Chapter 4

OptRISQL: Towards Performance Improvement of Time-Varying IoT Networks Using Q-Learning

IoT has the capability to facilitate the connection establishment and data exchange among a huge number of devices. These IoT networks can either be static or time-varying in nature, depending on the application scenario. Existing research on IoT mainly focuses on static devices like radio-frequency identification (RFID) sensors in buildings, traffic cameras on the road, security systems, and smart agriculture equipment. In addition to static devices, mobile devices have emerged as an essential part of IoT networks as mobile phones and automobiles are embedded with smart sensors [125]. The applications of IoT have now grown to help many real-time systems, such as intelligent transportation, smart mobile healthcare, and cognitive wearables.

To address the challenges of time-varying IoT networks, dynamic low power wide area networks (LPWAN) are being exhaustively investigated in the literature [73, 80, 65]. In dynamic LPWAN, due to changes in device position with time, a new set of wireless connections is established in the network towards data transfer at varying discrete

time instants. In most dynamic LPWANs, the data is directly transmitted to the gateway located far away from the source device. As a result, the network's energy-efficiency and data throughput performance suffer due to the drastic power consumption of individual devices [77]. To address this problem, multi-hop data transmission methods have been extensively researched in the literature to improve energy-efficiency and data throughput [126, 78, 76, 2]. Multi-hop data transmission over dynamic IoT networks is a critical task due to frequent changes in the network parameters. For instance, due to constant or transient changes in the locations of IoD, the wireless links in the network may either exist or not exist between selected IoDs in a random manner. Due to this phenomenon, it is difficult to design a single routing strategy for data transfer [71]. In addition, due to the unpredictable nature of link establishment, the IoDs are not able to identify an updated link between selected devices. Hence, the challenge here is to determine the relay IoDs to assist data transmission over a large network in which the device locations change over time. The selection of relay IoDs must be done in such a way that it results in the establishment of an energy-efficient and QoS-enhanced time-varying IoT network.

Towards this end, in this chapter, a novel method for the selection of relay IoD over a dynamic IoT network is proposed. The method utilizes a reinforcement learning technique called, Q-learning, which uses heuristic-based reward assignments for near-optimal relay IoD selection. The proposed Q-table originating from a reward matrix is built considering the inversely weighted averages of two distance heuristic functions. The method not only finds a multi-hop path in the network but also ensures that the selected relay node in the path ensures energy efficiency and improves performance. Rather than developing a fixed routing protocol for the entire dynamic IoT network, the proposed method updates the network's Q-table with the varying topology of the network. It ensures a near-optimal routing strategy customized to every changing network topology. The proposed method develops a minimum multi-hop path between

the IoD and the relay IoD. This enhances the residual energy of the IoD and minimizes data interference and data transmission delay. Due to this, data throughput is increased in an optimal way. This work presents a method that utilizes the current states of IoD to update the Q-matrix, with the aim of maximizing the cumulative reward value between selected device-gateway pairs. To achieve this, the proposed algorithm considers various states of an IoD, such as residual energy, data queue size, channel conditions, and transmission power levels. To achieve this goal, the main contributions of this chapter can be summarized as follows:

- In this chapter, a novel multi-hop data transmission framework is proposed over a time-varying IoT network using a heuristic-aided reinforcement learning technique, i.e., Q-learning technique. The proposed method of *Opt*imal *R*elay *I*oD *S*election using *Q*-*L*earning (OptRISQL), selects optimal IoD for relaying and minimum number of hops for data at each discrete time instant using network's topology snapshots.
- In order to enhance the energy-efficiency and QoS of the network, an optimization problem is formulated to maximize the accumulated reward at each IoD with respect to various network parameters such as overall residual energy, transmission delay, and bandwidth. In addition, a novel state-action model is provided for individual IoD that significantly increases the overall reward value between selected device-gateway pairs.
- By examining the most realistic scenarios for real-time applications of the proposed method, comprehensive measurement models are developed to compute the energy consumption of IoD, data latency between selected relay IoD and other IoD, data interference among IoDs, and throughput evaluation at the gateway.
- In addition to simulation results, the performance of the proposed method is also examined on various real-field datasets. The results obtained from the analysis of real-world datasets validate the practical applicability of the proposed method in

real-time scenarios, particularly in the context of large-scale IoT networks. The results obtained using the proposed method are also compared with well established existing methods such as direct transmission (DT) method [75], conventional routing (CR) method [72], learning-based multi-hop data routing (LMHR) method [79], and ring generator (RG) data routing method [35]. The obtained results demonstrate the significance of the proposed method over the existing methods.

The selected comparison methods represent both standard benchmarks and recent state-of-the-art solutions in energy-efficient and QoS-aware routing for IoT and LPWAN networks. DT [75] is a well-known geographic routing protocol optimized for energy savings with mobile sinks, while the conventional routing (CR) method [72] offers a deterministic energy-aware baseline. The learning-based multi-hop data routing (LMHR) method [79] and ring generator (RG) data routing method [35] address scalability and performance in constrained IoT environments, making them directly relevant to our objectives.

4.1 Network Model and Problem Formulation

In this section, the considered time-varying IoT network scenario is discussed first. Thereafter, the problem formulation for optimal relay IoD selection to achieve better energy efficiency and QoS over the developed network is presented.

4.1.1 Network Model

In this work, as shown in Figure 4.1, the time-varying IoT network is illustrated using a sequence of static graph snapshots, $\mathcal{G} = \mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_t, \ldots, \mathcal{G}_T$, where each $\mathcal{G}_t$ represents the network topology at time instant t. The vertical arrangement of snapshots indicates the progression of time, where each snapshot captures the network topology at a specific time instance. Changes across snapshots reflect link dynamics caused by node mobility

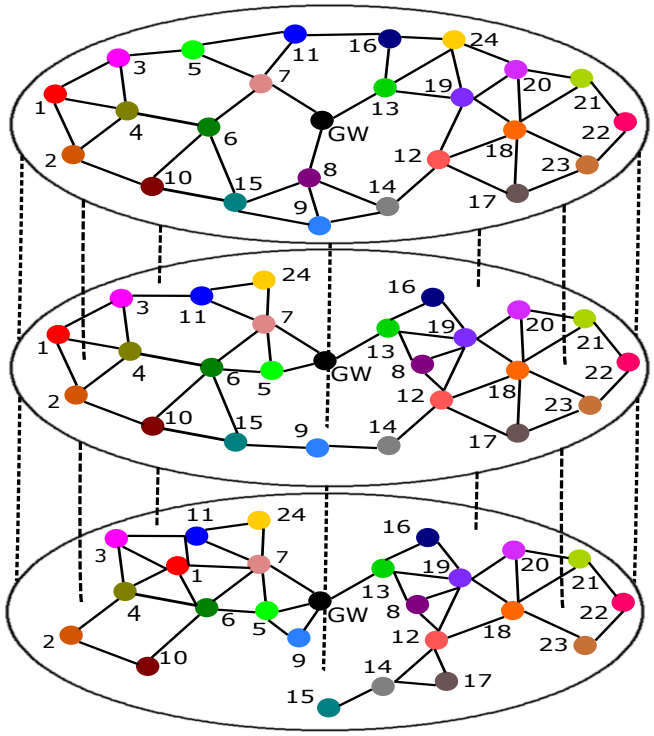


Figure 4.1: An illustration of the time-varying IoT network

or energy constraints, leading to the breaking of some connections and the formation of new ones. In Graph theory terms, these are unweighted, undirected graphs, where an edge denotes the presence of a direct communication link between two IoDs at that time. The gateway node (GW) remains static and acts as the central data sink, while the numbered nodes are simple identifiers for IoDs, representing connectivity without implying physical start or end roles. The locations of IoD follow Poisson's distribution with respect to the previous locations of the IoD within the network. Table 4.1 shows the terminologies used in this chapter. Here, T is the total network lifetime. The snapshot of the network at a time instant t is given as a two-dimensional graph $\mathcal{G}_t = (\mathcal{N}_t, \mathcal{M}_t, \mathcal{P}_t, \mathcal{E}_t^{res}, \mathcal{C}_t, \mathcal{B}_t)$. Where $\mathcal{N}_t$ is the set of time-varying IoDs, $\mathcal{M}_t$ represents a singleton set of the gateways, $\mathcal{P}_t = \left\{P_t^1, P_t^2, \ldots, P_t^n, \ldots, P_t^N\right\}$ denotes the set of transmission power capacities of different IoDs, $\mathcal{E}_t^{res} = \left\{E_t^1, E_t^2, \ldots, E_t^n, \ldots, E_t^N\right\}$ is the set of residual energies of the IoD, $\mathcal{C}_t$ is the set of channel coefficient between each IoD pair for data relaying, and $\mathcal{B}_t$ is the set of data queue sizes for each IoD in the network.

Table 4.1: Terminologies and definitions

Symbols/Parameters	Description
$\mathcal{N}_t, \mathcal{M}_t, \mathcal{P}_t$	Time-varying set of IoDs, singleton set of the gateway, set of transmission power capacities of different IoDs
$\mathcal{E}_t^{res}, \mathcal{C}_t, \mathcal{B}_t$	Residual energies of IoDs, channel coefficients between IoD pairs, data queue sizes for each IoD
$d_t^{n,m}$	Distance between IoD n and m at time t
$R_t^{n,m}$	Reward obtained by moving from n to m at time t
∂, r	Threshold energy value and transmission range of IoD
$B_{t,l}$	Bandwidth required for data transmission over link l
L, D_t^{tot}	Total number of available links, total data transmission delay
$D_{\text{tx}}^n, D_{\text{prop}}^n$	Transmission and propagation delays for IoD n
$D_{\text{proc}}^n, D_{\text{queue}}^n$	Processing and queuing delays for IoD n
$\mathbf{re}_t^n, \mathbf{qs}_t^n, \mathbf{cd}_t^n$	Vectors of residual energies, queue sizes, and channel properties for IoD n
$cs_t^n, pl_t^n, ls_t^n, br_t^n$	Channel status, power level, latency, and bandwidth matrices for IoD n
s^n, $\mathcal{E}_t$	Unique ID of IoD n and normalized energy parameters at time t
a_k	Action taken in state s^n at time t
$R(s^n, a)$	Reward for action a from state s^n to s^m
η_t^n	Sum of normalized energy parameters for IoD n
θ, $R_{N\times N}$, $Q_{N\times N}$	Data energy, reward matrix, Q-matrix
Q_{score}, $\zeta[\,]$	Accumulated reward over training, IoD array for efficient routing
$\pi(s^n)$	Optimal action taken in state s^n
α, γ	Learning rate and discount factor
E_{Tx}	Energy needed to transmit n-bit message over distance d

The IoD transfers the data packets to the *GW* through the relay IoDs in a multi-hop data transmission framework using heuristic-aided reinforcement machine learning (RML) technique at each instant of time t. These transferred packets are then forwarded by *GW* to the network server. Thereafter, data packets reach the application servers from the network servers via the TCP/IP protocol. For every snapshot graph, G_t at time t, every source IoD finds its set of optimal relay IoDs from $\mathcal{N}_t$ in its neighborhood and connects with them. Initially, two IoDs are connected if they are in the transmission range of each other. It is given as

$$d_t^{n,m} = \sqrt{(x_t^n - x_t^m)^2 + (y_t^n - y_t^m)^2}, \tag{4.1}$$

where $\{x_t^n, y_t^n\}$ and $\{x_t^m, y_t^m\}$ are the location of nth and mth IoD at tth time instant. Two IoDs n and m are connected, if $d_t^{n,m} \leq r$. Where r is the transmission range of IoD. The connections are created while keeping in mind the resource availability limit of the deployed network concerning real-time applications. Relay IoDs facilitate the multi-hop data transmissions to the *GW* from the source IoDs. The data transmissions between these IoD pairs exhaust power in proportion to the distance of transmission. The channel coefficient between any pair of IoD can be computed by any channel

estimation method, such as direction-of-arrival (DoA) estimation or received signal strength indicator (RSSI) measurement method [127]. Estimating the channel between the IoD pair is out of the scope of this work, and it is assumed that the channel is known. Furthermore, it is a common practice for each IoD to maintain a queue for received data packets, which can result in extra delays caused by congestion at the IoD. This work aims to design an algorithm that can optimize the performance of distributed data transmission by effectively utilizing the available information at each individual IoD.

4.1.2 Problem Formulation

The primary objective of this work is to optimally select relay IoDs for data transmission in the dynamic multi-hop transmission scenario to minimize data transmission delay, energy consumption, and number of hops. In order to achieve this, an RML-based algorithm is proposed. In the proposed RML framework, the agent performs an action according to the current state space, and consequently, a reward is generated. The reward is associated with each action taken by the agent in a way that the cumulative reward keeps adding up with each simultaneous action. Thus, the primary goal is to find the maximum cumulative reward for optimal selection of relay nodes. The reward function is defined as follows:

$$\mathcal{R}_t = \begin{cases} 0 & \textbf{if } m = n \\ R_t^{n,m} + 100 & \textbf{if } m = GW \\ R_t^{n,m} & Otherwise. \end{cases} \tag{4.2}$$

The expression of $R_t^{n,m}$ is given as:

$$R_t^{n,m} = [w \times d_t^{n,m} + (1 - w) \times d_t^{m,GW}]^{-1}, \tag{4.3}$$

where w is a hyper-parameter that determines the weight given to the distance between IoD n and m in a weighted average calculation. The reward expression is a function of the distances between the source IoD to relay IoD and the distance between the relay IoD to the gateway. The optimal weighting coefficient is obtained to ensure a higher reward for the shortest route. Hence, the main objective is to maximize the reward value. The problem of optimal relay IoD selection over the considered IoT network at time instant t is given as:

$$\begin{aligned}
&\max_{\forall n,m \in \mathcal{N}_t} \sum_{\substack{n,m=1 \\ n \neq m}}^{N} R_t^{n,m}, \\
\text{s.t.} \quad &\sum_{n=1}^{N} E_t^{n(\text{res})} \geq \partial, && ...(\text{a}) \\
&\sum_{l=1}^{L} B_{t,l} \leq B_{t,\max}, && ...(\text{b}) \\
&P^{GW}.\, p(t < \tau) \leq \sum_{n=1}^{N} P^n, && ...(\text{c}) \\
&D_t^{tot} \leq D_t^{conv}, && ...(\text{d}) \\
&t = 1, 2, \ldots, T, && ...(\text{e})
\end{aligned} \tag{4.4}$$

where the constraint (a) denotes that the total residual energy of all the n IoDs in the network at time instant t, i.e., $E_t^{n(res)}$ must be greater than a threshold energy value ∂. Constraint (b) guarantees that the bandwidth needed for data transmission $B_{t,l}$ over the existing links ($\forall l \in L$, where L represents the total number of available links) remains below the maximum bandwidth $B_{t,max}$ available to the network. Moreover, constraint (c) ensures that the cumulative data received by the gateway is either equal to or less than the total data generated by all IoD in the network. This in turn ensures that the total data delivered is less than or equal to the total data generated over the network. Constraint (d) denotes that the total data transmission delay (D_t^{tot}) which includes transmission delay, propagation delay, and queuing delay, observed over the network must be less than data latency associated with the conventional multi-hop data

routing method D_t^{conv}. The total data transmission delay is expressed as

$$\sum_{h=1}^{H}\sum_{n=1}^{N}\left(D_{\text{tx}}^{n}(h)+D_{\text{prop}}^{n}(h)+D_{\text{proc}}^{n}(h)+D_{\text{queue}}^{n}(h)\right), \tag{4.5}$$

where $h \in H$ is the number of hops. Finally, constraint (e) represents that the analysis is performed at discrete instants of time t.

4.2 Proposed Method

In this section, the proposed Q-learning method for optimal relay IoD selection in the multi-hop dynamic network is discussed first. Thereafter, the proposed algorithm to solve the formulated optimization problem is presented.

4.2.1 Q-Learning Based Multi-hop Data Routing Strategy

Q-learning is a model-free reinforcement learning technique. It does not require to know the probability distribution of actions taken at each IoD. Q-learning agents can perform well in a network setting where the transition distribution is often complex or unknown. Moreover, the key strength of the Q-learning method is its off-policy learning, enabling it to learn from older policies efficiently. Henceforth, due to the off-policy learning, the Q-learning agent's evaluation policy is always optimal, which guarantees optimal convergence. In the Q-learning approach for data routing, an agent selects the most suitable IoD to act as a relay for transmitting the data to the gateway at each time instant. The agent's choice of relay is influenced by the current state of the source IoD and the rewards obtained from the agent's actions. The proposed strategy conserves energy and ensures high QoS in the time-varying network. In this reinforcement learning problem, a tuple is defined as (S, SM, A, R) where S denotes state-space, SM denotes the state-mask mapping, A denotes the action-space, and R denotes the reward mapping at every instant t. A detailed description of the agent,

state, state-mask, actions, and rewards is given below.

4.2.1.1 Agent

An agent refers to the entity responsible for acquiring knowledge and making decisions. The things around this agent that it engages with are called its environment. A reinforcement learning (RL) agent uses deep learning techniques in combination with reinforcement learning algorithms to learn and make decisions in complex environments. Specifically, the RL agent utilizes the deep learning method to update the Q-matrix. RL agent has self-learning capabilities that focus on maximizing the long-term performance by interacting with environment and requires no awareness of system models.

4.2.1.2 States

In the proposed work, the best suitable relay IoD is determined by its state. At every discrete time instant t, the best course of action for data transfer is determined by the state and status of the device. The state of the nth IoD is denoted by s^n, and is given as $s_t^n = (\mathbf{re}_t^n, \mathbf{qs}_t^n, \mathbf{cd}_t^n)$, where $\mathbf{re}_t = \{re_t^n\}$ is a vector of residual energies, $\mathbf{qs}_t = \{qs_t^n\}$ is a vector of queue sizes for the IoDs, and $\mathbf{cd}_t$ indicates the channel properties for the IoDs and is denoted by $\mathbf{cd}_t = \{cd_t^n\}$, $\forall n$, where $cd_t^n = (cs_t^n, pl_t^n, ls_t^n, br_t^n)$. Here, cs_t^n, pl_t^n, ls_t^n, and br_t^n represent the matrices for the nth IoD's channel status, power transmission level, data latency, and bandwidth need at discrete time instant t, respectively. The state-space of the entire network is the collection of states of IoDs and is given as $S = (s^1, s^2, \ldots, s^n, \ldots, s^N)$. The state of each IoD serves as a unique ID for that IoD, i.e., s^n represents the nth IoD.

4.2.1.3 State-Mask

The energy-efficient multi-hop data transfer is determined by residual energy, queue size, channel conditions, power transmission levels, data latency, and bandwidth re-

quirements of the IoDs. The state-mask is a mapping function, M: $S \times \mathcal{E}_t \rightarrow S_t^*$, that determines which states from the state-space support data routing. Based on resource constraints, the function returns a subset of states that can be used for data transfer. $\mathcal{E}_t$ is the set of normalized energy parameters for each IoD at a discrete time instant t of the dynamic IoT network. Here, $S_t^* \subseteq S$ contains states satisfying energy constraints at time instant t. For every s^n and $\eta_t^n \in \mathcal{E}_t$

$$M(s^n, \eta_t^n) = \begin{cases} \phi, & \textbf{if } \eta_t^n < energy\ \ threshold \\ s^n, & \textbf{if } \eta_t^n \geq energy\ threshold, \end{cases} \tag{4.6}$$

where η_t^n is the sum of the normalized energy parameters of the nth IoD as described by $s_t^n = (\mathbf{re}_t^n, \mathbf{qs}_t^n, \mathbf{cd}_t^n)$ at time instant t.

4.2.1.4 Actions

The action space $A = (a_1, a_2, \ldots, a_k, \ldots, a_\lambda)$ is a vector space of all the possible actions from every state in the state-space. The states and time instant t have no impact on actions. As a result, any action executed by the model at any state s^n and at any time instant t is denoted as $a_k = (e_k)$, $\forall k \in \{1, 2, \ldots, |\lambda|\}$, where e_k signifies the routing choice. The metrics such as power transmission levels, bandwidth requirement, data latency, and interference between IoD-pairs are used to choose an action a_k for data transfer.

4.2.1.5 Rewards

A reward function assigns reward values corresponding to actions taken from the states with at least a threshold energy at a given time instant t. The reward function maps state-action pairs to a real value, R: $S_t^* \times A \rightarrow \mathbb{R}$. Based on the network resource constraints and the ease of data transmission at the transitioned state, a reward is assigned. Subsequently, for any action a taken from state s^n to s^m present in S_t^*, the

reward is given as

$$R(s^n, a) = F(s^n, s^m, \eta_t^n), \tag{4.7}$$

where $F(\cdot)$ is a function that calculates a real value as a weighted combination of state-value and IoD energy parameters. As a result, only activities that adhere to network energy limitations and improve data transmission capabilities are reinforced for the model to learn. The resulting route constitutes of optimal relay IoDs for data transmission at time instant t and is represented by a policy π, which maps π: $S_t^* \rightarrow \mathcal{A}$ with $\pi(s^n)$ being the optimal action taken in state s^n given the energy constraints. An optimal policy refers to a policy that maximizes the expected discounted reward and is given as:

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\ E\left[\sum_{n=0}^{N} \gamma R\left(s^n, \pi\left(s^n\right)\right)\right], \tag{4.8}$$

where γ is the discount factor. If $\gamma = 0$, the optimization problem becomes focused on maximizing the immediate reward at the current state. On the other hand, if γ is close to 1, the optimization problem takes future rewards into stronger consideration. In general, γ varies between 0 to 1.

4.2.2 Q-Table Update

The optimal relay routes are stored in a table called Q-table. The entries of this table are called the Q-values. A Q-matrix Q_t is computed for each snapshot of the dynamic network topology G_t at time instant t. This matrix is updated at each epoch to depict the data routes and selected relay IoDs. The Q-values for state-action pairs in these Q-matrices are updated as follows

$$\begin{aligned} Q_{t,(\text{new})}\left(s^n, a\right) \leftarrow (1-\alpha) Q_{t,(\text{old})}\left(s^n, a\right) + \alpha\{R_t\left(s^n, a\right) \\ +\gamma \max_{a_t} Q_{t,(\text{old})}\left(s^m, a\right)\}, \end{aligned} \tag{4.9}$$

where, α is the learning rate ($0 \leq \alpha \leq 1$). Learning rate controls the number of states where immediate future rewards are considered for the Q-value updation. In addition, γ is the discount rate, which adjusts the fraction of the maximum possible future cumulative reward that is taken into consideration. $Q_{t,(old)}(s^n, a)$ is the old Q-value for the state-action pair (s^n, a) in the network at time instant t, and $Q_{t,(new)}(s^n, a)$ is the new Q-value. In (4.9), the term $\{\max_{a_t} Q_{t,(old)}(s^m, a)\}$ is the maximum possible cumulative reward value that can be attained from state s^m, where $R_t(s^n, a)$ is the immediate reward received when the data is transmitted from state s^n to state s^m in the snapshot G_t. The core of the algorithm in the preceding equation uses the weighted average of the old and new Q values in an iterative update.

Algorithm 4.1 OptRISQL Algorithm (Optimum Relay IoD Selection using Q-Learning)

INPUT: A series of snapshots of static graphs at different time instants, $\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_t, \ldots, \mathcal{G}_T\}$. Where $\mathcal{G}_t = (\mathcal{N}_t, \mathcal{M}_t, \mathcal{P}_t, \mathcal{E}_t, \mathcal{C}_t, \mathcal{B}_t)$.

OUTPUT: A selection of paths of optimal relay IoDs for each IoD-gateway pair (π^*)

INITIALIZATION

1. Network of size $L \times W$ m^2 consisting of N IoDs, transmitting the data to M gateways.
2. Initialize the energy levels of IoD's with equal energy E.
3. Initiate θ: $[s^n = E^n]$ $\forall$ n
4. **for** t = 1,2,...,T **do** :
5. Create $\mathcal{R}_{N\times N}$ whose cells $R^{n,m}$ represent the *reward* given to the agent for moving from the *IoD* s^n to s^m.
6. **for all** transitions from s^n to s^m, set $\mathcal{R}_{N\times N}$ as:
7. **if** $s^m == s^n$
8. $R^{n,m} = 0$.
9. **else if** $s^m == GW$
10. $R^{n,m} = R^{n,m} + 100$.
11. **else**
12. Compute $R^{n,m}$ using (4.3).
13. Create $Q_{N\times N}$ whose cells $Q_{n,m}$ represent the accumulation of the *rewards* over

training (Q_{score}) for moving from IoD s^n to s^m and **do :**

14. **if** IoD n and m are neighbour

15. initialize $Q_{n,m} = 0$.

16. **else if** IoD n and m are non-neighbour

17. initialize $Q_{n,m}$ = -∞.

18. **for** each training *epoch* **do :**

19. Choose an IoD, and its *neighbor* with maximum Q_{score}.

20. Update Q_{score} using equation (4.9).

21. **while** (IoD != GW) **do :**

22. Initialize ζ [ ] array and add IoD

23. **if** $\theta > 0$

24. From $Q_{N\times N}$, select neighbor node of the IoD with the highest Q_{score} and store in the $chosen\_IoD$ and add to ζ [ ]

25. Initialize d with distance between IoD and $chosen\_IoD$.

26. **if** $d < d_{th}$ **then do :**

27. $E_{Tx} = n\varepsilon_{\text{elec}} + n\epsilon_{fs}d^2$

28. subtract E_{Tx} and update θ of $chosen\_IoD$.

29. **else :**

30. $E_{Tx} = n\varepsilon_{\text{elec}} + n\epsilon_{mp}d^4$

31. subtract E_{Tx} and update θ of $chosen\_IoD$.

32. Compute data latency, data throughput, and interference using the measurement models given in Section 4.3.

33. **else if** $\theta = 0$

34. From $Q_{N\times N}$ select neighbor node of the IoD with the next highest Q_{score} having $energy > 0$ and store in $chosen\_IoD$ and add to ζ[ ]

35. Compute E_{Tx} using steps 27 to 34.

36. **if** $chosen\_IoD = GW$

37. Compute E_{Tx} using steps 27 to 30.

38. **return** ζ[ ], **else**

40. *device* is assigned value of *chosen_IoD*.
41. **return** *optimal_path* (π^*)
42. **end**
43. **end**
44. **end**

4.2.3 Proposed Algorithm

The proposed algorithm starts with the initialization of every Q-matrix to an arbitrary fixed value. This matrix is trained individually by an agent for optimal data transmission and routing. In the proposed algorithm, during the training process, the learning algorithm first chooses its action based on the current state s^n, and then moves to the next state s^m. This transition receives the immediate and future rewards for every data transmission taken in the snapshot at tth time instant as defined by the reward matrix. An episode of the algorithm concludes when state s^m in the network snapshot G_t is a final or terminal state. *Algorithm 4.1* enumerates the process of selecting optimal relay IoD in a dynamic IoT network. First, at every time instant t, network snapshot G_t is considered, denoting the set of randomly deployed IoDs $\mathcal{N}_t$ with the fixed set of gateways in the network at that time instant. The power transmission levels, residual energy, queue size, and other channel conditions are assumed to be known at individual IoD across all the snapshots. Subsequently, the algorithm computes a series of Q-matrices, each associated with one snapshot G_t in an online learning mode. The initial values of every state-action pair in these Q-matrices are assigned according to distance metrics between the IoD pairs. The RL agent updates the Q-matrix for every state-action pair taken in the snapshot G_t. Subsequently, the learning method is applied to these snapshots to calculate the optimal path between selected IoD-gateway pairs. In the online learning method, the Q-matrix is used as a policy to decide which IoDs are to select for energy-efficient data transmissions at time instant t in the dynamic network.

The proposed online learning approach involves selecting the relay IoD with the highest Q-matrix values for data transfer in each snapshot. By following this strategy, the algorithm aims to determine an optimal path that can maximize the cumulative reward value between the IoD-gateway pairs at every time instant in the dynamic network. In other words, the method seeks to identify the most efficient route for data transmission that can lead to the highest possible collective reward for the system. In summary, the algorithm integrates Q-learning with dynamic Q matrices to adaptively optimize routing in a network of IoDs. It considers channel conditions, energy efficiency, and distance metrics, providing a comprehensive approach to routing strategy in dynamically changing network scenarios.

4.3 Measurement Models

This section discusses various measurement models that are used for evaluating the performance of the proposed method. The measurement model includes data latency, energy consumption, data throughput, and data interference measurement models.

4.3.1 Data Latency Measurement Model

The data latency measurement model is composed of the following delays in the network.

4.3.1.1 Packetization Delay

The delay that occurs during the process of breaking down a data stream into smaller packets for transmission across a network is known as packetization delay. Typically, larger packets will result in a longer packetization delay, as segmenting and encapsulating the data requires more time. The packetization delay can be expressed as

$$d_{\text{packet}} = \frac{\text{Packet size (bits)}}{\text{Transmission rate (kbps)}} = \frac{P}{R} \tag{4.10}$$

4.3.1.2 Propagation Delay

The time duration needed for data transfer between two IoD over a direct link is referred as propagation delay. This delay depends on the distance between the IoD pairs. Consequently, the propagation delay is expressed as

$$d_{\text{prop}} = \frac{\text{Node separation (m)}}{\text{Transmission speed (m/s)}} = \frac{\text{D}}{\text{S}} \tag{4.11}$$

For RF-based IoT communications, the transmission speed is the speed of light in free space. Therefore, in terrestrial IoT networks with typical communication ranges of a few hundred meters, the propagation delay is on the order of microseconds and is negligible compared to other delay components such as processing delay and queuing delay.

4.3.1.3 Processing Delay

The amount of time required by an IoD to process data packets is known as the processing delay. Bit-error detection and next-hop selection are part of the processing. Exponential random distribution is used to calculate the processing delay and is given by

$$d_{\text{process}}(x, \lambda) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ \\ 0, & x < 0, \end{cases} \tag{4.12}$$

where, $\lambda > 0$ is called the distribution rate parameter.

4.3.1.4 Queuing Delay

After being processed by an IoD, the delay experienced by data packets while waiting to be transferred via a network link is called queuing delay. If the network link is transmitting another data packet at the moment, the arriving data packet will suffer

queuing delay. It can be represented using the Poisson distribution and is given by

$$d_{\text{queue}} = P(k \text{ events}) = \frac{\mu^k e^{-\mu}}{k!}, \tag{4.13}$$

where μ represents the mean of the Poisson distribution under consideration. Consequently, the complete delay encountered by an IoD during the transmission of data can be calculated as:

$$d_{(\text{total})} = d_{\text{packet}} + d_{\text{prop}} + d_{\text{process}} + d_{\text{queue}}. \tag{4.14}$$

4.3.2 Energy Consumption Measurement Model

The IoDs dissipate their energy in sensing, processing, transmitting, and receiving operations. However, data transmission is the primary reason for energy consumption. The energy consumption model is used to calculate the amount of energy dissipated. Transmitting IoDs consume energy in the radio electronics and amplification. Whereas, receiving IoDs consume energy only in radio electronics. The number of data packets to be transmitted and the distance between the transmitter and receiver play a major role in calculating the energy dissipation. In this energy consumption model, free space (d^2 power loss) and multi-path fading (d^4 power loss) channel models are also taken into account. If the distance between the transmitter IoD and receiver IoD is less than a threshold distance d_{th}, then the free space (fs) model is used; else multi-path (mp) channel fading model is used. The energy required by transmitter to transmit n-bit message to receiver over a distance d, is given by [62]

$$\begin{aligned} E_{\text{Tx}}(n,d) &= \varepsilon_{\text{Tx-elec}}(n) + \varepsilon_{\text{Tx-amp}}(n,d) \\ &= \begin{cases} n\,\varepsilon_{\text{elec}} + n\epsilon_{fs} d^2, & d < d_{th} \\ n\,\varepsilon_{\text{elec}} + n\epsilon_{mp} d^4, & d \geq d_{th}. \end{cases} \end{aligned} \tag{4.15}$$

Moreover, the energy required by the receiver IoD to receive n-bit message is given by

$$E_{\mathrm{Rx}}(n) = n \times \varepsilon_{elec}, \tag{4.16}$$

where ε_{elec} depends on the network operations such as modulation, coding, and filtering. Amplifier energy $\epsilon_{fs}d^n$ depends on the distance d.

4.3.3 Data Throughput Measurement Model

Data throughput is the overall count of data packets transmitted to the gateway. Thus, the data throughput at each iteration of data transmission is equal to the total number of data packets received across the gateways. Following that, data throughput across the network is given by

$$\text{Data throughput} = \sum_{i=1}^{I} \sum_{m=1}^{M} P_m^{(i)}, \tag{4.17}$$

where, $P_m^{(i)}$ represents the count of packets received by the mth gateway at the ith iteration of data transmission.

4.3.4 Interference Measurement Model

Interference is the undesirable effect on the received data at an IoD due to external factors. Excessive data flow through IoD leads to more interference, which causes data loss and delayed data transmission. A large number of IoDs per gateway leads to more data collisions, which results in a high level of interference and affects the device's throughput. The measurement of interference in percentage at nth IoD is given by

$$I_n = \frac{\eta_n}{\sum_{m=1, m\neq n}^{N} \eta_m} \times 100, \tag{4.18}$$

where η_n is the number of transmissions made by the nth IoD in its lifetime. The interference level at the n^{th} IoD, denoted by I_n, is quantified as a percentage. This equation represents the interference in terms of the transmission activity of the n^{th} node relative

to the combined activity of all other nodes in the network. By excluding the n^{th} IoD from the denominator, the metric captures how heavily the node transmits in comparison to its peers. Multiplying the ratio by 100 expresses the interference as a percentage. A higher value of I_n indicates that the node is either contributing more interference to the network or is more prone to experiencing interference due to increased channel access. This model is particularly relevant in shared-channel IoT networks, where excessive transmissions lead to packet collisions, congestion, and degraded performance.

4.4 Performance Evaluation

The effectiveness of the proposed method is evaluated by examining both simulated and real-field datasets over a time-varying IoT network. To measure the efficacy of the optimal relay IoD selection method, the study examined the number of functioning and non-functioning IoDs over a dynamic network that evolves over time, resulting in distinct energy-efficient routes for each network snapshot. All of the snapshots in which a particular percentage of live IoDs could transmit data determine the lifespan of the dynamic network. Furthermore, the optimality of the route is evaluated in terms of the energy efficiency of the relay IoDs. This is calculated by considering the residual energies of the IoDs in time-varying IoT network snapshots. Further, the QoS of the proposed method is evaluated by measuring average data latency, successful packet transmission, network capacity, and packet collision rates for each IoD. Lastly, to show that the proposed method is the best way to choose relay IoDs, it is compared with existing methods like the direct transmission (DT) method, conventional routing (CR) protocol, learning-based multi-hop routing (LMHR) method, and ring generator (RG) protocol. Through this comparison, the effectiveness of the proposed method is presented.

4.4.1 Experimental Setup

The proposed method is evaluated over both the simulated and real-field IoT testbeds. The descriptions are given below.

4.4.1.1 Simulated IoT Testbed

The network size used for simulation is 4 km × 4 km. The network is populated with 400 dynamic IoDs that are evenly distributed across the region using a Poisson Point Process (PPP). The network also has a single IoT gateway installed at fixed coordinates (2 km, 2 km). After every bunch of data transmission, the topology of the network changes due to the movement of the IoDs. The random movement of the IoDs is confined within a circular region with the previous IoD location as the center. While it is generally assumed that IoDs can directly send data packets to the central hub, they also have the capability to convey information to all nearby IoDs within their signal radius. In every instance of data transmission, IoDs communicate with the central hub, either through a direct connection or via a multi-step approach. Each IoD has a 4000 mAh, 5 V battery and possesses 72,000 Joules (J) of energy. Data packets utilized for transmission are 400 bits in size. The LoRa communication framework utilizes specific parameters, such as power output, spreading factor, bandwidth, and coding rate. The values for these parameters are 14 dBm, 7, 125 kHz, and 4/5, respectively.

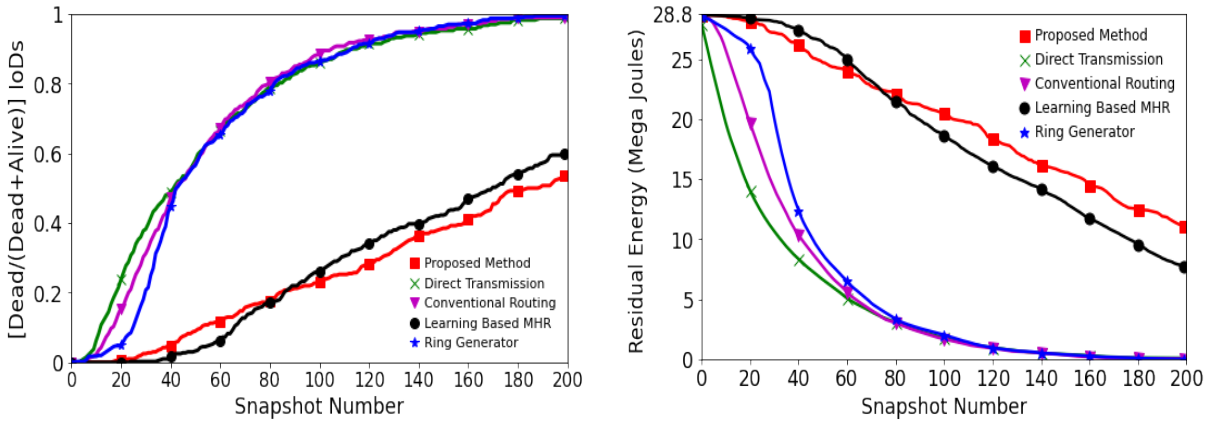
4.4.1.2 Real-Field IoT Testbed

A real-field IoT testbed is also used to conduct experiments. This real-field dataset is generated with the help of [128]. The experimental setup involves deploying 400 IoDs with a single gateway over an area of 4 km × 4 km. The initial placement of IoDs follows the measurement model of [128], but later the devices are allowed to move randomly within their radio range. The gateway remains stationary at (2 km, 2 km). Each IoD has an energy capacity of 72000 J with a 5V, 4000 mAh battery. Data transmission

utilizes the LoRa model with a 14 dBm transmit power and spreading factor of 7, with a bandwidth of 125 kHz and a coding rate of 4/5. The received power is measured to be -82 dBm for an IoD located 160 meters away from the source device, while it decreased to -123 dBm when the separation distance between devices was increased to 520 meters.

4.4.2 Energy-Efficiency Analysis

It is crucial to extend the lifespan of IoDs that have limited power. The objective of this section is to evaluate network longevity based on the count of transmitted snapshots to gateways. The assessment method utilizes the energy consumption metric model elaborated in **Section 4.3.2**. The results are obtained through the implementation of various data transmission methods, including RG method (represented in blue), the proposed method (represented in red), LMHR method (represented in black), DT method (represented in green), and CR method (represented in magenta) on a simulated IoT testbed shown in Figure 4.2. The analysis showcases the results obtained from both simulated and real-field datasets in Figure 4.2 and Table 4.2, respectively.



(a) (b)

Figure 4.2: An illustration of the (a) variation in the ratio of dead IoDs to total IoDs used for data transmission in the network with each snapshot taken at regular intervals of the mobile network's lifetime and (b) Residual energy with respect to the number of snapshots in a network

4.4.2.1 Network Lifetime Performance

The number of functional and non-functional IoDs in the network indicates network longevity. The network's longevity is directly related to the number of operational IoDs and inversely related to the number of inoperative IoDs present in the network.

As a result, the network's longevity is depicted by the counts of active and inactive IoDs throughout multiple snapshots. From Figure 4.2a, it is observed that, for RG, DT, and CR methods, all (100%) IoDs become non-functional after 200 snapshots. However, for the LMHR method, 60% IoDs are inoperative at 200 snapshots. In contrast, using the proposed approach, only 53% IoDs are inoperative under similar circumstances. Table 4.2 shows the results derived from the real-field dataset. It shows the number of dead and alive IoDs with respect to time. As per Table 4.2, the proposed method lowers the energy consumption for gathering and disseminating data packets. It is noted that 184 (46%) IoDs become inoperative during data transmission at $200th$ snapshot. In comparison, under the same conditions, the corresponding values for the DT, CR, LMHR, and RG methods are 393 (98.25%), 396 (99%), 268 (67%), and 379 (94.75%), respectively. Thus, the proposed method enhances energy efficiency by increasing the number of operational IoDs. The proposed method decreases the count of inoperative IoDs in the network, resulting in an extended network lifespan.

4.4.2.2 Residual Energy

The residual energy of a network snapshot is calculated by adding up the remaining energy of each IoD after complete data transmissions at that particular network snapshot. Residual energy represents the amount of remaining energy in the network. The IoDs are originally energized by a 5V, 4000mAh battery. The energy measurement model described in **Section 4.3.2** defines the energy that every transmission consumes.

Table 4.2: Variations in the number of dead IoT devices, alive IoT devices, residual energy, bandwidth utilization, data latency, and data throughput with varying number of iterations used for data transmission over a real field IoT testbed

	Snapshot number chosen for data	**Energy-Efficiency and Quality of Service Measurements over Real-field IoT Network Deployment**				
Data Tx. Model Used	Times the Data is Transferred	Dead Nodes (Numbers)	Alive Nodes (Numbers)	Residual Energy (Mega Joules)	Latency (Seconds)	Number of received Packets
Proposed Method	50	38	362	24.64	0.62	358
	100	84	316	20.47	0.33	315
	150	136	264	16.83	0.33	262
	200	184	216	12.96	0.63	216
Ring Generator Method (RG)	50	50	350	24.01	0.73	28
	100	108	292	20.60	0.29	13
	150	317	83	3.77	0.47	88
	200	379	21	0.49	0.36	20
Learning Based MHR Method (LMHR)	50	42	358	24.12	1.99	355
	100	139	261	16.24	3.19	260
	150	216	184	10.41	4.67	184
	200	268	132	6.38	0.14	132
Direct Transmission Method (DT)	50	211	189	7.44	0.48	191
	100	318	82	2.15	0.26	81
	150	375	25	0.35	0.24	24
	200	393	7	0.11	0.29	6
Conventional Routing Method (CR)	50	240	160	6.79	3.13	162
	100	364	36	1.29	0.89	36
	150	388	12	0.27	4.59	11
	200	396	4	0.07	7.23	3

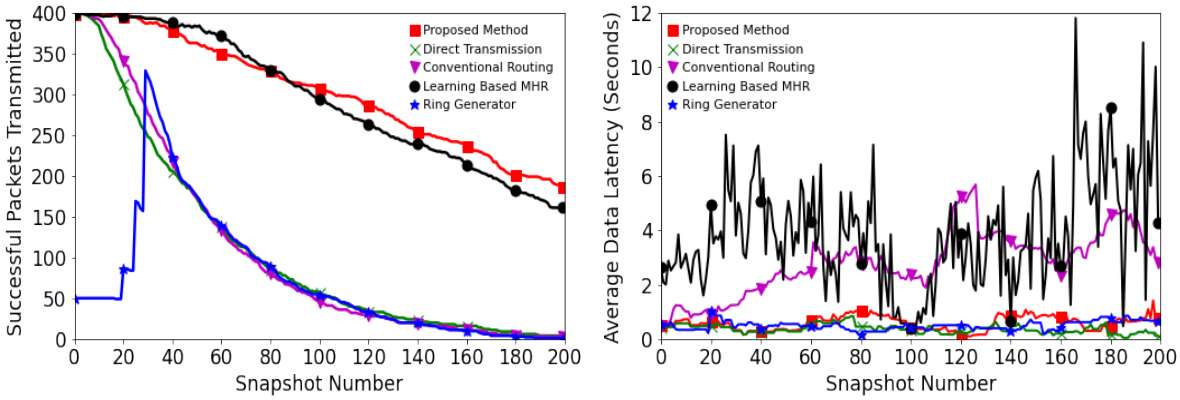
Figure 4.2b shows how the residual energy changes in a simulated IoT network environment, and Table 4.2 shows the residual energy in a real-field IoT testbed. In Figure 4.2b, after 150 iterations of data transmission, the residual energy in the network decreases to 0.2 MJ for CR method, 3.7 MJ for RG method, and 0.3 MJ for DT method. The residual energy for the proposed method and LMHR method is 58.43% and 36.1% of the initial energy, respectively, after 150 iterations. As shown in Figure 4.2b, after 200 iterations of data transmission, the residual energy in the network decreases to 0.07 MJ for CR method, 0.5 MJ for RG method, and 0.1 MJ for DT method. The residual energy for the proposed method and LMHR methods is 45% and 22.15% of the initial energy, respectively, after 200 iterations. The residual energy in a real-field IoT dataset, after 200 snapshots, is 12.9 MJ for proposed method, 6.3 MJ for LMHR method, 0.01 MJ for DT method, 0.07 MJ for CR method, and 0.5 MJ for RG method. This suggests that the proposed method can transfer data in a more energy-efficient way compared to other existing methods.

4.4.3 Quality of Service (QoS)

This section provides an evaluation of the network's QoS. To determine the proposed method's effectiveness, three measures are used, which are data throughput, average data latency, and data interference that are described in **Section 4.3.1, 4.3.3, and 4.3.4, respectively**. The analysis showcases the results obtained from both simulated and real-field datasets in Figure 4.3 and Table 4.2, respectively.

4.4.3.1 Data Throughput

Data throughput is defined as the overall quantity of data packets that have been successfully sent to the gateway. The analysis of data throughput in a simulated IoT network is presented in Figure 4.3a. While Table 4.2 provides the analysis of data throughput in the real-field dataset. Figure 4.3a suggests that the proposed method is more efficient than other methods like DT, CR, LMHR, and RG in transmitting



(a) (b)

Figure 4.3: An illustration of (a) data throughput and (b) average data latency with respect to the number of snapshots in a network with prevalent data transmission

data packets to the gateway. As it is clear from Figure 4.3a, in the simulated IoT testbed, the proposed method transferred 170 data packets to the gateway by the end of 200 network snapshots. On the other hand, the DT, CR, LMHR, and RG methods transferred fewer than 150 data packets successfully to the gateway in the simulated IoT testbed. Whereas in the real-field IoT testbed, this number is more than three million. The LMHR method transferred 130 data packets, which is only 76.5% of the number of successful transmissions made with the proposed method. The DT, CR, and RG methods transferred even fewer packets, with less than 10 successful transmissions, which is 94% less than the proposed method. In the case of the DT method, there are only 5 successful transmissions, which is approximately 2.94% of the proposed method.

Table 4.2 shows the data throughput performance of the proposed method in real-world scenarios. From Table 4.2, it is evident that the proposed approach outperforms other existing transmission methods such as CR, DT, LMHR, and RG. As shown in the table, after 200 snapshots, the proposed method transferred 216 data packets successfully to the gateway. In comparison, CR, DT, LMHR, and RG only transferred 1.4%, 2.8%, 61%, and 9% of the proposed method's success rate, respectively. These results signify an enhancement in data throughput achieved by the proposed method in contrast with other existing methods.

4.4.3.2 Data Latency

Using the measurement model outlined in **Section 4.3.1**, the average transmission delay for each snapshot is calculated. Figure 4.3b displays the average data latency for all transmissions within a simulated IoT testbed. In contrast to methods like LMHR and CR, the proposed method demonstrates average data latency, which is the lowest of all. In the proposed method, when tested in a simulated IoT environment, the average delay in transmitting data for each snapshot was recorded at just 1.1 seconds. On the other hand, the data latency for traditional LMHR and CR data transmission algorithms exceeds 1.1 seconds. Whereas, because of direct communication between the source IoD and the gateway, RG and DT methods demonstrate low data latency. As shown in Figure 4.3b, at the $160th$ snapshot, the RG method has a data latency of 0.4 seconds, which is 50% of the proposed method. However, at the $160th$ snapshot, the network has 95% fewer devices compared to the proposed method, resulting in an early network deterioration stage. Other models, such as CR and LMHR, have data latencies of 2.76 and 4.88 seconds, respectively. While under comparable conditions, the proposed method displays a data latency of 0.91 seconds.

Table 4.2 shows the data latency obtained from the real-field dataset. The use of the CR transmission method resulted in a latency of 7.2308 seconds, while the proposed method has a much lower latency of 0.6322 seconds. The DT, RG, and LMHR methods exhibit the lowest network latency of 0.2999 seconds, 0.3642 seconds, and 0.1371 seconds at the $200th$ snapshots. These values, obtained by adjusting the initial energy for various data transmission models, are also listed in Table 4.2, highlighting the advantage of the proposed method in terms of improved data latency.

4.4.3.3 Data Interference

The impact of data interference on the network is analyzed by calculating the percentage of time a node is occupied in data transmission relative to the total number of transmis-

sions made by all active nodes in the network. The data interference is computed using the measurement model described in **Section 4.3.4**. Figures 4.4a and 4.4b display the data interference analysis in both simulated and real-field IoT testbeds. The analysis reveals that the proposed method significantly reduces the level of data interference encountered by IoDs as well as the quantity of IoDs experiencing high interference by minimizing the repetition of node usage. Figure 4.4a shows that 45.5%, 14.75%, and 3% of IoDs using the LMHR method, CR method, and RG method, respectively, have higher interference compared to the proposed method in the simulated IoT testbed. Similarly, as shown in Figure 4.4b, in the real-field dataset, 42%, 14.75%, and 3.5% of nodes using the LMHR method, CR method, and RG method, respectively, have higher interference compared to the proposed method.

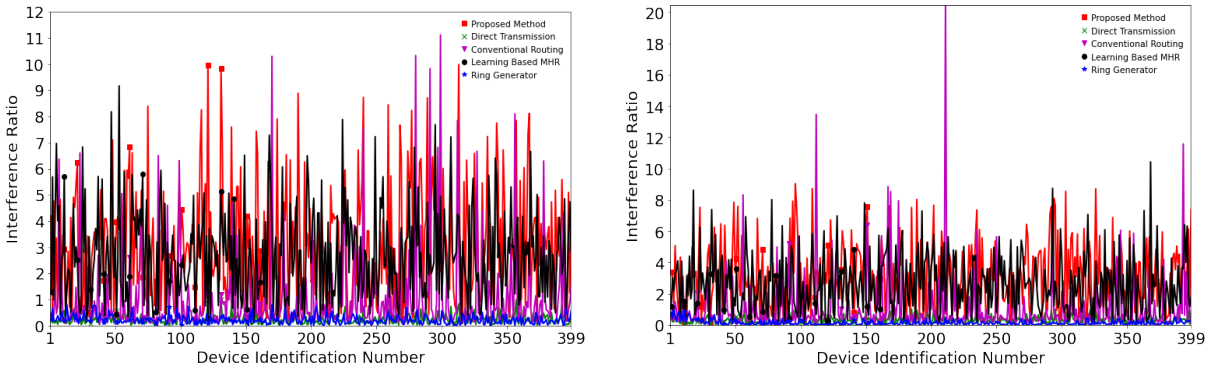


Figure 4.4: The analysis of data interference performance in both (a) simulated testbed and (b) real-field IoT testbed

4.5 Conclusions

The main task of this chapter is to create an innovative approach for data routing in dynamic IoT networks. The proposed method employs a Q-learning framework, which belongs to the category of reinforcement learning algorithms. The method enhances the data transmission policy, leading to improved network performance, including enhanced energy efficiency and quality of service (QoS), through optimal relay IoD selection.

QoS is assessed by examining data transmission delay, data throughput, and interference experienced by individual nodes. The effectiveness of the proposed approach is evaluated by comparing it with various existing techniques such as a conventional data routing framework, a direct transmission protocol, ring generator technique, and a learning based multi-hop routing method. The Performance is assessed in both simulated and real-field IoT testbed environments. The Obtained results demonstrate that the proposed method significantly improves energy-efficiency and QoS when compared to various existing methods, exhibiting strong robustness and adaptability in dynamic network environments.