Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Decentralized Multi-Metric Q-Learning Objective Function (MM-QLOF) for Efficient Routing in IoT Networks

Youness TALBI

youness_talbi@um5.ac.ma

Mohammed V University

Abdelhadi ELOUDRHIRI HASSANI

Mohammed V University

Adil SALBI

Mohammed V University

Issam BOUGANSSA

Mohammed V University

Research Article

Keywords: IoT, Routing, RPL, LLN, Q-learning, 6LOWPAN

Posted Date: June 29th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8996461/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Decentralized Multi-Metric Q-Learning Objective Function (MM-QLOF) for Efficient Routing in IoT Networks

Youness TALBI[1*], Abdelhadi ELOUDRHIRI HASSANI[1], Adil SALBI[1], Issam BOUGANSSA[1]

[1*]LASTIMI Laboratory, CeDOC Mohammadia School of Engineers (EMI), Mohammed V University in Rabat, Morocco.

*Corresponding author(s). E-mail(s): youness_talbi@um5.ac.ma;

Abstract

Under high-traffic conditions, IoT networks suffer from significant congestion and link quality degradation, making optimal parent node selection in RPL-based routing extremely difficult and challenging. This paper address this issue by proposing a decentralized, multi-metric Q-reinforcement learning objective function, named MM-QLOF, which incorporates packet success rate (PSR), expected transmissions (ETX), and congestion level into the preferred parent node selection process. MM-QLOF allows each node to autonomously learn optimal routing decisions through a reward-penalty mechanism based on observed network conditions. By continuously evaluating candidate parent nodes using the three metrics, the nodes update their Q values and select the parent node that maximizes the long-term reward, resulting in adaptive and congestion-sensitive routing behavior. The proposed objective function was evaluated under Cooja emulator. Simulation results prove that MM-QLOF significantly improves network performance. Compared to OF0, it improves packet delivery rate (PDR) by **10.7%**, reduces end-to-end latency by **18.7%**, and slightly decreases power consumption. Compared to MRHOF, MM-QLOF achieves an **11.1%** higher PDR, a **23.6%** reduction in latency, and increased energy efficiency.

Keywords: IoT, Routing, RPL, LLN, Q-learning, 6LOWPAN

1 Introduction

Internet of Things (IoT) has become an essential technology in various fields, including healthcare, industrial automation, smart infrastructure, and defense systems. IoT environments are typically composed of resource-constrained devices (limited processing and memory capacity, as well as limited power resources)[1]. In these networks, communication operations, particularly packet transmission and reception, constitute the primary source of energy consumption. Consequently, ensuring reliable, low-latency, and energy-efficient communication between nodes is a major challenge[2–4]. Routing protocols play a central role in ensuring efficient packet transmission while minimizing packet loss and delays in low-power and lossy networks. Recognizing these constraints, the Internet Engineering Task Force (IETF) introduced the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) in 2012, specifically designed to support resource-constrained IoT deployments.[5, 6].

1.1 RPL overview

RPL protocol has been designed to enable routing in wireless sensor networks. The fact that it is based on IPv6 makes it compatible with the IEEE 802.15.4 standard, enabling low-power devices to communicate efficiently[7, 8]. In RPL, avoiding routing loops is an obligation. To achieve this, RPL constructs a directed acyclic graph (DAG), where each DAG is structured around a root called DODAG (Destination Oriented DAG). An objective function is defined to guide the construction of the DODAG and select optimal routes based on precise metrics. RPL relies on three ICMPv6 control messages to form and maintain paths between nodes. The DIO (DODAG Information Object) message contains important topological information such as the current rank of a node, the root IPv6 address, and RPL instance parameters. The DIS (DODAG Information Solicitation) message is sent by a node when it needs to receive a DIO from a neighbor. The DAO (Destination Advertisement Object) message is used to manage traffic in the opposite direction, from the root to the nodes[9–13].

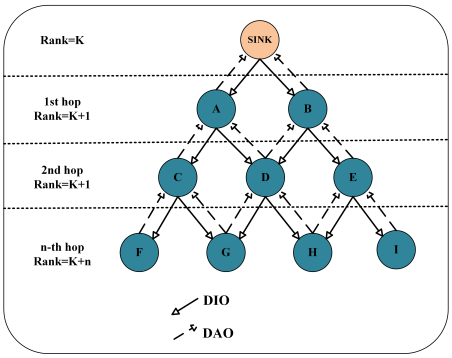


Fig. 1: Process of constructing a DODAG with RPL.

As shown in Fig. 1, the root node periodically sends DIO messages containing parameters such as rank, which indicates the distance of the node from the root; metric, which measures the quality of the link; routing cost, which represents the cost of the route to the root; and DODAGID, which uniquely identifies the DODAG[14–16]. Neighboring nodes evaluate this information using the objective function to decide whether to join the DODAG.When we talk about routing, we are referring to how a node selects its preferred parent (or next hop) to reach its destination. In general, the lower the cost of a path, the more optimal the routing decision. In this context, two widely used objective functions are the zero objective function (OF0) and the minimum rank objective function with hysteresis (MRHOF). OF0 is based on the minimum number of hops; in other words, it uses the number of hops to calculate the rank, so that the lower the number of hops, the more optimal the selected path. MRHOF, on the other hand, is based on link quality and the expected transmission count (ETX) metric, where a lower ETX indicates a more reliable and optimal path [17–19].

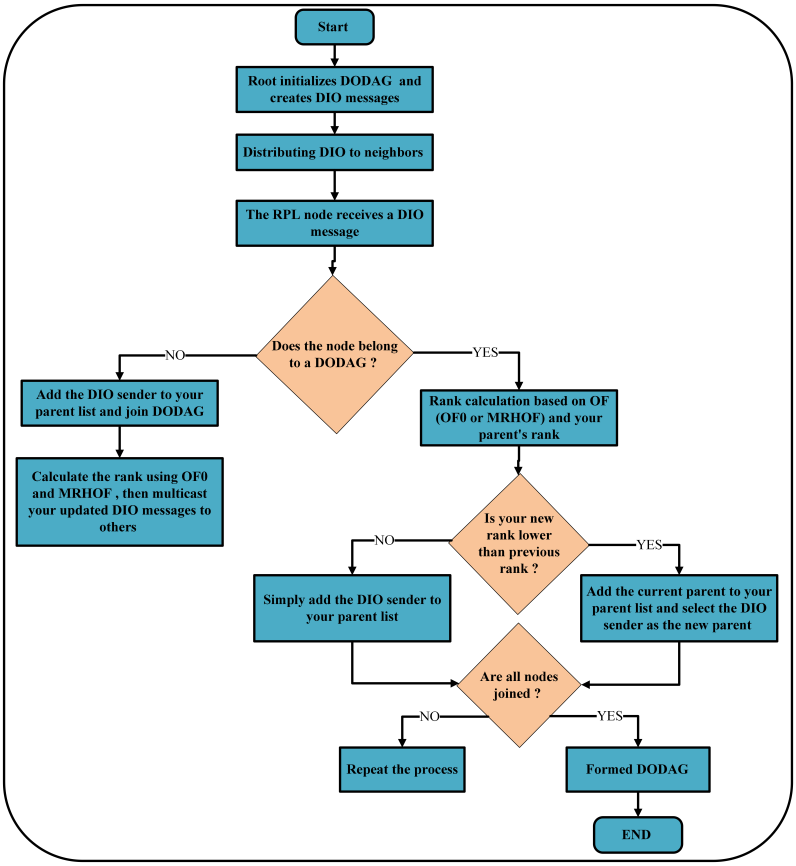


Fig. 2: Node rank calculation process.

Fig. 2 describes how nodes connect and form a DODAG RPL in IoT networks. It details the selection of parent nodes, the calculation of rank using objective functions (OF0 and MRHOF), and the propagation of DIO messages to build a stable topology.

1.2 Background on Reinforcement Learning

Machine learning is a field of artificial intelligence in which computers learn from data rather than being explicitly programmed. The principle is simple: we provide data to the machine, which then identifies patterns in order to make predictions or decisions[20–22]. Machine learning is used, for example, to recognize faces in photos, detect and classify objects in real time, recommend videos, or predict the weather. The more high-quality data the system has, the better it generally performs its task. There are three sub-domains of machine learning: supervised learning, unsupervised learning, and reinforcement learning. Focusing on the last one, reinforcement learning is a special type of machine learning[23].

Reinforcement learning (RL) is a subset of machine learning in which an agent learns to obtain the best strategy for an optimal solution by interacting with the environment. Unlike supervised machine learning algorithms, which rely on data processing, RL does not require data to learn[24, 25]. Instead, the agent learns from trials that can lead to either success or failure. It receives a reward each time it makes the right choice and a punishment when it makes the wrong choice,as shown in Fig. In this way, the agent learns from its mistakes and improves its strategy until it finds the best methodology to follow to achieve the final goal. The ultimate objective is to maximize the cumulative reward over time, and the agent learns through trial and error to select the actions that lead to the highest rewards. Several techniques are used in reinforcement learning, including multi-armed bandits, SARSA, and Q-learning. These techniques fall into one of two categories: on-policy or off-policy algorithms. Off-policy algorithms, such as Q-learning, allow the agent to act arbitrarily while still figuring out the best course of action; they do not need the agent to adhere to a particular policy during learning. SARSA and other on-policy algorithms depend on the agent knowing the current policy. Two policies are used in off-policy learning: the behavioral policy, which directs the investigation of all actions in all states, and the target policy, which stands for the ideal behaviors being learnt [26–28]. The Q-table, which is updated to reflect the anticipated future rewards, is where Q-learning stores state-action values, or Q-values. This last one can be updated according to Equation 1

$$Q_{\text{new}}(s_t, a_t) = (1 - \alpha)\, Q_{\text{old}}(s_t, a_t) + \alpha \left[R_t + \gamma \max \left(Q(s_{t+1}, a_{t+1}) \right) \right] \tag{1}$$

Where: s_t and a_t are the current state and action, respectively. The parameter $\alpha \in [0, 1]$ is the learning rate, which controls the influence of information acquired now in relation to estimates made. The reward obtained after action a_t is denoted by r_t, while $\gamma \in [0, 1]$ is the discount factor that determines the importance of future rewards. The term $\max_a Q(s_{t+1}, a_{t+1})$ corresponds to the maximum expected reward that can be achieved from the following state s_{t+1}. Reinforcement learning is widely

used to optimize the RPL protocol, particularly in the selection of optimal paths and parent nodes. By modeling the network as an environment composed of a set of interconnected nodes, each node can act as an agent capable of observing the network state (residual energy, latency, link quality, congestion, etc.) and consequently adapting its routing decisions. Thanks to the reward mechanism, the agent progressively learns from its mistakes in the current state and improves its future choices, thus dynamically optimizing network performance. This adaptive capability makes reinforcement learning particularly well-suited to RPL-based IoT networks, where network conditions are often variable and unpredictable.

1.3 Motivations and Contributions

The proposed objective function is specifically designed to handle high-traffic load conditions, where congestion and packet interference can significantly degrade network performance. It integrates congestion level, packet success rate (PSR), and expected number of transmissions (ETX) within a Q-learning framework , enabling a more comprehensive routing decision tailored to the traffic conditions. By jointly considering link quality (ETX), packet success rate (PSR), and queue occupancy (congestion level), the method achieves a balanced trade-off between reliability, latency, and energy consumption.

The main contributions of this work can be summarized as follows:

- We demonstrate that an appropriate combination of routing metrics: PSR, ETX, and congestion significantly improves network performance under-high traffic load.
- We design a reward function that incorporates both reinforcement and penalties, allowing the learning agent Q to adaptively select the most reliable parent based on the values of the aforementioned metrics.
- We provide an empirical analysis showing how different weightings of metrics influence the trade-off between PDR and latency.

The remainder of this article is structured as follows. Section 2 introduces related studies on RPL optimization and Q-learning-based routing approaches. Section 3 presents the proposed MM-QLOF approach, detailing the multi-metric design of our objective function. Section 4 describes the performance evaluation methodology and analyzes of the simulation results obtained. Finally, Section 5 concludes the article and outlines future work.

2 Related works

Researchers have shown considerable interest in the RPL protocol and have proposed various approaches to improve its reliability and energy efficiency. Consequently, several studies have addressed routing limitations under challenging network conditions. With regard to related work, many studies have focused on improving the performance of RPL in terms of packet delivery ratio (PDR), end-to-end latency, and energy consumption. Researchers have employed different techniques, such as fuzzy logic and reinforcement learning (RL), to optimize the selection of preferred parent

and enhance routing decisions in dynamic network environments. In this context, the potential of reinforcement learning (RL) to enhance communication-related decision making in Internet of Things networks was examined by the authors in [29], especially for resource-constrained devices using protocols like RPL. The article examines the advantages and drawbacks of using RL in large-scale IoT mesh networks and offers a theoretical framework that maps RL approaches to various network layers. The authors also go over future approaches and open research issues for combining RL with IoT routing and protocol design.

In order to facilitate seamless handoff in mobile IoT scenarios, the authors in [30] introduced MQW-RPL, a mobility-aware routing upgrade for RPL that chooses the preferred parent based on queue occupancy and workload while simultaneously taking ETX, residual energy, and RSSI into account. In terms of packet delivery ratio, delay, energy consumption, overhead, and convergence time in heterogeneous networks, the suggested method achieves an average gain of 13.5% over ERPL, 15.7% over mRPL, and 21.5% over standard RPL while maintaining backward compatibility with standard RPL.

The lack of coordination between the TSCH MAC layer and RPL routing was addressed by the authors in [31] by proposing ACI-RPL, an improved RPL solution for 6TiSCH networks. The suggested approach presents an enhanced initial cell allocation technique, a traffic load and cell-usage-aware goal function, and an adaptive parent modification mechanism based on cooperative Q-learning. ACI-RPL performs better than regular RPL in simulation and real-testbed evaluations, increasing packet delivery ratio and total received packets by 12% and 17%, respectively, while lowering energy usage by 23% and latency by 9%.

For resource-constrained mobile IoT networks, the authors of [32] presented QUERA, an energy-aware RPL routing algorithm based on Q-learning. Time-to-Reside (TTR), ETX, and RSSI are among the mobility and quality-aware measures that QUERA incorporates into its infinite-horizon Markov Decision Process routing formulation. It also uses an improved neighbor management technique to maintain stable candidate parents. According to experimental data, QUERA greatly enhances network performance, prolonging the entire lifespan of IoT devices in mobility scenarios while lowering energy consumption by up to 50% and increasing the packet delivery ratio by up to 12%.

For Advanced Metering Infrastructure (AMI) in Smart Grids, the authors of [33] proposed Q-RPL, a reinforcement learning-based improvement of the conventional RPL protocol. Q-RPL dynamically chooses the best next-hop forwarding nodes and adjusts to changing network conditions in wireless mesh deployments by incorporating Q-learning into RPL. When compared to regular RPL and other cutting-edge routing techniques, simulation results on actual map scenarios demonstrate that Q-RPL considerably increases the packet delivery ratio, decreases end-to-end latency, and boosts the compliance factor.

A mobile routing approach for MIoT networks that maximizes network lifetime by optimizing routing cost was developed by the authors in [34]. The suggested method uses a Q-learning algorithm to choose alternate routes when necessary and builds energy-efficient routing patterns based on residual energy and distance parameters. The suggested technique significantly improves residual energy, which improves data reachability and extends the total lifespan of the IoT network, according to performance evaluation against the standard RPL protocol.

In order to address energy consumption in low-power and lossy IoT networks, the authors of [35] presented CL-RPL-OF, a cross-layer energy-efficient objective function for RPL that incorporates MAC-layer transmission dynamics into routing decisions. In order to choose energy-optimal routing routes, the suggested objective function presents the energy-per-packet (EPP) metric in conjunction with ETX and strobe-per-packet rate utilizing fuzzy logic. In comparison to current RPL objective functions, CL-RPL-OF dramatically increases energy efficiency and packet delivery ratio while lowering strobe overhead, according to simulation findings in Cooja and experimental validation on the FIT IoT-LAB testbed.

RPL+ is a parent selection approach for RPL that addresses situations in which several candidate parents have the same ranking, according to the authors in [36]. The technique determines the most important routing parameters for selecting the best next-hop node using Random Forest feature importance analysis. The usefulness of ML-based routing strategies in Wireless Smart Grid Networks is demonstrated by the results, which show a considerable improvement in Packet Delivery Ratio when compared to typical RPL implementations.

Another study carried out where the authors [37] proposed a Q-Learning–based optimization of the RPL objective function in 6TiSCH wireless sensor networks to address frequent routing path changes. The method improves real-time performance, reduces packet loss, and increases network reliability by leveraging previous states and critical metrics. Simulation results in Contiki-NG/Cooja show significant gains in Packet Delivery Ratio (PDR) and network synchronization, outperforming standard MRHOF and OF-SR1, especially in networks of 20 and 40 nodes.

Using Cooja simulations, the authors in [38] assess RPL goal functions based on fuzzy logic in various topologies. Although it may require more parent switches, the results demonstrate that fuzzy logic can improve packet delivery ratio, decrease end-to-end latency, and lower energy consumption. Fuzzy logic works better in crowded or larger networks, whereas simpler approaches might be better for smaller networks.Table 1 summarizes the studies discussed in this section.

Table 1: Summary of related works on RPL optimization

Ref.	Protocol	Proposed Solution	Approach	Metrics Used
[29]	–	RL suitability for IoT (survey)	Survey	PDR, Latency, Energy
[30]	MQW-RPL	Queue-aware parent selection	Q-learning	ETX, Energy, RSSI
[31]	ACI-RPL	Cross-layer 6TiSCH RPL	Q-learning	Energy, Delay, PDR
[32]	QUERA	Energy and mobility-aware routing	Q-learning	ETX, RSSI, TTR
[33]	Q-RPL	RL-based RPL for AMI	Q-learning	PDR, Delay
[34]	–	Energy-efficient MIoT routing	Q-learning	Energy, Distance, Lifetime
[35]	CL-RPL-OF	MAC-aware objective function	Fuzzy Logic	ETX, EPP
[36]	RPL+	ML-based parent selection	Random Forest	PDR, Ranking
[37]	6TiSCH	Q-learning RPL optimization	Q-learning	PDR, Delay, Loss
[38]	FLOF	Fuzzy-based RPL evaluation	Fuzzy Logic	PDR, Power, Switches

3 MM-QLOF Approach

In static IoT networks, high traffic loads reduce network performance in terms of reliability and latency. As the number of packets increases, nodes compete more fiercely for channel access. This increases collisions and wait times at the MAC layer. Consequently, packets remain in the queue longer, increasing end-to-end delay. Simultaneously, buffer overflows and repeated retransmissions lead to packet loss and, consequently, a lower packet delivery rate (PDR). Ultimately, increased traffic loads negatively impact both latency and reliability[39].

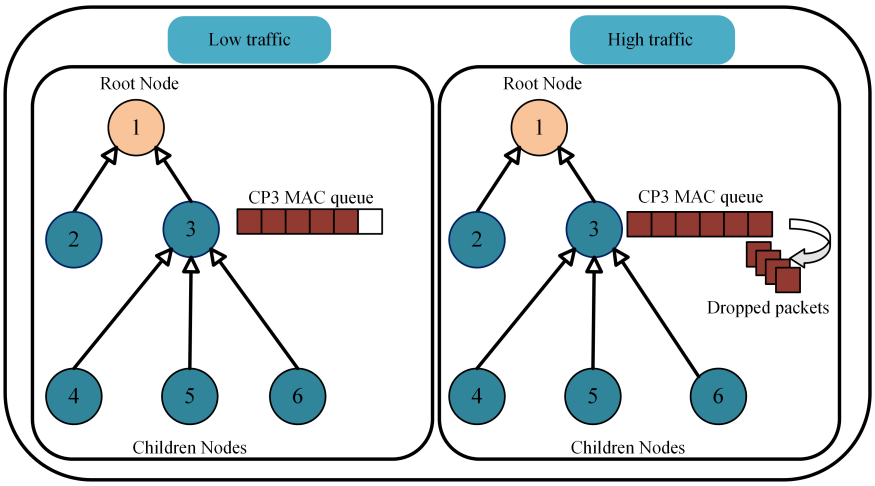


Fig. 3: Impact of Traffic Load on MAC Queue Occupancy and Packet Drops

As shown in Fig 3, when the network traffic load is moderate (30 packets/1min), congestion is minimal, resulting in fewer collisions and lower packet loss. However, when the traffic load becomes very high (120 packets/1min), the MAC queue of the parent

candidate becomes full, leading to packet loss due to the busy MAC layer. Consequently, parent candidate 3 (CP3) is no longer an optimal choice, as it serves nearby nodes, which increases the number of packets sent and, in turn, raises congestion and losses[5]. To solve this problem, a Q-learning-based objective function can be used, in which the reward function takes into account several network metrics to guide the selection of parents. This approach allows the routing process to adapt to changing network conditions (in this case, traffic load) and improves the reliability and efficiency of packet transmission.

3.1 Reward setup

MM-QROF is based on three metrics that it will use them to calculate the reward and then update the Q-values, these three metrics are: Packet Success Rate (PSR), Expected Transmission Count (ETX)and congestion level. The combination of those metrics was not randomly but each one gives an information about the network conditions. The PSR (Packet Success Rate) is an indicator that measures the proportion of packets successfully received by a specific neighboring parent (next hop). It provides an indication of the reliability of the link and can be calculated using Equation 2.

$$PSR = \frac{ACK}{T_x} \tag{2}$$

Where Tx: number of transmitted packets. ACK: number of acknowledged packets. It is important to distinguish between packet success rate (PSR) and packet delivery ratio (PDR). The PSR, as mentioned earlier, is calculated between a node and its immediate parent (next hop), reflecting the reliability of a single link. In contrast, the PDR measures end-to-end delivery, from source to destination, by measuring the overall packet delivery rate across the entire path. The second metric is ETX which indicate how many times the packet should be sent until it receives the destination successfully , the more ETX is low the more the link is optimal. The value can be calculated using Equation 3

$$ETX = \frac{1}{Df \cdot Dr} \tag{3}$$

where Df is the measured probability that a packet is received by the neighbor and Dr is the measured probability that the acknowledgment packet is successfully received. The final metric is Congestion level which is a MAC-layer metric that measures packet drops relative to transmissions at the link layer, reflecting how busy or congested the channel is [40].The value is calculated using Equation 4

$$CONGESTION = \frac{T_x}{T_x + Drops} \tag{4}$$

Where Tx: number of transmitted packets. Drops: number of dropped packets. ACK: number of acknowledged packets. When congestion is high, the node's transmission queue is likely full, leading to an increased number of dropped packets. This results in packet loss between the source and destination, reducing network reliability and potentially increasing latency as retransmissions occur.

The reward function 5 is formulated to include the metrics to be maximized in the numerator, namely NormalizedPSR and NormalizedCONGESTION, while the metrics to be minimized, notably the ETX, are placed in the denominator.

$$Reward = \frac{w_{psr} \cdot NormalizedPSR + w_{congestion} \cdot NormalizedCONGESTION}{w_{etx} \cdot ETX} \tag{5}$$

The use of network metrics mentioned in the reward function requires normalization, as each metric has its own range of values. They must be normalized to prevent one metric from overshadowing the others. Equations 6, 7 and 8 show us how this normalization is done:

$$NormalizedPSR = \frac{ACK}{TX} \times 100 \tag{6}$$

$$NormalizedCONGESTION = (1 - Congestion) \times 100 \tag{7}$$

$$NormalizedETX = ETX \tag{8}$$

In fact, an increase in NormalizedPSR and NormalizedCONGESTION (the more CONGESTION is low , the more NormalizedCONGESTION is high) leads to a higher reward, thus favoring the candidate parent. Conversely, a high ETX value without any normalization reduces the reward, acting as a penalty mechanism. Thus, the reward function combines both reinforcement and penalty effects, depending on the values of the candidate parents' metrics, allowing for adaptive selection of the most optimal parent.

3.2 MM-QLOF mechanism

As mentioned before, Q-learning is based on the definition of states and actions. In our case, the state corresponds to the IoT node seeking to select its preferred parent, while the action represents the choice of a preferred parent:Parent Selection Action (A_{PS}). Q-learning relies on a Q-table that associates a Q-value with each possible action in a given state. Thus, for a given node, the Q-table can be modeled as a vector of size (1, N), where N represents the number of candidate parents. Each candidate parent has a Q value that reflects the quality of that choice based on network conditions and the metrics considered and the candidate parent with highest Q value is selected as the preferred parent. When a node discovers a candidate parent, it first checks whether that parent already exists in the Q table. If the parent is already present, the associated Q value is updated according to the Q-learning rule to reflect the new network conditions. However, if the candidate parent does not yet exist in the table, a new entry is created for that parent. Each new candidate parent is initialized with a random Q value between 1 and 9 to encourage gradual learning by Algorithm.

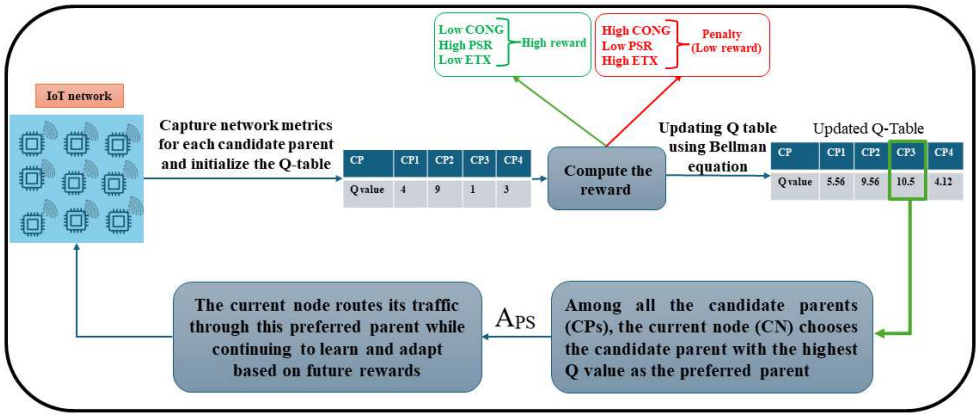


Fig. 4: Parent selection based on Q-learning in RPL routing using reward calculation.

Fig. 4 illustrates the steps of choosing a preferred parent, starting by initializing each candidate parent (CP) with a random Q value , then updating the Q values of each CP using the Bellman equation, and then applying the action of selecting the preferred parent (A_{PS}) with the highest Q value.

3.2.1 Parent Selection Process(PSP)

In this study we have two algorithms: the first algorithm is Q-Learning-Based Preferred Parent Selection algorithm(Algorithm1) that aims to select the optimal preferred parent by evaluating candidate parents using learned Q-values. The second algorithm is Q-Learning-Based Q-Value update algorithm (Algorithm2) its objective is to update the Q-values based on the received reward.

The Parent Selection Process (PSP) begins with the collection of all candidate parents from the RPL parent set, which is constructed from the DIO messages received (Algorithm 2, line 1). For each candidate parent, an entry in the Q table is created if it does not already exist, ensuring a correspondence between each parent and its associated Q value (Algorithm 2, line 2). Our Q table has dimensions $(1, N)$, where N is the number of candidate parents. Initial Q values are assigned to newly discovered neighbouring parents to encourage the learning process. In our case, we assign a random q value between 1 and 9 to each new candidate parent (Algorithm 2, lines 3-5) . Next, the link metrics for each candidate parent, including ETX, PSR, and congestion level , are captured (Algorithm 2, lines 6-9). These metrics are normalized to bring them to a common scale, after which a reward value is calculated by combining the normalized metrics using weighted coefficients (Algorithm 2, lines 10,11). The calculated reward is stored in the Q table and accumulated to update the cumulative reward of the corresponding parent (Algorithm 2, lines 12,13). The Q value is then updated using the Bellman equation, which incorporates the immediate reward and the maximum future Q value among all candidate parents (Algorithm 2, lines 14,15). The exploration rate is dynamically adjusted based on the convergence

Algorithm 1 Q-Learning-Based Preferred Parent Selection

Require: RPL parent set, Q-table, exploration rate ε, hysteresis threshold δ
Ensure: Selected preferred parent

1: $Best_Q \Leftarrow -99$
2: $Preferred_parent \Leftarrow NULL$
3: **for** each candidate parent p in RPL parent set **do**
4: **if** p belongs to another DAG **then**
5: **continue**
6: **end if**
7: **if** ETX or path cost of p exceeds threshold **then**
8: **continue**
9: **end if**
10: Update Q-value of parent p using Algorithm 2
11: Retrieve Q_p from Q-table
12: **if** $Q_p > best_Q$ **then**
13: $Best_Q \Leftarrow Q_p$
14: $Preferred_parent \Leftarrow p$
15: **end if**
16: **end for**
17: **if** $|Best_Q - Current_Q| < \delta$ **then**
18: Keep current parent
19: **else**
20: Generate random number $r \in [0, 1]$
21: **if** $r < \varepsilon$ **then**
22: Select a random acceptable parent
23: **else**
24: Select $preferred_parent$
25: **end if**
26: **end if**
27: **return** selected preferred parent

of the Q values, and the updated Q table is returned (Algorithm 2, lines 16-19). The preferred parent selection (PPS) process is described in Algorithm 1. It begins by initializing the best Q value with an infinite value ad the best parent is initialzed to NULL,(Algorithm 1, lines 1,2). Parents that belong to a different DAG or do not satisfy the eligibility constraints in terms of link and path cost are immediately excluded from the selection process (Algorithm 1, lines 3-9). For each acceptable parent, the Q value is updated (Algorithm 1, line 10) using the *update_q_ values* function described in Algorithm 1, and the candidate parent with the highest Q value is selected as the preferrred parent (Algorithm 1, lines 11-16). To avoid frequent parent changes, a hysteresis mechanism is applied by comparing the Q value of the best candidate with that of the current preferred parent; if the difference does not exceed a predefined threshold, the current parent remains. (Algorithm 1, lines 17-18). Finally, an ε-greedy

Algorithm 2 Q-Learning-Based Q-Value Update

Require: Parent p, learning rate α, discount factor γ
Ensure: Updated Q-value for parent p

1: Receive parent p from RPL parent set
2: Find or create Q-table entry for parent p
3: **if** Q_p is not initialized **then**
4: $\quad Q_p \Leftarrow$ random value between 1 and 9
5: **end if**
6: Retrieve link statistics of parent p
7: Compute ETX_p
8: Compute $PSR_p \Leftarrow ACK/TX$
9: Compute $Congestion_p \Leftarrow Drops/(TX + Drops)$
10: Normalize link metrics
11: Compute reward:

$$R_p \Leftarrow \frac{w_{PSR} \cdot PSR_p + w_{Cong} \cdot (1 - Congestion_p)}{w_{ETX} \cdot ETX_p}$$

12: Store link metrics and reward in Q-table
13: Update cumulative reward:

$$CumReward_p \Leftarrow CumReward_p + R_p$$

14: $Q_{max} \Leftarrow \max(Q_i)$
15: Update Q-value using Bellman equation:

$$Q_p \Leftarrow (1 - \alpha) \cdot Q_p + \alpha \cdot (R_p + \gamma \cdot Q_{max})$$

16: **if** Q-value convergence is detected **then**
17: $\quad$ Adapt exploration rate ε
18: **end if**
19: **return** updated Q-table

strategy is used to balance exploration and exploitation. With probability ε, a random acceptable parent is selected to explore other paths(Algorithm 1, lines 19-22), while with probability $1 - \varepsilon$, the parent with the highest Q value is chosen as the preferred parent (Algorithm 1, lines 23-27). Over time, the learning process stabilizes, resulting in a very small difference between the old Q value and the updated Q value. This indicates that the algorithm has sufficiently learned the quality of the preferred parent and that further updates would only bring negligible improvements. In this case, the exploration rate ε must be gradually reduced in order to favor the exploitation phase to choose the preferred parent with a high Q value, while maintaining a minimum level of exploration to be able to react to changes.

3.2.2 Exploration–Exploitation Strategy and Hysteresis Mechanism

As previously explained, whenever a candidate parent is discovered or already exists, its Q-value is updated. If updates are too frequent or too sensitive to small variations in routing metrics, one parent may appear slightly better than the previous one due to minor changes in the Q-value. This can trigger unnecessary parent changes, leading to network instability and packet loss.

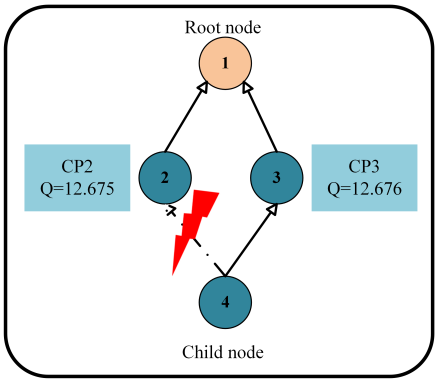


Fig. 5: Illustration of child-parent instability caused by slightly higher Q-values.

Fig. 5 shows us that child node will switch preferred parent, moving from actual parent (2) to a new parent (3), due to a slightly higher Q value. This will cause a communication interruption during packet transmission between the child and its parent, resulting in delays and packet loss.Therefore, to overcome this issue we used a hysteresis mechanism: a parent is selected only if its new Q value exceeds the previous Q value of the current preferred parent by at least 10%; otherwise, the previous parent is retained.

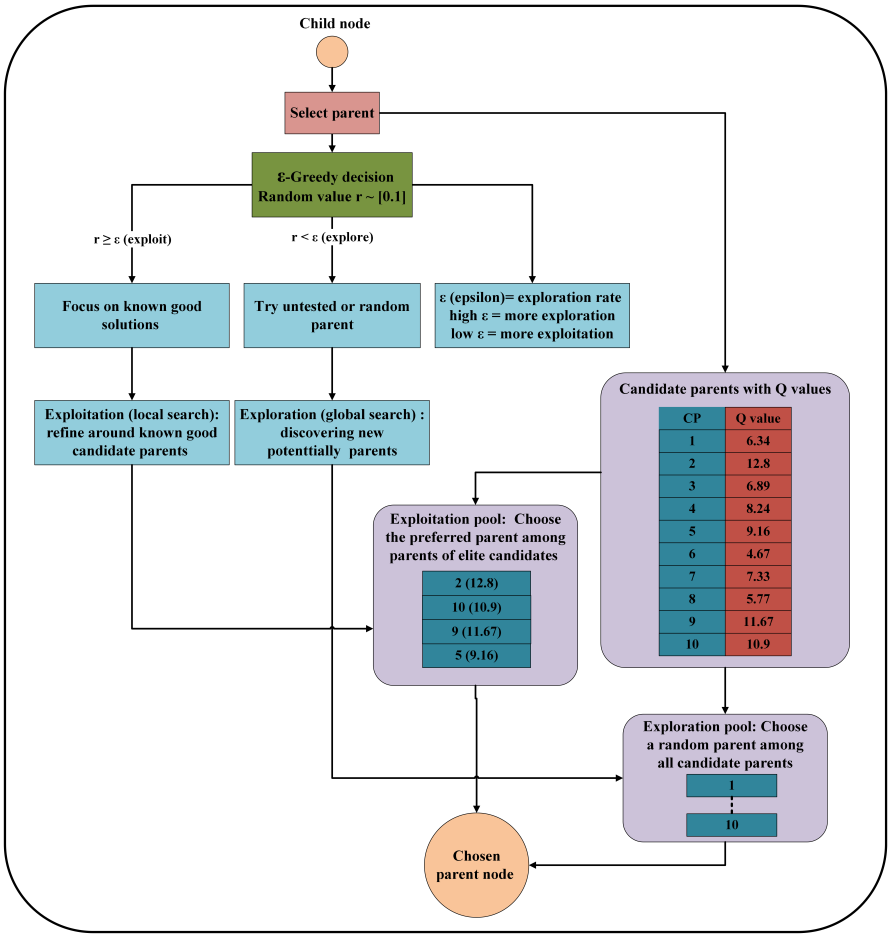


Fig. 6: Exploitation and exploration processes

Fig.6 illustrates the parent selection mechanism in MM-QLOF, using an ε-greedy strategy. Each child node selects a parent from a set of candidates, each associated with a Q value representing its quality according to the three previously mentioned metrics. A random value $r \in [0, 1]$ is generated: if $r < \varepsilon$, the node explores by selecting a random parent from the exploration set (local and global search) to discover potentially better parents. If $r \geq \varepsilon$, the node moves to the exploitation phase, choosing the parent with the highest Q value from among the parent candidates for exploitation (elite candidates). In our study, we used an exploitation ratio of 70% and an exploration ratio. To optimize learning, ε is dynamically adjusted based on the convergence of the Q values: when updates to the Q values become negligible ($|\Delta Q| < 0.001$), indicating learning saturation, ε is progressively decreased. This adaptive mechanism ensures a smooth transition from exploration to exploitation once stable network conditions are reached. This balance between exploration and exploitation allows MM-QLOF

to dynamically adapt to network conditions, ensuring optimal parent selection for improved network performance (high PDR and reduced latency).

4 Evaluation and results

4.1 Evaluation setup

The MM-QLOF is evaluated under Cooja, a widely used wireless sensor network emulator integrated into Contiki-NG. Cooja offers the ability to generate realistic IoT network scenarios with configurable topology, traffic patterns, and protocol parameters. Various simulation scenarios were developed to examine the influence of traffic load on network performance, the traffic load is adjusted from one packet every 3 seconds to one packet every 0.5 seconds, representing low to high traffic conditions.
In this study, a UDP client-server application is used on a network of 51 nodes, comprising a DODAG root node acting as a UDP server and 50 client nodes, distributed according to a linear topology based on a 100×100 m^2 grid. The root node is placed near the left edge of the grid and receives packets sent by the client nodes.
The evaluation focuses on key performance indicators, namely packet delivery rate (PDR), end-to end latency, and energy consumption.The results obtained with MM-QLOF are compared to the standard RPL objective functions OF0 and MRHOF in order to highlight the effectiveness of the proposed approach in terms of reliability, latency, and energy efficiency. The other parameters used in our study are shown in Table 2

Table 2: Simulation Parameters

Parameter	**Value/Description**
Emulator	Cooja
Radio medium	UDGM - Distance Loss
Mote type	Cooja mote
Topology	Linear
Network density	51 nodes
Area dimensions	100 m × 100 m
Operating system	Contiki-NG
Transport protocol	UDP
Routing	RPL Classic
	Objective Functions: MM-QLOF, MRHOF and OF0
MAC protocol	CSMA
Simulation duration	30 minutes
Send interval	3s 2s 1s 0.5s

4.2 Results and discussion

The reward function combines three metrics: PSR, ETX, and congestion level. To evaluate the impact of traffic load, we empirically calibrated the metric weights by

varying the packet transmission interval (3 s, 2 s, 1 s, 0.5 s) to maintain a balance between packet delivery ratio (PDR), latency, and energy consumption. The results demonstrate that different weight combinations affect network performance differently, highlighting the trade-offs among reliability, end-to-end delay, and energy efficiency. As

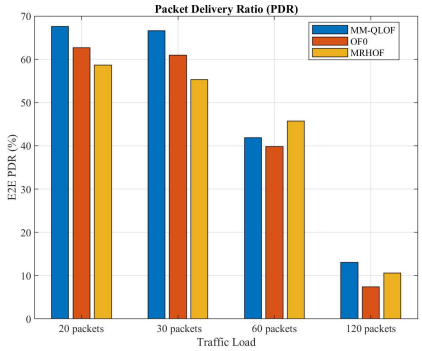


Fig. 7: Average PDR in different traffic load scenarios: MM-QLOF vs OF0 and MRHOF.

shown in Fig. 7, MM-QLOF maintains a high Packet Delivery Ratio (PDR) even under increased traffic load. On average, it improves the PDR by **10.7%** compared to OF0 and by **11.1%** compared to MRHOF. In the high-traffic scenario of 120 packets/1min, MM-QLOF outperforms OF0 by **4.65%** and MRHOF by **1.52%**, demonstrating the effectiveness of the selected parents in ensuring reliable packet delivery under heavy load.

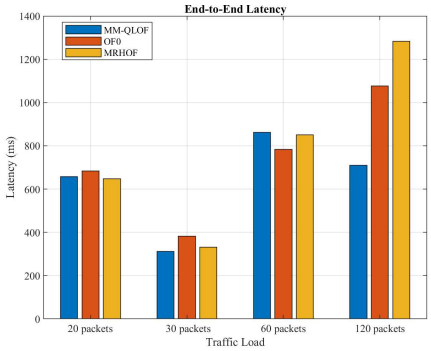


Fig. 8: Average end-to-end latency in different traffic load scenarios: MM-QLOF vs OF0 and MRHOF

In terms of latency, as illustrated in Fig. 8, MM-QLOF achieves lower end-to-end delay

than the default objective functions. On average, it reduces latency by **18.7%** compared to OF0 and by **23.6%** compared to MRHOF. In the high-traffic scenario of 120 packets/1min, MM-QLOF outperforms OF0 by **34.11%** and MRHOF by **44.66%**. demonstrating the effectiveness of the selected parents in ensuring reliable packet delivery under heavy load. These results confirm the effectiveness of the selected parent: packets are routed along less congested and more reliable paths, thus minimizing interference, collisions, and transmission delay.

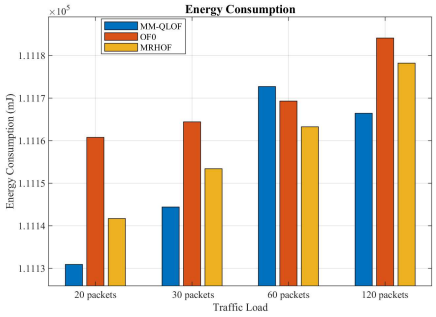


Fig. 9: Average energy consumption in different traffic load scenarios: MM-QLOF vs OF0 and MRHOF

The improved PDR and end-to-end latency indicate that MM-QLOF effectively avoids unreliable parent nodes and reduces retransmissions and transmission delays. Since retransmissions and delays directly increase energy consumption, this improvement has a positive impact on the overall energy efficiency of the network. As shown in Fig. 9, MM-QLOF enables a slight reduction in energy consumption compared to OF0 and MRHOF, thus confirming the benefit of better parent node selection under high-traffic conditions.

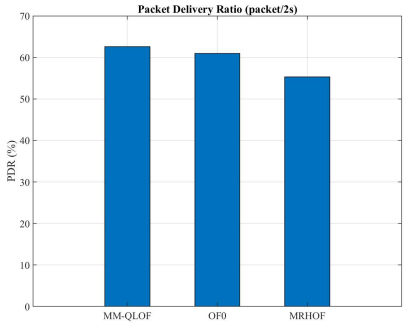
It can be noted that for the scenario of 60 packets/1min, MM-QLOF gives slightly less reliable results compared to MRHOF, particularly in terms of PDR and energy consumption. This can be explained by the fact that the weights associated with the metrics need to be calibrated, as the traffic load becomes somewhat critical. After several trials, we found more optimal weightings for scenarios where the traffic load becomes significant, which is clearly visible in the 120 packets/1min scenario, where MM-QLOF managed to outperform OF0 and MRHOF in terms of PDR ,latency and energy consumption. Speaking of optimal weightings, according to the empirical study we conducted, it is important to adapt the parameter weightings according to the scenario. When the traffic load is average and non-critical (20 packet/1min or 30 packets/1min), the ETX and PSR weightings should be higher than that of congestion. On the other hand, when the traffic load becomes heavy, it is necessary to prioritize congestion level, because when congestion is high , the node's transmission queue is likely full, leading to an increased number of dropped packets, this results in

packet loss between the source and the destination , reducing network reliability and potentially increasing latency and energy consumption as retransmissions occur.

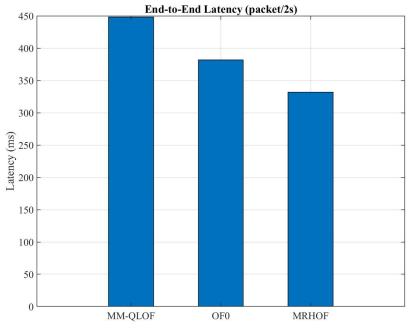
4.2.1 Impact of Metric Weighting on Performance in Empirical Studies

The effect of each metric on network performance is analyzed through a series of tests conducted under the scenario of one packet every 2 seconds. The results show that the choice of metric weightings can favor PDR at the expense of latency, or vice versa. For this end, two scenarios were created :

1. **The first scenario**: greater importance is given to the PSR metric ($w_{\text{psr}} = 0.7$, $w_{\text{etx}} = 0.1$, $w_{\text{congestion}} = 0.2$).



(a) Average PDR in packet/2s scenario : MM-QLOF vs. OF0 and MRHOF

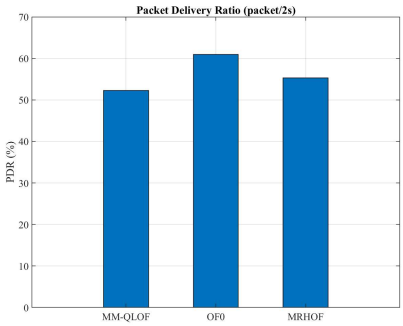


(b) Average latency in packet/2s scenario : MM-QLOF vs. OF0 and MRHOF

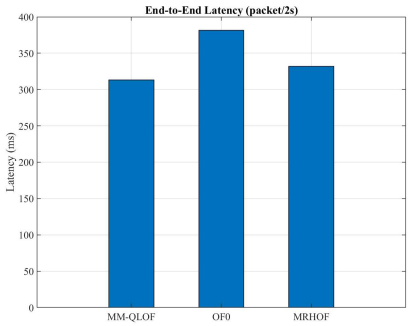
Fig. 10: Impact of PDR on End-to-End Latency in Comparison with OF0 and MRHOF under ($w_{\text{psr}} = 0.7$, $w_{\text{etx}} = 0.1$, $w_{\text{congestion}} = 0.2$).

The results obtained are presented in Figs. 10a and 10b, as can be seen the PDR achieved is very high compared to OF0 and MRHOF. This can be explained by the fact that the choice of parents becomes more optimal, as PSR provides clear indications of parent reliability. On the other hand, we note that latency becomes very high compared to OF0 and MRHOF, since congestion weighting is minimal, which increases the probability of losses and high queues.

2. **The second scenario**: priority is given to the congestion level metric ($w_{\text{psr}} = 0.2$, $w_{\text{etx}} = 0.1$, $w_{\text{congestion}} = 0.7$).



(a) Average PDR in packet/2s scenario : MM-QLOF vs. OF0 and MRHOF



(b) Average latency in packet/2s scenario : MM-QLOF vs. OF0 and MRHOF

Fig. 11: Impact of End-to-End Latency on PDR in Comparison with OF0 and MRHOF under ($w_{\text{psr}} = 0.2$, $w_{\text{etx}} = 0.1$, $w_{\text{congestion}} = 0.7$).

The corresponding results are presented in Figs. 11a and 11b, as can be seen, latency becomes more acceptable compared to OF0 and MRHOF, confirming the direct impact of the congestion metric on network responsiveness, but we obtained a PDR lower than that of OF0 and MRHOF.

Based on the empirical study conducted in the article, we are able to determine which performance parameter (e.g., PDR, latency, energy consumption etc.) is influenced by each metric used in MM-QLOF. PSR and ETX jointly improve PDR, as the path chosen by the parent has good link quality. Assigning them a high weight will therefore ensure reliable packet transmission and reception. The Congestion level metric has a greater influence on latency and energy consumption, as the node will avoid unavailable paths, thereby reducing collisions, packet loss, and retransmission, ensuring efficient and fast communication. Table 3 shows how the weights for each metric can be chosen to ensure a compromise between PDR, end-to-end latency, and energy consumption.

Table 3: Weights chosen to balance PDR, end-to-end latency, and energy consumption.

Low traffic	Medium traffic	High traffic load
PSR > ETX > Cong	PSR, ETX > Cong	Cong $\gg$ ETX, PSR

5 Conclusion

In this article, we present a decentralized objective function based on Q-learning for optimal parent node selection. Our approach is specifically designed to meet the high-traffic requirements of IoT applications by allowing nodes to dynamically adapt their parent node selection based on network conditions. Each node can independently choose its preferred parent from among all candidate parents that maximizes a reward function. The Q-learning algorithm continuously updates the Q values, ultimately selecting the parent with the highest value. For efficient parent selection, we have combined three key metrics: the number of expected transmissions (ETX), the packet success rate (PSR), and the congestion level. The combination of these indicators makes the selection process more optimal, especially under high traffic loads.

Under high traffic loads, the congestion metric plays a crucial role in parent selection by effectively identifying and avoiding candidate parents with saturated MAC queues. This proactive avoidance of congested candidate parents significantly reduces retransmissions, collisions, and packet loss, thus improving the reliability and speed of packet transmission and reception. The effectiveness of our objective function is demonstrated by empirical results, showing an 11.1% improvement in PDR and a 23.6% reduction in end-to-end latency compared to OF0 and MRHOF.

Furthermore, this approach allows us, through empirical studies, to understand how the weighting of these metrics influences overall network performance. This understanding provides important information for achieving an optimal balance between them, thus enabling a trade-off between packet PDR, latency, and energy consumption according to the requirements of each application.

For future work, the proposed objective function, based on Q-learning, can be extended to more dynamic environments. The model can be evaluated in mobile IoT scenarios to assess its adaptability to topology changes, and additional metrics, such as residual energy and delay, can be incorporated to improve the selection of optimal parents under heavy traffic.

6 Acknowledgments

The authors would like to thank the Centre National pour la Recherche Scientifique et Technique (CNRST), Morocco, for supporting this research through the PhD-Associate Scholarship (PASS) program.

7 Statements and Declarations

7.1 Funding

This research received no external funding.

7.2 Competing interests

The authors declare that they have no competing financial interests or personal relationships that could have apperaed to influence the work reported in this paper.

7.3 Ethics approval

Not applicable.

7.4 Consent to participate

Not applicable

7.5 Consent for publication

Not applicable

7.6 Data availability

The datasets generated and analyzed during the current study are available from the corresponding author on reasonable request.

7.7 Code availability

The Q-learning simulation code developed during this study is available from the corresponding author on reasonable request.

7.8 Authors' contributions

(YT - Youness TALBI; AEH - Abdelhadi ELOUDRHIRI HASSANI; AS - Adil SALBI; IB - Issam BOUGANSSA)
Conceptualization: YT, AEH
Data Curation: YT
Formal Analysis: YT
Software: YT
Methodology: YT, AEH
Validation: YT, AEH
Investigation: YT
Writing – Original Draft: YT
Writing – Review and Editing: AEH, AS
Supervision: IB
All authors read and approved the final manuscript.

References

[1] Abujassar, R.S.: A multi path routing protocol with efficient energy consumption in iot applications real time traffic. EURASIP Journal on Wireless Communications and Networking **2024**(1), 46 (2024) https://doi.org/10.1186/s13638-024-02377-1

[2] Ines Kechiche, Ines Bousnina, A.S.: A Review of RPL Objective Function based Enhancement Approaches. Advances in Science, Technology and Engineering Systems Journal **5**(5), 201–211 (2020) https://doi.org/10.25046/aj050525

[3] Verma, A., Ranga, V.: The impact of copycat attack on rpl based 6lowpan networks in internet of things. arXiv preprint arXiv:2303.00273 (2023) https://doi.org/10.48550/arXiv.2303.00273

[4] Haggag, A.: Implementation and evaluation of ipv6 with compression and fragmentation for throughput improvement of internet of things networks over ieee 802.15.4. Wireless Personal Communications **130**(2), 1449–1477 (2023) https://doi.org/10.1007/s11277-023-10340-4

[5] Gaddour, O., Koubâa, A.: Rpl in a nutshell: A survey. Computer Networks **56**(14), 3163–3178 (2012) https://doi.org/10.1016/j.comnet.2012.06.016

[6] Levin, S.L., Schmidt, S.: Ipv4 to ipv6: Challenges, solutions, and lessons. Telecommunications Policy **38**(11), 1059–1068 (2014) https://doi.org/10.1016/j.telpol.2014.06.008

[7] Wakili, A., Bakkali, S.: Aof: An adaptive algorithm for enhancing rpl objective function in smart agricultural iot networks. International Journal of Intelligent Networks **5**, 325–339 (2024) https://doi.org/10.1016/j.ijin.2024.09.001

[8] Kamble, S., Bhilwar, P., Chandavarkar, B.R.: Novel fuzzy-based objective function for routing protocol for low power and lossy networks. Ad Hoc Networks **144**, 103150 (2023) https://doi.org/10.1016/j.adhoc.2023.103150

[9] Chia, P.S., Kamis, N.H., Abdul Razak, S.F., Yogarayan, S., Yassin, W., Abdollah, M.F.: Routing protocols performance on 6lowpan iot networks. IoT **6**(1) (2025) https://doi.org/10.3390/iot6010012

[10] Hassani, A.E., Sahel, A., Badri, A.: Irh of: A new objective function for rpl routing protocol in iot applications. Wireless Personal Communications **119**, 673–689 (2021) https://doi.org/10.1007/s11277-021-08230-8

[11] Eloudrhiri Hassani, A., Sahel, A., Badri, A.: Ftc-of: Forwarding traffic consciousness objective function for rpl routing protocol. International Journal of Electrical and Electronic Engineering & Telecommunications **10**, 168–175 (2021) https://doi.org/10.18178/ijeetc.10.3.168-175

[12] Kim, S., Kim, H.-s., Kim, C.-K.: A3: Adaptive autonomous allocation of tsch slots. (2021). https://doi.org/10.1145/3412382.3458273

[13] Hkiri, A., Faraj, H., Bahri, O.B., Karmani, M., Alqurashi, S., Machhout, M.: The impact of network topologies and radio duty cycle mechanisms on the rpl routing protocol power consumption. Computers, Materials and Continua **79**(2),

1835–1854 (2024) https://doi.org/10.32604/cmc.2024.049207

[14] Hassani, A.E., Sahel, A., Badri, A.: Impact of rpl objective functions on energy consumption in ipv6 based wireless sensor networks. In: Colloque sur les Objets et Systèmes Connectés (2019)

[15] Rajasekar, V.R., Rajkumar, S.: A study on impact of dis flooding attack on rpl-based 6lowpan network. Microprocessors and Microsystems **94**, 104675 (2022) https://doi.org/10.1016/j.micpro.2022.104675

[16] Alotaibi, M.A., Alwakeel, S.S., Alyahya, A.N.: A novel reliable and trust objective function for rpl-based iot routing protocol. Computers, Materials and Continua **82**(2), 3467–3497 (2025) https://doi.org/10.32604/cmc.2025.060599

[17] Mardini, W., Ebrahim, M., Al-Rudaini, M.: Comprehensive performance analysis of rpl objective functions in iot networks. International Journal of Communication Networks and Information Security **9**, 323–332 (2017) https://doi.org/10.17762/ijcnis.v9i3.2515

[18] Abdel Hakeem, S.A., Hady, A.A., Kim, H.: Rpl routing protocol performance in smart grid applications based wireless sensors: Experimental and simulated analysis. Electronics **8**(2) (2019) https://doi.org/10.3390/electronics8020186

[19] Almusaylim, Z.A., Alhumam, A., Jhanjhi, N.Z.: Proposing a secure rpl based internet of things routing protocol: A review. Ad Hoc Networks **101**, 102096 (2020) https://doi.org/10.1016/j.adhoc.2020.102096

[20] Alzaman, C.: Product selection in drop-shipping: Reinforcement learning stacked on multiple machine learning algorithms. Applied Soft Computing **190**, 114515 (2026) https://doi.org/10.1016/j.asoc.2025.114515

[21] Schindlegger, F., Hupperich, T.: Reinforcement learning and movement prediction for adaptive routing in opportunistic networks. Computer Networks, 112084 (2026) https://doi.org/10.1016/j.comnet.2026.112084

[22] Moulay, M.A., Salbi, A., Bouganssa, I., Lasfar, A.: Deep learning for building segmentation and change detection in urban monitoring: 'study case: Zenata city'. Engineering Research Express **7**(3), 0352–6 (2025) https://doi.org/10.1088/2631-8695/ae024f

[23] Suresh, S.S., Prabhu, V., Parthasarathy, V., Senthilkumar, G., Gundu, V.: Intelligent data routing strategy based on federated deep reinforcement learning for iot-enabled wireless sensor networks. Measurement: Sensors **31**, 101012 (2024) https://doi.org/10.1016/j.measen.2023.101012

[24] Kumar, A., Singh, D.: Securing iot devices in edge computing through reinforcement learning. Computers & Security **155**, 104474 (2025) https://doi.org/10.

1016/j.cose.2025.104474

[25] Budania, S., Shenoy, M.V.: Rpl*: An explainable ai-based routing protocol for internet of mobile things. Internet of Things **27**, 101283 (2024) https://doi.org/10.1016/j.iot.2024.101283

[26] Bekar, T., Görmüş, S., Aydin, B., Aydın, H.: Q-learning algorithm inspired objective function optimization for ietf 6tisch networks, pp. 1–6 (2023). https://doi.org/10.1109/SmartNets58706.2023.10216124

[27] Terven, J.: Deep reinforcement learning: A chronological overview and methods. AI **6**(3) (2025) https://doi.org/10.3390/ai6030046

[28] Sharma, G., Grover, J., Verma, A.: Qsec-rpl: Detection of version number attacks in rpl based mobile iot using q-learning. Ad Hoc Networks **142**, 103118 (2023) https://doi.org/10.1016/j.adhoc.2023.103118

[29] Musaddiq, A., Olsson, T., Ahlgren, F.: Reinforcement-learning-based routing and resource management for internet of things environments: Theoretical perspective and challenges. Sensors **23**(19) (2023) https://doi.org/10.3390/s23198263

[30] Diniesh, v., Palaniswamy, B., Murali, L., N., M., Ramaswamy, K.: Improved energy efficient load balanced mobility management rpl protocol for mobile internet of things networks. Scientific Reports **15** (2025) https://doi.org/10.1038/s41598-025-27253-w

[31] Fawwaz, D., Chung, S.-H.: Intelligent parent change to improve 6tisch network transmission using multi-agent q-learning. IEEE Access (2024) https://doi.org/10.1109/ACCESS.2024.3433589

[32] Rezagholi Lalani, S., Safaei, B., Monazzah, A.M., Taghizadeh, H., Henkel, J., Ejlali, A.: Quera: Q-learning rpl routing mechanism to establish energy efficient and reliable communications in mobile iot networks. IEEE Transactions on Green Communications and Networking **PP**, 1–1 (2024) https://doi.org/10.1109/TGCN.2024.3399455

[33] Duenas Santos, C.L., Mezher, A.M., Astudillo León, J.P., Cardenas Barrera, J., Castillo Guerra, E., Meng, J.: Q-rpl: Q-learning-based routing protocol for advanced metering infrastructure in smart grids. Sensors **24**(15) (2024) https://doi.org/10.3390/s24154818

[34] Aziz, L., Achki, S., El Gourari, A., Zaouit, C.A., El Ghazouani, M.: A new version of rpl protocol to maximize lifetime in miot networks. Procedia Computer Science **265**, 236–243 (2025) https://doi.org/10.1016/j.procs.2025.07.177 . 20th International Conference on Future Networks and Communications/ 22nd International Conference on Mobile Systems and Pervasive Computing/15th International Conference on Sustainable Energy Information Technology (FNC/MobiSPC/SEIT

2025)

[35] Poornima, M., Vimala, H., . J, S.: Fuzzy-based novel cross-layer rpl objective function for energy-aware routing in iot. International Journal of Computational Intelligence Systems **18** (2025) https://doi.org/10.1007/s44196-025-00916-2

[36] Dueñas Santos, C., Astudillo, J., Mohamad-Mezher, A., Cárdenas-Barrera, J., Meng, J., Guerra, E.: Rpl+: An improved parent selection strategy for rpl in wireless smart grid networks, pp. 75–82 (2022). https://doi.org/10.1145/3551663.3558677

[37] Bekar, T., Görmüş, S., Aydin, B., aydın, H.: Q-learning algorithm inspired objective function optimization for ietf 6tisch networks, pp. 1–6 (2023). https://doi.org/10.1109/SmartNets58706.2023.10216124

[38] Alauddin: Performance evaluation of fuzzy logic-based rpl objective functions. International Journal of Innovations in Science and Technology **6** (2024) https://doi.org/10.33411/ijist/202462918929

[39] Srivastava, V., Paul, A., Sinha, S.: Mmof: A multi-metric objective function for congestion detection under varying transmission ranges in rpl-based wsn. SN Computer Science **5** (2024) https://doi.org/10.1007/s42979-024-03391-2

[40] Darabkh, K.A., Al-Akhras, M., Zomot, J.N., Atiquzzaman, M.: Rpl routing protocol over iot: A comprehensive survey, recent advances, insights, bibliometric analysis, recommendations, and future directions. Journal of Network and Computer Applications **207**, 103476 (2022) https://doi.org/10.1016/j.jnca.2022.103476