A Q-Learning Approach to Energy-Efficient Routing in BLE Mesh Network Based on Duty Cycle Scanning

Longrong Jiang[1,2,3], Jing Liu[1](✉), and Lan Wang[1]

[1] College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China
`liujing@szu.edu.cn`
[2] Research Institute of Tsinghua University in Shenzhen (RITS), Shenzhen, China
[3] Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing, Guilin, China

Abstract. The Bluetooth Special Interest Group (SIG) introduced the Bluetooth Low Energy Mesh (BLE Mesh) network specification in 2017, enabling multi-to-multi communication capability for devices operating on the Bluetooth Low Energy protocol. This specification has made BLE mesh network versatile for a range of Internet of Things (IoT) applications, particularly in building lighting and smart home systems. However, the existing BLE mesh network specification employs a managed-flood-based mechanism at the network layer for message dissemination, resulting in both message redundancy and unnecessary energy expenditure. This paper makes two innovative contributions to address these shortcomings: 1) Introduction of a broadcast routing protocol based on Q-learning algorithms. This approach enables network nodes to optimally select the next-hop relay node utilizing localized Q-value tables, thereby substantially mitigating data packet redundancy within the network. 2) Formulation of a comprehensive set of scanning-broadcasting strategies. These strategies not only ensure the reliable transmission of data packets but also facilitate a low-power standby mode for the majority of the network nodes' operational time, thereby enhancing the overall energy efficiency of the network. Based on the results of our simulation experiments, the proposed methodology significantly enhances the longevity of nodes while concurrently minimizing message redundancy within BLE mesh network.

Keywords: Bluetooth Low Energy mesh · Q-learning · energy consumption

1 Introduction

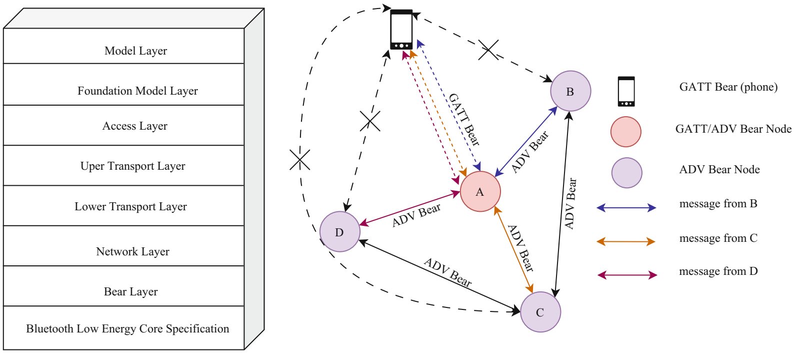
The architecture of the BLE mesh network is hierarchically designed, based on the core specifications of Bluetooth Low Energy (BLE). It consists of multiple layers, organized from top to bottom as follows: model layer, foundation model layer, access layer, upper

---

The Work Is Supported Partly By the Shenzhen Natural Science Foundation Under Grant JCYJ20200109143016563, and the Opening Project of Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing and Key Laboratory of Cognitive Radio and Information Processing, Ministry of Education Under Grant GXKL06220201.

X. Chen et al. (Eds.): IoTaaS 2023, LNICST 585, pp. 358–370, 2025.
https://doi.org/10.1007/978-3-031-70507-6_27

transport layer, lower transport layer, network layer, bearer layer, and the BLE core specification layer. In the bearer layer of the BLE mesh network, two types of communication methods are defined: advertising bearer (ADV bearer) and generic attribute profile bearer (GATT bearer) [1]. In the ADV bearer-based communication, three out of the 40 channels defined in the BLE core specification are used for message transmission. For devices that do not use broadcasting channels for data transmission, such as smartphones, communication is facilitated through GATT connections, enabling interaction with broadcast-based BLE mesh devices via a proxy. At the Network layer, the BLE mesh Profile mandates the use of a managed-flood-based routing mechanism. To prevent broadcast storms, network message cache and time to live (TTL) are employed to limit the number of times a message is forwarded within the network. The network message cache stores processed messages for a certain period or number of messages, preventing further processing of the same message. The TTL field is added to the message, decrementing by one each time the message is relayed. When the TTL value falls below 2, the message is no longer forwarded by the relay nodes. The BLE mesh network model and different communication methods within the network are illustrated in Fig. 1.



(a) (b)

Fig. 1. (a) BLE mesh network architecture. (b) Illustration of communication mechanisms between ADV bearer and GATT bearer nodes in the BLE mesh network.

Despite the constraints introduced by TTL and message caching in the managed flooding mechanism, nodes still engage in superfluous broadcasts, leading to a significant prevalence of duplicate messages in the network. Moreover, the absence of a synchronization mechanism among nodes necessitates that relay nodes maintain a 100% scanning duty cycle, exacerbating the energy consumption issue [2]. In forthcoming iterations of the BLE mesh specification, it is anticipated that the integration of advanced energy-efficient routing algorithms will significantly contribute to extending the overall lifespan of the network.

Research by Pai-Chet Ng and colleagues has proposed a method called BOM [3], which incorporates mesh functionality into Beacon networks. This method employs a bounded flooding algorithm based on the Received Signal Strength Indicator (RSSI) to mitigate broadcast redundancy. By determining the message forwarding probability through the received signal strength, the introduction of Beacon technology into BLE mesh network reduces the frequency of message forwarding based on the managed-flood-mechanism [4]. Furthermore, Emil and others [5], have introduced a relay node selection mechanism in BLE mesh network to reduce message redundancy. They conducted performance evaluations of BLE mesh network employing various relay node selection algorithms, such as K2 Pruning [6], Greedy Connect [7], and Dominator [8], tailoring the choice of algorithm to the network's dynamic nature and scale. Given the dynamic characteristics, energy constraints, and connectionless features of BLE mesh network, a routing mechanism based on Q-learning may offer a more suitable solution to the challenges of message redundancy and high energy consumption.

In fact, routing mechanisms based on Q-learning were initially proposed in telephone networks in the last century [9], taking into account network congestion and hop count to the destination to find the optimal routing path. In Q-learning-based routing algorithms, each node relies solely on local information and operates in a fully distributed manner. Due to its decentralized and adaptive nature, this routing algorithm has been widely applied in wireless sensor networks to optimize metrics such as network latency and energy consumption. A high-energy-efficient underwater sensor network routing algorithm (QELAR) [10] has been introduced, incorporating Q-learning to balance node energy consumption and extend network lifespan. To consider energy consumption more comprehensively, Xing Su and colleagues [11] have included node orientation, transmission distance, energy consumption for data transmission, and remaining node energy into the Q-learning reward function to achieve high-energy-efficient routing.

In summary, given that the existing routing mechanisms in BLE mesh network require further refinement, a Q-learning-based approach aligns well with the network's dynamic, distributed, and energy-constrained nature. This paper, therefore, concentrates on implementing next-hop relay node selection through Q-learning and introduces a set of scanning-broadcasting strategies aimed at minimizing energy consumption. The primary contributions of this paper are:

- The introduction of a Q-learning-based routing algorithm for BLE mesh network. This algorithm, grounded in managed-flood mechanisms, integrates both the hop count to the destination node and the residual energy of neighboring nodes into its reward function. The objective is to balance energy consumption across nodes while minimizing routing overhead, thereby selecting the most efficient node for packet forwarding.
- The proposal of a scanning-broadcasting strategy that enables nodes in asynchronous BLE mesh network to reduce scanning time and increase the duration spent in low-power standby mode, thereby enhancing energy efficiency and extending network longevity.

The remainder of this paper is structured as follows: Sect. 2 elucidates the mathematical model underlying routing in BLE mesh network. Section 3 introduces the basic principle of the proposed approach in detail. Section 4 analyzes simulation results; and Sect. 5 offers concluding remarks.

2 Mathematical Modeling of BLE Mesh Network Routing

To offer a more precise depiction of the interrelationships among relay nodes within BLE mesh network, we model the network as an undirected graph $G(R, E)$. In this graph, the vertices correspond to the relay nodes in the network and are denoted by the set $R = \{r_1, r_2, ..., r_m\}$. For each relay node r_i, its set of neighboring nodes is represented as $N_{r_i} = \{n_1, n_2, ...n_k, \forall n_i \in R\}$. The edges of the graph, which encapsulate the links between relay nodes and their neighbors, are captured by the set $E = \{e(r_i, n_j)|r_i \in R, n_j \in N_{r_i}\}$.

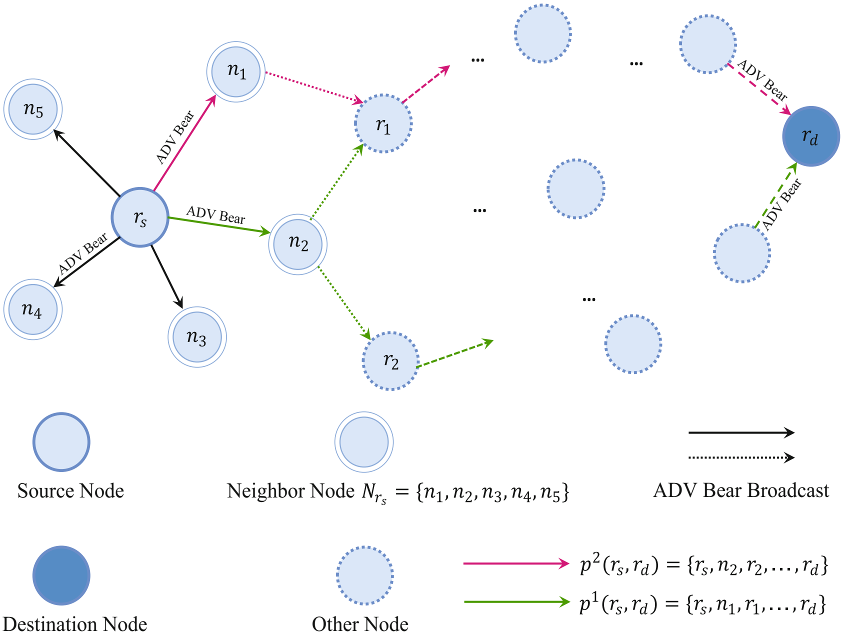


Fig. 2. Model of BLE mesh network routing.

The collection of all possible paths from a source node r_s to a destination node r_d is denoted as $P(r_s, r_d) = \{p^1(r_s, r_d), p^2(r_s, r_d), ..., p^q(r_s, r_d)\}$. Each specific path $p^i_{r_s, r_d}$ can be conceptualized as a sequence of relay nodes $p^i(r_s, r_d) = \{r_{s0}, r_{s1}, ..., r_{sk}, r_{s0} = r_s, r_{sk} = r_d, e(r_{si}, r_{s(i+1)}) \in E\}$ as illustrated in Fig. 2. Associated with each edge $e(r_{si}, r_{s(i+1)})$ are an energy cost $c(e(r_i, r_j))$ and a delay $d(e(r_i, r_j))$. Given a maximum permissible delay Δ_{delay} for the packet transmission from the source to the destination, we formulate the following constrained optimization problem:

$$\begin{gathered} \min_{p(r_s, r_d)\in P} \sum\nolimits_{e\in p(r_s, r_d)} c(e) \\ \text{s.t} \sum\nolimits_{e\in p(r_s, r_d)} d(e) \le \Delta_{\text{delay}} \end{gathered} \tag{1}$$

3 Proposed Approaches

3.1 Q-Learning-Based Routing Algorithm in BLE Mesh Network

The optimization problem under consideration exhibits considerable complexity, attributable to the exponential proliferation of path combinations as a function of network scale. This complexity is further exacerbated by the imposition of multiple constraints on the optimization objectives, such as energy efficiency and latency, as well as the inherently dynamic and stochastic nature of the network state. Moreover, the problem may manifest non-linear and potentially non-convex characteristics, rendering traditional optimization algorithms suboptimal for identifying globally optimal solutions. In light of these challenges, the Q-learning algorithm, renowned for its robust online learning capabilities and adaptability, emerges as an efficacious strategy for routing decisions within nodes.

First, the computational simplicity of the Q-learning algorithm renders it particularly amenable to implementation on resource-constrained devices, such as those prevalent in wireless sensor networks. Second, Q-learning, being a model-free reinforcement learning algorithm, possesses inherent adaptability. It learns an action-value function, denoted as $Q(s, a)$, through direct interactions with the environment. This function serves to estimate the expected long-term reward associated with executing a specific action a under a given state s. In light of the unique characteristics and constraints of BLE mesh network, the Q-learning algorithm's action-value function $Q(s, a)$ is adapted and updated according to the following equation:

$$Q(s,a) \leftarrow (1-\alpha) \cdot Q(s,a) + \alpha \cdot [f_r + \gamma \cdot \max_{a'} Q(s', a')] \tag{2}$$

In the Q-learning paradigm, two critical parameters are defined: α and γ, both of which are confined to the interval (0,1]. The learning rate, denoted by α, quantifies the proportion by which the newly acquired Q-value should influence the existing Q-value. Concurrently, the discount factor γ serves to weigh the significance of prospective rewards in the decision-making process. In the specialized context of BLE mesh network augmented with Q-learning algorithms, the network architecture can be abstracted as a dynamic environment. Within this environment, the state-space of a node is intricately linked to individual data packets. Specifically, when a node r_i is tasked with processing a data packet originating from a source node r_s and destined for r_d the state of r_i is explicitly defined as r_d, signifying its current engagement in routing a packet towards r_d. In this scenario, the action set A for r_i is constituted by its neighboring nodes, formally represented as $A = N_{r_i}$. Upon successful transmission of the data packet to a neighboring node n_j, the node r_i evaluates the reward associated with this particular action using a predefined reward function $f(r_i, n_j)$. Subsequently, the Q-value for r_i is updated in accordance with the Q-learning update formula, which, when tailored to address the nuances of this routing problem, is articulated as follows:

$$Q(r_d, n_j) \leftarrow (1-\alpha) \cdot Q(r_d, n_j) + \alpha \cdot [f(r_i, n_j) + \gamma \cdot \max_{a \in N_{n_j}} Q(n_j, a)] \tag{3}$$

In this equation$f(r_i, n_j)$ represents the reward function for the action of transmitting a packet from node r_i to n_j. This reward function incorporates both the hop count to the destination and the residual energy of the node, and is formulated as:

$$f(r_i, n_j) = -\beta \cdot h + (1 - \beta) \cdot c(n_j) \tag{4}$$

$$h = \begin{cases} 0, \ n_j = r_d \\ 1, \ n_j \neq r_d \end{cases} \tag{5}$$

$$c(n_j) = \frac{E_{n_j}}{\overline{E}_{N_{\mathrm{r}_i}}} \tag{6}$$

Here, h serves as a constant penalty, incentivizing nodes to minimize the hop count, thereby reducing latency. However, focusing solely on the shortest path may compromise network longevity. To address this, the reward function also incorporates the residual energy of nodes, normalized by the average energy of their neighbors $\overline{E}_{N_{r_i}}$, to achieve a balanced network load. The parameter β, where $\beta \in (0,1]$, is introduced to fine-tune the trade-off between energy efficiency and latency.

The proposed Q-learning-based routing algorithm retains the managed-flood's message caching and TTL mechanisms. It operates in two distinct phases, as delineated in the subsequent Fig. 3.

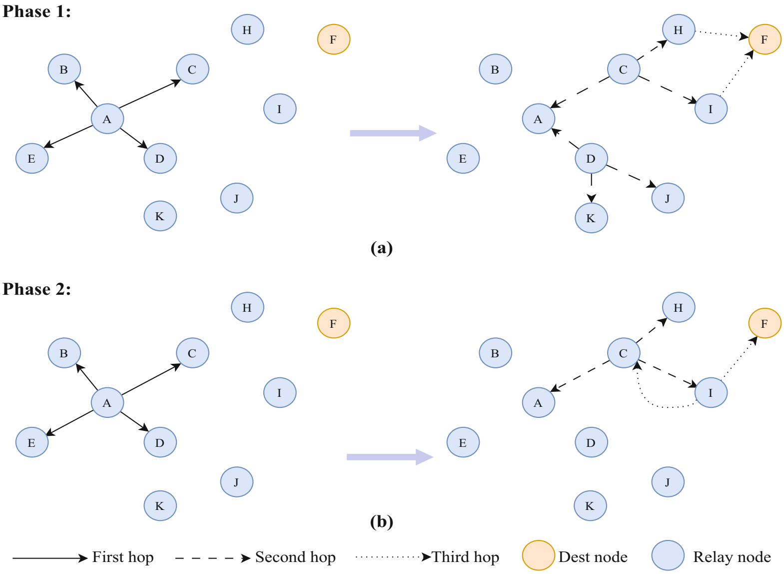


Fig. 3. (a) The initial broadcast-based Q-learning for rapid Q-value convergence among neighboring nodes. (b) The targeted packet forwarding based on the updated Q-table, facilitated by a caching mechanism to prevent redundant re-broadcasts.

Phase 1: The initial stage of the algorithm, termed as the preliminary phase, is characterized by a broadcast-oriented approach to data dissemination, particularly when a node is tasked with managing the initial set of data packets directed towards a specific destination node n_{r_d}. This phase is regulated by a threshold parameter,M_{r_d}, which determines the number of packets, m, a node handles before transitioning to the subsequent phase. Specifically, when $m < M_{r_d}$ the node employs a broadcast bearer, ensuring that all neighboring nodes within its vicinity receive the packet by scanning the designated broadcast channels. This mechanism, which essentially constitutes an exhaustive execution of the available action set, instigates an immediate update of all pertinent $Q(r_d, n_j)$, where $n_j \in N_{r_i}$ This broadcast-based Q-learning mechanism serves a dual purpose: it not only guarantees the reliable transmission of packets to the intended destination but also expedites the convergence of Q-values across the neighboring nodes by leveraging the widespread dissemination of packet information during the initial learning phase.

Phase 2: Upon concluding the swift learning phase, nodes refer to their Q-table to designate the next-hop neighbor for packet forwarding. This targeted forwarding ensures that even if other neighbors receive the packet, they will not re-broadcast it due to the existing caching mechanism. Concurrently, throughout the routing procedure, the Q-value table of nodes is adaptively updated utilizing a designated reward function, which strategically balances energy expenditure amongst neighboring nodes, ensuring an equitable distribution of energy consumption and mitigating premature node depletion within the network. The detailed procedure of the proposed algorithm is outlined in Algorithm 1.

Algorithm 1 The proposed Q-learning-Based routing approach

```
Input: neighbor nodes $N_{r_i}$
Output: next hop node
While $E_{r_i} > 0$ do
    if packet already cached or TTL < 2 do
        Drop packet
    else
        if the count of handle packet to destination node $m < M_{r_d}$ do
            for each $n_j \in N_{r_i}$ do
                Calculate $c(n_j)$ based on $E_{n_j}, \overline{E}_{N_{r_i}}$
                Calculate $f(r_i, n_j)$
                Node $r_i$ update $Q(r_d, n_j)$
            end
            return action $a = N_{r_i}$
        else
            Select $n_j$ with $max\ Q(r_d, n_j)$
            Calculate $c(n_j)$ based on $E_{n_j}, \overline{E}_{N_{r_i}}$
            Calculate $f(r_i, n_j)$
            Update $Q(r_d, n_j)$
            return action $a = n_j$
        end if
    end if
end
```

3.2 Duty Cycle Scanning Mechanism in BLE Mesh Network

BLE devices function within the 40 channels of the 2.4 GHz ISM frequency band. Specifically, nodes utilizing the broadcast bearer layer for communication employ three dedicated BLE broadcast channels—namely, channels 37, 38, and 39. According to BLE specifications [12], the operational states of BLE devices can be classified into initialization, standby, scanning, broadcasting, and connection. In BLE mesh network, relay devices operating on the broadcast bearer layer are perpetually in a connectionless state. This study is primarily concerned with the scanning, broadcasting, and standby states, which are most pertinent to the functioning of BLE mesh network.

When a node in the BLE mesh network necessitates data transmission, it disseminates the data via the broadcast channel. The managed-flood transmission mechanism in the BLE mesh Profile stipulates a 100% scanning duty cycle for relay nodes to ensure that devices exclusively supporting broadcast bearers can receive data in a connectionless milieu. This continuous scanning significantly escalates the energy expenditure of the nodes. To ameliorate this, we introduce a duty cycle scanning and repetitive broadcasting communication mechanism. This mechanism aims to optimize the time nodes spend in the Standby state, thereby prolonging their operational lifespan, while concurrently ensuring the reliable transmission of messages.

In the proposed mechanism, when a node has a message to transmit, it repetitively broadcasts the data packet thrice, transitioning to a Standby state for a duration t_{st} after each broadcast cycle. Adjacent nodes, to ensure the successful reception of at least one of the three broadcast messages, oscillate between Scanning and Standby states. The duration of the Scanning state is t_{sc} and the Standby duration t_{st} is synchronized with that of the broadcasting node. This ensures the reliable inter-node transmission of data packets in asynchronous scenarios, as illustrated in Fig. 4.

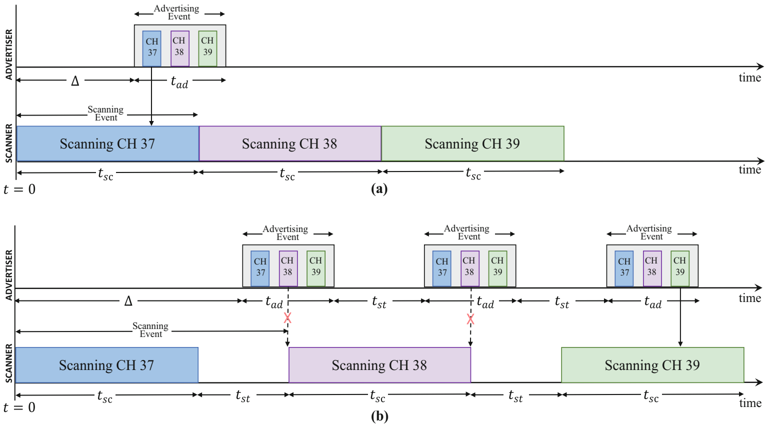


Fig. 4. Comparison of packet transmission mechanisms in BLE mesh network. (a) Managed-flood with 100% duty cycle scanning. (b) Proposed approach employing three broadcasts and sub-100% duty cycle scanning for enhanced packet reception.

To explore the optimal relationship between the scanning time t_{sc} and the standby time t_{st} this study employs the Monte Carlo simulation method. In the simulation, we assume $t_{sc} = k \cdot t_{st}$, where k is an adjustable coefficient. We further assume that the broadcasting node initiates the broadcasting of data packets at an arbitrary moment. The single-hop Packet Delivery Rate (PDR) is an instrumental metric, meticulously quantifying the efficacy of data packet transmissions over an immediate link, signifying the successful conveyance between nodes without intermediary interpositions. Our simulations revealed discernible fluctuations in the single-hop PDR across varied k coefficients. Preliminary findings suggest that, predicated on the scanning-broadcasting paradigm introduced herein, a scanning apparatus can adeptly intercept data packets dispatched by its broadcasting counterpart, given the condition that k exceeds 1, or equivalently, $t_{sc} > t_{st}$. The results are illustrated in the Fig. 5.

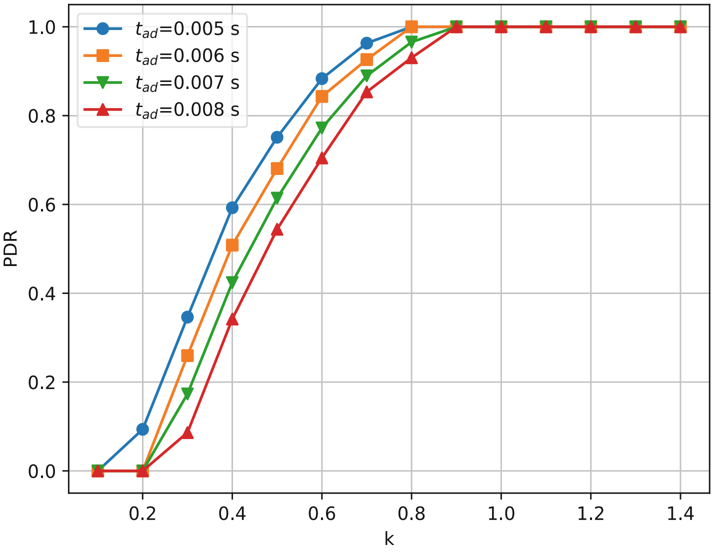


Fig. 5. Variations in single-hop PDR under different k values, where $t_{st} = 0.03$ s.

Taking into account channel congestion, unstable wireless environments, and potential channel conflicts that could inhibit successful packet reception, relay node r_i enters a backoff period t_b post-broadcast to monitor the broadcast channel. If a neighboring node n_j successfully receives and rebroadcasts the packet, r_i interprets this as an Acknowledgment (ACK) of successful data transmission. Conversely, if n_j fails to receive the packet, r_i reinitiates the broadcast cycle post-backoff. If the retransmission attempts exceed a predefined threshold, r_i designates an alternative neighboring node for packet forwarding and minimizes the Q-value for broadcasting to n_j. The retransmission mechanism is elucidated in Fig. 6.

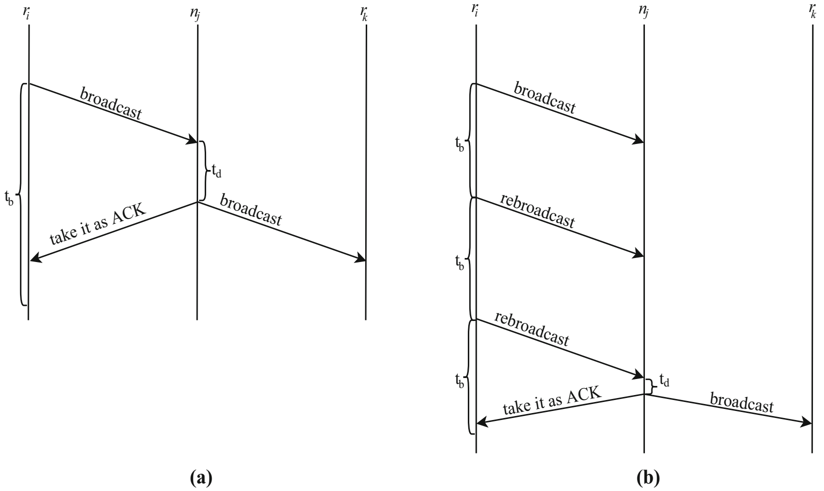


Fig. 6. (a) Successfully transfer packet. (b) Case of failure and retransmission.

4 The Result and Discussion

This section presents a comparative analysis of the simulation results for the proposed Q-learning-based BLE mesh network routing algorithm and the managed flooding mechanism. The simulations are conducted on the OMNeT++ platform, focusing on path overhead and average node lifetime.

4.1 Simulation Parameters

The parameters for network simulation are detailed in Table 1. Energy consumption data is sourced from [13]. To facilitate simulation, all intermediate physical-layer states during transitions between broadcasting, scanning, and standby modes are treated as standby states, and their energy consumption is averaged.

Table 1. Simulation parameters

Parameters	Value
t_{sc} Scanning duration	0.05 s
t_{st} Standby duration	0.03 s
t_{ad} Broadcasting duration	0.006 s
I_{sc} Current consumption on scanning	6.3 mA
I_{st} Current consumption on standby	1.9 mA

(*continued*)

Table 1. (*continued*)

Parameters	Value
I_{ad} Current consumption on broadcasting at 0 dBm	6.2 mA
N Numbers of BLE node	40
α, γ, β	0.8, 0.9, 0.7

4.2 Simulation Results

Average Path Overhead. Defined as the average number of times a data packet is forwarded from the source node to the destination node. Let M_i represent the number of times the i^{th} data packet is forwarded in the network, and m represent the number of data packets reaching the destination node. The formula is:

$$M = \frac{\sum_i M_i}{m} \tag{7}$$

In managed-flooding-based BLE mesh network, when a source node sends a data packet to a destination node, it broadcasts the packet to its neighboring nodes. These neighboring nodes, upon receiving the packet, further rebroadcast it. With a sufficiently large TTL, this ensures that all nodes in the network, including the destination node, receive the packet. The frequency of packet broadcasting is directly proportional to the number of nodes in the network. The current study introduces a Q-learning-based routing algorithm that takes full advantage of this broadcasting characteristic to implement a multi-action Q-learning process. This facilitates the rapid convergence of Q-values. During the packet forwarding phase, the algorithm employs these Q-values to designate, within the broadcast packet, which neighboring nodes should rebroadcast the packet. This targeted forwarding mechanism effectively minimizes packet redundancy within the network. The relationship between the number of packets generated in the network and the number of packets sent by the source node is depicted in the corresponding Fig. 7.

Node Lifetime. Correlated with the node's battery capacity $E_{battery}$ and average current $\overline{I}$. The average current can be calculated as the ratio of the total energy consumed during the simulation to the simulation time. The formula is:

$$\begin{gathered} T_{lifetime} = \frac{E_{battery}}{\overline{I}} \\ \overline{I} = \frac{I_{st} \cdot T_{st} + I_{sc} \cdot T_{sc} + I_{ad} \cdot T_{ad}}{T_{sc} + T_{sc} + T_{ad}} \end{gathered} \tag{8}$$

In addition to reducing packet redundancy and balancing energy consumption, the proposed broadcasting-scanning mechanism allows nodes to enter standby mode more frequently, thereby reducing energy consumption and extending node lifetime by at least 30%, as shown in Fig. 8.

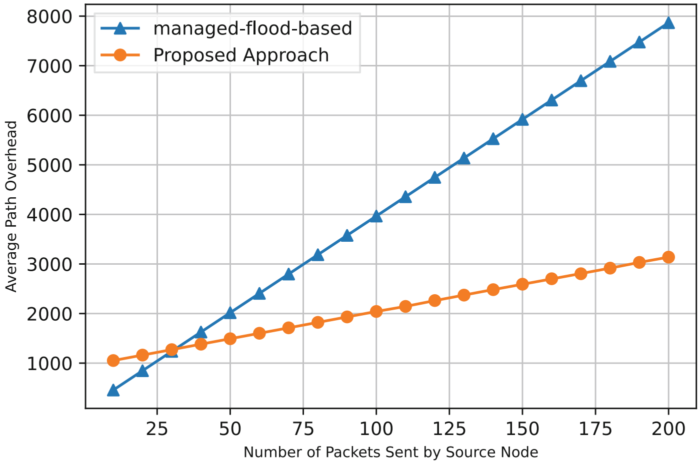


Fig. 7. The relationship between the number of data packets sent by the source node and the average path overhead in the network.

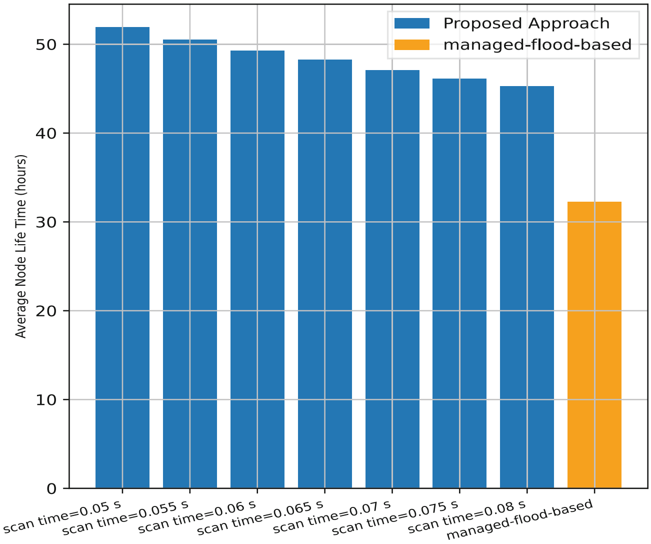


Fig. 8. Comparison of node average lifetime: the proposed approach with variable scanning durations vs. managed-flood-based.

5 Conclusion

This paper proposes a novel routing approach for BLE mesh network. The approach combines Q-learning algorithms with a duty cycle scanning mechanism and builds upon the existing managed flooding techniques. The algorithm incorporates both the hop count to the destination node and the residual energy of nodes into the reward function. This enables nodes to determine the next relay node based on their Q-values, thereby achieving a more balanced energy consumption across the network. To maximize node lifetime, we introduce a scanning-broadcasting mechanism that not only ensures reliable data packet transmission in asynchronous states but also allows nodes to enter a low-power standby mode as much as possible, thereby reducing their energy consumption. Simulation results indicate that our proposed method outperforms managed flooding in terms of path overhead and extends the lifetime of network nodes.

References

1. Mesh Profile. https://www.bluetooth.com/specifications/specs/mesh-profile-1-0-1/. Accessed 29 Aug 2023
2. Perez-Diaz-de-Cerio, D., Valenzuela, J.L., Garcia-Lozano, M., Hernández-Solana, Á., Valdovinos, A.: BMADS: BLE mesh asynchronous dynamic scanning. IEEE Internet Things J. **8**, 2558–2573 (2021)
3. Ng, P.C., She, J.: a novel overlay mesh with Bluetooth low energy network. In: 2019 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–6 (2019)
4. Ng, P.C., She, J., Spachos, P.: Energy-efficient overlay protocol for BLE beacon-based mesh network. IEEE Trans. Mob. Comput. **22**, 1 (2021)
5. Hansen, E.A.J., Nielsen, M.H., Serup, D.E., Williams, R.J., Madsen, T.K., Abildgren, R.: On relay selection approaches in Bluetooth mesh networks. In: 2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), pp. 1–5 (2018)
6. Wu, J., Li, H.: On calculating connected dominating set for efficient routing in ad hoc wireless networks. In: Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, pp. 7–14. ACM, Seattle Washington USA (1999)
7. Dhawan, A., Scoville, N., Tanco, M.: A Distributed greedy algorithm for constructing connected dominating sets in wireless sensor networks. In: International Conference on Sensor Networks (SENSORNETS 2014), pp. 181–187 (2014)
8. Kuhn, F., Wattenhofer, R., Zhang, Y., Zollinger, A.: Geometric ad-hoc routing: of theory and practice. In: Proceedings of the Twenty-Second Annual Symposium on Principles of Distributed Computing, pp. 63–72. Association for Computing Machinery, New York, NY, USA (2003)
9. Boyan, J., Littman, M.: Packet Routing in dynamically changing networks: a reinforcement learning approach. In: Advances in Neural Information Processing Systems. Morgan-Kaufmann (1993)
10. Hu, T., Fei, Y.: QELAR: A machine-learning-based adaptive routing protocol for energy-efficient and lifetime-extended underwater sensor networks. IEEE Trans. Mob. Comput. **9**, 796–809 (2010)
11. Su, X., Ren, Y., Cai, Z., Liang, Y., Guo, L.: A Q-learning based routing approach for energy efficient information transmission in wireless sensor network. IEEE Trans. Netw. Serv. Manag. **20**, 1 (2022)
12. Core Specification. Bluetooth® Technology Website. https://www.bluetooth.com/specifications/specs/core-specification-5-0/. Accessed 30 Aug 2023
13. Online Power Profiler for Bluetooth LE - opp - Online Power Profiler - Nordic DevZone. https://devzone.nordicsemi.com/power/w/opp/2/online-power-profiler-for-bluetooth-le. Accessed 31 Aug 2023