An Efficient and Scalable Reinforcement Learning-Driven Intelligent Resource Management and Secure Framework for LoRaWAN

Shaista Tarannum[1], Usha S.M[2]
Department of Electronics and Communication Engineering, JSS Academy of Technical Education, Bengaluru, India[1]
Visvesvaraya Technological University, Belagavi-590018, Karnataka, India[2]

***Abstract*—This study proposes a Q-learning-based adaptive duty cycle scheduling algorithm for LoRaWAN in a smart city eco-system to enhance the energy efficiency, reduce transmission delay, and handle dynamic traffic conditions. Additionally, it also incorporates an intelligent and efficient channel utilization scheme for LoRaWAN-enabled IoT networks and also integrates a lightweight security strategy at the edge (gateways), making it suitable for low-power, low-computation LoRaWAN environments. In this adaptive and intelligent LoRaWAN framework Q-learning agent dynamically selects various transmission actions based on the contextual states, including buffer size, energy levels, and channel conditions, which optimizes energy efficiency and also enhances the reliability of data transmission in LoRaWAN. The light-weight intrusion detection mechanism also filters suspicious packets using trust scores and payload analysis to ensure secure data delivery and adaptive, scalable, and proactive protection against several prevalent threats in LoRaWAN-driven IoT. It also incorporates a channel-aware scheduling to avoid congestion and improve overall transmission performance. Experimental outcome further confirms improvement over throughput, delay, bandwidth utilization, energy conservation, and resilience against malicious or faulty transmissions, demonstrating the framework's ability to optimize the resource allocation performance while balancing the above metrics adaptively.**

***Keywords*—LoRa; LoRaWAN; Q-learning; adaptive duty cycle; channel scheduling; energy efficiency; intrusion detection; trust score; resource management; IoT security**

I. Introduction

The rapid advancement of the Internet of Things (IoT) has created a demand for low-power, long-range wireless communication technologies. IoT involves connecting a wide range of end-devices, which are battery-powered sensor nodes, to the Internet and also enabling communication and data exchange between them [1] [2]. Therefore, the need arises to design the power usage profile carefully in order to extend the battery's lifetime. Also, the communication range needs to go from several hundred meters up to several kilometers as IoT end devices are distributed over a large area of operation. Considering all the aforementioned characteristics, this can be only realized by using the low power wide area network (LPWAN) technologies as LPWAN technologies can support resource management, throughput, and delay constraints in IoT [3]. There exist several LPWAN technologies present in the market, such as SigFox [4], Narrow Band (NB)-IoT [5], or Long-Range Wide Area Networks (LoRaWAN) [6] and Long-Term Evolution for Machines (LTE-M) [7]. The maximum data rate in SigFox is ~100 bps, whereas in NB-IoT it is ~250 kbps. On the other hand, in LoRaWAN, it is approximately ~50 kbps, and in the case of LTE-M, the maximum data rate is approximately ~1 Mbps. LoRaWAN [6] is among the leading LPWAN technologies as it offers the possibility for private network deployments and easy integration with a number of worldwide network platforms and has the ability to provide long-range communication with low power consumption. Due to this and its open access specification, LoRaWAN have gained significant attention from academia and industries for IoT [8] [9]. The LoRa physical layer has been patented by Semtech in the year 2014 [10]. LoRa is a radio frequency (RF) modulation technology that defines the physical layer features for long-range communications. However, the LoRaWAN medium access control (MAC) protocol is an open-source protocol standardized by the LoRa Alliance that runs on the top of LoRa physical layer.

LoRaWAN is also popular in smart city applications as it provides cost-effective, scalable, and power-efficient solutions for connecting thousands of widely distributed and low-data-rate devices. It supports long-range communication while covering ~2-15 km in rural areas and 1-5 km in urban settings. The end devices in LoRaWAN can last upto 5 to 10 years on battery, that reduces the need for frequent maintenance. It also offers low-cost unlicensed spectrum (ISM) bands, whereas LoRaWAN gateways are also affordable and can support thousands of devices. It also supports municipalities to deploy and control their own infrastructures in smart city applications. LoRaWAN also supports flexible network architecture with public, private, and hybrid networks, whereas designed for massive IoT deployments. Also, it uses AES-128 encryption schemes at the network and application layers while ensuring end-to-end security for data transmitted from sensors to applications [9] [10]. LoRaWAN is a popular communication protocol designed for low-power wide-area IoT deployments. While it excels in range and energy efficiency for small data transfers but it faces several inherent limitations and challenges due to its design trade-offs [20] [21]. The challenges arise due to low data rates, inefficient resource allocation, energy consumption, and network congestion problems [22]. It has also been observed that traditional fixed-duty cycle mechanisms suffer from suboptimal bandwidth utilization, increased transmission delays, and energy wastage, especially

in dynamic environments where real-time data transmission is crucial [23]. The root cause for the low data rate transmissions is chrip spread spectrum modulation, which prioritizes robustness and range over speed, and also spreading factor (SF) that results in long transmission time and poor latency [24]. LoRaWAN uses a pure ALOHA channel access method that also increases the chances of retransmissions. collisions and unfair resource usage. The frequent transmissions (TX), collisions, and high SF factors result in reduced battery life. LoRaWAN gateways operate in licensed ISM bands (868 MHz in EU, 915 MHz in US) with strict duty cycle limitations. As the network of IoT end devices grows with limited channel availability, that also results in frequent packet drops, low QoS, and scalability problems [25]. It is also observed that the LoRaWAN protocol suffers from security vulnerabilities to jamming and intrusions. No such in-built mechanism is found in LoRaWAN for intrusion detection or anomaly classification [15].

Machine Learning (ML) is a popularly growing field with many applications, including wireless communications [26] [27]. There are various studies which claim that ML could be used to improve the performance, efficiency, and security of LoRaWAN [28]. ML algorithms empower LoRaWAN networks to dynamically allocate resources, predict network traffic, mitigate interference, and optimize energy consumption, thereby enhancing network capacity, reliability, and battery life. With ML-driven insights, operators can proactively plan network expansions and ensure better quality of service (QoS), and also achieve self-optimizing networks that autonomously adapt to changing conditions. However, popular supervised learning approaches such as SVM, Decision Trees, Random Forest, and Logistic Regression models rely on a large amount of labelled data, which is often scarce and expensive in LoRaWAN due to limited sensing and reporting capability [29]. Also, models trained on static datasets may fail to generalize in dynamic, noisy environments with high traffic variability. Traditional supervised approaches also do not adapt well to real-time changes in topology or energy levels. These models also suffer from scalability issues and security bias limitations. It has been also observed that existing Deep Learning (DL) models, such as CNN, LSTM, DNN, auto encoders, require significant resources for training and interference which is not practical for low-powered LoRa nodes. DL models heavily rely on large datasets which is often unavailable in LoRaWAN due to low data rates and sparse feedback. Also, these DL models suffer from latency issues, overfitting problems, black-box nature, and deployment complexities [29] [30]. It has been also observed that the existing unsupervised ML techniques suffer from poor contextual awareness, false positives, and parameter tuning issues in large-scale LoRaWAN deployments.

The proposed system, therefore, aims to optimize the throughput, minimize transmission delay, and preserve energy while meeting the criteria for secure transmission, thereby enhancing the overall performance of LoRaWAN. However, the proposed work also finds the popularity of RL strategies and their scope towards improving the performance of LoRaWAN and further formulates a unique form of light-weight computational and analytical framework for Q-learning based adaptive duty cycle management in LoRaWAN, which is also further integrated with a security modeling. Here, the framework aims to optimize the LoRaWAN resources in dynamic conditions and also offers resiliency against different forms of adversaries in LoRaWAN, which affects the energy and QoS performance in LoRaWAN.

The key contribution of this work is listed as follows:

- Unlike traditional fixed duty cycling and static scheduling strategies, the proposed work employs Q-learning to dynamically adapt transmission decisions based on real-time network conditions and also significantly improves energy efficiency and throughput. Here Q-learning based adaptive duty-cycling dynamically adjusts transmission slots considering buffer, energy, channel, and duty cycle constraints.
- In contrast to the generic ML models that require extensive training data and centralized models, the proposed approach offers a light-weight and model-free reinforcement learning strategy that enables distributed and online learning at individual IoT nodes in LoRaWAN.
- The integration of trust-based intrusion detection modeling enhances the security by proactively filtering malicious data packets, a feature that is missing in existing standard ML approaches for resource management in LoRaWAN that focus solely on performance optimization without considering security. The framework also can be extended to incorporate trust scores from gateway-based intrusion detection, allowing the RL agent to make security-aware scheduling decision without compromising resource optimization.
- It offers an effective RL-driven approach for selecting transmission strategies that adaptively optimizes throughput, bandwidth utilization in response to dynamic traffic patterns. The proposed work shows superior performance in terms of delay, energy conservation, and secure transmission under variable traffic and channel conditions compared to popular baseline schemes.
- Unlike existing RL-based LoRaWAN studies, our framework uniquely integrates a Q-learning-driven adaptive duty cycle with trust-aware security with trust-aware security feedback. It combines Q-learning driven adaptive duty cycle for efficient resource management with a trust-aware security module. While the Q-learning agent optimizes transmission slots and power levels, the security layer independently evaluates trust scores from gateway-based intrusion detection that ensure reliable and secure data transmission alongside optimized network performance.
- In future work, the Q-learning agent could be extended to incorporate trust scores from gateway-based intrusion detection, which will enable joint resource management and security-aware decision making.

The aforementioned contributions are presented in a structured manner. The organization of the study is as follows: Section II reviews the existing methodologies, while the identified limitations are outlined in Section III. Section IV describes the system model, followed by the discussion of results in Section V. Finally, Section VI concludes the study with a summary of key findings.

II. Review of Literature

There exist various forms of related research studies that have also focused on improving the performance of LoRaWAN using ML approaches. The authors in [11] have proposed a load-balancing method for dense IoT networks such as smart city scenarios. The authors basically performed training of various ML techniques such as Multiple Linear Regression (MLR), Gaussian Naïve Bayes (GNB), Linear Discriminant Analysis (LDA), Decision Tree (DT), Random Forest (RF), and few others. These classifiers are applied to an urban IoT network, where the simulation results showed performance improvement over packet success ratio (PSR), and the framework also offered optimized energy consumption in the LoRaWAN network. In [12], the authors have introduced an SF allocation scheme considering SVM and DT that aims to resolve the collision issue in the LoRaWAN network. The training dataset was generated using simulator for LoRa SimLoRaSF, and a custom simulator was also designed for LoRaWAN using Python. The authors in [13] also emphasizes toward resource classification problem for static EDs in LoRaWAN using various ML techniques. Such as RF, SVM, logistic regression (LR), K-Nearest Neighbour (KNN), and few more others. The experimental outcome shows that the RF method accomplished the highest accuracy of 92% compared to other ML techniques. It has been observed that various DL methods were applied to improve the performance of LoRaWAN by optimizing resource parameters, predicting network traffic, mitigating inter-intra interferences, and optimizing energy constraints.

An extended Kalman Filter-based LSTM method based on regression is proposed for predicting collisions in the LoRaWAN network, which is introduced in the study of [14]. As a collision in the LoRa network is directly linked with the SF, hence SF has not been considered for adaptive configuration. As a result, the presented LSTM approach leads to underperformance when applied in a dynamic LoRaWAN network. The study in [15] proposed DeepLoRa, which is an environment-aware path loss model. On the other hand, in [16], a DL method is proposed for managing the transmission interval of IoT devices in LoRa networks by utilizing Intel Berkeley Research Lab Data. The paper [17] proposed a DL method for joint collision detection and resource management. Here, the presented work considers two DL methods: fully connected neural networks (FCNNs) for collision detection and CNN for SF management. The experimental results show improved prediction performance and energy consumption when compared with traditional ML methods. The paper [18] proposed an RL-based approach for optimizing and updating LoRa communication parameters. The authors utilized the RL method to derive optimal disseminating policies by aiming to maximize the accumulated average node throughput. The authors claim that their approach to the LoRaWAN has given a remarkable increase in the accumulated average per-node throughput of 147%.

The authors in [19] presented a novel method for resource allocation in LoRaWAN networks. Here, the method used Q-learning strategy of RL to learn the optimal resource allocation policy for each ED in the network. In the presented method, GW acts as an agent of Q-learning, where the Q-reward is based on the weighted sum of the number of successfully received packets in the proposed method. The Q-learning method was evaluated using simulation, and the outcome shows that the proposed method achieves PSR by ~20% compared to a random resource allocation scheme.

III. Research Problem

LoRaWAN networks significantly face challenges in balancing energy efficiency, reliable data transmission, and security against malicious activities. Traditional static duty cycle mechanism in LoRaWAN leads to inefficient bandwidth utilization and energy wastage, especially under dynamic traffic and channel conditions. Furthermore, LoRaWAN's lightweight nature makes it susceptible to various security threats, including spoofing and packet injection. The research addresses the need for an adaptive secure scheduling mechanism by formulating a Q-learning-based framework that can dynamically adjust the transmission duty cycle based on real-time state conditions. However, the challenge also arises to design a light-weight trust-aware intrusion detection system to block malicious packets in LoRaWAN. Despite the popularity, existing AI/ML-based optimization techniques often fail to adapt to the variability in network traffic, dynamic occupancy patterns, and energy constraints in large-scale LoRaWAN deployments.

IV. Research Method

The proposed work, therefore, aims to develop a novel, highly efficient, and scalable AI/ML-driven adaptive resource management and secure framework for LoRaWAN in a smart city eco-system where the resource management strategy is well-capable of optimizing the duty cycle allocation, bandwidth utilization, and energy-efficient transmission while maintaining high data throughput and minimal latency. The proposed framework of LoRaWAN operates in the MAC layer that effectively defines how LoRa devices communicate with the gateway and network servers. In the proposed work, the adaptive LoRaWAN framework (Fig. 1) is structured into three primary components, which are Q-learning-based scheduling, channel-aware management, and lightweight edge security. Here, Fig. 1 also shows a typical LoRa network deployment scenario. Before delving into algorithmic details, the section outlines the operational flow of these core components in LoRaWAN.

A. LoRaWAN-Based IoT Deployment

Let, $D = \{d_1, d_2, \ldots, d_n\} \in \mathbb{Z}^+$ be the set of n LoRa IoT devices in a LoRaWAN environment. The LoRa-based IoT network also consists of m LoRa gateways such as $G = \{g_1, g_2, \ldots, g_m\} \in \mathbb{Z}^+$. It is also assumed during the analytical modeling that the network should also consist of a central network server S. The total time steps for simulation is considered to be T. Therefore, each IoT device $d_i \in D$ is

located at position of $\mathrm{Loc}(d_i) \in \mathbb{R}^2$. And also, each gateway $g_j \in G$ has a coverage radius of R_j. The deployment assumption also considers that the devices and gateways are randomly distributed in a 2D space, and coverage check can be validated considering the following Eq. (1).

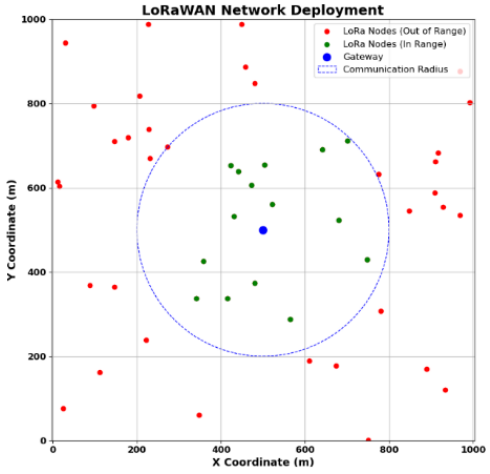
$$\left\|\mathrm{Loc}(d_i) - \mathrm{Loc}(g_j\right\| \leq R_j \quad (1)$$

At each time step of $t_i \in T$ each device d_i generates a data packet in the form of $P_i^{(t)} = \langle temp_i^{(t)}, hum_i^{(t)}, \tau_i^{(t)}, id_i \rangle$ where $temp_i^{(t)}$ represents the temperature readings, $hum_i^{(t)}$ represents the humidity readings, $\tau_i^{(t)}$ represents the current time stamp and id_i refers to the unique device ID. It also assumes that the packet $P_i^{(t)}$ is sent to a randomly selected gateway $g_j \in G$. Here it is assumed that IoT end devices periodically generate data packets, and that are transmitted to randomly selected gateways. The packet loss model also considers that each gateway g_j has a probability of packet loss of 0.05 and success of 0.95. The condition of data transmission is modeled using Eq. (2). If it is found that $X_{ij}^{(t)} = 1$, then the gateway forwards the packet to the server $P_i^{(t)} \rightarrow S$. The server further stores and processes the packets in Eq. (3).

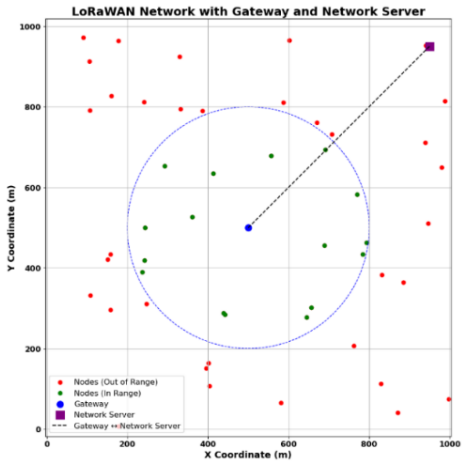
$$X_{ij}^{(t)} = \begin{cases} 1, & \text{if } g_j \text{ succesfully receives } P_i^{(t)} \\ 0 & \text{Otherwise} \end{cases} \quad (2)$$

$$\xi_s = \cup_{t=1}^{T} \left\{ P_i^{(t)} | X_{ij}^{(t)} = 1 \right\} \quad (3)$$

Here, in the above Eq. (3), ξ_s represents the cumulative set of successfully received packets over time from all IoT end devices in the LoRaWAN-enabled smart city eco-system. The proposed work emphasizes towards designing and developing a novel adaptive duty cycle slot allocation mechanism using Reinforcement Learning (RL) to enhance the performance of large-scale LoRaWAN in smart city eco-system. The following Fig. 1(a) shows the LoRaWAN network deployment scenario without a network server, and Fig. 1(b) shows the LoRaWAN network deployment scenario with the inclusion of a network server in the proposed work.



(a)



(b)

Fig. 1. LoRa network deployment in IoT smart city: a) LoRaWAN network with gateway placement, b) LoRaWAN with gateway and network server.

B. Improving LoRaWAN Performance Using Reinforcement Learning (RL) Algorithms

The proposed study emphasizes towards improving the performance of LoRaWAN via ML-based decision making for adaptive duty cycle slot allocation. The proposed work leverages RL techniques in LoRaWAN to intelligently manage the network resources and aims to maximize efficiency along with optimal delivery performance. Reinforcement Learning (RL) is a data-driven approach where it learns rules and policies from experience by interacting with the network and observing the results. RL is a dynamic approach that has the capability to adapt to the environmental changes. LoRaWAN networks are constantly changing owing to the dynamic scenarios and due to the underlying propagation environment. As a result, RL algorithms can be used to learn how to optimize the network parameters for these changes, ensuring that the network remains reliable and efficient. As RL is a scalable approach, it can be used to optimize large and complex networks [21]. Therefore, the proposed work realizes that in LoRaWAN, RL could be used to optimize resources by training agents for making decisions to maximize overall network efficiency and minimize interference, along with energy consumption. The proposed work in the first phase of design, therefore, introduces a novel and cost-effective adaptive duty cycle slot allocation using Q-learning modeling and also ensures dynamic slot assignment. In the second phase of design, it incorporates a scheme for bandwidth and efficiency optimization, followed by a security analysis in the third phase.

1) Adaptive duty cycle with RL in LoRaWAN: Duty cycle in LoRaWAN determines how often a device is capable of transmitting data packets. It has been observed that the static allocation paradigms often lead to congestion, collisions, or under-utilization of resources. Therefore, the proposed work introduces an approach of adaptive duty cycling using Q-learning to dynamically choose the best time slot based on traffic, battery, and buffer status, which in longer run reduces contention and also improves energy efficiency.

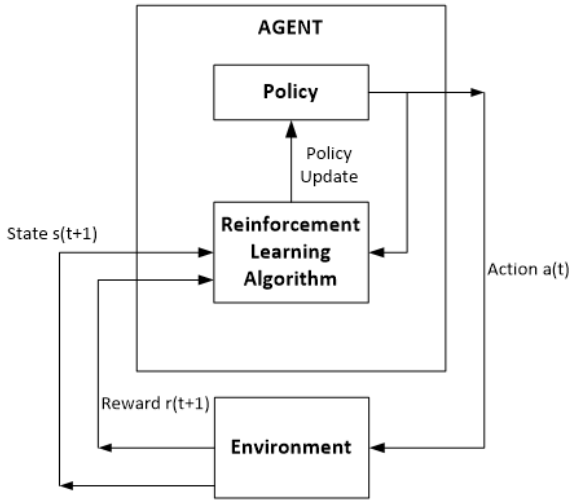


Fig. 2. Reinforcement learning in LoRaWAN.

2) State representation for Q-learning agent: In the proposed analytical modeling of LoRaWAN, each end device state in the context of IoT smart city is defined as a tuple of $s_t = \langle b_t, E_t, l_t, D_t, c_t \rangle$. Here, b_t refers to buffer size at time t, E_t implies the battery level at time t, whereas l_t implies time since last transmission. D_t on the other hand, implies current duty cycle slot and c_t represents channel status (i.e 0=idle, 1 = busy). Fig. 2 shows baseline RL concept which is adopted in the proposed LoRaWAN framework designing.

3) Action space definition in LoRaWAN: The proposed analytical strategy further defines the action space $A = \{TX_{high}, TX_{low}, wait\}$ as a set of high power transmission, which require high energy usage and lower delay denoted with TX_{high}, low power transmission, which require low energy usage but higher delay denoted with TX_{low}, and $wait$ implies skip transmission where the buffer grows and battery is conserved. The prime reason behind considering TX_{high} is that it is used for urgent data or poor channel conditions. It aims to minimize the delay and improve packet delivery probability (stronger signal) for urgent packets or unreliable channel conditions in LoaRaWAN. On the other hand, the action corresponds to TX_{low} imply when channel conditions are good or delay is tolerable for packet transmission. This action also conserves battery life while still sending data. Here, the action $wait$ refers to preserve battery or adhere to duty cycle limits. This action has got importance when the network is highly congested or data is not urgent. This approach not only conserve energy but also helps avoiding duty cycle violation. In the RL perspective the proposed work decides the action space to support light-weight decision model in resource constrained devices with three discrete actions. The learning agent learns a policy to select the best action at each time step depending on the above state space criteria highlighted in $s_t = \langle b_t, E_t, l_t, D_t, c_t \rangle$ to enhance the performance and resource management in LoRaWAN. In the proposed work Q-learning helps managing the resources and LoRaWAN traffic and enables adaptive traffic handling.

4) Q-learning in adaptive traffic handling: In the proposed approach of performance improvement of LoRaWAN the states encode the traffic contexts in the form of s_t. Here, the buffer size implies traffic backlog, which means how much data is waiting to be sent. On the other hand, E_t refers to the remaining battery level of end IoT devices. On the other hand, l_t helps controlling the latency factor. The factor of D_t is used as a duty cycle slot that measures the compliance constraint. These states help the learning agent understand the traffic load and network status accordingly RL mechanism effectively manages the resources. TX_{high} is learned when urgent traffic and high buffer build-up occur. On the other hand, TX_{low} is learned by the agent when traffic is moderate and conditions are favourable. The Q-learning learns to skip Tx for the action $wait$ and avoid penalty and transmit later when the condition improves. In the case of learning$wait$, the Q-learning model realizes that the traffic is low, that means there is no urgency to transmit, whereas transmission at that point might waste energy or duty cycle. $wait$ allows agents to accumulate more data and possibly send it together later. Q-learning learns that delaying when the buffer is small and doesn't incur a big penalty, and also may result in higher future rewards. The $wait$ is useful as it helps avoiding duty cycle violation, helps when the channel is busy ($c_t = 1$) and also useful in the case of battery E_t is critically low. The agent learns to wait for better conditions. Here, via Q-learning, the agent chooses the optimal actions to balance delay, energy usage, and duty cycle compliance, which also positively influence the throughput and data delivery outcome. The Q-Function updating is represented with Eq. (4).

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \right] \quad (4)$$

Here, α represents the learning rate, whereas γ implies the discount factor, and also r_t refers to the intermediate rewards at time t. Also, in the above expression, a' represents the next action, and further it leads to the Q-value for state action pair in $Q(s_t, a_t)$. While iteratively updating the above expression (4), the agent progressively learns when to transmit and at what power in LoRaWAN, and also learns when to wait to avoid congestion or conserve resources. It also adapts to dynamic traffic patterns in real-time. The agent evaluates b_t and if it is high, then it immediately sends in TX_{high} or TX_{low} depending upon the channel conditions. If E_t is low, then the agent prefers to conserve energy, so either it chooses or TX_{low} or $wait$. If l_t is high, then it avoids excessive delay and prefer any Tx. If D_t is found near max, then it avoids sending and prefers $wait$. Also, if $c_t = 1$ that implies the channel is busy and prefers $wait$. Here, $D_t \sim 1$ indicates that the end IoT node has used almost all of its allowed transmission time for the current regulatory duty cycle window. That's why it is safer to avoid transmitting until the cycle resets. In LoRaWAN, especially in the EU 868 MHz band is subject to strict regulatory duty cycle limits. If $D_t \sim 1$ in max and the learning agent still chooses to transmit that means it may get penalty, drop the packet and delay can occur for future transmissions. This results in low or negative reward, so over time the agent

learns that $wait$ is the optimal action, when $D_t \sim 1$. With approach the Q-table find which action yields the best long-term return and balances throughput, energy and reliability.

5) Reward function modeling: The design of reward function modeling encourages high rewards for timely transmission with low power. And also penalizes the agent for buffer overflows, delays, or duty violations. It is modeled using Eq. (5). Here, e = 5 for TX_{high} and e = 2 for TX_{low}:

$$r_t = \begin{cases} 10 - e & If\ TX\ succesful \\ -0.5 & if\ wait \\ -2 & if\ TX\ failed\ or\ battery\ low \end{cases} \quad (5)$$

The reward modeling in LoRaWAN encourages the learning agent with a high reward if fast delivery is needed. On the other hand, TX_{low} also gets a higher reward despite a higher delay, which promotes energy efficiency. Whereas TX_{high} is used when speed is critical but very less frequently. $wait$ is not rewarded, but the penalty is low, allowing it to be chosen in risky or sub-optimal states. The proposed Q-learning in LoRaWAN also penalizes risky transmissions, such as if Tx is fails or Tx when the battery is low. In such cases, the agent gets negative rewards, and further, this reward structure also pushes the agent to avoid such states.

6) LoRaWAN adaptive duty cycle mechanism: In the proposed work, the time is divided into 10 discrete duty cycle slots in the form of D_t. Here, each slot might represent a fixed duration, and the model cycles through them as time advances. The update rule for adaptive duty cycle is given using Eq. (6):

$$D_{t+1} = D_t + 1\ mod\ 10 \quad (6)$$

This equation shows that after each decision step (time step), the slot automatically advances to the next. In the proposed system content of LoRaWAN, the model behaviour simulates a rolling time window where duty cycle behaviour resets periodically, matching regulatory constraints (e.g. 1% per hour transmission time in EU868). In the proposed Q-learning-based adaptive duty cycling-based strategy, agents learn which slots are best for transmission based on success rate (i.e., whether Tx is successful), energy cost (TX_{high}, TX_{low}) , and channel conditions (e.g., fewer collisions). Through Q-learning, LoRaWAN builds state-action values $Q(s_t, a_t)$ that include D_t as a state component. It has to be noted that with this approach, the agent doesn't learn what to do, but it learns when to do it. The benefit of this adaptive duty cycling approach is that the agent learns to adapt to traffic patterns, interference, and energy constraints over time.

It has to be noted that the proposed framework operates in three phases: 1) Q-learning-based adaptive duty cycle assignment, 2) efficient channel scheduling, and 3) security-aware packet evaluation. Algorithm 1 to Algorithm 3 together represent the complete operational procedure in which the RL agent selects optimal transmission actions while considering energy levels, buffer status, and channel conditions, thereby achieving adaptive, secure, and efficient resource management in LoRaWAN. The RL agent is also capable of selecting optimal transmission actions while considering trust scores as well. The following algorithm shows Q-learning for adaptive duty cycle in LoRaWAN.

Algorithm 1: Q-learning for Adaptive Duty Cycle in LoRaWAN

Initialize $Q(s_t, a_t)$ arbitrarily for $\forall \langle s_t, a_t \rangle$ pairs
For each episode:
  Initialize environment (E) and state $s_t = \langle b_t, E_t, l_t, D_t, c_t \rangle$
  Repeat for each time step
    With probability ε select random action
    $a_t \in \{TX_{high}, TX_{low}, wait\}$
    Otherwise select $a_t = \arg\max_{a'} Q(s, a)$
    Execute action a, observe
      reward r_t,
      next state s_{t+1}.
      $D_{t+1} = D_t + 1 \bmod 10$ ← Adaptive Duty Cycle slot update
    Update Q(s,a) using:
      $Q(s_t, a_t) = Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \right]$
      $s \rightarrow s'$
      If terminal condition is met, break

The algorithm design and modeling ensure that the Q-learning agent adapts based on the dynamic state s_t. It also offers a duty cycle slot allocation where the reward strategy is encouraged for optimal time. Here, Q-learning automatically adapts policy using reward signals.

7) Optimization of bandwidth and data efficiency in LoRaWAN: It has to be noted that LoRaWAN in smart city eco-system suffers from collisions and idle listening as many IoT end devices share the same channel. Therefore, optimizing channel utilization (bandwidth) and efficient data scheduling can significantly improve the performance of LoRaWAN. If there are N number of end devices (ED) and P_t indicates the packets the successfully transmitted packets at time t ,whereas $\mathcal{U}_t$ represents the bandwidth utilization at time t. Then the bandwidth utilization can be modelled using Eq. (7) and Throughput T_t is measured using Eq. (8). The proposed work further evaluated the transmission delay using Eq. (9). Here, P_t represents the set of packets receiver at t and t_{rx}^p, t_{gen}^p represents the generation time and reception time of packets. The system also computes the battery energy level of IoT nodes or end devices in the form of E_i^t. Here, E_i^t represents the battery level of device i at step t.

$$\mathcal{U}_t = {}^{P_t}/_{N} \quad (7)$$

$$T_t = \sum_{i=1}^{N} \delta_i^t \text{ where } \delta_i^t = \begin{cases} 1 & if\ node\ i\ transsmission\ was\ succesful\ at\ t \\ 0 & otherwie \end{cases} \quad (8)$$

$$Delay_t = {}^{1}/_{|P_t|} \sum_{p \in P_t} (t_{rx}^p - t_{gen}^p) \quad (9)$$

In the proposed LoRaWAN framework Q-learning chooses TX_{high} and TX_{low} based on the congestion and energy. The bandwidth utilization is also tracked via $\mathcal{U}_t$ whereas the

proposed reward structure and simulation loop encourages maximized throughput. The proposed Algorithm 2 for efficient channel scheduling is represented as follows:

Algorithm 2: Efficient Channel Scheduling in LoRaWAN

For each time slot t:
  For each IoT node i:
    Estimate channel status c_t
    If $b_t > \phi$ and energy sufficient:
      Select TX_{high} and TX_{low} via Q-learning
    Else:
      Choose the $'wait'$ action
    Update packet success/failure
    Update metrics:
      Throughput, delay, bandwidth utilization

It has to be noted that the proposed work in LoRaWAN approximates an efficient channel scheduling (ECS) strategy by using a Q-learning agent to dynamically decide on transmission power or WAIT based on local buffer status, energy, duty cycle, and channel status. However, full-fledged ECS features like multi-channel assignment, real-time congestion estimation are marked as future extensions.

C. Security-Aware RL Approach in LoRaWAN

The extensive analysis on LoRaWAN shows that it is vulnerable to different forms of attacks, such as spoofing, flooding, and data injection. Here, in the proposed system, a security-aware RL strategy can detect malicious data patterns, it can also maintain trust scores, and drop packets based on suspicious behaviour. It also ensures secure routing and packet forwarding. The security modeling considers a functional strategy of malicious packet indicator $\phi(P_i^{(t)})$ which generates binary flags if packet $P_i \in s$ is spoofed or malicious, else it flags to 0. It can be represented using Eq. (10). If s_t denotes the set of received packets at time t. Then the intrusion detection rate is computed using Eq. (11):

$$\phi(P_i^{(t)}) = \begin{cases} 1 & if\ P_i^{(t)}\ is\ malicious \\ 0 & otherwise \end{cases} \quad (10)$$

$$I_t = {}^{1}\!/_{s_t} \sum_{P_i^{(t)} \in s_t} \phi(P_i^{(t)}) \quad (11)$$

The intrusion detection rate basically measures how many suspicious packets were found in the current time window. The value of I_t implies that the network is under attack or faulty devices are sensing data. In this proposed security-aware reinforcement learning approach, each IoT device is continuously monitored. And also, the system learns patterns of normal versus suspicious behavior. The devices that behave badly get low trust scores, and their packets are blocked. Here, the system scans the packets from all the IoT devices, and it incorporates a model to spot unusual behavior in a packet, such strange timing or values. The proposed work also implements a security filter in this adaptive resource management framework of LoRaWAN. Here, each gateway uses a rule-based intrusion detection function using Eq. (12). Here, T_d indicates the temperature reading, H_d indicates the humidity reading, whereas T_{max} = 100 degrees and H_{min} is ~5%. The trust score τ_i for IoT device i is evaluated using Eq. (13). The τ_i ranges between 0 and 1. Here 1 means fully trusted device, whereas 0 indicate highly suspicious device. When too many of a device's packets are flagged, then the trust score drops. The packet drop rate δ-drop is also estimated using Eq. (14). This mathematical model evaluates how many packets are rejected as the proposed framework could not trust the source device.

$$Detect(d) = \begin{cases} 1 & \text{if } T_d > T_{max}\ or\ H_d < H_{min} \\ 0 & otherwise \end{cases} \quad (12)$$

$$\tau_i = 1 - {}^{\text{Flagged } P_i^{(t)} \text{ from i}}\!/_{\text{Total } P_i^{(t)} \text{ from i}} \quad (13)$$

$$\delta_{drop} = {}^{P_i^{(t)} \text{ dropped due to low } \tau_i}\!/_{\text{Total received } P_i^{(t)}} \quad (14)$$

In the proposed work and security strategy, the model simulates malicious behavior in LoRaWAN while injecting fake data. It also simulates malicious IoT devices sending falsified or abnormal data, which represents active attacks like data injection. The proposed security filtering and black listing formulation is well capable of defending network flooding attacks or data corruption with fake sensor values. It also prevents the adversaries from resending previously valid packets. As the gateway/server could be extended to discard old or replayed packets by checking if the timestamp is within a valid range or not. If a node continuously refuses to transmit data, then the proposed security framework could be extended with a Q-learning agent that discourages excessive wait or non-transmitting states via negative rewards and mitigates routing blackhole by adapting duty cycles and penalizing inactivity in LoRaWAN. The system is also well capable of resisting Grayhole attack, Jamming attack, Sybil, and DoS intrusions in LoRaWAN. The next segment of the study further illustrates the results and discussion on the outcome obtained from simulating the above analytical algorithms through numerical computing and analysis.

Algorithm 3: Intrusion Detection and Secure Transmission in LoRaWAN

For each received $P_i^{(t)}$ from device i
  1. Extract key info: device_id, timestamp, data content (payload)
  2. Run anomaly or unusual behaviour detector on $P_i^{(t)}$
    - If $P_i^{(t)}$ is suspicious or trust_score $[i] < 0.6$
      Drop the packet
      Reduce trust_score$[i]$ by a penalty value.
      Blacklist ← Blacklist ∪ {Device$_i$}
    - Else
      Accept the $P_i^{(t)}$
      Increase trust_score $[i]$ by a reward value.

It has to be noted that in the proposed framework the security layer operates synergistically with the Q-learning agent, in which the trust score of each device is periodically updated considering packet consistency and transmission success rate. These trust updates in longer run will indirectly influence the agent's state vector and also enable adaptive scheduling decisions that prioritize reliable and trusted nodes within LoRaWAN. While the Q-learning agent primarily handles the scheduling, the trust mechanism ensures that

unreliable or malicious nodes are filtered that indirectly support more effective and reliable network operations.

V. Results and Discussion

The proposed adaptive duty cycle management framework using Q-learning for LoRaWAN is analytically modeled and scripted in Python 3.10. The simulations are carried out considering a Python-based environment using Jupyter IDE/Notebook. The simulation framework considers a custom discrete-event simulation modeling in Python for LoRaWAN. It also implements a custom implementation of Q-learning, considering visualization libraries, data handling libraries, and machine learning (ML) libraries. The security module also incorporates custom filtering and blacklist logic. The simulation system configuration considers an Intel Core i5 processor with 12 GB RAM and a Windows operating system. The custom Q-learning is CPU-light for conceptual modeling, so it doesn't require GPU processing.

The experimental evaluation considered a custom discrete-event simulation model developed in Python to emulate LoRaWAN communication under varying traffic and channel conditions. The proposed framework also includes realistic parameters such as bandwidth (125 kHz), transmission power levels consistent with LoRa Class-A devices. For comparison, the popular baseline schemes such as Static (fixed duty-cycle allocation), Random (stochastic transmission slot selection), and Round Robin (cyclic slot assignment) were implemented. All metrics, such as throughput (packets/sec) and energy (mJ), are normalized averages derived from the simulation experiments. The network topology configuration is shown in Table I.

TABLE I. Simulation Configuration

Parameters	Values
Number of IoT Devices	10-100
Number of LoRa gateways	1-2
Simulation Time Step	50
No. of Episodes	1000
Coverage Radius (per Gateway)	1000 meters
Device Transmission Slots	10 slot cyclic duty schedule
Random Packet Loss Probability	10-15%
Attack Injection Rate	10% (Fake Sensor Values)
Initial Battery Level	100 Joule
$\boldsymbol{TX_{high}}$ Energy Consumption	5 units
$\boldsymbol{TX_{low}}$ Energy Consumption	2 units
$\boldsymbol{wait}$ Energy Impact	0 (only incurs delay penalty)
Learning Rate α	0.1
Discount Factor $\boldsymbol{\gamma}$	0.9
Exploration Rate $\boldsymbol{\varepsilon}$	0.1

The proposed work observes the trends from the plots after implementing the strategy of adaptive duty cycle operation in LoRaWAN. The observed trends from Q-learning is illustrated in Fig. 3. Fig. 3(a) shows that trend in cumulative reward per episode is initially low, but improves and stabilizes over episodes. This indicates the Q-learning agent in the proposed system learns effective actions to maximize successful transmission and conserve energy while also avoids unnecessary transmissions as the encouragement with higher reward in proposed Q-learning leads to better decision policy.

The analysis of the buffer size also implies that the buffers remain within mid to low ranges, which clearly shows high variability as buffer size fluctuates significantly from episode to episode. The buffer size ranges from 0-4 packets [see Fig. 3(b)]. The high-frequency fluctuations across the episode range suggest that the buffer size is dynamic and frequently changing. The buffer size is rarely remains zero or maxim indicating the system is actively transmitting and receiving data. Values between 1 and 3 dominate suggest a moderately filled buffer across the learned period. The observed variability also indicates that the system adapts dynamically to changes in network load channel availability and energy constraints. Since learning policy involves exploration using ε-greedy policy hence fluctuations are expected as the agent explores different transmission strategies. Overtime although no clear smoothing is visible, the agent avoids extreme buffer overflows or starvation indicating policy convergence. The observed variation suggests a balance between energy and throughput.

Here, Fig. 4 indicates the number of successfully transmitted packets received by the gateway at each time step. Here, the throughput values fluctuate based on agent's learned behavior and environmental dynamics such as channel congestion or blacklist filtering. The Q-learning agent gradually improves its decision-making strategy between TX_{high} , TX_{low} and $wait$ to prioritize successful transmissions when the channel is idle and the battery is sufficient. This adaptive learning process prevents excessive collision or batter drain and ensure stable throughput over time. It also shows good responsiveness and adaptive behavior with increasing number of IoT End Devices (EDs).

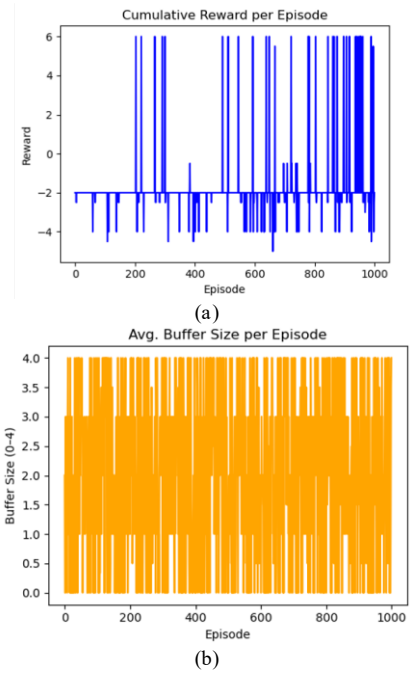


Fig. 3. Observed trends from Q-learning: a) analysis of cumulative rewards and b) average buffer size per episode.

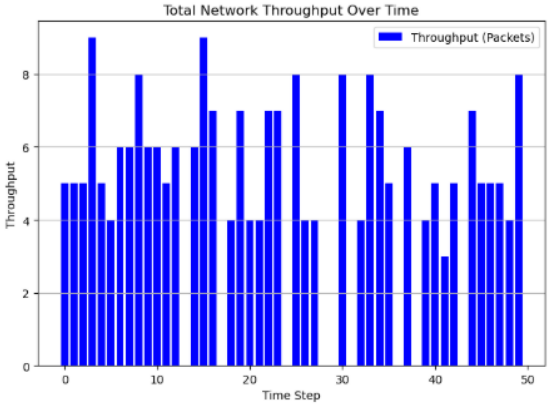


Fig. 4. Analysis of throughput in LoRaWAN.

The observed trends from Q-learning is illustrated in Fig. 5. The interpretation of the [Fig. 5(a)] plot shows ratio of packets successfully transmitted relative to total number of active devices at each time step. The proposed Q-learning based algorithm in LoRaWAN shows moderately efficient bandwidth utilization trend that clearly depicts the fact that the Q-agent balances the aggressive and conservative transmission attempts. Low utilization at certain intervals correlates with high interference (channel status busy) or conservative WAIT actions, which the agent learns to apply when risk is high or energy is low. This approach significantly helped reducing the congestion and packet loss. It offers ~65% utilization and the algorithm with this approach avoids overloading the network while maintains consistent data delivery and indicate balanced usage of resources. Fig. 5(b) captures the average battery level of all IoT devices over simulation steps. It highlights decreasing but controlled energy curve that signifies an energy-aware strategy in LoRaWAN. It also depicts that learning agent adaptively decide and favor low power transmission or defer transmission when essential as guided by the Q-learning optimal policy. The gradual descent also reflects sustainable energy usage rather than rapid depletion which verify the framework's suitability for long term operation in large-scale and constrained LoRaWAN.

The comparison of throughput for proposed Q-learning in (Fig. 6) shows that it achieves highest throughput in initial steps in LoRaWAN. Slowly it starts adapting to the environment dynamics and select transmission times that maximizes rewards. This leads to higher and smarter throughput over time. However, in random policy inconsistency is observed in the throughput outcome as it occasionally performs well by chance but lacks reliability and wastes resources due to frequent collisions or poor timing. However Round Robin offers stable but low throughput as it underutilizes idle slots and cannot adapt to traffic or channel conditions. Static duty cycle paradigms are fragile in dynamic environments and result poor reliability and wasted capacity. The Q-learning approach also occasionally yields zero throughput at specific time steps these reflect intelligent decision making such as avoiding unfavorable transmission slots or conserving energy. Here, Q-learning adapts its policy to maximize long-term throughput leading to overall superior performance despite transient dips in LoRaWAN.

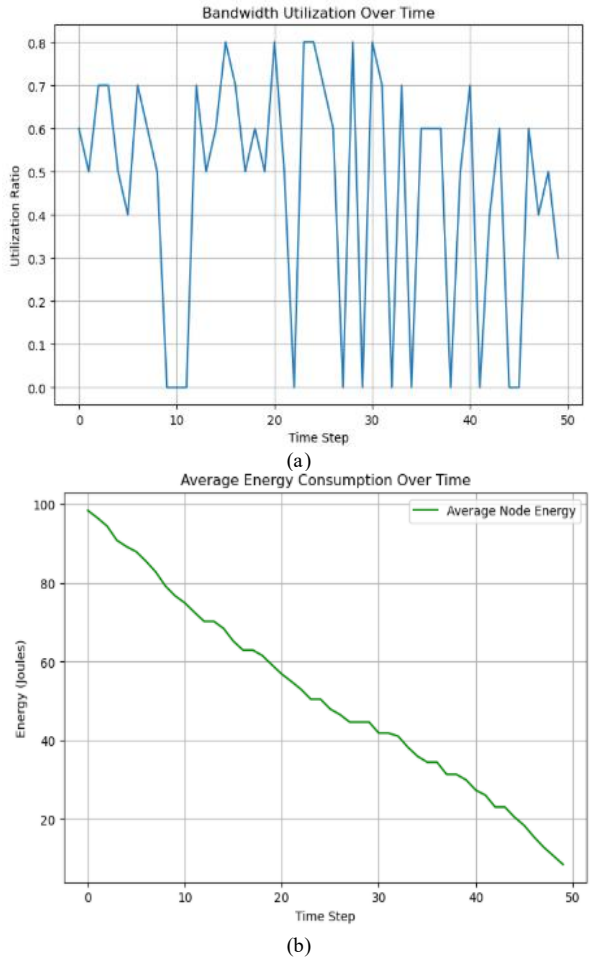


Fig. 5. Observed trends from Q-learning: a) analysis of bandwidth utilization ratio, and b) average energy consumption over time.

The delay outcome is found highly stable in the case of Q-learning strategy, as it learns from the reward feedback to choose actions (transmission slots) that minimizes delay. After brief learning phase, it converges to an efficient schedule with minimal delay. However, other methods such as Random, Round Robin and Static either transmit blindly or without adaption to repeated conflicts therefore, suffers from instability, higher delay variance (see Fig. 7).

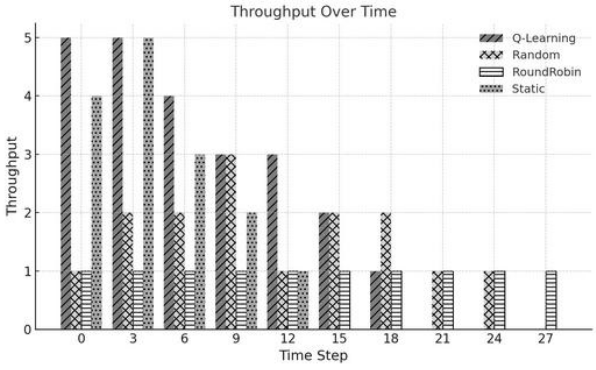


Fig. 6. Comparison of throughput with different approaches.

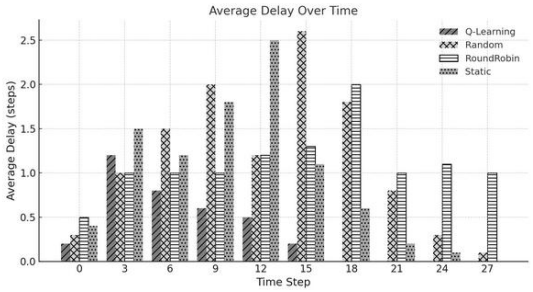


Fig. 7. Comparison of delay outcome with different approaches.

It can be seen from Fig. 8 that Q-learning rapidly explores its action space in early stages which consumes energy. However, once it learns the optimal transmission pattern then energy consumption of IoT devices significantly drops and remains stable. However, it also outperforms others by minimizing the energy use once the optimal policy is learned which static and non-adaptive schemes fail to achieve in LoRaWAN.

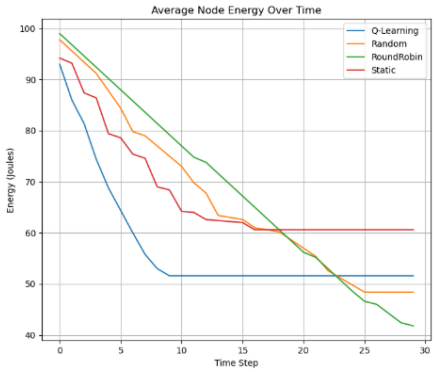


Fig. 8. Comparison of energy outcome with different approaches.

Fig. 9 reveals how many devices are being blacklisted throughout the system's operation in LoRaWAN. The blacklist count increases gradually which suggests that the security system is capable of detecting and isolating misbehaving or low-trust devices which mostly forward suspicious packets in LoRaWAN environment. Here, Q-learning plays a crucial role towards evaluating device behavior and taking penalizing actions over time. The trust-driven approach in the proposed scheme not only improves security and reliability but also contributes towards optimal throughput, lower delay and stable energy consumption.

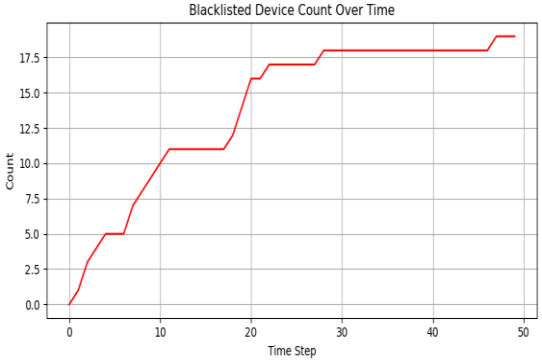


Fig. 9. Analysis of security in LoRaWAN.

VI. Conclusion

The proposed work introduces a Q-learning-based adaptive scheduling and duty cycle strategy to enhance the performance of LoRaWAN. Here, the proposed adaptive duty cycling approach incorporates Q-learning to dynamically choose the best time slot based on traffic, battery and buffer status reducing congestion and improving energy efficiency. Additionally, the unified framework also offers dynamic energy management and optimized transmission under traffic variability. An efficient channel scheduling algorithm is also proposed to enhance the bandwidth utilization, throughput and delay performance in LoRaWAN. The performance of the proposed Q-learning is also evaluated against Static, Random, and Round Robin approaches. The experimental results show that the Q-learning significantly improves the throughput performance (peak: 5 units), and also ensure lower average delay (converging to 0 within 10 steps), and improved energy efficiency (stabilizing at ~52J, ~15% better than Round Robin). It also offers a security analysis strategy that effectively identifies up to 19 malicious or underperforming devices via dynamic blacklisting. The outcome also conceptually justifies that the learning-based approach adapts well to traffic variations and security threats, outperforming static and randomized methods. The proposed framework demonstrates how reinforcement learning and lightweight edge security can jointly enhance LoRaWAN performance and ensure adaptive, energy-efficient, and secure communication in large-scale IoT deployments. In future work, we aim to extend this framework using deep reinforcement learning, multi-agent collaboration, and trust-aware reward models, with validation on more extensive LoRaWAN deployments.

References

[1] S. Devalal and A. Karthikeyan, "LoRa technology—An overview," in Proc. 2nd Int. Conf. Electron., Commun. Aerosp. Technol. (ICECA), Coimbatore, India, Mar. 2018, pp. 284–290. doi: 10.1109/ICECA.2018.8474803.

[2] N. Telagam, N. Kandasamy, and D. Ajitha, "Smart healthcare monitoring system using LoRaWAN IoT and machine learning methods," in Practical Artificial Intelligence for Internet of Medical Things, CRC Press, 2023, pp. 85–104. doi: 10.1201/9781003362725-6.

[3] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," ICT Express, vol. 5, no. 1, pp. 1–7, Mar. 2019. doi: 10.1016/j.icte.2019.01.005.

[4] C. Gomez, J. C. Veras, R. Vidal, L. Casals, and J. Paradells, "A Sigfox energy consumption model," Sensors, vol. 19, no. 3, p. 681, Feb. 2019. doi: 10.3390/s19030681.

[5] A. Farhad, D. H. Kim, B. H. Kim, A. F. Y. Mohammed, and J. Y. Pyun, "Mobility-aware resource assignment to IoT applications in long-range wide area networks," IEEE Access, vol. 8, pp. 186111–186124, 2020. doi: 10.1109/ACCESS.2020.3029947.

[6] D. Kjendal, "LoRa-Alliance regional parameters overview," J. ICT Standardization, vol. 9, no. 1, pp. 35–46, 2021. doi: 10.13052/jicts2245-800X.9113.

[7] S. R. Borkar, "Long-term evolution for machines (LTE-M)," in LPWAN Technologies for IoT and M2M Applications, Academic Press, 2020, pp. 145–166. doi: 10.1016/B978-0-12-818880-4.00010-1.

[8] J. Haxhibeqiri, E. De Poorter, I. Moerman, and J. Hoebeke, "A survey of LoRaWAN for IoT: From technology to application," Sensors, vol. 18, no. 11, p. 3995, Nov. 2018. doi: 10.3390/s18113995.

[9] K. L. Tsai, F. Y. Leu, I. You, S. W. Chang, S. J. Hu, and H. Park, "Low-power AES data encryption architecture for a LoRaWAN," IEEE Access, vol. 7, pp. 146348–146357, 2019. doi: 10.1109/ACCESS.2019.2945801.

[10] O. B. Seller and N. Sornin, Low Power Long Range Transmitter, U.S. Patent 9,252,834, Feb. 2, 2016.

[11] C. A. Gomez, A. Shami, and X. Wang, "Machine learning aided scheme for load balancing in dense IoT networks," Sensors, vol. 18, no. 11, p. 3779, 2018. doi: 10.3390/s18113779.

[12] T. Yatagan and S. Oktug, "Smart spreading factor assignment for LoRaWANs," in Proc. IEEE Symp. Comput. Commun. (ISCC), Barcelona, Spain, Jun. 2019, pp. 1–7. doi: 10.1109/ISCC47284.2019.8969645.

[13] S. U. Minhaj et al., "Intelligent resource allocation in LoRaWAN using machine learning techniques," IEEE Access, vol. 11, pp. 10092–10106, 2023. doi: 10.1109/ACCESS.2023.3241097.

[14] S. Cui and I. Joe, "Collision prediction for a low power wide area network using deep learning methods," J. Commun. Netw., vol. 22, no. 3, pp. 205–214, Jun. 2020. doi: 10.1109/JCN.2020.000016.

[15] L. Liu, Y. Yao, Z. Cao, and M. Zhang, "DeepLoRa: Learning accurate path loss model for long distance links in LPWAN," in Proc. IEEE INFOCOM, Vancouver, BC, Canada, May 2021, pp. 251–260. doi: 10.1109/INFOCOM42981.2021.9488831.

[16] S. Lee, J. Lee, J. Hwang, and J. K. Choi, "A novel deep learning-based IoT device transmission interval management scheme for enhanced scalability in LoRa networks," IEEE Wireless Commun. Lett., vol. 10, no. 11, pp. 2538–2542, Nov. 2021. doi: 10.1109/LWC.2021.3104133.

[17] S. I. A. Elkarim et al., "Deep learning based joint collision detection and spreading factor allocation in LoRaWAN," in Proc. IEEE 42nd Int. Conf. Distributed Comput. Syst. Workshops (ICDCSW), Bologna, Italy, Jul. 2022, pp. 187–192. doi: 10.1109/ICDCSW55796.2022.00054.

[18] R. M. Sandoval, A. J. Garcia-Sanchez, and J. Garcia-Haro, "Optimizing and updating LoRa communication parameters: A machine learning approach," IEEE Trans. Netw. Serv. Manag., vol. 16, no. 3, pp. 884–895, Sep. 2019. doi: 10.1109/TNSM.2019.2934601.

[19] N. Aihara, K. Adachi, O. Takyu, M. Ohta, and T. Fujii, "Q-learning aided resource allocation and environment recognition in LoRaWAN with CSMA/CA," IEEE Access, vol. 7, pp. 152126–152137, 2019. doi: 10.1109/ACCESS.2019.2947754.

[20] M. Jouhari, N. Saeed, M. S. Alouini, and E. M. Amhoud, "A survey on scalable LoRaWAN for massive IoT: Recent advances, potentials, and challenges," IEEE Commun. Surveys Tuts., vol. 25, no. 3, pp. 1841–1876, 2023. doi: 10.1109/COMST.2023.3244018.

[21] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of LoRaWAN," IEEE Commun. Mag., vol. 55, no. 9, pp. 34–40, Sep. 2017. doi: 10.1109/MCOM.2017.1600613.

[22] T. Deng, J. Zhu, and Z. Nie, "An improved LoRaWAN protocol based on adaptive duty cycle," in Proc. IEEE 3rd Inf. Technol. Mechatronics Eng. Conf. (ITOEC), Chongqing, China, Oct. 2017, pp. 1122–1125. doi: 10.1109/ITOEC.2017.8320815.

[23] A. Maleki, H. H. Nguyen, E. Bedeer, and R. Barton, "A tutorial on chirp spread spectrum modulation for LoRaWAN: Basics and key advances," IEEE Open J. Commun. Soc., 2024. doi: 10.1109/OJCOMS.2024.3384025.

[24] D. Kalugina and A. Pastukh, "Evaluation of the feasibility of implementing satellite IoT in the terrestrial IoT frequency bands of 868 MHz and 915 MHz," in Proc. Int. Sci. Tech. Conf. Modern Comput. Netw. Technol. (MoNeTeC), Oct. 2024, pp. 1–6. doi: 10.1109/MoNeTeC60266.2024.10406290.

[25] X. Yang, E. Karampatzakis, C. Doerr, and F. Kuipers, "Security vulnerabilities in LoRaWAN," in Proc. IEEE/ACM 3rd Int. Conf. Internet-of-Things Design and Implementation (IoTDI), Apr. 2018, pp. 129–140. doi: 10.1109/IoTDI.2018.0001.

[26] P. Yu, F. Zhou, X. Zhang, X. Qiu, M. Kadoch, and M. Cheriet, "Deep learning-based resource allocation for 5G broadband TV service," IEEE Trans. Broadcast., vol. 66, no. 4, pp. 800–813, Dec. 2020. doi: 10.1109/TBC.2020.3004470.

[27] S. U. Minhaj et al., "Intelligent resource allocation in LoRaWAN using machine learning techniques," IEEE Access, vol. 11, pp. 10092–10106, 2023. doi: 10.1109/ACCESS.2023.3241097.

[28] S. Lavdas, N. Bakas, K. Vavousis, W. El Hajj, and Z. Zinonos, "Evaluating LoRaWAN network performance in smart city environments using machine learning," IEEE Internet Things J., 2025. [Early Access]. doi: 10.1109/JIOT.2025.339412.

[29] M. Alkhayyal and A. M. Mostafa, "Enhancing LoRaWAN sensor networks: A deep learning approach for performance optimizing and energy efficiency," Computers, Materials & Continua, vol. 83, no. 1, 2025. doi: 10.32604/cmc.2025.046261.

[30] A. Farhad and J. Y. Pyun, "LoRaWAN meets ML: A survey on enhancing performance with machine learning," Sensors, vol. 23, no. 15, p. 6851, 2023. doi: 10.3390/s23156851.