ARTICLE

Deep Q-Learning Driven Protocol for Enhanced Border Surveillance with Extended Wireless Sensor Network Lifespan

Nimisha Rajput[1,#], **Amit Kumar**[1], **Raghavendra Pal**[1,#], **Nishu Gupta**[2,*], **Mikko Uitto**[2] and **Jukka Mäkelä**[2]

[1]Department of Electronics Engineering, Sardar Vallabhbhai National Institute of Technology, Surat, 395007, Gujrat, India
[2]Future Communication Networks, VTT Technical Research Centre of Finland Ltd., Oulu, 90590, Finland

*Corresponding Author: Nishu Gupta. Email: nishu.gupta@vtt.fi
#These authors contributed equally to this work

ABSTRACT: Wireless Sensor Networks (WSNs) play a critical role in automated border surveillance systems, where continuous monitoring is essential. However, limited energy resources in sensor nodes lead to frequent network failures and reduced coverage over time. To address this issue, this paper presents an innovative energy-efficient protocol based on deep Q-learning (DQN), specifically developed to prolong the operational lifespan of WSNs used in border surveillance. By harnessing the adaptive power of DQN, the proposed protocol dynamically adjusts node activity and communication patterns. This approach ensures optimal energy usage while maintaining high coverage, connectivity, and data accuracy. The proposed system is modeled with 100 sensor nodes deployed over a 1000 m × 1000 m area, featuring a strategically positioned sink node. Our method outperforms traditional approaches, achieving significant enhancements in network lifetime and energy utilization. Through extensive simulations, it is observed that the network lifetime increases by 9.75%, throughput increases by 8.85% and average delay decreases by 9.45% in comparison to the similar recent protocols. It demonstrates the robustness and efficiency of our protocol in real-world scenarios, highlighting its potential to revolutionize border surveillance operations.

KEYWORDS: Wireless sensor networks (WSNs); energy efficiency; reinforcement learning; network lifetime; dynamic node management; autonomous surveillance

1 Introduction

Many sensor nodes in wireless sensor networks (WSNs) [1] send and receive information about things like temperature, humidity, pressure, and movement. These sensor nodes can monitor and report data to a central sink node or base station since they are outfitted with sensing, processing, and communication capabilities. WSNs gather and monitor huge geographical areas in real-time in distant or inaccessible areas. Many fields have discovered uses for WSNs because of their adaptability, affordability, and scalability. These fields include healthcare, industrial automation, environmental monitoring, and military surveillance. The deployment of WSNs in these diverse fields highlights their significance in gathering critical information and facilitating informed decision-making. WSNs are an integral part of the Internet of things (IoT) [2].

Border surveillance is a crucial aspect of national security, aimed at preventing illegal activities such as smuggling, human trafficking, and unauthorized border crossings. Traditional methods of border surveillance involve the use of physical barriers, patrols by security personnel, surveillance cameras, and

radar systems. While these methods provide a degree of security, they are often limited by their coverage area, high operational costs, and susceptibility to environmental factors. Advancements in technology have led to the integration of unmanned aerial vehicles (UAVs) and satellite imagery, enhancing the ability to monitor vast and remote border areas. However, these methods still face challenges in terms of continuous real-time monitoring and resource allocation. Fig. 1 shows a general scenario of the use of WSNs in terrestrial border surveillance.
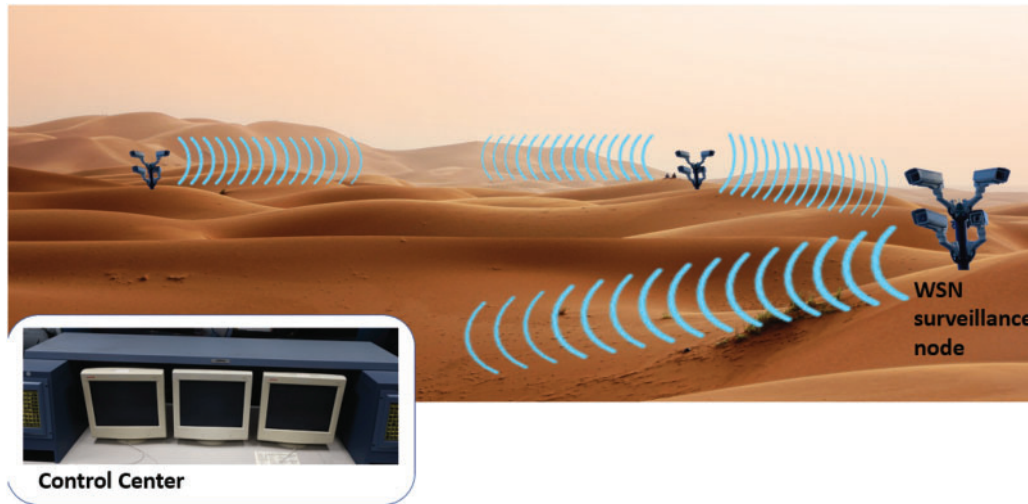

Figure 1: A general border surveillance wireless sensor network scenario

WSNs represent a highly effective technological approach for enhancing border surveillance. By enabling continuous, autonomous monitoring across large geographical regions, WSNs provide comprehensive oversight of border areas. Strategically deploying sensor nodes along boundary lines allows the construction of a robust surveillance framework capable of identifying and reporting unauthorized intrusions or suspicious movements in real-time. These networks are engineered to sense and transmit critical environmental data such as motion, vibrations, and other activity-specific signals, thereby improving situational awareness. Moreover, due to their resilience, WSNs remain functional in harsh climates and challenging terrains where conventional methods may underperform. The consistent and adaptive monitoring facilitated by WSNs substantially strengthens border security efforts, all while minimizing the reliance on human personnel.

Reinforcement Learning (RL) has emerged as an effective paradigm for optimizing the operation and control of WSNs [3]. Within these networks, RL algorithms have been successfully applied to manage core challenges such as optimal node deployment, efficient resource usage, and intelligent communication scheduling. Through interaction with their operating environment, RL agents learn adaptive strategies that enhance key network performance indicators, including energy efficiency, data throughput, and latency. For example, RL has been instrumental in dynamically modifying sensor node transmission power, timing data transmission events, and managing topological adjustments in response to evolving conditions. These practical implementations underscore the value of RL in improving the adaptability and efficiency of WSNs, particularly in unpredictable and dynamic deployment environments.

One of the most critical challenges facing WSNs in border surveillance is energy efficiency. As sensor nodes are generally powered by batteries, their maintenance–especially in remote or inaccessible areas– poses significant logistical difficulties. Consequently, extending the network's operational lifespan through

smart energy management becomes vital. Without such measures, high energy demands can result in early node failure, thereby degrading coverage, weakening communication links, and ultimately compromising the surveillance infrastructure.

To address this concern, reinforcement learning presents an intelligent and flexible solution. By designing Deep Q Learning (DQN)-driven energy aware routing protocols, it becomes possible to make real-time decisions about node activity and transmission behaviors. DQN-based models learn to conserve energy by determining optimal communication routes, adjusting transmission power as needed, and selectively activating nodes based on operational requirements. These capabilities help preserve battery life without sacrificing coverage or performance, resulting in significantly improved network sustainability. The protocol uses IEEE802.15.4 and is specifically designed for WSNs by incorporating energy models, duty cycling, and node-level decisions that reflect typical WSN constraints. Although the protocol is tailored for border surveillance applications using WSNs, its core learning mechanism can be adapted to other sensor-based or IoT deployments with similar energy and topology constraints.

The primary contributions of this paper can be summarized as follows:

1. A new energy-efficient protocol has been proposed based on reinforcement learning, specifically tailored to extend the lifespan of wireless sensor networks deployed in border surveillance applications.
2. A dynamic mechanism is introduced to govern node behavior and communication schedules in real-time, ensuring minimal energy usage while maintaining essential network parameters such as connectivity, coverage, and data fidelity. The dynamic mechanism refers to the Q-learning agent embedded in each node that, at each round, evaluates its current state (energy level, neighbor status, coverage need) and selects an action (sleep, forward, sense, or idle) to minimize long-term energy consumption while sustaining network performance.
3. The proposed protocol has been tested via simulation on a network of 100 sensor nodes distributed across a 1000 m × 1000 m area, reflecting realistic deployment scenarios.
4. Experimental results reveal considerable improvements over existing techniques, with enhancements including a 9.75% increase in network lifetime, an 8.85% gain in throughput, and a 9.45% reduction in average packet delay.

This paper is organized as follows: Section 2 provides a comprehensive literature review, highlighting recent studies and identifying gaps in the field. In Section 3, we outline the system model that serves as the foundation for our research. Section 4 details the proposed approach, including the methodology and techniques utilized. Section 5 presents an in-depth analysis of the simulation findings, featuring performance metrics and comparisons with existing methods. Finally, Section 6 concludes the paper by summarizing the main contributions and suggesting directions for future research.

2 Literature Review

The applications of WSNs in a variety of domains, such as industrial automation, healthcare, military surveillance, and environmental monitoring, have been thoroughly investigated in this literature review. The deployment of WSNs in border surveillance presents unique challenges and opportunities, particularly concerning energy efficiency and network lifetime.

Akyildiz et al. (2002) conducted one of the earliest comprehensive surveys of WSN architecture, applications, and design challenges [4]. Their study methodically reviewed the state of WSNs, identifying fundamental technical issues such as severe energy constraints, limited computation, and the need for scalability and robustness. As a result, they highlighted energy efficiency as a paramount concern and outlined potential techniques (e.g., duty cycling and data aggregation) to mitigate power consumption. The

strength of this seminal work lies in its broad, foundational insight–it established the key research problems and performance metrics that guided subsequent WSN research–however, it did not propose concrete solutions or quantitative evaluations. Studies such as [5] have highlighted the advantages of WSNs in terms of scalability, flexibility, and cost-effectiveness for large-scale monitoring applications. Authors in [6] present the use of edge computing in IoT networks.

Recent progress in the domain of WSNs has placed significant emphasis on improving system performance under adverse and dynamic operating conditions. For instance, the work of [7] examines multiple energy-conserving protocols aimed at extending the lifetime of WSNs. They reported that specialized routing techniques–for instance, forming a data-gathering chain among nodes with greedy neighbor selection–can drastically reduce redundant transmissions and thus conserve node energy. The results compiled in their survey showed that protocols like Power-Efficient Gathering in Sensor Information Systems (PEGASIS) achieve significantly longer network lifetime (by aggregating data to minimize communication cost) relative to shortest-path routing. The advantage of this work is its comprehensive taxonomy and comparative analysis of routing approaches, clearly identifying the strengths of each category (chain-based routing compresses data effectively, cluster-based routing balances load). Likewise, reference [8] delves into the integration of machine learning techniques to optimize various operational aspects of WSNs. When applied to border surveillance, WSNs demonstrate their capability to detect and relay unauthorized movements in real time, thereby enhancing situational awareness [9].

Conventional border surveillance infrastructures generally rely on a combination of physical barriers, patrolling personnel, video surveillance, and radar systems. While these tools do provide a level of protection, they suffer from inherent limitations such as restricted coverage, high maintenance costs, and vulnerability to environmental disruptions [10]. The incorporation of advanced technologies, such as unmanned aerial vehicles (UAVs) and satellite-based observation, has contributed to expanding surveillance reach in remote border zones [11]. Nonetheless, maintaining uninterrupted real-time monitoring and efficient resource allocation remains a major concern in such systems [12].

WSNs, in this context, offer multiple advantages. Their ability to function autonomously in isolated or difficult-to-access regions makes them suitable for persistent border surveillance. These networks facilitate real-time data acquisition and support adaptive monitoring based on situational needs. Empirical studies such as [11] confirm the utility of WSNs in tracking unauthorized crossings and illicit activity. Furthermore, WSNs can be effectively integrated with other surveillance technologies to establish a more resilient and holistic security framework [13].

The installation of sensor nodes along border areas enables the formation of an extensive surveillance grid that can continuously scan the surrounding environment. Data collected from such networks is typically forwarded to a centralized base station for further analysis. This persistent flow of information minimizes the need for constant manual oversight and enhances the efficiency of border security operations [11].

In recent years, advanced techniques from machine learning have been introduced to WSN problems, aiming to autonomously improve network decision-making and efficiency.

A notable contribution in this area is the introduction of deep Q-learning by [14], which applies neural networks to approximate Q-values, thus enabling better decision-making in complex operational spaces. Within WSNs, RL has been used to dynamically regulate transmission power, organize communication schedules, and restructure network topology based on current conditions [15].

Energy management remains one of the most pressing challenges in WSN applications, especially when these networks are deployed in remote areas with restricted access to power infrastructure. Prior

research, such as [7] and [16], explores different strategies and routing algorithms aimed at reducing energy consumption and prolonging network longevity.

Reinforcement learning provides a strategic approach to counter this issue by enabling the development of energy-aware routing protocols. These protocols adapt in real time by modifying transmission behavior and node activity levels based on environmental feedback [10].

Research outlined in [14] demonstrates how RL can support intelligent routing, energy-efficient power control, and adaptive duty cycling. These capabilities ensure continuous monitoring while significantly extending the service life of WSNs [16].

The study by [17] offers a detailed analysis of clustering techniques tailored to heterogeneous WSNs, aiming to improve energy efficiency. This scheme relies on a probabilistic model that incorporates parameters such as node density within sensing range and current energy levels–initial, residual, and dynamic. This clustering approach balances network load distribution and improves overall energy utilization. Simulation results validated through MATLAB reveal that the protocol surpasses traditional models like Stable Election Protocol (SEP), especially in terms of network stability and energy performance.

Additional contributions, such as those by [18], explore the concept of uneven clustering to address the hotspot problem and optimize energy usage. Their analysis supports dynamic cluster creation and hierarchical clustering as effective means to extend network lifetime while maintaining balanced energy distribution across the WSN.

The authors in [19] used Circle Rock Hyrax Swarm Optimisation and Levy flight-based momentum search to propose energy-efficient clustering and routing for wireless body area networks (WBANs). These approaches optimize cluster head selection and routing patterns by enhancing energy use and decreasing delays in healthcare monitoring.

A method to prevent primary user emulation attacks in Cognitive radio sensor networks (CRSNs) was suggested by [20] using adaptive sensing. By using hardware-based validation and dynamically adjusting sensing procedures, their solution improves energy economy, throughput and minimises the likelihood of sensing errors.

The authors in [21] investigated the use of WSNs in precision farming, emphasising communication and data collecting that uses less energy. They highlighted how IoT and AI technology may be used for sustainable agriculture monitoring and highlighted issues like energy harvesting and scalability.

Several energy-efficient protocols for WSNs have been proposed in recent years to prolong network lifetime. For instance, Saxena et al. [22] developed a fuzzy logic-based routing protocol for environmental monitoring WSNs, which allows nodes to periodically enter low-power sleep modes to conserve energy. Their hardware-implemented protocol achieved notable energy savings, though the fuzzy rules require careful tuning and the approach was evaluated in a relatively static scenario. Reinforcement learning has also been applied to WSNs–Kesavan et al. [23] introduce an RL-based multi-task scheduling model at the network edge for Internet-of-Everything applications, focusing on secure and efficient data collection. This method improved energy efficiency (achieving 89% energy efficiency and low latency) by dynamically allocating tasks, but it is tailored to Internet of Everything (IoE) edge computing rather than sensor node routing. Other recent works have explored dynamic clustering and sleep scheduling. For example, El-Shenhabi et al. [24] discussed an RL-based clustering and sleep scheduling algorithm that yields a 77% extension in network lifetime over traditional schemes, thanks to adaptive sleep cycles; however, the complexity of their compressive data gathering technique may limit scalability. In the medium access control (MAC) layer domain, Khan et al. [25] employ deep RL to optimize duty cycling (RL-BMAC), obtaining 35% better energy

efficiency and 58% higher throughput than the baseline B-MAC protocol. In Tables 1 and 2, we summarize key related works, comparing their techniques, performance, and limitations.

Table 1: Summary of related works

Reference	Method	Results	Strengths	Limitations
Akyildiz et al. (2002) [4]	Survey of WSN architecture	Identified key challenges	Foundational insight	No concrete solution
Yick et al. (2008) [5]	Scalability in WSN	Suited for large monitoring	Cost-effective	Outdated scenarios
Donta et al. [6]	Edge computing in IoT	Enhanced control	Real-time decisions	Limited to IoT
Pantazis et al. (2013) [7]	Routing protocol survey	PEGASIS more efficient	Routing taxonomy	Qualitative only
Kumar et al. (2019) [8]	Machine Learning (ML) in WSNs	Improved operations	Adaptive techniques	Complexity of ML
Mostafaei et al. (2018) [9]	WSN in surveillance	Detected intrusions	Real-time alerts	Needs integration
Zhao (2010) [10]	Sensing model	Reduced energy use	Environment-adaptive	Needs validation
Ibrahim et al. (2020) [11]	UAV in border zones	Extended reach	Remote monitoring	Expensive infrastructure
Sami and Chun (2016) [12]	Satellite border surveillance	Continuous data	Automated coverage	Resource-intensive
Bhadwal et al. (2018) [13]	Smart surveillance	Real-time tracking	Holistic framework	No RL component
Mnih et al. (2015) [14]	Deep Q-learning	Better decisions	Neural Q-approximation	Not WSN specific
Alsheikh et al. (2014) [15]	RL for control	Dynamic topology	Real-time adaptability	Training stability
Wang et al. (2015) [16]	Energy-aware routing	Increased life	Efficient protocols	Hardware not tested
Tripathi et al. (2022) [17]	Duty cycling in WSN	Longer life, better coverage	Scenario optimized	Limited prototype
Sharma et al. (2024) [18]	Uneven clustering	Hotspot avoidance	Balanced energy	Needs testing
Poornima et al. (2024) [19]	Hyrax + Levy optimization	Better routing in WBAN	Reduced delay	High complexity
Ram et al. (2023) [20]	Adaptive sensing in CRSN	Prevented attacks	Secure and energy-efficient	Focused on CRSN only
Aggarwal et al. (2024) [21]	WSN in agriculture	Sustainable monitoring	AI for precision	Energy harvesting challenge
Saxena et al. [22]	Fuzzy routing	Energy savings	Sleep scheduling	Static rules, tuning needed

(Continued)

Table 1 (continued)

Reference	Method	Results	Strengths	Limitations
Kesavan et al. [23]	RL edge multi-tasking	89% energy efficiency	Secure IoE control	Not for routing
El-Shenhabi et al. [24]	RL clustering/sleep	77% more lifetime	Adaptive cycles	High complexity
Khan et al. [25]	RL-BMAC protocol	58% higher throughput	Energy-efficient MAC	MAC-specific only

Table 2: Simulation parameters

Parameter	Value
Learning rate (α)	0.1
Transmission energy (E_{elec})	50 nJ/bit
Number of nodes	20, 40, 60, 80, 100
Discount factor (γ)	0.9
Network area	1000 m × 1000 m
Sink node position	Coordinates (0,0)
Exploration rate (ε)	0.1
Free space model (ε_{fs})	10 pJ/bit/m^2
Multipath model (ε_{mp})	0.0013 pJ/bit/m^4
Threshold distance (d_0)	87 m
Packet size	512 bytes

Research Gap

Despite significant advancements in WSNs and reinforcement learning, there remains a critical need to address the issue of energy consumption in border surveillance applications. Current methods, while effective in certain scenarios, often fall short in providing sustained energy efficiency and network longevity. This paper proposes a novel reinforcement learning-based energy-efficient protocol aimed at prolonging the lifetime of border surveillance WSNs. To improve the efficiency of border security operations, the suggested approach fills the current research gap by improving the WSN system's energy usage while preserving high performance in terms of coverage, connection, and data accuracy.

3 System Model

This section outlines the modeling framework used to simulate the WSN for border surveillance. It provides the network architecture, deployment assumptions, and the energy consumption model employed for transmitting, receiving, and processing data within the network. These models form the foundation for implementing and evaluating the proposed Q-learning based energy efficient protocol.

3.1 Network Architecture

The network's 100 sensor nodes are evenly spaced within a 1000 m × 1000 m region. At one corner of the space, precisely at coordinates (0, 0), lies the sink node, which is in charge of data processing and aggregation. The starting energy level E_0 of every sensor node is the same.

3.2 Energy Consumption Model

Energy sensing and communication are both included in the energy consumption model. Reference [26] provides the energy needed to send a k-bit packet across a distance d.

$$E_{tx}(k,d) = \begin{cases} k \cdot E_{elec} + k \cdot \epsilon_{fs} \cdot d^2 & \text{if } d < d_0 \\ k \cdot E_{elec} + k \cdot \epsilon_{mp} \cdot d^4 & \text{if } d \geq d_0 \end{cases} \tag{1}$$

In the definition of the threshold distance, represented by d_0, the energy dissipation factors of the free space and multipath fading channel models are represented by $epsilon_{fs}$ and $epsilon_{mp}$, respectively, while the energy wasted per bit to run the transmitter or receiver circuit is represented by E_{elec}.

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \tag{2}$$

The energy consumed to receive a k-bit packet is:

$$E_{rx}(k) = k \cdot E_{elec} \tag{3}$$

4 Proposed Method

This section presents the design and operation of the proposed DQN based energy efficient protocol for WSNs deployed in border surveillance. The method leverages DQN to dynamically manage node activity, transmission power, and communication decisions in real time. Detailed descriptions of the learning framework, state-action representations, reward formulation, and learning algorithm are provided in the following subsections.

4.1 Overview

The proposed method employs deep Q-learning (DQN) to dynamically manage the activity and communication patterns of the sensor nodes to minimize energy consumption while maintaining network performance. The DQN agent at a node learns an optimal policy to balance the trade-off between energy efficiency, coverage and connectivity.

4.2 Reinforcement Learning Framework

The reinforcement learning framework consists of the following elements:

- **State Space:** The state space S represents the energy levels of all sensor nodes and their relative positions to the sink node.
- **Action Space:** The action space A includes actions such as adjusting transmission power levels, changing active/inactive states of nodes, and selecting communication paths.
- **Reward Function:** The reward function R is designed to penalize high energy consumption and reward actions that extend network lifetime while maintaining coverage and connectivity.

4.3 State Representation

Each state $s_t \in S$ at time t is represented as a vector:

$$s_t = [E_1(t), E_2(t), \ldots, E_{100}(t), d_1(t), d_2(t), \ldots, d_{100}(t)] \tag{4}$$

where $E_i(t)$ is the residual energy of node i at time t, and $d_i(t)$ is the distance of node i from the sink node at time t.

4.4 Action Representation

The action $a_t \in A$ at time t is a vector indicating the adjustments made to the nodes, such as transmission power levels and active/inactive states:

$$a_t = [P_1(t), P_2(t), \ldots, P_{100}(t), A_1(t), A_2(t), \ldots, A_{100}(t)] \tag{5}$$

where $P_i(t)$ is the transmission power level of node i at time t, and $A_i(t)$ is the active state (1 for active, 0 for inactive) of node i at time t.

The transmission power level $P_i(t)$ for node i at time t can be calculated using the following equation:

$$P_i(t) = P_{\min} + (P_{\max} - P_{\min}) \left(\frac{d_i(t)}{d_{\max}} \right)^\phi \left(\frac{E_i(t)}{E_{\text{init}}} \right)^\chi \left(\frac{1}{LQ_i(t)} \right)^\psi \left(\frac{1}{H_i(t)} \right)^\omega \tag{6}$$

where:

- $P_{\min}$ and $P_{\max}$ are the minimum and maximum transmission power levels, respectively.
- $d_i(t)$ is the distance of node i from the sink node at time t.
- $d_{\max}$ is the maximum communication range.
- $E_i(t)$ is the residual energy of node i at time t.
- E_{init} is the initial energy of the nodes.
- $LQ_i(t)$ is the link quality of node i at time t.
- $H_i(t)$ is the number of hops from node i to the sink node at time t.
- ϕ, χ, ψ, ω are weighting factors that determine the importance of each parameter. We have considered all the values equal to 1.

4.5 Reward Function

The reward function $R(s_t, a_t)$ is defined as:

$$R(s_t, a_t) = - \left(\kappa \sum_{i=1}^{100} E_{tx,i}(t) + \mu \sum_{i=1}^{100} E_{rx,i}(t) \right) + \nu \cdot \text{Coverage}(t) + \xi \cdot \text{Connectivity}(t) \tag{7}$$

where κ, μ, ν, ξ are weighting factors that balance the importance of energy consumption, coverage, and connectivity. $E_{tx,i}(t)$ and $E_{rx,i}(t)$ are the transmission and reception energy consumptions of node i at time t. Coverage and connectivity are functions that measure the performance of the network. We have considered $\kappa = 0.5$, $\mu = 0.5$, $\nu = 1$ and $\xi = 1$.

Coverage at time t is a measure of how much of the monitored area is being covered by the active sensor nodes. It can be defined as the ratio of the effective area covered by the sensor nodes to the total area that

needs to be monitored, considering factors such as the sensing range, probability of detection, and overlap of coverage areas. The equation for coverage is given by:

$$\text{Coverage}(t) = \frac{\sum_{i=1}^{N_{\text{active}}(t)} \left(\pi r_i^2 \cdot P_d(i, t) \right) - A_{\text{overlap}}(t)}{A_{\text{total}}} \tag{8}$$

where:

- $N_{\text{active}}(t)$ is the number of active sensor nodes at time t.
- r_i is the sensing range of sensor node i.
- $P_d(i, t)$ is the probability of detection for sensor node i at time t, calculated as:

$$P_d(i, t) = P_{d0} \exp\left(-\frac{d_i(t)}{d_0} \right) \tag{9}$$

- $A_{\text{overlap}}(t)$ is the total overlap area covered by multiple sensor nodes at time t.
- A_{total} is the total area that needs to be monitored.

Here, P_{d0} represents the base probability of detection at zero distance, meaning the probability that the sensor node i will detect a target if the target is located directly at the sensor node's position. In ideal conditions, P_{d0} is typically set to 1, indicating maximum detection probability. In this work as well P_{d0} is set to 1. The parameter $d_i(t)$ is the distance between the sensor node i and the target at time t, and d_0 is a characteristic distance that determines how quickly the detection probability decreases with distance.

The total overlapping area $A_{\text{overlap}}(t)$ is calculated as:

$$A_{\text{overlap}}(t) = \sum_{i=1}^{N_{\text{active}}(t)-1} \sum_{j=i+1}^{N_{\text{active}}(t)} A_{\text{overlap}}(i, j, t) \tag{10}$$

where the overlapping area between two sensor nodes i and j is given by:

$$A_{\text{overlap}}(i, j, t) = \begin{cases} 0 & \text{if } d_{ij} \geq r_i + r_j \\ \pi \min(r_i^2, r_j^2) & \text{if } d_{ij} \leq |r_i - r_j| \\ r_i^2 \cos^{-1}\left(\frac{d_{ij}^2 + r_i^2 - r_j^2}{2 d_{ij} r_i} \right) + r_j^2 \cos^{-1}\left(\frac{d_{ij}^2 + r_j^2 - r_i^2}{2 d_{ij} r_j} \right) \\ \quad - \frac{1}{2} \sqrt{(-d_{ij} + r_i + r_j)(d_{ij} + r_i - r_j)(d_{ij} - r_i + r_j)(d_{ij} + r_i + r_j)} & \text{otherwise} \end{cases} \tag{11}$$

where:

- d_{ij} is the distance between the centers of the two circles (sensor nodes i and j).
- r_i and r_j are the sensing ranges of nodes i and j, respectively.

Connectivity at time t refers to the network's ability to maintain communication paths between sensor nodes and the sink node. It is defined as the ratio of the number of nodes that can successfully communicate with the sink to the total number of nodes in the network, considering factors such as energy levels, communication range, and link quality. The equation for connectivity is given by:

$$\text{Connectivity}(t) = \frac{\sum_{i=1}^{N_{\text{total}}} \left(\mathbf{1}_{\{E_i(t) > E_{\min}\}} \cdot \sum_{j=1}^{N_{\text{total}}} \mathbf{1}_{\{d_{ij} \leq R_c\}} \cdot LQ_{ij}(t) \right)}{N_{\text{total}}} \tag{12}$$

where:

- N_{total} is the total number of nodes in the network.
- $E_i(t)$ is the energy level of node i at time t.
- $E_{\min}$ is the minimum energy level required for a node to be considered active.
- d_{ij} is the distance between node i and node j.
- R_c is the communication range of the nodes.
- $LQ_{ij}(t)$ is the link quality between node i and node j at time t.
- $\mathbf{1}_{\{\cdot\}}$ is the indicator function that is 1 if the condition inside is true and 0 otherwise.

Link quality $LQ_{ij}(t)$ can be calculated using the following equation:

$$
\begin{aligned}
LQ_{ij}(t) = \quad & \left(\frac{\text{RSS}_{ij}(t)}{\text{RSS}_{\max}} \right)^{\zeta} \cdot \left(\frac{\text{SNR}_{ij}(t)}{\text{SNR}_{\max}} \right)^{\eta} \\
& \cdot \left(\frac{\text{PDR}_{ij}(t)}{\text{PDR}_{\max}} \right)^{\theta} \cdot \left(\frac{1-\text{BER}_{ij}(t)}{1-\text{BER}_{\min}} \right)^{\lambda} \\
& \cdot \left(\frac{d_{\max}-d_{ij}(t)}{d_{\max}} \right)^{\epsilon}
\end{aligned}
\tag{13}
$$

where:

- $\text{RSS}_{ij}(t)$ is the received signal strength between nodes i and j at time t.
- $\text{RSS}_{\max}$ is the maximum possible received signal strength.
- $\text{SNR}_{ij}(t)$ is the signal-to-noise ratio between nodes i and j at time t.
- $\text{SNR}_{\max}$ is the maximum possible signal-to-noise ratio.
- $\text{PDR}_{ij}(t)$ is the packet delivery ratio between nodes i and j at time t.
- $\text{PDR}_{\max}$ is the maximum possible packet delivery ratio.
- $\text{BER}_{ij}(t)$ is the bit error rate between nodes i and j at time t.
- $\text{BER}_{\min}$ is the minimum possible bit error rate (usually close to 0).
- $d_{ij}(t)$ is the distance between nodes i and j at time t.
- $d_{\max}$ is the maximum communication range.
- $\zeta, \eta, \theta, \lambda, \epsilon$ are weighting factors that determine the importance of each parameter. We have considered the values of all these parameters equal to 1.

4.6 Learning Algorithm

The Q-learning algorithm is utilized here, which is a type of model-free reinforcement learning method. Q-learning aids in determining the optimal policy. The Q-value, represented as $Q(s_t, a_t)$, signifies the expected cumulative reward for performing action a_t in state s_t and subsequently following the optimal policy.

The update rule for Q-learning is formulated as:

$$
Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \right]
\tag{14}
$$

Here:

- $Q(s_t, a_t)$ stands for the current Q-value, indicating the expected cumulative reward of taking action a_t in state s_t and later following the optimal policy.
- α is the learning rate, a parameter that tells the extent to which new experience overrides the old information. A larger value of α means the agent learns more rapidly by giving more importance to new experiences.
- r_t is the reward got after taking a_t in s_t. This reward serves as feedback to the agent about the efficacy of the action taken.

- The discount factor, γ, is a measure of the significance of future benefits. While a discount factor close to 0 forces the agent to concentrate more on current benefits, a factor close to 1 rates future rewards almost as highly as immediate rewards.
- s_{t+1} is the next state the agent moves to after taking action a_t in state s_t.
- $\max_{a'} Q(s_{t+1}, a')$ represents the maximum Q-value for the next state s_{t+1}, considering all possible actions a' in that state. This term signifies the best possible cumulative reward the agent can achieve from state s_{t+1} onwards, based on the current Q-values.

This method refines the Q-values iteratively based on the agent's experiences. The update rule modifies the Q-value $Q(s_t, a_t)$ by adding a correction term:

$$\alpha \left[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \right] \tag{15}$$

It represents the variation between the existing estimate of the Q-value and a revised estimate, which incorporates both the immediate reward and the maximum anticipated future Q-value, adjusted by the learning rate α. This update mechanism is referred to as temporal difference (TD) learning. It enables the learning agent to evolve through interaction with the environment, without requiring a predefined model of it.

The Q-values are progressively updated by striking a balance between exploration of new actions and exploitation of already learned behaviors. This iterative refinement guides the agent toward a policy that yields the highest expected reward over time. Within the Q-learning framework, the exploration rate, denoted as ϵ, plays a pivotal role. At the beginning of the learning process, ϵ is deliberately kept high to promote exploration of diverse actions and environmental states. As the learning advances, this rate is systematically reduced, allowing the agent to increasingly rely on the policies it has learned to be effective. The balance between exploration and exploitation is maintained using an ϵ-greedy strategy.

Algorithm 1 illustrates the step-by-step structure of the proposed reinforcement learning-based energy-efficient protocol tailored for WSNs. Initially, a Q-table $Q(s, a)$ is established to store the utility estimates for each state-action pair, which guides the agent in making decisions based on its experience. The initial state s is defined by the residual energy levels and positions of the sensor nodes in the network. Essential learning parameters such as the learning rate α, the discount factor γ, and the exploration rate ϵ are configured to control the agent's behavior during learning.

The agent operates in multiple learning episodes, where each episode comprises several time steps. During each time step, an action a_t is selected from the current state s_t using the ϵ-greedy approach. Upon execution of the action, the environment responds by transitioning the agent to a new state s_{t+1}, and a corresponding reward r_t is received. The Q-value for the state-action pair (s_t, a_t) is updated using the Q-learning formula, reflecting the most recent experience. The state s_t is updated to s_{t+1}, and the node's energy and positional data are refreshed to reflect the new environment state.

Following this, transmission power levels for each node are adaptively adjusted based on multiple parameters, including remaining battery energy, proximity to the sink node, and the observed link quality. Duty cycling is applied to dynamically switch nodes between active and inactive states to conserve energy while maintaining the integrity of the sensing operation. Communication routes are selected by evaluating several link quality indicators such as signal-to-noise ratio (SNR), packet delivery ratio (PDR), received signal strength (RSS), and hop count.

To stabilize the learning process, the agent records each transition as a tuple (s_t, a_t, r_t, s_{t+1}) in a replay buffer. This buffer is later used to sample mini-batches for training, reducing the effect of temporal

correlations between sequential experiences. For reinforcement learning models that use deep Q-networks (DQNs), an additional target network is periodically updated to improve learning convergence and avoid instability. The learning episode concludes prematurely if all sensor nodes exhaust their energy supply, signaling the failure of the network. At the end of each episode, the exploration rate ϵ is decayed to encourage the agent to exploit the learned policies more frequently as training progresses.

Algorithm 1: Reinforcement learning-based energy efficient protocol

---

1: Initialize Q-table $Q(s, a)$ with initial values
2: Initialize state s representing energy levels and positions of nodes
3: Set parameters: learning rate α, discount factor γ, and exploration rate ε
4: **for** each round **do**
5:      **for** each time step t **do**
6:          Choose a_t from s_t using ε-greedy policy
7:          Execute a_t then calculate r_t and s_{t+1}
8:          Update Q-value $Q(s_t, a_t)$ using the following equation:

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

9:          Update state $s_t \leftarrow s_{t+1}$
10:        Update energy levels and positions of nodes
11:        Adjust transmission power levels based on s_{t+1}
12:        Manage node duty cycles to conserve energy
13:        Optimize communication paths considering link quality $LQ_{ij}(t)$
14:        Integrate experience replay and store (s_t, a_t, r_t, s_{t+1}) in replay buffer
15:        Perform mini-batch updates from replay buffer to reduce correlation
16:        Update target network periodically for stable learning (in case of DQN)
17:        **if** all nodes are dead **then**
18:            Break from loop
19:        **end if**
20:      **end for**
21:    Decay exploration rate ε to balance exploration and exploitation
22: **end for**

---

5 Simulation and Results

To validate the effectiveness of the proposed reinforcement learning-based energy-efficient protocol, extensive simulations were conducted. Network simulator 3 was used for this purpose. The number of nodes varied from 20 to 100 in increments of 20. The simulation environment mimics the real-world scenario of border surveillance using WSNs. Table 2 shows the simulation parameters.

Rationale for Simulation Parameters: The simulation parameters listed in Table 2 were carefully selected based on widely accepted values in WSN simulations and reinforcement learning literature to ensure a balance between energy efficiency, learning convergence, and practical deployment feasibility.

The learning rate (α = 0.1) was chosen to provide a moderate balance between convergence speed and stability. A higher learning rate might cause instability, whereas a lower value may slow down convergence, especially in dynamic environments. The discount factor (γ = 0.9) prioritizes long-term rewards, which is essential in maximizing network lifetime over many rounds.

The range of node numbers (20 to 100) was selected to study the scalability of the proposed method under both sparse and dense deployment conditions. This range captures key behaviors such as reduced connectivity at low densities and increased contention at higher densities.

The network area of 1000 m × 1000 m reflects a realistic border surveillance region. The fixed sink node at (0,0) simplifies routing evaluation and ensures consistent baseline comparison across trials.

The transmission energy per bit (E_{elec} = 50 nJ/bit) aligns with standard energy models in WSN literature. Similarly, the values for free-space (ϵ_{fs} = 10 pJ/bit/m^2) and multipath fading (ϵ_{mp} = 0.0013 pJ/bit/m^4) models are adopted from the first-order radio model commonly used in energy analysis. The threshold distance d_0 = 87 meters is derived from these two values and represents the switching point between free-space and multipath channel models.

The exploration rate (ϵ = 0.1) was chosen to ensure sufficient exploration of actions while progressively favoring exploitation as learning stabilizes. A very high ϵ would lead to suboptimal policy learning due to excessive randomness, whereas a very low value might trap the agent in local optima.

A fixed packet size of 512 bytes was used to simulate uniform data reporting intervals, ensuring fairness in comparison across protocols.

Impact of Varying Parameters: Altering these parameters would affect the performance and convergence behavior of the protocol. For example, increasing the packet size would lead to higher energy consumption per transmission, possibly reducing network lifetime. A larger exploration rate could result in poor convergence of the Q-values, while a smaller one may prevent sufficient policy space exploration. Increasing the number of nodes without proportionate energy or range adjustments could lead to increased collisions and delay. Likewise, changing the energy model parameters would significantly affect transmission costs, thereby impacting the effectiveness of the learned energy-saving policies.

Hence, these values were selected to strike a practical balance between simulation tractability and realism in WSN-based border monitoring.

5.1 Performance Metrics

The performance of the proposed method is evaluated using the following metrics:

- **Packet Delivery Ratio (PDR):** The ratio of packets successfully delivered to the sink to the total packets sent.
- **Average Delay:** The average time taken for a packet to reach the sink node from the source node.
- **Average Remaining Energy after 2000 Rounds:** The average energy remaining in the sensor nodes after 2000 rounds of data transmission.
- **Throughput:** The total number of packets successfully delivered to the sink node per unit time.
- **Network Lifetime:** The time until the half of the nodes deplete their energy.

5.2 Simulation Implementation Details

To simulate the proposed reinforcement learning-based energy-efficient protocol, the following step-wise procedure was followed using NS-3.44 on Ubuntu 24:

1. **Simulation Setup:**

- A custom NS-3 simulation script titled `rl1.cc` was developed.
- IEEE 802.15.4 MAC/PHY layer was integrated via `LrWpanNetDevice`.
- 6LoWPAN and Internet Stack modules were used for IP-based communication.

2. **Application Layer Implementation:**

- A custom application class named `RlApp` was created, inheriting from `ns3::Application`.
- Each sensor node was equipped with:
    - A socket bound to the sink node address for packet delivery.
    - A reference to its `BasicEnergySource` object.
    - Initialization of energy level and unique node ID for tracking.

3. **Reinforcement Learning Logic:**

- Q-learning was used with discrete state-action representation.
- **State Space:** Discretized based on remaining energy as a fraction of initial energy.
- **Action Space:** Includes abstract transmission behavior actions.
- **Reward Function:** Designed to maximize utility using:
    (a) Negative energy consumption,
    (b) Coverage efficiency (detection probability and sensing radius),
    (c) Connectivity awareness (number of neighbors in communication range).
- Q-table is updated at each transmission step using standard temporal difference (TD) learning.

4. **Energy Consumption Modeling:**

- Instead of using `LrWpanRadioEnergyModel`, energy consumption was manually managed.
- A fixed transmission energy cost of 50 nJ/bit was deducted from the node's remaining energy.
- For each packet of 512 bytes, energy consumed was computed as:

$$E_{\text{tx}} = k \times 8 \times 50 \times 10^{-9} \tag{16}$$

where k is the packet size in bytes.

5. **Packet Logging and Debugging:**

- All transmissions and receptions were logged via `std::cout`, showing packet size, node ID, and energy state.
- Guard checks were added to avoid pointer dereferencing errors after simulation ends.
- The simulation was configured to stop gracefully upon encountering invalid pointer states.

5.3 Results and Discussion

The simulation results are compared with traditional methods [17] to demonstrate the advantages of the proposed Q-learning-based protocol. To provide clarity, the discussion is organized into subsections based on five key performance metrics.

5.3.1 Packet Delivery Ratio (PDR)

Fig. 2 shows the Packet Delivery Ratio (PDR) for increasing numbers of nodes. As the number of nodes increases, the PDR initially increases due to better connectivity and redundancy, as more nodes provide more alternative paths for data packets to reach the sink. However, with further increases in the number of nodes, the PDR decreases due to increased collisions and congestion, which occur when too many nodes are transmitting simultaneously. The proposed method outperforms the existing method in all scenarios mainly because of the consideration of coverage and connectivity in the calculation of the reward function. Another reason for this improvement is the dynamic adjustment of node activity and transmission power, which helps in reducing packet collisions and ensuring efficient data transmission by adapting to the current network conditions, optimizing routes, and managing the load on individual nodes.
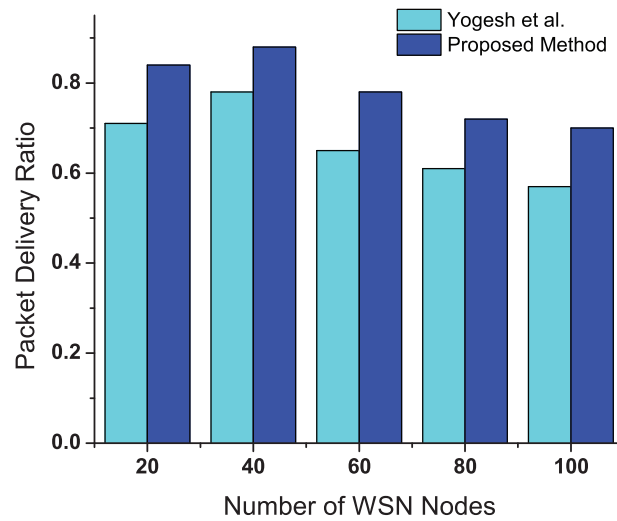
Figure 2: Packet Delivery Ratio (PDR) for different numbers of nodes compared with [17]

5.3.2 Average Delay

Fig. 3 illustrates the average delay experienced in the network. As the number of nodes increases, the average delay initially decreases due to improved path availability to the sink and reduced hop counts. However, beyond a certain point, the delay starts to increase due to higher network traffic and the network becomes congested with too many nodes attempting to communicate simultaneously. The proposed method consistently achieves lower delays compared to existing method because it optimizes the communication paths using coverage and connectivity terms, which further consider important parameters such as link quality, distance to the sink, residual energy, and node density. These optimizations ensure that data packets take the most efficient routes, avoiding congested or low-quality links, thereby reducing overall network delay.
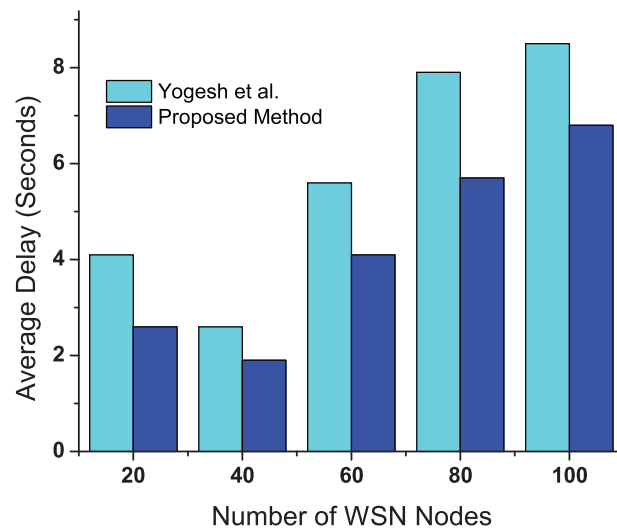
Figure 3: Average delay for different numbers of nodes compared with [17]

5.3.3 Average Remaining Energy after 2000 Rounds

Fig. 4 depicts the average remaining energy in the sensor nodes after 2000 rounds. As the number of nodes increases, the energy consumption is more evenly distributed, leading to higher average remaining energy across the network. The proposed method shows higher remaining energy compared to the existing method due to its efficient energy management strategies, such as adaptive transmission power adjustment, which optimizes energy usage based on node location, remaining battery life and several other parameters. Additionally, the method utilizes dynamic node activity scheduling to ensure that only necessary nodes are active at any given time, further conserving energy. This comprehensive approach to energy efficiency ensures that the sensor nodes maintain higher energy levels over prolonged periods, thereby extending the overall network lifetime and improving its sustainability.
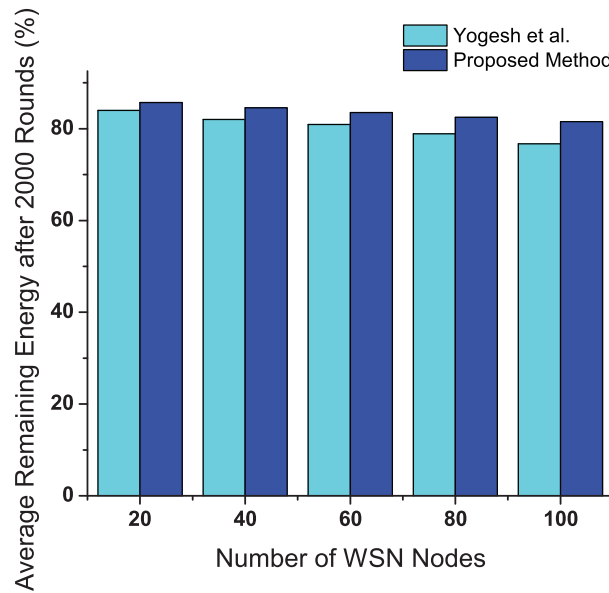
Figure 4: Average remaining energy after 2000 rounds for different numbers of nodes compared with [17]

5.3.4 Throughput

Fig. 5 shows the throughput of the network for different numbers of nodes. As the number of nodes increases, the throughput increases continuously due to the higher volume of data being transmitted. The proposed method achieves the higher throughput compared to the existing method. This improvement is primarily due to the consideration of coverage and connectivity in the calculation of the reward function, which ensures optimal routing and efficient utilization of network resources. Additionally, the dynamic adjustment of node activity and transmission power reduces packet collisions and congestion, leading to a more stable and higher data transmission rate. The adaptive nature of the proposed method allows it to handle increased network traffic effectively by optimizing communication paths based on real-time network conditions. This approach ensures that the network maintains high throughput even as the number of nodes grows.
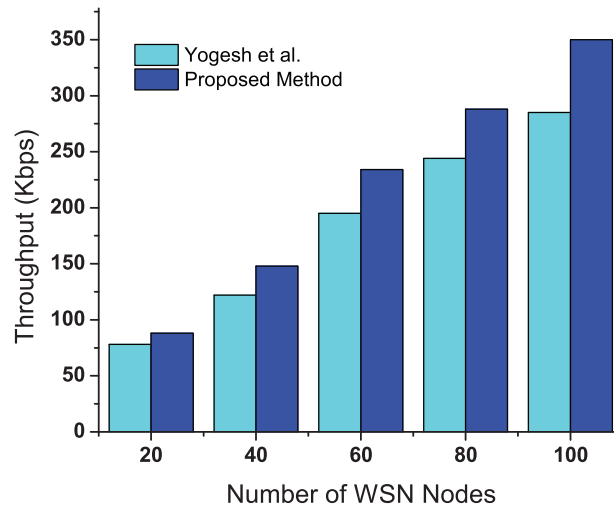
Figure 5: Throughput for different numbers of nodes compared with [17]

5.3.5 Network Lifetime

Fig. 6 presents the network lifetime for increasing numbers of nodes. Network lifetime is taken as the time since the starting of the simulation to the time at which the first node becomes dead. The proposed method significantly extends the network lifetime compared to existing method. This improvement is due to several factors. Firstly, the reinforcement learning-based approach continuously adapts to the network's state, optimizing node activity and communication patterns. Dynamic transmission power adjustments reduce unnecessary energy expenditure. Further, the reward function incorporates coverage and connectivity, balancing operational efficiency and energy conservation. This balance prevents overuse of certain nodes and distributes energy consumption evenly across the network. Then the adaptive node activity control prioritizes nodes with higher energy levels for data transmission, while nodes with lower energy levels are put into sleep mode or assigned less frequent transmission duties, conserving energy. Finally, optimal path selection avoids energy-depleted nodes and chooses the most energy-efficient routes, maintaining connectivity and performance over a longer period. The proposed method's ability to learn from real-time conditions ensures the network remains functional for an extended duration, showcasing the effectiveness of the reinforcement learning-based protocol.
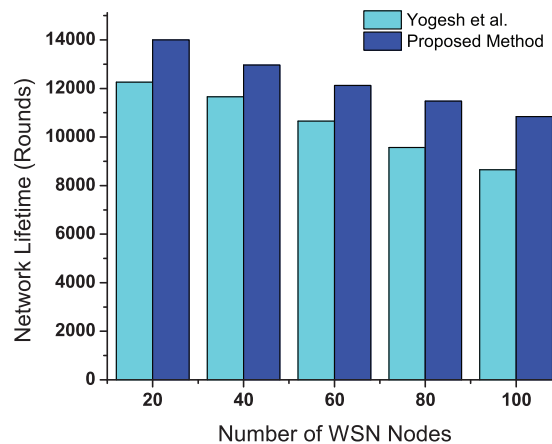

Figure 6: Network lifetime for different numbers of nodes compared with [17]

6 Conclusion

A reinforcement learning-based energy-efficient protocol is designed in this work to enhance the operational lifetime of WSNs for border surveillance. By leveraging the adaptive capabilities of reinforcement learning, the protocol dynamically optimizes node activity and communication patterns, ensuring optimal energy consumption without compromising coverage, connectivity, or data accuracy. Simulation results demonstrate significant improvements, with a 9.75% increase in network lifetime, an 8.85% boost in throughput, and a 9.45% reduction in average delay compared to recent similar protocols. This percentage improvement for each parameter is calculated using averaging the percentage improvement for each value of the corresponding parameters. These improvements directly translate to enhanced operational longevity and reliability in real-world border surveillance deployments. For instance, an increase in network lifetime reduces the frequency of battery replacement or re-deployment missions, which is particularly valuable in inaccessible or high-risk areas. Higher throughput and lower delay ensure faster and more reliable detection and reporting of security breaches or unauthorized intrusions along borders. Despite its promising performance, the proposed approach has certain limitations. First, the simulation assumes static sensor nodes and a fixed sink node; mobility and environmental dynamics have not yet been considered. Second, the DQN algorithm, although lightweight, may still present computational overhead for resource-constrained sensor nodes if scaled beyond moderate network sizes. Lastly, current evaluation is limited to simulation environments; real-world hardware implementations and field validations are required to assess robustness under actual deployment conditions. Future research will focus on incorporating advanced reinforcement learning techniques and addressing practical deployment challenges to further enhance the robustness and scalability of the protocol for real-world border surveillance applications. This approach offers a promising solution to revolutionize border surveillance operations.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Nimisha Rajput, Nishu Gupta and Raghavendra Pal; methodology, Nimisha Rajput; software, Raghavendra Pal; validation, Mikko Uitto, Jukka Mäkelä and Nishu Gupta; formal analysis, Nimisha Rajput; writing—original draft preparation, Amit Kumar; writing—review and editing, Amit Kumar and Raghavendra Pal; project administration, Raghavendra Pal; funding acquisition, Raghavendra Pal. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The related code can be found at https://drive.google.com/drive/folders/1sO5UJWo6V1hmCPs64zUSoKNK7Be9FDAU?usp=sharing (accessed on 2025 May 22).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Han CC, Kumar R, Shea R, Srivastava M. Sensor network software update management: a survey. Int J Netw Manag. 2005;15(5):283–94. doi:10.1002/nem.574.
2. Singh S, Anand V. Load balancing clustering and routing for IoT-enabled wireless sensor networks. Int J Netw Manag. 2023;33(5):e2244. doi:10.1002/nem.2244.

3. Kagade RB, Jayagopalan S. Optimization assisted deep learning based intrusion detection system in wireless sensor network with two-tier trust evaluation. Int J Netw Manag. 2022;32(4):e2196. doi:10.1002/nem.2196.

4. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor networks: a survey. Comput Netw. 2002;38(4):393–422. doi:10.1016/s1389-1286(01)00302-4.

5. Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. Comput Netw. 2008;52(12):2292–330. doi:10.1016/j.comnet.2008.04.002.

6. Donta PK, Monteiro E, Dehury CK, Murturi I. Learning-driven ubiquitous mobile edge computing: network management challenges for future generation Internet of Things. Int J Netw Manag. 2023;33(5):e2250. doi:10.1002/nem.2250.

7. Pantazis NA, Nikolidakis SA, Vergados DD. Energy-efficient routing protocols in wireless sensor networks: a survey. IEEE Communicat Surv Tutor. 2013;15(2):551–91. doi:10.1109/SURV.2012.062612.00084.

8. Kumar DP, Amgoth T, Annavarapu CSR. Machine learning algorithms for wireless sensor networks: a survey. Inform Fusion. 2019;49(4):1–25. doi:10.1016/j.inffus.2018.09.013.

9. Mostafaei H, Chowdhury MU, Obaidat MS. Border surveillance with WSN systems in a distributed manner. IEEE Syst J. 2018;12(4):3703–12. doi:10.1109/jsyst.2018.2794583.

10. Zhao G. Wireless sensor networks for industrial process monitoring and control: a survey. Netw Prot Algor. 2010;2(1):46–73. doi:10.5296/npa.v3i1.580.

11. Ibrahim SW. A comprehensive review on intelligent surveillance systems. Commun Sci Technol. 2016;1(1):7–14. doi:10.21924/cst.1.1.2016.7.

12. Sami DG, Chun S. Strengthening health security at ground border crossings: key components for improved emergency preparedness and response—a scoping review. Healthcare. 2024;12(19):2416. doi:10.3390/healthcare12191968.

13. Bhadwal N, Madaan V, Agrawal P, Shukla A, Kakran A. Smart border surveillance system using wireless sensor network and computer vision. In: International Conference on Automation, Computational and Technology Management (ICACTM). London, UK; 2019. p. 183–90. doi:10.1109/ICACTM.2019.8776749.

14. Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, et al. Human-level control through deep reinforcement learning. Nature. 2015;518(7540):529–33. doi:10.1038/nature14236.

15. Alsheikh MA, Lin S, Niyato D, Tan H. Machine learning in wireless sensor networks: algorithms, strategies, and applications. IEEE Communicat Surv Tutor. 2014;16(4):1996–2018. doi:10.1109/COMST.2014.2320099.

16. Wang J, Yang W, Liu J, Ding M. A survey on energy-efficient data aggregation routing in wireless sensor networks. ACM Transact Sensor Netw. 2015;11(3):1–41. doi:10.1145/1387663.1387666.

17. Tripathi Y, Prakash A, Tripathi R. A novel slot scheduling technique for duty-cycle based data transmission for wireless sensor network. Digit Communicat Netw. 2022;8(3):351–8. doi:10.1016/j.dcan.2022.01.006.

18. Sharma Y, Ahmed G, Saini D. Uneven clustering in wireless sensor networks: a comprehensive review. Comput Electr Eng. 2024;120:109844. doi:10.1016/j.compeleceng.2024.109844.

19. Poornima G, Amirtharaj S, Maheswaran M, Bhuvanesh A. Levy flight based momentum search and Circle Rock Hyrax swarm optimization algorithms for energy effective clustering and optimal routing in wireless body area networks. Comput Electr Eng. 2024;118:109461. doi:10.1016/j.compeleceng.2024.109461.

20. Ram S, Kumar Ram S, Deepan. Energy-efficient adaptive sensing for cognitive radio sensor network in the presence of primary user emulation attack. Comput Electr Eng. 2023;106:108619. doi:10.1016/j.compeleceng.2023.108619.

21. Aggarwal K, Reddy G, Makala R, Srihari T, Sharma N, Singh C. Studies on energy efficient techniques for agricultural monitoring by wireless sensor networks. Comput Electr Eng. 2024;113:109052. doi:10.1016/j.compeleceng.2023.109052.

22. Saxena P, Singh Bhadauria S. Hardware implementation of fuzzy logic-based energy-efficient routing protocol for environment monitoring application of wireless sensor networks. Int J Commun Syst. 2025;38(8):e70087. doi:10.1002/dac.70087.

23. Kesavan V, Thiruppathy T, Wong WK, Ng PK. Reinforcement learning based secure edge enabled multi-task scheduling model for Internet of Everything applications. Sci Rep. 2025;15(1):6254. doi:10.1038/s41598-025-89726-2.

24. El-Shenhabi AN, Abdelhay EH, Mohamed MA, Moawad IF. A reinforcement learning-based dynamic clustering of sleep scheduling algorithm (RLDCSSA-CDG) for compressive data gathering in wireless sensor networks. Technologies. 2025;13(1):25. doi:10.3390/technologies13010025.

25. Khan O, Ullah S, Khan M, Chao H-C. RL-BMAC: an RL-based MAC protocol for performance optimization in wireless sensor networks. Information. 2025;16(5):369. doi:10.3390/info16050369.

26. Heinzelman WB, Chandrakasan AP, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. IEEE Transact Wirel Communicat. 2002;1(4):660–70. doi:10.1109/twc.2002.804190.