IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 28, 2022, accepted April 8, 2022, date of publication April 13, 2022, date of current version April 29, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3167058

RLBEEP: Reinforcement-Learning-Based Energy Efficient Control and Routing Protocol for Wireless Sensor Networks

ALI FORGHANI ELAH ABADI[1], SEYYED AMIR ASGHARI[1], MOHAMMADREZA BINESH MARVASTI[1], GOLNOUSH ABAEI[2], MORTEZA NABAVI[3], AND YVON SAVARIA[3], (Fellow, IEEE)

[1]Department of Electrical and Computer Engineering, Kharazmi University, Tehran 14911-15719, Iran
[2]School of Information Technology, Monash University Malaysia, Subang Jaya 47500, Malaysia
[3]Department of Electrical and Computer Engineering, Polytechnique Montréal, Montreal, QC H3A 0E9, Canada

Corresponding author: Seyyed Amir Asghari (asghari@khu.ac.ir)

ABSTRACT One of the most important topics in the field of wireless sensor networks is the development of approaches to improve network lifetime. In this paper, an energy-efficient control and routing protocol for wireless sensor networks is presented. This algorithm is based on reinforcement learning for energy management in the network. This protocol seeks to optimize routing policies to maximize the long-term reward received by each node, using reinforcement learning, which is a machine learning approach. In order to improve the lifetime of wireless sensor network, three energy management approaches have been proposed. The first approach is to navigate correctly using reinforcement learning to reduce the length of the routes and to improve energy consumption. The second approach is to exploit a sleep scheduling technique to improve node energy consumption. The last approach is used to restrict data transmission of each node based on the received data change rate. Simulation results show that in terms of network lifespan, the proposed method significantly outperforms previous reported methods.

INDEX TERMS Wireless sensor network, network lifetime, reinforcement learning, energy management, routing policy, machine learning.

I. INTRODUCTION

Nowadays, wireless sensor networks can be considered as one of the most widely used methods for collecting and analyzing environmental data. Due to the energy limitations of the sensor nodes in these networks, using an optimal method for routing and controlling wireless sensor networks can be effective in increasing energy efficiency and network lifetime. In this paper, the reinforcement learning [1]–[4] technique is used to find the optimal routing and control procedures [5]–[7].

Wireless sensor networks are widely used for measuring, collecting, and analyzing environmental data in applications in areas such as agriculture, medicine, industry, as well as monitoring access environments. Typical networks comprise several nodes, most of which embed a sensor, a battery, some memory, and a microcontroller. The energy consumed by each node is supplied by its battery. Conserving energy and increasing battery life using approaches including energy management and recharging the battery is done using techniques such as using solar energy. Environmental data is measured using sensors, and the required information is stored in their node memory. Algorithms that allow saving energy are assumed to be executed on a suitable processing device such as a microcontroller, with each node interacting with others through existing communication mechanisms. The main node in these networks, called the Sink, can eventually obtain and maintain a global view of the desired environmental parameters by receiving all the results and aggregating them. Several widely used techniques in wireless sensor network control protocols are used to perform clustering, aggregation, compression, and sleep scheduling [8], [9]. In clustering techniques, in each region, a node is selected a cluster head, and this node collects data of surrounding nodes and then sends it to the sink node to exchange data through the cluster head [10], [11]. Aggregation and compression techniques are used to reduce the amount of data sent over the network with this approach. This reduces data storage required in leaf

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaojie Su.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

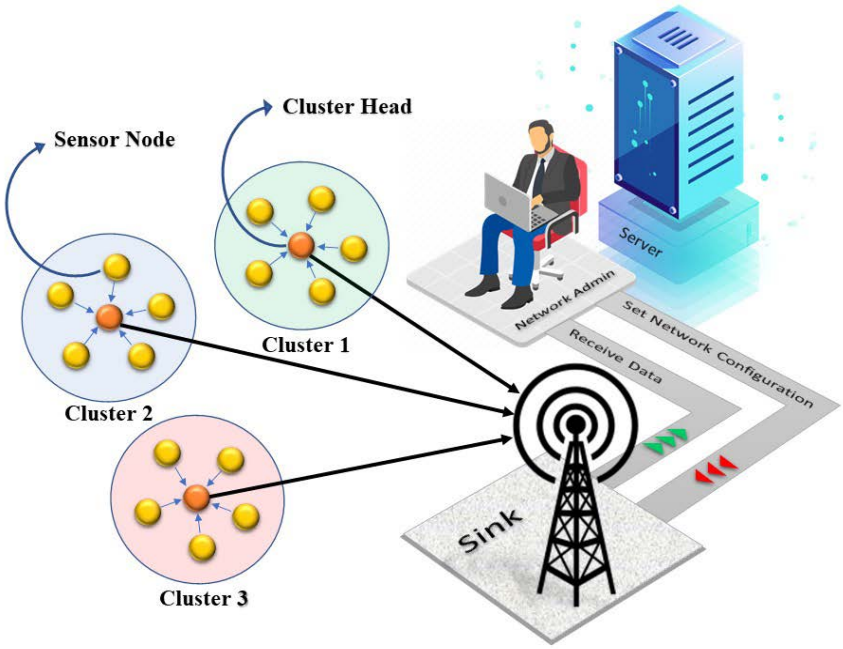

FIGURE 1. Wireless sensor networks general structure.

nodes. By disabling unused nodes using sleep scheduling techniques, further energy can be saved, which can lead to increase network lifetime [12]. The amount of saved with this method can be calculated from equation (1).

$$\begin{aligned} &\textit{Total amount of energy saving} \\ &= \sum_{i=0}^{\textit{Number of Nodes}} (\textit{Total Sleep Time}_i \\ &\quad \times \textit{Energy Consumption Constant}) \end{aligned} \tag{1}$$

When a network is subject to changing conditions, its administrator can adjust the network configuration and can analyze the data received from the server. These concepts are illustrated in Fig.1.

Proper management of energy consumption is one of the important issues in the development of wireless sensor networks. This has a direct impact on the lifetime and sustainable level of activity of sensor network systems. In general, four categories of techniques are used to improve the energy efficiency of sensor network systems. These techniques include designing the optimal topology for the network [13], using an appropriate routing method [14], apply sleep scheduling techniques [12], and finally use high-quality hardware nodes [15]. Interaction of these techniques at the system level is shown in Fig.2.

Reinforcement learning is one of the learning methods used in this field. Based on Markov's decision process [16], it tries to assign a suitable amount of reward (or punishment) to each agent by considering a reward for each action, so that the agent learns what to do in each state in order to maximize the total accumulated reward [17], [18]. With this method, there is a parameter, called the discount factor, that has a value between zero and one indicating that the reward of current stages should always affect the calculation of total reward more than the reward of expected future stages [19]. This can be expressed by Equation 2 [18].

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 r_{t+4} + \cdots = \sum_{i=t}^{\infty} \gamma^{i-t} r_{i+1} \tag{2}$$

where R_t is the amount of the received reward from time t based on future rewards. r_{t+1} indicates the received reward at time t+1 and γ is the discount factor. A typical Reinforcement Learning (RL) framework scenario is shown in Fig.3.

In this paper, an energy-efficient control and routing protocol in wireless sensor networks is presented. This algorithm is based on reinforcement learning for energy management approach in the network. This protocol seeks to optimize routing policies to maximize the long-term reward received by each node, using reinforcement learning, which is a machine learning approach. In addition, the proposed method uses the sleep scheduling technique and limits sending rate in nodes with low energy requirements. Notably, transmission can be restricted to changes in sensor values. The innovation of this paper is in fact the integration of three innovative techniques including sleep scheduling, data transmission restriction (data fusion) and packet routing using reinforcement learning.

This paper is organized as follows: a discussion on related works is presented in Section 2. A detailed description of the proposed method is given in Section 3.Simulation results validating the proposed method are given in Section 4. Finally, conclusions are presented in Section 5.

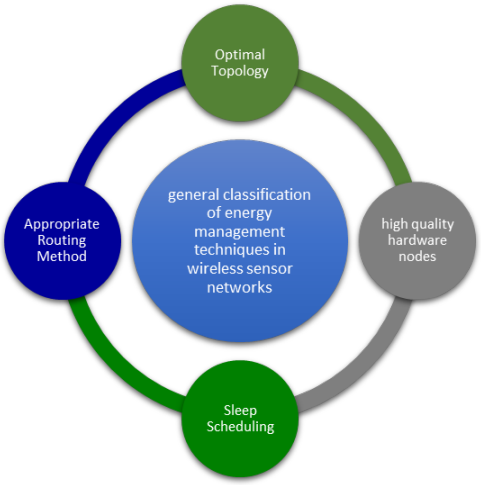

FIGURE 2. General classification of energy management techniques in WSN.

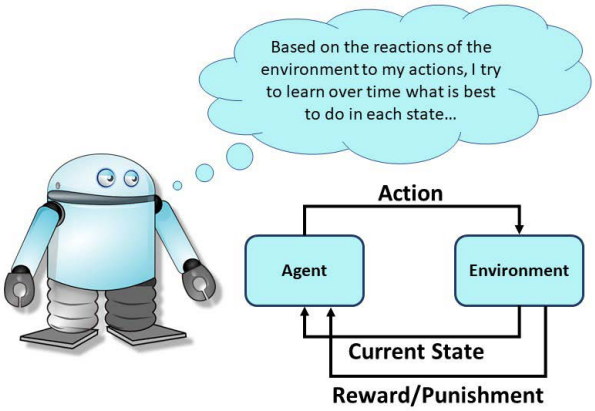

FIGURE 3. A typical Reinforcement Learning (RL) scenario.

II. RELATED WORKS

In the past, wireless sensor network control and routing protocols have been based on static approaches. Fig.4 shows the general classification of these methods.

In many applications of wireless sensor networks, it is not possible to assign a global identifier to each node due to their large numbers. However, sensor nodes are usually randomly distributed in the environment. This process makes it difficult to select some specified nodes in order to send them dedicated commands or to communicate with them privately. Routing protocols can use data aggregation and routing based on aggregation results to improve network performance and save energy. This approach called data-centric protocols sends a query to some desired area and waits for the response data to be received [20], [21].

By contrast, hierarchical protocols use clustering techniques. The main purpose of hierarchical routing is to conserve the energy of sensor nodes by engaging them in an intragroup communication and performing aggregation to reduce the number of transmitted messages to the source node [20]–[22].

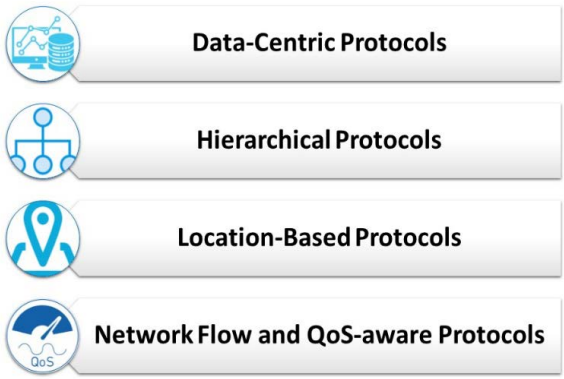

FIGURE 4. Classification of existing wireless sensor network routing protocols.

Often, location information is needed to calculate the distance between two specific nodes in order to estimate energy consumption. Because addressing procedures, such as IP, do not exist for sensor networks, spatial information can be used to route them. In another approach called location-based protocols, data can be sent to a specific location, reducing the number of data transfers significantly to reduce energy consumption. This method also supports dynamic network topology [20], [23].

Some of the previously reported routing protocols are aware of network flow and QoS. These protocols, while adjusting routes on the sensor network, take into account the delay requirements in the end-to-end transmission process [20], [24]. Routing methods in wireless sensor networks also underwent many changes when the approach of using artificial neural networks and deep learning emerged in the world. In ELDC [25], [26] Approach, back-propagation technique in neural networks have been used to select the cluster heads. Lee *et al.* [25], [27] proposed classification method for classify node degree based on deep learning. This protocols focuses on the connectivity of mobile nodes (MNs). Moon *et al.* [25], [28] proposed a cluster-ring approach for Energy efficient data collection. This approach is used to group a set of clusters into 'cluster-rings', which is a chain of clusters that are equal distance away from the sink, and conduct energy efficiency optimization at the cluster-ring level. In recent years, reinforcement learning based approaches have been designed to improve operation of wireless sensor networks as introduced below. For instance, Boyan and Littman [29] proposed the Q-Routing method based on Q-Learning. In this method, the Q-Value parameter is assigned to each action in each specific state, indicating the value of performing that action in that state. When a packet is received by a node, it is checked to find the highest Q-Value in the following nodes for sending the packet based on its neighbors. Finally, the packet is sent to the node with the maximum Q-Value. What stands out as an advantage in their method is the innovation of using the Q-Learning approach in

routing wireless sensor networks. The above method provides a simple means of using learning for routing in wireless sensor networks. This method emphasizes simplicity instead of optimality. This leaves room for significant improvements using other techniques.

Wang and Wang [30] proposed an Adaptive Routing (AdaR) method based on a combination of Least Square Policy Iteration (LSPI) and Q-Learning. This method is one of the methods based on reinforcement learnings independent of the problem model. Also, one of its features is the ability to search for the optimal policy with a small number of attempts.

Zhang and Huang [31] proposed the Learning-based Adaptive Routing Tree (ATP) method. In this method, the Q-Value parameter is calculated as the cost, and at each step, a node that has a value less than the value of other nodes is selected. Each node also stores the Q-Value parameter of its neighbors for its uses in the sending stage, depending on the appropriate neighbor, it assigns them the NQ-Value. These parameters are updated each time based on learning theory's main idea. This method is robust for un-predictable link failures and mobile sinks. One of the problems in this method is its need of hyper-parameters tuning.

Forster and Murphy [32] proposed the Feedback Routing for Optimizing Multiple Sink (FROMS) method. This method can find the optimal path for several target nodes. FROMS tries to set the limits on the number of steps for each node that must be taken to reach the target through the present node by forming a path-sharing tree.

Hu and Fei [33] proposed A Machine-Learning-Based Adaptive Routing Protocol for Energy-Efficient and Lifetime-Extended Underwater Sensor Networks (QELAR) method, this method is specifically designed for underwater wireless sensor networks that regardless of whether a node is selected as the next transmitter or not, it provides information such as residual energy and the group's average energy, and it achieves and updates these values in the list of local neighbors. In this algorithm, when the received packet is an information packet, and next node is examined. However, if the specified node is not eligible to send a packet, the packet is destroyed. However, if the specified sender is eligible to send data, then the new Q-Value values are calculated based on the current Q-Values and the action with the maximum Q-Value is selected based on them. Acknowledgment packets are used to detect unsuccessful sending in QELAR. This approach can be used in various applications and according to the reported results, this approach achieves good energy efficiency even in distributed networks. Also, due to the initial knowledge of the remaining energy distribution, it increased the network lifetime compared to other approaches.

Razzaque *et al.* [34] proposed a QoS-aware distributed adaptive cooperative routing (DACR) method. This method is used for selecting relay nodes from the Energy-aware Routing (EAR) low energy ad-hoc sensor networks protocol [35], which avoids entering the critical energy area of member nodes. DACR looks for n optimal path from end to end, i.e., the one that consumes the least amount of energy, which avoids the use of low-energy nodes to increase the network lifetime. The DACR algorithm helps identify the set of relay nodes between the source node and the intermediate nodes along the path. This set of nodes together forms the correct routes for data transfer. The DACR algorithm is based on the Ad-hoc On-Demand Distance Vector (AODV) [36] routing algorithm and makes changes to it. According to reported simulation results, this protocol improves performance compared to state-of-the-art protocols.

Kiani *et.al.* [37] proposed the FTIEE method. This method is based on the clustering idea and hierarchical routing methods. The number of clusters in this method is considered to be a constant value. The capacity of each cluster is determined by the distance of the current cluster from the cluster in which the Sink node is located. This method has used the Q-learning reinforcement to select nodes for each cluster and cluster heads. Upgrading the Q-value in this method is based on the two components of node's distance to the destination node and the remaining energy of the node. Another feature of this method is considering the tolerability of the desired shapes. It also uses a fault tolerance process that will maintain the algorithm's performance in situations such as the loss of a link in the path. This protocol has a good performance in terms of lifetime and number of delivered and lost packets. Based on the results of the simulations performed, it has been shown that this approach performs better than offline and concurrent methods such as HEED-NPF [38], LEACH [39], and EECS [40] in terms of packet delivery factor, network lifetime and delivery packet rate.

There are also QoS-aware approaches, including RescueNet [41], constraint-satisfied services composition method [42], QoS-aware distributed adaptive cooperative routing [34], that use reinforcement learning.

Renold and Chandrakala [43] proposed Multi-agent Reinforcement Learning-Based Self-Configuration and Self-Optimization Protocol for Unattended Wireless Sensor Networks (MRL-SCSO). This algorithm considers several states for each node and states that each node can be present in one particular state at any given time. Possible states include discovery, active, idle, and sleep. Also, this method, shows the importance of establishing a suitable tradeoff between exploration and exploitation, It is argued that the greedy method is suitable for this purpose and it was used to establish this tradeoff. Also, the value of the energy threshold in this method is determined as a factor of node's initial energy. This factor called eco-efficiency is set at a value of 0.5. If a node does not exceed this threshold, it will enter a low power state. Low-power nodes are managed by the algorithm to conserve energy. This algorithm divides nodes into two categories: convex and non-convex nodes. Convex nodes are determined using the Convex-Hull algorithm. These nodes are often in active mode. This method provides better QoS in terms of PDR, average end-to-end delay, and throughput. Also, initial energy consumption is higher in MRL-SCSO when compared with that of CTP.

Guo *et al.* [44] proposed the Reinforcement-Learning-Based Routing (RLBR) method. In their approach, after performing the initial network configuration, each node waits to receive a packet. Upon receiving a packet, it extracts the information and updates the table of its neighbors based on the obtained information. If there is no retrieval node near the present node, the packet is discarded. Otherwise, sink node's existence in the communication range of the present node will be checked first. If the node is in this range, the packet is sent directly to the sink node. Otherwise, if there is no suitable neighboring node, and if the node has enough energy, it tries to send the data directly to the sink node by adjusting its transmit power, but if not does not have enough energy, the packet is discarded. If there is a representative node to send data, the Q-Value parameter is obtained for all these representatives, and the node that has the highest Q-Value is selected as the next replication node. Finally, by updating the Q-Value and hop count values, as well as the packet header, it sends the packet to the next selected node. This method can also be applied to the large-scale WSNs as it can handle the routing phase in each cluster locally. It has been shown by the authors that this approach performs better compared than EAR, BEER, Q-Routing, and MRL-SCSO in terms of the proportion of live nodes, the connectivity to the sink, the number of packets delivered, and the energy efficiency.

Donta *et al.* [45] proposed the Delay-aware data fusion (DADF) method. this approach involve two phases namely hierarchical data fusion (HDF) and forwarding node selection (FDS). In HDP phase is used duplicate elimination and delete inconsistent data methods. This method manages sleep scheduling using an periodic cycle. Also, Q-Learning approach is used for routing process in this method.

Three techniques, including sleep scheduling, data transmission restriction (data fusion), and policy-based routing using Q-Learning based Algorithms, have been able to perform well. But this innovation that provides the basis for the simultaneous use of these three techniques is the method presented in this article. Table.1 compares these three techniques in the above methods.

TABLE 1. Latest Methods comparison based used techniques.

Methods	Sleep Scheduling	Data Fusion	Q-Learning
AdaR	×	×	✓
ATP	×	×	✓
FROMS	×	×	✓
QELAR	×	✓	✓
DACR	×	×	✓
FTIEE	×	×	✓
MRL-SCSO	×	✓	✓
RLBR	×	×	✓
DADF	✓	✓	✓
RLBEEP (proposed method)	✓	✓	✓

III. PROPOSED METHOD

A. OVERVIEW

The Reinforcement-Learning-Based Energy Efficient Protocol (RLBEEP) is a combination of techniques used in traditional wireless sensor network routing protocols with other modern machine learning methods. RLBEEP comprise three main phases: *routing*, *sleep scheduling*, and *restrict data transmission* as shown in Fig.5. The routing phase is based on the reinforcement-learning approach. Also, the data transmission restricting phase and sleep scheduling phase are based on traditional approaches.

B. MAIN APPROACH

In the present paper, the routing approach in the RLBR and DADF method has been used with some changes to improve its efficiency. This phase embeds the methods described below. It considers two main general scenarios. The first scenario is related to the execution procedure of each node in

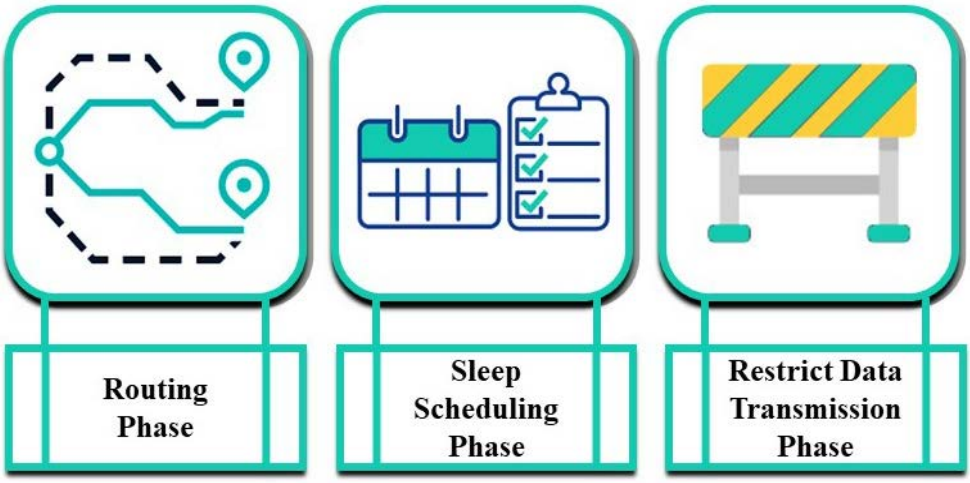

FIGURE 5. RLBEEP main phases.

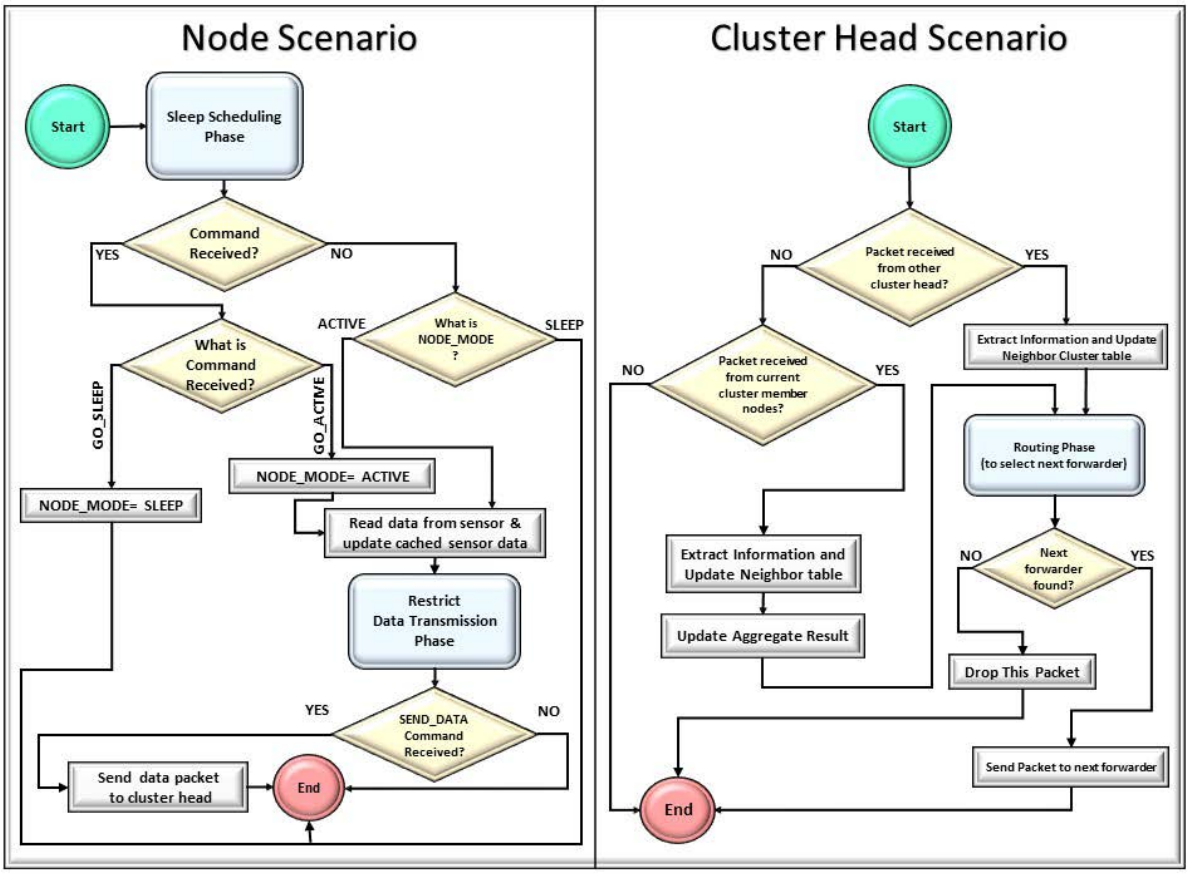

FIGURE 6. RLBEEP main scenarios in normal node and cluster head.

the network and the second scenario is related to the execution procedure in the cluster head nodes. This is in fact one of the main differences between the present method and the DADF algorithm from an architectural perspective These scenarios are illustrated in Fig.6.

In the node procedure scenario, at first, the node checks to see if a command has been sent to it by the sleep scheduling unit. If so, then the continuation of the process should be determined based on the type of command to set the node's state to sleep or active. If the received message is a change of node's state to sleep, or current node's state is equal to sleep, then it waits to receive a command from the sleep scheduling unit. If the received message is a change of node state to active or the current node state is equal to active, then the node reads data from the sensor and updates the cached sensor data. If the restricted data transmission unit is issued a data transmission license, then it sends the data packet to cluster head. IEEE 802.11 protocol is used for data transmissions between nodes in MAC layer.

In the cluster head procedure scenario, the cluster head first checks to see if it has received a packet from another cluster head or from members of its cluster, then it extracts information and updates its neighbor cluster table. In this procedure, packets received from other clusters are given more priority for processing. Also, if the packet is sent from a member of its cluster, an aggregation procedure will be performed to aggregate the new data with the previously received data. After that, it runs the routing process to find a suitable next forwarder. If this item is found, then it sends the packet to next forwarder. Otherwise, the packet is dropped. The pseudocode related to these two scenarios is given in Algorithm 1.

C. ROUTING PHASE

The approach presented in this phase is similar to the routing procedure in the RLBR algorithm, while the only difference is related to the reward function. This difference is intended to improve the learning process by better defining the reward function. The procedure for updating the Q-Value in the RLBR method is considered as Equation 3 [33].

$$Q_{new}(cur, nbr) = (1-\alpha)Q_{old}(cur, nbr) + \alpha(R(cur, nbr) + Q(nbr)) \quad (3)$$

where α is the learning rate, $Q(cur, nbr)$ represent the Q-Value of the path between the current node and the neighbor node, $R(cur, nbr)$ represents the reward received value when using this path, $Q(nbr)$ represents the Q-Value of the path from this neighbor node to the sink. $Q(nbr)$ is recursively

Algorithm 1 Proposed Method ELBEEP Method

INPUT

- *Received Packet*
- *Packet Receive Status Flag*
- *Sleep Scheduling Unit Received Command*
- *Sleep Scheduling Unit Receive Command Status Flag*

BEGIN

1. if node is cluster head
2. if packet received
3. extract informations
4. update neighbor cluster table
5. if packet received from other Cluster
6. jump to step 10.
7. else if packet received from cluster internal nodes
8. aggregates the new data with the previously received data
9. endif
10. run routing process for generate suitable next forwarder
11. if next forwarder is valid
12. send packet to next forwarder
13. else
14. drop this packet
15. endif
16. endif
17. else
18. if received command from sleep scheduling **unit**
19. if GO_SLEEP command received
20. NODE_MODE ← SLEEP
21. jump to step **1.**
22. else if GO_ACTIVE command received
23. NODE_MODE ← ACTIVE
24. go to step 32.
25. endif
26. else
27. if NODE_STATE == SLEEP
28. jump to step **1.**
29. else if NODE_STATE == ACTIVE
30. jump to step 32.
31. endif
32. end if
33. run restrict data transmission process
34. if SEND_DATA commmand received
35. rend data from sensor
36. update cached sensor data
37. send data packet to current cluster head
38. end if
39. end if

END

calculated from Equation 4 [44].

$$Q(cur) = max_{nbr \epsilon N} Q(cur, nbr) \tag{4}$$

The method of calculating the R(cur, nbr) is given in Equation 5 [45].

$$R(cur, nbr) = \frac{E(nbr)}{d^n(\text{cur, nbr}) \times \text{h(nbr)}} \tag{5}$$

where $E(nbr)$ represents residual energy of the neighboring node, $h(nbr)$ represent the hop count from the neighboring node to sink and $d(cur, nbr)$ represents the distance between the current node and the neighboring node. $h(nbr)$ is recursively calculated from Equation 6 [44].

$$h(cur) = h(nbr) + 1 \tag{6}$$

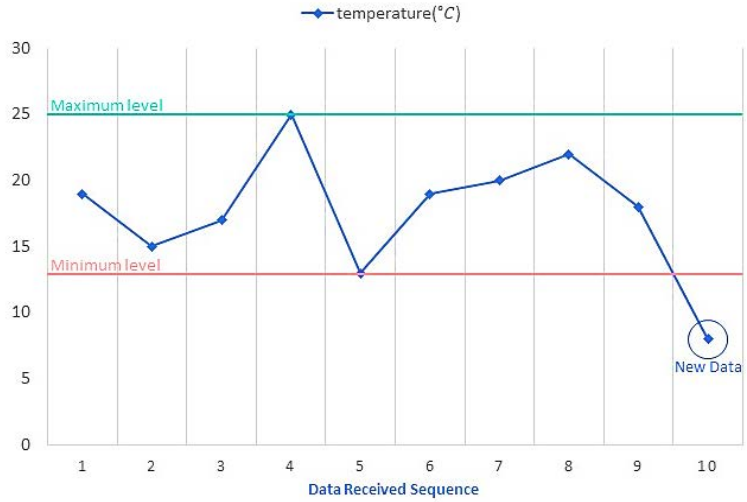

FIGURE 7. Temperature data received sequence.

The distance between the current node and the neighbor node is calculated from Equation 7 [44].

$$d(cur, nbr) = \sqrt{(x(nbr) - x(cur))^2 + (y(nbr) - y(cur))^2} \tag{7}$$

In this method, the procedure for calculating the parameter n is different from the RLBR method. It uses the normalized distance and distance factor range. The MND_{lon} means the maximum longitudinal network distance and MND_{lat} means the maximum latitudinal network distance. Calculation of the normalized distance is shown in Equation 8.

$$normalized\_distance(cur, nbr) = \frac{d(cur, nbr)}{\max(MND_{lon}, MND_{lat})} \tag{8}$$

Distance factor range is symbolized by DFR_{min} and DFR_{max}. Finally, n parameter is calculated from Equation 9.

$$n = (normalized\_distance(cur, nbr) \times (DFR_{MAX} - DFR_{min})) + DFR_{min} \tag{9}$$

D. RESTRICT DATA TRANSMISSION PHASE

The purpose of this phase is to control the data transmission rate based on a data-driven approach. In many cases, the changes in the information received from a sensor in the network are very small and may even be due to noise. Therefore, it is not always necessary to send these changes to the sink node accurately. This unit tries to manage the rate of transmitted data by examining the number of changes in the data received from the sensor related to each node in such a way that only useful data is transmitted to the sink node. This approach can reduce the energy consumption in each node. The method of detecting significant changes in the data received from the sensors according to Fig.7 is described below.

Algorithm 2 Restrict Data Transmission Unit Algorithm

INPUT
- *Received Sensor Data*
- *Change Threshold*

OUTPUT
- *Permission to Send Data Trigger*

BEGIN

```
if first run after change state to active
    min = MAX_INT
    max = MIN_INT
end if
if received_sensor_data < min
        min = received_sensor_data
else if received_sensor_data > max
        max = received_sensor_data
end if
if [((max-received_sensor_data)
        > CHANGE_THRESHOLD)
  or ((received_sensor_data-min)
        < CHANGE_THRESHOLD) ]
      permission to send data trigger = true
else
      permission to send data trigger = false
end if
```

END

Each time the node goes out of sleep and is activated, the minimum and maximum values received from the sensor are calculated until the data is sent. If the new received value is smaller than the current maximum or greater than the current minimum by the specified threshold, it means that changes made to the sensor values are significant. As a result, the current data will be sent to the sink node. The mentioned threshold is determined based on the noise level of the node, the appropriate measurement unit of the sensor, the percentage of sensor error, and other parameters that may be considered depending on the application. This approach is formalized in Algorithm.2.

Algorithm 3 Sleep Scheduling Unit Algorithm

INPUT

- ***Sending Permission Status***
- ***Sleep Restrict Repeat Threshold***

OUTPUT

- ***New Node Mode State***

BEGIN

```
1.  if node in active mode
2.      if not assign sending permission to node
3.          counter = counter + 1
4.          if counter == sleep_restrict_repeat_treshold
5.              change node state to sleep mode
6.          end if
7.      end if
8.  else if node in sleep mode
9.      if sleep interval finish
10.             change node state to active mode
11.     endif
12. endif
```

END

E. SLEEP SCHEDULING PHASE

This phase, manages the energy of the nodes in the network and tries to reduce energy consumption of the nodes, thus increasing the lifetime of network by using some appropriate control method to change the nodes state between sleep or active. In this approach, the head of each cluster never changes to the sleep state. Also, at certain intervals, the head of each cluster state changes. This is because the amount of data flowing in the cluster head is more than that of other nodes in the cluster. Therefore, by periodically changing the heads of clusters, energy consumption is spread more evenly. The sleep scheduling unit in this method tries to manage the nodes state by using the decisions of the unit to restrict data transmission. This function actually puts to sleep nodes in the network that have not sent data for some time. In order to prevent consuming energy in vain and to allow waking up after a certain suitable period.

IV. SIMULATION RESULTS

A. SIMULATION INTERFACE

RLBEEP was simulated with the Python language. Python provides implementations of the most popular libraries of data science and machine learning algorithms, as well as various tools for visualizing the results It is a very effective platform for simulating RLBEEP. In the performed simulations, a NumPy library is used to properly process and structure the data.

B. SIMULATION METRICS

Our measurement criteria in the simulation are selected in such a way that we can use them to properly measure the network lifetime and to show that the proposed method can be considered an efficient protocol for controlling wireless sensor networks. Our first performance measurement metric is the time of death of the first node in the sensor network.

This parameter indicates how long the network has been able to keep all the nodes alive based on each of the approaches. The second measurement parameter in this simulation is the change rate of the live nodes' percentage in the sensor network.

C. SIMULATION SETUP

The proposed protocol is simulated by considering certain hyper-parameters. These hyper-parameters include items such as the number of network nodes, the number of clusters, the permissible interval of point-to-point transmission of data packets, the initial energy level of nodes, the learning rate coefficient (alpha), the energy consumption in different cases, the maximum longitudinal and latitudinal interval of nodes positions, the simulation time and the number of iterations of the simulation (epoch). The values of these hyper-parameters are given in Table.2.

In the present simulation, wsn-indfeat-dataset [46] published wsn dataset is used to provide sensors data.

D. RESULTS AND COMPARISON APPROACHES

In this section, we present the results of the simulations performed to test the proposed approach. Here we simulate three methods. The first method is the RLBR algorithm, second method is the DADF method and the third method is proposed method (RLBEEP). We examined the death time of the first node in the network and found that our proposed method offers a good performance improvement compared to the RLBR and DADF method. Simulation results in 300 epochs (periods) are reported in Fig.8.

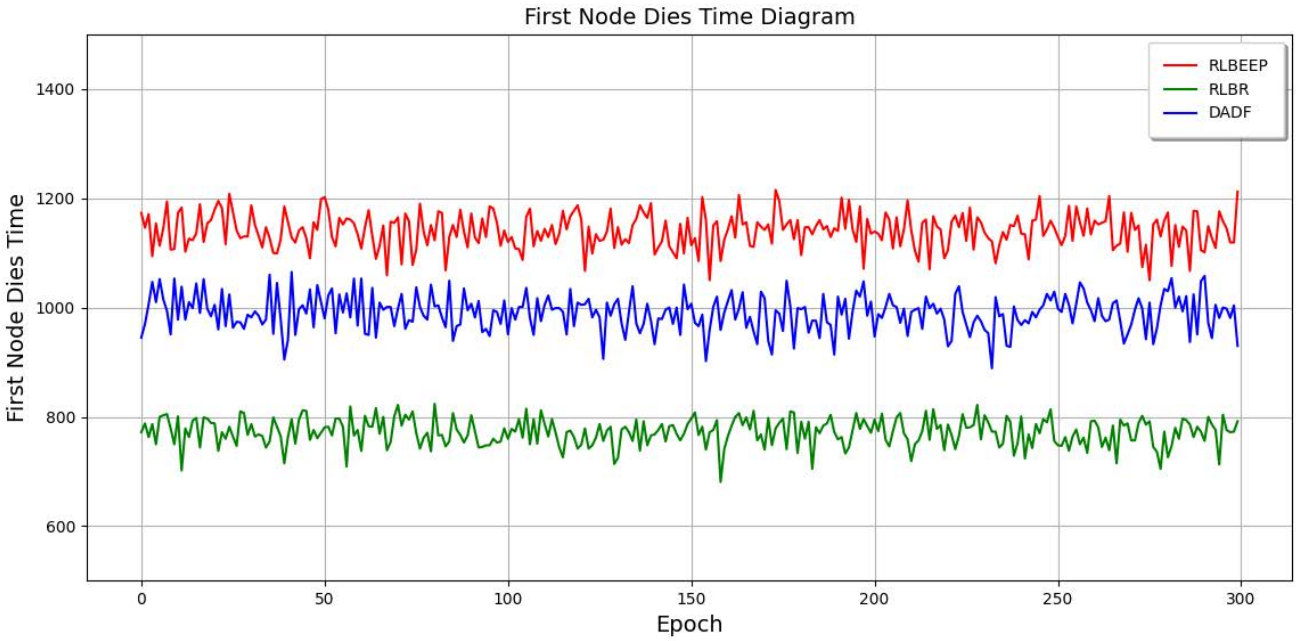

FIGURE 8. First node death time diagram.

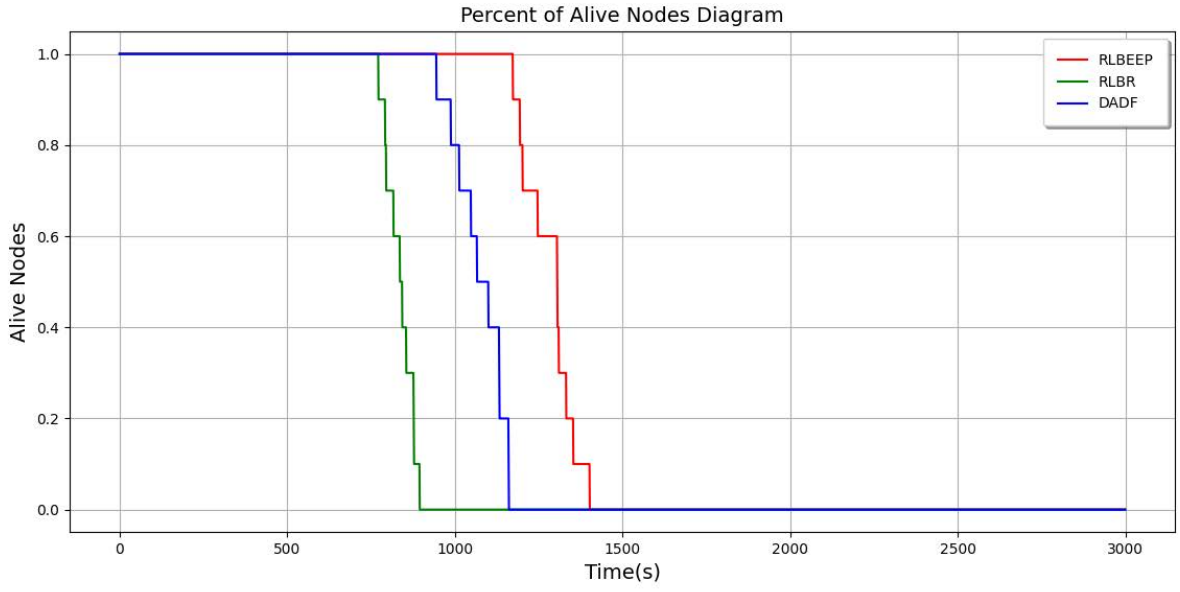

FIGURE 9. Percent of alive nodes diagram.

As shown in Fig.8, the time of death of the first node in the proposed RLBEEP method was more than those obtained with the reference method includes RLBR and DADF methods.

Another parameter that was examined during the simulations, as mentioned earlier, is the percentage of alive nodes in the network over time. An alive node at any given moment is a node whose energy is greater than zero at that moment. The performance of the proposed RLBEEP in terms of percentage of alive nodes is reported and compared with RLBR and DADF methods in Fig.9.Also, in Fig.10, the rate of change the network throughput in these methods are compared.

By achieving the optimal policy by using reinforcement learning and new rules in calculating the reward function, the routing performance has been improved and the data transmission path has been shortened. Also, by using two light techniques in terms of processing load, including sleep scheduling and restrict data transmission (data fusion), the energy loss of nodes in the network is prevented. As a result of the above, the lifetime of the network has increased significantly.

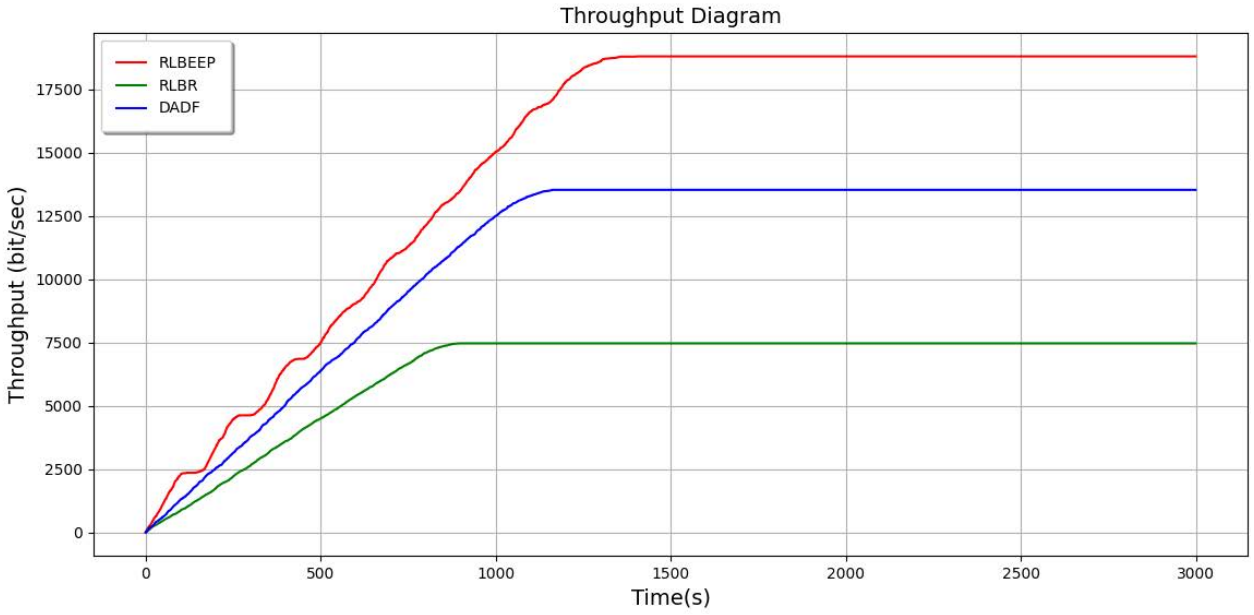

FIGURE 10. Throughput diagram.

TABLE 2. Simulation setup hyper-parameters value.

Hyper-Parameters	Value
Number of Nodes	10
Number of Clusters	4
Send Distance Range	10 m
Node Initial Energy	100 J
Alpha	0.5
Power consumption in send	0.3 J
Power Consumption in receive	0.2 J
Power Consumption in Active (Standby State)	0.1 J
Power Consumption in Sleep	0.05 J
Maximum Longitude	60.0 m
Maximum Latitude	60.0 m
DFR_{min}	5.0
DFR_{max}	55.0
Total Simulation Time	3000 s
Number of Epoch (Period)	300

As observed in this section, we showed that the proposed method has been able to improve performance both in terms of increasing the time to death of the first node and also increasing the survival time of nodes in the network compared to the RLBR and DADF methods. Therefore, the proposed method increases significantly the lifetime of the network that is an important metric in wireless sensor networks compared to the RLBR and DADF methods. This method Integrating three approaches, including reinforcement learning-based routing, node sleep scheduling, and restrict data transmission based on data changes, makes the current approach different from other recent approaches and more efficient.

While the present method increases the network lifetime, but this method requires a powerful processor in the sink node to go through the learning process to achieve the optimal policy. This requirement is in fact one of the limitations of the proposed method.

V. CONCLUSION

In this paper, we proposed the RLBEEP method to increase the lifetime in wireless sensor networks. The proposed method was shown to increase the network lifetime compared to the RLBR and DADF methods that is known the best reinforcement learning-based approach in wireless sensor networks power consumption reduction. The method combines known energy management approaches with learning-based algorithms as well as network transmission management procedures. This approach reduce computational load by simplify the architecture of sleep scheduling techniques and restrict data transmission (data fusion). Also, Improve the process of creating optimal policy in reinforcement learning for better routing. Finally, we showed that the proposed method significantly improves the time of death of the first node as well as the percentage of alive nodes as compared to RLBR and DADF. The proposed algorithm can be enhanced by improving the reward function and other functions of the learning algorithm, as well as improving the energy management and transmission management procedures in the network.

REFERENCES

[1] N. Mazyavkina, S. Sviridov, S. Ivanov, and E. Burnaev, "Reinforcement learning for combinatorial optimization: A survey," *Comput. Oper. Res.*, vol. 134, Oct. 2021, Art. no. 105400, doi: 10.1016/j.cor.2021.105400.

[2] J. Czech, "Distributed methods for reinforcement learning survey," in *Reinforcement Learning Algorithms: Analysis and Applications* (Reinforcement Learning Algorithms: Analysis and Applications). Berlin, Germany: Springer, 2021, pp. 151–161, doi: 10.1007/978-3-030-41188-6_13.

[3] S. Pateria, B. Subagdja, A.-H. Tan, and C. Quek, "Hierarchical reinforcement learning: A comprehensive survey," *ACM Comput. Surveys*, vol. 54, no. 5, pp. 1–35, Jun. 2021, doi: 10.1145/3453160.
[4] A. T. D. Perera and P. Kamalaruban, "Applications of reinforcement learning in energy systems," *Renew. Sustain. Energy Rev.*, vol. 137, Mar. 2021, Art. no. 110618, doi: 10.1016/j.rser.2020.110618.
[5] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Comput. Netw.*, vol. 52, no. 12, pp. 2292–2330, Aug. 2008, doi: 10.1016/j.comnet.2008.04.002.
[6] M. Ahmed, M. Salleh, and M. I. Channa, "Routing protocols based on node mobility for underwater wireless sensor network (UWSN): A survey," *J. Netw. Comput. Appl.*, vol. 78, pp. 242–252, Jan. 2017, doi: 10.1016/j.jnca.2016.10.022.
[7] S. Dhiviya, A. Sariga, and P. Sujatha, "Survey on WSN using clustering," in *Proc. 2nd Int. Conf. Recent Trends Challenges Comput. Models (ICRTCCM)*, Tindivanam, India, Feb. 2017, pp. 121–125, doi: 10.1109/ICRTCCM.2017.87.
[8] D. Kandris, C. Nakas, D. Vomvas, and G. Koulouras, "Applications of wireless sensor networks: An up-to-date survey," *Appl. Syst. Innov.*, vol. 3, no. 1, p. 14, Feb. 2020, doi: 103390/asi3010014.
[9] C. Nakas, D. Kandris, and G. Visvardis, "Energy efficient routing in wireless sensor networks: A comprehensive survey," *Algorithms*, vol. 13, no. 3, p. 72, Mar. 2020, doi: 10.3390/a13030072.
[10] G. Vishnupriya and R. Ramachandran, "A survey on tree based energy efficient wireless sensor network," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Chennai, India, Jul. 2020, pp. 242–245, doi: 10.1109/ICCSP48568.2020.9182161.
[11] P. K. Mishra and S. K. Verma, "A survey on clustering in wireless sensor network," in *Proc. 11th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Kharagpur, India, Jul. 2020, pp. 1–5, doi: 10.1109/ICCCNT49239.2020.9225420.
[12] K. P. Mhatre and U. P. Khot, "Energy efficient opportunistic routing with sleep scheduling in wireless sensor networks," *Wireless Pers. Commun.*, vol. 112, no. 2, pp. 1243–1263, Jan. 2020, doi: 10.1007/s11277-020-07100-z.
[13] A. A. Sheikh and E. Felemban, "Optimal topology generation for linear wireless sensor networks based on genetic algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 1, pp. 1–10, 2020, doi: 10.14569/IJACSA.2020.0110184.
[14] M. Shafiq, H. Ashraf, A. Ullah, and S. Tahira, "Systematic literature review on energy efficient routing schemes in WSN—A survey," *Mobile Netw. Appl.*, vol. 25, no. 3, pp. 882–895, Feb. 2020, doi: 10.1007/s11036-020-01523-5.
[15] M. Hempstead, M. J. Lyons, D. Brooks, and G.-Y. Wei, "Survey of hardware systems for wireless sensor networks," *J. Low Power Electron.*, vol. 4, no. 1, pp. 1–10, 2008, doi: 10.1166/jolpe.2008.156.
[16] M. A. Alsheikh, D. T. Hoang, D. Niyato, H.-P. Tan, and S. Lin, "Markov decision processes with applications in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1239–1267, Apr. 2015, doi: 10.1109/COMST.2015.2420686.
[17] T. Yang, L. Zhao, W. Li, and A. Y. Zomaya, "Reinforcement learning in sustainable energy and electric systems: A survey," *Annu. Rev. Control*, vol. 49, pp. 145–163, Apr. 2020, doi: 10.1016/j.arcontrol.2020.03.001.
[18] S. Ravichandiran, *Hands-On Reinforcement Learning With Python*, vol. 3. Birmingham, U.K.: Packt, 2018, ch. 3, sec. 1, pp. 41–56.
[19] I. Althamary, C.-W. Huang, and P. Lin, "A survey on multi-agent reinforcement learning methods for vehicular networks," in *Proc. 15th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Tangier, Morocco, Jun. 2019, pp. 1154–1159, doi: 10.1109/IWCMC.2019.8766739.
[20] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 325–349, May 2005, doi: 10.1016/j.adhoc.2003.09.010.
[21] A. Boukerche, X. Cheng, and J. Linus, "A performance evaluation of a novel energy-aware data-centric routing algorithm in wireless sensor networks," *Wireless Netw.*, vol. 11, no. 5, pp. 619–635, Sep. 2005, doi: 10.1007/s11276-005-3517-6.
[22] N. Sabor and M. Abo-Zahhad, "A comprehensive survey of intelligent-based hierarchical routing protocols for wireless sensor networks," in *Nature Inspired Computing for Wireless Sensor Networks* (Nature Inspired Computing for Wireless Sensor Networks). Kolkata, India: Springer, Feb. 2020, pp. 197–257, doi: 10.1007/978-981-15-2125-6.
[23] A. Kumar, H. Y. Shwe, K. J. Wong, and P. H. J. Chong, "Location-based routing protocols for wireless sensor networks: A survey," *Wireless Sensor Netw.*, vol. 9, no. 1, pp. 25–72, 2017, doi: 10.4236/wsn.2017.91003.
[24] N. Sharma, B. M. Singh, and K. Singh, "QoS-based energy-efficient protocols for wireless sensor network," *Sustain. Comput., Informat. Syst.*, vol. 30, Jun. 2021, Art. no. 100425, doi: 10.1016/j.suscom.2020.100425.
[25] D. P. Kumar, T. Amgoth, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," *Inf. Fusion*, vol. 49, pp. 1–25, Sep. 2019.
[26] A. Mehmood, Z. Lv, J. Lloret, and M. M. Umar, "ELDC: An artificial neural network based energy-efficient and robust routing scheme for pollution monitoring in WSNs," *IEEE Trans. Emerg. Topics Comput.*, vol. 8, no. 1, pp. 106–114, Jan. 2020.
[27] Y. Lee, "Classification of node degree based on deep learning and routing method applied for virtual route assignment," *Ad Hoc Netw.*, vol. 58, pp. 70–85, Apr. 2017.
[28] S.-H. Moon, S. Park, and S.-J. Han, "Energy efficient data collection in sink-centric wireless sensor networks: A cluster-ring approach," *Comput. Commun.*, vol. 101, pp. 12–25, Mar. 2017.
[29] J. A. Boyan and M. L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, Dec. 1993, pp. 1–8.
[30] P. Wang and T. Wang, "Adaptive routing for sensor networks using reinforcement learning," in *Proc. 6th IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, Sep. 2006, p. 219.
[31] Y. Zhang and Q. Huang, "A learning-based adaptive routing tree for wireless sensor networks," in *Proc. IEEE 3rd Consum. Commun. Netw. Conf.*, May 2006, pp. 12–21.
[32] A. Forster and A. L. Murphy, "FROMS: Feedback routing for optimizing multiple sinks in WSN with reinforcement learning," in *Proc. 3rd Int. Conf. Intell. Sensors, Sensor Netw. Inf.*, Dec. 2007, pp. 371–376.
[33] T. Hu and Y. Fei, "QELAR: A machine-learning-based adaptive routing protocol for energy-efficient and lifetime-extended underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 6, pp. 796–809, Jun. 2010, doi: 10.1109/TMC.2010.28.
[34] M. A. Razzaque, M. H. U. Ahmed, C. S. Hong, and S. Lee, "QoS-aware distributed adaptive cooperative routing in wireless sensor networks," *Ad Hoc Netw.*, vol. 19, pp. 28–42, Aug. 2014, doi: 10.1016/j.adhoc.2014.02.002.
[35] R. C. Shah and J. M. Rabaey, "Energy aware routing for low energy ad hoc sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf. Record (WCNC)*, Orlando, FL, USA, Mar. 2002, pp. 350–355, doi: 10.1109/WCNC.2002.993520.
[36] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop Mobile Comput. Syst. Appl. (WMCSA)*, New Orleans, LA, USA, Feb. 1999, pp. 90–100, doi: 10.1109/MCSA.1999.749281.
[37] F. Kiani, E. Amiri, M. Zamani, T. Khodadadi, and A. A. Manaf, "Efficient intelligent energy routing protocol in wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 3, pp. 5–27, May 2014, doi: 10.1155/2015/618072.
[38] H. Taheri, P. Neamatollahi, M. Naghibzadeh, and M.-H. Yaghmaee, "Improving on HEED protocol of wireless sensor networks using non probabilistic approach and fuzzy logic (HEED-NPF)," in *Proc. 5th Int. Symp. Telecommun.*, Dec. 2010, pp. 193–198, doi: 10.1109/ISTEL.2010.5734023.
[39] N. G. Palan, B. V. Barbadekar, and S. Patil, "Low energy adaptive clustering hierarchy (LEACH) protocol: A retrospective analysis," in *Proc. Int. Conf. Inventive Syst. Control (ICISC)*, Jan. 2017, pp. 1–12, doi: 10.1109/ICISC.2017.8068715.
[40] V. Saranya, S. Shankar, and G. R. Kanagachidambaresan, "Energy efficient clustering scheme (EECS) for wireless sensor network with mobile sink," *Wireless Pers. Commun.*, vol. 100, no. 4, pp. 1553–1567, Jun. 2018, doi: 10.1007/s11277-018-5653-1.
[41] E. K. Lee, H. Viswanathan, and D. Pompili, "RescueNet: Reinforcement-learning-based communication framework for emergency networking," *Comput. Netw.*, vol. 98, pp. 14–28, Apr. 2016.
[42] L. Ren, W. Wang, and H. Xu, "A reinforcement learning method for constraint-satisfied services composition," *IEEE Trans. Services Comput.*, vol. 13, no. 5, pp. 786–800, Sep. 2020.
[43] A. Renold and S. Chandrakala, "MRL-SCSO: Multi-agent reinforcement learning-based self-configuration and self-optimization protocol for unattended wireless sensor networks," *Wireless Pers. Commun.*, vol. 96, pp. 5061–5079, Oct. 2016, doi: 10.1007/s11277-016-3729-3.

[44] W. Guo, C. Yan, and T. Lu, "Optimizing the lifetime of wireless sensor networks via reinforcement-learning-based routing," *Int. J. Distrib. Sensor Netw.*, vol. 15, no. 2, Feb. 2019, Art. no. 155014771983354, doi: 10.1177/1550147719833541.

[45] P. K. Donta, T. Amgoth, and C. S. R. Annavarapu, "Delay-aware data fusion in duty-cycled wireless sensor networks: A Q-learning approach," *Sustain. Comput., Informat. Syst.*, vol. 33, Jan. 2022, Art. no. 100642, doi: 10.1016/j.suscom.2021.100642.

[46] *A Dataset Containing Features Extracted From Measurements Collected From Multi-Hop WSN Deployments*. Accessed: Nov. 9, 2016. [Online]. Available: https://github.com/apanouso/wsn-indfeat-dataset

ALI FORGHANI ELAH ABADI received the B.Sc. degree in computer software engineering from Sadjad University of Technology, Mashhad, Iran, in 2019. He is currently pursuing the M.Sc. degree in artificial intelligence with Kharazmi University, Tehran, Iran

SEYYED AMIR ASGHARI received the B.Sc. degree (hardware engineering major) and the M.Sc. and Ph.D. degrees (computer architecture major) from the Amirkabir University of Technology, in 2007, 2009, and 2013, respectively. He has served as a Faculty Member with the Department of Electrical and Computer Engineering, Kharazmi University. His current research interests include fault-tolerant design and real-time embedded system design.

MOHAMMADREZA BINESH MARVASTI received the M.Sc. degree from the Department of Electrical and Computer Engineering, University of Tehran, Iran, in 2007, and the Ph.D. degree in electrical and computer engineering from McMaster University, Canada, in 2013. He has served as a Faculty Member with the Department of Electrical and Computer Engineering, Kharazmi University. His research interests include computer architecture, low-power digital design, FPGAs, approximate computing, and on-chip interconnection networks.

GOLNOUSH ABAEI received the master's degree in computer applications from the University of Mysore, India, in 2008, and the Ph.D. degree in computer science from University Technology Malaysia, Malaysia, in 2015. She is currently a Lecturer with Monash University Malaysia. Her research interests include artificial intelligence, soft computing, software testing, and software defect prediction.

MORTEZA NABAVI received the B.S. degree in computer engineering from the Amirkabir University of Technology, Tehran, Iran, in 2008, the M.A.Sc. degree in computer engineering from Boston University, Boston, MA, USA, the M.A.Sc. degree in electronics from Carleton University, Ottawa, ON, Canada, in 2012, and the Ph.D. degree from the University of Waterloo, Waterloo, ON, Canada. He joined the Research and Development Group of Sidense Company, Ottawa, where he involved in designing subthreshold one-time programmable (OTP) memories. His current research interests include subthreshold circuits and memories.

YVON SAVARIA (Fellow, IEEE) received the B.Ing. and M.Sc.A. degrees in electrical engineering from École Polytechnique Montréal, in 1980 and 1982, respectively, and the Ph.D. degree in electrical engineering from McGill University, in 1985. Since 1985, he has been with Polytechnique Montréal, where he is currently a Professor with the Department of Electrical Engineering. He is also affiliated with the Hangzhou Innovation Institute, Beihang University. He has carried out work in several areas related to microelectronic circuits and microsystems, such as testing, verification, validation, clocking methods, defect and fault tolerance, effects of radiation on electronics, high-speed interconnects and circuit design techniques, CAD methods, reconfigurable computing, applications of microelectronics to telecommunications, aerospace, image processing, video processing, radar signal processing, and the acceleration of digital signal processing. He is currently involved in several projects related to embedded systems in aircraft, radiation effects on electronics, asynchronous circuit design and testing, green IT, wireless sensor networks, virtual networks, software-defined networks, machine learning, computational efficiency, and application-specific architecture design. He holds 16 patents, has published 185 journal articles and 470 conference papers, and was the thesis advisor of 170 graduate students who completed their studies. He was the Program Co-Chairperson of NEWCAS'2018. He has been working as a consultant or was sponsored for carrying out research by Bombardier, Ciena, CNRC, Design Workshop, Dolphin DREO, Ericsson, Genesis, Gennum, Huawei, Hyperchip, Intel, ISR, Kaloom, LTRIM, Miranda, MiroTech, Nortel, Noviflow, Octasic, PMC-Sierra, Technocap, Thales, Tundra, and Wavelite. He is a member of the Regroupement Stratégique en Microélectronique du Québec (RESMIQ), the Ordre des Ingénieurs du Québec (OIQ), and CMC Microsystems Board. He is a fellow of the Canadian Academy of Engineering. In 2001, he was awarded the Tier 1 Canada Research Chair (www.chairs.gc.ca) on the designs and architectures of advanced microelectronic systems that he held until June 2015. He also received the Synergy Award of the Natural Sciences and Engineering Research Council of Canada, in 2006.