EBR-RL: Energy Balancing Routing protocol based on Reinforcement Learning for WSN

Vially Kazadi Mutombo
Soongsil University
Seoul, Korea
mutombo.kazadi@gmail.com

Sung Y. Shin
South Dakota State University
South Dakota, USA
sung.shin@sdstate.edu

Jiman Hong*
Soongsil University
Seoul, Korea
jiman@ssu.ac.kr

ABSTRACT

A Wireless Sensor Network (WSN) is a wireless network that monitors physical environment conditions through resource-constrained sensor nodes and delivers data to a sink node through the network. One of the most important constraints on sensor nodes is their limited power source, which consists of small and irreplaceable batteries. Energy conservation is thus a dominant factor in WSN. Therefore, when designing a routing protocol for WSNs, it is necessary to consider the energy constraint of sensor nodes. In this paper, we consider the energy constraint of sensor nodes and propose an Energy Balancing Routing Protocol using reinforcement learning. The performance of the proposed protocol is compared to other existing energy-efficient routing protocols and the results show that the proposed protocol performs better with regards to energy saving and network lifetime.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

Wireless Sensor Network, Energy efficiency, Reinforcement Learning

ACM Reference Format:
Vially Kazadi Mutombo, Sung Y. Shin, and Jiman Hong. 2021. EBR-RL: Energy Balancing Routing protocol based on Reinforcement Learning for WSN. In *The 36th ACM/SIGAPP Symposium on Applied Computing (SAC '21), March 22–26, 2021, Virtual Event, Republic of Korea.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3412841.3442063

1 INTRODUCTION

A Wireless Sensor Network(WSN) is a network made up of several inexpensive devices with sensing capabilities, ranging from a few to hundreds or even thousands. WSNs have been initially used for military applications to monitor battlefields. In recent years, they have received much attention and have been introduced to a wide range of applications such as smart homes, object tracking, disaster management, environmental monitoring, healthcare [1].

A WSN merges sensing, computation, and communication modules into a single tiny sensor node to make it cooperate with others together to form a WSN. Generally, a sensor node is composed of four units, namely, power unit, sensing unit, processing unit, and communication unit [2, 3]. The power module consists of a small battery that supplies power to the remaining three modules. The sensing module is responsible for sensing data from the surrounding environment, whereas the processing module carries out the computation tasks. Finally, the communication module is in charge of sending packets across the network. Logically, the power module does not consume any energy but supplies energy to other modules. The sensing module and processing module also consume negligible energy, whereas the communication module is the most energy-consuming.

Efficient energy utilization is crucially important to maintain a fully operational network for the longest period of time possible. Especially, energy-efficient routing protocols which find proper paths for transmission of sensed data are known to manage the consumption of WSN available energy. Energy-efficient routing protocols also are known to assist to extend the lifetime of WSN despite some of the limitations of sensor nodes in a network and the harsh environments [4], in which the sensor nodes are to operate.

In a typical WSN, sensor nodes can either communicate directly with the sink node or communicate with the sink node via intermediate sensor nodes using multi-hop communication. The sensor nodes far away from the sink, transmit packets through some intermediate sensor nodes to save energy. The major issue of the multi-hop approach is selecting the next hop to which data packets will be sent next. Therefore, to solve this issue, some multi-hop flat routing protocols have been studied, where sensors collaborate based on a special policy. For example, some policies can select the next hop based on the distance to sink, some can consider the residual energy or delay time, signal strength. Also, other policies can optimally combine a few considerations to reduce energy consumption or maximize the quality of service.

However, the multi-hop flat routing protocols often suffer from energy consumption, data redundancy, and network scalability [4]. Various clustering protocols have been studied to solve these problems and improve the performance of routing protocols for WSNs [4–10].

A clustering protocol divides the WSN network into small subsets of sensor nodes, with each subset forming a cluster of sensor nodes around a powerful sensor node, commonly referred to as cluster head. Each cluster head aggregates data from different sensor nodes also called cluster members and passes them to the sink node

*Corresponding author

SAC '21, March 22–26, 2021, Virtual Event, Republic of Korea

ACM ISBN 978-1-4503-8104-8/21/03.
https://doi.org/10.1145/3412841.3442063

[4, 8, 11]. Clustering protocols can provide energy efficiency and network scalability and increase network lifetime[6, 10]. However, the performance of clustering protocols varies from design and implementation, and each design and implementation have advantages and disadvantages depending on the various performance factors considered in the performance evaluation.

Therefore, in this paper, we propose EBR-RL which is an Energy Balancing Routing Protocol for WSN based on Reinforcement Learning. The proposed EBR-RL balances the energy dissipation between sensor nodes in a WSN to maximize the energy efficiency and network lifetime. It also computes the optimal paths using feedback obtained as a reward from each routing decision and updates the routing table for a better choice of the next-hop in the future. The reward function is computed using the residual energy and hop count to the sink. Furthermore, the hop count parameter used in EBR-RL can reduce the end-to-end delay time and eventually it can extend the network lifetime.

To evaluate the performance of EBR-RL, a simulation is carried out. The evaluation result shows that EBR-RL achieves an efficient energy consumption of the sensor nodes by optimally transmitting packets to the sink node at a low energy cost. EBR-RL is also compared to LEACH [15] and PEGASIS [9] and shows that it is outperformed them by providing a better energy balance and extending the network lifetime.

The remainder of this paper is organized as follows: In section 2, we give an overview of Reinforcement Learning(RL) and its application in routing, then we discuss the existing solutions on routing protocol using RL in section 3. In section 4, we describe our proposed solution. The performance evaluation of the proposed solution is presented in section 5, and finally, comes the conclusion and future work in section 6.

2 OVERVIEW OF REINFORCEMENT LEARNING(RL)

RL is another type of machine learning which addresses the problems of an agent that takes actions that map with a given environment from which it gets a reward from the action performed[12]. The obtained reward determines whether the action performed by the agent was good or bad, and the goal of an RL agent is to maximize the cumulative reward. Initially, the agent does not have any knowledge of the environment, but it learns it and discovers how to react based on some criteria called policy. In recent years, RL is applied to different areas that deal with similar problems. For example, in the routing protocol RL was introduced to make proper routing decisions and adapt to network topology and condition changes by learning independently to adapt to the change without any explicit coordination [9, 13]. In routing for WSN, RL can be useful when it comes to selecting the next-hop based on some criteria notably residual energy, signal strength, distance.

The problem of RL is described as a Markov Decision Process(MDP) with a tuple (S, A, P, R) where S represents a set of states an agent can be in at a given time t, A denotes a set of possible actions an agent can take at a given time [12]. P is transition probability that an agent is in a given state s(t) enters in state s(t+1) by performing an action a(t) at a given time t. R is the reward obtained by the agent by performing an action.

3 RELATED WORK

RL for the network routing protocol was first introduced by Michael and Justin in [13]. They proposed the Q-routing algorithm which is a delivery time-optimal solution that aimed to find routes that deliver packets from source to destination within a minimal time. The Q-routing algorithm showed that network routing network protocol is one of the natural applications of RL.

In [14], G. Oddi et al. proposed an Energy balancing in multi-hop Wireless Sensor Networks using RL. They optimized the network lifetime by balancing the sensor nodes' energy dissipation and reducing the network overhead. However, they considered only residual energy of sensor nodes, thus the protocol did not ensure a good balance for multi-hop communication in a long run.

Jafarzadeh and Moghaddam proposed EQR-RL, an energy-aware QoS routing protocol using RL in [15]. In EQR-RL, nodes periodically broadcasted Heartbeat packets that include the delivery ratio estimate and the residual energy of the sender. This information enabled each sensor nodes to compute the next hop using a probability function. The simulation results showed a good delivery ratio and low end-to-end delay, while supporting sink mobility. However, network isolation was likely to happen due to the avoidance of any sensor node from the routing table that does not respond.

In [16], Foster and Murphy proposed FROMS, Feedback Routing for Optimizing Multiple Sinks in WSN. FROMS is an RL based multiple sinks routing protocol. In FROMS, a node's local information was shared as feedback to neighboring sensor nodes without extra network overhead. FROMS also provided a recovery mechanism after node failure to deal with packet loss. However, FROMS could suffer from packets loss in case of sink mobility, especially in case of high-speed mobility as links kept dropping and new links adding. This might lead to routing errors and extra energy consumption.

4 EBL-RL : ENERGY BALANCING ROUTING PROTOCOL USING RL

4.1 Network Set-up phase

The proposed EBR-RL consists of two phases: Network Set-up Phase, and Communication Phase. In the network Set-up phase, each sensor node including the sink node broadcasts a "Heart-Beat" packet containing its initial Q-Value and location coordinate in the packet header. Also, every sensor node overhears the neighbors' packets and creates its initial routing table using the data collected from its neighboring sensor nodes. The initial Q-value in the routing table is set based only on the hop count as for a homogeneous WSN, all sensor nodes have the same communication range and the same energy level. However, if the energy level of each sensor node is different, the residual energy is also reflected in the Q-value.

$$D_{link} = D_{i,j} + D_{j,sink} \tag{1}$$

$$D_{link} = (\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2)) + (\sqrt{(x_j - x_{sink})^2 + (y_j - y_{sink})^2)) \tag{2}$$

$$N_h \cong \frac{D_{link}}{R} \tag{3}$$

To compute the hop count, the distance link and the communication range are used as presented in [4]. The distance of a sensor node S_i to the sink node via an intermediate sensor node S_j,is denoted as D_{link} computed as in Eq. 1 and Eq. 2. This distance D_{link} can also equivalent to $N_h \times R$, where N_h denoted as the hop count, and R the communication range [4]. From the aforementioned, the estimate hop count can be evaluated as in Eq. 3.

Algorithm 1: Initial routing Algorithm

```
for i ← 1 to n do
    nID=1 /* Neighbor ID                                   */
    for j ← 1 to n do
        dist = Euclidean(S(i), S(j))
        if dist ≤ R then
            Neig[i,nID].append(S_j)
            S_dist(i, nID).append(dist)
            nID ← nID + 1
        End
    End
End
for i ← 1 to n do
    E_min = min(Neig(i, :))
    E_max = max(Neig(i, :))
    for j ← 1 to n do
        if S(j) ∈ Neig(i, :) then
            for n ← 1 to len(Neig(i, :)) do
                if E_min == E_max then
                    Q(i,n) = 1/S(j).hop
                else
                    Q(i,n) =
                      p×(S(j).E−E_min)/(E_max−E_min) + (1 − p) × 1/S(j).hop
                End
            End
        End
    End
End
for i ← 1 to n do
    nID =1
    maxQ = max(Q(i,niD))
    for j ← 1 to n do
        if S(j) ∈ Neig(i, :) then
            if Q(i, nID) == maxQ then
                return S(j)
            End
            nID ← nID + 1
        End
    End
End
```

The procedure of computing the initial route is described as in Algorithm 1. The first part of the algorithm is finding the neighboring sensor nodes, any sensor node S_j within S_i communication range is considered neighbor of S_i. The second part of the algorithm is about the computation of the Q-Value of each sensor node. The third and last part find the neighbor with the maximum Q-value to be appointed as a next-hop.

4.2 Communication Phase: Application of RL

The communication phase is the most important step of the EBR-RL, discussing the learning process. The learning process consists of updating the Q-Value and finding the best policy to speed up convergence. It uses three functions such as the energy consumption model, the reward function and the update function for Q-value. The energy consumption model allows updating the residual energy by subtracting the energy dissipated for the packet transmission. The updated residual energy together with the hop count are then used to evaluate the reward. Finally, The reward is also used as an argument in the Q-value function to update the Q-value.

4.2.1 *Energy Consumption Model.*

The energy consumption model used in EBR-RL is as same as the the one proposed in LEACH [5]. When a packet is transmitted from a sensor node to another, the energy is dissipated from both the transmitter and the receiver. However,the energy consumed by the transmitter is larger than the one by the receiver because of the signal amplification over the distance. The energy consumption model is presented in Eq. 4.

$$\begin{cases} E_{Tx}(k,d) = E_{elec} \times k + E_{amp} \times k \times d^m \\ E_{Rx}(k) = E_{elec} \times k \end{cases} \tag{4}$$

where $E_{Tx}(k,d)$ and $E_{Rx}(k)$are the energy consumed by the transmitter and receiver respectively.

The transmission (or reception) energy E_{elec} estimated at $50nJ/bit$ is the energy consumed to run the transmitter or receiver circuit, whereas amplification energy estimated at $100pJ/bit/m_2$ is the energy consumed to amplify the signal over the distance. Finally m =2 or 4 depending upon the distance.

4.2.2 *Funtion for Rewarding routing decisions.*

The reward function is the cost representing the goodness or badness of the action taken by the agent at a given state. In EBR-RL, the action refers to the selection of a neighbor as next-hop and its cost to the reward obtained for the selection of that neighbor as next-hop to the sink. This cost is estimated based on the factors mentioned in the previous phase including hop count (N_h) and the residual energy (E_r). The hop count can differ from a sensor node to another depending on the distance between that sensor node and the destination sensor node or sink node. For optimal routing, the reward function will be used to update the Q-Value. Eq. 5 shows the computation of the reward for packet transmission to the next-hop.

$$r_{t+1} = \begin{cases} \frac{1}{N_h} & \text{if } E_{min} = E_{max} \\ p \times (\frac{E_r - E_{min}}{E_{max} - E_{min}}) + (1-p) \times \frac{1}{N_h} & \text{if } E_{min} \neq E_{max} \\ -100 & \text{if } E_r \leq 0 \end{cases} \tag{5}$$

where $0 \leq p \leq 1$ is the probability factor, which defines the impact of the E_r in contrast to N_h.

To ensure energy efficiency and low latency, a good value of p is required, this will decrease the probability of selecting sensor nodes

with low energy level or very far from the sink. Furthermore,if $E_r \leq 0$ in this case, S_i will assign a negative reward to S_j and will select another sensor node for the next-hop. Similarly, S_j also repeats the same process when forwarding the packet to the next-hop and sends a feedback to S_i. The reward is encapsulated in the packet header, so that all the sensor nodes in the neighborhood including S_i can overhear the packet forwarded and update their routing table. S_i will also take it as a feedback of the packet sent.

4.2.3 *Function for updating Q-value .*

To learn the real cost of selecting the next-hop by a sensor node sending a packet, the sensor node updates the Q-value by incorporating the reward received from the selected neighbor sensor node as next-hop.Eq. 6 shows the equation for updating the Q-value.

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha(r_{t+1}(s,a) + \gamma max_a Q(s',a') - Q_t(s,a)) \quad (6)$$

where α is the learning rate, which in most cases is set to 1 to speed up the learning process, and $r_{t+1}(s,a)$ is the immediate reward computed using Eq. 5.

The policy used is such that, the sender select the neighbor with the highest Q-value denoted as $max_a Q(s',a')$, to maximize the reward, and end up in state s'. The discount factor γ whose value is between 0 and 1, defines the importance of the long-term rewards against the immediate one. If γ approaches 1, it means that the agent emphasizes the future reward rather than the immediate reward. Therefore most RL based protocols set a value that approaches 1 to give high importance to the taken in future reward. In the EBR-RL as well we use $\gamma = 0.95$. However, a discount factor of 0 means that the agent is more concerned only with maximizing the immediate reward, in routing this can reduce the network lifetime.

Algorithm 2 describes the data transmission procedure. In algorithm 2, sensor nodes whose residual energy is less than zero are considered dead, thus cannot transmit data. However, sensor nodes with the highest Q-value in their neighborhood and sensor nodes with the sink node in their communication range can communicate directly with the sink node without any intermediate sensor node. On the other hand, sensor nodes far from the sink node select the next-hop from their neighborhood based on the Q-Value, in this case, the next hop is the sensor node with the highest Q-value. The selected next-hop aggregates data from the source sensor node, and pass it either to the sink node directly or through another sensor node close to the sink node. At each packet transmission, a reward is computed and sent back to the source, and each sensor node updates its Q-value.

4.2.4 *Updating Routing Table.*

During the initial step, each sensor node computes an optimal route based on the distance only as described in Step 1 of EBR-RL. In homogeneous WSN, the route computed in step 1 is as same as the shortest path and does not consider the energy level. However, in order to optimize energy consumption, the learning step will consist of updating the routing table every time there is a change in the neighborhood. For this reason, residual energy is also another factor to be considered at this step.

Algorithm 2: Data Transmission

```
for i ← 1 to n do
    if S(i).E>0 then
        maxQ = max(Q(source,:));
        if S(i).Q >= maxQ or S(i).d <= 20 then
            if Any sensornode ∈ NET has S(i) as NextHop then
                Aggregate data
                send data to sink
            else
                send data to sink
            End
            Compute Reward
            Update Q-value
        else
            Nneig = 1
            for j ← 1 to n do
                if S(j) ∈ Neigh(i,:) then
                    if Q(i,Nneig) == maxQ then
                        if Any sensornode ∈ NET has S(i) as NextHop then
                            Aggregate data
                            send data to S(j)
                        else
                            send data to S(j)
                        End
                        Compute Reward
                        Update Q-value
                        End
                    Increment Nneig
                End
            End
        End
    End
End
```

5 PERFORMANCE EVALUATION

In order to evaluate the performance of the proposed EBR-RL, Simulations were conducted using MATLAB, where 100 sensor nodes were randomly distributed over a sensing field of 100×100m following the normal distribution [17]. The sink node was placed in the middle of the field with (50,50) coordinates. Furthermore, we assumed that all sensor nodes have the same initial energy and communication range as in a homogeneous WSN. The simulation parameters are presented in Table 1, and Figure 1 shows the distribution of sensor nodes within the sensing field.

Table 1: Parameters of Simulation

Parameters	Values
Sensing field size	$100 \times 100m$
Number of sensor nodes	100
Sink position(m)	(50,50)
Communication range	20 m
Initial Energy	2 Joules
Data size	4000 bits
E_{elec}	$50 \times 10^{-9} Joules/bit$
E_{amp}	$100 \times 10^{-12} Joules/bit/m^2$
α	1
γ	0.95

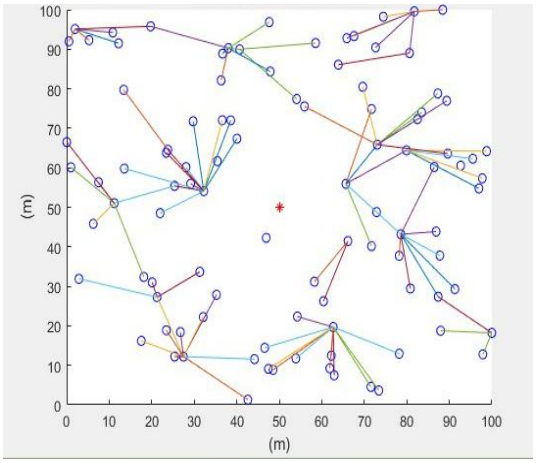

Figure 1: Initial Routing

5.1 Impacts of probabilistic factors on energy consumption

As noted above, the proposed protocol takes the hop count and residual energy into consideration in section 4, some probabilistic parameters such as p and $q = 1 - p$ were assigned to both residual energy and hop count respectively. A high value of p gives the sensor nodes with high energy levels a high probability to be selected, similarly, a high value of q increases the probability of sensor nodes with less hop count to sink node to be selected. Therefore, to optimize the performance of EBR-RL, different values of these parameters were tested to show their impact on the assessment of the proposed protocol. The performance evaluation showed slightly different results with different values of p and q. However, with p and q equal to 0.3 and 0.7 respectively, the network lifetime was extended more, while keeping a good energy balance. As a result, this prolonged the time until the first sensor node dies. Figure 2 shows the performance evaluation results of EBR-RL using different values of p and q.

5.2 Energy efficiency evaluation

We have also compared the performance of the proposed EBR-RL with LEACH [5] and PEGASIS [7] clustering protocols. We used the

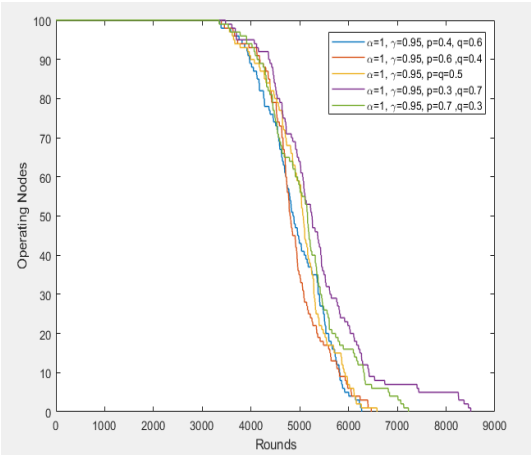

Figure 2: Performance evaluation of EBR-RL using different value of p and q

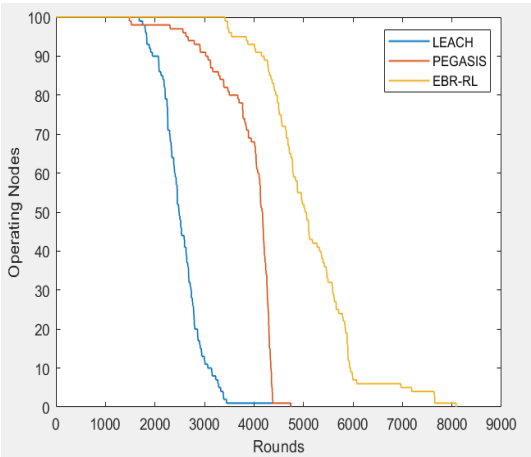

Figure 3: Number of alive sensor nodes per round

following metrics for comparison:(1) Number of alive sensor node per round, this metric also helps to evaluate the network lifetime.(2) Energy consumed per round, it is the sum of energy consumed by all the sensor nodes at each round.(3) Average energy consumed per round, the average energy consumed by a node at each round, this quotient of total energy consumed, and the number of alive sensor nodes.

The performance results showed that the proposed EBR-RL outperformed LEACH and PEGASIS in terms of energy consumption and network lifetime. In addition, providing a good energy balance between sensor nodes was the key idea in the work, as this help to extend the network lifetime. Figure 3 shows a good energy balance which results in extending the time until the first node dies. In Figure 4, EBR-RL shows high energy consumption in the beginning, due to the learning process. Therefore, we set up the learning rate α to 1 to speed up the learning process. Consequently, after the learning process, the energy consumed per round was considerably reduced. On the other hand, we set up the discount factor γ to

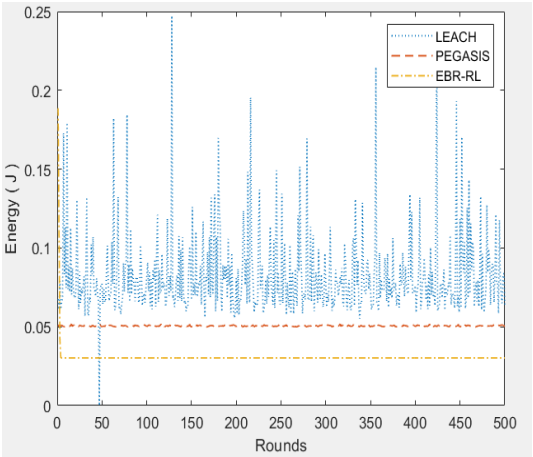

Figure 4: Energy consumed per round

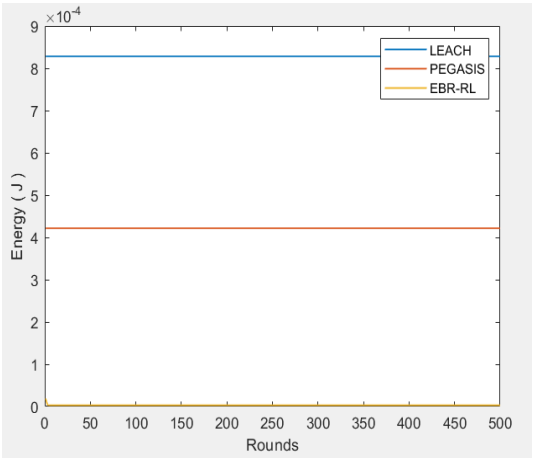

Figure 5: Average energy consumed by sensor node per round

0.95 to focus more on the future reward. This played an important role in balancing energy consumption in the long run as it allows minimizing the energy consumed by each sensor node and extends the network lifetime as shown in Figure 5.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we proposed EBR-RL, a reinforcement learning-based energy balancing routing protocol for WSN. The main focus of the EBR-RL protocol was to balance the energy consumption between sensor nodes and maximize the network lifetime.

EBR-RL was designed in two phases such as network set-up in which we considered the hop count factor to compute the initial route, the shortest path to the destination was the result of this phase. On the other hand, the data transmission phase which is characterized by the learning using reinforcement learning provided an energy-efficient routing where both residual energy of sensor nodes and hop count where considered. The simulation results showed that EBR-RL achieved a better performance in terms of energy consumption and network lifetime compared to LEACH and PEGASIS and provided a good energy balance.

In the future, we plan to extend EBR-RL to a cluster-based routing protocol and add more considerations such as heterogeneity of sensor nodes, scalability. We would like also to assess the proposed EBR-RL using the packet delivery ratio and end-to-end delay.

ACKNOWLEDGMENTS

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT (No. NRF-2017M3C4A7069432)

REFERENCES

[1] K. A. Yau, H. G. Goh, D. Chieng, and K. H. Kwong, "Application of reinforcement learning to wireless sensor networks: models and algorithms," *Computing*, vol. 97, no. 11, pp. 1045–1075, 2015.
[2] S.-H. Yang, *Hardware Design for WSNs*, ch. 3, pp. 49–72. Springer London, 2014.
[3] "Internet of things: Wireless sensor networks," white paper, IEC, 2014.
[4] M. S. Obaidat and S. Misra, *Inside a wireless sensor node: structure and operations*, ch. 2, p. 14–29. Cambridge University Press, 2014.
[5] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pp. 1–10, 2000.
[6] J. Hong, J. Kook, D. K. S. Lee, and S. Yi, "T-leach: The method of threshold-based cluster head replacement for wireless sensor networks," *Information Systems Frontiers*, vol. 11, no. 5, pp. 513–521, 2008.
[7] S. Lindsey and C. Raghavendra, "Pegasis: Power-efficient gathering in sensor information systems," in *Proceedings of the IEEE Aerospace Conference*, pp. 1125–1130, 2002.
[8] S. Yi, J. Heo, Y. Cho, and J. Hong, "PEACH: power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks," *Comput. Commun.*, vol. 30, no. 14-15, pp. 2842–2852, 2007.
[9] D. Oužecki and D. Jevtić, "Reinforcement learning as adaptive network routing of mobile agents," in *Proceedings of the 33rd International Convention MIPRO*, pp. 479–484, 2010.
[10] L. Xu, R. Collier, and G. M. P. O'Hare, "A survey of clustering techniques in wsns and consideration of the challenges of applying such to 5g iot scenarios," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1229–1249, 2017.
[11] Y. Touati, A. Ali-Chérif, and B. Daachi, *Routing Information for Energy Management in WSNs*, ch. 3, pp. 23 – 51. ISTE Press - Elsevier, 2017.
[12] R. S. Sutton and A. G. Barto, *Reinforcement Learning : An Introduction*. Cambridge: The MIT Press, 2nd. ed., 2020.
[13] M. Littman and J. Boyan, "A distributed reinforcement learning scheme for network routing," in *Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications*, pp. 1–6, 1993.
[14] G. Oddi, A. Pietrabissa, and F. Liberati, "Energy balancing in multi-hop wireless sensor networks: an approach based on reinforcement learning," in *Proceedings of the NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, pp. 262–269, 2014.
[15] S. Z. Jafarzadeh and M. H. Y. Moghaddam, "Design of energy-aware qos routing algorithm in wireless sensor networks using reinforcement learning," in *Proceedings of the 4th International Conference on Computer and Knowledge Engineering (ICCKE)*, pp. 722–727, 2014.
[16] A. Forster and A. L. Murphy, "Froms: Feedback routing for optimizing multiple sinks in wsn with reinforcement learning," in *Proceedings of the 3rd. International Conference on Intelligent Sensors, Sensor Networks and Information*, pp. 371–376, 2007.
[17] C. F. Gauss, "Normal distribution," 2001. [Online; accessed 13-April-2020].