



Article Dynamic Topology Model of Q-Learning LEACH Using Disposable Sensors in Autonomous Things Environment

Jae Hyuk Cho * D and Hayoun Lee

School of Electronic Engineering, Soongsil University, Seoul 06978, Korea; leehayoun0916@ssu.ac.kr * Correspondence: chojh@ssu.ac.kr; Tel.: +82-2-811-0969

Received: 10 November 2020; Accepted: 15 December 2020; Published: 17 December 2020



Abstract: Low-Energy Adaptive Clustering Hierarchy (LEACH) is a typical routing protocol that effectively reduces transmission energy consumption by forming a hierarchical structure between nodes. LEACH on Wireless Sensor Network (WSN) has been widely studied in the recent decade as one key technique for the Internet of Things (IoT). The main aims of the autonomous things, and one of advanced of IoT, is that it creates a flexible environment that enables movement and communication between objects anytime, anywhere, by saving computing power and utilizing efficient wireless communication capability. However, the existing LEACH method is only based on the model with a static topology, but a case for a disposable sensor is included in an autonomous thing's environment. With the increase of interest in disposable sensors which constantly change their locations during the operation, dynamic topology changes should be considered in LEACH. This study suggests the probing model for randomly moving nodes, implementing a change in the position of a node depending on the environment, such as strong winds. In addition, as a method to quickly adapt to the change in node location and construct a new topology, we propose Q-learning LEACH based on Q-table reinforcement learning and Fuzzy-LEACH based on Fuzzifier method. Then, we compared the results of the dynamic and static topology model with existing LEACH on the aspects of energy loss, number of alive nodes, and throughput. By comparison, all types of LEACH showed sensitivity results on the dynamic location of each node, while Q-LEACH shows best performance of all.

Keywords: wireless sensor networks; disposable IoT; reinforcement learning; Q-learning; artificial intelligence; LEACH; fuzzy clustering

1. Introduction

The most potent and influential technology of 21st century is Wireless Sensor Network (WSN). This is a key technology in the ubiquitous network, and is becoming more important as sensors can connect everything to the Internet and process important information. It also collects information about the surrounding environment (temperature, humidity, etc.) with a specific purpose and transmits it to the base station through various processes. With the advancements of WSN disposable sensors shows enhanced performance on the fields where the user cannot access or include large area, such as forest or deep sea [1]. In addition, the size of the sensor node should be small as it includes batteries, making it impossible to charge or replace, and often forming a wireless network by deploying a large number of sensors. And because it uses a battery, it cannot be charged or replaced. In addition, the size of the sensor node must be small as a large number of sensors are placed to form a wireless network. In addition, the price should be low, so there is a limit to the ability to process data and the amount of power that can be supplied to the nodes [2].

In general, in a wireless network environment, disposable sensor nodes that make up the network must perform routing and sensing roles together, so each sensor node always has an energy burden [3]. Due to these characteristics of disposable sensor network, the efficient use of limited power is of paramount importance in WSN design. Thus, various studies are currently being conducted to maximize energy consumption efficiency and to increase the overall life span.

Since routing protocols in WSN are very important to efficiently transmit detected data to the Base Station (BS), clustering-based routing is preferred to obtain the advantage of efficient communication. As a common low-energy cluster-based protocol, Low-Energy Adaptive Clustering Hierarchy (LEACH) was proposed [4]. This is a technique that utilizes random rotation of BS to distribute network sensors and energy loads evenly. Only Cluster Heads (CH) interact with BS, which saves energy and increases network lifetime [5]. Thus, the idea of clustering-based routing is aimed at saving energy on sensor nodes by using CH's information aggregation mechanism to reduce the amount of data transmission and energy. Thus, in the case of LEACH, the advantage of ensuring even energy use is achieved by using the probability function, but all nodes should take part in the CH election process every round, and, if a few CHs are elected, the number of nodes to be managed increases, resulting in high network traffic and increased energy consumption. Therefore, various LEACHs were developed to compensate for these problems in the preceding studies.

In recent years, extensive research has been conducted on WSN's various systems of artificial intelligence approaches, called Reinforcement Learning (RL), to improve network performance, and there is much interest in applying them [6]. A Q-learning-based cooperative sensing plan is proposed to enhance the performance of spectrum in a wireless network environment [7]. In Autonomous Things environment, the interests on disposable sensors with characteristics that can be easily sprayed and moved is emerging. Therefore, efficient energy management is required considering these topology changes, in real life.

As limits exist for applying multiple LEACH algorithm with fixed topology state in previous studies, the development of a new dynamic topology model is required. In addition, as opposed to traditional LEACH and D-LEACH, which performs LEACH in its jurisdiction by dividing the zones, D-LEACH is efficient in the static topology simulation; however, in dynamic topology simulation, with higher uncertainty, the advantage of D-LEACH is not expected. The F-LEACH we developed for testing to secure dynamic uncertainty consists of two processes: first, it performs flexible clustering, centering nodes toward the cluster center by upper/lower membership values which compensate uncertainty and sensitivity, while producing optimized fuzzifier constant by histogram. Then, LEACH method is performed on the clustered node location. Q-LEACH the LEACH method using Q-learning (reinforced learning) rewards the success of the agent progress from multi nodes to the cluster head, and it derives this success in the form of a performance probability. From the results of energy loss, number of alive nodes, and throughput between LEACH, D-LEACH, F-LEACH, and Q-LEACH, finally, we invented the topology model, which updates the changes of sensor location in a dynamic or static autonomous things environment and definitize Q-LEACH, which increases the efficiency of sensor energy consumption, extending sensor life.

2. Materials and Methods

2.1. Wireless Sensor Network (WSN)

In simple terms of a wireless sensor network, it means that sensors operating wirelessly form a network to transmit data. As shown in Figure 1, when an event occurs in a sensor node in each cluster, event status information and location information are transmitted to the head of the cluster. And then the head of the cluster is configured to immediately send all information to BS or Sink.



Figure 1. Diagram of Wireless Sensor Network (WSN).

The latest WSN technology enables two-way communication to control sensor activity. The types of WSN can be roughly divided into homogenous [6,7] and heterogeneous network [8,9]. Previous studies using wireless sensor networks have limited energy because sensor nodes operate with batteries. Therefore, for efficient use of the battery, a clustering technique was developed that considers network overlapping prevention [8]. People who use the technique could cite several tasks that deal with monitoring [9–11]. This is not only for medical use, but WSN is the same in civil areas, such as health monitoring and smart agriculture, as well as enemy tracking and military areas. Remote sensing execution is greatly simplified to be useful in inexpensive sensors and essential software packages. Therefore, the user can monitor and control the basic environment from a remote location, and it will be used in various ways besides safety accident and prevention training [12,13].

2.2. Disposable IoT Sensors

Currently, various technologies have been fused and evolved into IoT technologies. Existing IoT sensors have limitations in price, size, and weight, so it is difficult to precisely monitor a wide range of physical spaces. To solve these problems, technology of disposable IoT that has reduced size, price, and weight and has been constantly studied and developed. In the late 1990s, The University of California developed a very small sensor with a size of 1–2 mm called "Smart Dust" [14]. It has a technology that can detect and manage surrounding physical information (temperature, humidity, acceleration, etc.) through a wireless network through disposable IoT sensors (micro sensors). Therefore, it is possible to accurately sense the entire space by using a low-cost micro sensor compared to the existing sensor. In smart dust system using disposable sensors, LEACH is one of the key technologies for effective data collecting. In addition, disposable sensors broadcast collected data without polling process in order to reduce energy consumption and the size of send/receive payloads are relatively small. Disposable IoT sensor can be applied to a wide range of fields, such as weather, national defense, safety of risks, and detection of forest fire, as it can judge parts that are difficult for user to access [15].

With the advancements of wireless sensor networks, it is anticipated that real time forest fire detection systems can be developed for high precision and accuracy using wireless sensors data. Thousands of disposable sensors can be densely scattered over a disaster-prone area to form a wireless sensor network in a forest [16].

For monitoring the environment, disposable sensors are distributed in large areas by aerial dispersion using airplanes or dispersion in water stream. Distributed disposable sensors can be dislocated from original point by environmental condition, such as winds or movement of animals, since its size is only a maximum of 1 cm², as shown Figure 2.



Figure 2. Image of disposable sensor.

2.3. LEACH (Low-Energy Adaptive Clustering Hierarchy)

The structure of a network can be classified according to node uniformity, in Flat Networks Routing Protocols (FNRP) and Hierarchical Networks Routing Protocol (HNRP) [17].

LEACH is one of HNRP in sensor networks proposed by Wendi Heizehan [18]. This is a typical routing protocol that effectively reduces transmission energy consumption by forming a hierarchical structure between nodes. As shown in Figure 3, LEACH is a method in which the cluster head collects and processes data from the member nodes of the cluster and delivers it directly to the BS. If this is used, the cluster head is selected by the ratio of all sensor nodes, and the cluster head is determined by calculation inside the sensor node, so it is a distributed system.



Figure 3. Diagram of WSN. (**a**) Flat networks routing protocols. (**b**) Hierarchical routing protocols (Low-Energy Adaptive Clustering Hierarchy (LEACH)).

The operation of LEACH is divided into rounds, and each round begins with a setup phase when the cluster is formed, and consists of a steady state phase when data is transmitted to the base station. In order to minimize overhead, the steady state phase is longer than the set phase, as shown in Equation (1) [19]. In Equation (1), the setup is: each node decides to depend on CH of the current round. This decision is based on choosing a random number between 0 and 1 if the number is less than the threshold T(n), and the node becomes the cluster head of the current round. The threshold is set as follows. If the cluster head is selected, the state is reported using the Carrier Sense Multiple Access (CSMA) Medium Access Control (MAC) protocol. The remained nodes make decisions about the current round's cluster head according to the received signal strength of the advertisement message. The Time Division Multiple Access (TDMA) scheduling is applied to all members of the cluster group to send a message to the CH and then from the cluster head to the base station. As soon as a cluster head is selected for a region, the steady state phase begins. Steady: the cluster is created when the TDMA scheduling is fixed and data transfer can begin. If the node always has the data to send, it sends

it to the cluster head for the allotted transmission time. Each head node can be turned off until the node allocates the transmission time, thus minimizing energy loss in these nodes [20].

$$T(n) = \begin{cases} \frac{\rho}{1 - \rho(r \mod(\frac{1}{\rho}))}, & \text{if } n \in G\\ 0, & \text{otherwise} \end{cases}$$
(1)

T(n) = Threshold; r = current round; G = The set of nodes that have not been CH; $\rho =$ the percentage of cluster – heads.

2.3.1. Enhancements of LEACH

There are several drawbacks with LEACH. The setup phase is non-deterministic due to randomness. It may be unstable during setup phase that depending on the density of sensor nodes. It is not applicable on large networks as it uses single hop for communication. CH, which is located far from BS, will consume huge amount of energy. It does not guarantee the good CH distribution and it involves assumption of uniform energy consumption of CH during setup phase. And also, the main problems with LEACH lie in the random selection of CH. Various studies have been conducted to improve the LEACH protocol to improve network lifetime, energy savings, and performance and stability [21–28]. LEACH-Centralized (LEACH-C) method was developed for improving the lifetime of the network in the setting stage and also each node transmits information related to the location and energy level to the BS. The BS determines the cluster, as well as the CH and each cluster node. Through this, it was possible to extend the life of the network and improve energy savings [21]. In addition, in S-LEACH, solar power is used to improve WSN life. Typically, the remained energy is maximized to select a node with redundant power, and the node transmits the solar state to the BS along with its energy level. As a mechanism, nodes with higher energy are selected as CH, and the higher the number of recognition nodes, the better the performance of the sensor network. According to a Simulation results, S-LEACH significantly extends the lifetime of WSNs [23]. In addition, in the case of a LEACH, a new node concept that reduces and provides a uniform distribution of dissipated energy by dividing routing work and data aggregation, and supports CH in multi-hop routing is introduced. Energy-efficient multi-hop pathways are established for nodes to reach the BS to extend the life of the entire network and minimize energy dissipation [28]. As a method for saving energy, H-LEACH was developed with various algorithms that consider the concept of minimizing the distance between data [21]. It adds a CH of LEACH to act as the Master Cluster Head (MCH) to pass the data to the BS. In addition, it is proposed I-LEACH to save energy, while communicating within the network. In order to increase the stability of the network, we proposed Optimized-LEACH(O-LEACH), which optimizes CH selection [29]. This means that the selected CH is based only on the remaining dynamic energy. If the energy of the node is greater than 10% of the minimum residual energy, the node is selected as CH, otherwise the existing CH is maintained. In order to deal with situations in which the diameter of the network increases beyond certain limits, D-LEACH randomly places nodes with a high probability of being located close to each other, called twin nodes. It is necessary to keep one node asleep until the energy of another node depletes. Therefore, D-LEACH has uniform distribution of CH so that it does not run out of energy when longer distance transmission takes place [21]. The D-LEACH method ensures that the selected cluster heads are distributed to the network. Therefore, it is unlikely that all cluster heads will be concentrated in one part of the network [25].

2.3.2. Clustering in LEACH

Clustering means defining a data group by considering the characteristics of given data and finding a representative point that can be represented. In LEACH, the sensor detects the events and then sends the collected data to a faraway BS. Because the cost of information transmission is higher

than the calculation, nodes are clustered into groups to save energy. It allows data to be communicated only to the CH, then the CH routes the aggregated data to the BS. The CH, which is periodically elected using a clustering algorithm, aggregates the data collected from cluster members and sends it back to the BS, which can be used by the end user. To transmit data over long distances in this way, only a few nodes are required, and most nodes need to complete short distance transmissions. So, more energy is saved, and the WSN life is extended. The main idea of a hierarchical routing protocol is to divide the entire network into two or more hierarchical levels, each of which performs different tasks [20]. In order to create these hierarchical levels, the clustering functions as a critical role in LEACH.

2.4. Interval Type-2 Possibilistic Fuzzy C-Means (IT2-PFCM)

It is known that the synthesis of Fuzzy C-Means (FCM) and T2FS gives more room to handle the uncertainties of belongingness caused by noisy environment. In addition, a Possibilistic C-Means (PCM) clustering algorithm was presented, which allocates typicality using an absolute distance between one pattern and one central value. However, the PCM algorithm also has a problem that clustering results respond to the initial parameter values sensitively. To address this sensitivity problem, PFCM algorithm combining FCM and PCM by weighted sum was proposed. However, the PFCM method also has the uncertainty problem of determining the value of the purge constant. It is an important issue to control the uncertainty of the fuzzy constant value because the fuzzy constant value plays a decisive role in obtaining the membership function. To control the uncertainty of these fuzzy constants, hybrid algorithms are suggested, which includes the general type-2 FCM [30–32], Interval Type-2 FCM (IT2-FCM) [33], kernelized IT2-FCM [34], interval type-2 fuzzy c-regression clustering [35], interval type-2 possibilistic c-means clustering [36,37], interval type-2 relative entropy FCM [38], particle swarm optimization based IT2-FCM [39], interval-valued fuzzy set-based collaborative fuzzy clustering [40], and others. These T2FS based algorithms have been successfully applied to areas like image processing, time series prediction and others. Interval Type-2 FCM (IT2-FCM): In fuzzy clustering algorithms, like FCM, the fuzzification coefficient *m* plays an important role in determining the uncertainty of partition matrix. However, the value of *m* is usually hard to be decided upon in advance. IT2-FCM considers the fuzzification coefficient as an interval (m_1, m_2) and solves two optimization problems [41].

Fuzzifier Value

IT2 PFCM is expressed as the sum of the weights of FCM and PCM method. Therefore, it is clustered in the direction of minimizing PFCM objective function, as follows in Equation (2).

In Equation (3), u_{ik} represents a membership value where the input pattern k belongs to cluster

$$J_{m,\eta}(U,T,V;X) = \sum_{k=1}^{n} \sum_{i=1}^{c} (au_{ik}^{\ m} + bt_{ik}^{\ \eta}) \times ||x_k - v_i||^2 + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{n} ((1 - t_{ik})^{\eta}),$$
(2)

$$\sum_{i=1}^{c} u_{ik} = 1, 0 \le u_{ik}, t_{ik} \le 1, m = >1, \eta > 1, \gamma_i > 0.$$
(3)

i. x_i is the *k*-th input pattern, and v_i is the center value of the *i*-th cluster. m is a constant representing the degree of fuzziness and satisfies the condition of $m \in (1, \infty)$. tik represents typicality that the input pattern *k* belongs to cluster *i*, which is a feature of PFCM method using absolute distance. γ_i is a scale defining point, where typicality of the *i*-th cluster is 0.5, and the value is an arbitrary number.

For PFCM cluster method, in above Equation (5) the objective function should be minimized with respect to the membership function u_{ik} . Membership that is obtained by Equation (2). To draw m,

you must create the lowest and highest membership functions using the primary membership function. The highest and lowest membership functions of PFCM according to m are as follows.

$$\overline{u}_{ik} = \left(\frac{1}{\sum_{j=1}^{1} \left(\frac{d_{ik}}{d_{ij}}\right)^{\frac{2}{m_{1}-1}}}\right),\tag{4}$$

$$\overline{u}_{ik} = \left(\frac{1}{\sum_{j=1}^{1} \left(\frac{d_{ik}}{d_{ij}}\right)^{\frac{2}{m_2 - 1}}}\right),\tag{5}$$

As shown in Equation (6), the lowest and highest membership values where m_1 and m_2 are representing objective function, the value γ_i also changes according to the lowest and highest membership functions. Using γ_i , the lowest and highest typicality is,

$$\gamma = \frac{\log\left(\frac{1}{u_j}\right) - \log\left(1 + \sum_{k=2}^{C} \left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m_{old}} - 1}\right)}{\log\left(\frac{d_{ij}}{d_{ik}}\right)}$$
(6)

$$m_{jnew} = 1 + \frac{2}{\gamma} \tag{7}$$

For updating the center value, as shown in Equation (7), the type reduction process of changing type-2 fuzzy takes up type-1 using the K-means algorithm that it is performed, and the center value of each cluster is updated.

$$m_1 = \left(\sum_{i=1}^N m_{1i}\right) / N, \ m_2 = \left(\sum_{i=1}^N m_{2i}\right) / N$$
 (8)

Table 1 shows the symbol and original values of IT2-PFCM.

Symbol	Explanation
С	Bulky cluster
v	Cluster center
v_i, V	Cluster prototype
т	Fuzzifier value
и	Membership function
U	Partition matric
d _{ik} /d _{ij}	Euclidean distance value
δ	Threshold of fuzzifier constant
\widetilde{A}	Secondary membership degree
J	PFCM objective function
x	Input pattern
t _{ik}	Represents typicality, the input pattern k belongs to cluster i
γi	Scale defining point where typicality of the i-th cluster is 0.5
X_i	Input space
$\Phi(X_j)$	Kernel property space
K	Input space for kernel
S	Number of kernels

Figure 4 shows that histograms and Footprint of Uncertainty (FOU) examples are determined by class and dimension. Upper membership function (UMF) histogram and lower membership function (LMF) histogram are obtained according to class and dimension. A new membership function from the Gaussian Curve Fitting (GF-F) method can be applied to calculate the adaptive fuzzifier value.



Figure 4. Histogram for fuzzifier value. (a) LMF histogram (b) UMF histogram

2.5. Reinforcement Learning (RL)

RL is one of the unsupervised learning methods which tries to find out some policies from interaction with the environment. It is the problem faced by an agent that has to learn behavior through trial and error interactions with vigorous environments. And it was applied successfully in many agent systems [42–44]. The aim of reinforcement learning is to find out a useful behavior by evaluating the reward. Q learning is a RL method adequately appropriate for learning from interaction, where the learning is performed through a reward mechanism. This method was applied to WSN optimization problems [45]. Research into applying intelligent and machine learning methods for power management was considered in Reference [46], with Reference [47] being among those specially targeting the area of dynamic power management for energy harvesting embedded systems. A learner interacted with the environment and autonomously determined required actions [47]. An RL-based Sleep Scheduling for Coverage (RLSSC) algorithm is for sustainable time-slotted operation in rechargeable sensor networks [48]. Then, the learner was rewarded by the reward function to respond to different environmental states. RL can be applied to both single and multi-agent systems [49]. In previous studies, the efficiency of RL was improved by developing RL applications in WSN [50]. An independent RL approach for resource management in WSN is proposed [51]. WSN tasks through random selection can provide better performance in terms of cumulative compensation over time and residual energy of the network [52].

Q-Learning

The most well-known reinforcement learning technique is Q-learning. Q-learning is a representative algorithm of RL prosed by Watkins [53]. As shown in Figure 5, data is collected directly by the agent acting in a given environment. In other words, the agent takes an action (a) in the current state (s) and gets a reward (r) for it.

In Equation (9), after Q-learning $Q(s_t, a_t)$ is initialized to an arbitrary value, it is repeatedly updated with the following formula value according to learning progresses. R_{t+1} is the reward that gets it from current state(s_t) to next state (s_{t+1}). γ is a discount factor, which is the largest value among the action value function values that can be obtained in the next state (s_{t+1}) [54].

$$Q(s_t, a_t) = R_{t+1} + \gamma \max_{a+1} Q.$$
(9)



Figure 5. Diagram of Q-learning.

2.6. Proposed Modification in LEACH

Figure 6 shows the workflow of this study, first, we define the deployment of sensor node in the matrix then, choose the model according to sensor movement, if the movement of sensor topology is not considered as traditional LEACH modeling, static topology model is chosen; otherwise, the dynamic topology model is applied to imply the disposable sensor movement. In the chosen model, LEACH protocol algorism is applied, and each LEACH protocol selects Cluster Head (CH) in its own way, while the calculation is repeated until the system is terminated as the node dying.



Figure 6. Workflow of research.

2.6.1. Dynamic Topology Modeling

The simulation of the dynamic topology model is signed with the changing node location on the field during operation. This reflects the phenomenon that the micro-sensors exposed to the real environment show movements. As round (r) in Equation (10) increase, the position of the node (X_r , Y_r),

$$X_r = X_{r+1} + R_{random}Y_r = Y_{r+1} + R_{random} - m_{random} \le R_{random} \le m_{random}$$
(10)

every node updates its location as (X_{r+1}, Y_{r+1}) , while R_{random} is a random number between $-m_{random}$ and $+m_{random}$. In the proposed method, one step is inserted before the network moves to the steady state step. After cluster formation is complete, the cluster topology is improved using the various methods (Q-LEACH: reinforcement learning, F-LEACH: IT2 PFCM). The aim of the improvement is to still maintain network connectivity with minimal energy loss in the network. All sensors set the transmit power to the minimum level but keep the received signal power of CH above the E_b/I_o (Threshold) value so that the sensor transmits with an acceptable error rate.

In order to improve the topology of the cluster, the simulation is set up as follows in Equation (11). E_t is the total energy consumed in the network, E_{tx_i} is the energy of the sensor *i* transmitting

$$E_t = min(\sum_{i=1}^{N} E_{tx,i} + \sum_{k=1}^{n} E_{CH,k})$$
(11)

data to the CH, and *k* is the energy of the CH receiving data from the sensor.

In Equation (12), Signal Interference Noise Rate (SINR) is the received signal-to-noise-noise ratio, where *P* is the transmit power of the sensor to CH to transmit synchro data, *I* is the interference power, and N is the noise density. As the distance between nodes gets closer, I and P values increase, and, as the distance between nodes increases, *I* and *P* values decrease. The energy the sensor consumes to send data to the cluster head depends on the distance between the sensor and the CH. If the distance is less than d_0 , the energy depends on the free space. However, if the distance is greater than d_0 , the energy is dependent on multipath fading. In addition to the transmission energy, the sensor consumes energy to process data inside the hardware (E_{elec}) in Equation (13).

$$SINR = \frac{P}{I+N'}$$
(12)

$$E_{tx} = \begin{cases} LE_{elec} + LE_{EFS}d^2, & d \le d_0\\ LE_{elec} + LE_{EMP}d^4, & d > d_0 \end{cases}$$
(13)

$$E_{CH} = L(E_{RX} + E_{elec} + E_{DA}) + LE_{MP}d^2.$$
(14)

The CH receives L-bit data from the sensor with the energy E_{RX} , processes the data inside the hardware with the energy E_{elec} , aggregates the data with the energy E_{DA} , and transmits the data to the sink with the energy E_{MP} in Equation (14).

2.6.2. F-LEACH

The disadvantage of the existing D-LEACH is that the distribution method is too simple as 1/N. Therefore, the proposed F-LEACH in this paper provides more effective distributed clustering compared to the D-LEACH protocol. To do this, dispersion proceeds through the introduction of Interval Type-2 FCM (IT2-FCM). The entire sensor field is divided into sub-areas by fuzzifier centroid. Fuzzifiers (m_{1i} , m_{2i}) for each data point calculation. (function of upper and lower membership) As the effect of the fuzzifier value on the cluster center position, it is clustered by automatic calculation of the fuzzifier value at the cluster center position, as shown in the following the algorithm (Algorithm 1) of the F-LEACH protocol.

Algorithm 1. F-LEACH

1: node (x, y): initial position input to csv (or generate random position) 2: N; number of alive nodes 3: numClust: number of CH 4: CHpl: packet size of CH 5: NonCHpl: packet size of node (not CH) 6: **for** round r = 1: R 7: Set the initial any value of fuzzifier (m_1, m_2) 8: (m_1, m_2) in interval type-2 PFCM and calculate membership of each data point. 9: select histogram for upper and lower membership. 10: Generate a histogram for individual clusters. 11: Use curve fitting over these histograms to obtain upper and lower memberships. 12: Normalization (m_{upper}, m_{lower}) = (1,0) 13: Take mean of (m_{1i}, m_{2i}) and update (m_1, m_2) 14: if there is negligible change in the resulting fuzzifier values or terminating criteria is met, Stop simulation end 15: Save node Location, CH Location on Coordinate basis << Calculate Energy of nodes>> 16: D = distance from CH(x,y)-node(x,y) 17: $E(CH) = E(CH) - (((Etrans+Eagg) * CHpl) + (Efs * CHpl * (D.^2)) + (NonCHpl * Erec * d(N/numClust)));$ 18: E(~CH&~Dead) = 19: E(~CH&~Dead)-((NonCHpl * Etrans) + (Efs * CHpl * (D^2)) + ((Erec+Eagg) * CHpl)); 20: E(Dead) = 0; 21: if R: N < 5% Stop simulation end 22: end

2.6.3. Q-LEACH

Q-LEACH is based on Q-table. As a reward, SINR is considered, while states are selected using ϵ -greedy.

Reward

The proposed algorithm sets SINR as state and chooses to use transmit power as action. In order to evaluate efficient SINR and transmission power, when the SINR is less than the threshold(*T*) and it is greater than the threshold(*T*), it is divided into two independent elements. After that, calculate the compensation function r_{t+1} . ω is the weighting factor, a_{max} is the maximum transmission power factor of node *i*, and at is the power for node *i* at the present time. The reward value can have a positive or negative value, depending on *T* in Equation (15).

$$r_{t+1,i} = \begin{cases} \omega_1 \text{SINR} + \omega_2(a_{max} - a_t), & \text{SINR} \le T\\ \omega_1(T - \text{SINR}) + \omega_2(a_{max} - a_t), & \text{SINR} > T \end{cases}$$
(15)

Action Selection

Performed operations in all states are selected using the ϵ -greedy method in Equation (16).

$$a = \begin{cases} \text{random action}(a), & \text{with probability } \varepsilon \\ maxQ(s,a), & \text{with probability } 1 - \varepsilon \end{cases}$$
(16)

 ε is the exploration factor [0,1]. With probability ε , the sensor starts an exploration in which the task is chosen at random and performs a random action(*a*) to figure out how the environment reacts to

One of the reinforcement models, Temporal Difference (TD), is used to estimate the Q-value in Equation (9). The Q-value is updated as follows in Equation (17).

$$Q_{t+1} = Q_t(s_t, a_t) + \beta(r_{t+1} + \gamma \max(Q_t(s_{t+1}, a)) - Q_t(s_t, a_t)).$$
(17)

 β is a learning factor that determines how fast learning occurs (usually set to value between 0 and 1) and γ is a discount factor. The *Q*-value function is calculated using the existing estimate, not the final return. The Q value of the *t*+1 state is the value of the current state $Q(s_t, a_t)$ multiplied by the discount factor that controls the importance of the future value by selecting the max value among the action values in the *t*+1 state. (s_t, a_t) is subtracted. It can be obtained by adding the value obtained by multiplying the value by the learning factor to the value obtained by adding the importance for future value, and the closer to 0. This Q-LEACH (Algorithm 2) is as follows.

Algorithm 2. Q-LEACH

- 2: Source (*x*, *y*) -> target (*x*, *y*) initial position input to csv (or generate random position) # q table generation (Update one space of Q each time)
- 3: Make a map to unit dimension
- 4: Initialize Q table

- 6: discount rate = 0.99, maximum iteration number before regrouping (episodes i = 2000)
- 7: **Repeat (for** each episode *i*):
- 8: state reset (reset environment to a new, random state)
- 9: Initialize reward value
- 10: Initialize While flag done = false
- 11: while not done:
- 12: $\varepsilon = 1./((i/100) + 1)$
- 13: Create a probability value for ε -Greedy (Set to Decay as step passes)
- 14: if Random action when $\varepsilon < 1$ (small probability)
- 15: else Action to the side where max state aggregate is present end
- 16: return new (state, reward, done) from action
 - <<Q table update>>
 - # Update Q-Table with new knowledge using learning rate
- 17: Q table = Max value of reward + (discount * new state)
- 18: Upon arrival from Node to CH, Reward +1
- 19: Save reward value, update state
- 20: reward value appends in array
- 21: Move node to higher reward in Q table
- 22: if node = CH end
- 23: if duplicate nodes exist or move to another location end
- 24: Explore other dimensions
- 25: While complete the solution in total field (1000×1000)

Declaring target matrix from the original topology is preceded as shown in Figure 7. Before applying Q-table, total field is divided into several unit dimension (which is the same method as D-LEACH) and Q-table is calculated in each unit dimension in the space of target matrix, in this study, defined as 15×15 . Q-probability, drawn out in four directions at node position on target matrix, gives the direction of node (toward direction with the highest probability). In the developed Q-table, state demonstrates the Q-probability that of approach (4 direction: up, down, left, right) of node toward CH, as the hole and other nodes in the dimension can act as the causes of lowing the probability. 1 step action occurs

^{1:} Save node Location, CH Location on Coordinate basis

^{5:} Set the γ value multiplied by max Q (state max, action max)

according to the state-direction with the biggest probability. Eventually, the system earns 1 reward when node arrives to CH.



Figure 7. Q-LEACH: Target matrix from original topology and Q-table probability.

The detailed parameters were chosen according to previous studies that analyzed WSN parameters in order to select the optimal value for the experiment [52–55]. Proposing LEACH protocol simulation parameters are shown in Table 2.

Parameter	Value	Unit
Field dimension	1000×100	m ²
Number of sensor (N)	110	sensors
Initial Energy	0.1	Joule
EEFS	10^{-12}	Joule
EEMP	0.0013×10^{-13}	Joule
Eelec	50×10^{-9}	Joule
EDA	50×10^{-9}	Joule
Etrans	50×10^{-9}	Joule
E_{RX}	50×10^{-9}	Joule
Message length (L)	4000	Bits
Round (r)	2000	-
α	0.1	-
β	0.2	-
γ	0.3	-
ε	0.99	-
а	1,2,3 10	-
Threshold(T)	8	dB
m _{random}	0.1	m

Table 2. Simulation parameters of the proposed LEACH protocol.

3. Results

As shown in Figure 8b, the node location in the static topology model is immovable, while it was observed that nodes were constantly moving random distances in random directions in the dynamic



Figure 8. The node location (a) dynamic topology model, (b) static topology model.

AS shown in Figure 9, we calculate energy loss, number of alive nodes and data throughput as evaluation index of LEACHs. D-LEACH, Q-LEACH and F-LEACH which sets CH in their own way, perform better than LEACH, which is characterized as random setting of CH.



Figure 9. LEACH results on energy loss, number of alive nodes, throughput. (**a**–**c**) is the result from dynamic topology model and (**d**–**f**) is the result from static topology model.

In the static topology model, he performance of D-LEACH and F-LEACH were similar, but in the case of moving nodes in the dynamic topology model, the performance of F-LEACH is significantly lowered compared to D-LEACH. In particular, the significant reduction in the life time of the node in F_LEACH at the beginning increases energy consumption and decreases throughput significantly.

As a result of Q-LEACH & F-LEACH and existing LEACH (LEACH, D-LEACH) on the aspects of energy loss, number of alive nodes and throughput, Q-LEACH showed the best performance both when a node was immobile/mobile. Q-LEACH on the static node model shows constantly better performance; on the other hand, at the beginning of operation, Q-LEACH and D-LEACH in the dynamic node model shows a similar tendency; however, over time, it can be found that the performance of Q-LEACH has been greatly improved by the reinforced learning effect.

Figure 10 shows the comparison of various LEACH protocols, and the percentage of the life time of node each system compared to original LEACH is marked [56,57].



Figure 10. The comparison with related work.

As in this study, Q-table is applicable since the table is only 4,000,000 (1000 (x grid) \times 1000 (y grid) \times 4 (up, down, left, right)) but in the most of application such as invade game the overall calculation is determined by the number of pixels. As related studies on comparison of RL algorithms shows that performance of RL algorithms are showing environment dependence [58]. The environment of this study has the same structure as the OpenAI gym environment [59], and this allows developers to easily and quickly implement the environment by sharing the environment. As can be seen in the pseudocode of Q-LEACH, we tried to improve the performance optimized for the situation by customizing the logic in the openAI environment. In addition, we presented as experimental results that the Q-table showed better performance than other RL methods. The Q-network is applied to a vast number of cases situation, and the Q-table is more suitable for this study because it shows better results in finite situations. As mentioned above, performance improvement is observed in Q-LEACH compared to other LEACH methods.

4. Discussion

Existing research presupposes static topology. However, in real-world environments where LEACH is applied, nodes change dynamically over time. Thus, in this study, the efficiency of LEACH within the dynamic topology was increased through Q-LEACH.

As the reinforcement learning is an alternative solution for optimizing the routing. It can achieve low latency, high throughput, and adaptive routing [60]. Various type of reinforcement learning method are applied to routing [61]. Among these RL methods, Q-table was applied in this paper to perform Q-learning in OpenAI environment. Existing research required extra effort to build its own environment, but we can contribute to performance improvement by configuring an optimized situation with an open source environment applying OpenAI's gym environment.

In order to optimize routing in each network, it is necessary to develop this study by detailing the specific tasks defined for existing networking problems. In addition, it is necessary to introduce a number of recently developed reinforcement learning methods (Deep Deterministic Policy Gradient, Advantage Actor Critic, etc.) into LEACH to be applied in real life.

In the static topology model, there is the need to consider LEACH with multi-hop, whereas, in the dynamic topology model, a multi-hop system is not efficient in the aspect of energy consumption as nodes composing the topology are constantly moving. It seems necessary to calculate LEACH through more well-defined dynamic modeling by applying various variables that occur in actual situation.

5. Conclusions

Extending network lifetime is still an important issue in WSN of Autonomous Things Environment. Our study aimed the extended the network lifetime even if there is node topology changes. To simulate and evaluate, the basic LEACH protocol and implementation of the proposed algorithm are used in the statistic and dynamic topology models. The rationality of the dynamic node modeling is shown in the similarity on tendency of energy consumption of both models. The difference between the static topology model and the dynamic topology model proved limitations of LEACH on fixed topology. It is shown in Figure 9a against Figure 9d.

In detail, the dynamic node model, the initial life time of node significantly overall decreased as the distance between nodes continuously increasing, and the effect of energy consumption, which is calculated from the distance between node and CH, also increased. In other words, selecting cluster heads efficiently is more important in the dynamic node model than that of the static model. As the nodes move continuously, the location of all nodes tends to diverge. D-LEACH divides a given field into a constant division, so the location of the node is not significantly affected by the divergence; therefore, with both models, D-LEACH shows similar results, while F-LEACH is backward in the dynamic model. In the diverging case with F-LEACH, the effect of uncertainty problem is emerged as the CH is defined from the fuzzifier values and elimination of outliers. This is seen as the result of uncertainty of fuzzy constant, which makes it difficult to select CH because the fuzzy constant plays a decisive role in the membership function. While F-LEACH is a calculation of approaching to the CH, Q-LEACH reflects the random location of a node by calculating the probability of all directions through rewards of SINR. As a result, Q-LEACH eventually achieved the best throughput with the least energy.

Author Contributions: Conceptualization, J.H.C.; methodology, J.H.C.; software, H.L.; validation, J.H.C., H.L.; formal analysis, H.L.; investigation, H.L.; resources, J.H.C.; data curation, J.H.C.; writing—original draft preparation, H.L.; writing—review and editing, J.H.C.; visualization, H.L.; supervision, J.H.C.; project administration, J.H.C.; funding acquisition, J.H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Institute of Korea Health Industry Development Institute (KHIDI), grant number HI19C1032 and The APC was funded by Ministry of Health and Welfare (MOHW).

Acknowledgments: This work was supported by Institute of Korea Health Industry Development Institute (KHIDI) grant funded by the Korea government (Ministry of Health and Welfare, MOHW) (No. HI19C1032, Development of autonomous defense-type security technology and management system for strengthening cloud-based CDM security).

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Hefeeda, M.; Bagheri, M. Forest fire modeling and early detection using wireless sensor networks. *Ad Hoc Sens. Wirel. Netw.* **2009**, *7*, 169–224.
- 2. Hong, S.H.; Kim, H.K. The Algorithm for an Energy-efficient Particle Sensor Applied LEACH Routing Protocol in Wireless Sensor Networks. *J. Korea Soc. Simul.* **2009**, *18*, 13–21.
- 3. Lee, J.-Y.; Jung, K.-D.; Moon, S.-J.; Jeong, H.-Y. Improvement on LEACH protocol of a wide-area wireless sensor network. *Multimed. Tools Appl.* **2017**, *76*, 19843–19860.
- 4. Cho, D.H. The DHL-LEACH for Improving the Absence of Cluster Head Nodes in a Layer. *J. Inst. Electron. Eng. Korea* **2020**, *57*, 109–115.
- 5. Yassein, M.B.; Aljawarneh, S.; Al-huthaifi, R.K. Enhancements of LEACH protocol: Security and open issues. In 2017 International Conference on Engineering and Technology (ICET); IEEE: Piscataway, NJ, USA, 2017; pp. 1–8.
- 6. Ma, X.; Tao, Z.; Wang, Y.; Yu, H.; Wang, Y. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transp. Res. Part C Emerg. Technol.* **2015**, *54*, 187–197. [CrossRef]
- 7. Kim, D.Y.; Choi, J.W.; Choi, J.W. Reinforce learning based cooperative sensing for cognitive radio networks. *J. Korea Inst. Electron. Commun. Sci.* **2018**, *13*, 1043–1049.
- 8. Choi, H.; Cho, H.; Baek, Y. An Energy-Efficient Clustering Mechanism Considering Overlap Avoidance in Wireless Sensor Networks. In *ISCN'08*; Boğaziçi University: Istanbul, Turkey, 2008; p. 84.
- 9. Shnayder, V.; Chen, B.-R.; Lorincz, K.; Fulford-Jones, T.R.; Welsh, M. *Sensor Networks for Medical Care*; Technical Report; Harvard University: Cambridge, MA, USA, 2005.
- Wood, A.; Virone, G.; Doan, T.; Cao, Q.; Selavo, L.; Wu, Y.; Stankovic, J. ALARM-NET: Wireless Sensor Networks for Assisted-Living and Residential Monitoring; University of Virginia Computer Science Department Technical Report; University of Virginia: Charlottesville, VA, USA, 2006; Volume 2, p. 17.
- 11. Mišić, J.; Mišić, V.B. Implementation of security policy for clinical information systems over wireless sensor networks. *Ad Hoc Netw.* 2007, *5*, 134–144. [CrossRef]
- 12. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Netw.* **2002**, *38*, 393–422. [CrossRef]
- 13. Lee, C.-S.; Lee, K.-H. A contents-based anomaly detection scheme in WSNs. *J. Inst. Electron. Eng. Korea CI* **2011**, *48*, 99–106.
- 14. Jain, S.K.; Kesswani, N. Smart judiciary system: A smart dust based IoT application. In *International Conference* on *Emerging Technologies in Computer Engineering*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 128–140.
- Sathyan, S.; Pulari, S.R. A deeper insight on developments and real-time applications of smart dust particle sensor technology. In *Computational Vision and Bio Inspired Computing*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 193–204.
- Mal-Sarkar, S.; Sikder, I.U.; Konangi, V.K. Application of wireless sensor networks in forest fire detection under uncertainty. In Proceedings of the 2010 13th International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 23–25 December 2010; IEEE: Piscataway, NJ, USA; pp. 193–197.
- 17. Patil, R.; Kohir, V.V. Energy efficient flat and hierarchical routing protocols in wireless sensor networks: A survey. *IOSR J. Electron. Commun. Eng. (IOSR–JECE)* **2016**, *11*, 24–32.
- Heinzelman, W.R.; Chandrakasan, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2000; IEEE: Piscataway, NJ, USA; Volume 2, p. 10.
- 19. Singh, S.K.; Kumar, P.; Singh, J.P. A survey on successors of LEACH protocol. *IEEE Access* 2017, 5, 4298–4328. [CrossRef]
- 20. Renugadevi, G.; Sumithra, M. An analysis on LEACH-mobile protocol for mobile wireless sensor networks. *Int. J. Comput. Appl.* **2013**, *65*, 38–42.
- Tripathi, M.; Gaur, M.S.; Laxmi, V.; Battula, R. Energy efficient LEACH-C protocol for wireless sensor network. In Proceedings of the Third International Conference on Computational Intelligence and Information Technology (CIIT 2013), Mumbai, India, 18–19 October 2013.
- 22. Heinzelman, W.B. Application-Specific Protocol Architectures for Wireless Networks. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, June 2000.

- Voigt, T.; Dunkels, A.; Alonso, J.; Ritter, H.; Schiller, J. Solar-aware clustering in wireless sensor networks. In Proceedings of the ISCC 2004. Ninth International Symposium on Computers and Communications (IEEE Cat. No. 04TH8769), Alexandria, Egypt, 28–31 July 2004; IEEE: Piscataway, NJ, USA; Volume 1, pp. 238–243.
- 24. Biradar, R.V.; Sawant, S.; Mudholkar, R.; Patil, V. Multihop routing in self-organizing wireless sensor networks. *Int. J. Comput. Sci. Issues (IJCSI)* **2011**, *8*, 155.
- Kumar, N.; Kaur, J. Improved leach protocol for wireless sensor networks. In Proceedings of the 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing, Wuhan, China, 23–25 September 2011; IEEE: Piscataway, NJ, USA; pp. 1–5.
- Kumar, N.; Bhutani, P.; Mishra, P. U-LEACH: A novel routing protocol for heterogeneous Wireless Sensor Networks. In Proceedings of the 2012 International Conference on Communication, Information & Computing Technology (ICCICT), Mumbai, India, 19–20 October 2012; IEEE: Piscataway, NJ, USA; pp. 1–4.
- 27. Sindhwani, N.; Vaid, R. V LEACH: An energy efficient communication protocol for WSN. *Mech. Confab* **2013**, 2, 79–84.
- Kumar, S.V.; Pal, A. Assisted-leach (a-leach) energy efficient routing protocol for wireless sensor networks. *Int. J. Comput. Commun. Eng.* 2013, 2, 420–424. [CrossRef]
- 29. Khediri, S.E.; Nasri, N.; Wei, A.; Kachouri, A. A new approach for clustering in wireless sensors networks based on LEACH. *Procedia Comput. Sci.* **2014**, *32*, 1180–1185. [CrossRef]
- 30. Xu, R.; Wunsch, D. Clustering; John Wiley & Sons: Hoboken, NJ, USA, 2008.
- Cherkassky, V.; Ma, Y. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Netw.* 2004, 17, 113–126. [PubMed]
- 32. Linda, O.; Manic, M. General type-2 fuzzy c-means algorithm for uncertain fuzzy clustering. *IEEE Trans. Fuzzy Syst.* **2012**, *20*, 883–897. [CrossRef]
- 33. Hwang, C.; Rhee, F.C.-H. Uncertain fuzzy clustering: Interval type-2 fuzzy approach to \$ c \$-means. *IEEE Trans. Fuzzy Syst.* **2007**, *15*, 107–120. [CrossRef]
- Kaur, P.; Lamba, I.; Gosain, A. Kernelized type-2 fuzzy c-means clustering algorithm in segmentation of noisy medical images. In Proceedings of the 2011 IEEE Recent Advances in Intelligent Computational Systems, Trivandrum, Kerala, India, 22–24 September 2011; IEEE: Piscataway, NJ, USA; pp. 493–498.
- 35. Zarandi, M.F.; Gamasaee, R.; Turksen, I. A type-2 fuzzy c-regression clustering algorithm for Takagi–Sugeno system identification and its application in the steel industry. *Inf. Sci.* **2012**, *187*, 179–203. [CrossRef]
- Raza, M.A.; Rhee, F.C.-H. Interval type-2 approach to kernel possibilistic c-means clustering. In Proceedings of the 2012 IEEE International Conference on Fuzzy Systems, Brisbane, QLD, Australia, 10–15 June 2012; IEEE: Piscataway, NJ, USA; pp. 1–7.
- Rubio, E.; Castillo, O.; Valdez, F.; Melin, P.; Gonzalez, C.I.; Martinez, G. An extension of the fuzzy possibilistic clustering algorithm using type-2 fuzzy logic techniques. *Advances in Fuzzy Systems* 2017, 2017, 1–23. [CrossRef]
- Zarinbal, M.; Zarandi, M.F.; Turksen, I. Interval type-2 relative entropy fuzzy C-means clustering. *Inf. Sci.* 2014, 272, 49–72. [CrossRef]
- Rubio, E.; Castillo, O. Optimization of the interval type-2 fuzzy C-means using particle swarm optimization. In Proceedings of the 2013 World Congress on Nature and Biologically Inspired Computing, Fargo, ND, USA, 12–14 August 2013; IEEE: Piscataway, NJ, USA; pp. 10–15.
- 40. Ngo, L.T.; Dang, T.H.; Pedrycz, W. Towards interval-valued fuzzy set-based collaborative fuzzy clustering algorithms. *Pattern Recognit.* **2018**, *81*, 404–416. [CrossRef]
- 41. Yang, X.; Yu, F.; Pedrycz, W. Typical Characteristics-based Type-2 Fuzzy C-Means Algorithm. *IEEE Trans. Fuzzy Syst.* **2020**. [CrossRef]
- 42. Kaelbling, L.P.; Littman, M.L.; Moore, A.W. Reinforcement learning: A survey. J. Artif. Intell. Res. 1996, 4, 237–285. [CrossRef]
- 43. Hsu, R.C.; Liu, C.-T. A reinforcement learning agent for dynamic power management in embedded systems. *J. Internet Technol.* **2008**, *9*, 347–353.
- 44. Chitsaz, M.; Woo, C.S. Software agent with reinforcement learning approach for medical image segmentation. *J. Comput. Sci. Technol.* **2011**, *26*, 247–255. [CrossRef]
- 45. Prabuchandran, K.; Meena, S.K.; Bhatnagar, S. Q-learning based energy management policies for a single sensor node with finite buffer. *IEEE Wirel. Commun. Lett.* **2012**, *2*, 82–85. [CrossRef]
- 46. Ernst, D.; Glavic, M.; Wehenkel, L. Power systems stability control: Reinforcement learning framework. *IEEE Trans. Power Syst.* **2004**, *19*, 427–435.

- 47. Hsu, R.C.; Liu, C.-T.; Wang, H.-L. A reinforcement learning-based ToD provisioning dynamic power management for sustainable operation of energy harvesting wireless sensor node. *IEEE Trans. Emerg. Top. Comput.* **2014**, *2*, 181–191. [CrossRef]
- 48. Chen, H.; Li, X.; Zhao, F. A reinforcement learning-based sleep scheduling algorithm for desired area coverage in solar-powered wireless sensor networks. *IEEE Sens. J.* **2016**, *16*, 2763–2774. [CrossRef]
- 49. Sharma, A.; Chauhan, S. A distributed reinforcement learning based sensor node scheduling algorithm for coverage and connectivity maintenance in wireless sensor network. *Wirel. Netw.* **2020**, *26*, 4411–4429. [CrossRef]
- Yau, K.-L.A.; Goh, H.G.; Chieng, D.; Kwong, K.H. Application of reinforcement learning to wireless sensor networks: Models and algorithms. *Computing* 2015, 97, 1045–1075. [CrossRef]
- Shah, K.; Kumar, M. Distributed independent reinforcement learning (DIRL) approach to resource management in wireless sensor networks. In Proceedings of the 2007 IEEE International Conference on Mobile Adhoc and Sensor Systems, Pisa, Italy, 8–11 October 2007; IEEE: Piscataway, NJ, USA; pp. 1–9.
- Liu, T.; Martonosi, M. Impala: A middleware system for managing autonomic, parallel sensor systems. In Proceedings of the Ninth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, San Diego, CA, USA, 11–13 June 2003; pp. 107–118.
- 53. Sutton, R.S.; Barto, A.G. Reinforcement Learning: An Introduction; MIT Press: Cambridge, MA, USA, 2011.
- 54. Solaris. Learning Deep Learning Using Tensorflow; Youngjin.com publisher: Seoul, Korea, 2019; pp. 322–324.
- 55. Dembla, D.; Shivam, H. Analysis and implementation of improved-LEACH protocol for Wireless Sensor Network (I-LEACH). *IJCSC IJ* 2013, *4*, 8–12.
- 56. Khan, F.A.; Ahmad, A.; Imran, M. Energy optimization of PR-LEACH routing scheme using distance awareness in internet of things networks. *Int. J. Parallel Program.* **2020**, *48*, 244–263. [CrossRef]
- Alnawafa, E.; Marghescu, I. IMHT: Improved MHT-LEACH protocol for wireless sensor networks. In Proceedings of the 2017 8th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 4–6 April 2017; IEEE: Piscataway, NJ, USA; pp. 246–251.
- Sharma, J.; Andersen, P.A.; Granmo, O.C.; Goodwin, M. Deep Q-Learning with Q-Matrix Transfer Learning for Novel Fire Evacuation Environment. In *IEEE Transactions on Systems, Man, and Cybernetics: Systems*; IEEE: Piscataway, NJ, USA, 2020.
- 59. Dutta, S. Reinforcement Learning with TensorFlow: A Beginner's Guide to Designing Self-Learning Systems with TensorFlow and OpenAI Gym; Packt Publishing Ltd: Birmingham, UK, 2018.
- 60. Sun, P.; Hu, Y.; Lan, J.; Tian, L.; Chen, M. TIDE: Time-relevant deep reinforcement learning for routing optimization. *Future Gener. Comput. Syst.* **2019**, *99*, 401–409.
- 61. Yu, C.; Lan, J.; Guo, Z.; Hu, Y. DROM: Optimizing the routing in software-defined networks with deep reinforcement learning. *IEEE Access* 2018, *6*, 64533–64539. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).