Centralized Q-Learning based Routing in EH-WSNs with Dual Alternative Batteries

Tian Zhao¹*, XiaoBin Xu¹,a, ShangGuang Wang²,b

¹Beijing Advanced Innovation Center for Future Internet Technology, Beijing University of Technology, Beijing, China
²State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China

*xuxiaobin@bjut.edu.cn, sgwang@bupt.edu.cn

ABSTRACT Energy harvesting wireless sensor network (EH-WSN) uses a large number of sensor nodes to conduct environmental perception, data collection and data transmission through wireless links. However, the battery capacity of traditional energy harvesting wireless sensor is limited. In this paper, a kind of energy collection wireless sensor of dual-battery system is studied, and a centralized energy-aware routing algorithm based on reinforcement learning is proposed. Considering the energy of the overall link, each source node chooses a good path at the beginning of data transmission. After all the data is transmitted to the sink node, it selects the path according to the whole network state, and learns several times until it finds the best path choice. The simulation results show that the end-to-end delay of this protocol is lower than that of random packet forwarding.

1. INTRODUCTION
As Wireless Sensor Networks (WSNs) has a broad application prospect, it has been a research topic [1]. A WSN consists of a large number of small and low-cost sensor nodes. Sensor nodes can cooperatively perceive the environment and collect data [2][3]. Data transmission is conducted between sensor nodes through wireless links. The data collected by sensor nodes is transmitted over multiple hops to one or more sink nodes for further processing. Due to the diverse environmental adaptability of WSNs, WSNs are widely used in environmental monitoring, military supervision or disaster management for information collection and data transmission [4][5]. However, the energy supply of traditional WSNs is limited, many challenging research problems need to be solved, such as energy optimization, topology control, power control, routing, cross-layer optimization and coverage [6].

In the energy harvesting wireless sensor network (EH-WSN), the sensor nodes are equipped with rechargeable batteries [7]. Energy harvesting technology allows sensor nodes to harvest energy from their surroundings and store it in rechargeable batteries, which can be used to power their own loads (such as data collection and transmission). Unlike traditional WSNs with sensors equipped with non-rechargeable batteries, the EH-WSN has longer network lifetime [8] and the maintenance cost of farmland [9]. Therefore, the energy collection wireless sensor network is suitable for applications with long operating cycles [10]. However, sensor nodes in traditional EH-WSNs cannot simultaneously harvest energy and power the load.
In order to optimize the use of sensor nodes energy, in this paper, we mainly consider a routing algorithm in dual-battery system EH-WSNs. In this type of EH-WSNs, the sensor batteries are equipped with a dual-battery system [11], where the nodes can both harvest energy and power their load. The dual-battery system is used to address the problem whereby a sensor node cannot discharge and charge its battery simultaneously. From Figure 1, we see that battery I is recharged by the solar energy harvester, and in the meantime, battery II is being discharged to power the load (such as maintaining the work of the sensor node and energy consumption for data transmission). Another important consideration is that a battery loses its capacity if it is recharged when it is only partially charged [12]. This involves the two pieces of battery charging and discharging role transition conditions. Thus, batteries must change role only when a battery is fully charged and the other is fully discharged [11].

Figure 1: An EH-WSN where nodes have a dual alternative battery model.

Sensor nodes must route their collected data to a sink or a fusion center. In this respect, there are many works that aim to maximize network throughput or minimize the energy consumption of a EH-WSN by designing an energy aware routing protocol. Example works include [13], [14] and [15]. The following work mainly involves the idea of centralized algorithm and the use of energy aware to design the routing protocol.

2. RELATED WORK

In this section, we discuss previous works that address the routing problem in EH-WSNs. Authors in [16] proposed a modified energy aware routing protocol based on Ad-hoc On-demand Multipath Distance Vector (EA-AOMDV) for mobile ad-hoc networks. In this protocol, paths are selected based on their energy metric instead of hop-count metric to maximize the network operational time and minimize the average consumed energy. EA-AOMDV is a centralized algorithm based on multipath energy metric. The best path with the highest energy metric will be used first, the other paths are backups. The routing algorithm uses the method of comprehensive measurement, it avoids the case that single metric selection makes the nodes in the path lose power quickly.

In [17], authors propose a loss-aware routing algorithm for EH-WSNs. The main idea is to calculate the link cost based on the energy of sensor nodes. The routing decision is made based on energy cost which consists of the energy consumed by data transmission and network energy wastage due to overcharging of batteries with finite capacity. Through predicting the amount of future harvested energy and energy consumption, the total residual energy of the network can be calculated by adding the current residual energy and the harvested energy of all nodes, then subtracting the consumption of data transmission. The algorithm selects the route that maximizes the total residual energy of the network.

Authors in [18] proposed a centralized Trust-based Efficient Routing protocol for wireless sensor networks, called CENTER. This protocol utilizes the powerful sink base station (BS) to identify and disable different types of misbehaving nodes that may interrupt or abuse the functionality of the WSN. BS calculates different quality measures, namely the level of malice, cooperation, and compatibility, approximates the life of battery, and evaluates Data Trust and Forward Trust values for each node. The BS then isolates all nodes with bad states, whether misbehaving or malicious, based on their history. Finally, the BS uses an efficient method to propagate updated routing information, indicating the up-
links and the next hop downlink for every node. Through its centralized approach, the protocol provides more efficient and secure routing while considering about the energy-constrained sensor nodes.

However, in previous work, it was assumed that the sensor node was equipped with a single battery. Hence, conventional routing algorithms cannot be directly applied to EH-WSNs with a dual-battery system. This is because they do not take into account the difference in charging and discharging behavior of nodes with a dual-battery system. In particular, nodes on a route may have completely discharged one battery while the other waits to be fully charged; in this case, nodes experience an energy outage. Compounding to this challenge is that the energy arrivals at sensor nodes exhibit spatio-temporal behaviors, and are generally random [7].

Henceforth, we propose a centralized Q-learning based routing algorithm that choose the optimal path combination and select several paths as alternatives after learning. Our goal is to route packets on paths that have minimal energy outages; equivalently, each source node learns the best path combination that have the lowest end-to-end delay to the sink. To the best of our knowledge, this work is the first to consider a learning approach to route data over sensor nodes with dual batteries. Compared to existing energy-aware routing algorithms, this work has the following advantages: 1) it considers a practical charging and discharging EH model via a dual-battery system; this ensures that the nodes maintain battery life for as long as possible [19], 2) the proposed routing algorithm operates over random energy arrivals, and 3) the source nodes only need to select the path in the initial stage, and the intermediate sensor nodes can transfer data according to the initial selection, so that the next hop selection is not necessary; thus our proposed protocol is suitable for use in large-scale EH-WSNs.

Next, we present a model of EH-WSNs that use a dual-battery system. After that in Section IV, we present our proposed centralized Q-learning based routing algorithm (CQR) and describe a novel reward function. The simulation results in Section V show that nodes are able to make the optimal routing decision with lower end-to-end delay than a competing routing algorithm. Section VI concludes the paper.

3. PRELIMINARIES

We model an EH-WSN as a graph \( G(\mathbf{v}, \mathbf{e}) \), where \( \mathbf{v} \) denotes the set of nodes indexed by \( i = \{1, ..., |\mathbf{v}|\} \), and \( \mathbf{e} \) denotes the set of links. Let \( \mathbf{S} \subset \mathbf{v} \) denote the set of source nodes. We consider \( G \) as a Destination Oriented Directed Acyclic Graph (DODAG) [14]. Each DODAG has only one root node denoted by \( s \in \mathbf{v} \); it has unlimited energy and data storage buffer. A DODAG can be generated using Routing Protocol for Low-Power and Lossy Networks (RPL) [20], where it is used to construct a DODAG that minimizes the number of hops from a node to the root. RPL also provides each node with a rank value that indicates its location relative to other nodes and the root node. The root node has a rank of zero. In addition, each node maintains a parent set containing neighbors with a smaller rank. Each node periodically sends its Destination Advertisement Object (DAO) to its selected parent(s) as feedback to maintain the generated DODAG. The set of parents for node \( i \) is denoted as \( \mathbf{N}_i \subset \mathbf{v} \).

Data flow from one source node can be split and forwarded through multiple paths, so called multi-path routing. Time is slotted and indexed by \( t \in T \), and each one-hop transmission will take one time slot. Let \( \mathbf{d}^t_i \) indicate the data storage level at node \( i \) in slot \( t \). We assume all nodes have unlimited data storage. As our focus is on routing, we assume each node is able to transmit/receive to/from multiple neighbors at the same time using techniques such as FDMA [21] or CDMA [22]. For simplicity, we do not consider the energy cost of data reception and system operation.

Each node is equipped with dual alternating batteries [11], which we refer to as \( A \) and \( B \). Both batteries have capacity \( B_{max} \). We assume each node \( i \) in each slot \( t \) is able to harvest a random amount of energy \( \Delta E \in [0,B_{max}] \) [23]. Also, the switching time of the dual alternative batteries is negligible. We use \( R_{IA}^t \) and \( R_{IB}^t \) to respectively denote the residual energy in battery \( A \) and \( B \) in time slot \( t \) at node \( i \). We define an indicator \( \mathbb{I}_{ix}^t \in \{0,1\} \) to denote whether battery \( A \) is discharging at time \( t \). For example, if battery \( A \) is discharging and battery \( B \) is recharging, then \( \mathbb{I}_{ix}^t \) is one; otherwise,
the value of $I_{iA}^t$ is zero. We use $E_{iA}^t$ to denote the residual energy at node $i$ in slot $t$, and it is defined as,

$$E_{iA}^t = I_{iA}^t \left( R_{iA}^t + \frac{R_{iB}^t + \Delta t^t}{B_{max}} B_{max} \right)$$

$$+ \left( 1 - I_{iA}^t \right) \left( R_{iB}^t + \frac{R_{iB}^t + \Delta t^t}{B_{max}} B_{max} \right)$$

(1)

According to Eq. (1), if battery $A$ of node $i$ is discharging, the value of $E_{iA}^t$ is $R_{iA}^t + \frac{R_{iB}^t + \Delta t^t}{B_{max}} B_{max}$. Moreover, if battery $B$ is fully charged, e.g., $R_{iB}^t + \Delta t^t \geq B_{max}$, the residual energy of node $i$ is $R_{iA}^t + B_{max}$; otherwise, $E_{iA}^t$ equals $R_{iA}^t$. On the other hand, if battery $B$ is discharging and $A$ is recharging, $E_{iA}^t$ equals $R_{iB}^t + B_{max}$ when battery $A$ is fully charged, and it equals $R_{iB}^t$ if $A$ is not fully charged.

4.1 CENTRALIZED Q-LEARNING BASED ROUTING ALGORITHM

Firstly, we briefly introduce a Markov Decision Process (MDP) [26]. A MDP uses a tuple $(S, A, P_a(s, s'), R_a(s, s'))$, where $S$ is a finite set of states, $A$ is a finite set of actions, $R_a(s, s')$ is the transition probability from state $s$ to $s'$, and $R(s, a)$ is the immediate reward for choosing action $a$ in state $s$. For each state $s$, an action is selected using policy $\pi(s)$. Each agent learns and stores the value for each action $a$, so called Q-value $Q(a)$, which corresponds to the reward for selecting action $a$ at an episode. The value of $Q(a)$ is calculated as per [23]:

$$Q(a') \leftarrow (1 - \alpha)Q(a) + \alpha [R(a) + \gamma MAXQ(a')]$$

(2)

Where $\alpha$ is the learning rate, $\gamma$ is the discount factor, $R(a)$ denotes the reward, and $MAXQ(a')$ indicates the maximum value in the next episode. An agent then chooses the action $a$ with the highest $Q(a)$ value when it is in an episode.

We are ready to present our centralized reinforcement learning based multi-path routing algorithm, named CQR. Each source node selects an action using the $\epsilon$-greedy algorithm [24] based on stored Q-values. It then forwards data to the next hop(s) parent(s) contained in the selected action. The node transfers the data along the path in the action. Each sensor node in the selected path(s) then calculates and updates the energy and data information at the time slot. After transmission, each source node then calculates and updates the Q-value of the action according to the residual energy of nodes in the action and the end-to-end time delay used.

Actions: All source nodes are considered as a whole and act as an agent. The paths that all source nodes can choose are randomly combined into multiple sets that does not repeat. An action of an agent is defined as a set. That is, an action represents one or more paths. Thus, a source node can select one or several paths. In each episode, an agent applies the $\epsilon$-greedy algorithm to select an action. That is, in each episode, an agent randomly generates a probability value $p \in [0,1]$. If $p$ is greater than or equal to $\epsilon$, the agent will select the action with the maximum $Q(a)$ value in the episode. If $p$ is less than $\epsilon$, this node $i$ will randomly select an action $a$.

Updating Q-values: The agent updates the $Q(a)$ value for action $a$ as follows,

$$Q(a') \leftarrow (1 - \alpha)Q(a) + \alpha R(a)$$

(3)

where $Q(a)$ and $Q(a')$ are the Q-values of action $a$ at the agent in the current episode and the next episode respectively. The last term computes the reward $R(a)$ value in action $a$.

Reward function: The agent calculates the reward of each action based on the information after all the data is transferred to the sink. But we are still calculating the time that it takes for the data transmission to occur within each time $t$. In each slot $t$, each node $i$ sends an information tuple $S_i^t = \{E_i^t, d_i^t\}$ to its children in the selected path, where $E_i^t$ denotes the residual energy, and $d_i^t$ indicates the amount of data in its buffer. Let $e_t$ denote the energy cost of transmitting one unit of data in one time slot. Each node $i$ first calculates the estimated residual energy $w_j^t$ of parent $j \in N_i$ in slot $t$ as,

$$w_j^t = E_j^t - d_j^t e_t$$

(4)
The computation of these variables involves the time slots for data transmission. At the end of a learning loop, $MRE$ denotes the minimum residual energy of a node in the path. $MRE$ will be updated according to Eq. (5).

$$MRE = \min_{n \in P} E(n)$$  \hspace{1cm} (5)

Where $P$ is the set of all nodes in the path, $E(n)$ is the residual energy of node $n$. Then the residual energy of the intermediate node is updated according to Eq. (6).

$$PRE = \sum_{n \in P} E(n)$$  \hspace{1cm} (6)

Where $PRE$ is the path residual energy, that is, the residual energy of nodes on this path. After the data transmission, based on the value of $MRE$ and $PRE$, the agent calculates the energy metric of this path according to Eq. (7) and Eq. (8).

$$ARE = \frac{PRE}{N}$$  \hspace{1cm} (7)

$$EM(P) = 0.6(MRE) + 0.4(ARE)$$  \hspace{1cm} (8)

Where $ARE$ is the path average residual energy and $EM$ is the path energy metric, $N$ denotes the number of nodes on the path. $P$ indicates the path in the selected action. $MRE$ is given higher weight than $ARE$ to affect more in the path discovery as our proposed protocol tries to avoid paths with low power nodes.

Recall that an action $a$ contains one or more paths. Reward $R(a)$ of action $a$ is defined as the mean of the energy metric of the contained path minus the time slots consumed by the action $a$. Mathematically,

$$R(a) = \frac{\sum_{P \in a} EM(P)}{N(a)} - t$$  \hspace{1cm} (9)

Intuitively, if the action $a$ takes less time slots, the reward value is higher.

Algorithm 1 shows the steps of the proposed centralized Q-learning based routing algorithm. Initially, all the source nodes are considered as an agent, and each action of the agent generates a Q-value. There will be one or more transmission paths from the source nodes to sink, then we combine all the alternative paths of the source nodes without repeating. At the start of the learning round, the agent uses the $\epsilon$-greedy algorithm to select an action. An action contains a set of multiple paths, that is, each source node can choose one or more paths for data transmission. When the transmission starts, the node can only forward the data according to the paths in the action, and calculate the used time slots after all the data is transferred to sink.

**Algorithm 1** The proposed Centralized Q-learning based Routing Algorithm

**Input:** set of all sensor nodes $\nu$; the total amount of data $D$; battery information $E_i$; data information $d_i$; $Q$-value; number of rounds;

**Output:** A feasible routing scheme

1. for $i \in \nu$ do
2. exchange node information including $E_i$ and $d_i$;
3. end
4. Find the available path from source nodes to the sink;
5. Initializes the $Q(a)$ for all Action;
6. repeat (for each round):
7. choose $a$ using $\epsilon$-greedy policy derived from $Q$;
8. take action $a$, transmit data;
9. while data is all transmitted to sink do
\[ Q(a') \leftarrow (1 - \alpha)Q(a) + \alpha R(a) \]

calculate the time slots of data transmission \( t \);

update node information;

end while

until round is over;

Find the optimal Action.

Next, node \( i \) forwards data to the parent(s) included in the selected action \( a \). Let \( A_i^{t+} \) denote the maximal amount of data that node \( i \) transmits in slot \( t \) that is limited by its residual energy in the previous time slot \( E_i^{t-1} \) and the amount of data in its buffer \( d_i^{t-1} \). Formally,

\[ A_i^{t+} = \text{MIN}(d_i^{t-1}, E_i^{t-1}/\epsilon_t) \tag{10} \]

The total energy consumption of data transmission is thus \( A_i^{t+} \epsilon_t \). If there are several parents in \( a \), data will be split and forwarded to different parents. Moreover, the amount of data for each parent node is proportional to the residual energy of that parent. Let \( \lambda_{ij}^t \) denote the amount of data transmitted from node \( i \) to parent \( j \) in slot \( t \), which is calculated as follows,

\[ \lambda_{ij}^t = \frac{\Lambda_i^{t+} E_i^{t-1}}{\Sigma_{j \epsilon a} E_j^{t-1}} \tag{11} \]

The sum of forwarded data from \( i \) to parent nodes must equal to the total transmitted data. Thus, we have,

\[ \Lambda_i^{t+} = \sum_{j \epsilon a} \lambda_{ij}^t \tag{12} \]

We use \( \Lambda_i^{t-} \) to denote the total amount of data received by node \( i \) in slot \( t \). The data storage level of \( i \) in slot \( t \) is,

\[ \Lambda_i^{t-} = \sum_{j \epsilon a} \lambda_{ij}^t \tag{13} \]

We use \( \Lambda_i^{t+} \) denote the total amount of data received by node \( i \) in slot \( t \). The data storage level of \( i \) in slot \( t \) is,

\[ d_i^t = d_i^{t-1} - \Lambda_i^{t+} + \Lambda_i^{t-} \tag{14} \]

At each node, the residual energy in both batteries is updated. The dual alternative batteries may change role if a battery is fully discharged and the other is fully charged. Let \( I_{i,s}^t \) denote whether two batteries switch role at node \( i \) in slot \( t \). In this case, \( I_{i,s}^t \) is one. Otherwise, \( I_{i,s}^t \) is zero. The residual energy in battery \( A \) and \( B \) at node \( i \) is updated as follows,

\[ \begin{align*}
R_{iA}^t &= I_{iA}^t(E_i^{t-1} - \Lambda_i^{t+} \epsilon_t) + (1 - I_{iA}^t)(1 - I_{iB}^t)R_{iA}^{t-1} \tag{15} \\
R_{iB}^t &= (1 - I_{iA}^t)(E_i^{t-1} - \Lambda_i^{t+} \epsilon_t) + I_{iA}^t(1 - I_{iB}^t)R_{iB}^{t-1} \tag{16}
\end{align*} \]

we have \( I_{iA}^t = 1 \) if battery \( A \) is discharging after data transmission. According to Eq. (15) and (16), the energy level of the discharging battery is \( E_i^{t-1} - \Lambda_i^{t+} \epsilon_t \). If two batteries switch role, i.e., \( I_{i,s}^t = 1 \), that means one battery is empty and will be recharged, e.g., \( R_{iB}^t = 0 \) and \( I_{iA}^t = 1 \). If two batteries does not switch role, the residual energy level in the discharging battery is the same, e.g., \( R_{iA}^t = R_{iB}^{t-1} \); a if \( I_{i,s}^t \) is zero and \( I_{iA}^t \) is zero. Recall that the residual energy at node \( i \) in slot \( t \) can be calculated using Eq. (1). At the end of slot \( t \), each node \( i \) calculates the estimated residual energy \( w_i^j \) of each parent \( j \in N_i \). At the end of learning episode, the reward \( R(a) \) for action \( a \) according to Eq. (5) through (9). Agent then updates \( Q(a) \) using Eq. (3).

5.EVALUATION

Our experiments are conducted using Matlab with the Matgraph toolbox [27]. On a 100 m² area, we generate a DODAG with 65 sensor nodes including six sources and one sink. Each source node has
one unit of data to transmit to the sink. The communication range of each node is 25 meters. If two nodes are in each their communication range, they can establish links. We set the battery capacity $B_{\text{max}}$ to one Joule. Unless stated explicitly, we use exploration probability $\epsilon = 0.5$. We set the value of $\epsilon$ to be slightly larger, so that the probability of finding a new path is greater. The $\alpha$ in Eq. (3) are set to 0.7 [23]. We define the period of transmitting all data from sources to the sink as one round. The length of a round is measured in slots, which is also the end-to-end delay. We note that the end-to-end delay of a path is mainly affected by the energy outage at nodes.

We first study the effect of the number of source nodes on the end-to-end delay of CQR. We first consider the number of source nodes is four, six and eight, respectively. That is $|S| = \{4,6,8\}$. From Figure 2, when $|S| = 4$, the minimum end-to-end delay is 14 slots. As for $|S| = 6$, the minimum end-to-end delay is 17 slots. When $|S| = 8$, the maximum end-to-end delay is 26 slots, which is 5 slots higher than the maximum end-to-end delay when there are four sources. When the number of sources increases, both the maximum and minimum end-to-end delays increase because the total amount of data that needs to be transmitted to the sink increases. The end-to-end delay fluctuates because the energy harvested by the nodes at each time slot is random.

Figure 2: End-to-end delays with different number of sources.

Next, we study the effect of the number of nodes on the end-to-end delay of CQR. We consider the number of nodes to be 30, 65 and 100, that is $|v| = \{30,65,100\}$. As is shown in Figure 3, when the number of nodes is 30, the minimum end-to-end delay is 17 slots. As for the number of nodes is 65, the minimum end-to-end delay is 12 slots. When the number of nodes is 100, the minimum end-to-end delay is 10 slots. As the number of nodes increases, the minimum end-to-end delay decreases. This is because each sensor node has more next hops, and nodes have more options when establishing path connections. Therefore, for each node, there are more available and shorter paths to choose.
Finally, Figure 4 demonstrates the performance of CQR against a random routing algorithm whereby each node randomly chooses one or more parents to forward its data. From Figure 4, we can see the minimum end-to-end delay of CQR is 12 slots. Moreover, the average end-to-end delay of CQR is 15.57 slots. In comparison to nodes that select parents randomly, the resulting end-to-end delays fluctuate between 21 and 56 slots. Its average end-to-end delay is 38.69 slots, which is larger than the average result of CQR. The end-to-end delay of the random routing algorithm fluctuates because nodes always randomly choose the next hop(s) without learning from experience nor consider the state of its parents. This may lead to energy outage at nodes. On the other hand, in the learning process, CQR results in a high end-to-end delay of 21 slots due to exploration.

6. CONCLUSION
This paper has proposed a centralizes reinforcement learning based multipath routing protocol for EH-WSNs that employ a dual-battery system. Its optimization goal is to minimize the total end-to-end delay caused by energy outages. The node use our protocol to learn iteratively select the best routing set. The results show that the protocol achieves lower end-to-end delay than random packet forwarding. In the future work, we will jointly study routing and link scheduling to minimize the end-to-end latency of the dual battery system in EH-WSNs.

REFERENCES
[1] Bauer, P., Sichitiu, M., Istepanian, R., and Premaratne, K. 2000. The mobile patient: wireless distributed sensor networks for patient monitoring and care. *IEEE Embs International Conference on Information Technology Applications in Biomedicine*. (Nov. 2000), pp. 17-21.
[2] Passmann, C., Brenzel, C., and Meschenmoser, R. 2001. Wireless Inter-Vehicle Communication for Hazard Warning. *IFAC Proceedings Volumes*, 34, 9 (July 2001), 275-277.

[3] Zhang, S., and Zhang, H. 2012. A review of wireless sensor networks and its applications. Automation and Logistics (ICAL), 2012 IEEE International Conference on. (Aug.2012), 386-389.

[4] Atzori, L., Iera, A., and Morabito, G. 2010. The internet of things: a survey. *Computer Networks*, 54, 15 (Oct. 2010), 2787-2805.

[5] Zhao, F., and Guibas, L. 2005. Wireless sensor networks: an information processing approach. Sensor Review, 25, 2 (Jun. 2005), 269 - 270.

[6] Seah, K., Eu, Z., and Tan, H. 2009. Wireless Sensor Networks Powered by Ambient Energy Harvesting (WSN-HEAP)–Survey and Challenges. *IEEE International Conference on Wireless Communication*, 5, 8 (Jun. 2009), 1-5.

[7] Adu-Manu, S., Adam, N., Tapparello, C., Ayatollahi, H., and Heinzelman, B. 2018. Energy-harvesting wireless sensor networks (EH-WSNs): A review. *ACM TOSN*, 14, 2 (Jul 2018), 1-50. DOI=http://doi.acm.org/10.1145/3183338.

[8] Shaikh, K., and Zeadally, S. 2016. Energy harvesting in wireless sensor networks: A comprehensive review. *Renewable and Sustainable Energy Reviews*, 55 (Mar 2016), 1041-054.

[9] Ingelrest, F., Barrenetxea, G., Schaefer, G., VetterliM., and Parlangue, M. 2010. SensorScope: Application-specific sensor network for environmental monitoring. *ACM Transactions on Sensor Networks*, 6, 2 (Feb 2010),1–32. DOI= http://doi.acm.org/10.1145/1689239.1689247.

[10] Cho, S., Park, J. W., Jang, S. A., Jo, H., and Jr, B. F. S. 2011. Structural health monitoring of a cable-stayed bridge using wireless smart sensor technology: Data analyses. *Smart Structures and Systems*, 6, 5 (Jan 2011), 439–459.

[11] Bhat, V., Motani, M., Murthy, R., and Vaze, R. 2018. Energy harvest-ing communications using dual alternating batteries. *arXiv preprint arXiv: 1801.03813*.

[12] Tsuyoshi, S., Yoshio, U., and Petr, N. 2013. Memory effect in a lithium-ion battery. *Nature Materials*, 12, 6 (Dec 2013), 569-675.

[13] Bao, X. and Ding, G. A routing algorithm for maximizing network performance in energy harvesting wireless sensor network. *IEEE ICISCE*, Beijing, China, Jul. 2016.

[14] Han, G., Dong, Y., Hui, G., Lei, S., and Wu, D. 2015 Cross-layer optimized routing in wireless sensor networks with duty cycle and energy harvesting. *Wireless communications & mobile computing*, 15, (Feb 2014).

[15] Luo, S., Zhang, R., and Teng, L. 2013. Optimal save-then-transmit protocol for energy harvesting wireless transmitters. *IEEE Transactions on Wireless Communications*, 12, 3 (Mar 2013)1196-1207.

[16] Shawara, M., Sarhan, A., and Elfishawy, A. 2017. Energy aware Ad-Hoc On Demand Multipath Distance Vector (EA-AOMDV). *13th International Computer Engineering Conference (ICENCO)*, Cairo, (Dec 2017), 317-322.

[17] Martinez, G., Li, S., and Chi, Z. 2014. Wastage-aware routing in energy-harvesting wireless sensor networks. *IEEE Sensors Journal*, 14, 9 (Sept 2014), 2967-2974.

[18] Tajeddine, A., Kayssi, A., and Chehab, A. 2012. CENTER: A Centralized Trust-Based Efficient Routing protocol for wireless sensor networks. Tenth International Conference on Privacy. *IEEE Computer Society*, (July 2012).

[19] Sawada, H., Kuriyama, H., Yusa, N., Mizuno, T., and Mineno, H. 2012. Mutually complementary communication protocol based on destination oriented directed acyclic graph. in *IEEE CCNC*, Las Vegas, NV, USA.

[20] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Postel, K., Syriuk, R., Vasseur, J., and Alexander, R. 2012. *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*. Internet Engineering Task Force (IETF), RFC 6550 Std.
[21] Sari, H., Levy, Y., and Karam, G. 1997. An analysis of orthogonal frequency-division multiple access. In IEEE Globecom, 3 (Dec1997), 1635-1639.
[22] Fattah, H., and Leung, C. 2002. An overview of scheduling algorithms in wireless multimedia networks. IEEE wireless communications, 9, 5 (Nov 2002), 76-83.
[23] Mehrabi, A., and Kim, K. 2016. General framework for network throughput maximization in sink-based energy harvesting wireless sensor networks. IEEE Transactions on Mobile Computing, 16, 7 (Sep 2016), 1881-1896.
[24] Sutton, S., and Barto, G. 2018. Reinforcement learning: An introduction. MIT press.
[25] Watkins, J., and Dayan, P. 1990. Q-learning. Machine learning, 8, 3-4 (May 1992), 279-292.
[26] mathworks, “Matgraph,” https://ww2.mathworks.cn/matlabcentral/fileexchange/19218-matgraph, 2000.
[27] Dong, S., Agrawal, P., and Sivalingam, K. 2007. Reinforcement learning based geographic routing protocol for UWB wireless sensor network. In IEEE Globecom, Washington, DC, USA.