Link Scheduling in Rechargeable Wireless Sensor Networks with Battery Memory Effects

Tony Tony, Sieteng Soh, Mihai Lazarescu
School of Electrical Engineering, Computing and Mathematical Sciences
Curtin University
Perth, WA, Australia
tony@postgrad.curtin.edu.au, {s.soh, m.lazarescu}@curtin.edu.au

Kwan-Wu Chin
School of Electrical, Computer and Telecommunications Engineering
University of Wollongong
Wollongong, NSW, Australia
kwanwu@uow.edu.au

***Abstract*—This paper considers deriving a link schedule for rechargeable WSNs. Unlike past works, it considers: (i) the time required by nodes to harvest energy, and (ii) deterioration in battery lifetime due to memory effects. It presents a greedy heuristic that schedules links according to the earliest time in which the batteries at each link's end nodes are fully discharged. Our results show that considering memory effects via a battery cycle constraint and energy harvesting time increases the link schedule by up to 30.43% and reduces the number of charge/discharge cycles of battery by up to 84.05%. Hence, it helps to increase a battery's lifetime. On the other hand, an increase in energy harvesting time linearly increases link schedules but it does not affect the number of charge/discharge cycles. Finally, increasing a battery's depth of discharge reduces its number of charge/discharge cycles by up to 399.84%, while lengthening the link schedule by up to 7.03% only.**

***Index Terms*—Battery cycle constraint, harvesting time, link schedule, wireless sensor networks.**

I. Introduction

Wireless Sensor Networks (WSNs) form the sensing layer of Internet of Things (IoTs) [1]. They have been used for environmental or habitat monitoring [2], [3], [4]. They are also widely used in various industries such as manufacturing [5]. A well-known issue faced by WSNs is that nodes have limited energy. In many applications, it is impractical to replace the batteries of nodes, especially when there is a large number of sensor nodes and they are deployed in difficult-to-reach locations.

To this end, rechargeable WSNs (rWSNs) are now of great interest because sensor nodes are able to harvest energy from their environment, e.g., sunlight. However, nodes may experience time-varying energy arrivals, meaning when a node exhausts its energy, it will have to spend time harvesting energy before it is able to continue executing tasks. The said energy harvesting time is affected by the type of energy source as well as a node's location [6]. For instance, assume a solar panel has a power density of $15,000$ μW/cm^3 and 20 μW/cm^3 for outdoor and indoor settings, respectively [7]. Hence, a node with a volume of 20 cm^3 will have a corresponding energy harvesting rate of 300 mJ/s (outdoor) and 0.4 mJ/s (indoor). Assuming a Mica2 mote [8], which requires 30 mJ/s of energy to transmit/receive a packet, it will need to harvest for 0.1 s (outdoor) or 75 s (indoor) before it can transmit/receive one packet.

Another important issue of interest recently is the lifetime of rechargeable batteries. Among others, one factor that affects the lifetime of batteries is *memory effects* [9], which decrease their usable capacity if they are charged and discharged repeatedly after a partial discharge and charge, respectively. Another factor is the percentage of discharged energy relative to a battery's overall capacity, which is also called the battery's *Depth of Discharge (DoD)* [10]. Further, frequent battery charge and discharge affect a battery's lifetime [11]. To prolong the lifetime of a battery, we can impose a *battery cycle* constraint, i.e., a node must charge (discharge) its battery completely before fully discharging (charging) its battery again [12].

A fundamental problem in rWSNs is channel access or *link scheduling* [13]. This paper considers nodes that use a Time Division Multiple Access (TDMA) link schedule; this ensures nodes do not experience collisions, which waste energy, and they only need to be active during their allocated time slot. A link scheduler allocates links into slot(s), and links in each slot do not suffer from excessive interference. Ideally, each slot should have a high number of links which ensures a high network capacity. Moreover, a short schedule means a link transmits frequently.

In this paper, we consider a novel research aim: derive a short TDMA link schedule that considers nodes with varying energy harvesting rates and the memory effects of their batteries. We now discuss the problem at hand with the aid of Fig. 1a, 1b, and 1c. Note that links (v_1, v_2), (v_1, v_3), and (v_1, v_4) interfere with each other and thus cannot be

scheduled concurrently. We consider the Harvest-Store-Use battery charging model [6], where the harvested energy in slot t can only be used in slot $t+1$, $t+2$, ... We see that node v_1 needs to wait for three time slots to accumulate one unit of energy; denoted as $v_1|3$, and this energy can only be used after slot 3. First, consider the case where nodes have unlimited battery capacity; denoted by $\sim$ in Fig. 1a. Node v_1 can use its stored energy in time slot $t = 4$. However, none of its links can be scheduled at time 4 because its neighbors have insufficient energy to receive a packet. For example, link (v_1, v_2) can be scheduled no earlier than at slot $6+1 = 7$. After node v_1 transmits a packet to node v_2 at time 7, its remaining energy is sufficient to transmit a packet to node v_3 and v_4. Thus, links (v_1, v_3), and (v_1, v_4) can be scheduled at time slots $t = 8$ and $t = 10$, respectively.

Next, we consider the case where each battery has a capacity of one unit; see Fig. 1b. This means the battery at node v_1 can be recharged only after it is used at time $t = 7$. Thus, node v_1 can transmit the second packet no earlier than at time $t = 7 + 3 + 1 = 11$, i.e., after it has harvested sufficient energy. Further, it can transmit the third packet no earlier than time $11 + 4 = 15$. Consequently, the schedule length is 15.

Lastly, we consider the case where each battery has a *battery cycle* constraint; see Fig. 1c. Node v_1 has a battery with three units of energy and v_2, v_3, v_4 have a battery with two units of capacity. Thus, node v_1 needs to wait until slot $t = 3 \times 3 + 1 = 10$ to fully recharge its battery before it can transmit one packet. However, it cannot do so because its neighbors' battery is yet to be fully recharged. That is node v_2, v_3, and v_4 have to wait until slot $t = 6 \times 2 + 1 = 13$, $t = 15$, and $t = 17$, respectively before their battery can be discharged. The schedule length in this case is 17.

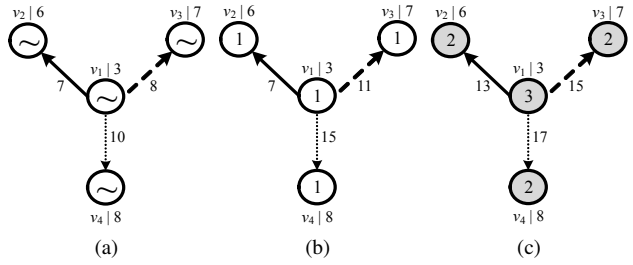


Figure 1. An example (a) with interference and harvesting time, (b) plus battery capacity, and (c) plus battery cycle constraints. The number inside the nodes shows the capacity of its battery, the number next to each link denotes its activation time, and $v_x|z$ denotes node x requires z time slots to accumulate one unit of energy.

This paper contains the following *contributions*. First, we propose a TDMA link scheduler to maximize the network capacity in a rWSN whereby (i) sensor nodes have a different energy harvesting rate, (ii) sensor nodes have finite battery capacity, (iii) each battery at nodes has a *battery cycle* constraint, and (iv) each link i has weight $w_i \geq 1$, meaning it must be scheduled at least w_i times in the derived schedule. To the best of our knowledge, no works have considered all of the above factors when deriving a link schedule. Second, we develop a heuristic technique, whereby its main idea is to schedule links that can be activated at the earliest time when the battery at their end nodes can be discharged to transmit/receive a packet.

The rest of this paper is organized as follows. Section II discusses related works. Section III contains the network model and problem at hand. Our solution is described in Section IV, and its performance evaluation is reported in Section V. Finally, Section VI concludes the paper and provides future research directions.

II. Related Works

Except for reference [14], [15] and [16], there is no research that focuses on link scheduling in rWSNs where nodes require varying amount of time to harvest energy. Sun *et al.* [14] propose two link schedulers that correspond to links with and without a weight. They aim to maximize network throughput and use the Harvest-Store-Use (HSU) [6] battery charging model, where a node must first store before using its harvested energy. Recently, the authors of [15] consider the Harvest-Use-Store (HUS) [17] battery usage protocol, which uses a super capacitor to store harvested energy. This allows a node to use its harvested energy immediately as well as allowing it to store any remaining energy for later use. In a subsequent work, in [16], the authors consider imperfect batteries that leak and have storage inefficiency. Each battery has a recharging time that defines when a node has sufficient energy for one packet data transmission/reception. The authors of [14] consider infinite battery capacity whereas reference [15] and [16] have a limited battery size. These works reported in [14], [15], and [16] consider rechargeable battery with *shallow recharge* [6] which allows *partial* recharge and/or discharge. As reported in [11], shallow charging causes *memory effects* that reduce a battery's lifetime. Therefore, works such as [14], [15], and [16] do not consider memory effects or incorporate a battery cycle constraint to avoid battery degradation, which ultimately reduces the lifetime of a rWSN.

III. Preliminaries

We first describe our rWSN model, introduce key notations before formalizing the problem at hand.

A. Network Model

We model a rWSN as a directed graph $G(V, E)$, where each node $v_i \in V$ is a sensor node and each link $l_{i,j} \in E$ denotes a directional link from v_i to v_j. Each node v_i has a transmission range of $\mathcal{R}_i$. Let $||v_i - v_j||$ be the Euclidean distance between nodes v_i and v_j. A node v_i can transmit or receive the packets to / from v_j if $||v_i - v_j|| \leq \mathcal{R}_i$. Each link $l_{i,j} \in E$ has weight $w_{i,j} \geq 1$, meaning the link must be activated $w_{i,j}$ times in the generated schedule. For instance, in Fig. 2a, $w_{2,4} = 3$. Let ϵ (in Joule) be the energy consumed to transmit or receive one packet. We assume equal energy usage for transmission and reception.

We assume the protocol interference model [18], which considers (i) primary interference, where each node is half-duplex, and (ii) secondary interference, where a node, say A, while receiving a packet from its neighbor, say B, also receives a transmission from node C to D. The interference between links is modeled by a conflict graph $C_G(V', E')$ [19], which can be constructed for a graph $G(V, E)$ as follows: (i) each vertex in V' represents a link in E, i.e., $|V'| = |E|$, and (ii) each edge in E' represents two links of G that experience primary or secondary interference if they are active together. Fig. 2b shows the conflict graph C_G for the example rWSN depicted in Fig. 2a.

A TDMA superframe or a link schedule consists of equal sized time slots. All links in each slot do not experience primary and secondary interference. Let $\mathcal{S}$ represent the superframe and $|\mathcal{S}|$ denote its length (in slots). Each slot is either *empty* or contains one or more non-interfering, concurrently active links. A slot is empty when all sensor nodes experience energy outage.

Nodes use the HSU [6] battery usage protocol. Specifically, a node v_i is equipped with a harvester that scavenges energy from environment, e.g., solar, and a rechargeable battery with a capacity of b_i (in unit ϵ). Let $r_i \geq 1$ (in slots) be the *harvesting time* or the total number of slots that is required by a node v_i to accumulate 1ϵ of energy.

The battery capacity of nodes is assume to be at least the required energy to transmit/receive one packet, i.e., $b_i \geq 1\epsilon$. Each node uses a single battery with a *battery cycle* constraint. This constraint requires the battery of node v_i to be (i) charged to its maximum capacity, $b_{i,max}$, before it can be used, for $1 \leq b_{i,max} \leq b_i$, and (ii) discharged to its minimum capacity, $b_{i,min} \geq 1$, before it can be charged, where $b_{i,min} < b_{i,max}$. Consequently, the battery can be in one of two modes: (i) *charging*, or (ii) *discharging*. Without loss of generality, we assume the battery at node v_i has an initial energy level of $b_{i,min}$. Further, we assume $b_{i,min}$ and $b_{i,max}$ are integers.

We use $\tilde{t}_i^+$ and $\tilde{t}_i^-$ to denote respectively the start and end charging time of node v_i's battery. Similarly, t_i^+ and t_i^- denote respectively the start and end discharging time of node v_i's battery. Further, $\tilde{\tau}_i$ and τ_i respectively are the *charging time interval* and *discharging time interval* at node v_i, which is computed as $\tilde{\tau}_i = \tilde{t}_i^- - \tilde{t}_i^+$ and $\tau_i = t_i^- - t_i^+$. In other words, the battery at node v_i is being charged during time interval $\tilde{\tau}_i$ and being discharged during time interval τ_i. Each battery follows a sequence of charge-discharge cycle. Thus, we have $t_i^+ = \tilde{t}_i^- + 1$ and $\tilde{t}_i^+ = t_i^- + 1$. Note that, for each charge/discharge cycle, the value of $\tilde{\tau}_i$ is dependent on r_i and $b_{i,max}$, while the length of τ_i is affected by $b_{i,min}$ and the number of times the battery is used to transmit / receive packets at the cycle. We note that time interval $\tilde{\tau}_i$ has equal length in any cycle because r_i and $b_{i,max}$ are constants over all cycles. In contrast, the value of τ_i may vary at different cycles due to different energy usage between cycles. We use $\tilde{b}_{i,t}$ and $b_{i,t}$ (in unit ϵ) to denote the amount of energy that the battery at node v_i have at the beginning of slot t, for charge and discharge cycle, respectively. Thus, we have $\tilde{b}_{i,\tilde{t}_i^-} = b_{i,max}$ and $b_{i,t_i^-} = b_{i,min}$. The battery level of node v_i at the beginning of each *charging* cycle is given as $\tilde{b}_{i,\tilde{t}_i^+} = b_{i,min}$. On the other hand, the battery level of node v_i at the beginning of each *discharging* cycle is $b_{i,t_i^+} = b_{i,max}$.

Let T_i be the earliest time slot when the battery at node v_i is in discharging mode. The earliest time in which link $l_{i,j}$ can be scheduled is at time $t_{i,j} = \mathbf{max}(T_i, T_j)$, which corresponds to the discharging time of the battery at the end nodes of link $l_{i,j}$. For each node v_i, we initialize $T_i = \tilde{\tau}_i + 1$. Note that, the battery is initially in charging mode at the beginning of time slot zero and it takes $\tilde{\tau}_i$ slots to reach level $b_{i,max}$. Further, following the HSU protocol, the energy can be used only one slot later. It is updated when the battery at node v_i is discharged to transmit/receive a packet.

As an example of our network model, Fig. 2a shows the value of $b_{i,min}$, $b_{i,max}$, b_i, and r_i for each node v_i as well as the weight of each link $l_{i,j}$. The battery level of node v_1 in charging and discharging mode are $b_{1,t_1^+} = b_{1,max} = 5$ and $\tilde{b}_{1,\tilde{t}_1^+} = b_{1,min} = 1$, respectively. As shown later in Section IV, one can compute the charging time interval for each battery to obtain $\tilde{\tau}_1 = 8$, $\tilde{\tau}_2 = 10$, $\tilde{\tau}_3 = 16$ and $\tilde{\tau}_4 = 9$. Thus, we have $T_1 = \tilde{\tau}_1 + 1 = 9$, $T_2 = 11$, $T_3 = 17$, and $T_4 = 10$. Therefore the earliest time each link $l_{i,j}$ can be scheduled is computed as $t_{1,2} = \mathbf{max}(9, 11) = 11$, $t_{2,4} = 11$, $t_{3,1} = 17$.

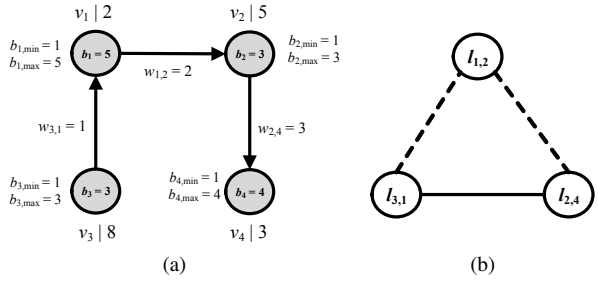


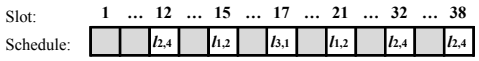
Figure 2. A rWSN model. (a) Graph G, and its (b) conflict graph C_G. Also shown are primary (dashed lines) and secondary (solid line) interference.

B. Problem Statement

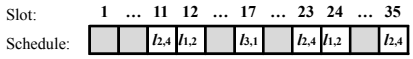
Our problem is to generate the TDMA link schedule $\mathcal{S}$ with the minimum length $|\mathcal{S}|$ for a rWSN such that (i) the battery at each node $v_i \in V$ satisfies the *battery cycle* constraint, (ii) each link $l_{ij} \in E$ can be scheduled at time slot t only if its end nodes are in discharging mode, and (iii) each link $l_{i,j} \in E$ is scheduled at least $w_{i,j}$ times in $\mathcal{S}$.

To illustrate the effect of link scheduling on $|\mathcal{S}|$, consider the example in Fig. 2. Fig. 3a shows one *feasible* link schedule with 38 slots. A schedule is *feasible* if it satisfies constraints (i), (ii), and (iii). The figure shows only non empty slots, i.e., each empty slot is represented as "...". Figure 3b shows a shorter *feasible* schedule of $|\mathcal{S}| = 35$ slots. Our problem aims to generate the shortest *feasible* schedule $\mathcal{S}$. Note that, link scheduling in wireless networks

in general is known to be NP-hard [20], where nodes do not have energy constraint. In our case, observe that a special case of our problem is where nodes have ample energy. Consequently, our problem remains NP-hard.



(a) One feasible TDMA link schedule.



(b) The optimal TDMA link schedule.

Figure 3. TDMA link schedules for the rWSN in Fig. 2 where nodes have battery cycle constraint. Gray colored slots show no transmissions/receptions.

IV. Solution

This section first describes three propositions relied upon by our greedy algorithm. Proposition 1 and 2 concern a battery in charging mode, while Proposition 3 is for a battery in discharging mode.

A. Key Properties

Proposition 1. *The energy level of a battery in charging mode at node v_i at time slot t, for $\tilde{t}_i^+ \leq t \leq \tilde{t}_i^-$, is $\tilde{b}_{i,t} =$ **min**$(b_{i,max}, b_{i,min} + (t - \tilde{t}_i^+)/r_i)$.*

Proof. The maximum amount of energy that can be harvested from $\tilde{t}_i^+$ to t is $\frac{t-\tilde{t}_i^+}{r_i}$. However, $\tilde{b}_{i,t}$ is bounded by the upper limit of battery capacity $b_{i,max}$, which implies $\tilde{b}_{i,t} \leq b_{i,max}$. □

Proposition 2 computes the number of slots required by node v_i to charge its battery from $b_{i,min}$ to $b_{i,max}$.

Proposition 2. *The charging time interval for the battery of node v_i is computed as $\tilde{\tau}_i = r_i(b_{i,max} - b_{i,min})$.*

Proof. We set $t = \tilde{t}_i^-$ in Proposition 1 to generate the maximum energy level of the battery, i.e., $b_{i,max}$. We have $b_{i,max} = b_{i,min} + \tilde{\tau}_i/r_i$. Thus, $\tilde{\tau}_i = r_i(b_{i,max} - b_{i,min})$ as shown in Proposition. □

Proposition 3 computes the next earliest time in which the battery at node v_i is ready for discharging.

Proposition 3. *Let $t \geq \tilde{\tau}_i + 1$ be the time in which node v_i last used its battery. The next earliest time slot before it can use its battery again to transmit/receive a packet is $T_i = t + \sigma_{i,t} \times \tilde{\tau}_i + 1$.*

Proof. The next earliest time T_i depends on the remaining battery level at node v_i at time t, i.e., $b_{i,t}$. We consider two cases: (i) $b_{i,t} = b_{i,min}$, and (ii) $b_{i,t} > b_{i,min}$. For case (i), the battery needs to be recharged and $\sigma_{i,t}$ is set to one. Then, we use Proposition 2 to compute the charging time interval, $\tilde{\tau}_i$. Thus, $T_i = t + \tilde{\tau}_i + 1$ because the harvested energy needs to be stored first before it can be used.

For case (ii), the battery at node v_i still can be discharged to transmit/receive another packet at time t and $\sigma_{i,t}$ is set to 0. However, as a node is half-duplex, the next earliest time the battery at node v_i can be discharged to transmit/receive a packet is in the next slot, i.e., $T_i = t + 1$. □

B. Algorithm

We are now ready to explain LSBCC. The algorithm aims to schedule all non-interfering links at the earliest possible time slot when the battery at its end nodes can be used, i.e., in discharging mode. Starting at $t = 0$, the battery of each node is initially in charging mode.

In *Lines 1-3*, LSBCC calls *INIT*(.) to initialize the following four parameters for each node v_i: (i) it initializes the energy level of battery of node v_i to $b_{i,min}$, i.e., $\tilde{b}_{i,0} = b_{i,min}$, (ii) it uses Proposition 2 to compute $\tilde{\tau}_i$, i.e., the charging time interval for the battery at node v_i, (iii) it sets T_i to $\tilde{\tau}_i + 1$, recall that T_i is the earliest time when the battery at node v_i can be discharged to transmit or receive one packet, and (iv) it initializes b_{i,T_i}, the energy level of battery at node v_i at time T_i, to $b_{i,max}$.

Lines 4-6 compute $t_{i,j}$, i.e., the earliest time link (i, j) can be scheduled. *Line 7* creates a set K that saves links (i, j) that have the earliest activation time. *Line 8* then uses function *ORDER*(K) to sort links in set K in order of decreasing weight $w_{i,j}$. Links with equal $w_{i,j}$ are sorted in decreasing node degree of its end nodes and for a tie, links are sorted in increasing order of their node labels. *Line 9* sets t with the earliest slot, i.e., **min**$\{t_{i,j}\}$. *Lines 10-32* repeatedly schedule each link $l_{i,j} \in K'$ in order. Each selected link in *Line 12* does not cause interference or is interfered by links that have been scheduled in slot t; see the condition in *Line 11*. Each slot in $\mathcal{S}$ is initially empty. Note that, function *CONFLICT()* uses a matrix M of size $|E|^2$ that contains Boolean variables to represent the conflict graph of the network; i.e., $M[a, b]$ is set to one if there is an interference between links a and b. *Line 13* decreases the weight of each selected link $l_{i,j}$ by one. Once the link weight is equal to zero, *Line 15* removes the link from contention; see *Lines 14-16*.

Further, *Lines 17-18* compute the energy level of the battery at node v_i and v_j at time t after one packet transmission/reception. *Line 19* updates the next earliest time the battery at node v_i can be discharged, while *Lines 20-24* recompute its energy level depending on the remaining energy level at the current time t. LSBCC computes the next earliest time and energy level of the battery at node v_j in *Lines 25-30*. Finally, the steps from *Line 4* is repeated until all links have weight $w_{i,j} = 0$.

As an example, consider the rWSN and conflict graph C_G shown in Fig. 2. *Lines 1-3* of LSBCC use the function *INIT*(.) to set $\tilde{b}_{1,0} = \tilde{b}_{2,0} = \tilde{b}_{3,0} = \tilde{b}_{4,0} = 1$. The function applies Proposition 2 to compute $\tilde{\tau}_1 = 8$, $\tilde{\tau}_2 = 10$, $\tilde{\tau}_3 = 16$, and $\tilde{\tau}_4 = 9$. It sets $T_1 = 9$, $T_2 = 11$, $T_3 = 17$, and $T_4 = 10$. Finally, it initializes $b_{1,T_1} = 5$, $b_{2,T_2} = 3$, $b_{3,T_3} = 3$, and $b_{4,T_4} = 4$. *Lines 4-6* compute $t_{1,2} =$ **max**$(T_1, T_2) = 11$, $t_{2,4} = 11$, and $t_{3,1} = 17$. *Line 7* inserts links $l_{1,2}$ and $l_{2,4}$ into the set K. *Line 8* obtains $K' = \{l_{2,4}, l_{1,2}\}$ because

$w_{2,4} > w_{1,2}$, while *Line 9* sets $t = 11$. Further, *Line 12* inserts $l_{2,4}$ into $\mathcal{S}[11]$. *Line 13* decreases $w_{2,4}$ by one and hence $w_{2,4} = 2$. *Lines 17-18* compute $b_{2,11} = 2$ and $b_{4,11} = 3$. *Lines 19-24* set $T_2 = 12$ and $b_{2,T_2} = 2$, while *Lines 25-30* obtain $T_4 = 12$ and $b_{4,T_4} = 3$. *Line 33* repeats the steps from *Line 4* until all links are scheduled. Finally, LSBCC produces the link schedule $\mathcal{S}$ in Fig. 3b, i.e., $\mathcal{S} = [\ \mathcal{S}[11] = \{l_{2,4}\},\ \mathcal{S}[12] = \{l_{1,2}\},\ \mathcal{S}[17] = \{l_{3,1}\},\ \mathcal{S}[23] = \{l_{2,4}\},\ \mathcal{S}[24] = \{l_{1,2}\},\ \mathcal{S}[35] = \{l_{2,4}\}\ ]$. The generated schedule contains 29 empty slots as the battery of each node needs time to charge to its maximum level before it can be used to transmit/receive packets.

Proposition 4. *The time complexity of LSBCC is* $O(W|E|^2)$*, where*

$$W = \sum_{(i,j)\in|E|} (w_{i,j}). \tag{1}$$

Proof. Lines 1-3 take $O(|V|)$ and *Lines 4-6* require $O(|E|)$. *Line 7* takes $O(|E|)$. *Line 8* sorts all links in K using the function *ORDER*(K) that requires $O(|E|\ log\ |E|)$. *Line 9* takes $O(1)$. *Line 11* requires $O(|E|^2)$ to construct a matrix M which represents the conflict graph C_G. Function *CONFLICT*$(l_{i,j}, \mathcal{S}[t])$ in *Line 11* uses the matrix at most $|E|$ times. Hence, it takes $O(|E|)$. *Lines 12-30* take $O(1)$ each. The for loop in *Lines 10-32* is repeated at most $|E|$ times, and thus, the loop requires at most $O(|E|^2)$. *Line 33* repeats *Lines 4-32* W times. Thus, the time complexity of LSBCC is $O(W|E|^2)$. □

V. Evaluation

We have implemented LSBCC in C++ and conducted our experiments on a computer with an Intel Core i7 CPU @ 3.4 GHz and 16 GB of RAM. We consider arbitrary networks with 10 to 50 nodes randomly deployed on a 40×40 m^2 area. Each node has a transmit and interference range of 15 and 30 meters, respectively. Our results are an average over 100 random node deployments. The average number of links $|E|$ is 28, 125, 273, 470, and 758 for 10, 20, 30, 40, and 50 nodes respectively. Note, as noted in Section II our problem is new. Consequently, we did not compare our algorithm against any other solutions because no prior works have used our battery cycle constraint model.

A. LSBCC vs LSNBC

This section compares the performance of LSBCC against LSNBC in terms of superframe length $|\mathcal{S}|$ and the total number of charge/discharge cycles. Briefly, LSNBC is similar to LSBCC but without the battery cycle constraint. For this experiment, we consider rWSNs with 10 to 50 nodes with harvesting time $r_i = 5$. We set the battery capacity b_i to 3ϵ of energy, its minimum energy level $b_{i,min}$ to 1ϵ, and maximum energy level $b_{i,max}$ to 3ϵ. Each link weight is randomly fixed to a value between 1 and 5, i.e., $w_{i,j} = [1, 5]$.

Algorithm 1 LSBCC

Input: $G(V, E)$, r_i, b_i, $b_{i,max}$, $b_{i,min}$ of each node $v_i \in V$, weight $w_{i,j}$ of each link $l_{i,j} \in E$, and conflict graph C_G
Output: Superframe $\mathcal{S}$

1: **for** each node $v_i \in V$ **do**
2: *INIT*$(\tilde{b}_{i,0}, \tilde{\tau}_i, T_i, b_{i,T_i})$
3: **end for**
4: **for** each link $l_{i,j} \in E$ **do**
5: $t_{i,j}$ = **max**(T_i, T_j)
6: **end for**
7: K = {node $l_{i,j}$ in C_G with **min**$\{t_{i,j}\}$}
8: K' = *ORDER*(K)
9: $t \leftarrow$ **min**$\{t_{i,j}\}$
10: **for** each $l_{i,j} \in K'$ **do**
11: **if** NOT *CONFLICT*$(l_{i,j}, \mathcal{S}[t])$ **then**
12: $\mathcal{S}[t] \leftarrow \mathcal{S}[t] \cup l_{i,j}$
13: $w_{i,j} \leftarrow w_{i,j} - 1$
14: **if** $w_{i,j}$ = 0 **then**
15: remove node $l_{i,j}$ from C_G
16: **end if**
17: $b_{i,t} \leftarrow b_{i,T_i} - 1$
18: $b_{j,t} \leftarrow b_{j,T_j} - 1$
19: $T_i \leftarrow t + 1$
20: **if** $b_{i,t} = b_{i,min}$ **then**
21: $T_i \leftarrow T_i + \tilde{\tau}_i$
22: $b_{i,T_i} \leftarrow b_{i,max}$
23: **else** $b_{i,T_i} \leftarrow b_{i,t}$
24: **end if**
25: $T_j \leftarrow t + 1$
26: **if** $b_{j,t} = b_{j,min}$ **then**
27: $T_j \leftarrow T_j + \tilde{\tau}_j$
28: $b_{j,T_j} \leftarrow b_{j,max}$
29: **else** $b_{j,T_j} \leftarrow b_{j,t}$
30: **end if**
31: **end if**
32: **end for**
33: **repeat** Line 4-32 **until** all $w_{i,j}$ = 0

LSBCC vs LSNBC on Superframe Length: Fig. 4 shows that the superframe length $|\mathcal{S}|$ produced by LSBCC is longer than LSNBC. As an example, in a rWSN with 10 nodes, LSBCC produces 42 more slots (30.43% longer) as compared to when using LSNBC. The results are consistent for other networks, i.e., $|V| = 20, 30, 40, 50$ nodes. LSBCC produces superframes that are 19.4%, 17.9%, 17.94%, and 17.47% longer than in LSNBC, respectively. The reason is because LSBCC needs to wait for each battery to be charged to its maximum level before it can be discharged.

LSBCC vs LSNBC on Charge/Discharge Cycles: As shown in Fig. 5, the number of charge/discharge cycles produced by LSBCC is less than LSNBC. For example, when $|V| = 10$, LSBCC has 64 cycles less than in LSNBC; i.e., 77.11% less. The results are consistent for $|V| = 20, 30, 40, 50$. More specifically, LSBCC requires 78.86%, 81.03%, 82.62%, and 84.05% fewer cycles than LSNBC,

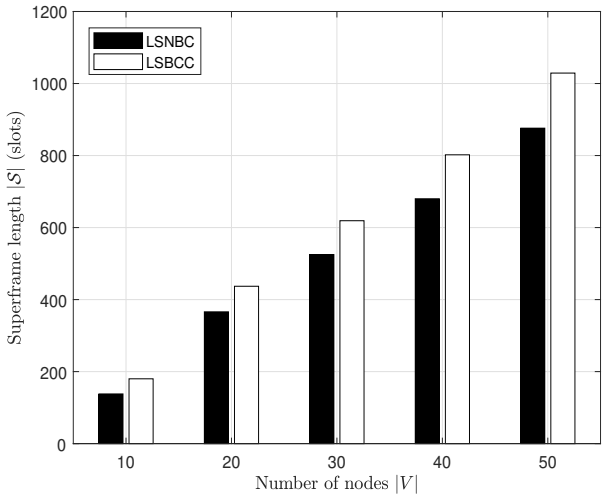


Figure 4. LSBCC vs LSNBC on $|\mathcal{S}|$.

respectively. The reason is because each battery in LSBCC needs to be charged only when its energy level reaches the minimum. Notice that, there is a tradeoff between longer link schedules and lesser charge/discharge cycles. For example, for rWSN with $|V| = 30$, the number of cycles is reduced by 81.03% in the expense of 17.9% longer superframe length. In the remaining experiments, we only consider LSBCC.

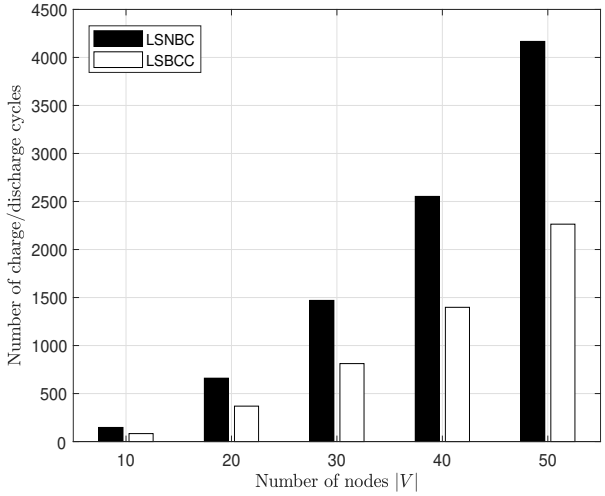


Figure 5. LSBCC vs LSNBC on charge/discharge cycles.

B. Effect of Harvesting Time

This section investigates the effect of harvesting time r_i on $|\mathcal{S}|$ and charge/discharge cycles. In this simulation, we consider various r_i values, namely $1, 5, 10, 15, 20$ slots in a rWSN with 10 to 50 nodes. We set $b_i = 3\epsilon$, $b_{i,min} = 1\epsilon$, $b_{i,max} = 3\epsilon$, and each link has weight $w_{i,j} = 3$.

Effect on Superframe Length: From Fig. 6, we see that energy harvesting time has a significant effect on $|\mathcal{S}|$, i.e., increasing the harvesting time of nodes results in a longer superframe. Specifically, for a rWSN with 10 nodes, when r_i is increased by four slots, i.e., from one to five, $|\mathcal{S}|$ jumps from 68 to 173 slots - an increase of 1.54 times. Similarly, when r_i increases from 5 to 20 with an interval of five, i.e., from five to 10, 10 to 15, and 15 to 20, $|\mathcal{S}|$ is further increased by 139, 140, and 140 slots, meaning the link schedule increases by 0.8, 0.45, and 0.31 times, respectively. We observe similar trends in rWSN with 20, 30, 40, and 50 nodes. For example, for a rWSN with $50(100)$ nodes, when r_i increases from 1 to 20 with an interval of five, $|\mathcal{S}|$ is increased by $1.33(1.34)$, $0.72(0.72)$, $0.42(0.42)$, $0.3(0.3)$ times respectively. The increase in $|\mathcal{S}|$ is because each battery needs more time to be charged to its maximum level before it can be used to transmit/receive a packet. Also notice that the $|\mathcal{S}|$ for each network size increases almost linearly when r_i is increased from one to 20. Further, the rate of increase (in slots) in smaller networks, e.g., $|V| = 10$, is less than that of larger networks, e.g., $|V| = 50$. The reason is because more nodes mean more links need to be scheduled. Also, more links will have to wait for sufficient energy before they can be activated.

Effect on Charge/Discharge Cycles: In our experiment, for $|V| = 10, 20, 30, 40, 50$, LSBCC produces the number of charge/discharge cycles of 86, 377, 821, 1412, and 2276, respectively for all values of r_i. The results show that increasing r_i does not affect the number of battery charge/discharge cycles. The reason is because charge/discharge cycles depend on the energy level of battery, not on r_i. Hence, the varying value of r_i does not impact on the cycles.

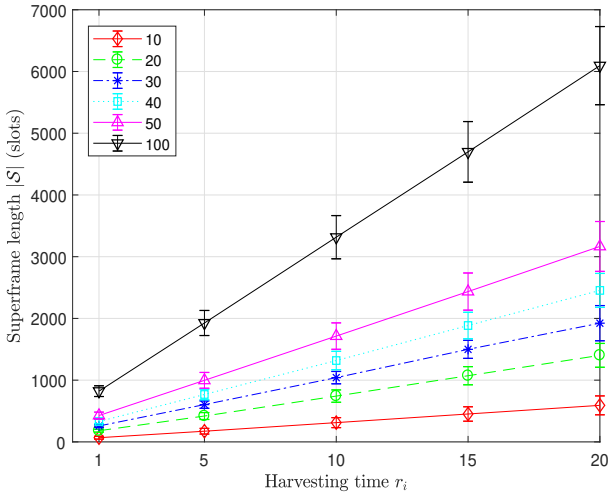


Figure 6. Effect of harvesting time r_i on $|\mathcal{S}|$.

C. Effect of Battery's DoD

This section studies the effect of Depth of Discharge (DoD) on $|\mathcal{S}|$ and the number of charge/discharge cycles. Recall that DoD corresponds to the percentage of battery

capacity that has been discharged. This experiment considers the battery of each node v_i with equal capacity of $b_i = 20\epsilon$, and DoD of 5%, 25%, 50%, 75% and 100%; thus, the usable energy for each discharge cycle is 1ϵ (5% of 20ϵ), 5ϵ, 10ϵ, 15ϵ, and 20ϵ, respectively. For the five DoD values, LSBCC fixes the value of $b_{i,max}$ to 20, and uses five different values of $b_{i,min}$, i.e., 19, 15, 10, 5, 0, for DoD of 5%, 25%, 50%, 75% and 100%, respectively. We set $r_i = 5$ and $w_{i,j} = 10$ in a rWSN with 10, 20, 30, 40, and 50 nodes.

Effect on Superframe Length: Fig. 7 shows that increasing DoD has an insignificant effect on $|\mathcal{S}|$. As an example, when $|V| = 10$ and $|V| = 20$, increasing DoD from 5% to 25% enlarges $|\mathcal{S}|$ by 6.58% (547 to 583) and 0.57% (1394 to 1402), respectively. Similarly, for $|V| = 30, 40, 50$, there is only a decrease in $|\mathcal{S}|$ of 0.39% (2033 to 2025), 0.12% (2583 to 2580), and 0.12% (3338 to 3334) in $|\mathcal{S}|$, respectively. Further, when DoD increased from 25% to 100%, the superframe length $|\mathcal{S}|$ increases only by 7.03% (583 to 624), 4.21% (1402 to 1461), 1.14% (2025 to 2048), 0.12% (2580 to 2583), and 0.27% (3334 to 3343) for $|V| = 10, 20, 30, 40, 50$, respectively. The reason is because battery's DoD corresponds to the battery's usable capacity, e.g., a battery with maximum capacity of 20 and DoD of 30% has usable energy of six, while the same battery with DoD of 80% has a larger usable energy of 16. Thus, increasing a battery's DoD is equivalent to increasing the battery's usable energy or its capacity. As reported in [15], battery capacity has an insignificant effect on $|\mathcal{S}|$.

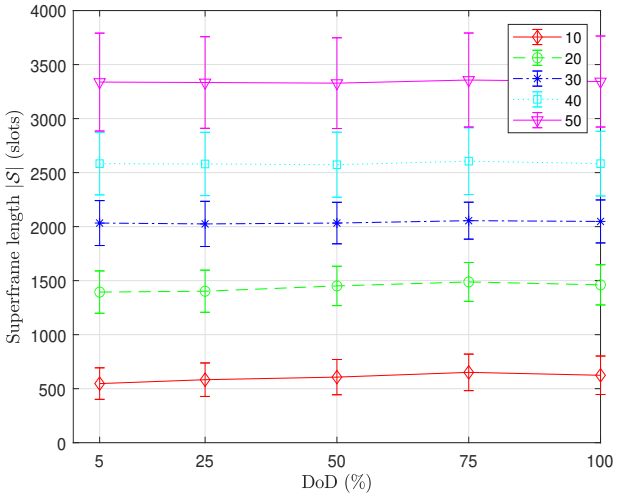


Figure 7. Effect of battery's DoD on $|\mathcal{S}|$.

Effect on Charge/Discharge Cycles: As shown in Fig. 7, increasing DoD decreases the number of cycles. For example, when DoD increases from 5% to 25% in a rWSN with 10 nodes, the number of charge/discharge cycles decreased by 394.78% (from 569 to 115). Similarly, for rWSNs with $|V| = 20, 30, 40, 50$ nodes, there are 399% (2505 to 502), 399.45% (5469 to 1095), 399.73% (9405 to 1882), and 399.84% (15165 to 3034) decreases in the number cycles, respectively. In addition, increasing DoD from 25% to 100% reduces the number of charge/discharge cycles from 115 to 29 (a decrease of 74.78%), 502 to 126 (74.9%), 1095 to 274 (74.98%), 1882 to 471 (74.97%), and 3034 to 759 (74.98%) for $|V| = 10, 20, 30, 40, 50$, respectively. The decrease in the number of charge/discharge cycles when DoD increases is because the battery of each node has more usable energy. Thus, larger DoD values are preferable because the capacity of a battery tends to drop if it has a large number of charge/discharge cycles [21].

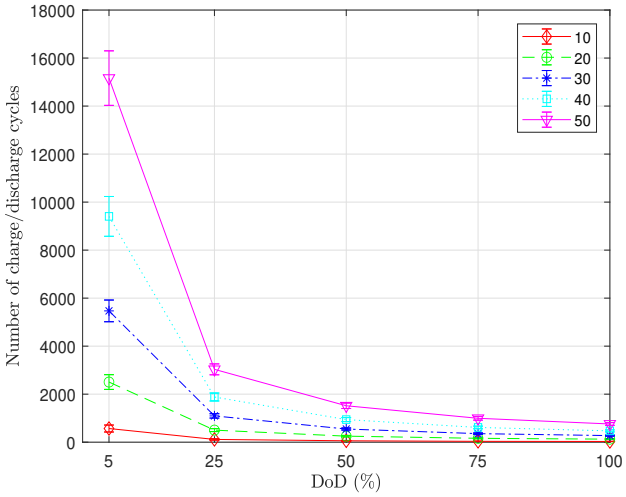


Figure 8. Effect of battery's DoD on charge/discharge cycles.

VI. Conclusion

This paper has considered link scheduling in rWSNs where the battery of each sensor node uses the HSU recharging model. Unlike past works, it considers the memory effects that degrade the lifetime of a node's battery. To this end, it proposes an algorithm called LSBCC that considers a battery constraint when scheduling links. Our simulations show that: (i) the battery cycle constraint increases superframe length $|\mathcal{S}|$ by up to 30.43% and reduces the number of battery charge/discharge cycles by up to 84.05% as compared to without the cycle constraint, (ii) increasing harvesting time linearly increases $|\mathcal{S}|$, but it does not affect the number of charge/discharge cycles, (iii) increasing a battery's DoD extends $|\mathcal{S}|$ only by up to 7.03%, and reduces the number of charge/discharge cycles by up to 399.84%. An immediate future work is to consider battery leakage. Another research direction is to consider a dual alternate battery system.

Acknowledgment

The first author acknowledges support from Indonesia Lecturer Scholarship (BUDI) by Indonesia Endowment Fund for Education (LPDP), Ministry of Finance, Republic

of Indonesia.

REFERENCES

[1] J. A. Manrique, J. S. Rueda-Rueda, and J. M. Portocarrero, "Contrasting internet of things and wireless sensor network from a conceptual overview," in *iThings and GreenCom and CPSCom and SmartData*, Chengdu, China, Dec. 2016, pp. 252–257.

[2] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proc. 1st ACM Int. Workshop Wireless Sensor Netw. Applicat. (WSNA)*, Atlanta, Georgia, USA, Sep. 2002, pp. 88–97.

[3] A. Boubrima, W. Bechkit, and H. Rivano, "Optimal wsn deployment models for air pollution monitoring," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 2723–2735, May 2017.

[4] K. S. Adu-Manu, C. Tapparello, W. Heinzelmanm, F. A. Katsriku, and J.-D. Abdulai, "Water quality monitoring using wireless sensor networks: Current trends and future research directions," *ACM Trans. Sensor Netw.*, vol. 13, no. 1, pp. 4:1–4:41, Feb. 2017.

[5] V. J. Hodge, S. O'Keefee, M. Weeks, and A. Moulds, "Wireless sensor networks for condition monitoring in the railway industry: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1088–1106, Jun. 2015.

[6] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: survey and implications," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 3, pp. 443–461, 2011.

[7] H. Kim, Y. Tadesse, and S. Priya, *Piezoelectric Energy Harvesting, in Energy Harvesting Technologies*, S. Priya and D. J. Inman, Eds. New York: Springer, 2009.

[8] [Online]. Available: https://www.eol.ucar.edu/isf/facilities/isa/internal/CrossBow/DataSheets/mica2.pdf

[9] R. A. Huggins, "Mechanism of the memory effect in nickel electrodes," *Solid State Ionics*, vol. 177, no. 26, pp. 2643–2646, 2006.

[10] M. I. Hlal, V. K. Ramachandaramurthy, A. Sarhan, A. Pouryekta, and U. Subramaniam, "Optimum battery depth of discharge for off-grid solar pv/battery system," *Journal of Energy Storage*, vol. 26, no. 100999, 2019.

[11] J. Yang and J. Wu, "Optimal transmission for energy harvesting nodes under battery size and usage constraints," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Aachen, Germany, Jun. 2017, pp. 819–823.

[12] G. M. Siddesh, G. C. Deka, K. G. Srinivasa, and L. M. Patnaik, *Cyber Physical Systems- A Computational Perspective*. Chapman & Hall /CRC, 2015.

[13] S. Gandham, M. Dawande, and R. Prakash, "Link scheduling in wireless sensor networks: Distributed edge-coloring revisited," *J. Parallel Distrib. Comput.*, vol. 68, no. 8, pp. 1122–1134, 2008.

[14] G. Sun, G. Qiao, and L. Zhao, "Efficient link scheduling for rechargeable wireless ad hoc and sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 1, no. 1, pp. 1–14, Dec. 2013.

[15] Tony, S. Soh, K.-W. Chin, and M. Lazarescu, "Link scheduling in rechargeable wireless sensor networks with harvesting time and battery capacity," in *Proc. IEEE 43rd Conf. Local Computer Networks (LCN)*, Chicago, IL, USA, Oct. 2018, pp. 235–242.

[16] ——, "Link scheduling in rechargeable wireless sensor networks with imperfect batteries," *IEEE Access*, vol. 7, pp. 104 721–104 736, 2019.

[17] M.-L. Ku, W. Li, Y. Chen, and K. J. R. Liu, "Advances in energy harvesting communications: Past, present, and future challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1384–1412, 2016.

[18] S. Ramanathan, "A unified framework and algorithm for channel assignment in wireless networks," *Wireless Netw.*, vol. 5, no. 2, pp. 81–94, Mar. 1999.

[19] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact on interference on multi-hop wireless network performance," in *ACM MobiCom*, San Diego, CA, USA, Sep. 2003, pp. 66–80.

[20] B. Hajek and G. Sasaki, "Link scheduling in polynomial time," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 910–917, 1988.

[21] [Online]. Available: https://batteryuniversity.com/learn/article/how_to_prolong_lithium_based_batteries