IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 27, 2020, accepted August 1, 2020, date of publication August 6, 2020, date of current version August 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3014645

Energy-Saving Link Scheduling in Energy Harvesting Wireless Multihop Networks With the Non-Ideal Battery

SIQI GUAN[1], JIANHUI ZHANG[1], (Member, IEEE), ZHENGBING SONG[1], BEI ZHAO[1], AND YANJUN LI[2], (Member, IEEE)

[1]Key Laboratory of Brain Machine Collaborative Intelligence of Zhejiang Province, College of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China

[2]School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China

Corresponding author: Bei Zhao (zhaobei@hdu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61473109, Grant 61772472, and Grant 61877015; and in part by the Fundamental Research Funds for the Provincial Universities of Zhejiang under Grant RF-A2019002.

ABSTRACT Emerging energy harvesting technology can harvest and convert ambient energy into electrical power. It is a quite effective way to extend the lifetime of Wireless Multihop Networks (WMNs), and has been widely applied in WMNs. Meanwhile, the interference-free link scheduling is an indispensable operation in WMNs to reduce energy waste caused by link collision. In the energy harvesting WMNs with link scheduling, a few works take into account the impact of non-ideal battery on energy efficiency and network lifetime. This article studies the harvested energy-saving link scheduling problem in the energy harvesting WMNs under the imperfect storage efficiency. We establish the energy model to index the link weight with the harvested energy consumed on it. Furthermore, we propose a Centralized Scheduling (CS) algorithm, a Centralized Scheduling with Minimum interference weights (CSM) algorithm and a distributed one to activate the links with the maximum weights so as to maximize the utilization of the harvested energy and to minimize the energy stored in the battery. Finally, this article conducts the simulation experiment to verify the effectiveness of our algorithms and the correctness of our theoretical analysis.

INDEX TERMS Energy harvesting, link scheduling, storage efficiency, TDMA, wireless multihop networks.

I. INTRODUCTION

Wireless Multihop Networks (WMNs) are widely used in the communication field and greatly increase network capacity due to the advantages of instant communication, easy deployment, and powerful self-organization capabilities [1]. They have many specific applications, such as wireless sensor networks, edge networks and edge computing. However, energy is a key factor restricting the network lifetime [2]. Especially with the development of 5G technology, high-data-rate multimedia wireless services will cause battery-powered devices to consume more energy to meet the needs of various transmission services [3]. An effective technique to keep networks working for sustainable time is energy harvesting. In general, the nodes harvest ambient energy (*e.g.* solar [4], wind [5], thermal energy [6], *etc.*) from the surroundings and convert it into electrical power. But the energy distribution of nodes is uneven due to the difference of node deployment locations and environmental influences, which severely limits the throughput of the network. Another technology SWIPT was proposed to solve the uneven energy distribution in the environment [7], [8]. In view of the fact that radio frequency signals can carry energy, SWIPT provides energy to surrounding nodes while transmitting information. However, maximizing the worst-case signal-to-interference-and-noise ratio constrained by harvested energy constraint under the imperfect channel state information is still worth studying [9]. Therefore, when operating WMNs, a key concern is to manage the energy harvested by nodes to maximize the network efficiency.

Theoretically, WMNs may operate permanently with extra energy supply from ambient energy [10]. But in many cases, the harvested energy is often limited and unevenly distributed. For example, solar energy is sufficient when the weather is

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

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

clear, but it is insufficient in cloudy days. Once the ambient energy is not abundant, the network performance will be greatly reduced. The same problem also exists in SWIPT due to the loss of the signal when it transmits. To extend network lifetime or even to support permanent network operation, this article further considers saving energy on the basis of energy harvesting.

Typically, the nodes harvest energy and store it in their energy storage units such as batteries. Nevertheless, the battery has the imperfect storage efficiency λ ($0 < \lambda \leq 1$), which brings new challenges in the effective utilization of the harvested energy [11]. For instance, the round-trip energy storage efficiency of the lithium-ion battery is about 90%-95% [12], and the sodium-sulfur battery has about 89% energy storage efficiency [13]. In [14], the chemical limitations of the NiMH battery can even cause the storage efficiency to be as low as 66%. In recent years, supercapacitors have been widely used in energy storage systems because of their high power density, wide operating temperature range, and no environmental pollution [15]. However, compared with traditional rechargeable batteries, supercapacitors have lower energy density and higher self-discharge rate, which means that they store less energy and leak more energy [16]. This is detrimental to energy-neutral operation in WMNs. In addition, the charging efficiency is determined by the energy conserved by supercapacitors and transmitted by the charging circuit, so the supercapacitors also have imperfect charging efficiency [17]. The energy conversion efficiency of energy receiver in SWIPT is also imperfect [18]–[20]. To reduce the energy waste caused by the low storage efficiency, node prefers to use the harvested energy directly rather than storing it in battery [21]–[25].

In the energy harvesting WMNs, the prerequisite for successful data transmission is to schedule links without interference. A popular approach for interference free transmission is to adopt the TDMA MAC protocol, which can directly support low duty cycle operation [26]. To be specific, the TDMA protocol divides one network period into several time slots with constant size, and activates interference-free links simultaneously in each time slot to avoid energy waste caused by link collision. However, the time slot to schedule link activation is not always available for energy harvesting. The stored energy must be discharged to support link activation so the energy is indirectly lost because of the imperfect storage efficiency. It brings challenges to link scheduling in the energy harvesting networks.

As far as we know, very few efforts have fully considered the limited battery storage efficiency and interference-free link scheduling. Only the work in [23] has jointly considered the two factors. Since each node had a different energy harvesting time, they derived a formula for calculating the earliest time slot that each link could activate by introducing the battery storage efficiency and the leakage rate. Nevertheless, their goal is to schedule all links in the least time slots instead of saving energy. Furthermore, once the energy harvesting time of each node is determined, it will not be changed in the subsequent time slots, which is idealistic compared to the real situation. Because in the actual situation, the uneven energy distribution of node is not only caused by differences in deployment locations, but also caused by time-varying environmental factors, so the energy harvesting time of node should change with time.

In this article, we propose the energy model including both the battery storage efficiency limitation and interference-free link scheduling. In each time slot, each node harvests energy randomly. We introduce a novel link scheduling problem called the harvested energy-saving link scheduling problem to reduce energy waste. In our scheduling, each link is assigned with a weight representing the amount of consumed energy when it activates. The more energy consumed, the greater the weight is. Due to the impact of the battery storage efficiency on energy waste in link scheduling, this article always gives the highest priority to activate the links with the maximum weights so as to minimize the amount of energy to be stored in the battery. When a link activates with no harvested energy, it costs battery energy.

To avoid link collision, we construct a weighted-conflict graph based on the interference model in this article, and utilize interval vertex coloring for link scheduling. All link nodes with the same color can activate in the same time slot. Note that the goal of this article is to activate interference-free links with the maximum weights, rather than activating as many links as possible at the same time.

The contributions are as follows:

1) This article proposes a new problem, called the harvested energy-saving link scheduling problem and proves its NP-hardness. We then construct the energy model to formulate it.
2) We design CS to schedule links by arranging them into a non-increasing with weight, and design CSM to handle the case when some links have the same weight. We also propose DA to activate links in the distributed networks and analyze the performance of the algorithms.
3) This article conducts simulations to analyze the link duty cycle, energy waste rate and link activation time. The results show that the link duty cycle closes to a high ratio 2, and the energy waste rate eventually can be maintained at an average of about 19%. The link activation time is at least 0.987τ on average, where τ is the maximum time length that the energy harvested by one node during one time slot can support the node to communicate. Besides, we compare our algorithms with LS-rWSN in [23], which maximizes throughput under the ideal energy harvesting time. The results show that under the same topology, our algorithm CS costs more time than LS-rWSN, and takes at most 5.69% extra time than LS-rWSN to schedule all links. It indicates that our algorithms have comparable throughput with that obtained by LS-rWSN.

Some key symbols and their meanings are listed in Table. 1

TABLE 1. Symbol and meaning.

Symbol	Description
$G = (V, E)$	Directed graph with node set V and link set E.
v_i	Node i.
l_{ij}	Directed link from v_i to v_j.
λ	Battery storage efficiency.
R_i	Interference range.
R_c	Communication range.
r	Ratio between R_i and R_c, $i.e.$, $r = \frac{R_i}{R_c}$.
$d(v_i, v_j)$	Euclidean distance between v_i and v_j.
t	Time slot.
e	Energy harvested by one node in one time slot.
τ	Time length for communication with e energy.
e_p	Harvested energy consumption for link activation.
e_b	Battery energy consumption for link activation.
$H_i(t)$	Energy harvested by v_i during t.
$B_i(t)$	Battery energy of v_i at the beginning of t.
$C_i(t)$	Sum of energy consumed and wasted by v_i during t.
w_{ij}	Weight of link l_{ij}.
$l_{ij}(w_{ij})$	Link node corresponding to l_{ij}.
G_w	Weighted-conflict graph.
γ_{ij}	Ratio of link l_{ij}.
L	Set of all unscheduled links.
L_c	Set of the link nodes corresponding to links in L.
A_w	Set of link nodes with the same color.
A	Set of activated links.
N_i	Set of links indicating from v_i.
t_{max}	The time slot that is not assigned to the interference links and has the greatest link weight.
t_{last}	The last time slot that all links can be scheduled before this time slot.

The rest of this article is organized as follows. Section II reviews the related works. Section III presents the system model and then formulates the harvested energy-saving link scheduling problem. Section IV and Section V introduce the CS, CSM, and DA in detail. Section VI describes the simulation experiment and analyzes the experimental results. Finally, Section VII concludes the paper.

II. RELATE WORK

In recent years, the mature energy harvesting technology has greatly extended the lifetime of WMNs. Some algorithms are designed to balance energy consumption and network performance. Liu *et al.* designed the algorithm for energy management to keep the battery at a certain target level and maximize network utility [27]. Mao *et al.* proposed the optimal energy allocation policy to solve the energy allocation problem of sensing data and transmitting [28]. Peng *et al.* proposed the real-time adaptive energy management based on the information observed in the past to extend the network lifetime [29]. They also proposed the energy harvest-store (use) model to reduce energy waste caused by storing efficiency.

In the energy harvesting WMNs, an appropriate link scheduling protocol can effectively reduce the energy waste caused by link interference. [21], [30], and [31] performed the link scheduling. Sun *et al.* proposed two efficient link scheduling algorithms based on whether or not to assign weight for each link to improve network throughput [30]. They utilized the harvest-store-use model, in which the harvested energy must be stored in the battery before it was used to ensure that the nodes have enough energy to transmit. Tony *et al.* proposed the greedy heuristic to schedule links through the harvest-use-store model first introduced in [25], and took the effect of limited battery capacity on link scheduling into consideration [21]. Chen *et al.* designed three interference-free link scheduling algorithms to solve the problem of Minimum Latency Aggregation Scheduling in the energy harvesting WMNs with battery capacity limitation [31]. All of the above can effectively improve network throughput and reduce energy waste. However, their default battery storage efficiency is 100%, which is merely impossible in real scenarios. None of them take into account the impact of battery storage efficiency on energy waste.

The works in [14], [22], and [24] allow for the impact of battery storage efficiency. In [14], the double-threshold policy was proved to be the optimum policy to maximize energy efficiency. When the harvested energy was higher than the upper threshold, the energy was stored in the battery. When it was lower than the lower threshold, the energy was discharged from the battery. Otherwise, the harvested energy was directly used. Yuan *et al.* argued that the harvest-store-use model would cause low energy efficiency due to the battery storage loss in the real world. Therefore, they adopted the harvest-use-store model for point-to-point data transmission [22]. On the basis of the static and block fading channels, the optimal energy policies were proposed to maximize throughput, and extended to minimize the energy consumption subject to the storage efficiency and delay constraints [24].

So far, only [23] has considered the effect of battery storage efficiency on link scheduling. Tony *et al.* proposed LS-rWSN to minimize the time slots for link scheduling in harvest-use-store. Each node had a different energy harvesting time. They calculated the earliest time slot for each node to harvest enough energy and the earliest time slot that each link could activate when the battery was not perfect (the storage efficiency was below 100% and there existed energy leakage). However, their works focus on saving time rather than reducing energy waste. Furthermore, once the energy harvesting time of each node is determined, it will remain unchanged in the following scheduling, which is ideal compared with the real situation. With the previous works, our algorithms divide the network period into several equal-size time slots, and each node randomly harvests energy in each time slot. Based on the harvested energy consumed when the link activates, we assign a weight to each link so as to reduce energy waste.

III. SYSTEM MODEL AND PROBLEM FORMULATION

This section introduces the system model including the network model, the interference model, and the energy model. We formulate the harvested energy-saving link scheduling problem and prove it is NP-hard.

A. SYSTEM DESCRIPTION

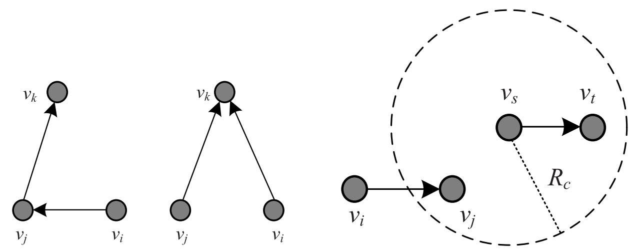
1) NETWORK MODEL

A WMN consists of a sink and some nodes with the functions of data gathering, sensing, energy harvesting and storing, communication and computing. To show the data communication relationship among nodes clearly, the WMN is represented by a directed graph $G = (V, E)$, where node $v_i \in V$, and directed link from v_i to v_j $l_{ij} \in E$. Define R_c as the communication range and $d(v_i, v_j) = ||v_i - v_j||$ as the Euclidean distance between v_i and v_j. A successful transmission from v_i to v_j must satisfy $d(v_i, v_j) \leq R_c$.

In this article, two types of network topologies will be discussed: data gathering tree (DGT) rooted at the sink and direct acyclic graph (DAG). In DGT, the intermediate node collects data from its child nodes and forwards to its parent node. There is only one path for any intermediate node transmitting data to the sink. In DAG, it has no directed cycles, and each node can choose multiple paths for data transmitting to the sink.

2) INTERFERENCE MODEL

In the energy harvesting WMNs, every node communicates the neighbors within the range of R_c, which can also cause interference. The interference model in this article is the protocol interference model [32]. All nodes have the same communication range R_c and interference range R_i. Define $r = \frac{R_i}{R_c}$ as the ratio between R_i and R_c. In general, $2 \leq r \leq 4$.



(a) Primary interference. (b) Secondary interference.

FIGURE 1. Primary interference and secondary interference.

When a node starts to transmit data, there are two types of interference: primary interference and secondary interference [33]. As shown in Fig. 1a, if v_j sends data to v_k while receiving data from v_i, or v_k receives data from v_i and v_j simultaneously, it will cause the primary interference. Besides, as shown in Fig. 1b, when there is a transmission from v_i to v_j, the secondary interference occurs when the receiver node v_j is within the communication range of another transmission's sender node v_s.

If two links do not interfere with each other, the two links are interference-free. Both links can activate in one time slot.

3) ENERGY MODEL

In the energy model, the uneven distribution and time variability of environmental energy lead to different energy harvested by each node in energy harvesting WMNs. To simplify the complex harvest situation, we divide the network period into a series of equal-size time slots. In each time slot t, assume that each node with rich ambient energy can harvest one unit of energy e [11]. Let $H_i(t)$ denote whether v_i harvests energy at t, it can be defined by Eq. (1):

$$H_i(t) = \begin{cases} e, & v_i \textit{ can harvest energy at } t; \\ 0, & v_i \textit{ cannot harvest energy at } t. \end{cases} \tag{1}$$

The harvested energy of each node in each time slot is divided into three parts: one for link activation consumption, one for storage, and the last one for energy waste caused by the limitation of battery storage efficiency.

Consumption. This article only considers the energy consumption of data transmission when the link activates. It is reasonable because the energy consumption of a node for its communications is much greater than the energy consumption for its sensing and computing, *e.g.*, the energy consumption of communication, sensing and computing is 180.1 mJ, 17.242 mJ, and 5.2 mJ respectively [34]. In addition, the energy consumption of nodes during transmission and reception is basically equivalent [35]. All the harvested energy of v_i is consumed for activation, and it can support v_i to send or receive for τ time. For example, if v_i and v_j can harvest energy simultaneously, they are so called harvested nodes, and the harvested energy of v_i and v_j can be used to support link l_{ij} to activate for τ time.

Storage. There are two cases in which the harvested energy needs to be stored in the battery. Firstly, the harvested energy of the inactive harvested node needs to be stored in the battery. For example, if v_i harvests energy and l_{ij} does not activate in t, the harvested energy of v_i will be stored in its battery. Secondly, not all the harvested energy of v_i is consumed when l_{ij} activates. For instance, v_i is a harvested node, and the battery energy of the non-harvested node v_j is less than e. To minimize the energy waste, v_i spends e_p harvested energy, which equals to the v_j's battery energy to maintain l_{ij} activation for $\frac{e_p}{e}\tau$ time. The remaining $e - e_p$ energy of v_i is stored in v_i's battery.

Waste. The harvested energy is wasted when stored in the battery due to the imperfect storage efficiency of batteries, the leakage rate and the energy saturation in batteries [23]. Assuming that the battery has no leakage rate and the battery capacity is ideal, energy waste is only caused by the imperfect battery storage efficiency. Represent the battery storage efficiency with λ $(0 < \lambda \leq 1)$. If all the harvested energy is stored, in fact, only λe energy will be stored successfully, and $(1 - \lambda)e$ energy is wasted.

It is necessary to calculate the battery energy of each node v_i at the beginning of each time slot t. It can be updated by Eq. (2):

$$B_i(t) = B_i(t-1) + H_i(t-1) - C_i(t-1) \tag{2}$$

where $B_i(t-1)$ is the battery energy of v_i at the beginning of $t - 1$. $H_i(t-1)$ has been defined in Eq. (1). $C_i(t-1)$ represents the sum of energy consumed and wasted by v_i in $t - 1$. Specifically, when v_i harvests energy e and costs the harvested energy for l_{ij} activation in $t - 1$, the consumption is e and waste is 0. Therefore, $C_i(t-1) = e$. If v_i consumes e_p $(e_p < e)$ harvested energy for l_{ij} activation, $(1 - \lambda)(e - e_p)$

energy will be wasted while $\lambda(e - e_p)$ energy will be stored in the battery, and $C_i(t-1) = e_p + (1-\lambda)(e - e_p)$. In addition, if v_i sleeps, all the harvested energy will be stored, $(1-\lambda)e$ energy is wasted and $C_i(t-1) = (1-\lambda)e$. When v_i cannot harvest energy in $t-1$, it will discharge battery energy for l_{ij} activation, which will not cause energy waste. Define e_b as the battery energy consumed by v_i when l_{ij} activates, $C_i(t-1) = e_b$. Otherwise, if v_i sleeps, it will have neither energy consumption nor waste and $C_i(t-1) = 0$.

B. PROBLEM FORMULATION

For a given energy harvesting WMN, harvested energy-rich links and harvested energy-poor links coexist due to uneven energy distribution. In the case of limited battery storage efficiency, scheduling as many harvested energy-rich links as possible can greatly reduce the energy that needs to be stored in the battery, thus reducing waste. We refer to such an interference-free link scheduling that can save energy as the harvested energy-saving link scheduling.

For the sake of solving our problem, each link should be assigned a weight to indicate the amount of harvested energy that it needs to consume for its activation.

Definition 1: For any link $l_{ij} \in E$, let w_{ij} denote the weight of the link l_{ij}, and w_{ij} equals to the consumption of the harvested energy when l_{ij} activates.

Before calculating the weight of each link, we first discuss the energy consumption, storage, and waste of different categories of links.

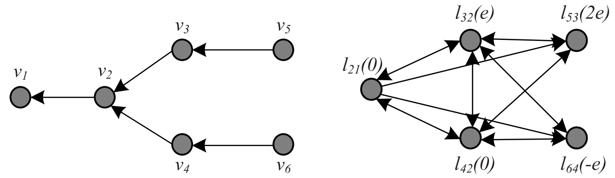
Each node $v_k \in V$ has four states in t according to whether v_k harvests energy and how much the battery energy is in v_k:
1) v_k is a harvested node ($H_k(t) = e$);
2) v_k is a non-harvested node and has battery energy with at least energy e ($H_k(t) = 0, B_k(t) \geq e$);
3) v_k is a non-harvested node and has battery energy with at most energy e ($H_k(t) = 0, 0 < B_k(t) < e$);
4) v_k is a non-harvested node and has no battery energy ($H_k(t) = 0, B_k(t) = 0$).

According to the harvested energy consumption on link activation, the nodes in the four states can form up to five categories of links. We take l_{ij} as an example to further discuss links in the five categories.

Category 1: $H_i(t) = e$ and $H_j(t) = e$. l_{ij} costs all the harvested energy of both v_i and v_j without energy waste when it activates. All the harvested energy can support l_{ij} to activate for τ time.

Category 2: There is only one harvested node in l_{ij}, and the battery energy of the non-harvested node is no less than e. Let $H_i(t) = e$, $H_j(t) = 0$, and $B_j(t) \geq e$. It consumes all the harvested energy of v_i and e battery energy of v_j to activate l_{ij}, which causes no energy waste. l_{ij} can activate for τ time.

Category 3: There is only one harvested node in l_{ij}, and the battery energy of the non-harvested node is less than e but more than 0. Let $H_i(t) = e$, $H_j(t) = 0$, and $0 < B_j(t) < e$. It spends a part of the harvested energy of v_i and all battery energy of v_j to activate l_{ij}. Let $e_p (e_p = B_j(t))$ denote the part of the harvested energy that v_i consumes in this time



(a) Communication graph. (b) Weighted-conflict graph.

FIGURE 2. Communication graph and weighted-conflict graph.

slot, and the remaining energy $e - e_p$ is stored in the battery, resulting in the waste of $(1-\lambda)(e - e_p)$ energy. The value of e_p depends on the battery energy of v_j. Therefore, the links that belong to this category may have different weights. But the weights are always positive and smaller than e. When l_{ij} activates, it wastes less while the weight is higher. A part of harvested energy of v_i and all battery energy of v_j can support l_{ij} to activate for $\frac{e_p}{e}\tau$ time.

Category 4: Both v_i and v_j are non-harvested nodes, and the battery energy is sufficient. l_{ij} costs battery energy for activation. Let e_b represent the battery energy consumption when l_{ij} activates. v_i and v_j take $e_b = min\{B_i(t), B_j(t), e\}$ energy from their batteries to support l_{ij} activation for $\frac{e_b}{e}\tau$ time without energy waste.

Category 5: There is no less than one node in l_{ij} having neither harvested energy nor battery energy (*e.g.* $H_i(t) = 0$, $B_i(t) = 0$). The link that belongs to this category cannot activate because of insufficient energy although it may have harvested energy. If v_j is a harvested node, l_{ij} will waste $(1-\lambda)e$ energy. Otherwise, it will not waste energy.

The link weights in the five categories are as follows. Note that when the link belongs to *Category* 5, we set $-e$ as its weight and have the following:

$$w_{ij} = \begin{cases} 2e, & l_{ij} \text{ belongs to Category 1;} \\ e, & l_{ij} \text{ belongs to Category 2;} \\ e_p, & l_{ij} \text{ belongs to Category 3;} \\ 0, & l_{ij} \text{ belongs to Category 4;} \\ -e, & l_{ij} \text{ belongs to Category 5.} \end{cases} \tag{3}$$

Given a communication graph $G = (V, E)$ and the interference model, a conflict graph can be obtained [36]. On this basis, we propose a weighted-conflict graph $G_w = (V', E')$ to model the harvested energy-saving link scheduling. In G_w, nodes are called link nodes, and each link node $l_{ij}(w_{ij}) \in V'$ corresponds to a directed link from v_i to v_j in G with a weight w_{ij}. There is a link $e' \in E'$ between two link nodes if the corresponding links of the link nodes in G interfere with each other. The weight of each link is time-varying because of the time-varying harvested energy, which results in the time-varying weighted-conflict graph. To avoid link interference when scheduling, this article models the link scheduling as the interval vertex coloring. In G_w, any pair of adjacent link nodes have different colors.

For example, there is a communication graph as shown in Fig. 2a. The communication range R_c is one hop and the interference range R_i is two hops. In time slot t, $H_3(t) = e$, $H_5(t) = e$. The battery energy of non-harvested nodes is $B_1(t) = \lambda e$, $B_2(t) = e + \lambda e$, $B_4(t) = \lambda e$, $B_6(t) = 0$. The weight of each link can be calculated by Eq. (3). Consequently, the weighted-conflict graph is constructed, as shown in Fig. 2b. Obviously, we can get an effective harvested energy-saving link scheduling by selecting the same color link nodes with the greatest weight and activating the corresponding links in G.

Theorem 1: The harvested energy-saving link scheduling problem is NP-hard.

Proof: In our scheduling, we always pick out the link nodes that are interference-free with the non-increasing order of link weights in each time slot to form a new independent set, thereby maximizing the sum of the link weights in the independent set. It can be formulated as finding a Maximum Weighted Independent Set (MWIS) in the weighted-conflict graph under the protocol interference model. The MWIS problem is proved to be NP-hard [37]. So our problem is NP-hard. □

IV. CENTRALIZED ALGORITHMS AND PERFORMANCE ANALYSIS

This section describes the CS and CSM for the harvested energy-saving link scheduling problem in detail. To reduce the harvested energy waste, CS is proposed to schedule links with the non-increasing order of weights. Furthermore, CSM is designed for link scheduling when the weights of links are the same.

A. CS ALGORITHM DESIGN

In each time slot, the interference-free links with sufficient energy can activate simultaneously. We propose a centralized algorithm called Centralized Scheduling (CS) to schedule links. In CS, all unscheduled links in G are recorded in the set $L = \{l_1, l_2, l_3, \ldots, l_m\}$, and their corresponding link nodes in G_w are in the set $L_c = \{l_1(w_1), l_2(w_2), l_3(w_3) \ldots l_n(w_n)\}$. Afterward, CS sorts the link nodes in L_c with the non-increasing order of the link weights. Starting from the first one, the link nodes are colored one by one and then those with the same color are picked out to form an independent set A_w. The links corresponding to the link nodes in A_w are recorded in the set A for simultaneous activation. However, when the weight of the link node in A_w is $-e$, corresponding link will not appear in A because of insufficient energy for activation. The link can activate in the subsequent time slot with sufficient energy.

At the beginning of each time slot, the battery energy of each node and the weight of each link are renewed. After the links activate, L is updated by deleting the links that have been scheduled. The CS is described in Algorithm 1.

Taking the communication graph and G_w shown in Fig. 2 as an example, the process of CS is described as follows. Define a matrix H to display the harvested energy of nodes in each time slot as shown below:

$$H = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{array} \begin{array}{c} \begin{array}{ccccc} t_0 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} e & 0 & e & e & 0 \\ e & e & e & 0 & 0 \\ e & 0 & 0 & e & e \\ e & e & 0 & e & 0 \\ 0 & 0 & 0 & 0 & e \\ e & e & 0 & 0 & e \end{pmatrix} \end{array} \tag{4}$$

Another matrix W is introduced to show the weights of links in each time slot and is defined as follows:

$$W = \begin{array}{c} \\ l_{21} \\ l_{32} \\ l_{42} \\ l_{53} \\ l_{64} \end{array} \begin{array}{c} \begin{array}{ccccc} t_0 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} 2e & -e & 2e & \lambda(1-\lambda)e & 0 \\ 2e & \lambda e & \lambda e & \lambda(1-\lambda)e & \lambda(1-\lambda)e \\ 2e & 2e & \lambda e & \lambda(1-\lambda)e & 0 \\ -e & -e & -e & -e & 2e \\ 2e & 2e & 0 & e & \lambda e \end{pmatrix} \end{array} \tag{5}$$

Firstly, all links are in the set $L = \{l_{21}, l_{32}, l_{42}, l_{53}, l_{64}\}$. Then the interference-free links with sufficient energy are scheduled in each time slot. In t_0, links updates weights based on battery energy and harvested energy. According to the harvested energy matrix H, only v_5 is the non-harvested node. Therefore, all link weights except l_{53} is $2e$. After updating the link weights, the link nodes corresponding to the links in L are recorded in L_c with the non-increasing order of weights, *i.e.*, $L_c = \{l_{21}(2e), l_{32}(2e), l_{42}(2e), l_{64}(2e), l_{53}(-e)\}$. Only $l_{21}(2e)$ and l_{21} can be placed in the sets A_w and A, respectively. Therefore, only l_{21} activates in this time slot. The remaining

Algorithm 1 CS Algorithm Design

Input: A communication graph $G = (V, E)$
Output: A valid centralized link scheduling

```
Construct the weighted-conflict graph G_w;
Set a set L to record all links in G;
Set three sets L_c, A_w, A;
while L ≠ ∅ do
    Each node updates the battery energy;
    Each link updates the weight in G_w;
    L_c = ∅, A_w = ∅, A = ∅;
    L_c records the link nodes corresponding to the links
    existing in L from G_w with the non-increasing order
    of weights;
    for Each link node l_i(w_i) in L_c do
        if l_i(w_i) is not the neighbor of any link node in A_w
        then
            Put l_i(w_i) to A_w;
        end if
        if w_i ≠ −e then
            Put l_i to A;
        end if
    end for
    Activate links in A;
    Delete the links in A from L.
end while
```

link nodes $l_{32}(2e)$, $l_{42}(2e)$, $l_{64}(2e)$, $l_{53}(-e)$ are neighbors of $l_{21}(2e)$, and their corresponding links (*i.e.*, $l_{32}, l_{42}, l_{64}, l_{53}$) interfere with l_{21}. After the activation of l_{21}, L is updated by deleting l_{21} and becomes a new set $L = \{l_{32}, l_{42}, l_{53}, l_{64}\}$. In t_1, L_c records the link nodes corresponding to links existing in L, *i.e.*, $L_c = \{l_{42}(2e), l_{64}(2e), l_{32}(\lambda e), l_{53}(-e)\}$. Obviously, $l_{42}(2e)$ and l_{42} are placed in A_w and A, respectively. Repeat these steps to activate one or more links in each time slot till all links are scheduled. Note that in t_3, although $l_{53}(-e)$ is in A_w, l_{53} is not be placed in A since its weight is $-e$.

B. CSM ALGORITHM DESIGN

According to the CS, we can obtain an effective link scheduling by activating the links with the non-increasing order of weights. But it does not handle the situation when the weights of some links are same. For example, there is a link l_{ij} interfering with both links l_{pq} and l_{st}, while l_{pq} and l_{st} are interference-free. All of their weights are $2e$ in t. If l_{ij} activates, l_{pq} and l_{st} have to sleep in this time slot, causing the four nodes to store the harvested energy in their batteries. As a result, the total waste is $4(1-\lambda)e$.

Conversely, if l_{pq} and l_{st} activate firstly, only l_{ij} needs to sleep. The total waste is $2(1-\lambda)e$. Obviously, it can save energy by activating the link with the smallest sum of the interfering link weights if the links have the same weight. We also get an efficient harvested energy-saving link scheduling by defining a ratio for each link and call it Centralized Scheduling with Minimum interference weights (CSM).

Definition 2: Suppose $I(l_{ij})$ is the set of links interfering with l_{ij}. For any link $l_{ij} \in E$, let $\gamma_{ij} = \frac{w_{ij}}{\sum w_{mn}+e}$ denote the ratio of l_{ij}, where w_{ij} is the weight of l_{ij}, w_{mn} is the weight of l_{mn} existing in $I(l_{ij})$ and $w_{mn} \neq -e$.

By comparing the ratios instead of the weights, the activation priority of links can be sorted. In the above example, the ratios of l_{ij}, l_{pq} and l_{st} are $\gamma_{ij} = \frac{2}{5}$, $\gamma_{pq} = \frac{2}{3}$, $\gamma_{st} = \frac{2}{3}$, respectively. Evidently, l_{pq} and l_{st} should activate firstly. Nevertheless, when the weight of l_{st} changes into $-e$, the ratio of l_{ij} becomes $\gamma_{ij} = \frac{w_{ij}}{w_{pq}+e} = \frac{2}{3}$, l_{ij} and l_{pq} have the same activation priority. Note that $\sum w_{mn} + e$ is used instead of $\sum w_{mn}$ in CSM because $\sum w_{pq} = 0$ is considered. The CSM is described in Algorithm 2.

For example, recalling the harvested energy matrix H and the weight matrix W in Eq. (4) and Eq. (5), the ratio matrix γ can be obtained as follows:

$$\gamma = \begin{array}{c} \\ \gamma_{21} \\ \gamma_{32} \\ \gamma_{42} \\ \gamma_{53} \\ \gamma_{64} \end{array} \begin{array}{c} \begin{array}{ccccc} t_0 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} \frac{2}{7} & \frac{\lambda}{5+\lambda} & \frac{2}{1+\lambda} & -\frac{1}{1+3\lambda} & -\frac{1}{3+\lambda} \\ \frac{2}{7} & \frac{\lambda}{5+\lambda} & \frac{\lambda}{3} & \frac{\lambda}{1+2\lambda} & -\frac{1}{3+\lambda} \\ \frac{2}{7} & \frac{2}{3+2\lambda} & -\frac{1}{3+\lambda} & \frac{\lambda}{1+2\lambda} & -\frac{1}{3+\lambda} \\ -\frac{1}{5} & -\frac{1}{3+\lambda} & -\frac{1}{1+\lambda} & -\frac{1}{1+2\lambda} & 2 \\ \frac{2}{5} & \frac{2}{3+\lambda} & -\frac{1}{1+\lambda} & \frac{\lambda}{1+2\lambda} & \lambda \end{pmatrix} \end{array} \tag{6}$$

Firstly, all unscheduled links are in the $L = \{l_{21}, l_{32}, l_{42}, l_{53}, l_{64}\}$. Then the interference-free links with sufficient energy are scheduled in each time slot. In t_0, after updating the link weights, the links in L update their ratios according to the sum of the weights of their interference links. The calculation results of the ratios are shown in Eq. (6). The link nodes are recorded in the $L_c = \{l_{64}(2e), l_{21}(2e), l_{32}(2e), l_{42}(2e), l_{53}(-e)\}$ with the non-increasing order of ratios. Only $l_{64}(2e)$ and l_{64} can be placed in the sets A_w and A, respectively. l_{64} activates in this time slot. After activation, L is updated to delete l_{64}. Repeat these steps in each time slot until all links are scheduled.

Algorithm 2 CSM Algorithm Design

Input: A communication graph $G = (V, E)$
Output: A valid centralized link scheduling

Construct the weighted-conflict graph G_w;
Set a set L to record all links in G;
Set three sets L_c, A_w, A;
Initialize $sum = 0$;
while $L \neq \emptyset$ **do**
  Each node updates the battery energy;
  Each link updates the weight in G_w;
  $L_c = \emptyset, A_w = \emptyset, A = \emptyset$;
  L_c records the link nodes in G_w corresponding to the links existing in L;
  for Each link node $l_i(w_i)$ in L_c **do**
    $sum = 0$;
    for Each link node $l_j(w_j)$ ($j \neq i$) in L_c **do**
      if $l_j(w_j)$ is the neighbor of $l_i(w_i)$ and $w_j \neq -e$ **then**
        $sum+ = w_j$;
      end if
    end for
    Calculate $\gamma_i = \frac{w_i}{sum+e}$;
  end for
  Sort the link nodes in L_c with the non-increasing order of ratios;
  for Each link node $l_i(w_i)$ in L_c **do**
    if $l_i(w_i)$ is not the neighbor of any link node in A_w **then**
      Put $l_i(w_i)$ to A_w;
    end if
    if $w_i \neq -e$ **then**
      Put l_i to A;
    end if
  end for
  Activate links in A;
  Delete the links in A from L.
end while

C. PERFORMANCE ANALYSIS

Define $r = \frac{R_i}{R_c}$ as the ratio of R_i to R_c. Let $d(v_i, v_j)$ be the Euclidean distance between v_i and v_j.

Theorem 2: If there is a set L to record all links in G, for any link l_{ij} in L and its interference links set $I(l_{ij})$, at least $\frac{|I(l_{ij})|}{C} + 1$ time slots are needed to schedule all links in L, where $C = \frac{(3r+1)^2}{(r-1)^2}$, $|I(l_{ij})|$ is the number of links in $I(l_{ij})$.

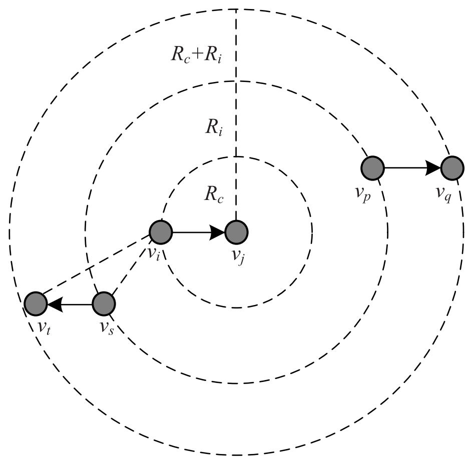


FIGURE 3. Interference links in Theorem 1.

Proof: As shown in Fig. 3, If v_i can communicate with v_j, v_i must be within the R_c range of v_j. If link l_{pq} interferes with l_{ij}, the sender node v_p is within the R_i range of v_j, and the receiver node v_q is within the $R_i + R_c$ range of v_j. The distance between two sender nodes v_i and v_s transmitting simultaneously without interference must satisfy $d(v_i, v_s) \geq d(v_i, v_t) - d(v_t, v_s) \geq R_i - R_c$. At most $C = \frac{\pi[R_i+R_c+0.5(R_i-R_c)]^2}{\pi[0.5(R_i-R_c)]^2} = \frac{(3r+1)^2}{(r-1)^2}$ links with radius $\frac{(R_i-R_c)}{2}$ can be placed within the $R_i + R_c$ range of v_j [38]. It means that there are $\frac{|I(l_{ij})|}{C}$ links interfering with each other in the set $I(l_{ij})$. Therefore, at least $\frac{|I(l_{ij})|}{C} + 1$ time slots are needed to schedule all links in L. □

Theorem 3: Let η_{waste} denote the waste rate of the harvested energy that schedules all links in a network period, and always have $\eta_{waste} \leq 1 - \lambda$, where λ is the battery storage efficiency.

Proof: Let $T = \{t_1, t_2, t_3, \ldots, t_n\}$ denote the time slots in a network period. In each time slot $t_i (t_i \in T)$, $A_i = \{a_{i1}, a_{i2}, \ldots, a_{im}\}$ records the harvested nodes that consume all the harvested energy for link activation. $P_i = \{p_{i1}, p_{i2}, \ldots, p_{ik}\}$ records the harvested nodes that cost e_p $(0 < e_p < e)$ energy for link activation. $S_i = \{s_{i1}, s_{i2}, \ldots, s_{ip}\}$ records the harvested nodes that sleep. The nodes in A_i cannot waste energy, while the nodes in other sets waste energy. Specifically, the node in P_i wastes $(1-\lambda)(e-e_p)$ amount of energy when it costs e_p energy for link activation. Different e_p of each node in P_i leads to different energy waste. Otherwise, the node in S_i wastes $(1 - \lambda)e$ amount of energy. Define $|A_i|$, $|P_i|$, $|S_i|$ as the number of nodes in A_i, P_i, S_i. Therefore, the total energy waste E_{waste} in a network period can be calculated as Eq. (7):

$$\begin{aligned} E_{waste} &= \sum_{i=1}^{n} (0 \cdot |A_i| + \sum_{p_{i1}}^{p_{ik}} (1-\lambda)(e-e_p) + (1-\lambda)e|S_i|) \\ &\leq (1-\lambda)e \sum_{i=1}^{n} (|P_i| + |S_i|) \end{aligned} \tag{7}$$

The total harvested energy E_{sum} in a network period is:

$$E_{sum} = \sum_{i=1}^{n} (|A_i| + |P_i| + |S_i|)e. \tag{8}$$

Thus, the energy waste rate η_{waste} is computed as Eq. (9):

$$\begin{aligned} \eta_{waste} &= \frac{E_{waste}}{E_{sum}} \leq (1-\lambda) \frac{\sum_{i=1}^{n} (|P_i| + |S_i|)}{\sum_{i=1}^{n} (|A_i| + |P_i| + |S_i|)} \\ &\leq (1-\lambda) \end{aligned} \tag{9}$$

□

Theorem 4: In the worst case, the time complexity of CS and CSM is $O(m^3)$, where m is the number of links.

Proof: Given that there are n nodes and m links in the network, CS first takes $O(n)$ to update the battery energy and $O(m)$ to update the link weights. Sorting the link nodes in L_c costs at most $O(m^2)$. It takes at most $O(m^2)$ to select a link node from L_c which is not the neighbor of any link node in A_w, because CS needs to compare the selected link node with all link nodes in A_w. Obviously, the time complexity of CS to schedule all links is $O(m^3)$.

In CSM, it is at most $O(m^2)$ more than CS to calculate the link ratios, and the total costs is at most $O(m^3)$ steps more than CS after scheduling all links. Therefore, the time complexity of CSM is also $O(m^3)$. □

V. DISTRIBUTED ALGORITHMS

This section proposes the Distributed Algorithm (DA) for the harvested energy-saving link scheduling problem. For each node v_i, let $N_i = \{l_{i1}, l_{i2}, \ldots, l_{im}\}$ record the links indicating from v_i (the sender node of the links is v_i). In DA, a random order is used to schedule links. Suppose there is a contention-based MAC for a node to compete. If v_i competes successfully, each link l_{ij} $(1 \leq j \leq m)$ in N_i finds the time slot t_{max} that is not assigned to the interference links and has the greatest weight. The weights of the links associated with v_i and v_j are changed after t_{max}, because the battery energy of v_i and v_j decreases after l_{ij} activates in t_{max}. After all links in N_i are scheduled, v_i no longer participates in the next competition. Let t_{last} represent the last time slot, and all links can be scheduled before t_{last}. The DA is described in Algorithm 3.

Also taking the harvested energy matrix H in Eq. (4) as an example to introduce the process of DA. Before link scheduling, the link weights are initialized by assuming that the harvested energy of each node is stored in its battery, and the following initial weight matrix W_0 is obtained:

$$W_0 = \begin{array}{c} \\ l_{21} \\ l_{32} \\ l_{42} \\ l_{53} \\ l_{64} \end{array} \begin{array}{c} \begin{array}{ccccc} t_0 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} 2e & \lambda e & 2e & e & 0 \\ 2e & \lambda e & \lambda e & e & e \\ 2e & 2e & e & e & 0 \\ -e & -e & -e & -e & 2e \\ 2e & 2e & 0 & e & e \end{pmatrix} \end{array} \tag{10}$$

Suppose v_2 successfully competes for the channel, and the link in $N_2 = \{l_{21}\}$ finds t_{max}. In this example, t_0 is assigned

Algorithm 3 DA Algorithm Design

Input: All nodes and their next-hop nodes

Output: A valid distributed link scheduling

1: Set a set T_a;
2: Each node v_i has a set N_i to record all links indicating from v_i;
3: **while** Exists $N_i \neq \emptyset$ **do**
4:   Each node v_i where $N_i \neq \emptyset$ competes channels;
5:   **if** v_i competes successfully **then**
6:     **for** each link l_{ij} in N_i **do**
7:       $T_a = \emptyset$;
8:       **for** $(t = 0; t < t_{last}; t++)$ **do**
9:         **if** t has not been assigned to the interference links **then**
10:           Put t to T_a;
11:         **end if**
12:       **end for**
13:       l_{ij} finds the time slot t_{max} ($t_{max} \in T_a$) with the maximum weight for activation;
14:       l_{ij} broadcasts the information, and the interference links cannot activate in t_{max};
15:       All links related to v_i and v_j update the weights in the time slots after t_{max};
16:     **end for**
17:     $N_i = \emptyset$;
18:   **else**
19:     v_i goes to step 4;
20:   **end if**
21: **end while**

to l_{21} for activation because l_{21} has the greatest weight in t_0. All links that interfere with l_{21} cannot activate in t_0 (*i.e.*, $l_{32}, l_{42}, l_{53}, l_{64}$ cannot activate in t_0 because of interference). After activation, w_{21}, w_{32}, and w_{42} in the time slot after t_0 need to be updated, because the harvested energy originally stored in the batteries of v_1 and v_2 is consumed in t_0, which causes the battery energy of v_1 and v_2 to decrease by λe after t_{max}. The updated weight matrix W_1 is as follows:

$$W_1 = \begin{array}{c} \\ l_{21} \\ l_{32} \\ l_{42} \\ l_{53} \\ l_{64} \end{array} \!\!\begin{array}{c} \begin{array}{ccccc} t_0 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} 2e & -e & 2e & e & 0 \\ 2e & \lambda e & \lambda e & e & e \\ 2e & 2e & e & e & 0 \\ -e & -e & -e & -e & 2e \\ 2e & 2e & 0 & e & \lambda e \end{pmatrix} \end{array} \tag{11}$$

Repeat the above steps until each link is assigned to the appropriate time slot for activation.

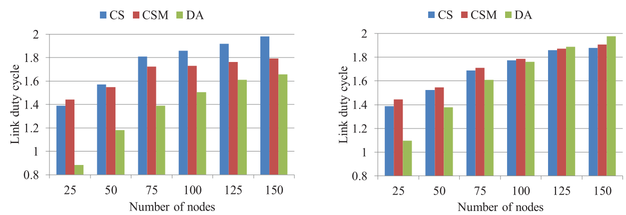
Theorem 5: In the worst case, the time complexity of DA is $O(nm^3)$, where n is the number of nodes and m is the number of links.

Proof: Given that there are n nodes and m links in the network, DA first costs $O(m)$ to initialize N_i. When there is a node v_i successfully competing for the channel, it takes at most $O(m)$ for each link in N_i to find t_{max} to activate. In addition, when the link finds a suitable time slot to activation, it takes at most $O(m^2)$ to update the weights of links related to it. Therefore, it takes at most $O(m^3)$ to process all links in N_i. When all nodes have successfully competed, DA costs at most $O(nm^3)$. □

VI. SIMULATION RESULTS

In this section, we use C++ to build a simulation platform to study the performance of CS, CSM, and DA on the harvested energy-saving link scheduling problem by analyzing link duty cycle, energy waste rate and link activation time. We also study the network throughput by comparing with the greedy algorithm LS-rWSN in [23].

In the simulation, the battery storage efficiency is set to 80%. Nodes with a communication range of 15m and an interference range of 30m are randomly deployed in a square area of 100m × 100m. We build a data gathering tree (DGT) and a directed acyclic graph (DAG) as the test topologies. When the number of nodes varies from 25 to 150 in steps of 25, each topology is generated 50 times randomly to test the average performance. In DA, t_{last} is set to twice the number of links, which has proved to be sufficient in the experiment. The simulation results are as follows.

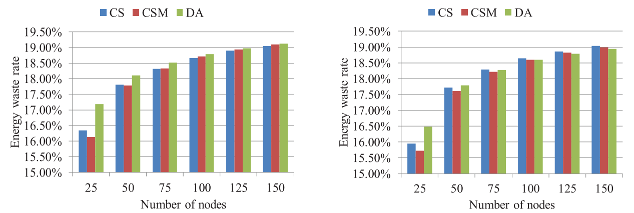


(a) DGT. (b) DAG.

FIGURE 4. Link duty cycle.

Define the link duty cycle (*i.e.*, the ratio of the number of links to the number of time slots required to schedule these links) as an indicator for analyzing the link scheduling performance in each time slot. Fig. 4 shows the link duty cycle of the algorithms. In both DGT and DAG, the link duty cycle increases when the number of nodes increases from 25 to 150. In DGT, DA has the lowest link duty cycle compared with CS and CSM. However, the link duty cycle of CS, CSM, and DA is close to each other as the number of nodes increases, and reaches a high ratio close to 2. It means that CS, CSM and DA do not need to consume too much time slots to schedule all links though their goal is to reduce energy waste rather than scheduling links as quickly as possible. The practical feasibility of the proposed algorithms can be derived from the simulation results.

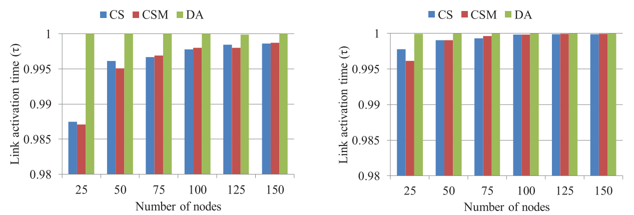
Fig. 5 shows the energy waste rate of the algorithms. From Theorem 3, it can be obtained that the energy waste rate of the simulation experiment does not exceed 20%. As the number of nodes increases, the energy waste rate becomes higher. The reason for this phenomenon is that the increase in the number of nodes in the same area leads to a denser deployment of nodes, which increases the number of interference links per link. But the growth rate of the energy waste rate in both



(a) DGT. (b) DAG.

FIGURE 5. Energy waste rate.

DGT and DAG becomes smaller. We can see from Fig. 5 that the energy waste rate eventually remains at an average of about 19%. Besides, the energy waste rate is inversely proportional to the link duty cycle under the same topology. For example, when the number of nodes is 25, the energy waste rate of DA in DGT is the greatest, while that of CSM is the smallest. Corresponding to the link duty cycle, DA has the smallest one while CSM has the largest one. Because the low duty cycle requires more time slots to schedule all links, resulting in a large number of scheduled nodes storing the harvested energy in their batteries, thereby increasing the energy waste rate.



(a) DGT. (b) DAG.

FIGURE 6. Link activation time.

The link activation time of the algorithms is shown in Fig. 6. It is used to measure the throughput of links in one time slot. The longer the link activation time, the more data can be transferred. In our scheduling, it is at most τ. In both DGT and DAG, the link activation time increases when the number of nodes increases. Furthermore, it is always the greatest in DA, *i.e.*, close to or equal to τ. Because each link in DA has enough time slots to select a time slot with a weight of $2e$ or e for activation. Meanwhile, the link activation time in DGT is shorter compared with the DAG under the same number of nodes. However, it is no less than 0.987τ in DGT and DAG. It can be concluded from the simulation results that CS, CSM and DA have good throughput.

Fig. 7 shows the number of time slots required to schedule all links on DAG. We compare our algorithms with the LS-rWSN which aims to minimize the number of time slots needed to schedule all links. When simulating LS-rWSN, the energy harvesting time of each node is 2, the weight of each link is 1, and the leakage is 0. Further, the battery capacity of each node is $1000000e$, which is enough to accommodate all the harvested energy in the experiment, so that the experimental results is not affected by the battery capacity limitation. It can be concluded from Fig. 7 that when there

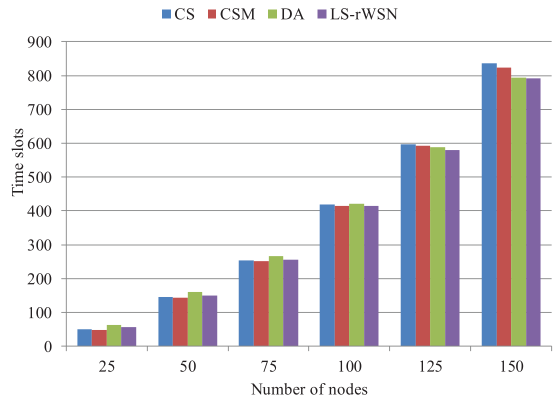


FIGURE 7. Number of time slots required to schedule all links.

are many nodes (the number of nodes is not less than 100), the CS, CSM, and DA need more time slots than LS-rWSN since the node in CS, CSM, and DA cannot harvest energy in two consecutive time slots while it in the LS-rWSN can harvest enough energy every two time slots. Nevertheless, the gap between the number of time slots required by the algorithms in this article and that of LS-rWSN is not large. For example, when the number of nodes is 150, CS needs 836 time slots, while LS-rWSN needs 791 time slot. CS spends 5.69% extra time slots than LS-rWSN to schedule all links under the same topology. It means the CS, CSM, DA spend comparable time slots as LS-rWSN while CS, CSM and DA can reduce energy waste.

VII. CONCLUSION

This article introduces a harvested energy-saving link scheduling problem with the non-ideal battery in the energy harvesting WMNs to reduce energy waste caused by low battery storage efficiency. This problem is proved to be NP-hard, and solved by assigning a weight to each link, which represents the amount of the harvested energy consumption when the link activates. At the same time, a weighted-conflict graph based on the interference model is constructed for interference-free link scheduling. Furthermore, this article proposes the CS, CSM, and DA to implement the link scheduling. Generally, the link with the greatest weight activates preferentially to decrease the harvested energy stored in the battery, so as to reduce energy waste. Finally, simulation results confirm the effectiveness of the algorithms and the correctness of the theoretical analysis. Our algorithms can effectively save energy and maintain good throughput in the energy harvesting WMNs.

REFERENCES

[1] S. Toumpis and A. J. Goldsmith, "Capacity regions for wireless ad hoc networks," *IEEE Trans. Wireless Commun.*, vol. 24, no. 5, pp. 736–748, May 2003.

[2] W. K. G. Seah, Z. A. Eu, and H.-P. Tan, "Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)–Survey and challenges," in *Proc. 1st Int. Conf. Wireless Commun., Veh. Technol., Inf. Theory Aerosp. Electron. Syst. Technol.*, May 2009, pp. 1–5.

[3] Z. Chu, F. Zhou, P. Xiao, Z. Zhu, D. Mi, N. Al-Dhahir, and R. Tafazolli, "Resource allocation for secure wireless powered integrated multicast and unicast services with full duplex self-energy recycling," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 620–636, Jan. 2019.

[4] X. Jiang, J. Polastre, and D. Culler, "Perpetual environmentally powered sensor networks," in *Proc. 4th Int. Symp. Inf. Process. Sensor Netw. (IPSN)*. Los Angeles, CA, USA: UCLA, Apr. 2005, pp. 463–468.

[5] C. Park and P. H. Chou, "Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes," in *Proc. 3rd Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw.*, Reston, VA, USA, vol. 1, Sep. 2006, pp. 168–177.

[6] Y. K. Tan and S. K. Panda, "Energy harvesting from hybrid indoor ambient light and thermal energy sources for enhanced performance of wireless sensor nodes," *IEEE Trans. Ind. Electron.*, vol. 58, no. 9, pp. 4424–4435, Sep. 2011.

[7] Z. Zhu, Z. Chu, N. Wang, S. Huang, Z. Wang, and I. Lee, "Beamforming and power splitting designs for AN-aided secure multi-user MIMO SWIPT systems," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 12, pp. 2861–2874, Dec. 2017.

[8] Z. Chu, Z. Zhu, and J. Hussein, "Robust optimization for AN-aided transmission and power splitting for secure MISO SWIPT system," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1571–1574, Aug. 2016.

[9] Z. Zhu, S. Huang, Z. Chu, F. Zhou, D. Zhang, and I. Lee, "Robust designs of beamforming and power splitting for distributed antenna systems with wireless energy harvesting," *IEEE Syst. J.*, vol. 13, no. 1, pp. 30–41, Mar. 2019.

[10] J. Zhang, Z. Li, and S. Tang, "Value of information aware opportunistic duty cycling in solar harvesting sensor networks," *IEEE Trans. Ind. Informat.*, vol. 12, no. 1, pp. 348–360, Feb. 2016.

[11] J. Zhang, S. Zheng, T. Zhang, M. Wang, and Z. Li, "Charge-aware duty cycling methods for wireless systems under energy harvesting heterogeneity," *ACM Trans. Sensor Netw.*, vol. 16, no. 2, pp. 1–23, Apr. 2020.

[12] P. Li, "Energy storage is the core of renewable technologies," *IEEE Nanotechnol. Mag.*, vol. 2, no. 4, pp. 13–18, Dec. 2008.

[13] Z. Zhao, "The review of energy storage technologies selection," *J. Electr. Electron. Syst.*, vol. 5, no. 172, pp. 796–2332, 2016.

[14] K. Tutuncuoglu, A. Yener, and S. Ulukus, "Optimum policies for an energy harvesting transmitter under energy storage losses," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 3, pp. 467–481, Mar. 2015.

[15] L. Wang, J. Guo, and F. Ji, "Energy management strategy for super capacitor energy storage system based on phase shifted full bridge converter," in *Proc. IEEE Int. Power Electron. Appl. Conf. Expo. (PEAC)*, Nov. 2018, pp. 1–6.

[16] S. Kim and J. Heidemann, "Energy harvesting with supercapacitor-based energy storage," in *Smart Sensors and Systems*. Cham, Switzerland: Springer, 2015, pp. 215–241.

[17] J. Zhang, J. Wang, and X. Wu, "Research on supercapacitor charging efficiency of photovoltaic system," in *Proc. Asia–Pacific Power Energy Eng. Conf.*, Mar. 2012, pp. 1–5.

[18] Z. Zhu, Z. Chu, Z. Wang, and I. Lee, "Outage constrained robust beamforming for secure broadcasting systems with energy harvesting," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7610–7620, Nov. 2016.

[19] Z. Zhu, Z. Chu, F. Zhou, H. Niu, Z. Wang, and I. Lee, "Secure beamforming designs for secrecy MIMO SWIPT systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 424–427, Jun. 2018.

[20] Z. Chu, Z. Zhu, M. Johnston, and S. Y. Le Goff, "Simultaneous wireless information power transfer for MISO secrecy channel," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 6913–6925, Sep. 2016.

[21] Tony, S. Soh, M. Lazarescu, and K.-W. Chin, "Link scheduling in rechargeable wireless sensor networks with harvesting time and battery capacity constraints," in *Proc. IEEE 43rd Conf. Local Comput. Netw. (LCN)*, Chicago, IL, USA, Oct. 2018, pp. 235–242. [Online]. Available: https://ieeexplore.ieee.org/document/8638233

[22] F. Yuan, Q. T. Zhang, S. Jin, and H. Zhu, "Optimal harvest-use-store strategy for energy harvesting wireless systems," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 698–710, Feb. 2015.

[23] T. Tony, S. Soh, K.-W. Chin, and M. Lazarescu, "Link scheduling in rechargeable wireless sensor networks with imperfect batteries," *IEEE Access*, vol. 7, pp. 104721–104736, 2019.

[24] F. Yuan, S. Jin, K.-K. Wong, Q. T. Zhang, and H. Zhu, "Optimal harvest-use-store design for delay-constrained energy harvesting wireless communications," *J. Commun. Netw.*, vol. 18, no. 6, pp. 902–912, Dec. 2016.

[25] R. Rajesh, V. Sharma, and P. Viswanath, "Capacity of fading Gaussian channel with an energy harvesting sensor node," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Houston, TX, USA, Dec. 2011, pp. 1–6.

[26] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proc. 21st Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, New York, NY, USA, Jun. 2002, pp. 1567–1576.

[27] R. Liu, P. Sinha, and C. E. Koksal, "Joint energy management and resource allocation in rechargeable sensor networks," in *Proc. 29th Int. Conf. Comput. Commun., Joint Conf. IEEE Comput. Commun. Soc.*, San Diego, CA, USA, Mar. 2010, pp. 1–9.

[28] S. Mao, M. H. Cheung, and V. W. S. Wong, "An optimal energy allocation algorithm for energy harvesting wireless sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Ottawa, ON, Canada, Jun. 2012, pp. 265–270.

[29] S. Peng and C. P. Low, "Throughput optimal energy neutral management for energy harvesting wireless sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Paris, France, Apr. 2012, pp. 2347–2351.

[30] G. Sun, G. Qiao, and L. Zhao, "Efficient link scheduling for rechargeable wireless ad hoc and sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2013, no. 1, p. 223, Dec. 2013.

[31] Q. Chen, H. Gao, Z. Cai, L. Cheng, and J. Li, "Energy-collision aware data aggregation scheduling for energy harvesting sensor networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Honolulu, HI, USA, Apr. 2018, pp. 117–125.

[32] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[33] S. Ramanathan and E. L. Lloyd, "Scheduling algorithms for multihop radio networks," *IEEE/ACM Trans. Netw.*, vol. 1, no. 2, pp. 166–177, Apr. 1993.

[34] M. Razzaque and S. Dobson, "Energy-efficient sensing in wireless sensor networks using compressed sensing," *Sensors*, vol. 14, no. 2, pp. 2822–2859, Feb. 2014.

[35] J.-P. Vasseur and A. Dunkels, *Interconnecting Smart Objects With IP: The Next Internet*. San Mateo, CA, USA: Morgan Kaufmann, 2010.

[36] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," *Wireless Netw.*, vol. 11, no. 4, pp. 471–487, Jul. 2005.

[37] P. Wan, "Multiflows in multihop wireless networks," in *Proc. 10th ACM Int. Symp. Mobile Ad Hoc Netw. Comput. (MobiHoc)*, New Orleans, LA, USA, May 2009, pp. 85–94.

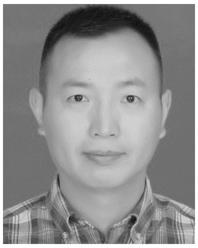
[38] J. Ma, W. Lou, Y. Wu, X. Li, and G. Chen, "Energy efficient tdma sleep scheduling in wireless sensor networks," in *Proc. 28th IEEE Int. Conf. Comput. Commun., Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, Rio de Janeiro, Brazil, Apr. 2009, pp. 630–638.

SIQI GUAN received the B.S. degree in Internet of Things engineering from the School of Computer Science and Technology, Anhui University of Technology, Ma'anshan, China, in 2018. She is currently pursuing the M.S. degree in computer technology with Hangzhou Dianzi University, Hangzhou, China.

JIANHUI ZHANG (Member, IEEE) received the B.S. degree in mechanotronics and the M.S. degree in fluid mechanics from Northwestern Polytechnical University, China, in 2000 and 2003, respectively, and the Ph.D. degree in control theory and engineering from Zhejiang University, Hangzhou, China, in 2008.

He is currently working as a Professor with the School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou. His research interests include algorithm design, wireless networks, and so on.

ZHENGBING SONG received the B.S. degree in communication engineering from the School of Information Science and Engineering, Chengdu University, Chengdu, China, in 2019. He is currently pursuing the M.S. degree in computer technology with Hangzhou Dianzi University, Hangzhou, China.



BEI ZHAO received the B.S. and Ph.D. degrees from Zhejiang University, Hangzhou, China.

He is currently a Lecturer with the School of Computer Science and Technology, Hangzhou Dianzi University, China. He has published over ten technical articles in journals. His research interests include high-performance embedded computing, digital signal processing, wireless sensor networks, mobile computing, and related aspects.

YANJUN LI (Member, IEEE) received the B.S. degree in automation and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2004 and 2009, respectively, and the Ph.D. degree in information science from Nancy University, Villers-lès-Nancy, France, in 2010.

She is currently an Associate Professor with the School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou. She has authored or coauthored over 50 research article in journals and conferences, including the IEEE Transactions on Vehicular Technology, the IEEE Internet of Things Journal, the IEEE Transactions on Industrial Informatics, the IEEE Communications Letters, the IEEE Network, INFOCOM, and WCNC. Her current research interests include battery-free networks, indoor localization, and the Internet of Things.