Charge-Aware Duty Cycling Methods for Wireless Systems under Energy Harvesting Heterogeneity

JIANHUI ZHANG, SIWEN ZHENG, TIANHAO ZHANG, and MENGMENG WANG, Hangzhou Dianzi University
ZHI LI, Stony Brook University

Recent works have designed systems containing tiny devices to communicate with harvested ambient energy, such as the ambient backscatter and renewable sensor networks. These systems often encounter the heterogeneity and randomness of ambient energy. Meanwhile, the energy storage unit, such as the battery or capacitor, has the inherent property of imperfect charge efficiency λ ($\lambda \leq 1$), which is usually low when the power of the ambient energy is weak or variable. These features bring new challenges in using the harvested energy efficiently. This article calls it the *stochastic duty cycling problem* and studies it under three cases—offline, online, and correlated stochastic duty cycling—to maximize utilization efficiency. We design an offline algorithm[1] for the offline case with optimal performance. An approximation algorithm with the ratio $1 - e^{-\gamma}$ is designed for the online case. By adding initial negotiation among devices, we present a correlated algorithm and prove its approximation ratio theoretically. Experiment evaluation on our real energy harvesting platform shows that the offline algorithm performs over the other two algorithms. The correlated algorithm may not perform over the online one under the impacts of the three metrics: heterogeneity, charge efficiency, and energy harvesting probability.

CCS Concepts: • **Computer systems organization** → **Sensor networks**; Embedded systems; • **Hardware** → *Renewable energy*;

Additional Key Words and Phrases: Stochastic duty cycling, energy harvesting platform, bipartite matching, wireless communication

ACM Reference format:
Jianhui Zhang, Siwen Zheng, Tianhao Zhang, Mengmeng Wang, and Zhi Li. 2020. Charge-Aware Duty Cycling Methods for Wireless Systems under Energy Harvesting Heterogeneity. *ACM Trans. Sen. Netw.* 16, 2, Article 15 (January 2020), 23 pages.
https://doi.org/10.1145/3372800

[1]This article shortens the term *offline stochastic duty cycling algorithm* to *offline algorithm*, and similarly for the online and correlated algorithms.

This work was supported by the general program of the National Natural Science Foundation of China (NSFC) under grants 61473109 and 61572164, the major program of the NSFC under grant 61190113, the Zhejiang Major Science and Technology Program under grant 2018C04012, and the Graduate Scientific Research Foundation of Hangzhou Dianzi University.
Authors' addresses: J. Zhang, S. Zheng, T. Zhang, and M. Wang, College of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, Zhejiang, 310018, China; emails: jh_zhang@hdu.edu.cn, siwenzheng@outlook.com, tianhao.hdu@gmail.com, mm1028_wang@163.com; Z. Li, Department of Computer Science, Stony Brook University, Stony Brook, NY 11794; email: zhili_pro@163.com.

1550-4859/2020/01-ART15 $15.00
https://doi.org/10.1145/3372800

1 INTRODUCTION

Energy harvesting and communication for tiny devices have been hot topics in recent years [Chen et al. 2017; Karaaslan et al. 2017; Liu et al. 2018; Zheng et al. 2019]. A wide range of devices, such as ambient backscatter [Darsena et al. 2017], ring [Karaaslan et al. 2017], and renewable sensor node [Chen et al. 2017, 2018; Shen et al. 2013], have been developed to harvest diverse ambient energy, such as TV signals [Darsena et al. 2017], Near Field Communication (NFC) signals from phones [Boada et al. 2018], and renewable energy [Chen et al. 2017]. With these tiny devices, the power of harvested energy is usually weak and limited [Shen et al. 2013]. To support long-term system operation with it, the duty cycle adjustment is an indispensable technique to allocate energy efficiently [Chen et al. 2017]. Furthermore, it varies with time and is heterogeneous among devices. A few works suggest the dynamic duty cycling according to its randomness [Tong and Pan 2017].

However, a few works consider the impacts of the imperfect charge efficiency and the heterogeneity of the ambient energy when adjusting the duty cycle in energy harvesting wireless systems. For example, the charge efficiency of the rechargeable battery for a solar-powered sensor node is often less than 75% [Ding et al. 2000], whereas the capacitor suffers from high leakage [Zhu et al. 2009]. The imperfect charge efficiency results in infrastructure energy consumption to store harvested energy inevitably. The ambient energy is heterogeneous (i.e., nonuniform over time and different among the devices). The previous works on dynamic duty cycling exhibit their limitation on the utilization efficiency of harvested energy since they consider only the randomness rather than the heterogeneity of harvested energy among devices and the imperfect charge efficiency. For example, if two neighboring devices u and v can wake up and harvest energy simultaneously, they need not store its harvested energy and thus waste energy on the imperfect charge efficiency. It thus requires a new technique to not only adjust the duty cycle but also determine the moments to wake up by considering the imperfect charge efficiency caused by randomness, heterogeneity of the harvested energy, and the limited and weak power. This article presents the new technique using the term *stochastic duty cycling*.

To improve the utilization efficiency of the harvested energy, this work proposes the stochastic duty cycling algorithms for three cases—offline, online, and correlation—by considering the imperfect charge efficiency that caused by randomness, heterogeneity of the harvested energy, and the limited and weak power. Under the offline case, the profile of harvested energy in each period is assumed to be known in advance. This assumption can be realistic when the profile in one period is the same as or quite similar to that in the previous periods. The offline algorithm serves as the benchmark to compare to the online and correlated cases. Under the online case, each device operates its duty cycling independently so as to save energy on message negotiation among nodes. Under the correlated case, each device can communicate its strategy with its neighbor at the beginning of each period, then adjusts its duty cycle independently. The goal of our study for the three cases is to maximize the utilization efficiency of the harvested energy, which is measured by the metric Common Active Time (CAT) in each period. Suppose that two devices u and v are scheduled to be active respectively in time sets s_u and s_v. The CAT between u and v is then the intersection $s_u \cap s_v$ of the two sets. The main contributions of this work are summarized as follows:

(1) This article proposes the new problem of how the charge efficiency caused by the randomness, heterogeneity of the harvested energy, and the limited and weak power affects its utilization efficiency under duty cycling.
(2) We design the offline algorithm for the offline case, which is proved to be optimal. The online algorithm is designed for the online case with the approximation ratio $1 - e^{-\gamma}$. For

the correlated case, we propose the correlated algorithm and give its approximation ratio in different cases.

(3) The experiment is conducted based on our real energy harvesting platform to evaluate the performance of the offline, online, and correlated algorithms. We extract three metrics—heterogeneity, charge efficiency, and energy harvesting probability (randomness)—from the experiment data to dissect their impacts on the three algorithms. Our analysis shows that the online and correlated algorithms have their own advantages under the three metrics.

Road map. The rest of this article is organized as follows. Section 2 reviews the related work on the energy harvesting systems and duty cycling techniques. Section 3 describes the motivation of this work. The model and the stochastic duty cycling problem are formulated in Section 4. We design the optimal solution for the offline case in Section 5 and the online and correlated algorithms in Sections 6 and 7, respectively. The experiment evaluation based on our real energy harvesting platform is conducted in Section 8. Section 9 concludes the article.

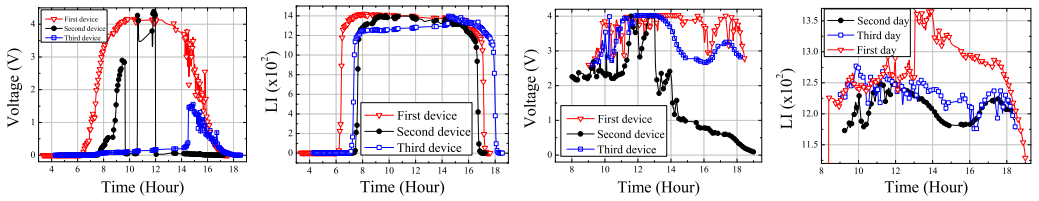
2 RELATED WORK

In recent years, many works have been devoted to energy harvesting and its management and applications [Darsena et al. 2017; Mehrabi and Kim 2016; Wang et al. 2013; Yang et al. 2015; Zhang et al. 2015, 2016]. This section reviews the related work on the energy harvesting system and the duty cycling technique.

Energy harvesting system. Tiny energy harvesting devices have been widely researched in recent years. Darsena et al. [2017] give a modeling and performance analysis about the ambient backscatter devices, which can communicate by backscattering ambient RF with the energy harvested from TV signals. Li et al. [2018] design a self-sustaining sensing system called *Trinity*. It harvests energy from the airflow produced by HVAC outlets to power the sensor nodes, which are then able to communicate with each other. Gummeson et al. [2014] design a ring form-factor wearable device for gesture input to harvest energy from an NFC-enabled phone held in a user's hand. Other typical systems are able to harvest solar, vibration, and wind energy [Bito et al. 2017; Habibzadeh et al. 2017]. Among the existing systems, some devices have no storage components and thus use the harvested energy directly [Huang and Neely 2013]. Some devices store the harvested energy in their batteries, whereas others take the capacitor as the primary buffer and the battery as the second buffer [Jiang et al. 2005]. The battery suffers from low charge efficiency and long charging duration, and the capacitor has high leakage. For example, the 2000F ultra-capacitor has a high leakage rate up to 43.8% during the first month [Zhu et al. 2009]. Although there are some works on the scheduling of energy networks, such as smart grid, their methods cannot be applied to ours directly.

These devices and systems can harvest a small amount of energy from the natural resources continually, which is not enough to support the full duty cycle. The harvested energy among devices or the systems is heterogeneous because of the spatio-temporal variation of the natural resources [Chang et al. 2017; Heesen and Madlener 2018]. Frequent charging/discharging suffers from much energy loss and harms the battery working life.

Duty cycling. In energy harvesting systems, the duty cycling technique remains necessary because of the limited hardware and insufficient intensity of the natural resources [Chen et al. 2017; Tong and Pan 2017]. Previous works on duty cycling can be classified into two groups: deterministic and stochastic duty cycling. The former one calculates the amount of active time in each period, such as the adaptive duty cycling [Tong and Pan 2017]. A few works consider not only the amount of active time but also the moments to wake up (i.e., stochastic duty cycling) [Ghidini



(a) Voltage of three devices. (b) LI of three devices. (c) Voltage in 3 days. (d) LI in 3 days.

Fig. 1. The energy harvested by different three devices in 3 days.

and Das 2011]. However, they do not consider that energy harvesting is a random process, and the energy profile of each sensor node is time varying and different among the devices [Zhu et al. 2009]. Furthermore, these previous works adjust the duty cycle only according to the predicted amount of harvested energy in a duration, and they do not consider the moments to be active since the low charge efficiency has a great impact on energy efficiency. Different from previous works, this work addresses stochastic duty cycling by determining not only the amount of active time but also the moments to be active over a period. We consider the imperfect charge efficiency caused by the phenomena about strong randomness, heterogeneous energy availability, and the weak and limited power of harvested energy, which has not yet been studied.

3 MOTIVATION

This section presents the analysis for the data collected with our energy harvesting platform, SolarMote [Zhang et al. 2015]. Its technical details will be presented in Section 8.1. Some phenomena are then observed from the data, including the strong randomness, heterogeneous energy availability, and the weak and limited power of harvested energy. They result in the low charge efficiency.

3.1 Randomness and Heterogeneous Energy Availability

The randomness and heterogeneity of harvested energy appear quite often when the platform is deployed outdoors [Chang et al. 2017; Heesen and Madlener 2018]. For example, solar energy can be harvested during the day and needs to be stored partially to support operation during the night. Unpredictable environmental factors, such as the shadowing of a tree, building, or cloud, lead to diverse availability profiles of the harvested energy as illustrated in Figure 1(a) and (b). One device may have different energy availability profiles over different periods even under similar weather conditions as illustrated in Figure 1(c) and (d). This observation is defined as the randomness of harvested energy, which has been modeled in previous models as either an *i.i.d.* [Huang and Neely 2013] or Markov process [He et al. 2012]. Furthermore, as shown in Figure 1(a) and (b), the energy profiles for different devices vary significantly over the same period. Therefore, different devices may have diverse chances to harvest energy because of shadowing. This observation is defined as the heterogeneity among devices. A device wastes energy if it cannot wake up with its neighbor simultaneously.

3.2 Power Limitation and Weakness

The stored power is limited and weak because of the randomness and heterogeneity of harvested energy availability under unstable and weak ambient light conditions [Shen et al. 2013]. The voltage to support the normal operation of a tiny device, such as SolarMote, is usually about 3 V. In a good solar condition, the time to harvest energy is about 7 hours as shown in Figure 1(a) and (b). It can be much worse, however, as shown in the second and third devices of harvested energy. Thus, harvested energy is limited to support device operation for a full day. The power of har-

vested energy is weak because its voltage is about 4 V or less as shown in Figure 1(c) and (d). The voltage value of harvested solar power has significant variation in different periods between different devices. Sometimes the voltage of harvested solar power is quite weak. The voltage value is between $[0, 1]$V for most of the day as shown in Figure 1(a) and (c). Especially in Figure 1(a), the voltage value of harvested energy of the third device is close to 0. The weakness of harvested energy makes the amount of it limited even further.

3.3 Low Charge Efficiency

Charge efficiency can be very low because of hardware limitations or the power weakness and randomness of harvested energy. The most popular energy storage units in tiny energy harvesting devices are the battery and capacitor. The charge efficiency of a rechargeable battery for a solar-powered sensor node is often less than 75% [Ding et al. 2000], whereas the capacitor suffers from high leakage. The randomness of harvested energy decreases the chance to charge the battery because of its lower power, which needs a node to switch frequently between storing the harvested energy and using battery energy. The weak harvested energy results in even worse charge efficiency.

Notice that devices can communicate with each other only when they are active simultaneously. Based on the preceding observations, it is beneficial for utilization efficiency of harvested energy to use it directly rather than store it first under the low charge efficiency. However, the randomness of energy availability often prevents the devices from using harvested energy directly. The heterogeneity prevents devices from making decisions with the statistics of historical information. Such difficulties make it quite challenging to plan active time slots for devices. These observations motivate us to design new duty cycling algorithms to improve the utilization efficiency of harvested energy by considering factors including randomness, heterogeneity of the energy availability, weak and limited power, and imperfect charge efficiency.

4 PRELIMINARIES

This section provides the energy harvesting and system models, presents the study objective, and then formulates the stochastic duty cycling problem.

4.1 Model

In the energy harvesting system, each device can harvest ambient energy to communicate with its neighbor. It can store its harvested energy with charge efficiency λ or use it directly. To improve energy harvesting efficiency, most energy harvesting systems fix a switch among the ambient power source, the reservoir capacitor, and the battery instead of connecting the former one directly to either of the latter two [Shen et al. 2009]. Thus, by controlling the switch, each device can wake up by using the stored or harvested energy to support its normal communications. Each device can choose any one of two resources—stored or harvested energy—to support its normal operation by controlling the switch. This article focuses on the usual case that the power of the ambient energy cannot support simultaneous operation of communication and battery charging—for instance, the harvested consumption power is not higher than the energy consumption power ρ. In each period, each device can harvest energy with probability γ. The chance to harvest energy could be arbitrary and nonuniform.

Let T denote the period and θ_u denote the energy availability state of u, where $\theta_u(\tau) = 1, \tau \in T$, if u can access energy sources with sufficient intensity at time slot τ, and $\theta_u(\tau) = 0$ otherwise. Assume that each device can harvest one unit of energy at each time slot τ ($\tau \in T$) with probability γ. The harvested energy is relatively weak, and thus we can set $\rho \geq 1$. Let $\alpha_u(\tau)$ denote the decision of device u at time slot τ. If u is active, $\alpha_u(\tau) = 1$ and 0 otherwise. Let $r_u(\tau)$ denote u's residual

stored energy until the end of slot τ. Combining with the harvested energy and previous decisions, we can calculate $r_u(\tau)$ by the following equation:

$$r_u(\tau) = \sum_{t=1}^{\tau} \{\lambda\theta_u(\tau)[1-\alpha_u(\tau)] - \beta_u(\tau)[1-\theta_u(\tau)]\alpha_u(\tau)\}. \tag{1}$$

When $\theta_u(\tau) = 1$ and $\alpha_u(\tau) = 0$, device u can remain asleep and store λ units of energy. When $\theta_u(\tau) = 1$ and $\alpha_u(\tau) = 0$, device u has to consume its residual stored energy to wake up and keep active in time slot τ. The consumed energy of the devices u and v at time slot τ is denoted by $\beta(\tau)$. $\beta(\tau)$ is given by the following equation:

$$\beta(\tau) = \begin{cases} 0, & if \quad \alpha(\tau) = 0, \\ 1, & if \quad \alpha(\tau)\cdot\theta(\tau) = 1, \\ min\{1, r_u(\tau-1)\}, & otherwise. \end{cases} \tag{2}$$

Device u will not consume energy if it decides to remain asleep in time slot τ. It will consume one unit of harvested energy if it decides to wake up and the ambient energy is available. However, if it decides to wake up and ambient energy is not available, it will consume the residual stored energy. Notice that this work considers the point-to-point (P2P) communication model (i.e., one device communicates with the other). Thus, we can assume that a pair of neighboring devices can have the same energy consumption power: $\rho_u = \rho_v$. Let $\rho = \rho_u = \rho_v$ ($\rho \geq 1$). The P2P model is very useful in wide applications, such as communication in the ambient backscatter [Darsena et al. 2017] and the neighboring nodes of sensor networks [Zhang et al. 2016]. For other models, such as tree-like topology and the energy consumption model, we provide some discussion in Section 9.

4.2 Objective

Suppose that a period T is divided into some equal time slots $\tau = 1, 2, \ldots, |T|$. Denote the set of device u's active time slots by s_u, where $s_u \in T$. The duty cycle is the ratio of active time to the period length, such as $|s_u|/|T|$. The technique of duty cycle adjustment in a previous work [Chen et al. 2017] is to adjust the ratio. To differ from the duty cycle adjustment, this work introduces the concept of duty cycling, which finds a set of active time slots, such as s_u, in the given period for each device so as to improve energy efficiency. This section thus defines the metric, CAT, as follows:

Definition 1 (Common Active Time). Given the neighboring devices u and v, and their sets s_u and s_v of active time slots in period T, their CAT $I(T)$ in the period T is the cardinality of the intersection $s_u \cap s_v$ of two sets.

The CAT is determined by the energy availability state and the decision of each device. The CAT between u and its neighbor v is calculated as follows. When $\theta_u(\tau)\cdot\theta_v(\tau) = 1$, named *condition one* (c_1), and $\alpha_u(\tau)\cdot\alpha_v(\tau) = 1$, named *condition two* (c_2), the CAT is set to be $1/\rho$ (i.e., $I(\tau) = 1/\rho$). When $\alpha_u(\tau)\cdot\alpha_v(\tau) = 0$, named *condition three* (c_3), the CAT is set to be 0. When $\theta_u(\tau)\cdot\theta_v(\tau) = 0$, named *condition four* (c_4), and $\alpha_u(\tau)\cdot\alpha_v(\tau) = 1$, at least one of devices u and v has to use the residual energy. Thus, $I(\tau) = \min\{1, r_v(\tau)/\rho, r_u(\tau)/\rho\}$, where $r_v(\tau) \geq 0$ and $r_u(\tau) \geq 0$. Therefore, the CAT at each time slot τ is given by the following equation:

$$I(\tau) = \begin{cases} 1/\rho, & if \quad c_1 \& c_2, \\ 0, & if \quad c_3, \\ \min\{1, r_v(\tau)/\rho, r_u(\tau)/\rho\}, & if \quad c_2 \& c_4, \end{cases} \tag{3}$$

where c_1, c_2, c_3, and c_4 are four conditions given in the previous content. Each device must determine a sequence of decisions $\alpha(\tau)(\forall\tau \in T)$ so that the expected overall CAT $\mathbb{E}(I(T)) = \Sigma_{\tau=1}^{T}\mathbb{E}(I(\tau))$

Table 1. Symbol and Meaning

Symbol	Description	Symbol	Description	Symbol	Description
T	Period	τ	Slot	u, v	Devices
r	Stored energy	U, V	Vertex sets	$\mathbb{E}$	Expectation
E	Edge set	G	Graph	λ	Charge efficiency
I	CAT	θ	Energy state	C	Synchronous edge set
α	Device decision	p	Probability	D	Asynchronous edge set
c	Condition	n	No. of balls	A	Bin subset
β	Power	ρ	Consumed energy in one slot	S	No. of nonempty bins
m	No. of bins	s	Active slot set		
k, l	Index	γ	Energy harvesting probability		

in T can be maximized. In a period T, each node needs to make a sequence of decisions to sleep or to be active with its random and heterogeneous harvested energy while the charge efficiency $\lambda < 1$. The objective of this work is to maximize the overall CAT in each period T. With Equation (3), the stochastic duty cycling problem can be formulated as the following optimization problem:

$$\max \quad \mathbb{E}(I(T)), \tag{4}$$

$$\text{s.t.} \quad \beta(\tau) \leq \theta(\tau - 1) + [1 - \theta(\tau)]r(\tau), \quad \forall \tau \in T; u, v, \tag{5}$$

$$\theta(\tau) \in \{0, 1\} \quad \forall \tau \in T, \tag{6}$$

where the first constraint in Equation (5) indicates that the amount of consumed energy should less than the harvested energy when the ambient energy is available or less than the residual stored energy when it is not available. Since we can define the new slot and period as τ/ρ and T/ρ, the following context omits the scale ρ for the succinct presentation. Most symbols used in this article are summarized in Table 1.

4.3 Energy State Graph

This work develops the *energy state graph*, denoted by $G(U, V, E)$, to describe the states $\theta_u(\tau)$ and $\theta_v(\tau)$, $\forall \tau = 1, \ldots, T$, of the ambient energy available to device u and its neighbor v, and all feasible decisions $\alpha_u(\tau)$ and $\alpha_v(\tau)$, $\forall \tau = 1, \ldots, T$. In the energy state graph, the time slot at which a device can harvest energy is called a *vertex*. U and V represent the vertex sets of devices u and v, respectively. E represents the set of edges among the vertices respectively from the sets U and V, whose construction will be described in the next section. The graph characterizes the timing and interaction relations between the states of harvested energy of neighboring devices and thus is a bipartite graph.

Let the red circle in Figure 2 denote the vertex. For example, the red vertex $\tau_1(u)$ indicates that u can harvest ambient energy at time slot τ_1 (i.e., $\theta_u(\tau_1) = 1$). Each device can consume harvested energy directly or store it. Correspondingly, each vertex of device u may connect to any vertex of its neighbor v as shown in Figure 2. If two vertices come from two vertex sets, respectively, such as $\tau_1(u) \in U$ and $\tau_1(v) \in V$ in Figure 2, and appear at the same time slot, they are called *synchronous vertices*. Otherwise, they are called *asynchronous vertices*—that is, at least one device uses the stored energy to establish a link with the neighboring device. If devices u and v are active at the time slot of synchronous vertices, the edge connecting the synchronous vertices is called a *synchronous edge*. Otherwise, it is called an *asynchronous edge*, which indicates that an earlier vertex should wait for a latter one. Therefore, devices v and u could wake up at the same slot. For example, the asynchronous edge $(\tau_4(v), \tau_5(u))$ indicates that v harvests the energy (i.e., the

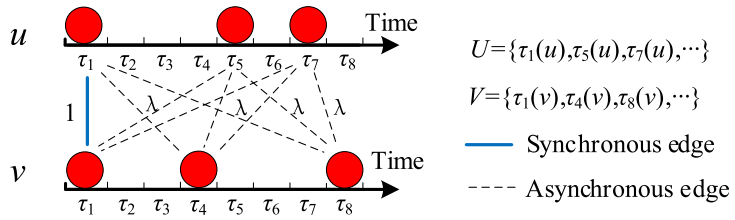


Fig. 2. Energy state graph $G(U, V, E)$. u and v are neighboring devices.

vertex $\tau_4(v)$) at slot τ_4 and wakes up by consuming it at τ_5. At the same time slot τ_5, u wakes up by consuming the directly harvested energy (i.e., $\tau_5(u)$). If one device does not wake up at any time slot, no edge is established. Figure 2 shows all possible edges. The properties analyzed in the following context illustrate that some edges are not feasible.

In the case of synchronous edge, the harvested energy is directly consumed, so no energy is wasted. However, some energy is wasted in the case of an asynchronous edge because of low charge efficiency. By recalling the assumption in Section 4.1, each device can harvest one unit of energy at each single time slot. The CAT of each synchronous edge is 1, so we assign it with weight 1. Similarly, each asynchronous edge is assigned weight λ. The example in Figure 2 illustrates these weight properties. The synchronous edges $(\tau_1(u), \tau_1(v))$ are assigned weight 1. The asynchronous edges, $(\tau_1(u), \tau_4(v))$, $(\tau_5(u), \tau_1(v))$, and $(\tau_5(u), \tau_4(v))$, and so on, are assigned weight λ. We have the following property.

Property 1. *All synchronous edges have the same weight: 1. All asynchronous edges have the same weight: λ.*

Notice that the energy harvested in one time slot can support the device's communication for at most one slot. Accordingly, each vertex can connect one feasible edge at most. We have the following property.

Property 2. *A feasible duty cycling schedule can choose at most one edge incident to each vertex.*

Thus, the edges $(\tau_5(u), \tau_1(v))$ and $(\tau_5(u), \tau_4(v))$ cannot exist simultaneously in the example of Figure 2 according to Property 2. Recall that each device can harvest energy at each time slot with probability γ. Accordingly, the probability that device u has a synchronous or an asynchronous edge with device v is γ at each time slot as well, since each vertex of device u needs to connect with one of device v. The probability that there is a synchronous edge between a pair of devices at each time slot is γ^2. Thus, the probability that each device has an asynchronous edge at each time slot is $\gamma - \gamma^2$. We therefore have the following property.

Property 3. *The probabilities to have synchronous and asynchronous edges at each time slot respectively are γ^2 and $\gamma(1-\gamma)$ between a pair of neighboring devices.*

5 OFFLINE STOCHASTIC DUTY CYCLING

Under the offline case, each device has complete knowledge of the harvested energy of its own and its neighbor. The expected numbers of synchronous and asynchronous edges in Property 3 can be known in advance. First, we construct an energy state graph for each device. The overall CAT is the sum of a subset of edges of the energy state graph, which is selected by a duty cycling algorithm. The optimal algorithm is to find such a subset of edges so that the overall weight of the edges is maximized. In the energy state graph in Figure 2, each vertex in U may connect with any vertex in V and vice versa. Each edge can be taken as a matching between two vertices that it connects with. Thus, we have the following claim.

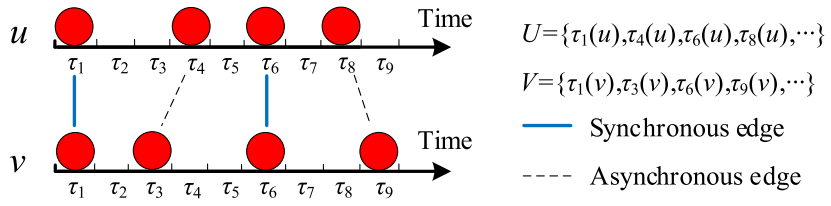


Fig. 3. Offline duty cycling graph $G(U, V, E_f)$.

Claim 1. *The optimal algorithm for the offline duty cycling problem is equivalent to finding the maximum weighted bipartite matching between two sets U and V in the energy state graph.*

Since there are two kinds of weights—1 and λ—it is easy to find the maximum weighted bipartite matching in the energy state graph with Properties 1 and 2. The example in Figure 3 illustrates the way to find the maximum weighted matching between the sets U and V, which leads to a new graph, named the *offline duty cycling graph*. Recall Property 2 in which each vertex can only connect with one edge, so we can construct the offline duty cycling graph by two steps. In the first step, all synchronous edges are selected because the synchronous edge has higher weight than the asynchronous one according to Property 1. The selection of the synchronous edges, such as $(\tau_1(u), \tau_1(v))$ and $(\tau_6(u), \tau_6(v))$ in Figure 3, corresponds to the decision making, $\alpha_u(\tau_1) = \alpha_v(\tau_1) = 1$ and $\alpha_u(\tau_6) = \alpha_v(\tau_6) = 1$. In the second step, the asynchronous edges are chosen. The vertex $\tau_3(v)$ searches one vertex in U to add an asynchronous edge. $\tau_3(v)$ can only search the vertices in U, which appears later than $\tau_3(v)$, since the device can use the energy only after it is harvested and stored. According to Property 1, $\tau_3(v)$ can select the vertex $\tau_4(u)$ so the asynchronous edge $(\tau_4(u), \tau_3(v))$ is established—that is, making decisions $\alpha_v(\tau_3) = 0$, $\alpha_u(\tau_4) = \alpha_v(\tau_4) = 1$. For each asynchronous edge $(\tau_k(u), \tau_l(v))$, its selection means that u and v make the decisions to be active at τ_k if $l < k$ by Rule 1. The offline algorithm is summarized in Algorithm 1.

Rule 1. *For each asynchronous edge $(\tau_k(u), \tau_l(v))$, u sleeps and stores its harvested energy at slot τ_k and then wakes up at τ_l if $k < l$. Otherwise, it wakes up at τ_k.*

We show that the offline algorithm given in Algorithm 1 is optimal by Theorem 1.

Theorem 1 (Optimality). *The edge set E_f of the offline duty cycling graph is the maximum weighted bipartite matching between the two sets U and V.*

ALGORITHM 1: Offline Duty Cycling Algorithm

Input: The harvested energy states of devices u and v: U and V;
Output: The decision sequence $\alpha(\tau)$, $\tau \in T$, for each device, and the edge set E_f.

1: Construct the energy state graph $G(U, V, E)$ based on U and V;
2: Find all synchronous edges in E and add them into E_f;
3: Remove all edges linked to the synchronous vertices from E;
4: **while** $E \neq \varnothing$ **do**
5: Select an arbitrary edge $(\tau(u), \tau(v))$ from E and add it into E_f;
6: Remove all edges linked to the vertices $\tau(u)$ and $\tau(v)$;
7: **end while**
8: At each time slot τ, $\tau \in T$, corresponding to synchronous edge, let $\alpha(\tau) = 1$ for both the devices u and v;
9: At all time slots corresponding to an asynchronous edge, each device makes a decision according to Rule 1. If a sleeping decision is made, $\alpha(\tau) = 0$ and 1 otherwise.

Proof. We prove the theorem by contradiction. Assume that the optimal algorithm leads to the maximum weighted matching, denoted by E_p. The overall weight of all edges in E_p must be no less than that in E_f, where E_f is obtained by Algorithm 1. Since each edge has two kinds of weights—1 and λ—we divide the edge set into two subsets: E^1 with weight 1 and E^λ with weight λ. Accordingly, $E_f = E_f^1 \cup E_f^\lambda$ and $E_p = E_p^1 \cup E_p^\lambda$. According to the assumption, we must have $|E_p^1| > |E_f^1|$ or $|E_p^\lambda| > |E_f^\lambda|$, where $|\cdot|$ is the set cardinality. For the case $|E_p^1| > |E_f^1|$, there is at least one another synchronous edge, which is not included in E_f^1. But it contradicts Algorithm 1 since the algorithm searches all synchronous edges and adds them into E_f first in Step 2. Therefore, we have $|E_p^1| = |E_f^1|$. For the case $|E_p^\lambda| > |E_f^\lambda|$, E_p must contain more edges than E_f. Let $(\tau(u), \tau(v))$ denote an arbitrary edge that is in E_p but not in E_f. The edge $(\tau(u), \tau(v))$ must also be in the set E. If $(\tau(u), \tau(v))$ is selected by Step 5 of Algorithm 1, it is in E_f. It leads to contradiction. If $(\tau(u), \tau(v))$ is not selected by Step 5 of Algorithm 1, the vertex ($\tau(u)$ or $\tau(v)$) must connect to another vertex, such as $\tau(v))'$ or $(\tau(u)'$. It creates a new edge in E_f, which leads to contradiction. Therefore, we can prove the case $|E_p^\lambda| = |E_f^\lambda|$, and the offline algorithm is optimal. □

Although the energy is harvested with probability, the energy state of each device is assumed to be known in advance in the offline case. According to Property 3, the maximal CAT is given by the following theorem.

Theorem 2 (Maximal CAT). *The expected maximal CAT over the period T by Algorithm 1 is $|T|\gamma[\gamma + \lambda(1-\gamma)]$ per device.*

Proof. By recalling the assumption in Section 4.1, each device can harvest energy in each time slot with the charge efficiency γ. The expected number of synchronous edges over the period is $\sum_{\tau_i=1}^{T} \gamma^2$. The expected number of asynchronous edges over the period is $\sum_{\tau_i=1}^{T} \gamma(1-\gamma)$. Thus, the expected overall CAT is $\mathbb{E}(I(T)) = |T|\gamma[\gamma + \lambda(1-\gamma)]$. □

There are some existing algorithms to find the maximum weighted bipartite matching [Madry 2013]. Their complexity is quite high. In this work, the time complexity of Algorithm 1 is not as high as in previous works. In Step 1 of Algorithm 1, each device costs $|T|$ steps to identify the harvested energy state of each time slot. In Step 2, it takes at most $\max\{|U|, |V|\}$ steps to establish synchronous edges among the vertices. From Step 4 to Step 7, it takes at most $|U||V|$ steps to establish asynchronous edges among the vertices. Therefore, the overall time complexity is $\max\{|T|, |U||V|\}$.

Theorem 3 (Time complexity). *The time complexity of Algorithm 1 is $O(\max\{|T|, |U||V|\})$.*

6 ONLINE STOCHASTIC DUTY CYCLING

This section describes the online stochastic duty cycling problem, and presents the online algorithm and its theoretical performance.

6.1 Problem Description

Under the online case, each device and its neighbor have no future information of the profiles of their harvested energy in advance. Accordingly, the vertices in both sets U and V arrive online in the energy state graph. Before the arrival of each vertex, each device must independently make an online decision to be active or sleep beforehand. The challenging points are tripartite: (1) each device does not know the future energy state of its own and its neighbor beforehand, (2) each device must make decision independently according to its own energy harvesting probability, and (3) it is not accurate to make a decision with the historical information because of the heterogeneity

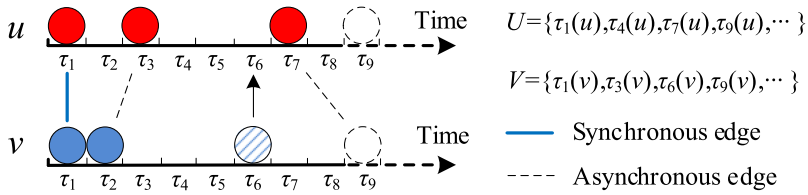


Fig. 4. Online duty cycling graph $G(U, V, E_o)$. Current time slot is τ_8.

of the harvested energy. The online duty cycling graph in Figure 4 illustrates an example to show the challenging points. In this figure, the current time slot is τ_8. The dashed circles $\tau_9(u)$ and $\tau_9(v)$ denote the possible coming vertices, which are unknown for both u and v. u is not sure whether $\tau_9(u)$ will arrive or not since it is random. u cannot estimate the arrival probability of $\tau_9(u)$ since it cannot count the probability according to the arrival of $\tau_1(u)$ and $\tau_3(u)$, which arrive unevenly. But u must decide to be active or not at time slot τ_9. If $\tau_9(u)$ and $\tau_9(v)$ arrive and only one of u and v is not active, the energy $\tau_9(u)$ or $\tau_9(v)$ is wasted. The online stochastic duty cycling problem is how each device makes a sequence of online decisions independently so that the overall CAT in each period can be maximized.

6.2 Algorithm Design

By analyzing the online duty cycling graph in Figure 4, we find that the vertices $\tau(u) \in U$ and $\tau(v) \in V$ arrive in arbitrary orders. Each device has to make online decision to be active or not for next time slot to establish weighted edges. The online duty cycling problem is similar to the online weighted bipartite matching problem but more difficult. The idea of the online algorithm is to establish the weighted edges by waking up each device with certain probability p_o at each time slot. If the decision is active at time slot τ, then $\alpha(\tau) = 1$ and $\alpha(\tau) = 0$ otherwise. We summarize the online duty cycling algorithm in Algorithm 2, where the decision probability p_o will be analyzed in the next section. In this algorithm, the decision sequence of each device is implemented by the second and third steps. Other steps help to find the edge set E_o and are not needed for each device's decision.

6.3 Theoretical Performance

The theoretical performance of Algorithm 2 can be analyzed by mapping as an extension of the typical balls-in-bins problem [Raab and Steger 1998]. In the typical balls-in-bins problem, n balls are thrown into m bins randomly. Suppose that a vertex of device u in an energy state graph is a ball and a vertex of device v is a bin. Since the ambient energy arrives randomly, balls are thrown

ALGORITHM 2: Online Duty Cycling Algorithm

Input: The probability γ and the period T.
Output: The decision sequence and edge set E_o.

1: **for** $\tau = 1, \ldots, T$ **do**
2: At $\tau - 1$, each device u makes a decision $\alpha_u(\tau) = 1$ with probability p_o and $\alpha_u(\tau) = 0$ otherwise;
3: If C_1 and C_2 are satisfied, add the edge $(\tau(u), \tau(v))$ to E_o and assign it with weight 1;
4: If C_1 and C_3 are satisfied, add the edge $(\tau'(u), \tau''(v))$ to E_o and assign it with weight λ, where τ' and τ'' $(\tau', \tau'' \leq \tau)$ are two different time slots and indicate that u or v uses the energy stored at slot τ' or τ'' respectively.
5: If $\alpha_u(\tau) = 0$ and $\theta_u(\tau) = 1$, u stores harvested energy.
6: **end for**

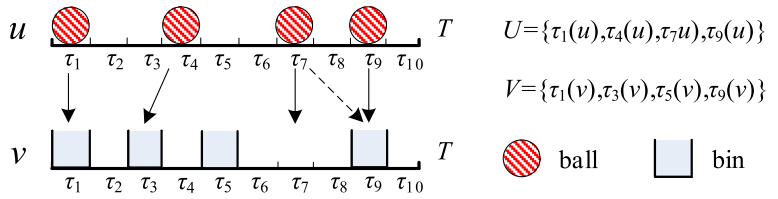


Fig. 5. Map to balls-in-bins problem.

into bins randomly. Different from the typical balls-in-bins problem, both balls and bins appear randomly in this work. By Algorithm 2, each device has two kinds of states—sleep and active—depending on their decisions and available energy. The possible edge connection, correspondingly the balls are thrown into bins, includes the following cases:

(1) If u and v are active and have available energy at the same time slot, we say that a ball is thrown into an empty bin. For example, the ball $\tau_1(u)$ is thrown into the bin $\tau_1(v)$ and the ball $\tau_4(u)$ is thrown into the bin $\tau_3(v)$ in Figure 5. Two edges are consequently connected.

(2) If u or v has unavailable energy at the same time slot, we say no ball or no bin. For example, u cannot harvest energy at time slot τ_2 and thus has no ball. v cannot harvest energy at τ_2 and thus has no bin.

(3) If only one of u and v is active at the same time slot and has available energy, we say there is an empty bin or a missing-thrown ball, such as the empty bin $\tau_5(v)$ and the ball $\tau_7(u)$ in Figure 5. We treat the missing-thrown ball as the ball thrown into a nonempty bin. For example, the missing-thrown ball $\tau_7(u)$ is taken to be thrown into the nonempty bin $\tau_9(v)$ in Figure 5.

For the newly defined extension of the balls-in-bins problem, we concern the nonempty bins containing at least one ball because this kind of bin represents the edge in the online duty cycling graph. Notice that if a nonempty bin contains more than one ball, the CAT of the bin is not accumulated according to Property 2. For example, if bin $\tau_9(v)$ contains two balls $\tau_7(u)$ and $\tau_9(u)$, the CAT is still 1. The probability of the number of nonempty bins can be given by the following fact by using Azuma's inequality on appropriately defined Doob martingales. The extension is given in the following fact.

Fact 1. *Suppose that there are n balls and m bins. The balls are thrown into bins randomly. Each ball may not be thrown and kept with the probability* $1 - p_o$. *Let A be a certain subset of bins, and let S be a random variable that equals the number of bins from A with at least one ball. With probability at least* $1 - 2e^{-\varepsilon^2 m/2}$, *for any* $\varepsilon > 0$, *we have:*

$$|A|[1 - e^{-\frac{n}{m}}] - \varepsilon p_o n \leq \mathbb{E}(S). \tag{7}$$

Proof. The probability that each bin has at least one ball is $1 - (1 - \frac{p_o}{m})^n$, so the expectation of S is $\mathbb{E}(S) = \sum_{a \in A}[1 - (1 - \frac{p_o}{m})^n]$. Since $(1 - \frac{1}{m})^n > 1/e$, we have the following equation with high probability:

$$\sum_{bin \in A} \left[1 - e^{-\frac{np_o}{m}}\right] \leq \mathbb{E}(S). \tag{8}$$

Notice that S is the function of the placements of the $p_o n$ balls. It satisfies the 1-Lipschitz condition since each ball can change the number of empty bins by no more than 1. We may apply Azuma's inequality to the Doob martingale (i.e., the changed number of empty bins after each ball placement is no more than 1) and obtain the probability inequality: $p(|S - \mathbb{E}(S)| \geq \varepsilon m) \leq 2e^{-\varepsilon^2 m/2}$. □

The next theorem presents the theoretical performance of Algorithm 2 by approximation analysis. An algorithm for a maximization problem is said to be an a-approximate algorithm if the

solution of the algorithm is more than the factor a times of the optimal algorithm solution. The factor a is called the *approximation ratio*.

Theorem 4. *The online algorithm given in Algorithm 2 has the approximation ratio* $1 - e^{-\gamma p_o}$, *where* γ *is the probability able to harvest energy.*

Proof. Algorithm 2 results in the edge set E_o. Let $E_o = C + D$, where sets C and D contain the synchronous and asynchronous edges, respectively. By recalling that the energy harvesting probability is γ, each device can harvest energy at $n = \gamma m$ time slots on average i.e., n vertices), where $m = |T|$. Suppose that there are two devices u and v. Each of them has m bins. Among these bins, there are n bins at which u and v can harvest energy. The edge can be established when the ball is thrown into one of these n bins. Therefore, among the m bins, there are γn and $(1-\gamma)n$ bins on average that are able to establish the synchronous and asynchronous edges, respectively. In Algorithm 2, each device wakes up with probability p_o (i.e., has $p_o n$ balls on average). To bound the total number of edges in set C, we have a balls-in-bins problem with $p_o n$ balls and n out of m bins. We are interested in lower bounding the number of bins that have at least one ball. Applying the result in Fact 1, we obtain that $C \geq [1 - e^{-\frac{p_o n}{m}}]\gamma n - \varepsilon m$ with probability $1 - e^{-\Omega(n)}$. Similarly, we can obtain that $D \geq [1 - e^{-\frac{p_o n}{m}}]n' - \varepsilon m$ with probability $1 - e^{-\Omega(n')}$, where $n' = (1-\gamma)n$. Thus, the weight of the set E_o is $I_o(T) = |C| + \lambda|D| \geq (1 - e^{-p_o\gamma})(\gamma n + \lambda n') - 2\varepsilon m$. Since $\gamma \leq 1$, $(1 - e^{-p_o\gamma})(\gamma n + \lambda n') - 2\varepsilon m \geq (1 - e^{-p_o\gamma})\gamma(n + \lambda n') - 2\varepsilon m$.

Let E_{OPT} denote the edge set obtained by the optimal algorithm. $E_{OPT} = C_{OPT} + D_{OPT}$, where set C_{OPT} contains the edges with weight one and set D_{OPT} contains the edges with weight λ. The device and its neighbor have $m\gamma^2$ common time slots to harvest energy in expectation and thus have $m\gamma^2$ edges with weight one in expectation (i.e., $C_{OPT} = m\gamma^2 = n\gamma$). Similarly, it has expected $m(1-\gamma)\gamma$ edges with weight λ (i.e., $D_{OPT} = m(1-\gamma)\gamma = \gamma n'$). Thus, the weight of set E_{OPT} is $I_{OPT}(T) = |C_{OPT}| + \lambda|D_{OPT}| = \gamma^2 m + 2\lambda(1-\gamma)\gamma m = \gamma(n + \lambda n')$.

By summarizing the preceding analysis, we can obtain that the overall CAT (i.e., the overall edge weight) obtained by Algorithm 2 is $I_n(T)/I_{OPT}(T) = (|C_n| + \lambda|D_n|)/(|C_{OPT}| + \lambda|D_{OPT}|) \geq 1 - e^{-p_o\gamma} - 2\varepsilon m/(|C_{OPT}| + \lambda|D_{OPT}|)$. We can take the value of ε to be sufficiently small, and then the approximation ratio of Algorithm 2 is $1 - e^{-\gamma p_o}$. □

From the preceding theorem, it is easy to find that the performance of the online algorithm is affected by the parameter p_o. When $p_o = 1$, the theoretical performance of Algorithm 2 reaches its best value.

Lemma 1. *Algorithm 2 reaches its best performance with the approximation ratio* $1 - e^{-\gamma}$, *when the active probability* $p_o = 1$.

Figure 6 shows the numerical evaluation to verify the preceding lemma. The CAT by the online algorithm increases with the active probability and the energy harvesting probability, and reaches its maximal value for each energy harvesting probability when the active probability $p_o = 1$, as shown in Figure 6.

7 CORRELATED STOCHASTIC DUTY CYCLING

This section presents the correlated algorithm, under which each device makes decision independently after the initial negotiation. Its theoretical performance is close to the optimal one and better than the online algorithm.

7.1 Correlated Duty Cycling Algorithm

The advantage of the online algorithm is that devices can make decisions independently. To keep the advantage and improve the performance of the online algorithm, this section presents the

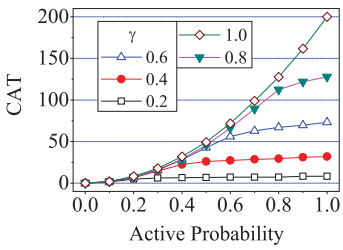


Fig. 6. Performance of the online algorithm. $\lambda = 0.8$, and T contains 200 slots.

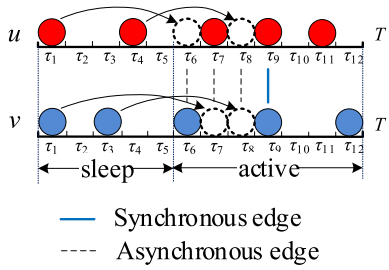


Fig. 7. Correlated duty cycling graph.

correlated algorithm, by which all devices negotiate at the beginning of each period and then make decisions independently. Figure 7 shows an example of the correlated algorithm. u and v sleep in the first five slots and store harvested energy. After slot τ_5, the stored energy supports devices' waking up at the time slots when they cannot harvest energy. u stores the energy harvested at τ_1 and τ_4, and consumes them at τ_6 and τ_8. v stores the energy harvested at τ_1 and τ_3, and consumes them at τ_7 and τ_8. They can establish edges at τ_6, τ_7, and τ_8 so the energy corresponding to vertices $\tau_7(u)$ and $\tau_6(v)$ can be fully explored. Let variable $p_s, p_s \in [0, 1]$, be the ratio to sleep in each period. A basic fact is that the energy can be stored or support device operation only after it is harvested. In summary, the correlated algorithm works as follows:

> Each device negotiates the value of p_s with its neighbor at the beginning of each period T, and stays asleep to store the harvested energy in the first $p_s|T|$ time slots and wakes up with the support of stored or harvested energy in the rest time slots of T. When there is no available ambient energy to harvest in the rest time slots, devices can wake up with the support of stored energy until it is exhausted.

The key problem is to determine the value of p_s so that the overall CAT in each period can be maximized. Notice that $\gamma p_s|T|$ units of energy can be stored during the first $p_s|T|$ time slots on average. The energy can establish $\gamma p_s|T|$ asynchronous edges in the rest time slots. In the rest $(1 - p_s)|T|$ time slots, each device keeps active and can harvest $\gamma(1 - p_s)|T|$ units of energy on average. Among them, there are $\gamma^2(1 - p_s)|T|$ synchronous on average by the concept of the energy state graph in Section 4.3. These asynchronous vertices can establish asynchronous edges with the support of the energy stored in the first $p_s|T|$ slots. For example, u stores energy $\tau_1(u)$ and $\tau_4(u)$ in Figure 7. The stored energy $\tau_1(u)$ and asynchronous vertex $\tau_6(v)$ establish an asynchronous edge at τ_6. Meanwhile, both the stored energy $\tau_4(u)$ and $\tau_3(v)$ also establish another asynchronous edge at slot τ_8. The number of asynchronous edges is $\gamma p_s|T| + \gamma^2 p_s|T|$ on average. In the rest $(1 - p_s)|T|$ time slots of T, there are $(1 - \gamma^2)(1 - p_s)|T|$ time slots left for asynchronous edges since there are $\gamma^2(1 - p_s)|T|$ synchronous vertices on average in the rest $(1 - p_s)|T|$ slots. Therefore, the overall CAT in the period T can be calculated by the following equation:

$$I(T) = \gamma^2(1 - p_s)|T| + \lambda \min\{(1 - \gamma^2)(1 - p_s)|T|, (\gamma + \gamma^2)p_s|T|\}, \tag{9}$$

where the item $(1 - \gamma^2)(1 - p_s)|T|$ decreases linearly with p_s and the item $(\gamma + \gamma^2)p_s|T|$ increases linearly with p_s in Equation (9). Given the fixed values of γ and λ, $I(T)$ can be maximized in Equation (9) when $p_s = 1 - \gamma$.

Lemma 2. *Given the fixed values of γ and λ, the overall CAT can be maximized by the correlated algorithm when $p_s = 1 - \gamma$.*

7.2 Theoretical Performance

Because of Lemma 2, the correlated algorithm can achieve its best performance, denoted by $I_c(T)$. According to Equation (9), $I_c(T) = [\gamma^2 + \lambda(1-\gamma^2)]\gamma|T|$. Recall that the online algorithm achieves its best performance when the active probability is 1. The best CAT of the online algorithm is $I_o(T) = \gamma^2|T|$ on average. By comparing $I_c(T)$ to $I_o(T)$, we have the following lemma.

Lemma 3. *When $\gamma < \frac{\lambda}{1-\lambda}$ and $\gamma \neq 0$ and 1, the performance of the correlated algorithm is always better than the online algorithm.*

Comparing to the offline algorithm, the theoretical performance of the correlated algorithm can be given in the following theorem.

Theorem 5. *The correlated algorithm has the approximation ratio as follows:*

$$\begin{cases} \gamma, & \gamma \text{ is a constant,} \\ 1 - \dfrac{1-\sqrt{\lambda}}{1+\sqrt{\lambda}}, & \lambda \text{ is a constant.} \end{cases} \tag{10}$$

Proof. The maximal CAT of the offline algorithm is $I_f(T) = \gamma^2|T| + \lambda(1-\gamma)\gamma|T|$ on average. Define a function $f(\cdot)$ of the two variables: γ and λ. Let $f(\gamma,\lambda) \triangleq \frac{I_c(T)}{I_f(T)} = \frac{[\gamma^2+\lambda(1-\gamma^2)]\gamma|T|}{[\gamma^2+\lambda(1-\gamma)\gamma]|T|} = 1 + \gamma - \frac{\gamma}{\gamma+\lambda(1-\gamma)}$. To find the approximate ratio is equivalent to finding the least value of $f(\gamma,\lambda)$, so we can look for the extremum of the function by its derivation.

Its first-order partial derivatives are $f_\gamma^{'}(\gamma,\lambda) = 1 - \frac{\lambda}{[\gamma+\lambda(1-\gamma)]^2}$; $f_\lambda^{'}(\gamma,\lambda) = \frac{\gamma(1-\gamma)}{[\gamma+\lambda(1-\gamma)]^2}$.

Let $f_\gamma^{'}(\gamma,\lambda) = 0$ and $f_\lambda^{'}(\gamma,\lambda) = 0$, and the critical points are derived: $(\gamma = 0, \lambda = 1)$ or $(\gamma = 1, \lambda = 1)$.

Its second-order partial derivatives are $f_{\gamma\gamma}^{''}(\gamma,\lambda) = \frac{2\lambda(1-\lambda)}{[\gamma+\lambda(1-\gamma)]^3}$; $f_{\gamma\lambda}^{''}(\gamma,\lambda) = \frac{\lambda-\gamma-\lambda\gamma}{[\gamma+\lambda(1-\gamma)]^3}$; $f_{\lambda\lambda}^{''}(\gamma,\lambda) = \frac{-2\gamma(1-\gamma)^2}{[\gamma+\lambda(1-\gamma)]^3}$.

Let $\mathbb{A} \triangleq f_{\gamma\gamma}^{''}(\gamma,\lambda)$; $\mathbb{B} \triangleq f_{\gamma\lambda}^{''}(\gamma,\lambda)$; $\mathbb{C} \triangleq f_{\gamma\lambda}^{''}(\gamma,\lambda)$.

As to the critical point $(\gamma = 0, \lambda = 1)$, we have $\mathbb{A} = 0, \mathbb{B} = 1, \mathbb{C} = 0$, and then $\mathbb{AC} - \mathbb{B}^2 \leq 0$. Thus, the critical point $(\gamma = 0, \lambda = 1)$ is not an extreme value point.

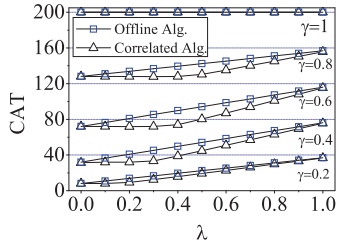
As to the critical point $(\gamma = 1, \lambda = 1)$, we have $\mathbb{A} = 0, \mathbb{B} = -1, \mathbb{C} = 0$, and then $\mathbb{AC} - \mathbb{B}^2 \geq 0$ and $\mathbb{A} = 0$. We cannot determine if $\gamma = 1, \lambda = 1$ is an extreme value point or not.

Since the extremum of the function $f(\gamma,\lambda)$ does not exist, we discuss the impacts of γ and λ on the function, respectively.

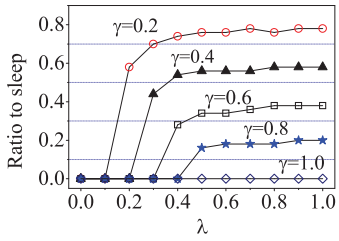
Let λ be a constant and $f(\gamma) \triangleq 1 + \gamma - \frac{\gamma}{\gamma+\lambda(1-\gamma)}$, where $0 \leq \lambda \leq 1$. Its first-order derivative is $f_\gamma^{'}(\gamma) = 1 - \frac{\lambda}{[\gamma+\lambda(1-\gamma)]^2}$. Let $f^{'}(\gamma) = 0$, and we can derive $\gamma = \frac{\sqrt{\lambda}}{1+\sqrt{\lambda}}$. When $\gamma \in [0, \frac{\sqrt{\lambda}}{1+\sqrt{\lambda}}]$, $f^{'}(\gamma) < 0$. When $\gamma \in [\frac{\sqrt{\lambda}}{1+\sqrt{\lambda}}, 1]$, $f^{'}(\gamma) > 0$. $f(\gamma)$ is monotonously decreasing when $\gamma \in [0, \frac{\sqrt{\lambda}}{1+\sqrt{\lambda}}]$. It is monotonously increasing when $\gamma \in [\frac{\sqrt{\lambda}}{1+\sqrt{\lambda}}, 1]$. Therefore, $f_{min}(\gamma) = 1 - \frac{1-\sqrt{\lambda}}{1+\sqrt{\lambda}}$ when $\gamma = \frac{\sqrt{\lambda}}{1+\sqrt{\lambda}}$.

Let γ be a constant and $f(\lambda) \triangleq 1 + \gamma - \frac{\gamma}{\gamma+\lambda(1-\gamma)}$, where $0 \leq \gamma \leq 1$. Its first-order derivative is $f^{'}(\lambda) = \frac{\gamma(1-\gamma)}{[\gamma+\lambda(1-\gamma)]^2}$. When $\lambda \in [0,1]$, we can derive that $f^{'}(\lambda) > 0$. Thus, when $\lambda \in [0,1]$, $f(\lambda)$ is monotone increasing. When $\lambda = 0$, $f_{min}(\lambda) = \gamma$. □

The rest of this section illustrates the performance of the offline, online, and correlated algorithms by the numerical evaluation. The performance of both the offline and correlated algorithms

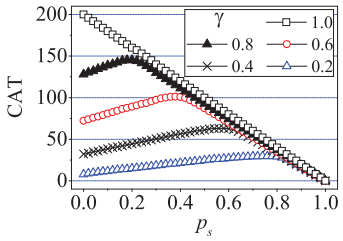


(a) Charge efficiency affection.

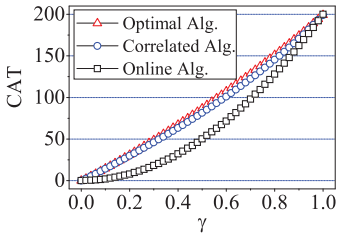


(b) Best sleep ratio.

Fig. 8. The CAT of the correlated algorithm with charge efficiency. T contains 200 time slots.



(a) The correlated algorithm.



(b) Performance comparison.

Fig. 9. T contains 200 slots. $\lambda = 0.8$, and $p_o = 1$.

increases with increasing charge efficiency as illustrated in Figure 8(a), where each group represents that γ takes a specific value, and contains two curves corresponding to the CATs of the offline and correlated algorithms, respectively.

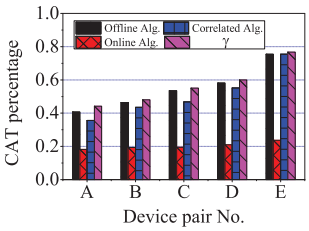
Given the charge efficiency λ and the energy harvesting probability γ, there is the best sleep ratio p_s by which the correlated algorithm can obtain its maximal CAT in Figure 9(a). It shows a case where the charge efficiency $\lambda = 0.8$, under which the best value of p_s is 0.79, 0.60, 0.41, and 0.18 when γ is 0.2, 0.4, 0.6, and 0.8, respectively. It is quite close to the theoretical result $p_s = 1 - \gamma$. Figure 8(b) shows the relation of the best sleep ratio with λ and γ. In this figure, p_s decreases with γ and increases with the charge efficiency λ. Figure 9 shows the performance of the correlated algorithm under the case $\lambda = 0.8$. Figure 9(b) compares the performance of the offline, online, and correlated algorithms. The active probability is set to be 1 for the online algorithm since it can achieve its best performance according to the analysis in Section 6.3. The CAT of the correlated algorithm reaches at least 93.81% of the online algorithm (i.e., the optimal algorithm), can be up to 84.29% better than the online algorithm when $\gamma = 0.48$.

8 EXPERIMENTAL EVALUATION

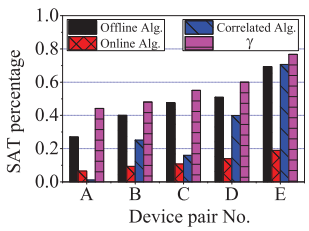
This section gives the detailed experiment setting and evaluates the performance of the offline, online, and correlated algorithms based on our real energy harvesting platform. By analyzing the data collected from the platform, we extract the features—heterogeneity and randomness—to evaluate the experimental performance of the three algorithms.

8.1 Experiment Setting

The real solar harvesting platform—SolarMote—consists of four main components: the solar panel, the battery, the controller, and the TelosB sensor node [Zhang and Li 2014]. The solar panel harvests energy to power the latter two components. The TelosB can sense data and communicate



(a) CAT percentage.



(b) SAT percentage.

Fig. 10. CAT and SAT percentages under the three algorithms when $\lambda = 0.8$ and $p_o = 50\%$ for the online algorithm.

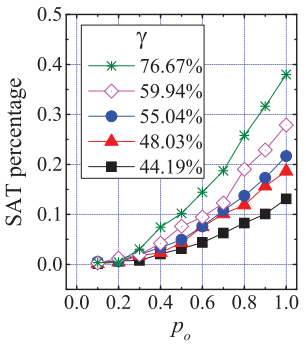


Fig. 11. The SAT of the online algorithm vs. active probability.

with others. The controller can manage the power direction: to power TelosB or to charge the battery. There are a total of 16 SolarMotes in the experiment. Seven of them have solar panels of size 70×28 mm and the rest have ones of size 80×38 mm so as to make the heterogeneity of the harvested energy obvious.

In this experiment, we consider the communication among device pairs, which can be applied to some popular applications such as data aggregation [Chang et al. 2017] or packet delivery [Lu et al. 2017]. Fifteen devices and one sink are deployed to harvest solar energy, and to sample and deliver the data of the temperature and solar luminous density. There are a total of eight pairs of devices, among which seven pairs of solar sensor devices run the three algorithms. The sink connects with a personal computer to collect data from devices. The period and the slot respectively are set to be 10 hours and 1 minute.

8.2 Result Analysis

This section measures the parameter CAT to evaluate the overall performance of the offline, correlated, and online algorithms. We introduce some metrics to extract the features of the experiment data including Synchronous Active Time (SAT), active probability, and heterogeneity, and analyze their impacts on the CAT. The SAT is the sum of the synchronous edges' weight, so it is a part of the CAT. Heterogeneity is defined as the percentage of the non-SAT out of the CAT between a pair of sensor nodes. Higher heterogeneity results in less SAT among devices. These metrics are determined by the energy harvesting probability γ, but also the three algorithms since the algorithms may result in different CAT and SAT. By selecting data from five pairs of nodes out of seven, which last 7 days of the experiment, we present the analysis result of the experiment data as the following figures. Notice that the total experiment time of the five pairs of nodes are not exactly the same, so we present the CAT percentage instead of the CAT in these figures. The former equals the fraction of the CAT divided by the experiment time. For example, if the experiment time for one node pair is 10 hours and the CAT is 2 hours, the CAT percentage is 20%.

CAT and SAT percentage. Figure 10(a) shows the CAT percentage obtained by the offline, online, and correlated algorithms among the five pairs of devices, which are denoted by A, B, C, D, and E under the different energy harvesting probability γ. The energy harvesting probabilities of the five pairs are 44.19%, 48.03%, 55.04%, 59.94%, and 76.67%. The CAT percentages of the three algorithms increase with γ. The correlated algorithm performs over the online algorithm and quite close to the offline algorithm. It verifies the theoretical results in Figure 9(b). The SAT of the online algorithm reaches its maximal value when $p_o = 1$ as shown in Figure 11. Similarly, from the experiment data

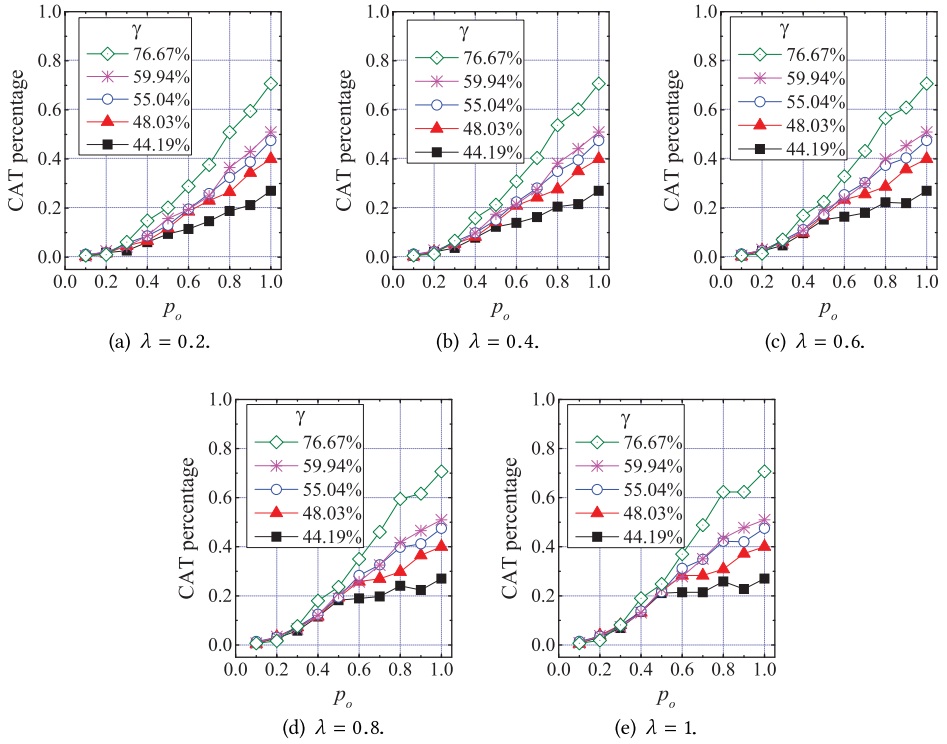


Fig. 12. The CAT of the online algorithm changes vs. active probability.

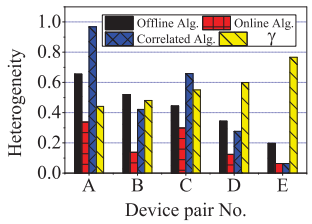
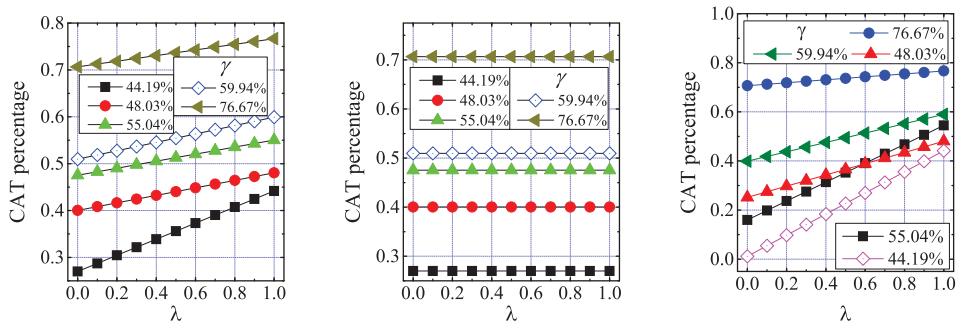


Fig. 13. Heterogeneity of the three algorithms when $\lambda = 0.8$ and the active probability $p_o = 50\%$.

in Figure 12, we can find that the online algorithm gains its maximal CAT under each energy harvesting probability when it sets the active probability $p_o = 1$, which is the same as the theoretical conclusion in Lemma 1.

Figure 10(b) shows the SAT percentage of the three algorithms when the active probability of the online algorithm is 50%. Among the nodes pairs A, B, C, D, and E, SATs of the offline algorithm occupy 66.27%, 86.21%, 88.76%, 87.65%, and 91.8% of their CAT s, respectively, whereas those of the online algorithm are 34.45%, 48.06%, 55.56%, 65.61%, and 80.28% of their CATs. SATs of the offline algorithm are 3.22%, 57.89%, 34.12%, 72.34%, and 93.64% of their CATs, respectively. In most cases, the CAT by the three algorithms mainly consists of the SAT, except the pairs A and C under the offline algorithm. Furthermore, the SAT percentage out of the CAT percentage does not increase with the energy harvesting probability. These two phenomena are caused by heterogeneity.



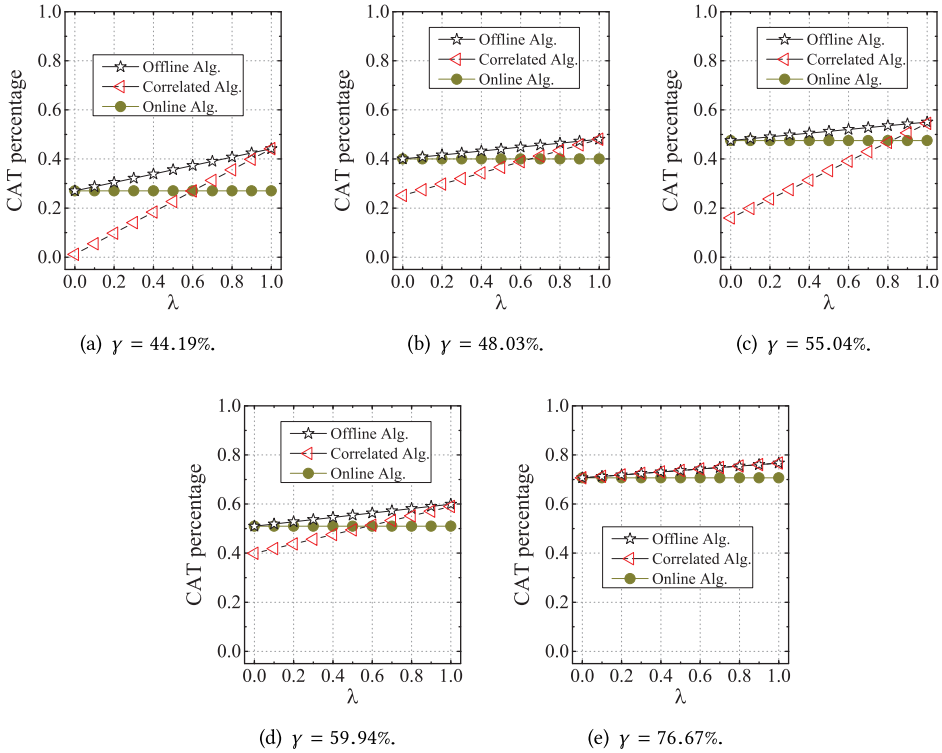
(a) The offline algorithm. (b) The online algorithm $p_o = 1$. (c) The correlated algorithm.

Fig. 14. CAT percentages of the three algorithms change with charge efficiency under different energy harvesting probabilities.

Heterogeneity. Heterogeneity causes the occurrences of the asynchronous edges and their nonuniform distribution in the energy state graph. The heterogeneities of the three algorithms are presented in Figure 13. First, all of the three algorithms lose their CAT due to the heterogeneity and the imperfect charge efficiency. For example, although the offline algorithm is optimal, its CAT percentage can reach the energy harvesting probability up to at most 98.43% as shown in Figure 10(a). Second, the heterogeneity causes the nonuniform distribution of the edges in the energy state graph. It brings an obvious effect on the correlated algorithm. Figure 10(b) shows that the SAT of the correlated algorithm between the node pair A is as low as 1.15%, and lower than 6.55% of the online algorithm. We can see that the heterogeneity of the correlated algorithm between pair A is high, up to 96.78% as shown in Figure 13. By checking the experiment data, we find that node pair A can harvest energy almost during the earlier part of each period, during which the correlated algorithm sets nodes to sleep as in the example in Figure 7. By comparing Figures 10(b) and 13, it is easy to discover that the SAT percentage of the correlated algorithm does not increase with the energy harvesting probability, but with the decreasing of its heterogeneity. Heterogeneity has an inversely proportional effect on the SAT of the online and correlated algorithms but not the online algorithm.

Charge efficiency. Charge efficiency is the direct factor to reduce the CAT. Figure 14 shows the CAT percentages of the three algorithms with charge efficiency under five energy harvesting probabilities, where the online algorithm sets the active probability to be 1. The charge efficiency is hard to measure directly, especially when the natural energy arrives randomly. To evaluate the impact of the charge efficiency, this section takes the value from 0 to 1 with the interval 0.2 to estimate its real value. With Figure 14, we can find that the CAT percentage of the offline and correlated algorithms increases with charge efficiency linearly under the five energy harvesting probabilities.

Figure 14(a) and (c) show that the CAT percentages of the offline and correlated algorithms change linearly with the charge efficiency under the five energy harvesting probabilities. In Figure 14(a), the increasing rates of these five curves are 0.17185, 0.08006, 0.07923, 0.07971, and 0.06, corresponding to the five energy harvesting probabilities $\gamma = 44.19\%$, 48.03%, 55.04%, 59.94%, and 76.67%. So we can find that the higher energy harvesting probability results in the lower increasing rate by the offline algorithm. In Figure 14(c), the increasing rates of these five curves are 0.4304, 0.2288, 0.3949, 0.1907, and 0.06, corresponding to the five energy harvesting probabilities. Recall that the heterogeneities of the correlated algorithm are 96.78%, 42.11%, 65.88%, 27.66%, and 6.36% under the five probabilities. The higher heterogeneity results in a higher increasing rate by the



(a) $\gamma = 44.19\%$. (b) $\gamma = 48.03\%$. (c) $\gamma = 55.04\%$.

(d) $\gamma = 59.94\%$. (e) $\gamma = 76.67\%$.

Fig. 15. CAT percentages of the three algorithms change with charge efficiency when the online algorithm sets the active probability to 1.

correlated algorithm. Figure 15 shows the comparison on the CAT percentage with charge efficiency under the five energy harvesting probabilities. The energy harvesting probability γ almost has no effect on the increasing ratio of the online and online algorithms but not the correlated algorithm. In Figure 15, the increasing ratio of the correlated algorithm is almost consistent with that of the offline algorithm. The performance of the correlated algorithm is lower than that of the online and offline algorithms when the energy harvesting probability is low as shown in Figure 15.

Energy harvesting probability. Energy harvesting probability has fundamental impacts on the three algorithms. In this section, we label the energy harvesting probability on each figure. Figure 10 shows that both the offline and online algorithms can obtain more CAT and SAT when the energy harvesting probability is higher. Figure 12 shows that the CAT of the online algorithm increases with the energy harvesting probability when the algorithm takes different active probabilities and the charge efficiency is set to be several values. Figure 14 shows that the CAT percentages of the three algorithms increase with the energy harvesting probability no matter the charge efficiency. Meanwhile, the SAT of the algorithm has a similar property—that is, higher energy harvesting probability results in more SAT under different active probabilities, as shown in Figure 11.

9 CONCLUSION

This article studies the duty cycling strategy by considering factors including the randomness, heterogeneity of the energy availability, the power weak and limited, and the imperfect charge

efficiency. It brings up a new duty cycling problem, which we call the *stochastic duty cycling problem*. To solve the problem, three algorithms are proposed with the goal to improve the utilization efficiency of the harvested energy. We find that some parameters, the energy harvesting probability and heterogeneity, including the charge efficiency, also have great effects on our algorithms. Theoretical and experimental analysis are presented to evaluate the performance of the algorithms. The results indicate that the offline algorithm is optimal. The correlated algorithm has the approximation ratio and is very close to the offline algorithm but is affected easily by the heterogeneity. The online algorithm has the approximation ratio $1 - e^{-\gamma}$ but is not easily affected by the heterogeneity. The correlated and online algorithms are more suitable to the practical applications than the offline algorithm. The online algorithm is more tolerable to the heterogeneity than the correlated algorithm.

The P2P model can be extended to the networked case, such as multihop networks, where each device may have more than one neighbor. In multihop networks, the different pairs of neighboring nodes can be assigned with different periods by the coloring technique [Wang et al. 2008]. For example, device u_1 has two neighbors, v_1 and v_2, and their communication can be operated with the TDMA mechanism. One pair of nodes, u_1 and v_1, can communicate in period T_1. The other pair of nodes, u_1 and v_2, can operate in period T_2. In this way, both of the two node pairs work under the P2P model.

There are other methods and directions to deal with the problem discussed in this article. For example, the neighboring nodes can communicate in the same period, and each of them is active in one subset of time slots in the period. This method can allocate the time slots precisely but leads to more complex calculation than ours. One possible direction for the harvested energy allocation is to consider the network dynamic. Imperfect charge efficiency is one aspect of the network dynamic, which means that the energy efficiency can be changed because of not only the charge efficiency but also other factors, such as dynamic link quality and transmission power.

REFERENCES

Jo Bito, Ryan Bahr, Jimmy G. Hester, Syed Abdullah Nauroze, Apostolos Georgiadis, and Manos M. Tentzeris. 2017. A novel solar and electromagnetic energy harvesting system with a 3-D printed package for energy efficient Internet-of-Things wireless sensors. *IEEE Transactions on Microwave Theory and Techniques* 65, 5 (Feb. 2017), 1831–1842. DOI: https://doi.org/10.1109/TMTT.2017.2660487

Martí Boada, Antonio Lazaro, Ramon Villarino, and David Girbau. 2018. Battery-less soil moisture measurement system based on a NFC device with energy harvesting capability. *IEEE Sensors Journal* 18, 13 (May 2018), 5541–5549. DOI: https://doi.org/10.1109/JSEN.2018.2837388

Chen Chang, Neng Zhu, and Jihong Shang. 2017. The study of occupant behavior analysis of Inner Mongolia in regard to heating energy consumption. *Procedia Engineering* 205 (2017), 915–922. DOI: https://doi.org/10.1016/j.proeng.2017.10.122

Quan Chen, Hong Gao, Zhipeng Cai, Lianglun Cheng, and Jianzhong Li. 2018. Energy-collision aware data aggregation scheduling for energy harvesting sensor networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM'18)*. IEEE, Los Alamitos, CA. DOI: https://doi.org/10.1109/INFOCOM.2018.8486366

Zhuangbin Chen, Anfeng Liu, Zhetao Li, Young-June Choi, and Jie Li. 2017. Distributed duty cycle control for delay improvement in wireless sensor networks. *Peer-to-Peer Networking and Applications* 10, 3 (May 2017), 559–578. DOI: https://doi.org/10.1007/s12083-016-0501-0

Donatella Darsena, Giacinto Gelli, and Francesco Verde. 2017. Modeling and performance analysis of wireless networks with ambient backscatter devices. *IEEE Transactions on Communications* 65, 4 (Jan. 2017), 1797–1814. DOI: https://doi.org/10.1109/TCOMM.2017.2654448

Yi Ding, Robert Michelson, and Charles Stancil. 2000. Battery state of charge detector with rapid charging capability and method. US Patent 6,094,033. http://hdl.handle.net/1853/57341.

Giacomo Ghidini and Sajal K. Das. 2011. An energy-efficient Markov chain-based randomized duty cycling scheme for wireless sensor networks. In *Proceedings of the 31st IEEE International Conference on Distributed Computing Systems (ICDCS'11)*. IEEE, Los Alamitos, CA, 67–76. DOI: https://doi.org/10.1109/ICDCS.2011.86

Jeremy Gummeson, Bodhi Priyantha, and Jie Liu. 2014. An energy harvesting wearable ring platform for gesture input on surfaces. In *Proceedings of the 12th ACM Annual International Conference on Mobile Systems, Applications, and Services (MobiSys'14)*. ACM, New York, NY, 162–175. DOI: https://doi.org/10.1145/2594368.2594389

Mohamadhadi Habibzadeh, Moeen Hassanalieragh, Akihiro Ishikawa, Tolga Soyata, and Gaurav Sharma. 2017. Hybrid solar-wind energy harvesting for embedded applications: Supercapacitor-based system architectures and design trade-offs. *IEEE Circuits and Systems Magazine* 17, 4 (Feb. 2017), 29–63. DOI: https://doi.org/10.1109/MCAS.2017.2757081

Shibo He, Jiming Chen, David K. Y. Yau, Huanyu Shao, and Youxian Sun. 2012. Energy-efficient capture of stochastic events under periodic network coverage and coordinated sleep. *IEEE Transactions on Parallel and Distributed Systems* 23, 6 (Oct. 2012), 1090–1102. DOI: https://doi.org/10.1109/TPDS.2011.242

Florian Heesen and Reinhard Madlener. 2018. Consumer behavior in energy-efficient homes: The limited merits of energy performance ratings as benchmarks. *Energy and Buildings* 172 (Aug. 2018), 405–413. DOI: https://doi.org/10.1016/j.enbuild.2018.04.039

Longbo Huang and M. J. Neely. 2013. Utility optimal scheduling in energy-harvesting networks. *IEEE/ACM Transactions on Networking* 21, 4 (Aug. 2013), 1117–1130. DOI: https://doi.org/10.1109/TNET.2012.2230336

Xiaofan Jiang, Joseph Polastre, and David Culler. 2005. Perpetual environmentally powered sensor networks. In *Proceedings of the 4th IEEE International Symposium on Information Processing in Sensor Networks (IPSN'05)*. ACM, New York, NY, 463–468. DOI: https://doi.org/10.1109/IPSN.2005.1440974

Muharrem Karaaslan, Mehmet Bağmancı, Emin Ünal, Oguzhan Akgol, and Cumali Sabah. 2017. Microwave energy harvesting based on metamaterial absorbers with multi-layered square split rings for wireless communications. *Optics Communications* 392 (June 2017), 31–38. DOI: https://doi.org/10.1016/j.optcom.2017.01.043

Feng Li, Yanbing Yang, Zicheng Chi, Liya Zhao, Yaowen Yang, and Jun Luo. 2018. Trinity: Enabling self-sustaining WSNs indoors with energy-free sensing and networking. *ACM Transactions on Embedded Computing Systems* 17, 2 (April 2018), 57. DOI: https://doi.org/10.1145/3173039

Xiaoying Liu, Kechen Zheng, Luoyi Fu, Xiao-Yang Liu, Xinbing Wang, and Guojun Dai. 2018. Energy efficiency of secure cognitive radio networks with cooperative spectrum sharing. *IEEE Transactions on Mobile Computing* 18, 2 (May 2018), 305–318. DOI: https://doi.org/10.1109/TMC.2018.2836902

Rongxing Lu, Kevin Heung, Arash Habibi Lashkari, and Ali A. Ghorbani. 2017. A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced IoT. *IEEE Access* 5 (Aug. 2017), 3302–3312. DOI: https://doi.org/10.1109/ACCESS.2017.2677520

Aleksander Madry. 2013. Navigating central path with electrical flows: From flows to matchings, and back. In *Proceedings of the 54th IEEE Annual Symposium on Foundations of Computer Science*. IEEE, Los Alamitos, CA, 253–262. DOI: https://doi.org/10.1109/FOCS.2013.35

Abbas Mehrabi and Kiseon Kim. 2016. Maximizing data collection throughput on a path in energy harvesting sensor networks using a mobile sink. *IEEE Transactions on Mobile Computing* 15, 3 (March 2016), 690–704. DOI: https://doi.org/10.1109/TMC.2015.2424430

Martin Raab and Angelika Steger. 1998. "Balls into bins"—A simple and tight analysis. In *Randomization and Approximation Techniques in Computer Science*. Springer, 159–170. https://link.springer.com/chapter/10.1007/3-540-49543-6_13.

Xingfa Shen, Cheng Bo, Jianhui Zhang, Guojun Dai, Xufei Mao, and XiangYang Li. 2009. SolarMote: A low-cost solar energy supplying and monitoring system for wireless sensor networks. Poster. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys'09)*. ACM, New York, NY, 413–414. DOI: https://doi.org/10.1145/1644038.1644128

Xingfa Shen, Cheng Bo, Jianhui Zhang, Shaojie Tang, Xufei Mao, and Guojun Dai. 2013. EFCon: Energy flow control for sustainable wireless sensor networks. *Elsevier Ad Hoc Networks* 11, 4 (June 2013), 1421–1431. DOI: https://doi.org/10.1016/j.adhoc.2011.07.003

Fei Tong and Jianping Pan. 2017. ADC: An adaptive data collection protocol with free addressing and dynamic duty-cycling for sensor networks. *Mobile Networks and Applications* 22, 5 (Oct. 2017), 983–994. DOI: https://doi.org/10.1007/s11036-017-0850-9

Xinbing Wang, Sihui Han, Yibo Wu, and Xiao Wang. 2013. Coverage and energy consumption control in mobile heterogeneous wireless sensor networks. *IEEE Transactions on Automatic Control* 58, 4 (June 2013), 975–988. DOI: https://doi.org/10.1109/TAC.2012.2225511

Yu Wang, Weizhao Wang, Xiang-Yang Li, and Wen-Zhan Song. 2008. Interference-aware joint routing and TDMA link scheduling for static wireless networks. *IEEE Transactions on Parallel and Distributed Systems* 19, 12 (Dec. 2008), 1709–1726. DOI: https://doi.org/10.1109/TPDS.2008.53

Qianqian Yang, Shibo He, Junkun Li, Jiming Chen, and Youxian Sun. 2015. Energy-efficient probabilistic area coverage in wireless sensor networks. *IEEE Transactions on Vehicular Technology* 64, 1 (Jan. 2015), 367–377. DOI: https://doi.org/10.1109/TVT.2014.2300181

Jianhui Zhang and Xiangyang Li. 2014. Energy-harvesting technique and management for wireless sensor networks. In *Rechargeable Sensor Networks: Technology, Theory and Application—Introducing Energy Harvesting to Sensor Networks*, J. Chen, S. He, and Y. Sun (Eds). Hackensack, NJ, 107–168. DOI: https://doi.org/10.1142/9789814525466_0004

Jianhui Zhang, Zhi Li, and Shaojie Tang. 2015. Value of information aware opportunistic duty cycling in solar harvesting sensor networks. *IEEE Transactions on Industrial Informatics* 12, 1 (Dec. 2015), 348–360. DOI: https://doi.org/10.1109/TII.2015.2508745

Yongmin Zhang, Shibo He, and Jiming Chen. 2016. Data gathering optimization by dynamic sensing and routing in rechargeable sensor networks. *IEEE/ACM Transactions on Networking* 24 3, (June 2016), 1632–1646. DOI: https://doi.org/10.1109/TII.2015.2508745

Kechen Zheng, Xiao-Yang Liu, Luoyi Fu, Xinbing Wang, and Yi-Hua Zhu. 2019. Energy efficiency in multihop wireless networks with unreliable links. *IEEE Transactions on Network Science and Engineering* (Jan. 2019). Available at DOI: https://doi.org/10.1109/TNSE.2018.2890430

Ting Zhu, Ziguo Zhong, Yu Gu, Tian He, and Zhili Zhang. 2009. Leakage-aware energy synchronization for wireless sensor networks. In *Proceedings of the 7th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys'09)*. ACM, New York, NY, 319–332. DOI: https://doi.org/10.1145/1555816.1555849

Received July 2018; revised July 2019; accepted November 2019