This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

Online Throughput Maximization for Energy Harvesting Communication Systems with Battery Overflow

Weiwei Wu, *Member, IEEE,* Jianping Wang, *Member, IEEE,* Xiumin Wang, *Member, IEEE,* Feng Shan, *Member, IEEE,* Junzhou Luo, *Member, IEEE*

***Abstract*—Energy harvesting communication system enables energy to be dynamically harvested from natural resources and stored in capacitated batteries to be used for future data transmission. In such a system, the amount of future energy to harvest is uncertain and the battery capacity is limited. As a consequence, battery overflow and energy dropping may happen, causing energy underutilization. To maximize the data throughput by using the energy efficiently, a rate-adaptive transmission schedule must address the trade-off between a high-rate transmission which avoids energy overflow and a low-rate transmission which avoids energy shortage. In this paper, we study an online throughput maximization problem without knowing future information. To the best of our knowledge, this is the first work studying the fully-online transmission rate scheduling problem for battery-capacitated energy harvesting communication systems. We consider the problem under two models of the communication channel, a static channel model that assumes the channel status is stable, and a fading channel model that assumes the channel status varies. For the former, we develop an online algorithm that approximates the offline optimal solution within a constant factor for all possible inputs. For the latter that the channel gains vary in range $[h_{min}, h_{max}]$, we propose an online algorithm with a proven $\Theta(\log(\frac{h_{max}}{h_{min}}))$-competitive ratio. Our simulation results further validate the efficiency of the proposed online algorithms.**

***Index Terms*—Energy harvesting, battery capacity, energy overflow, energy-efficient rate scheduling, throughput maximization, online algorithm, competitive ratio.**

I. Introduction

With salient features of energy harvesting, including self-sustainability and perpetual operation, more and more wireless devices are being equipped with energy-harvesting capacities. In such a system, energy is dynamically harvested and stored to the battery for future use. However, any battery has a limited capacity, which may result in energy *overflow* due to a sudden large amount of future energy arrival.

If future energy arrivals are fully known in advance, a device can use more energy to transmit data before a large amount of energy is harvested in the near future, so as to avoid energy overflow. However, energy harvested is *dynamic* in nature and becomes hard to be predicted when devices harvest energy from hybrid renewable resources, *e.g.*, solar, vibration, and wild. Thus, it becomes difficult to decide how a device shall consume the energy. On one hand, a conservative energy usage that uses energy at a low rate in each stage may result in energy overflow upon new energy arrival; on the other hand, an optimistic energy usage that consumes energy at a high rate in each stage may result in energy shortage.

Therefore, with dynamic energy arrivals and limited battery capacities, it is crucial to consider the energy efficiency and the trade-off between energy overflow and energy shortage. In this paper, we study online rate scheduling algorithms for wireless devices, equipped with rate-adaptive capacities, to maximize the throughput by efficiently utilizing the energy.

In rate-adaptive wireless systems, it is well-known that, for most encoding schemes [3],[4], the data rate achieved at any time is a concave increasing function of the transmission power allocated, which implies that, by consuming the same amount of energy, transmitting in a longer period with a lower rate always achieves higher data throughput than transmitting in a shorter period with a higher rate. Thus, besides the trade-off between energy-overflow and energy-shortage, another trade-off between high rate transmission and low rate transmission should be also well-addressed in order to maximize the throughput.

In this paper, we address the aforementioned two challenges and study the throughput-maximization rate scheduling problem in the online setting. In the literature, there are two types of online models for scheduling problems. In the *partially-online* model, partial future information is known, *e.g.*, distribution or prior information of the future energy arrival; in the *fully-online* model, no prior/distribution information of future energy arrivals is available. In this paper, we consider the fully-online rate scheduling problem. The output of the online algorithm will be measured by competitive analysis, which compares the throughput of the online algorithm with the optimal offline solution.

In the literature, the online rate scheduling with dynamic energy arrivals has been understudied. Before we are to have a detailed literature survey in Section II, here we make a discussion on some most-related work, to emphasize the contribution of our work. In prior work, Sharma *et al.* [18] develop energy management policies that keep the data queue stable and achieve the maximum throughput with the assumption of known mean energy and infinite battery capacity. Gatzianas *et al.* [17] investigate the data rate maximization problem

W. Wu, J. Luo and F. Shan are with School of Computer Science and Engineering, Southeast University, Nanjing, Jiangsu, P.R.China (Emails: {weiweiwu, jluo, shanfeng}@seu.edu.cn).

J. Wang is with Department of Computer Science, City University of Hong Kong, Hong Kong (Email: jianwang@cityu.edu.hk).

X. Wang is with School of Computing and Engineering, HeFei University of Technology, Hefei, Anhui, P.R.China (Email: wxiumin@hfut.edu.cn).

in multihop networks with the assumption of distribution information for energy arrivals. Vaze *et al.* [22] present the first work to study the fully-online setting without any distribution information to maximize the throughput before a known deadline T, in which a T-competitive online algorithm is developed. Due to its dependency on the assumption of infinite battery capacity, it fails to address the trade-off between energy overflow and energy shortage. Furthermore, it relies on the assumption of known number of time slots (deadline). In this paper, we drop these two restricted assumptions and attempt to develop more efficient algorithms.

In order to address the trade-off between energy overflow and energy shortage, we propose a novel method to estimate future energy arrivals in our algorithm design. That is, we partition the battery capacity into two equal parts, where the first part is taken as an estimator to obtain the time period during which the energy harvested is doubled compared to the prior time period, and the second part is used to deplete the energy accumulated. Instead of assuming known mean energy or distribution information, such a method provides a new way of estimation without relying on any prior information. Moreover, it further makes the arbitrary energy arrivals well-structured, which helps to approach the optimal offline solution.

The contributions of the paper are summarized as follow,

- We study the throughput-maximization rate scheduling problem in battery-capacitated energy-harvesting communication systems to address the trade-off between energy overflow and energy shortage. Two fully-online models are investigated, i.e. static channel model and fading channel model, which does not assume any distribution/future information of energy arrivals or channel status. This is the first work studying the fully-online rate scheduling problem for battery-capacitated energy-harvesting communication systems.
- For the static channel model with unlimited battery capacity, we analyze the lazy schedule proposed in [22] and prove it is constant competitive, which significantly improves the result of T-competitiveness in [22].
- As the main result of this paper, we consider the static channel model with finite battery capacity where energy overflow may occur. We develop a novel and efficient online algorithm, which is the first constant competitive algorithm for the most general setup in the literature.
- For the fading channel model, by adopting the algorithm developed for the static channel model as a building block, we derive a $\Theta(\log(\frac{h_{max}}{h_{min}}))$-competitive algorithm where h_{min}, h_{max} are the minimum/maximum value of channel status, which is the first logarithmic competitive algorithm and meanwhile asymptotically optimal competitive, substantially improving the linear competitive algorithm (T-competitive, close to the number of time slots) that assumes infinite battery capacity in the literature.

The rest of the paper is organized as follows. Section II reviews the related work in the literature. Section III presents the preliminaries. Section IV studies the online algorithms for the static channel model with infinite battery capacity where no overflow occurs. Section V develops an online algorithm for the static channel model with energy overflow addressed. The general fading channel model is then investigated in Section VI. The simulation results are presented in Section VII. Finally, we conclude the paper in Section VIII.

II. Related work

Tremendous research efforts have been made to design energy-efficient rate scheduling algorithms. A comprehensive review on the work in energy-harvesting wireless communications can be find in a recent survey [6]. We only review the most related ones due to space limit. In prior works, [7]-[16] investigate the rate scheduling algorithms for energy harvesting communication systems with various objectives, such as minimizing the completion time/energy consumption or maximizing the throughput. Tutuncuoglu and Yener [10] propose optimal algorithms to maximize the throughput with the consideration of finite battery capacity. Shan *et al.* [16] develop optimal max-throughput algorithm that satisfies individual deadline constraints.

All the above studies are limited within the scope of offline rate scheduling, which assumes known full information about future energy arrivals or channel states. However, future information may be hard to be obtained/predicted in some cases. Online models are more suitable for these cases. Unlike the offline setting where no energy overflow occurs, the trade-off between energy overflow and energy shortage should be addressed in the online setting. In prior work, Sharma *et al.* [18] and Gatzianas *et al.* [17] investigate energy management policies that keep the data queue stable and achieve the maximum throughput in single channel and multi-hop networks. Their result, however, relies on the assumption of distribution information and hence is only *partially-online*. To the best of our knowledge, [19], [20], [21], [22] are among the first works to theoretically study the *fully-online* algorithms that do not rely on any future/distribution information. Buchbinder *et al.* [19],[20] develop efficient max-throughput algorithms with $\Theta(\log(\frac{h_{max}}{h_{min}}))$-competitiveness for non-energy-harvesting systems. Vaze *et al.* [21] examine the completion time minimization problem for energy-harvesting systems and derive efficient competitive algorithms.

The most related work to this paper is Vaze *et al.* [22], which presents the first work to study the fully-online setting and maximize the throughput, in which a T-competitive online algorithm is proposed. The proposed algorithm works under known time slot information T and infinite battery capacity, without addressing energy overflow, which is pseudo-online and not practical enough. This paper attempts to remove these restrictions and develop more efficient algorithms with energy overflow addressed.

III. Preliminaries

In this section, we will introduce the system model in data transmission first and then formulate the problem.

A. System model

We consider the point-to-point data transmission where a transmitter needs to transmit as much data as possible to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

a receiver with dynamic energy arrivals and channel state changes. Assume that, a given time interval is partitioned into consecutive time slots, $0, 1, \ldots, T-1$. In an online problem, T can be either unknown or known in advance. We model the energy harvesting as a sequence of discrete time events. Let $\mathcal{H}$ be the harvesting instance that is composed of m harvestings $\{H_1, H_2, ..., H_m\}$. A harvesting $H_i = (E_i, t_i)$ is composed of E_i units of energy and an occurrence time $t_i \in \{0, 1, \ldots, T-1\}$. To be specific, we assume that the energy harvested with an amount E_i is immediately available to use in time slot t_i. Accordingly, we say that a harvesting event occurs at time t_i. The occurrence time of a harvesting event is also called a harvesting point. Assume that $0 = t_1 < t_2 < ... < t_m < T$ and the initial energy level is E_1 at the beginning time $t_1 = 0$.

The *channel gain* reflects the status of the channel, which determines the data rate of communication when the transmitter assigns a certain amount of energy for transmitting. We consider two channel gain models in this paper. In the static channel model, the channel gain is stable and equals h at any time, while in the fading channel model, the channel gain varies over time (known as fading effect). Denote by h_t the channel gain at time t, and correspondingly denote by h_{min} and h_{max} the minimum channel gain and the maximum channel gain respectively.

The transmitter can adaptively change its transmission rate r, which is related to the *power allocation* p and channel gain h through a function $r = g(p, h)$ called *power-rate* function. Such a power-rate function is concave and increasing on p in many systems with realistic encoding/decoding schemes [3],[4]. In this paper, we target at the typical logarithmic power-rate functions $r = a\log(1 + b \cdot h \cdot p) = g(p, h)$ with parameters $a, b > 0$, and a general concave increasing power-rate function $r = G(p, h)$ of which the first derivation satisfies $\frac{\partial G(p,h)}{\partial p} > 0$ and the second derivation satisfies $\frac{\partial^2 G(p,h)}{\partial^2 p} \leq 0$. We will write $g(p, h) = g(p)$ and $G(p, h) = G(p)$ for short if no ambiguity arises.

B. Problem formulation

A schedule needs to decide the *power allocation* $E(t)$ at each time $t \in [0, T)$ where $E(t)$ is also called *power allocation function*. If the battery capacity is infinite, no overflow will occur and a feasible schedule should satisfy the *energy constraints*. That is, the depleted power by time t should be at most the total energy harvested by that time,

$$\sum_{0 \leq t' \leq t} E(t') \leq \sum_{k: t_k \leq t} E_k, \ \forall t \in [0, T). \tag{1}$$

When the battery has a finite capacity B, the energy stored in the battery at each time slot should not exceed the capacity B. When the amount of energy harvested at time t_k is E_k and the empty space in the battery is less than E_k, we say that *energy overflow* occurs. Assume that at time t_k, the amount of energy recharged to the battery is $e(t_k)$, i.e., $0 \leq e(t_k) \leq E_k$, then the amount of energy lost is $d(t_k) = E_k - e(t_k) \geq 0$ due to overflow. Accordingly, a feasible schedule needs to satisfy both the *capacity constraint* that the amount of energy stored in the battery after recharging is at most B,

$$\sum_{0 \leq t' \leq t} e(t') - \sum_{0 \leq t' \leq t-1} E(t') \leq B, \ \forall t \in [0, T), \tag{2}$$

and the *energy constraint* that the amount of energy depleted is no more than that of the energy recharged for any time t,

$$\sum_{0 \leq t' \leq t} E(t') \leq \sum_{0 \leq t' \leq t} e(t'), \ \forall t \in [0, T). \tag{3}$$

The objective is to maximize the data throughput in the period $[0, T)$,

$$\sum_{t \in [0, T)} G(E(t), h_t), \tag{4}$$

where $h_t = h$ for $t \in [0, T)$ in the static model and h_t varies in the fading channel model.

When the number of time slots is known, the problem is to maximize the data throughput with known deadline T; when the number of time slots is unknown, the problem is generalized to maximize the data throughput with unknown future time slots.

We adopt the paradigm of competitive analysis to measure the worst-case performance of online algorithms, where an online algorithm ALG is compared to the optimal offline solution OPT that knows the entire information of the request sequence σ (e.g., energy arrivals and channel gains in this paper).

We say that an online rate scheduling algorithm is λ-*competitive* if it always achieves a throughput within λ times of the optimal offline solution for any input σ. That is,

$$\max_{\sigma} \frac{OPT(\sigma)}{ALG(\sigma)} \leq \lambda, \tag{5}$$

where $ALG(\sigma), OPT(\sigma)$ are the data throughput of ALG and OPT respectively. We say that an algorithm is constant competitive if the worst-case performance is always within an $O(1)$ factor compared to the optimal solution. In this paper, we aim at developing algorithms with asymptotically optimal performance bound, without optimizing the constant factor.

IV. Static channels: Throughput maximization without energy overflow

In this section, we consider the problem in the static channel model. The problem is to maximize the throughput before a known deadline T with the consideration of infinite battery capacity where no energy overflow occurs, same as in [22]. In such setting, we note that a lazy online algorithm REPA for the fading channel model is proposed in [12], [22] and it is proved to be T-competitive in [22]. In this section, we improve their results and prove that, as a fact, REPA is constant competitive (independent of the input size) in the static channel model.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

A. Online algorithm

Before presenting the algorithm, we introduce the properties of the optimal solution first. We start by introducing a basic concept called *equalization*. Suppose that there are two intervals with lengths T_1, T_2. The allocated power at each time slot inside these two intervals is constantly p_1, p_2 respectively and $p_1 \neq p_2$. If averaging all the power in these two intervals by a constant power $\frac{p_1 T_1 + p_2 T_2}{T_1 + T_2}$ does not violate the energy constraints over time, we can replace the original schedule in these two intervals by a new schedule with a single constant power to increase the throughput, by applying the concavity of the power-rate function. This will be called *equalization* and we say that the power in these two intervals are *equalized*. The time interval between any two adjacent harvesting points is defined to be an *epoch*. As stated in [7], [8], one basic property of the optimal solution is that the power allocation function is non-decreasing.

Lemma 1. *In the optimal schedule, the power allocation in any epoch keeps constant; the overall power allocation function is a non-decreasing step function.*

Algorithm REPA adopts naturally the lazy strategy to average all the current harvested energy over the remaining time and transmit with the total average energy at each time. Specifically, as presented in Algorithm 1, it computes the average amount of energy, $e_i = \frac{E_i}{T - t_i}$, on the occurrence of a harvesting, and then transmits with rate $G(\sum_{i:t_i \leq t} e_i)$ on time t.

Note that no energy overflow occurs since all energy harvested can be recharged to the battery. The resulting power allocation function is non-decreasing as time goes. The schedule in the proposed algorithm does not rely on any future information of the harvestings.

Algorithm 1 REPA

```
1: E = 0.
2: for on the occurrence of a harvesting H_i on time t_i do
3:     compute the average amount of energy e_i = E_i/(T - t_i).
4:     compute E = E + e_i.
5:     transmit at rate G(E, h) till the next harvesting point.
6: end for
```

B. Competitive analysis of the online algorithm

Now we prove that the algorithm is constant competitive in the static channel model.

Let $E^{opt}(t)$ and $E^{RE}(t)$ be the power allocation (function) computed in the optimal solution and the online algorithm REPA respectively. Let $E^{\mathcal{H}}(t)$ be the amount of harvested energy in $\mathcal{H}$ at time t. $E^{\mathcal{H}}(t)$ can be viewed as the *energy harvesting function* generated by $\mathcal{H}$. Note that $E^{opt}(t)$ and $E^{RE}(t)$ are computed with the input of harvestings $\mathcal{H}$, or equivalently, with the input of energy harvesting function $E^{\mathcal{H}}(t)$. Let $E^{RE,\mathcal{H}}(t)$ be the computed schedule of the online algorithm running on the input of the energy harvesting function $E^{\mathcal{H}}(t)$. Obviously, $E^{RE,\mathcal{H}}(t) = E^{RE}(t)$.

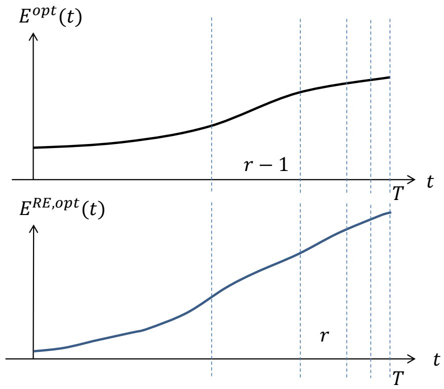

Fig. 1. An example that shows the regions in $E^{RE,opt}(t)$ and $E^{opt}(t)$.

Notice that $E^{opt}(t)$ is the optimal energy allocation at time t. On one hand, in order to maximize the throughput, the harvested energy $E^{\mathcal{H}}(t)$ is not utilized immediately at time t; instead, the optimal power allocation $E^{opt}(t)$ is the reallocation of the harvested energy which may properly postpone its usage to some later time. On the other hand, we can use function $E^{opt}(t)$ to represent an energy harvesting function in which the amount of energy harvested at time t is $E^{opt}(t)$. For the sake of analysis, we will abuse the notation and take a power allocation function $E^{(\cdot)}(t)$ as an energy harvesting function if it is required. Accordingly, we use $E^{RE,opt}(t)$ to denote the computed schedule of the online algorithm running on the input of the energy harvesting functions $E^{opt}(t)$. In the following analysis, we use $w(E^{(\cdot)})$ to denote the overall throughput achieved by energy allocation function $E^{(\cdot)}(t)$ over all time t, i.e. $w(E^{(\cdot)}) = \sum_{t \in [0,T)} G(E^{(\cdot)}(t))$.

Ideally, to prove the competitiveness, we need to build the relationship between the optimal throughput $w(E^{opt})$ and the throughput $w(E^{RE,\mathcal{H}})$ achieved by the online algorithm. Note that both $E^{RE,\mathcal{H}}(t)$ and $E^{opt}(t)$ are computed based on $\mathcal{H}$. Also, note that $E^{RE,opt}(t)$ is computed based on $E^{opt}(t)$ and $E^{opt}(t)$ is computed based on $\mathcal{H}$. This implies that $E^{RE,opt}(t)$ is computed indirectly based on $\mathcal{H}$ as well. To build the final relationship, we will use $E^{RE,opt}(t)$ as a bridge to compare $w(E^{opt})$ with $w(E^{RE,\mathcal{H}})$. That is, in the following analysis, we will first derive a bound between throughput $w(E^{opt})$ and $w(E^{RE,opt})$ in Lemma 2, and then further build the relationship between throughput $w(E^{RE,opt})$ and $w(E^{RE,\mathcal{H}})$ in Lemma 3.

We first establish the bound between the data throughput $w(E^{opt})$ achieved by the optimal power allocation and $w(E^{RE,opt})$ achieved by the online algorithm running on the energy harvesting function $E^{opt}(t)$. The idea is to observe the non-decreasing property of the two power allocation functions and partition the interval into regions such that throughput achieved in $E^{RE,opt}(t)$ in a later region is at least a constant factor of the optimal solution in its prior region.

Lemma 2. *The throughput achieved by* $E^{RE,opt}(t)$ *satisfies*

$$w(E^{RE,opt}) \geq \frac{w(E^{opt})}{18}. \tag{6}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

Proof. We partition the time axis $[0, T)$ into several regions as follows. For brevity, we assume T to be a value of 2 to some constant power first, say $T = 2^U$, and discuss the case of arbitrary value T later. The regions are respectively the intervals $[0, T-\frac{T}{2}), [T-\frac{T}{2}, T-\frac{T}{4}), ..., [T-\frac{T}{2^{r-1}}, T-\frac{T}{2^r}), ..., [T-1, T)$, where region r has length $\frac{T}{2^r}$. Fig. 1 shows an example of the regions.

To bound the ratio between $w(E^{RE,opt})$ and $w(E^{opt})$, we compare the total rate obtained from $E^{RE,opt}$ and E^{opt} respectively in every two adjacent regions, say region r and $r-1$. In region $r-1$ of E^{opt}, the total rate is $\sum_{T-\frac{T}{2^{r-2}} \le t < T-\frac{T}{2^{r-1}}} G(E^{opt}(t))$. In region r of $E^{RE,opt}(t)$, the total rate obtained is $\sum_{T-\frac{T}{2^{r-1}} \le t < T-\frac{T}{2^r}} G(\sum_{i \le t} \frac{E^{opt}(i)}{T-i})$, which is at least

$$\sum_{T-\frac{T}{2^{r-1}} \le t < T-\frac{T}{2^r}} G(\sum_{t-\frac{T}{2^r} \le i < t} \frac{E^{opt}(i)}{T-i}) \tag{7}$$

$$\ge \sum_{T-\frac{T}{2^{r-1}} \le t < T-\frac{T}{2^r}} G(\sum_{t-\frac{T}{2^r} \le i < t} \frac{E^{opt}(t-\frac{T}{2^r})}{\frac{T}{2^{r-2}}}) \tag{8}$$

$$\ge \sum_{T-\frac{T}{2^{r-1}} \le t < T-\frac{T}{2^r}} G(\frac{E^{opt}(t-\frac{T}{2^r})}{4}) \tag{9}$$

$$\ge \sum_{T-\frac{T}{2^{r-1}}-\frac{T}{2^r} \le t < T-\frac{T}{2^{r-1}}} G(\frac{E^{opt}(t)}{4}) \tag{10}$$

$$\ge \frac{1}{2} \sum_{T-\frac{T}{2^{r-2}} \le t < T-\frac{T}{2^{r-1}}} G(\frac{E^{opt}(t)}{4}) \tag{11}$$

$$\ge \frac{1}{8} \sum_{T-\frac{T}{2^{r-2}} \le t < T-\frac{T}{2^{r-1}}} G(E^{opt}(t)). \tag{12}$$

The first inequality holds because $E^{opt}(i)$ is non-decreasing by Lemma 1 and $T-i \le T-(t-\frac{T}{2^r}) \le \frac{T}{2^{r-1}} + \frac{T}{2^r} \le \frac{T}{2^{r-2}}$. The second inequality is correct since the total number of items in the inner summation is at least $\frac{T}{2^r} \ge \frac{1}{4}\frac{T}{2^{r-2}}$. The third one holds by rewriting index t and the fact that the length of region r is $\frac{T}{2^r}$. The second last inequality is because $E^{opt}(t)$ is non-decreasing and the length $\frac{T}{2^r}$ of region r is at least $\frac{1}{2}$ of that of region $r-1$. The last inequality holds by $4G(\frac{E^{opt}(t)}{4}) \ge G(E^{opt}(t))$ by the concavity of the power-rate function.

Thus, the total rate obtained in region r in $E^{RE,opt}(t)$ is at least $\frac{1}{8}$ of that in region $r-1$ in $E^{opt}(t)$. Furthermore, the last region has length 1 and the energy allocated in that time in $E^{RE,opt}(t)$ is at least that of $E^{opt}(t)$. Consequently, summing up the rate in all regions of $E^{RE,opt}(t)$, we have

$$10 \cdot w(E^{RE,opt}) \tag{13}$$

$$\ge 8 \sum_{r=1}^{U} \sum_{t=T-\frac{T}{2^{r-1}}}^{T-\frac{T}{2^r}} G(\sum_{t-\frac{T}{2^r} \le i < t} \frac{E^{opt}(i)}{T-i}) + 2w(E^{RE,opt}) \tag{14}$$

$$\ge \sum_{1 \le r \le U} \sum_{t=T-\frac{T}{2^{r-2}}}^{T-\frac{T}{2^{r-1}}} G(E^{opt}(t)) + \sum_{T-2 \le t < T} G(E^{opt}(t)) \tag{15}$$

$$\ge w(E^{opt}). \tag{16}$$

Now we discuss the case that T is an arbitrary value. We partition the time axis $[0, T)$ into several regions as follows. The regions are respectively the intervals $[0, T-\lceil\frac{T}{2}\rceil), [T-\lceil\frac{T}{2}\rceil, T-\lceil\frac{T}{4}\rceil), ..., [T-\lceil\frac{T}{2^{r-1}}\rceil, T-\lceil\frac{T}{2^r}\rceil), ..., [T-1, T)$, where region r has length $\lceil\frac{T}{2^{r-1}}\rceil - \lceil\frac{T}{2^r}\rceil$. The detailed computation would give that the length of region r is at least $\frac{1}{4}$ that of region $r-1$. Following similar deduction as above, the total rate obtained in region r in $E^{RE,opt}(t)$ is at least $\frac{1}{16}$ of that in region $r-1$ in $E^{opt}(t)$. Finally, summing up the rates in all regions, we have $w(E^{RE,opt}) \ge \frac{1}{18} w(E^{opt})$. This completes the proof. □

Then, we further establish the relationship between the throughput achieved in $E^{RE} = E^{RE,\mathcal{H}}$ and $E^{RE,opt}$, which respectively take $\mathcal{H}$ and $E^{opt}(t)$ as the input. The idea is to observe that, as the inputs of the online algorithm, $E^{opt}(t)$ is generated by properly postponing the usage of harvested energy in $\mathcal{H}$.

Lemma 3. $w(E^{RE,\mathcal{H}}) \ge w(E^{RE,opt})$.

Proof. We start by investigating the following splitting process applied to a harvesting. The harvesting, say H_k, is split into a harvesting $\hat{H}_k$ with amount e, which occurs at time $\hat{t}_e > t_k$, and a remaining harvesting $\bar{H}_k$ with amount of $e_k - e$ that still occurs at time t_k. Denote by $\mathcal{H}$ the instance $\{H_1, ..., H_k, ..., H_m\}$ before the splitting and $\mathcal{H}'$ the instance after the splitting. When running algorithm REPA respectively with the input of $\mathcal{H}$ and with the input of $\mathcal{H}'$, the part of energy with amount e in H_k is averaged over interval $[t_k, T)$ before the splitting while it is averaged over interval $[\hat{t}_e, T)$ after the splitting. Note that $E^{RE,\mathcal{H}}(t)$ is non-decreasing. Compared with averaging e over interval $[\hat{t}_e, T)$, averaging e over interval $[t_k, T)$ would move part of energy with amount e from $[\hat{t}_e, T)$ in $E^{RE,\mathcal{H}'}(t)$ to $[t_k, \hat{t}_e)$. Moreover, the resulting function $E^{RE,\mathcal{H}}(t)$ is non-decreasing. This implies that $w(E^{RE,\mathcal{H}}) \ge w(E^{RE,\mathcal{H}'})$ by the concavity of the power-rate function.

Observing this, we generalize the analysis above to show $w(E^{RE,\mathcal{H}}) \ge w(E^{RE,opt})$. The harvested energy can only be utilized after its occurrence time. The optimal solution $E^{opt}(t)$ properly delays the usage of some energy to make the energy assigned in non-decreasing manner, and hence achieves the optimal throughput. The function $E^{opt}(t)$ can be taken as another harvesting instance generated by properly delaying the occurrence time or partial harvested energy in $\mathcal{H}$. Now, consider $E^{RE,opt}(t)$ and $E^{RE,\mathcal{H}}(t)$ that is obtained by running REPA with the input of $E^{opt}(t)$ and with the input of $\mathcal{H}$ respectively. $E^{RE,\mathcal{H}}(t)$ can be generated by gradually and properly moving part of the energy of $E^{RE,opt}(t)$ to some early time by applying the splitting process to multiple harvestings. The resulting function $E^{RE,\mathcal{H}}(t)$ is still non-decreasing, which would induce the desired bound that $w(E^{RE,\mathcal{H}}) \ge w(E^{RE,opt})$. □

Finally, combining the results in Lemma 2 and Lemma 3, we have

$$w(E^{RE,\mathcal{H}}) \ge w(E^{RE,opt}) \ge \frac{w(E^{opt})}{18}. \tag{17}$$

This has derived the constant competitiveness of Algorithm REPA, as summarized in the following theorem.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

Theorem 1. *Algorithm REPA is constant competitive in the static channel model for general concave power-rate functions.*

We note that scheduling with known/fixed deadline T and infinite battery capacity is *pseudo-online* and not practical enough. Thus, in the next section, we will remove such assumptions to design a more practical algorithm.

V. Static channels: throughput maximization with energy overflow

In this section, we study the online throughput maximization problem in static channels with a capacitated-battery, where energy overflow may occur and the trade-off between energy overflow and energy shortage has to be considered. Moreover, we consider a more general setting in which the future time slot information is unknown. This practically models the general online setup where the transmitter needs to maximize the data throughput without knowing a deadline, which will be further adopted as a key building block to deal with the scheduling in fading channels in the next section.

When the battery capacity constraints are incorporated, we can see that the following different properties arise. First, the harvested energy may have to be dropped due to overflow at some time. Second, to achieve higher throughput, the rate may be increased in advance in case of energy dropping/overflow in the next time at which a large amount of energy is harvested. Third, the rate in the optimal solution is no longer non-decreasing and hence the monotonic property stated in the previous section fails to hold.

Furthermore, with unknown information of future time slots, it is even impossible to apply the averaging process like REPA, which divides the residual energy over the number of the remaining time slots. These together bring us many difficulties in designing online algorithms with proven good performance bounds.

In the following subsections, we will propose a novel algorithm and examine the structural properties brought by the design of the algorithm first, and then prove its constant competitiveness based on those properties.

A. Online algorithm design and its structural properties

To derive an online algorithm with constant competitiveness, we propose a novel method to deal with the battery capacity and unknown information of future time slots.

The idea is to equally divide the battery capacity into the cumulative part (which receives the energy) and executive part (which depletes the energy received by the cumulative part). When the accumulated energy is doubled or exceeds its capacity, i.e., $\frac{B}{2}$, the cumulative part would empty its received energy to the executive part. When the capacity $\frac{B}{2}$ is exceeded, the energy harvested cannot be accumulated/recharged any more and energy overflow occurs. The cumulative part is indirectly utilized as a deterministic *estimator* to guess the time period in which the accumulated energy of the harvestings is doubled or exceeds the capacity $\frac{B}{2}$. Such an estimation method would partition the online energy arrivals into sessions so that the energy in each session and the accumulated energy over sessions would have well-structured properties, which could be utilized to approach the optimal offline solution.

Algorithm 2 Cumulative-Guessing($\mathcal{H}$,B)

1: Divide the battery into two parts, the cumulative part with capacity $\frac{B}{2}$ and the executive part with capacity $\frac{B}{2}$. Initially set the amount in the cumulative part and executive part to be $E_c = 0$ and $E_e = 0$.
2: Set index $u = -1$. Let T_u denote the number of time slots in session u. Initially set all $T_u = 0$ and write the initial cumulative amount of energy to be $X_u = 0$. Set the time index $\tau = 0$.
3: **while** on the arrival of time t **do**
4:   If the harvested energy at time t is E_i, add E_i to the cumulative part, $E_c = E_c + E_i$.
5:   $\tau = \tau + 1$.
  // identify the first time at which E_c exceeds the amount $\min\{2X_u, \frac{B}{2}\}$.
6:   **if** $E_c \geq 2 \cdot X_u$ or $E_c \geq \frac{B}{2}$ **then**
7:     Report that t is the first time at which E_c is doubled or at least $\frac{B}{2}$ (critical time point). Set $X_{u+1} = \min\{E_c, \frac{B}{2}\}, T_u = \tau - 1$.
8:     Reset the executive part $E_e = X_{u+1} = \min\{E_c, \frac{B}{2}\}$ and cumulative part $E_c = 0$.
9:     Reset the time index $\tau = 1$ and set $u = u+1$. Session $u+1$ begins.
10:   **end if**
11:   Deplete the power from execute part and transmit at rate $r(X_u, \tau) = g((\frac{(3-2\sqrt{2})X_u}{4 \cdot h \cdot 2^{\lceil \log \tau \rceil}})^{\frac{1}{2}})$ at time t as the procedure in [19] to approximate (without information T_u) the local optimal solution $T_u \cdot g(\frac{X_u}{T_u})$ in session u.
12: **end while**

Algorithm Cumulative-Guessing presents the design of the online algorithm, which takes the harvestings $\mathcal{H}$ and capacity B as the input. We use E_c and E_e to denote the amount of energy in the cumulative part and the executive part. The online algorithm will divide the time interval into sessions. We use X_{u+1} to denote the amount of energy at the beginning of session $u+1$ in the executive part. We call the time at which the amount of energy in the cumulative part exceeds $2X_u$ or $\frac{B}{2}$ (Step 6) to be the *critical time point*. The interval between any two critical time points is defined to be a *session*. In the **while** loop, the cumulative part starts to receive/accumulate energy from the beginning of each session and hence $E_c = E_c + E_i$ when the amount of energy harvested is E_i. If E_c exceeds $\min\{2X_u, \frac{B}{2}\}$ at the critical time point, the executive part of the battery would recharge energy with amount of E_c from the cumulative part and hence empty the cumulative part, which is achieved by resetting the executive part to be $E_e = X_{u+1} = \min\{E_c, \frac{B}{2}\}$ and the cumulative part to be $E_c = 0$. Accordingly, session $u+1$ starts at the first time point at which $X_{u+1} \geq \min\{2X_u, \frac{B}{2}\}$ (also, the time index τ is reset to be zero at that time) and ends right before the first time point at which $X_{u+2} \geq \min\{2X_{u+1}, \frac{B}{2}\}$. Note that X_u is non-decreasing on u and energy overflow occurs when the cumulative part fully uses its capacity $\frac{B}{2}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

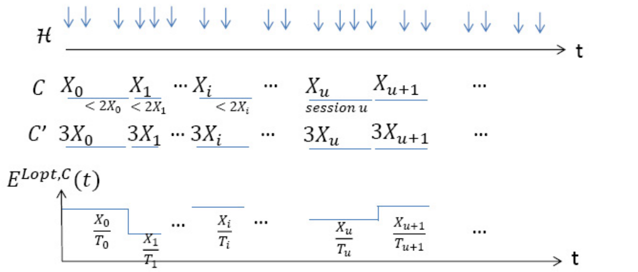

Fig. 2. An example that demonstrates the accumulated instance $\mathcal{C} = \{(C_i, T_i), 0 \leq i \leq U\}$ and the local optimal solution in session u, $T_u \cdot g(\frac{X_u}{T_u})$. In the accumulated instance $\mathcal{C}$, the interval covered by a line is the duration of a session and X_u is the energy of the executive part when session u begins. The accumulated instance $\mathcal{C}'$ is assumed to be $\{(3C_i, T_i), 0 \leq i \leq U\}$.

Note that at the beginning of each session u, the amount X_u of energy is known at that time, while the *length* (total number of time slots) T_u of session u is *future information* and not known until the end of that session. Thus, the online algorithm can only schedule without using the future information T_u. The maximum achievable total data rate to allocate the energy with amount X_u in T_u time slots is $T_u \cdot g(\frac{X_u}{T_u})$, which will be called *local optimal solution of session* u. We notice that [19] provides an online procedure to approximate such a local optimal solution within a constant factor of around 48 without relying on the future information T_u, where the rate assigned at the τ-th time among the T_u time slots is denoted by $r(X_u, \tau)$ in this paper.

Thus, we set an index τ in our algorithm and run that procedure to transmit at rate $r(X_u, \tau)$, to approximate the local optimal solution of session u, $T_u \cdot g(\frac{X_u}{T_u})$, in Step 10.

Denote by $C_i = (X_i, T_i)$ an accumulated event where T_i is the length of session i. Let $\mathcal{C} = \{C_0, C_1, C_2, ..., C_U\}$ be the accumulated instance. Fig. 2 presents an example for the accumulated instance and the local optimal solution $T_u \cdot g(\frac{X_u}{T_u})$ of session u.

The design of our online algorithm implies the following structural properties:

- **P1**: (**Session property**) between every two adjacent critical time points, the total amount of harvested energy between the two critical time points is less than $\min\{2X_u, \frac{B}{2}\}$, while the accumulated amount of energy satisfies $X_{u+1} \geq \min\{2X_u, \frac{B}{2}\}$ at the latter critical time point, since the latter one is the first time point at which the accumulated energy exceeds $\min\{2X_u, \frac{B}{2}\}$.
- **P2**: (**Accumulation property**) if session $u+1$ is the first session with $X_{u+1} \geq \frac{B}{2}$, then $X_i \geq 2X_{i-1}$ for $i \leq u$, and moreover, simple deduction would derive that the amount of harvested energy in session i is at least that of the total energy harvested by session $i-1$, i.e. $X_i \geq \sum_{j=0}^{i-1} X_j$ for any $i \leq u$.
- **P3**: (**Local-rate property**) in each session u, the algorithm allocates the power at rate $r(X_u, \tau)$ to approximate the local optimal solution $T_u \cdot g(\frac{X_u}{T_u})$ within a constant factor.

The session property states the property of the energy accumulated in the critical time point. The accumulation property states the structural property of the energy accumulated in the current session and the energy harvested by that session. The local-rate property provides an approximation for the local maximum achievable rate in each session. These properties altogether do help the online algorithm to approach the optimal solution, which will be shown in the next subsection.

B. Constant competitiveness of the online algorithm

Based on the structural properties brought by the design of the algorithm, we will prepare *four* equalities/inequalities to derive the constant competitiveness of our online algorithm CUMULATIVE-GUESSING.

First, we build a relationship between the output of the algorithm running on the original harvestings $\mathcal{H}$ and that of the algorithm running on the harvestings transformed from the accumulated instance $\mathcal{C}$. Let $E^{CG,\mathcal{H}}(t)$ be the computed schedule returned by CUMULATIVE-GUESSING running on $\mathcal{H}$. Let X_0 be the energy at the beginning. Session 1 starts at the first time point at which $X_1 \geq \min\{2X_0, \frac{B}{2}\}$ and the interval before that time point is treated as session 0. Algorithm CUMULATIVE-GUESSING individually tackles every accumulated event C_i to approximate the local optimal solution without the length information T_i of the session. Thus, $\mathcal{C}$ can be transformed into another harvesting instance

$$\mathcal{H}(\mathcal{C}) = \{(X_0, 0), (X_1, T_0), ...(X_U, \sum_{i=0}^{U-1} T_i)\}. \tag{18}$$

Let $E^{CG,\mathcal{C}}(t)$ be the computed schedule returned by CUMULATIVE-GUESSING running on such a harvesting instance $\mathcal{H}(\mathcal{C})$. Then, we have the first relationship,

$$w(E^{CG,\mathcal{H}}) = w(E^{CG,\mathcal{C}}). \tag{19}$$

Second, by applying the session property (P1), we derive a relationship between the optimal solution with the input of the original harvestings $\mathcal{H}$ and that with the input of the harvestings transformed from the accumulated instance $\mathcal{C}$. Note that $\mathcal{C}$ is generated in the cumulative part by running CUMULATIVE-GUESSING on $\mathcal{H}$, and moreover, $\mathcal{C}$ can be transformed into another harvesting instance

$$\mathcal{H}(\mathcal{C}) = \{(X_0, 0), (X_1, T_0), ...(X_U, \sum_{i=0}^{U-1} T_i)\}, \tag{20}$$

where the harvested energy at time $\sum_{i=0}^{u-1} T_i$ is X_u (which is generated by postponing the energy harvesting in $\mathcal{H}$). Denote by $E^{opt,\mathcal{H}}(t)$ and $E^{opt,\mathcal{C}}(t)$ respectively the optimal schedule with the input of harvesting instance $\mathcal{H}$ and that with the input of $\mathcal{H}(\mathcal{C})$ transformed from $\mathcal{C}$. Obviously,

$$w(E^{opt,\mathcal{C}}) \leq w(E^{opt,\mathcal{H}}). \tag{21}$$

Let $\mathcal{C}' = \{(3X_i, T_i), 0 \leq i \leq U\}$ where the amount $3X_u$ of energy at time $\sum_{i=0}^{u-1} T_i$ corresponds to the upper bound of total energy in the accumulative part at that time if the energy harvested in session u of $\mathcal{H}$ (which is at most $2X_u$ according to the session property (P1)) is advanced to that time, which

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

is illustrated in Fig. 2. Moreover, $\mathcal{C}'$ can be transformed into a harvesting instance

$$\mathcal{H}(\mathcal{C}') = \{(3X_0, 0), (3X_1, T_0), ...(3X_U, \sum_{i=0}^{U-1} T_i)\}. \tag{22}$$

Obviously, it is true that

$$w(E^{opt,\mathcal{H}}) \leq w(E^{opt,\mathcal{C}'}). \tag{23}$$

Therefore, we have the second relationship

$$w(E^{opt,\mathcal{H}}) \leq w(E^{opt,\mathcal{C}'}) \leq 3w(E^{opt,\mathcal{C}}), \tag{24}$$

where the last inequality holds by applying the concavity of the power-rate function and the fact that the length of each session is not changed while the harvested energy is enlarged three times in $\mathcal{C}'$ than $\mathcal{C}$ does.

Third, based on the local-rate property (P3), we examine the relationship between the throughput of the online algorithm and the *local optimal solution*, which is defined to be the local maximum data rate $\sum_{0\leq i\leq U} T_i g(\frac{X_i}{T_i})$ achieved in all sessions by equalizing an amount X_i of energy over T_i time slots for each accumulated event $C_i = (X_i, T_i)$. Because by the equalization method, if the number of time slots is known, the optimal solution for allocating the energy with an amount E over T time slots is to equalize the energy to achieve $T \cdot g(\frac{E}{T})$ data rate. For each individual accumulated event $C_i = (X_i, T_i)$, as stated in the local-rate property (P3), algorithm CUMULATIVE-GUESSING runs the procedure in [19] with rate $r(X_u, \tau)$ to approximate (without information T_i) the local optimal solution $T_i g(\frac{X_i}{T_i})$ within a constant factor. Accordingly,

$$w(E^{CG,\mathcal{C}}) = \sum_{0\leq i\leq U} \sum_{1\leq\tau\leq T_i} r(X_u, \tau). \tag{25}$$

Let $E^{LOpt,\mathcal{C}}(t)$ be the corresponding local optimal power allocation function for all the sessions and $w(E^{LOpt,\mathcal{C}})$ be the local optimal solution, i.e.,

$$w(E^{LOpt,\mathcal{C}}) = \sum_{0\leq i\leq U} T_i g(\frac{X_i}{T_i}). \tag{26}$$

Applying the competitiveness in [19] that

$$T_i g(\frac{X_i}{T_i}) \leq O(1) \sum_{1\leq\tau\leq T_i} r(X_u, \tau), \tag{27}$$

we have the following third relationship,

$$w(E^{LOpt,\mathcal{C}}) = \sum_{0\leq i\leq U} T_i g(\frac{X_i}{T_i}) \leq O(1) w(E^{CG,\mathcal{C}}). \tag{28}$$

Last, by using the structural information of the sessions implied by the accumulation property (P2), we prove the relationship between the local optimal solution $w(E^{LOpt,\mathcal{C}})$ and the optimal solution $w(E^{opt,\mathcal{C}})$. The proof is moved to Appendix.

Lemma 4. *Running on the accumulated instance $\mathcal{C}$, the throughput achieved by the local optimal power allocation function satisfies*

$$w(E^{LOpt,\mathcal{C}}) \geq \frac{1}{4} w(E^{opt,\mathcal{C}}). \tag{29}$$

Remark: *The idea of the proof is to observe two facts derived by the accumulation property (P2). First, before the first time point at which the accumulated energy exceeds the capacity $\frac{B}{2}$, a subset of sessions that contribute high data rate can be carefully selected and proved with a good throughput compared to the optimal solution. Second, after that time point, the energy in the executive part is $\frac{B}{2}$ and sufficiently large to approximate the throughput achievable for the optimal solution.*

Consequently, combining all the inequalities (19)(24)(28)(29) above, we have

$$\begin{aligned} w(E^{opt,\mathcal{H}}) &\leq w(E^{opt,\mathcal{C}'}) && (30)\\ &\leq 3 \cdot w(E^{opt,\mathcal{C}}) && (31)\\ &\leq 12 \cdot w(E^{LOpt,\mathcal{C}}) && (32)\\ &\leq 12 \cdot O(1) w(E^{CG,\mathcal{C}}) && (33)\\ &= O(1) w(E^{CG,\mathcal{H}}). && (34) \end{aligned}$$

This has derived the constant competitiveness of the algorithm's wost-case performance, as concluded in the following theorem, which is the first algorithm achieving the constant competitive ratio. Such a performance bound is independent of the input size.

Theorem 2. *Algorithm* CUMULATIVE-GUESSING *is constant competitive in the static channel model with finite battery capacity and unknown future information.*

Discussion 1: *We note that [18] developed partially-online algorithms under the assumption of known (future) mean of harvested energy and infinite battery capacity. Interestingly, although such assumption is removed in this paper, the estimation method developed is able to obtain similar information (the harvested energy in the near future). That is, half of the battery capacity is used to estimate the time period in which the energy harvested is doubled. In such a sense, our method provides a new method in guessing/obtaining future information and helps to approach the optimal offline solution, which is powerful and surprising at first glance. We believe that such a method is of independent interest for studying other related problems in energy harvesting systems.*

VI. FADING CHANNELS: THROUGHPUT MAXIMIZATION WITH ENERGY OVERFLOW

In this section, we investigate the fading channel model with finite battery capacity where energy overflow may occur. The algorithm needs to schedule with the input of varying channel gains and dynamic energy arrivals. We will develop a logarithmic competitive algorithm.

Algorithm PARTITION presents the details of the proposed algorithm. The idea is to partition the channel gains that arrive over time into $L = \lceil \log \frac{h_{max}}{h_{min}} \rceil$ levels where level l is composed of the channel gains ranging in $[h_{min}2^{l-1}, h_{min}2^l)$. The channel gains in the same level l differ at most by a factor of 2 and this allows us to treat these channel gains to be the smallest one $h_{min}2^{l-1}$ which will not lose too much competitive ratio in later analysis. The harvested energy and the battery capacity are partitioned equally into L parts.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

Denote $\frac{\mathcal{H}}{L}$ to be one of the partitioned harvesting instance where $\frac{\mathcal{H}}{L} = \{(\frac{E_i}{L}, t_i), 1 \le i \le m\}$. Then, the channel gains in the same level l are treated as fixed value $h = h_{min}2^{l-1}$ as in the static channel model.

However, due to the partition strategy applied above, each level has unknown number of time slots. Fortunately, we have derived an online algorithm CUMULATIVE-GUESSING for static channel model with unknown number of time slots in the previous section, which will be denoted as CUMULATIVE-GUESSING($\frac{\mathcal{H}}{L}, \frac{B}{L}, h$), where h is the static channel gain. Thus, we can tackle each level of channel gains with the support of $\frac{B}{L}$ battery capacity and the partitioned harvesting instance $\frac{\mathcal{H}}{L}$ in an online manner. That is, run procedure CUMULATIVE-GUESSING($\frac{\mathcal{H}}{L}, \frac{B}{L}, h_{min}2^{l-1}$) on the arrival of channel gains in level l.

Algorithm 3 PARTITION

1: partition the channel gains arrived over time into $L = \lceil \log \frac{h_{max}}{h_{min}} \rceil$ levels where level l is composed of the channel gains ranging in $[h_{min}2^{l-1}, h_{min}2^{l})$. Treat the channel gains in level l to be $h_{min}2^{l-1}$.
2: divide the harvested energy over time and battery capacity equally into L parts. Denote $\frac{\mathcal{H}}{L}$ to be one of the partitioned harvesting instance $\frac{\mathcal{H}}{L} = \{(\frac{E_i}{L}, t_i), 1 \le i \le m\}$.
3: **for** on the arrival of a new channel gain h_t at time t **do**
4:   **if** the channel gain belongs to level l **then**
5:     run CUMULATIVE-GUESSING($\frac{\mathcal{H}}{L}, \frac{B}{L}, h_{min}2^{l-1}$) for that level.
6:   **end if**
7: **end for**

We will prove that the proposed algorithm is optimal with respect to competitiveness in the following subsections. We will first prepare a bound between the optimal solution for the original harvesting instance/capacity and a partitioned harvesting instance/capacity, and then, based on this bound, we will derive the competitiveness of the online algorithm.

A. Optimal properties of partitioned instance

Now we examine the optimal solution for the partitioned instance. Denoted by $OPT(\mathcal{H}, B, h)$ the optimal solution for maximizing the throughput with harvestings $\mathcal{H}$, battery capacity B, and static channel gain h. Denote $\frac{\mathcal{H}}{L}$ to be the partitioned harvesting instance $\frac{\mathcal{H}}{L} = \{(\frac{E_i}{L}, t_i), 1 \le i \le m\}$. Similarly, denote by $OPT(\frac{\mathcal{H}}{L}, \frac{B}{L}, h)$ the optimal solution for maximizing the throughput with harvestings $\frac{\mathcal{H}}{L}$ and capacity $\frac{B}{L}$. We will develop an optimal algorithm COMPARE-DENSITY to build the critical relationship between the optimal solution $OPT(\mathcal{H}, B, h)$ and the optimal solution for the partitioned instance $OPT(\frac{\mathcal{H}}{L}, \frac{B}{L}, h)$.

We will claim that $OPT(\mathcal{H}, B, h)$ (and $OPT(\frac{\mathcal{H}}{L}, \frac{B}{L}, h)$) can be characterized by the schedule returned by Algorithm COMPARE-DENSITY$(0, T, \mathcal{H}, B)$ (and COMPARE-DENSITY$(0, T, \frac{\mathcal{H}}{L}, \frac{B}{L})$). We note that Tutuncuoglu *et al.* [10] provides an algorithm to compute the optimal solution $OPT(\mathcal{H}, B, h)$. Our optimal algorithm COMPARE-DENSITY is developed independently with a different method by comparing the energy density and iteratively deleting the intervals, which provides a simple way to prove the relationship between $OPT(\mathcal{H}, B, h)$ and $OPT(\frac{\mathcal{H}}{L}, \frac{B}{L}, h)$ in Lemma 5, since scaling down the harvested energy in the input just affects the density of intervals selected in the iterations.

Given an interval (t_i, T), we define the *mix-density* to be

$$mix\_density(t_i, T) = g(\frac{\sum_{k \ge i} E_k}{T - t_i}). \tag{35}$$

Given an interval (t_i, t_j) with $t_j < T$, define *minus-density* to be

$$minus\_density(t_i, t_j) = g(\frac{\sum_{i \le k \le j} E_k - B}{t_j - t_i}). \tag{36}$$

Algorithm COMPARE-DENSITY works as follows. It compares mix-density and minus-density to find the interval with the largest rate in the optimal solution. Observe that the energy harvested at time outside the interval with maximum rate will not be allocated to that interval in the optimal solution. Algorithm COMPARE-DENSITY deletes the maximum rate interval, updates the remaining harvestings, and iteratively computes the largest rate in the remaining intervals till the schedule in all intervals is fixed. The proof of its optimality is presented in Appendix.

The following lemma states the desired relationship between the optimal solution $OPT(\mathcal{H}, B, h)$ and $OPT(\frac{\mathcal{H}}{L}, \frac{B}{L}, h)$.

Algorithm 4 COMPARE-DENSITY$(t_{min}, t_{max}, \mathcal{H}, B)$

1: **while** there are some epoches not fixed yet **do**
2:   compute all the possible minus-densities of the intervals, $minus\_density(t_i, t_j) = g(\frac{\sum_{i \le k \le j} E_k - B}{t_j - t_i}), t_{min} \le t_i \le t_{max}, t_{min} \le t_j \le t_{max}$.
3:   compute all the possible mix-densities of the intervals, $mix\_density(t_i, t_{max}) = g(\frac{\sum_{t_i \le t_k < t_{max}} E_k}{t_{max} - t_i}), t_{min} \le t_i \le t_{max}$.
4:   find the interval, say (t_u, t_v), of which the corresponding density achieves the value $\max_{t_{min} \le t_i, t_j \le t_{max}} \{mix\_density(t_i, t_{max}), minus\_density(t_i, t_j)\}$.
5:   **if** (t_u, t_v) is selected from mix-densities **then**
6:     fix interval $[t_u, t_v = t_{max})$ and transmit at rate $mix\_density(t_u, t_v)$ in that interval.
7:     update $E_u = 0$ and treat the remaining harvestings in interval $[t_{min}, t_u)$ as new input instance.
8:     run COMPARE-DENSITY$(t_{min}, t_u, \mathcal{H}, B)$.
9:   **end if**
10:   **if** (t_u, t_v) is selected from minus-densities **then**
11:     fix interval $[t_u, t_v)$ and transmit at rate $minus\_density(t_u, t_v)$ in that interval.
12:     update $E_u = 0, E_v = B$.
13:     treat the refined harvestings in the remaining intervals $[t_{min}, t_u), [t_v, t_{max})$ as new input instances.
14:     run COMPARE-DENSITY$(t_{min}, t_u, \mathcal{H}, B)$, COMPARE-DENSITY$(t_v, t_{max}, \mathcal{H}, B)$ on the new input instances.
15:   **end if**
16: **end while**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

Lemma 5. $OPT(\mathcal{H}, B, h) \leq L \cdot OPT(\frac{\mathcal{H}}{L}, \frac{B}{L}, h)$ *with* $L \geq 1$.

Proof. Since COMPARE-DENSITY returns the optimal solution, in order to build the desired relationship, it is sufficient to compare the schedule returned by the optimal algorithm COMPARE-DENSITY$(0, T, \mathcal{H}, B)$ and COMPARE-DENSITY$(0, T, \frac{\mathcal{H}}{L}, \frac{B}{L})$. When running COMPARE-DENSITY$(0, T, \frac{\mathcal{H}}{L}, \frac{B}{L})$, the E_i amount of energy harvested at t_i and the battery capacity are simultaneously scaled down by a factor L compared to running COMPARE-DENSITY$(0, T, \mathcal{H}, B)$.

In the first iteration, we notice that the intervals found in Step 4 in these two algorithms are always identical since the interval that achieves the density $\max_{t_{min} \leq t_i, t_j \leq t_{max}} \{mix\_density(t_i, t_{max}), minus\_density(t_i, t_j)\}$ is not affected by the scaling operation. This makes the intervals found in the second iteration keep the same, which also holds for all intervals found in later iterations. Although the intervals found in each iteration are identical, the amount of energy invested at each time in the same interval is scaled down by a factor L. According to the concavity of the power-rate function, the rate $g(p_t) \leq L \cdot g(\frac{p_t}{L})$ when scaling down p_t at time t by a factor L with $L \geq 1$. Summing up the rates over all time slots would derive the desired bound

$$OPT(\mathcal{H}, B, h) \leq L \cdot OPT(\frac{\mathcal{H}}{L}, \frac{B}{L}, h). \tag{37}$$

□

B. Performance analysis

Now we are ready to derive the logarithmic competitiveness of Algorithm PARTITION. The idea is to merge the performance achieved by the optimal algorithm and the online algorithm running on the partitioned instance. In fact, PARTITION is asymptotically optimal competitive, because even for non-energy-harvesting systems, it is proved that any online algorithm is $\Omega(\log(\frac{h_{max}}{h_{min}}))$-competitive for maximizing the throughput [19].

Theorem 3. *Algorithm* PARTITION *is optimal* $\Theta(\log(\frac{h_{max}}{h_{min}}))$*-competitive for the throughput maximization problem in fading channels with logarithmic power-rate function.*

Proof. We prepare three bounds first. Assume that $OPT(\mathcal{H}, B)$ is the optimal offline throughput with the input of harvestings $\mathcal{H}$ and capacity B. Let $OPT_j(\mathcal{H}, B)$ be the optimal achievable throughput over the channel gains in level j with known harvestings $\mathcal{H}$ and capacity B. Obviously, we have the first bound

$$OPT(\mathcal{H}, B) \leq \sum_{j=1}^{L} OPT_j(\mathcal{H}, B). \tag{38}$$

Let $ALG_j(\frac{\mathcal{H}}{L}, \frac{B}{L})$ be the throughput achieved by the channel gains in level j with partitioned harvestings $\frac{\mathcal{H}}{L}$ and capacity $\frac{B}{L}$. Let $ALG(\mathcal{H}, B)$ be the throughput returned by Algorithm PARTITION. Then,

$$ALG(\mathcal{H}, B) = \sum_{j=1}^{L} ALG_j(\frac{\mathcal{H}}{L}, \frac{B}{L}). \tag{39}$$

By applying the constant competitiveness of Algorithm CUMULATIVE-GUESSING, we have the second bound

$$OPT_j(\frac{\mathcal{H}}{L}, \frac{B}{L}, h_{min}2^{j-1}) \leq O(1) ALG_j(\frac{\mathcal{H}}{L}, \frac{B}{L}), \tag{40}$$

where $OPT_j(\frac{\mathcal{H}}{L}, \frac{B}{L}, h_{min}2^{j-1})$ is the optimal achievable throughput of the channel gains in level j (which are treated as the smallest value $h_{min}2^{j-1}$ in that level) with partitioned harvestings $\frac{\mathcal{H}}{L}$ and capacity $\frac{B}{L}$.

Recall $L = O(\log(\frac{h_{max}}{h_{min}}))$. Since the channel gains in $OPT_j(\mathcal{H}, B)$ are at most $h_{min}2^j$, we have

$$OPT_j(\mathcal{H}, B) \leq OPT_j(\mathcal{H}, B, h_{min}2^j). \tag{41}$$

Moreover, when the channel gains differ by a factor of 2, it is true that

$$OPT_j(\mathcal{H}, B, h_{min}2^j) \leq 2OPT_j(\mathcal{H}, B, h_{min}2^{j-1}). \tag{42}$$

Recall Lemma 5 that states the relationship

$$OPT_j(\mathcal{H}, B, h_{min}2^{j-1}) \leq L \cdot OPT_j(\frac{\mathcal{H}}{L}, \frac{B}{L}, h_{min}2^{j-1}), \tag{43}$$

which are the optimal solutions under the same channel gain $h = h_{min}2^{j-1}$. Combining these inequalities, we have the third bound

$$\begin{aligned} OPT_j(\mathcal{H}, B) &\leq OPT_j(\mathcal{H}, B, h_{min}2^j) && (44) \\ &\leq 2OPT_j(\mathcal{H}, B, h_{min}2^{j-1}) && (45) \\ &\leq 2L \cdot OPT_j(\frac{\mathcal{H}}{L}, \frac{B}{L}, h_{min}2^{j-1}). && (46) \end{aligned}$$

Merging the three bounds (38)(40)(46) above, we have

$$\begin{aligned} OPT(\mathcal{H}, B) &\leq \sum_{j=1}^{L} OPT_j(\mathcal{H}, B) && (47) \\ &\leq 2L \sum_{j=1}^{L} OPT_j(\frac{\mathcal{H}}{L}, \frac{B}{L}, h_{min}2^{j-1}) && (48) \\ &\leq O(1) \cdot 2L \sum_{j=1}^{L} ALG_j(\frac{\mathcal{H}}{L}, \frac{B}{L}) && (49) \\ &= O(\log(\frac{h_{max}}{h_{min}})) ALG(\mathcal{H}, B). && (50) \end{aligned}$$

This has derived the final $O(\log(\frac{h_{max}}{h_{min}}))$-competitiveness of the online algorithm.

Recall that even for non-energy-harvesting systems, any online algorithm is $\Omega(\log(\frac{h_{max}}{h_{min}}))$-competitive for maximizing the throughput with dynamic channel gains [19]. Therefore, our online algorithm is $\Theta(\log(\frac{h_{max}}{h_{min}}))$-competitive and asymptotically optimal. □

Discussion 2: PARTITION *is the first logarithmic competitive algorithm for throughput maximization in energy harvesting systems, improving upon the* T*-competitive algorithm REPA in [22].* PARTITION *is more flexible and efficient than REPA in the following aspects: (1) REPA relies on the information* T *of future time slots and hence is a pseudo-online algorithm, while* PARTITION *is not resorting to any future information and hence more practical and flexible; (2) for*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

the worst-case performance bound, REPA is linear competitive (close to the number of time slots), while PARTITION *is logarithmic competitive and thus asymptotically improves upon REPA; (3) the factor* $\log(\frac{h_{max}}{h_{min}})$ *is usually bounded and small since there is a smallest acceptable rate or upper limit for the channel gain in the transmission channel, while the factor* T *can approach infinity as time goes by.*

Discussion 3: *Note that the upper bound of logarithmic competitive ratio (*$O(\log(\frac{h_{max}}{h_{min}}))$*-competitive) derived for algorithm* PARTITION *does not contradict the linear lower bound (*$\Omega(T)$*-competitive) derived in [22], since the proof in [22] assumes that the channel gain is a function of the number of time slots* T*, while our upper bound is derived with respect to the varying range of channel gains. Moreover, the* $O(\log(\frac{h_{max}}{h_{min}}))$*-competitiveness of* PARTITION *verifies that the fading coefficient information can be important and properly utilized to derive more efficient online algorithms with better theoretical guarantees. This provides a new viewpoint different from [22], which does not use the channel gain information in algorithm REPA and states that "the optimal competitive ratio is invariant to the availability of the information about the past/present fading coefficients, and shows that the causal fading coefficient information is actually not useful".*

VII. SIMULATIONS

Our theoretical analysis has bounded the worst-case performance of our proposed online algorithms. In this section, we conduct simulations and examine the average performance of our online algorithms, CUMULATIVE-GUESSING and PARTITION.

First, we perform simulations on Algorithm CUMULATIVE-GUESSING for the static channel model. We compare the results with the optimal offline solution OPT (which can be computed by Algorithm COMPARE-DENSITY), since no online algorithms have addressed the throughput maximization problem in the same setting with battery capacity and unknown number of future time slots for static channels. We conduct the simulation with the power-rate function in AWGN channels. The energy harvesting occurrence time is assumed to be a uniform random integer between 1 and 500. The amount of harvested energy is assumed to follow uniform distribution $U(1, 20)$. The battery capacity is set to be $B = 100$. Fig. 3 demonstrates the simulation results of our online algorithm. Each point in the figure is a mean value of 100 random instances where h is a fixed value chosen from normal distribution $N(4.65, 4)$ in each instance. The ratio between the throughput achieved by the online algorithm and the optimal solution is stable and varies in range $[1.40, 1.75]$, which validates the efficiency of our online algorithm.

Then, we examine the average performance of our online algorithm PARTITION in the fading channel model. Algorithm PARTITION works with unknown future time slots, while in prior work, algorithm REPA works with known number of future time slots T [22]. PARTITION does not rely on the information T of future time slots and thus generalizes the setup of REPA. To directly compare these two algorithms is unfair to PARTITION. We notice that an intuitive way to apply

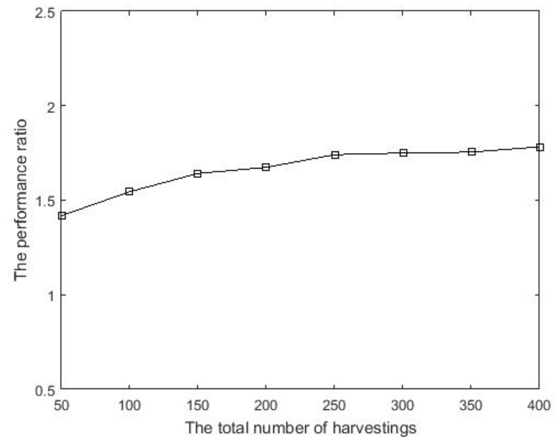

Fig. 3. The performance ratio of the throughput achieved by the optimal offline solution OPT to that of CUMULATIVE-GUESSING.

REPA without the information T is to guess the number of future time slots, say T_{guess}, and average the available energy over the remaining time slots by T_{guess}, where similar idea is adopted to the throughput maximization problem in non-energy-harvesting systems [19]. We thus compare the results of PARTITION to that of REPA by examining the influence of its guessing accuracy T/T_{guess}. In this simulation, energy harvesting occurrence time is assumed to be a uniform random integer between 1 and 500. The amount of harvested energy is assumed to follow uniform distribution $U(1, 20)$. The channel gain is assumed to follow normal distribution $N(4.65, 4)$ with range in $[0.1, 9.2]$. The battery capacity is set to be $B = 100$ and the number of harvestings is 200. Fig. 4 demonstrates the simulation results of our online algorithm where each point is a mean value of 100 random instances. The solid line shows the ratio between the throughput achieved by PARTITION and REPA when $T = 1000$. It can be seen that our algorithm approaches REPA when the guessing accuracy is high and outperforms REPA when the guessing accuracy is less than 0.7. Moreover, when the number of time slots is increased to 2000 (dotted line), our algorithm can even obtain higher throughput than REPA. This is possibly because REPA is too pessimistic to be adaptive to the change of channel gains in large number of time slots. This shows that our online algorithm PARTITION is efficient and can outperform REPA in general setting since PARTITION does not rely on the information of future time slots, while REPA is pseudo-online due to the dependence on the number of future time slots and actually the number of transmission time slots is hard to predict in nature.

Together with the theoretical worst-case guarantees, the simulation results on the average performance validate the efficiency of our online algorithms.

VIII. CONCLUSION

This paper studies the fundamental online throughput maximization problem in battery-capacitated energy harvesting systems. We address the trade-off between energy overflow and energy shortage, and model the fully-online setup without

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2539163, IEEE Transactions on Mobile Computing

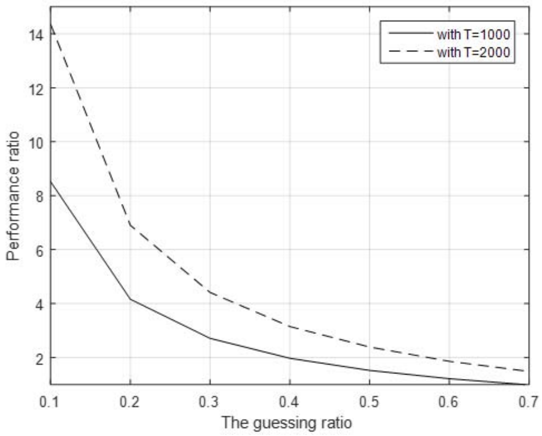

Fig. 4. The performance ratio of PARTITION to REPA.

assuming any information of future time slots, energy arrivals or channel gains. We develop the first online algorithm with proven constant competitive ratio for static channels. Adopting it as a building block, we propose an online algorithm which is proved optimal $\Theta(\log(\frac{h_{max}}{h_{min}}))$-competitive for fading channels, improving the linear T-competitive algorithm in prior work that depends on the time slot information T. Our simulation results further validate the efficiency of our online algorithms.

REFERENCES

[1] A. Jalali, R. Padovani, R. Pankaj, Data throughput of CDMA-HDR a high effciency high data rate personal communication wireless system, *in Proc. of IEEE VTC*, 2000.

[2] S. Nanda, K. Balachandran and S. Kumar, Adaptation techniques in wireless packet data services, *IEEE Communications Magazine*, vol.38, pp. 54-64, 2000.

[3] Berry R., Gallager R. Communication over fading channels with delay constraints, *IEEE Tran. on Inf. Theory*, vol. 48, no. 5, 2002.

[4] B. Prabhakar, E. Biyikoglu, A. El Gamal, Energy-efficient transmission over a wireless link via lazy packet scheduling, *in Proc. of IEEE INFOCOM*, 2001.

[5] A. Borodin and R. El-Yaniv, Online computation and competitive analysis, *Cambridge University Press*, 1998.

[6] Ulukus, S., Yener, A., Erkip, E., Simeone, O., Zorzi, M., Grover, P., Huang, K. (2015). Energy Harvesting Wireless Communications: A Review of Recent Advances. *IEEE Journal on Selected Areas in Communications*, 33(3), 360-381.

[7] J. Yang and S. Ulukus, Transmission completion time minimization in an energy harvesting system, *in Proc. of IEEE CISS*, 2010.

[8] J. Yang and S. Ulukus, Optimal Packet Scheduling in an Energy Harvesting Communication System, *IEEE Trans. On Comm.*, vol. 60, no 1, pp. 220-230, 2012.

[9] K. Tutuncuoglu, and A. Yener, Short-term throughput maximization for battery limited energy harvesting nodes, *in Proc. of IEEE ICC*, 2011.

[10] K. Tutuncuoglu, and A. Yener, Optimum transmission policies for battery limited energy harvesting nodes, *IEEE Trans. on Wireless Comm.*, no. 11, pp. 1180-1189, 2012.

[11] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener, Resource management for fading wireless channels with energy harvesting nodes, *in Proc. of IEEE INFOCOM*, 2011.

[12] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener. Transmission with energy harvesting nodes in fading wireless channels: Optimal policies, *IEEE J. on Sel. Areas in Comm.*, vol. 29, no. 8, pp. 1732-1743, 2011.

[13] M. A. Antepli, E. Uysal-Biyikoglu, and H. Erkal. Optimal packet scheduling on an energy harvesting broadcast link, *IEEE Journal on Sel. Areas in Comm.*, vol. 29, no. 8, pp. 1721-1731, 2011.

[14] O. Ozel, J. Yang, and S. Ulukus. Optimal broadcast scheduling for an energy harvesting rechargeable transmitter with a finite capacity battery, *IEEE Trans. on Wireless Comm.*, vol. 11, no. 6, pp. 2193-2203, 2012.

[15] H. Erkal, F. Ozcelik and E. Uysal-Biyikoglu, Optimal offline broadcast scheduling with an energy harvesting transmitter, *J. on Wireless Comm. and Networking*, no. 197, 2013.

[16] F. Shan, J. Luo, W. Wu, M. Li, X. Shen. "Discrete Rate Scheduling for Packets with Individual Deadlines in Energy Harvesting Systems." *J. on Selected Areas in Communications(JSAC)*, vol. 33, no. 3 (2015): 438-451.

[17] M. Gatzianas, L. Georgiadis, and L. Tassiulas, Control of wireless networks with rechargeable batteries, *IEEE Trans. on Wireless Comm.*, vol. 9, no. 2 pp. 581-593, 2010.

[18] V. Sharma, U. Mukherji, V. Joseph, S. Gupta, Optimal energy management policies for energy harvesting sensor nodes. *IEEE Trans. on Wireless Comm.*, 9(4), 1326-1336, 2010.

[19] N. Buchbinder, L. Lewin-Eytan, I. Menache, S. Naor and A. Orda, Dynamic power allocation under arbitrary varying channels - an online approach, *in Proc. of IEEE INFOCOM*, 2009.

[20] N. Buchbinder, L. Lewin-Eytan, I. Menache, S. Naor and A. Orda, Dynamic power allocation under arbitrary varying channels - the multi-user case, *in Proc. of IEEE INFOCOM*, 2010.

[21] R. Vaze, Competitive ratio analysis of online algorithms to minimize packet transmission time in energy harvesting communication system, *in Proc. of IEEE INFOCOM*, 2013.

[22] R. Vaze, R. Garg, and N. Pathak, Dynamic Power Allocation for Maximizing Throughput in Energy-Harvesting Communication System, *IEEE/ACM Trans. on Netw.*, vol. 22, no. 5, pp. 1621-1630, 2014.

Weiwei Wu received his BSc degree in South China University of Technology and the PhD degree from City University of Hong Kong (CityU, Dept. of Computer Science) and University of Science and Technology of China (USTC) in 2011, and went to Nanyang Technological University (NTU, Mathematical Division, Singapore) for post-doctorial research in 2012. He is currently an associate professor in Southeast University, P.R. China. His research interests include wireless communications, cloud computing, optimizations and algorithm analysis, game theory and network economics.

Jianping Wang received the BSc and MSc degrees in computer science from Nankai University, Tianjin, China in 1996 and 1999, respectively, and the PhD degree in computer science from the University of Texas at Dallas in 2003. She is currently an associate professor at the Department of Computer Science, City University of Hong Kong. Her research interests include dependable networking, wireless networking, optical networking, cloud computing, service oriented networking, data center networks, integration of Fiber-Wireless networks, and network coding. She is currently an associate editor of IEEE Transactions on Mobile Computing.

Xiumin Wang received her B.S. from the Department of Computer Science, Anhui Normal University, China, in 2006. She received the Ph.D. degree from both the Department of Computer Science, University of Science and Technology of China, Hefei, China, and the Department of Computer Science, City University of Hong Kong, under a joint PhD program. She is now an Associate Professor in School of Computer and Information, Hefei University of Technology, Hefei, China. Her research interests include wireless networks, routing design, and network coding.

Feng Shan received his B.S. degree in Computer Science from Hohai University, Nanjing, China, in 2008. He is currently pursuing the Ph.D. degree in computer science and engineering at Southeast University, Nanjing, China. He was a Visiting Scholar at the School of Computing and Engineering, University of Missouri-Kansas City, Kansas City, MO, United States, from 2010 to 2012. His research interests are in the areas of energy harvesting, wireless power transfer, algorithm design and analysis.

Junzhou Luo received his B.S. degree in Applied Mathematics from Southeast University in 1982. From then on he has been a faculty member at the School of Computer Science and Engineering, Southeast University. He got his M.S. and Ph.D. degree in Computer Network both from Southeast University in 1992 and in 2000, respectively. He now is a full professor and Dean at the School of Computer Science and Engineering in Southeast University in China. He is an IEEE member and co-chair of IEEE SMC Technical Committee on Computer Supported Cooperative Work in Design. His research interests are the next generation network architecture, protocol engineering, network security and management, wireless LAN, and grid and service computing.