

Article

A Robust Transmission Scheduling Approach for Internet of Things Sensing Service with Energy Harvesting

Jie Hao ^{1,2,†} , Jing Chen ^{1,†}, Ran Wang ^{1,2} , Yi Zhuang ¹  and Baoxian Zhang ^{3,*} 

¹ College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

² Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 210023, China

³ School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China

* Correspondence: bxzhang@ucas.ac.cn; Tel.: +86-010-88256565

† These authors contributed equally to this work.

Received: 24 May 2019; Accepted: 10 July 2019; Published: 12 July 2019



Abstract: Maximizing the utility under energy constraint is critical in an Internet of Things (IoT) sensing service, in which each sensor harvests energy from the ambient environment and uses it for sensing and transmitting the measurements to an application server. Such a sensor is required to maximize its utility under the harvested energy constraint, i.e., perform sensing and transmission at the highest rate allowed by the harvested energy constraint. Most existing works assumed a sophisticated model for harvested energy, but neglected the fact that the harvested energy is random in reality. Considering the randomness of the harvested energy, we focus on the transmission scheduling issue and present a robust transmission scheduling optimization approach that is able to provide robustness against randomness. We firstly formulate the transmission scheduling optimization problem subject to energy constraints with random harvested energy. We then introduce a flexible model to profile the harvested energy so that the constraints with random harvested energy are transformed into linear constraints. Finally, the transmission scheduling optimization problem can be solved traditionally. The experimental results demonstrate that the proposed approach is capable of providing a good trade-off between service flexibility and robustness.

Keywords: energy harvesting; transmission scheduling; Internet of Things; robustness; uncertainty

1. Introduction

The Internet of Things (IoT) has drawn much attention due to its massive applications. One important IoT application is environment sensing, such as climate change sensing and water quality monitoring. In such IoT applications, a number of IoT devices equipped with sensors (we refer to the IoT devices as sensors hereafter) with wireless communication interfaces are deployed to sense the environment and report the readings to the application server. The communication network is typically a single-hop network with a star typology. Low-Power Wide Area Network (LPWAN) technologies including IEEE 802.11ah, Long Range (LoRa), SigFox, Narrow Band (NB)-IoT, and so on, can be used for single-hop communication. Recently, there has been growing concern about energy efficiency for IoT services [1]. Energy harvesting is one promising method to address this concern.

An energy-harvesting sensor can harvest ambient energy (e.g., solar, wind, thermal, vibration, etc.) to recharge its battery and thus is capable of maintaining an eternal lifetime. This allows sensors to be suitable to be deployed in tough circumstances. The main design objective for an energy-

harvesting sensor is to fulfill its functionalities by adequately utilizing the harvested energy, i.e., achieve the maximum utility subject to the harvested energy constraint. The objective involves two main elements, utility and energy constraint. Utility can stand for a number of performance metrics, such as duty cycle [2,3], sensing coverage [4], throughput [5–7], data quality [8–10], decision fusion [11–13], and so on. The energy constraint regarding energy harvesting is well known as Energy Neutral Operation (ENO), that is the consumed energy of a sensor should not exceed the harvested energy after a finite horizon. In environment-sensing IoT applications, the main functionalities of a sensor are to sense environment and transmit the readings. The more readings transmitted, the higher sensing quality can be obtained. Therefore, we focus on the transmission scheduling design that aims to achieve the maximum data quality subject to the harvested energy constraint.

However, there exists a big challenge in the transmission scheduling design, i.e., the unknown harvested energy profile. Although there has been extensive research on profiling the harvested energy, either predicting the harvested energy or modeling it as a random process, the harvested energy profile cannot be precise, and the impact of the inaccurate harvested energy profile on the performance has not been considered. Figure 1 shows the deviation of a solar energy model, Exponentially-Weighted Moving-Average (EWMA), proposed in [14] from the ground truth. The uncertainty in harvested energy makes the transmission scheduling design based on an exact energy model stray from the design objective. To improve the robustness against the harvested energy randomness, instead of performing transmission scheduling based on an exact harvested energy model, this paper presents an Uncertain Harvested Energy-based Transmission Scheduling approach (UHETS), which considers the harvested energy to be of a distribution fluctuating around an energy model. In this way, UHETS is able to achieve high robustness against the inaccuracy of the harvested energy model. The main contributions of this paper include:

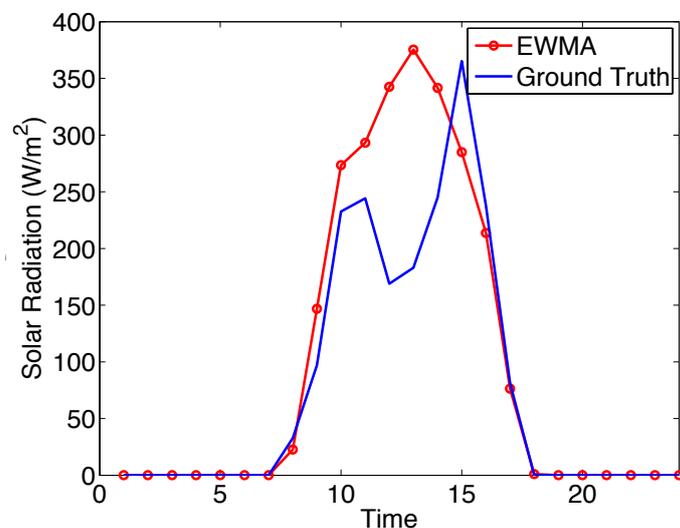


Figure 1. The energy model deviates from the ground truth. EWMA, Exponentially-Weighted Moving-Average.

- We formulate the transmission scheduling problem as a utility optimization problem subject to the energy constraints.
- We introduce a flexible model to describe the harvested energy, which is robust against the inaccuracy of an exact energy model.
- We conduct extensive experiments on real-world data and explore the impacts of the prediction model and various parameters; numerical results demonstrate the robustness of the proposed scheduling approach.

The remainder of this paper is organized as follows. Section 2 introduces related work. In Sections 3 and 4, we present the proposed scheduling approach and adaptive transmission approach, respectively. Section 5 summarizes the presented approaches. The evaluation results are shown in Section 6 followed by the conclusion in Section 7.

2. Related Work

In this section, we introduce the related literature and focus on the interplay between the utility optimization and the energy profile. We categorize the related work into three classes.

2.1. Utility Optimization without Prior Knowledge of the Harvested Energy Profile

In this category, utility optimization is performed mainly according to the available energy. Vigorito et al. [14] aimed to optimize the average duty cycle and minimize the variance while maintaining ENO. They proposed a harvested energy model-free approach that kept track of the battery level instead of the harvested energy level and made adaptive duty cycle decisions based on adaptive control theory. Sharma et al. [7] claimed that the greedy policy that transmits the most data bits allowed by available energy had promising performance in terms of both throughput and mean delay. Yoon et al. [15] compared the harvested energy and its desired energy level and used the excess energy to compress and transmit data packets. As these approaches only considered the current available energy, they cannot guarantee global optimization. Djenouri et al. [16] treated the energy-harvesting sensor as an unlimited energy sensor and only used harvesting nodes as relay nodes to maintain network connectivity. The connectivity problem is equivalent to finding the minimum weighted connected dominating set in a vertex weighted graph. This assumption is impractical in many cases. For example, solar energy on a rainy day might be insufficient for full-time operation.

2.2. Utility Optimization Based on a Known Harvested Energy Profile

Simultaneous Wireless Information and Power Transfer (SWIPT), as a major type in this category, mainly addresses the resource allocation problem as an optimization problem subject to a energy harvesting model. The main effort is made to solve the optimization problem, for example to overcome the non-convexity of the optimization. Conventionally, the harvested energy is assumed as a linear function of input energy [17–20]. Based on the linear model, the energy/information efficiency is optimized, including the Rate-Energy (R-E) tradeoff in [17], the maximum achievable energy efficiency in [19], the users' weighted sum uplink rate in [20], and the joint optimization of transmit power energy harvesting efficiency and interference power leakage-to-transmit power ratio in [18]. Vamvakas et al. [21] consider users' actual needs in energy consumption during the wireless energy transfer phase and solved the optimization problem of each user's utility function with respect to the optimal customized transmission power. However, the linear model is only applicable for the specific scenario in which the received powers are constant. In practice, the work in [22] showed that the wireless energy conversion efficiency varies with different input power energy levels. Based on the non-linear energy harvesting model, the resource allocation problem will be more complex [23–26]. In [23], the cross-layer multi-level optimization to maximize the power efficiency, user fairness, and channel non-reciprocity was studied. Boshkovska et al. [24] took the uncertainty of the Channel State Information (CSI) into account and addressed robust sum throughput maximization and maximization of the minimum individual throughput against imperfect CSI knowledge. The work in [25] aimed to minimize the transmit power in the scenario when multiple heterogeneous users existed. Xiong et al. [26] followed the work of [17], but derived the boundaries of the R-E region for MIMO broadcasting channels. For more literature surveys on SWIPT, please refer to [27].

Gunduz et al. [28] focused on offline transmission scheduling for both half-duplex and full-duplex two-hop communication scenarios and assumed the energy was harvested at known instants. Fan et al. [29] firstly used a known prediction model for the energy profile and modeled a multiple-hop network as a node-weighted space-time graph. Upon such a graph, the awake sensor node set

with minimum cost was found to maintain the network connectivity. However, the simulation was conducted based on a randomly-generated topology instead of using an energy prediction model.

Maemoto et al. [30] considered the non-uniform harvesting rates for the sensor nodes and proposed a clustering control scheme, which consisted of adaptive cluster size control and adaptive backoff window control. EWMA [2] and its variances are widely used as (solar) energy prediction models. In EWMA, the predicted harvested energy is a weighted combination of historical data of previous days. Using EWMA, the work in [2] formulated the energy allocation problem as a duty cycle maximization problem subject to energy management constraints. Apparently, the assumption of the known harvested energy profile is not realistic since the energy harvesting process is naturally random.

2.3. Utility Optimization Assuming the Energy Harvesting Process as a Random Process

To address the uncertainty of harvested energy, some recent work modeled the harvested energy as a random process. The Markov chain is a promising tool for modeling a random process [31,32]. Tunc et al. [31] assumed a finite-state Continuous-Time Markov Chain (CTMC) model for characterizing the energy harvesting process and proposed an adaptive sensing policy that maximized the sensing rate of an IoT device while meeting the first battery outage probability constraint. Gorlatova et al. [33] treated the independent and identically distributed random harvested energy as a Markov decision process and developed an algorithm for calculating the maximum energy spending rate for a sensor node. The Markov model was applied for decentralized hypothesis testing [34] as well. In [34], the sensors made observations of a phenomenon and performed transmission scheduling to minimize the error probability of the hypothesis on the target phenomenon.

Ho et al. [6] aimed to maximize the throughput for point-to-point, flat-fading, single-antenna communication via power allocation. The energy harvesting and the time-varying wireless channel were modeled as a random process with a joint distribution to address their unpredictable nature. Huang et al. [35] assumed an energy harvesting model under which the energy arrival time and the harvested amount were known prior to transmissions and addressed the throughput maximization problem under the Gaussian relay channel. Yadav et al. [36] focused on the problem that the randomness and arrivals of harvested energy may induce interruptions to single-hop communications. To overcome the interruptions, the work in [36] assumed that the arrival of harvested energy followed a Bernoulli random process and proposed an adaptive retransmission scheme. The retransmission scheme can highly improve the communication reliability against the uncertainty of harvested energy. Aman et al. [2] modeled the solar energy profile by using the average harvesting rate and lower/upper bound. In order to guarantee ENO, the lower bound of harvested energy was used to maximize the duty cycle of a sensor node. This lower bound-based guarantee would incur inefficient energy usage and degraded utility. Chamanian et al. [37] focused on vibration energy and maximized the duty cycle subject to ENO. Hao et al. [3] adopted a robust energy model in which the harvested energy fluctuated around a reference distribution to pursue the maximum duty cycle. However, they did not consider the fact that the uncertainty of cumulative harvested energy might change, and their proposed duty cycle maximization approach may result in an overly strict energy maintaining policy. Although the randomness of harvested energy was considered in the above work, the impact of the deviation of a hypothetical random energy model from the ground truth on the network performance was not studied.

3. Robust Transmission Scheduling

This section will describe how to formulate the transmission scheduling problem and how to solve it in detail.

3.1. Network Model

Consider that in an environment sensing IoT application, each sensor equipped with an energy-harvesting battery with a non-ideal energy buffer senses a random field and generates packets to be transmitted to the application server.

Time is slotted, and each slot lasts for ΔT , while one decision cycle lasts for T . During the t th slot, n samples are supposed to be taken and transmitted by the sensor. However, due to the energy constraint, only a subset of the samples, i.e., $n(t) < n$ samples, is eligible for sensing and transmission in the t th slot. This can be considered as a matrix completion problem. Given an original $n \times T$ data sample matrix $\mathbf{M}_{n \times T}$, for each slot t , only $n(t)$ samples are transmitted to the application server. That is, the transmission scheduling decides an indicator matrix $\mathbf{\Omega}_{n \times T}$, and each entry $\Omega(i, j)$ denotes whether or not the corresponding entry in $\mathbf{M}_{n \times T}$ is selected for transmission. If the entry in location (i, j) is selected: $\Omega(i, j) = 1$, otherwise, $\Omega(i, j) = 0$. Thus, the received data matrix by the application server is:

$$\mathbf{O} = \mathbf{\Omega} \bullet \mathbf{M},$$

where \bullet denotes the Hadamard product (the element-wise product of two matrices). Upon the received matrix $\mathbf{O}_{m \times n}$, the application server then utilizes a matrix completion algorithm to reconstruct a complete matrix $\hat{\mathbf{M}}$. Obviously, the transmission scheduling scheme has a big impact on data recovery quality. Generally speaking, the more transmitted samples, the higher the recovery quality. Therefore, we aimed to design a transmission scheduling scheme that leads to a high recovery quality. The design of the matrix completion method is out of the scope of our work. In this paper, we just adopt a classic algorithm, Singular-Value Thresholding (SVT) [38], as the matrix completion method. Considering that the energy cost for computation is negligible compared with that for communication and sensing (for energy-hungry sensors) [39], we assume that sensing and transmission consume most of the energy in a sensor node and ignore the other consumption (The proposed approach is suitable for the sensor nodes whose energy cost for sensing is trivial. In such cases, the sensor node can sense the environment at a high rate, but only transmits a subset of the sensing readings based on the proposed transmission scheduling.).

Each sensor node in the network has finite battery capacity E_{max} . Let E_t denote the remaining energy of the battery at slot t , $E_h(t)$ denote the harvested energy during slot t , and E_c denote the energy cost for single-packet sensing and transmission. Given E_c , $E_h(t)$ at slot t , then at slot t , the energy buffer will be updated as:

$$E_t = E_0 - \sum_{i=1}^t n(i)E_c + \sum_{i=1}^t E_h(i), \quad (1)$$

with the constraints:

$$0 \leq E_t \quad \forall t \in [0, T] \quad (2)$$

$$E_t \leq E_{max} \quad \forall t \in [0, T] \quad (3)$$

$$n(t) \leq n. \quad (4)$$

Moreover, ENO imposes an additional constraint that over a decision period T , the sensor node is expected to be energy neutral. In other words, the remaining energy of a sensor node should be higher than the initial energy. This can be formulated as:

$$E_T \geq E_0, \quad (5)$$

where T is generally set to be a twenty-four-hour duration if we adopt the solar energy source.

The variables used in this paper are summarized in Table 1.

Table 1. Notations used in this paper.

Symbol	Definition
T	Decision period, $T = 24$ h in this paper
$t \in [1, T]$	Slot
ΔT	Duration of a slot, $\Delta T = 1$ h in this paper
E_{max}	Battery capacity
E_0	Initial battery energy
E_t	Residual battery energy at slot t
E_c	Energy consumption for a single packet
$E_h(t)$	Harvested energy at slot t
$f(*)$	Real-world distribution
$g(*)$	Gaussian distribution
n	The number of high-resolution samples generated at each slot
$n(t)$	The number of samples transmitted in slot t
\mathbf{N}	Transmission decision $\mathbf{N} = [n(1), n(2), \dots, n(T)]$

3.2. Optimization Formulation

In this paper, we aim to optimize the transmission scheduling and further optimize the data recovery quality subject to the energy constraints over the time horizon. To this end, we hoped that the overall transmission number could be as high as possible. The more samples transmitted, the higher the data quality a sensor node can provide. Moreover, we also desired fairness among all slots, that is we absolutely did not expect $n(t) = 0$ for slot t . In the case of a solar harvesting sensor node, there is no harvested energy during the night, which might lead to zero transmissions during the night. Obviously, this will lead to extreme high information loss during the night. Motivated by this, the optimization problem of transmission scheduling subject to the energy constraints is defined as follows:

$$\begin{aligned} \max \quad & \text{sum}(\mathbf{N}) - \text{var}(\mathbf{N}) \\ \text{s.t.} \quad & (1), (2), (3), (4), (5) \end{aligned} \quad (6)$$

where $\mathbf{N} = [n(1), n(2), \dots, n(T)]$, sum denotes the summation operation, and var is the variance of \mathbf{N} . We believe this optimization objective will yield high data reconstruction quality by using matrix completion.

3.3. Handling the Random Variable

As illustrated in Section 1, the constraints involve a random variable $E_h(t)$, which brings in a big challenge to solve the optimization problem (6). In this subsection, we shall illustrate how to handle the uncertainty of $E_h(t)$ in detail.

For a constraint with a random variable, a robust way to guarantee the constraint is to guarantee the worst case probability. For example, if we have a constraint $x > y$ where y is a random variable uniformly distributed within $[0,1]$, the worst case for $x > y$ is when $y = 1$. Therefore, if we replace y with its upper bound of one, we can guarantee $x > y$ with a probability of 100%. However, in most applications, it is too strict or infeasible to guarantee 100%. We can approximately guarantee the original constraint with a probability of $1 - \epsilon$, where $\epsilon \in [0, 1]$. In this case, we can firstly calculate a threshold y^u such that $P(y^u \geq y) = 1 - \epsilon$ and then replace $x > y$ with $x > y^u$. The original constraint is transformed into a traditional linear constraint.

Likewise, we can tackle $E_h(t)$ in the same way. However, we notice that our problem is different from the example in two aspects. One is that E_t involves the sum of $E_h(i)$, $1 \leq i \leq t$. It is too strict to guarantee the worst case probability for every $E_h(i)$, $t \in [1, t]$ considering that the sum of $E_h(i)$ will compensate the other, especially when t is large enough. The other is that the distribution of the random variable in our problem is not known. We had to use a novel model to obtain the threshold

and transform the original constraints to traditional linear constraints. To address these two difficulties, we introduced a new variable $S_h(t) = \sum_{i=1}^t E_h(i)$ and explored the threshold regarding $S_h(t)$.

3.3.1. Transforming the Constraints

We firstly processed the constraint $E_t \geq 0$ in Constraint (2). Constraints $E_t \leq E_{max}$ in (3) and $E_T \geq E_0$ in (5) can be transformed in the same way. The transformation processes of Constraints (3) and (5) are omitted here due to space limitations.

With $S_h(t)$, Equation (1) can be transformed to:

$$E_t = E_0 - \sum_{i=1}^t n(i)E_c + S_h(t). \quad (7)$$

From $E_t \geq 0$, we have:

$$S_h(t) \geq \sum_{i=1}^t n(i)E_c - E_0. \quad (8)$$

We let $s_h(t) = \sum_{i=1}^t n(i)E_c - E_0$ and then obtain:

$$S_h(t) \geq s_h(t). \quad (9)$$

This constraint can be guaranteed with a probability of (higher than) $1 - \epsilon$ by:

$$S_h^u(t) \geq s_h(t) \quad (10)$$

$$P(S_h(t) \geq S_h^u(t)) = 1 - \epsilon, \quad (11)$$

where $P(*)$ denotes the probability of condition $*$ and $\epsilon \in [0, 1]$ is a sufficiently small value. Since the distribution of $S_h(t)$ is unknown, it is difficult to derive the exact value of $P(S_h(t) \geq S_h^u(t))$. Therefore, in order to satisfy Constraint (9) with a probability of (higher than) $1 - \epsilon$, we should guarantee Condition (9) regardless of $f(S_h(t))$ by guaranteeing:

$$\min_{f(S_h(t))} P(S_h(t) \geq S_h^u(t)) = 1 - \epsilon, \quad (12)$$

where $f(S_h(t))$ is the ground truth Probability Density Function (PDF) of $S_h(t)$. In this paper, we use the PDF and distribution interchangeably. We can see that ϵ indicates the robustness level against the uncertainty, that is we can adapt the robustness by tuning ϵ .

In the next step, we need to cope with the unknown $f(S_h(t))$ and calculate the threshold $S_h^u(t)$ to satisfy Equation (12).

3.3.2. Calculating Threshold $S_h^u(t)$

Although there is no precise prediction model or distribution model to describe $S_h(t)$, $S_h(t)$ can be considered to be close to a prediction, and its probability density is close to a known distribution [40]. The former can be demonstrated in Figure 1, in which EWMA can provides a close estimate. Figure 2 shows that the PDF of $S_h(t)$ was close to a Gaussian distribution [40,41]. The data used in Figure 2 were the values of $S_h(10)$ (i.e., the accumulative harvested energy until 10:00) in January from 2001–2019. We omitted the other results since they performed similarly. Therefore, we assumed that the real-world distribution of $S_h(t)$ was close to a Gaussian distribution in which the mean value was a prediction based on some model (e.g., EWMA) and the variance was obtained empirically based on historical data. Moreover, the Kullback–Leibler divergence (KL divergence) between the Gaussian distribution and real-world distribution of $S_h(t)$ for every slot was only about

0.1. Thus, we can use the Gaussian distribution along with its difference from the ground truth to describe $S_h(t)$. In this way, the ground truth distribution of $S_h(t)$ belongs to a domain \mathbf{R} ,

$$\mathbf{R} = \{f(S_h(t)) | \text{Distance between } f(S_h(t)) \text{ and } g(S_h(t)) \text{ is less than a limit}\}, \tag{13}$$

where $g(S_h(t))$ is the empirical Gaussian distribution based on a prediction model. Figure 3 is an illustrative figure to show how we formulated the unknown $f(S_h(t))$. In Figure 3, as we assumed the ground truth distribution was close to a Gaussian distribution, it lied within the dotted region \mathbf{R} in which each distribution had a distance from the Gaussian distribution lower than a limit D_{KL} .

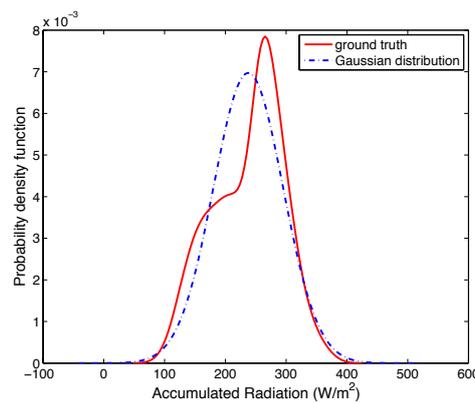


Figure 2. The probability density of $S_h(10)$ is close to a Gaussian distribution.

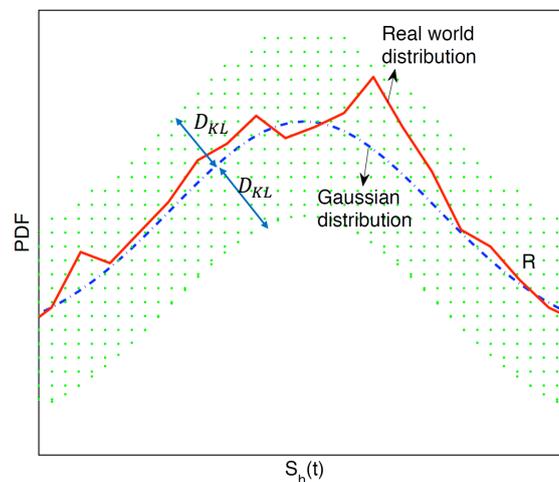


Figure 3. The formulation of $f(S_h(t))$. The red solid curve is the ground truth PDF, which is unknown in practice. The blue dotted curve is an empirical Gaussian distribution. The dotted area denotes \mathbf{R} in which each distribution has a distance from the Gaussian distribution lower than a limit D_{KL} . Since the ground truth distribution is close to the Gaussian distribution, we say that the ground truth distribution belongs to \mathbf{R} . Please note that this figure is for illustrative purposes only.

The distance between the two distributions can be calculated by the KL divergence, which can be formulated as:

$$\mathbb{E}_f[\ln f(S_h(t)) - \ln g(S_h(t))] = \int_0^{+\infty} [\ln f(S_h(t)) - \ln g(S_h(t))] f(S_h(t)) dS_h(t), \tag{14}$$

where $\mathbb{E}_f(\ast)$ denotes the integral of expression \ast with $f(S_h(t))$. As a result, \mathbf{R} can be formulated as:

$$\mathbf{R} = \{f(S_h(t)) | \mathbb{E}_f[\ln f(S_h(t)) - \ln g(S_h(t))] \leq D_{KL}\}, \tag{15}$$

where D_{KL} is the distribution difference limit. The more uncertainty in $S_h(t)$, the less confidence on the Gaussian distribution, and thus, the larger D_{KL} .

Therefore, with $f(S_h(t)) \in \mathbf{R}$, to obtain $S_h^u(t)$ satisfying Equation (12) to obtain $S_h^u(t)$ such that:

$$\min_{f(S_h(t)) \in \mathbf{R}} P(S_h(t) \geq S_h^u(t)) = 1 - \epsilon, \tag{16}$$

which is further equivalent to:

$$\max_{f(S_h(t)) \in \mathbf{R}} P(S_h(t) < S_h^u(t)) = \epsilon. \tag{17}$$

We denoted the left part of Equation (17) as $P_m(S_h^u(t))$ and divided the solution of Constraint (17) into two subproblems, which were calculating $P_m(S_h^u(t))$ as the function of variable $S_h^u(t)$ and calculating $S_h^u(t)$ such that $P_m(S_h^u(t)) = \epsilon$. We first solved the first subproblem, which can be formalized as:

$$P_m(S_h^u(t)) = \max_{f(S_h(t)) \in \mathbf{R}} P(S_h(t) < S_h^u(t)) = \max_{f(S_h(t)) \in \mathbf{R}} \int_0^{S_h^u(t)} f(S_h(t)) dS_h(t), \tag{18}$$

$$\text{s.t. } \mathbb{E}_f[\ln f(P_h(t)) - \ln g(P_h(t))] \leq D_{KL} \tag{19}$$

$$\mathbb{E}_f[1] = 1. \tag{20}$$

In order to avoid the variable in the limits of the integral, $P_m(S_h^u(t))$ in Equation (18) can also be transformed to:

$$\max_{f(S_h(t)) \in \mathbf{R}} \int_0^{+\infty} w(S_h(t), S_h^u(t)) f(S_h(t)) dS_h(t), \tag{21}$$

where $w(S_h(t), S_h^u(t))$ is an auxiliary function as:

$$w(S_h(t), S_h^u(t)) = \begin{cases} 0 & S_h(t) \geq S_h^u(t); \\ 1 & S_h(t) < S_h^u(t). \end{cases} \tag{22}$$

Fortunately, the problems (19)–(21) is convex. The proof of the convexity is shown in Appendix A. Based on the convexity of the optimization problem and Slater’s condition, we can utilize the duality to solve (18)–(20). By applying the Lagrangian method, we can obtain $P_m(S_h^u(t))$ as follows:

$$P_m(S_h^u(t)) = \min_{\alpha, \beta} \max_{f(S_h(t))} \mathbb{E}_f \left[w(S_h(t), S_h^u(t)) - \alpha (\mathbb{E}_f[\ln f(P_h(t)) - \ln g(P_h(t))] - D_{KL}) - \beta (\mathbb{E}_f[1] - 1) \right] \tag{23}$$

$$= \min_{\alpha, \beta} \max_{f(S_h(t))} \mathbb{E}_f \left[w(S_h(t), S_h^u(t)) - \beta - \alpha \ln \frac{f(S_h(t))}{g(S_h(t))} \right] + \alpha D_{KL} + \beta, \tag{24}$$

where $\alpha \geq 0$ and β are Lagrangian multipliers associated with Constraints (19) and (20), respectively. The optimal distribution function $f^*(S_h(t))$ and α^*, β^* for the optimization problem (23) can be derived as in Appendix B in detail. We only present the derivation results here. The optimal distribution function is:

$$f^*(S_h(t)) = g(S_h(t)) \exp \left(\frac{w(S_h(t), S_h^u(t)) - \beta^*}{\alpha^*} - 1 \right), \tag{25}$$

where α^*, β^* are obtained by solving the following.

$$U(S_h^u(t))e^{\beta^*/\alpha^*} + V(S_h^u(t))e^{(1-\beta^*)/\alpha^*} = 0, \quad (26)$$

$$V(S_h^u(t))e^{(1-\beta^*)/\alpha^*} - \beta^* - \alpha^*(1 + D_{KL}) = 0, \quad (27)$$

where $U(S_h^u(t)) = G(S_h^u(t))e^{-1}$, $V(S_h^u(t)) = (1 - G(S_h^u(t)))e^{-1}$, and $G(*)$ is the cumulative distribution of $g(S_h(t))$.

With the obtained α^*, β^* , we then solved the second subproblem, i.e., calculating the threshold $S_h^u(t)$ from $P_m(S_h^u(t)) = \epsilon$.

However, it is difficult to calculate an explicit solution of $S_h^u(t)$ and α^*, β^* by solving Equations (26) and (27). A heuristic algorithm to calculate $S_h^u(t)$ and α^*, β^* is required. We note that $P_m(S_h^u(t))$ is non-decreasing regarding $S_h^u(t)$, and a binary search method is presented in Algorithm 3 in Appendix C to search for proper solutions.

4. Adaptive Transmission Scheduling

UHETS can be performed based on a 24-h solar energy prediction just once per day. As the real-world harvested energy values may vary from the predicted values, we can adapt the transmission scheduling based on the observed harvested energy. One method is to update the predicted energy profile in real time and perform UHETS every slot, which is computationally consumptive, obviously. Another alternative is to adaptively tune the transmission scheduling in a more lightweight way. The basic idea is if $S_h(t) > S_h'(t)$, where $S_h(t)$ is the observed harvested energy and $S_h'(t)$ is the predicted value, the extra energy can be used to increase the transmissions of the latter slots. Otherwise, the transmissions of the latter slots should be decreased to some extent accordingly.

As we hoped to maximize the objective $sum(\mathbf{N}) - var(\mathbf{N})$, if the excessive energy or energy shortage was fixed, the change in $sum(\mathbf{N})$ was fixed as well as $|(S_h(t) - S_h'(t))|/E_c$. Maximizing $sum(\mathbf{N}) - var(\mathbf{N})$ now becomes minimizing $var(\mathbf{N})$, which desires \mathbf{N} to be more even by adjusting the transmission scheduling. The detailed procedure of the adaptive transmission scheduling algorithm based on UHETS (UHETS-adaptation) is presented in Algorithm 1. We take an example to explain the working procedure.

Algorithm 1 UHETS-adaptation

Input: $ct, \mathbf{N} = [n(ct + 1), \dots, n(T)]$, $S_h(ct)$, $S_h'(ct)$

Output: Adjusted transmission decision for the remaining slots, i.e., $\mathbf{N} = [n(ct + 1), \dots, n(T)]$.

- 1: \mathbf{N}_a is sorted \mathbf{N} in the ascending order of the number of samples transmitted.
- 2: \mathbf{Q}_a is the index of \mathbf{N}_a .
- 3: \mathbf{N}_d is sorted \mathbf{N} in the descending order of the number of samples transmitted.
- 4: \mathbf{Q}_d is the index of \mathbf{N}_d .
- 5: **if** $S_h(t) > S_h'(t)$ **then** // If the harvested energy is more than the expectation
- 6: $\Delta S_h = S_h(t) - S_h'(t)$ // excessive energy
- 7: $\Delta n = \Delta S_h / E_c$ // Δn need to be added to \mathbf{N} in total
- 8: **for** $k = 1 \rightarrow T - ct$ **do**
- 9: $Avg = (\sum_{j=1}^k (\mathbf{N}_a(j) + \Delta n) / k$
- 10: **if** $\mathbf{N}_a(k+1) < Avg$ **then** // if involving $\mathbf{N}_a(k+1)$ would make \mathbf{N} more even
- 11: continue; // involve $\mathbf{N}_a(k+1)$ in the adjustment
- 12: **else** // adjust $\mathbf{N}_a(1)$ to $\mathbf{N}_a(k)$
- 13: **for** $j = 1 \rightarrow k$ **do**
- 14: $\mathbf{N}_a(j) = floor(Avg)$ // assign $\mathbf{N}_a(1)$ to $\mathbf{N}_a(k)$ uniformly
- 15: **end for**
- 16: $Residual = mod(\sum_{j=1}^k \mathbf{N}_a(j) + \Delta n, k)$

```

17:         for  $j = k \rightarrow k - Residual$  do
18:              $\mathbf{N}_a(j) = \mathbf{N}_a(k) + 1$  //assign the residual transmissions
19:         end for
20:         for  $k = 1 \rightarrow T - ct$  do
21:              $n(\mathbf{Q}_a(k)) = \mathbf{N}_a(k)$  //restore the order of N
22:         end for
23:         break;
24:     end if
25: end for
26: else // if the harvested energy is less than the expectation
27:      $\Delta S_h = S'_h(t) - S_h(t)$  // energy shortage
28:      $\Delta n = \Delta S_h / E_c$  //  $\Delta n$  need to be deduced from N in total
29:     for  $k = 1 \rightarrow T - ct$  do
30:          $Avg = (\sum_{j=1}^k n(\mathbf{Q}_d(j)) - \Delta n) / k$ 
31:         if  $\mathbf{N}_d(k+1) > Avg$  then //if involving  $\mathbf{N}_d(k+1)$  would make N more even
32:             continue; //involve  $\mathbf{N}_d(k+1)$  in the adjustment
33:         else //adjust  $\mathbf{N}_d(1)$  to  $\mathbf{N}_d(k)$  evenly
34:              $Residual = mod(\sum_{j=1}^k n(\mathbf{Q}_d(j)) - \Delta n, k)$ 
35:             for  $j = 1 \rightarrow k$  do
36:                  $\mathbf{N}_d(j) = ceil(Avg)$ 
37:             end for
38:             for  $j = k + 1 - Residual \rightarrow k$  do
39:                  $\mathbf{N}_d(j) = \mathbf{N}_d(j) + 1$ 
40:             end for
41:             for  $k = 1 \rightarrow T - ct$  do
42:                  $n(\mathbf{Q}_d(k)) = \mathbf{N}_d(k)$  //restore the order of N
43:             end for
44:             break;
45:         end if
46:     end for
47: end if

```

At Slot 19 ($ct = 19$, where ct means the current slot), we observed that more energy was harvested than expectation, and $\Delta n = 4$ transmissions should be added in the subsequent slots (from Slots 20–24). The original transmission decision for the subsequent slots was $\mathbf{N} = [4, 8, 1, 1, 1]$. In order to decrease $var(\mathbf{N})$, a straightforward way is to utilize four to make the elements in \mathbf{N} as even as possible.

We firstly sorted N in an ascending order as $\mathbf{N}_a = [1, 1, 1, 4, 8]$ as shown in Line 1 in Algorithm 1. Starting from one ($k = 1$ in Line 8), we decided if we should add four to $\mathbf{N}_a(1)$ or evenly distribute four to $\mathbf{N}_a(1)$ and $\mathbf{N}_a(2)$ by comparing if $\mathbf{N}_a(2) < (\mathbf{N}_a(1) + 4)/1$. Obviously, the latter will deduce a less $var(\mathbf{N})$ if $\mathbf{N}_a(2) < (\mathbf{N}_a(1) + 4)/1$. We should involve $\mathbf{N}_a(2)$ in the adjustment. That is, Lines 10–11 were triggered, and we proceeded to Line 8 with $k = 2$. Next, we decided if we should involve $\mathbf{N}_a(3)$. As $\mathbf{N}_a(3) < (\mathbf{N}_a(1) + \mathbf{N}_a(2) + 4)/2$, Lines 10–11 were triggered the second time, and we proceeded to Line 8 with $k = 3$. At this time, we found $\mathbf{N}_a(4) > (\mathbf{N}_a(1) + \mathbf{N}_a(2) + \mathbf{N}_a(3) + 4)/3$, which means involving $\mathbf{N}_a(4)$ did not help with reducing $var(\mathbf{N})$. Therefore, now, we should execute Lines 13–23 to distribute $\Delta n = 4$ to $\mathbf{N}_a(1)$, $\mathbf{N}_a(2)$, and $\mathbf{N}_a(3)$ as evenly as possible. As the summary of $\mathbf{N}_a(1)$, $\mathbf{N}_a(2)$, and $\mathbf{N}_a(3)$ is fixed as $\sum_{j=1}^3 \mathbf{N}_a(j) + \Delta n = 7$. In order to distribute seven to the three elements uniformly, Lines 13–15 firstly assign $floor(Avg) = floor(7/3) = 2$ to all three elements, and $Residual = 1$ is left for assignment. Lines 16–19 assign $Residual = 1$ to $Residual = 1$ elements, one for each element, and $\mathbf{N}_a(1) = 2 + 1 = 3$. Finally, $\mathbf{N} = [4, 8, 3, 2, 2]$ is obtained. The procedure of adjustment in the case of energy shortage is similar, which is omitted.

5. Summary and Discussion

In this section, we will firstly summarize the working procedure of UHETS and then discuss the scalability and complexity.

5.1. Working Procedure Summary

The overall working procedure of UHETS is summarized in Algorithm 2. Firstly, D_{KL} and ϵ are pre-set according to the application service, and σ as the variance of $S_h(t)$, $t \in [1, T]$ is obtained based on historical data. At the beginning of every decision cycle, as shown in Lines 2–3, we firstly updated $g(S_h(t))$ for this decision cycle by using a prediction model (e.g., EWMA [2]). As we chose $T = 24$ for solar energy, in this case, we firstly updated $g(S_h(t))$, $t \in [1, T]$ of the incoming day using an existing prediction model. Consequently, Lines 4–5 transform the original constraints in the optimization problem (6) to linear constraints. Using the linear constraints, Line 6 solves (6) and obtains the transmission scheduling $\mathbf{N} = [n(1), \dots, n(T)]$. Upon the transmission scheduling, at each slot, the transmission scheduling is adjusted depending on the observed harvested energy by using UHETS-adaptation, as shown in Lines 7–9. Finally, the transmission scheduling is carried out, and matrix completion is used to recover the original samples.

Algorithm 2 Working procedure of UHETS

Input: D_{KL} and ϵ required by the application, the variance of $S_h(t)$, $t \in [1, 24]$ based on historical data denoted as σ

Output: $\mathbf{N} = [n(1), \dots, n(T)]$

```

1: for each decision cycle (every day) do
2:   Calculate the harvested energy prediction of  $S_h(t)$ ,  $t \in [1, 24]$  as  $\hat{S}_h(t)$  using a prediction model
3:   Update the Gaussian distribution of  $S_h(t)$ ,  $t \in [1, 24]$  as  $g(S_h(t)) = N(\hat{S}_h(t), \sigma)$ 
4:   Calculate  $S_h^u(t)$ ,  $t \in [1, 24]$  using Algorithm 3
5:   Replace  $S_h(t)$  with  $S_h^u(t)$  in the original constraints in (6) so that the constraints become linear
6:   Solve the optimization problem (6) and obtain the transmission scheduling  $\mathbf{N}$ 
7:   for  $slot \leftarrow 1 : T$  do
8:     Perform UHETS-adaptation to adjust  $\mathbf{N}$  for each slot
9:   end for
10: end for

```

5.2. Scalability

The proposed UHETS was designed for a single-hop star-topology network in which each sensor node independently decides its sensing and transmission scheduling. Although we describe UHETS only using one sensor node and one application server, UHETS is suitable for a multi-sensor network inherently. As UHETS does not involve multihop routing or cooperation among multiple sensor nodes, it inherently accommodates mobility [42]. For the same reason, UHETS is also suitable for clustered networks [43]. However, for a clustered network, cooperation among cluster members can be applied to further improve the data quality subject to the energy constraints.

In this paper, we do not consider the transmission failure caused by interference or channel failure. Retransmissions upon transmission failure apparently influence the transmission scheduling. However, matrix completion adopted in UHETS will alleviate the impact of transmission failure since it only requires the reception of sufficient data packets. As a result, retransmissions can be depressed if sufficient data packets are transmitted successfully. The detailed transmission implementation considering interference and retransmission scheme will be explored in future work.

5.3. Complexity Analysis

The complexity of UHETS contains three main components, the complexity of transforming the original constraints to the linear constraints, i.e., the complexity of Algorithm 3, the complexity to solve the convex Quadratic Programming (QP) optimization problem with linear constraints, and the complexity of UHEST-adaption in Algorithm 1.

Algorithm 3 has two loops, the outer loop being a binary search and the inner loop Newton's method. The former has a complexity of $O(\log_2(\gamma/\rho))$, while it would make more sense to discuss the convergence regarding Newton's method. In Figure 4, we randomly choose one decision cycle in January 2018 and plot the convergence time of Newton's method in this decision cycle. As seen in the figure, Newton's method can converge in less than 10 iterations. In fact, Newton's method can always converge in about 10 iterations in every decision cycle.

When solving the convex QP problem, considering that integer optimization requires relatively high computational overhead and the transmission scheduling in this paper is for low-cost embedded systems, we removed the integer constraint of \mathbf{N} when implementing the presented transmission scheduling and rounded down the resulting \mathbf{N} to obtain the final scheduling scheme. The best-known running time if the interior point method is used for convex QP is $O(m^{1/2}L)$ iterations, where L is the length of input data in Turing machine theory and $m = 3T + 1$ ($T = 24$) is the number of constraints. However, the practical convergence is more optimistic. The average iteration number was about 30. An example of the convergence in one randomly-chosen decision cycle is shown in Figure 5.

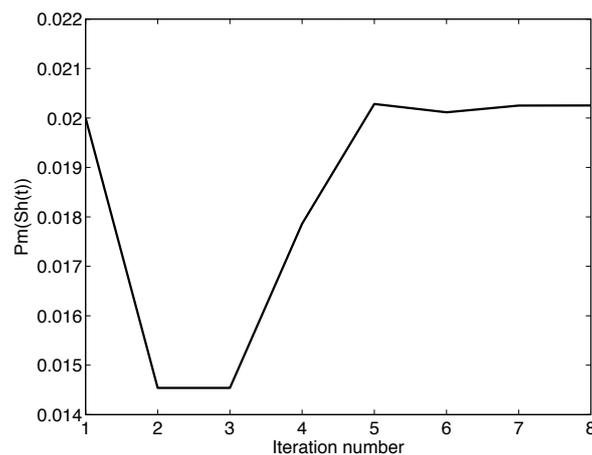


Figure 4. The convergence of Newton's method.

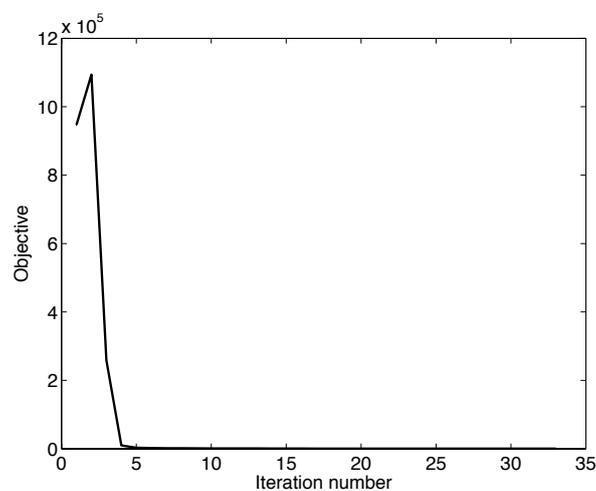


Figure 5. The convergence of the QP method.

In Algorithm 1, the sorting process in Lines 1–4 is dominant regarding complexity. Since Algorithm 1 is performed every slot, the overall complexity is $O(T^3)$, which is trivial for one decision cycle (one day).

In general, as UHETS has a limited scale, the computational complexity is acceptable for low-cost IoT sensors. In order to present the running time of UHETS, we also implemented UHETS using CVXGEN [44], an open-source optimization solver, on a low-cost MCU, STM32L4. For each decision cycle ($T = 24$), UHETS took roughly 900 ms at a current of 8 mA and voltage of 3.6 V. Considering that the optimization is solved only once everyday, this cost is acceptable for this ultra-low-cost MCU.

6. Evaluation

In the evaluation, we supposed that in an environment sensing IoT application, each sensor senses the solar radiation and transmits the readings via the NB-IoT communication interface. We used the data of solar radiation spanning from 1 January 2001 to 31 December 2018 as the historical data for obtaining empirical parameters and the data from 1 January 2018–31 March 2018 for performance evaluation. The solar radiation data are published in units of W/m^2 by NREL (National Renewable Energy Laboratory) [45]. The solar panel size was assumed as $10\text{ cm} \times 10\text{ cm}$, and the charging efficiency was 0.7 [4]. The harvested energy was the product of solar irradiation, solar panel size, time, and charging efficiency. For each slot, 60 packets (i.e., one packet per minute), each having eight bytes, were generated, and selected packets were transmitted via a Quectel BC95 NB-IoT module [46]. The NB-IoT module works at a typical voltage of 3.6 V, current of 230 mA in the transmission state and a data rate of 4 kbps. Thus, each transmission cost about 13 mJ. The battery capacity was $7.7 \times 10^3\text{ J}$ (1800 mAh).

The performance metrics included the objective function and failure rate. The objective function is $sum(\mathbf{N}) - var(\mathbf{N})$, and we refer to this value as utility for simplification hereafter. The failure rate is defined as the ratio between the decisions violating ENO and all decisions, which is evaluated to demonstrate the robustness of UHETS against the harvested energy randomness.

6.1. Prediction Model

Since the Gaussian distribution is built based on a prediction model of harvested energy, this prediction model has a big impact on the performance. In this subsection, we explore two prediction models for evaluation. One is EWMA and the other is a Weather Forecast (WF)-based prediction model [47].

EWMA predicts the amount of harvested energy during slot t on day d as:

$$\hat{E}_h^{d+1}(t) = w\hat{E}_h^d(t) + (1-w)E_h^d(t), \quad (28)$$

where $E_h^d(t)$ denotes the amount of energy harvested during slot t on day d , $\hat{E}_h^d(t)$ is the estimated amount, and w is a weighting factor. We set $w = 0.5$ since it can yield the least prediction error [2].

WF assumes the harvested solar energy for every slot can be formulated as a quadratic function and further influenced by the cloud coverage. Specifically, $\hat{E}_h(t)$ can be estimated as:

$$\hat{E}_h(t) = (a(t-b)^2 + c) \times (1 - sky_condition), \quad (29)$$

where a, b, c are empirical parameters obtained from historical data and $sky_condition$ is the percentage of the sky that is covered by cloud, which was obtained from the weather forecast station. The dataset of $sky_condition$ was obtained from the National Weather Station (NWS) [48]. For every 3 h, NWS publishes a forecast for 3 h, 24 h, and 72 h. In our experiment, we utilized the forecast for 24 h as the decision period in UHETS is 24 h. Based on the meteorological data, a, b, c were obtained by minimizing the mean squared error, as shown in Table 2. Figure 6 shows the difference between the fitting curve by Equation (29) and the ground truth on a sunny day without cloud cover in January 2018.

Table 2. Parameters in WF.

	a	b	c
January	−25.88	12.61	592.21
February	−21.87	12.50	738.19
March	−23.89	12.70	818.22

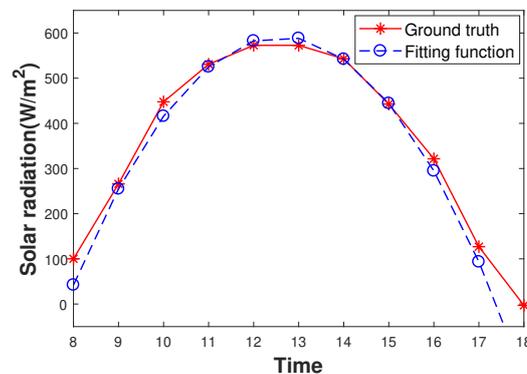
**Figure 6.** Fitting curve of the Weather Forecast (WF).

Table 3 shows the comparison result between EWMA and WF in which the Mean Squared Error (MSE) represents the expected value of the square of the difference between the predicted values by EWMA and WF and the ground truth, and Root Mean Squared (RMS) is the the arithmetic square root of MSE. From Table 3, we can see that WF was more accurate than EWMA, the MSE of WF being roughly 1.09×10^{-3} , while that of EWMA was 1.44×10^{-3} .

Table 3. Comparison of the two prediction models.

	MSE	RSE
EWMA	1.44×10^{-3}	3.80×10^{-2}
WF	1.09×10^{-3}	3.30×10^{-2}

6.2. The Impacts of Parameters

As mentioned in Section 3, the parameters ϵ , D_{KL} indicate the robustness level against the harvested energy randomness. Thus, in Figure 7 we explored their impacts on the performance of UHETS based on the two prediction models described in Section 6.1.

Intuitively, the higher ϵ , the higher the tolerance on ENO violation, thus a higher failure rate and utility. Higher D_{KL} indicates that we assumed the real-world PDF was more different from the Gaussian PDF and that UHETS will yield a stricter worst case bound. In this way, the failure rate and the utility will decrease with increasing D_{KL} . The experimental results were two-fold. Firstly, UHETS enabled flexible transmission scheduling, which traded between energy maintenance and utility. Secondly, by setting small D_{KL} and ϵ , UHETS was robust against the harvested energy randomness. This conclusion holds for both UHETS based on EWMA and WF.

When we compared the performance between the two prediction models, we observed that with a similar utility, the failure rate achieved by WF was slightly lower than EWMA. This conforms to the expectation that WF is superior to EWMA in terms of prediction accuracy. Furthermore, we observed that in order to achieve the same utility, WF requires smaller D_{KL} and higher ϵ . This is also because of the higher prediction accuracy of WF.

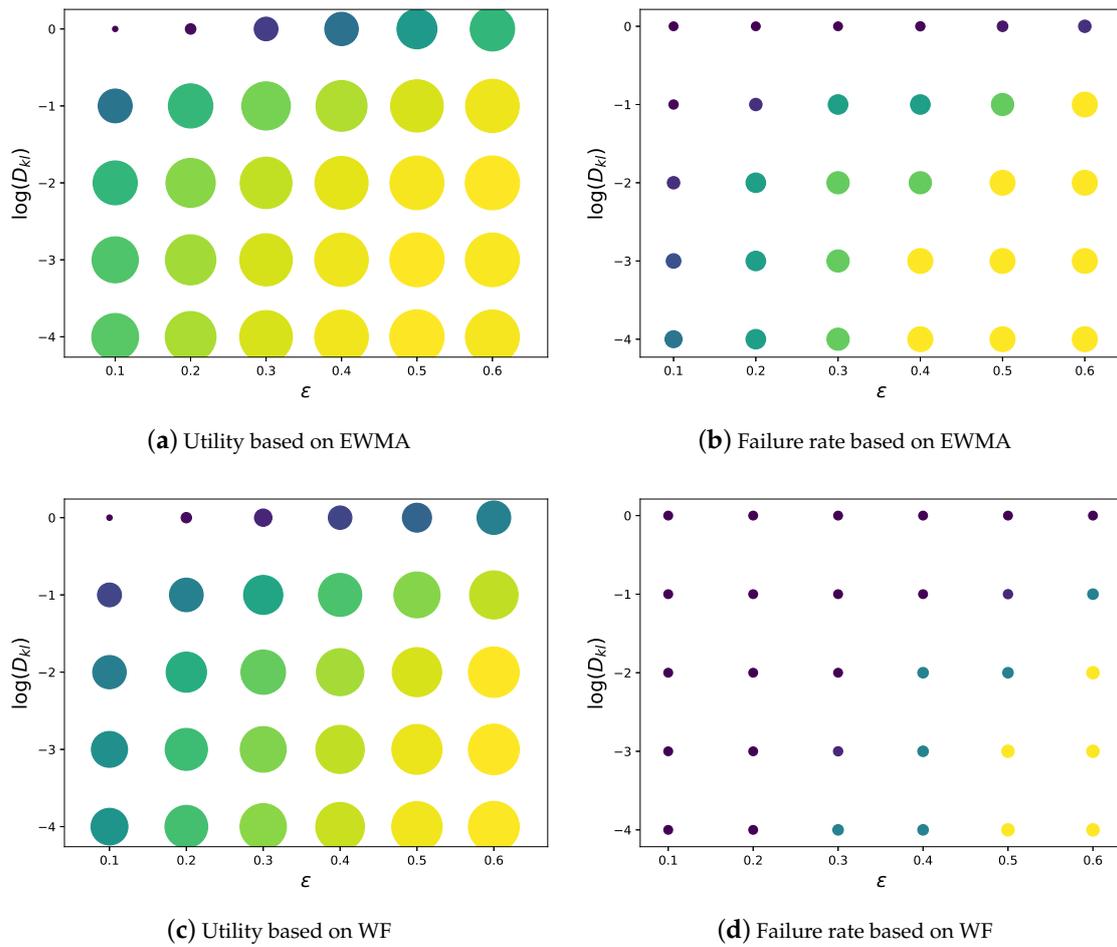


Figure 7. Impact of parameters ϵ and D_{KL} . We present the amount of utility in (a,c) and the failure rate in (b,d) using the size of each circle, respectively.

A more reasonable way is to set D_{KL} empirically based on historical data. The empirical values of D_{KL} were calculated according to the definition of KL divergence $D_{KL} = \mathbb{E}_f[\ln f(S_h(t)) - \ln g(S_h(t))]$ where $f(S_h(t))$ was obtained from historical data and $g(S_h(t))$ was obtained as mentioned in Section 3.3.2. Among all the values of D_{KL} for each day, we chose the maximum value as the final divergence limit. Figure 8 plots the empirical values of D_{KL} by EWMA and WF. In the later experiments, we chose to set different D_{KL} for each slot.

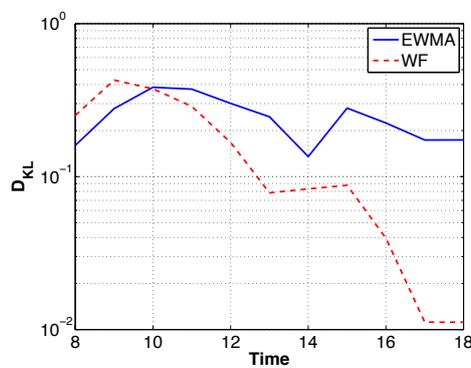


Figure 8. Empirical D_{KL} .

6.3. Performance

In this subsection, we compare the proposed UHETS with three approaches. The first approach is OPTworking as the benchmark. In OPT, the optimization problem is solved directly based on the ground truth data. Thus, it yields the optimal transmission scheduling that achieves the most transmissions with a zero failure rate. The second one solves the optimization problem based on EWMA. With a slight abuse of notation, we refer to this approach as EWMA hereafter. The third one is a Lower Bound-based Transmission Scheduling approach (LBTS). In LBTS, we adopt the energy model presented in [2], the harvested energy is characterized by a $(\rho, \sigma_1, \sigma_2)$ function, where ρ is the average rate of the harvested energy over a long duration, and the burstiness is bounded by σ_1 and σ_2 . Specifically, the harvested energy until slot t is lower bounded by $\rho t - \sigma_2$. LBTS uses this lower bound of harvested energy in the energy constraints.

We evaluated the approaches in terms of failure rate, utility, and recovered quality in the case of tight initial battery energy (i.e., $E_0 = 2$ J in this paper) and lenient initial battery energy (i.e., $E_0 = 2 \times 10^3$ J). Please recall that the failure rate is defined as the ratio between the decisions violating ENO and all decisions and utility is the objective function. Regarding utility, to straightforwardly illustrate the comparison results of different approaches, we show the ratio between the utility of different approaches and that of OPT in Figure 9. The recovered data quality was evaluated by the reconstruction error by using matrix completion, which is defined as:

$$e = \|\mathbf{M} - \mathbf{M}'\|_2 / \|\mathbf{M}\|_2,$$

where \mathbf{M} denotes the original data matrix and \mathbf{M}' is the reconstructed data matrix by using matrix completion. Since advanced matrix completion design is out of the scope of this paper, we used a classic matrix completion method SVT [38] for this purpose.

SVT recovers the original data matrix \mathbf{M} by solving the following optimization problem:

$$\min \quad \tau \|\mathbf{X}\|_* + \frac{1}{2} \|\mathbf{X}\|_F^2 \quad (30)$$

$$s.t. \quad \mathbf{\Omega} \bullet \mathbf{X} = \mathbf{\Omega} \bullet \mathbf{M} \quad (31)$$

where \mathbf{X} is the recovered \mathbf{M}' , $\tau > 0$, $\|\cdot\|_*$ denotes the nuclear norm, and $\|\cdot\|_F$ is the Frobenius norm. Fix $\tau > 0$ and δ of the scalar step size, starting from $\mathbf{Y}_0 = 0 \in \mathbb{R}^{n \times T}$, at the k th iteration, SVT induces:

$$\mathbf{X}_k = \mathcal{D}_\tau(\mathbf{Y}_{k-1}) \quad (32)$$

$$\mathbf{Y}_k = \mathbf{Y}_{k-1} + \delta \mathbf{\Omega} \bullet (\mathbf{M} - \mathbf{X}_k) \quad (33)$$

until the stopping criterion is reached, where $\mathcal{D}_\tau(\cdot)$ is the soft-thresholding operator on the singular values of matrix \cdot . When implementing SVT, we used the source code from [49] with parameters $\tau = 10\sqrt{nT}$, $\delta = \|\mathbf{N}\|_1 / (10nT)$, maximum number of iterations 10^3 , and stopping criterion $\epsilon = 10^{-4}$.

We firstly focus on the performance UHETS based on EWMA, as shown in Figure 9a,c,e.

In the case of tight initial battery energy, all approaches were expected to yield lower utility, consequently higher data recovery error and a lower failure rate, as well. This is because lower E_0 has a stronger impact on Constraint (2), while higher E_0 will impose a weak effect on (2). Our experimental results confirmed this expectation. As shown in Figure 9a, UHETS obtained a growing utility with increasing ϵ and approached that of EWMA with a high ϵ . Consequently, the data error of UHETS decreased and approached that of EWMA with increasing ϵ . We also observed that when ϵ exceeded 0.5, there was a turning point regarding both the utility and data error. This is because $n(t)$ for $t \in [0, 8]$ was mainly determined by the tight initial energy. As ϵ increased, $n(t)$ for the latter slots rose consequently, which will lead to the augmentation of $var(N)$. As a result, the utility decreased slightly, as did the recovery data error. With slightly lower utility and data error as compared with EWMA, the failure rate of UHETS always stayed below the 28% achieved by EWMA.

Among all, LBTS obtained a zero failure rate, but the lowest utility and data quality. This lower bound-based performance guarantee method imposes such an overly tight constraint that the utility degrades significantly, which makes LBTS impractical to use in reality. We can conclude that UHETS was able to achieve flexible transmission scheduling and energy maintenance.

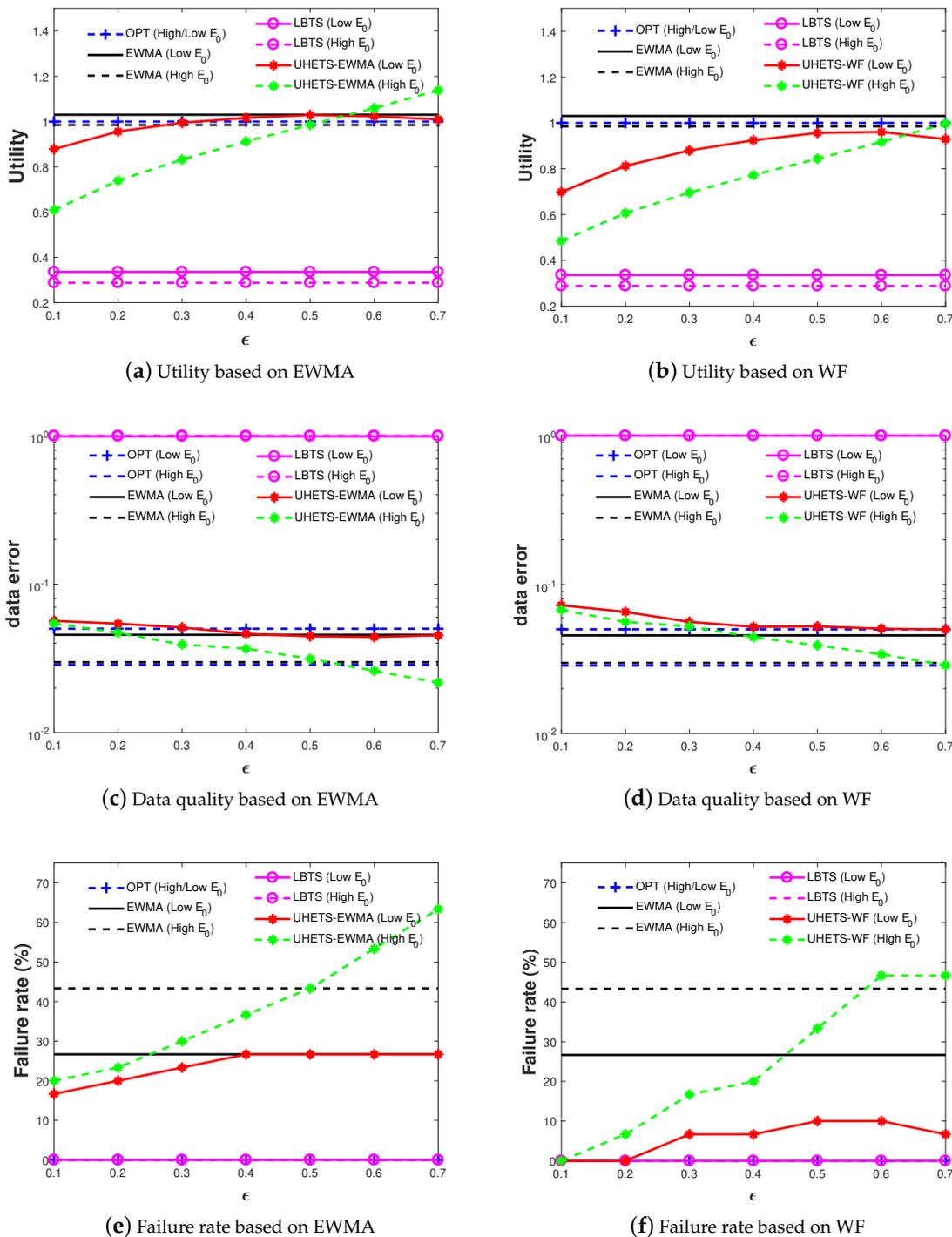


Figure 9. Statistic performance: comparison of Uncertain Harvested Energy-based Transmission Scheduling (UHETS) based on EWMA or WF with pure EWMA, Lower Bound-based Transmission Scheduling (LBTS), and OPT.

In the case of lenient initial battery energy, the utility would increase, data error decrease, and failure rate grow consequently as well for both UHETS and pure EWMA, while LBTS would remain unchanged. Specifically, the failure rate of EWMA was boosted to roughly 43%, while UHETS had varying failure rates from 18–43% under $\epsilon \leq 0.5$.

UHETS based on WF as shown in Figure 9b,d,f showed similar experimental results except that the utility and data rate were inferior and the failure rate was superior to UHETS based on EWMA under the same ϵ .

In summary, by comparison of the experimental results in Figure 9, it was demonstrated that UHETS was capable of providing flexible service performance by tuning parameter ϵ .

6.4. UHETS-Adaptation

This subsection compares the performance of UHETS-adaptation based on EWMA (as shown in Figure 10 to OPT, EWMA, and UHETS. We only plot the performance in the case of tight initial battery energy based on EWMA since the performance for other cases showed similar results. As shown in Figure 10a, by adaptively adjusting the transmission scheduling, the utility of UHETS-adaptation was improved greatly compared with UHETS. With increasing ϵ , the data error approached OPT and EWMA. Figure 10c clearly shows that the failure rate of UHETS-adaptation was lower than pure UHETS. This figure demonstrates that UHETS-adaptation was capable of both improving the utility and reducing the failure rate with lightweight extra computational overhead.

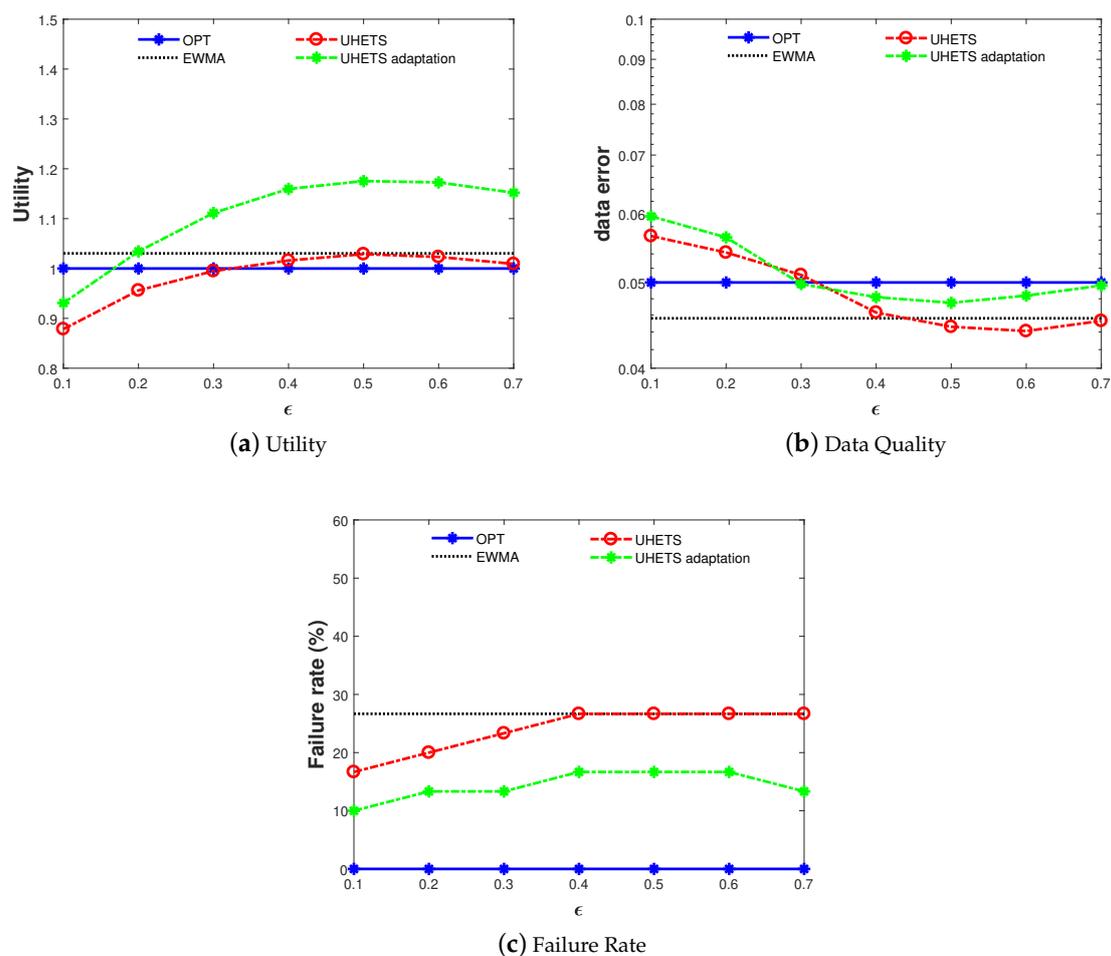


Figure 10. Performance of UHETS-adaptation.

7. Conclusions

In this paper, we focused on the environment sensing IoT applications and aimed to address the utility maximization problem considering the randomness of the harvested energy. To handle the randomness gracefully, we firstly formulated the utility maximization problem subject to energy constraints with the harvested energy as a random variable. Profiling the harvested energy as a random process biased by a known distribution, we replaced the random variable with a constant so that the utility maximization problem could be solved using a traditional method. We also proposed an adaptive transmission scheduling approach that could adjust the transmission scheduling based on real-time harvested energy. Numerical experiment results demonstrated that UHETS was able to provide flexible transmission scheduling, which can achieve a good trade-off between ENO guarantee and utility maximization.

In this paper, data quality was implicitly optimized in the design of the transmission scheduling approach. In future work, we plan to directly use data quality as an explicit optimization objective, and the joint design of matrix completion and transmission scheduling has potential to further improve the data quality. Furthermore, as discussed in Section 5.2, the transmission failure caused by interference or channel failure is not considered in UHETS. We plan to design cross-layer transmission scheduling considering an interference and retransmission scheme in the future.

Moreover, we only focused on solar energy as the energy source. However, there is a rich body of energy sources such as vibration energy, wind energy, and so on, that are difficult to model with a known distribution. The application of UHETS with these energy sources still needs to be explored in the future.

Author Contributions: Methodology, J.H. and R.W.; resources, Y.Z.; software, J.C.; writing, original draft, J.H. and B.Z.; writing, review and editing, B.Z.

Funding: This research was funded by the Natural Science Foundation of Jiangsu Province (BK20160807), the National Natural Science Foundation of China (61602242), the China Postdoctoral Science Foundation (2017M611805), the Jiangsu Planned Projects for Postdoctoral Research Funds (1701138B), the China Postdoctoral Science Foundation No. 11 Special Fund (2018T110498), the Natural Science Foundation of Jiangsu Province (BK20160812), the China Postdoctoral Science Foundation (2017M611806), the Jiangsu Planned Projects for Postdoctoral Research Funds (1701137A), the Aeronautical Science Foundation of China (2016ZC52030), and the National Natural Science Foundation of China (61572253).

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

We now prove that problems (19)–(21) is a convex optimization problem.

Proof. We let y be $S_h(t)$ and y' be $S_h^u(t)$ and rewrite (19)–(21) as follows:

$$\max_{f(y)} \int_0^{+\infty} w(y, y') \cdot f(y) dy \quad (\text{A1})$$

$$\text{s.t.} \quad \int_0^{+\infty} [\ln f(y) - \ln g(y)] f(y) dy \leq D_{KL} \quad (\text{A2})$$

$$\int_0^{+\infty} f(y) dy = 1. \quad (\text{A3})$$

The objective function (A1) and Constraint (A3) are affined with respect to $f(y)$. Next, we show that Constraint (A2) is convex.

We define a convex function $f(y) = -\ln y$. As we know, f and its perspective function g should be convex or concave at the same time (The readers can refer to [40,50] for a more detailed proof.), and its perspective function is the relative entropy of y and z , i.e.,

$$g(y, z) = -z \ln(y/z) = z(\ln z - \ln y), \quad (\text{A4})$$

which is convex as well. Then, we have that the KL divergence $\int_{y \in S} [\ln f(y) - \ln g(y)] f(y) dx$ between distribution $f(y)$ and $g(y)$ is convex in $f(y)$ and also $g(y)$. As a result, the constraint (A2) is convex with respect to $f(y)$.

Finally, we claim that the optimization problem is convex. \square

Appendix B

We let:

$$\mathcal{P}(S_h(t), f, \alpha, \beta) = \mathbb{E}_f \left[w(S_h(t), S_h^u(t)) - \beta - \alpha \ln \frac{f(S_h(t))}{g(S_h(t))} \right]. \quad (\text{A5})$$

Then, the derivative of $\mathcal{P}(S_h(t), f, \alpha, \beta)$ with respect to f can be derived as:

$$\begin{aligned} \frac{\partial \mathcal{P}}{\partial f} &= \lim_{t \rightarrow 0} \frac{1}{t} \left[\mathcal{P}(f(S_h(t)) + t \cdot g(S_h(t))) - \mathcal{P}(f(S_h(t))) \right] \\ &= \int_0^{+\infty} \left(w(S_h(t), S_h^u(t)) - \alpha \ln \frac{f(S_h(t))}{g(S_h(t))} - \beta - \alpha \right) g(S_h(t)) dS_h(t). \end{aligned}$$

Applying the Karush–Kuhn–Tucker condition, we have:

$$w(S_h(t), S_h^u(t)) - \alpha \ln \frac{f(S_h(t))}{g(S_h(t))} - \beta - \alpha = 0 \quad (\text{A6})$$

$$\int_0^{+\infty} f(S_h(t)) dS_h(t) = 1 \quad (\text{A7})$$

$$\mathbb{E} \left[\ln \frac{f(S_h(t))}{g(S_h(t))} \right] - D_{KL} \leq 0 \quad (\text{A8})$$

$$\alpha \left(D_{KL} - \mathbb{E} \left[\ln \frac{f(S_h(t))}{g(S_h(t))} \right] \right) = 0 \quad (\text{A9})$$

From (A6), the optimal distribution function can be expressed as follows:

$$f^*(S_h(t)) = g(S_h(t)) \exp \left(\frac{w(S_h(t), S_h^u(t)) - \beta}{\alpha} - 1 \right). \quad (\text{A10})$$

The Lagrangian multipliers α, β in (A10) should be chosen properly such that Conditions (A7)–(A9) are satisfied. By substituting (A10) to Expressions (A7)–(A9), we can straightforwardly obtain (α, β) by solving the following:

$$U(S_h^u(t)) e^{\beta/\alpha} + V(S_h^u(t)) e^{(1-\beta)/\alpha} = 0, \quad (\text{A11})$$

$$V(S_h^u(t)) e^{(1-\beta)/\alpha} - \beta - \alpha(1 + D_{KL}) = 0, \quad (\text{A12})$$

where $U(S_h^u(t)) = G(S_h^u(t)) e^{-1}$, $V(S_h^u(t)) = (1 - G(S_h^u(t))) e^{-1}$, and $G(*)$ is the cumulative distribution of $g(S_h(t))$.

Appendix C

We use binary search and Newton's method to seek a proper solution of $S_h^u(t), \alpha^*, \beta^*$ as shown in Algorithm A1. The search range of $S_h^u(t)$ is $[0, \gamma]$, and iteration tolerance is ρ . Initially, we started the search with lower bound $l = 0$ and upper bound $u = \gamma$ (u must be chosen large enough such that $P_m^u(u) > \epsilon$). We need to check if $P_m^u(m) > \epsilon$ or $P_m^u(m) < \epsilon$ where $m = (l + u)/2$ and halve the search radius as shown in Line 23 and Line 25, respectively. During each search, we use Newton's method as seen in Lines 10–17 to calculate $P_m^u(m)$ such that Equations (A11) and (A12) are satisfied. The binary search terminates when the upper bound u converges to the lower bound l .

Algorithm A1 The iterative method to calculate $S_h^u(t), \alpha^*, \beta^*$

Input: $g(S_h(t))$, search range $[0, \gamma]$, iteration tolerance ρ, ϵ .

Output: $S_h^u(t)$.

```

1:  $l \leftarrow 0, u \leftarrow \gamma$  // initialize the search range as  $[l, u] = [0, \gamma]$ 
2:  $H_1(S_h^u(t), \alpha, \beta) \leftarrow U(S_h^u(t))e^{\beta/\alpha} + V(S_h^u(t))e^{(1-\beta)/\alpha}$ ,
3:  $H_2(S_h^u(t), \alpha, \beta) \leftarrow V(S_h^u(t))e^{(1-\beta)/\alpha} - \beta - \alpha(1 + D_{KL})$ ,
4:  $\mathbf{H}(S_h^u(t), \alpha, \beta) \leftarrow [H_1(S_h^u(t), \alpha, \beta), H_2(S_h^u(t), \alpha, \beta)]^T$ 
5: while  $|u - l| > \rho$  do
6:   //Until the search range converges
7:    $k \leftarrow 1$ 
8:    $m \leftarrow (l + u)/2$  //  $m$  is the middle of the search range
9:    $\alpha_1 = 2, \beta_1 = -2$  //choose a random start for  $\alpha, \beta$ 
10:  while  $\mathbf{H}(m, \alpha_k, \beta_k) > \rho$  do
11:    //Newton iteration to calculate  $\alpha^*$  and  $\beta^*$  such that  $\mathbf{H}(m, \alpha^*, \beta^*) \approx 0$ 
12:    evaluate  $\mathbf{H}(m, \alpha, \beta)$  and Jacobian matrix  $J(m, \alpha, \beta)$ 
13:    solve  $J(m, \alpha, \beta)[\Delta\alpha_k, \Delta\beta_k]^T = -\mathbf{H}(m, \alpha, \beta)$ 
14:     $\alpha_{k+1} \leftarrow [\alpha_k + \Delta\alpha_k]^+$  //update  $\alpha$ 
15:     $\beta_{k+1} \leftarrow \beta_k + \Delta\beta_k$  //update  $\beta$ 
16:     $k \leftarrow k + 1$ 
17:  end while
18:  //now  $\mathbf{H}(m, \alpha_{k+1}, \beta_{k+1}) \leq \rho$ 
19:   $\alpha^* \leftarrow \alpha_{k+1}$ 
20:   $\beta^* \leftarrow \beta_{k+1}$ 
21:   $P_m^u(m) \leftarrow \alpha^* D_{KL} + \beta^*$  //update  $P_m^u(m)$  using  $\alpha^*$  and  $\beta^*$ 
22:  if  $P_m^u(m) < \epsilon$  then //if  $P_m^u(m) < \epsilon$ 
23:     $l \leftarrow m$  //halve the search range to  $[m, u]$ 
24:  else
25:     $u \leftarrow m$  //halve the search range to  $[l, m]$ 
26:  end if
27:  if  $|P_m^u(m) - \epsilon| < \rho$  then // If  $P_m^u(m) \approx \epsilon$ , the binary search terminates.
28:    break
29:  end if
30: end while
31:  $S_h^u(t) \leftarrow m$  //  $S_h^u(t)$  such that  $P_m^u(S_h^u(t)) = \epsilon$  is obtained.

```

References

1. Kaur, N.; Sood, S.K. An energy-efficient architecture for the Internet of Things (IoT). *IEEE Syst. J.* **2017**, *11*, 796–805. [[CrossRef](#)]
2. Kansal, A.; Hsu, J.; Zahedi, S.; Srivastava, M.B. Power management in energy harvesting sensor networks. *ACM Trans. Embed. Comput. Syst.* **2007**, *6*, 32. [[CrossRef](#)]
3. Hao, J.; Wang, R.; Zhuang, Y.; Zhang, B. A Flexible Network Utility Optimization Approach for Energy Harvesting Sensor Networks. In Proceedings of the IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, UAE, 9–13 December 2018.
4. Kar, K.; Krishnamurthy, A.; Jaggi, N. Dynamic node activation in networks of rechargeable sensors. In Proceedings of the IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Miami, FL, USA, 13–17 March 2006; Volume 14, pp. 15–26.
5. Ozel, O.; Ulukus, S. Information-Theoretic Analysis of an Energy Harvesting Communication System. In Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications Workshops, Istanbul, Turkey, 26–30 September 2010; pp. 330–335.

6. Ho, C.K.; Zhang, R. Optimal Energy Allocation for Wireless Communications with Energy Harvesting Constraints. *IEEE Trans. Signal Process.* **2011**, *60*, 4808–4818. [[CrossRef](#)]
7. Sharma, V.; Mukherji, U.; Joseph, V.; Gupta, S. Optimal Energy Management Policies for Energy Harvesting Sensor Nodes. *IEEE Trans. Wirel. Commun.* **2010**, *9*, 1326–1336. [[CrossRef](#)]
8. Tapparello, C.; Simeone, O.; Rossi, M. Dynamic Compression-Transmission for Energy-Harvesting Multihop Networks with Correlated Sources. *IEEE/ACM Trans. Netw.* **2014**, *22*, 1729–1741. [[CrossRef](#)]
9. Mao, Z.; Koksals, C.E.; Shroff, N.B. Near Optimal Power and Rate Control of Multi-Hop Sensor Networks with Energy Replenishment: Basic Limitations with Finite Energy and Data Storage. *IEEE Trans. Autom. Control* **2012**, *57*, 815–829.
10. Liu, R.S.; Sinha, P.; Koksals, C.E. Joint Energy Management and Resource Allocation in Rechargeable Sensor Networks. In Proceedings of the IEEE Infocom, San Diego, CA, USA, 14–19 March 2010; pp. 902–910.
11. Ciunozzo, D.; Aubry, A.; Carotenuto, V. Rician MIMO channel-and jamming-aware decision fusion. *IEEE Trans. Signal Process.* **2017**, *65*, 3866–3880. [[CrossRef](#)]
12. Ciunozzo, D.; Rossi, P.S.; Dey, S. Massive MIMO channel-aware decision fusion. *IEEE Trans. Signal Process.* **2014**, *63*, 604–619. [[CrossRef](#)]
13. Ciunozzo, D.; Rossi, P.S. Quantizer design for generalized locally optimum detectors in wireless sensor networks. *IEEE Wirel. Commun. Lett.* **2017**, *7*, 162–165. [[CrossRef](#)]
14. Vigorito, C.M.; Ganesan, D.; Barto, A.G. Adaptive control of duty cycling in energy-harvesting wireless sensor networks. In Proceedings of the 2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, San Diego, CA, USA, 18–21 June 2007; pp. 21–30.
15. Yoon, I.; Kim, H.; Dong, K.N. Adaptive Data Aggregation and Compression to Improve Energy Utilization in Solar-Powered Wireless Sensor Networks. *Sensors* **2017**, *17*, 1226. [[CrossRef](#)]
16. Djenouri, D.; Bagaa, M. Energy harvesting aware relay node addition for power-efficient coverage in wireless sensor networks. In Proceedings of the IEEE International Conference on Communications, London, UK, 8–12 June 2015; pp. 86–91.
17. Zhou, X.; Zhang, R.; Ho, C.K. Wireless Information and Power Transfer: Architecture Design and Rate-Energy Tradeoff. *IEEE Trans. Commun.* **2013**, *61*, 4754–4767. [[CrossRef](#)]
18. Ng, D.W.K.; Lo, E.S.; Schober, R. Multiobjective Resource Allocation for Secure Communication in Cognitive Radio Networks with Wireless Information and Power Transfer. *IEEE Trans. Veh. Technol.* **2016**, *65*, 3166–3184. [[CrossRef](#)]
19. Wu, Q.; Tao, M.; Kwan Ng, D.W.; Chen, W.; Schober, R. Energy-Efficient Resource Allocation for Wireless Powered Communication Networks. *IEEE Trans. Wirel. Commun.* **2016**, *15*, 2312–2327. [[CrossRef](#)]
20. Ju, H.; Zhang, R. Optimal resource allocation in full-duplex wireless-powered communication network. *IEEE Trans. Commun.* **2014**, *62*, 3528–3540. [[CrossRef](#)]
21. Vamvakas, P.; Tsiropoulou, E.E.; Vomvas, M.; Papavassiliou, S. Adaptive power management in wireless powered communication networks: A user-centric approach. In Proceedings of the 2017 IEEE 38th Sarnoff Symposium, Newark, NJ, USA, 18–20 September 2017; pp. 1–6.
22. Boshkovska, E.; Ng, D.W.K.; Zlatanov, N.; Schober, R. Practical Non-Linear Energy Harvesting Model and Resource Allocation for SWIPT Systems. *IEEE Commun. Lett.* **2015**, *19*, 2082–2085. [[CrossRef](#)]
23. Tran, V.; Georges, K.; Trung, T.K. Resource Allocation in SWIPT Networks under a Non-Linear Energy Harvesting Model: Power Efficiency, User Fairness, and Channel Non-Reciprocity. *IEEE Trans. Veh. Technol.* **2018**, *67*, 8466–8480. [[CrossRef](#)]
24. Boshkovska, E.; Ng, D.W.K.; Zlatanov, N.; Koelpin, A.; Schober, R. Robust Resource Allocation for MIMO Wireless Powered Communication Networks Based on a Non-Linear EH Model. *IEEE Trans. Commun.* **2017**, *65*, 1984–1999. [[CrossRef](#)]
25. Jiang, R.; Xiong, K.; Fan, P.; Zhang, Y.; Zhong, Z. Optimal Design of SWIPT Systems with Multiple Heterogeneous Users under Non-linear Energy Harvesting Model. *IEEE Access* **2017**, *5*, 11479–11489. [[CrossRef](#)]
26. Xiong, K.; Wang, B.; Liu, K.J.R. Rate-Energy Region of SWIPT for MIMO Broadcasting Under Nonlinear Energy Harvesting Model. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 5147–5161. [[CrossRef](#)]
27. Ponnimbaduge Perera, T.D.; Jayakody, D.N.K.; Sharma, S.K.; Chatzinotas, S.; Li, J. Simultaneous Wireless Information and Power Transfer (SWIPT): Recent Advances and Future Challenges. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 264–302. [[CrossRef](#)]

28. Gündüz, D.; Devillers, B. Two-hop communication with energy harvesting. In Proceedings of the 2011 4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), San Juan, Puerto Rico, 13–16 December 2011; pp. 201–204.
29. Fan, L.; Chen, S.; Tang, S.; Xiao, H.; Yu, W. Efficient Topology Design in Time-Evolving and Energy-Harvesting Wireless Sensor Networks. In Proceedings of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems, Hangzhou, China, 14–16 October 2013.
30. Maemoto, D.; Mori, K.; Sanada, K. Adaptive clustering control for energy-harvesting WSNs with non-uniform energy harvesting rate. In Proceedings of the 2017 IEEE Sensors, Glasgow, UK, 29 October–1 November 2017; pp. 1–3.
31. Tunc, C.; Akar, N. Markov fluid queue model of an energy harvesting IoT device with adaptive sensing. *Perform. Eval.* **2017**, *111*, 1–16. [[CrossRef](#)]
32. Guntupalli, L.; Li, F.Y.; Gidlund, M. Energy Harvesting Powered Packet Transmissions in Duty-Cycled WSNs: A DTMC Analysis. In Proceedings of the IEEE Global Communications Conference, Singapore, 4–8 December 2018.
33. Gorlatova, M.; Wallwater, A.; Zussman, G. Networking low-power energy harvesting devices: Measurements and algorithms. *IEEE Trans. Mob. Comput.* **2013**, *12*, 1853–1865. [[CrossRef](#)]
34. Tarighati, A.; Gross, J.; Jaldén, J. Decentralized hypothesis testing in energy harvesting wireless sensor networks. *IEEE Trans. Signal Process.* **2017**, *65*, 4862–4873. [[CrossRef](#)]
35. Huang, C.; Zhang, R.; Cui, S. Throughput Maximization for the Gaussian Relay Channel with Energy Harvesting Constraints. *IEEE J. Sel. Areas Commun.* **2013**, *31*, 1469–1479. [[CrossRef](#)]
36. Yadav, A.; Goonewardena, M.; Ajib, W.; Dobre, O.A.; Elbiaze, H. Energy Management for Energy Harvesting Wireless Sensors with Adaptive Retransmission. *IEEE Trans. Commun.* **2017**, *65*, 5487–5498. [[CrossRef](#)]
37. Chamanian, S.; Baghaee, S.; Uluşan, H.; Zorlu, Ö.; Uysal-Biyikoglu, E.; Külah, H. Implementation of Energy-Neutral Operation on Vibration Energy Harvesting WSN. *IEEE Sens. J.* **2019**, *19*, 3092–3099. [[CrossRef](#)]
38. Cai, J.F.; Candès, S.E.; Shen, Z. A Singular Value Thresholding Algorithm for Matrix Completion. *SIAM J. Optim.* **2008**, *20*, 1956–1982. [[CrossRef](#)]
39. Anastasi, G.; Conti, M.; Francesco, M.D.; Passarella, A. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.* **2009**, *7*, 537–568. [[CrossRef](#)]
40. Wang, R.; Wang, P.; Xiao, G. A robust optimization approach for energy generation scheduling in microgrids. *Energy Convers. Manag.* **2015**, *106*, 597–607. [[CrossRef](#)]
41. Salcedo-Sanz, S.; Casanova-Mateo, C.; Muñoz-Marí, J.; Camps-Valls, G. Prediction of Daily Global Solar Irradiation Using Temporal Gaussian Processes. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 1936–1940. [[CrossRef](#)]
42. Tsiropoulou, E.; Koukas, K.; Papavassiliou, S. A socio-physical and mobility-aware coalition formation mechanism in public safety networks. *EAI Endorsed Trans. Future Internet* **2018**, *4*, 154176. [[CrossRef](#)]
43. Lin, C.R.; Gerla, M. Adaptive clustering for mobile wireless networks. *IEEE J. Sel. Areas Commun.* **1997**, *15*, 1265–1275. [[CrossRef](#)]
44. Mattingley, J.; Boyd, S. CVXGEN: A code generator for embedded convex optimization. *Optim. Eng.* **2012**, *13*, 1–27. [[CrossRef](#)]
45. Nrel: National Renewable Energy Laboratory. Available online: http://www.nrel.gov/midc/nwtc_m2/ (accessed on 25 April 2019).
46. Available online: <https://www.quectel.com/product/bc95.htm> (accessed on 12 February 2019).
47. Sharma, N.; Gummesson, J.; Irwin, D.; Shenoy, P. Cloudy Computing: Leveraging Weather Forecasts in Energy Harvesting Sensor Systems. In Proceedings of the 2010 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), Boston, MA, USA, 21–25 June 2010; pp. 1–9.
48. Nws: National Oceanic and Atmospheric Administration’s National Weather Service. Available online: <https://w2.weather.gov/climate/> (accessed on 26 April 2019).
49. SVT. Available online: <http://svt.stanford.edu/code.html> (accessed on 6 February 2019).
50. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.

