Big Data Transmission in Industrial IoT Systems With Small Capacitor Supplying Energy

Xiaolin Fang, Junzhou Luo, *Member, IEEE*, Guangchun Luo, Weiwei Wu, *Member, IEEE*, Zhipeng Cai, *Member, IEEE*, and Yi Pan, *Member, IEEE*

***Abstract*—Transmission is crucial for big data analysis and learning in industrial Internet of Things (IoT) systems. To transmit data with limited energy is a challenge. This paper studies the problem of data transmission in energy harvesting systems with capacitor to supply energy where the energy receiving rate varies over time. The energy receiving rate is slower when the capacitor receives more energy. Based on this characteristic, we study the problem of how to transmit more data when the energy receiving time is not continuous. Given many packets that arrive at different time instances, there is a tradeoff between transmitting the packet right now or saving the energy to transmit the future arriving packets. We formalize two types of problems. The first one is how to minimize the total completion time when there is enough energy to transmit all the packets. The second one is how to transmit as many packets as possible when the energy is not enough to transmit all the packets. For the first problem, we give a $1+\alpha$ approximation offline algorithm when all the information of the packets and the energy receiving periods is known in advance, and a $\max\{2, \beta\}$ competitive ratio online algorithm where the information is not known in advance. For the second problem, we study three cases and give a $6+\lceil\frac{h}{b/R}\rceil$ approximation offline algorithm for the general situation. We also prove that there does not exit a constant competitive ratio online algorithm.**

***Index Terms*—Capacitor energy, data communication, energy harvesting, wireless energy transfer.**

Manuscript received February 7, 2018; revised April 4, 2018, June 10, 2018, and July 7, 2018; accepted July 27, 2018. Date of publication August 2, 2018; date of current version April 3, 2019. This work was supported in part by the National Key Research and Development Programm of China under Grant 2017YFB1003000, in part by the National Natural Science Foundation of China under Grant 61502099, Grant 61632008, Grant 61320106007, Grant 61572130, Grant 61502100, Grant 61532013, Grant 61602084, Grant 61672154, and Grant 61602111, in part by the National Science Foundation under Grant 1252292, Grant 1741277, Grant 1829674, and Grant 1704287, in part by the Jiangsu Provincial Natural Science Foundation of China under Grant BK20150637, in part by the Jiangsu Provincial Key Laboratory of Network and Information Security under Grant BM2003201, in part by the Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grant 93K-9, in part by the Jiangsu Provincial Key Technology R&D Program under Grant BE2014603, in part by the Collaborative Innovation Center of Novel Software Technology and Industrialization, Special Project on Youth Science and Technology Innovation Research Team of Sichuan Province under Grant 2015TD0002, in part by the Science and Technology Innovation Seed Project of Sichuan Province under Grant 2017RZ0008, in part by the International cooperation and exchange Project of Sichuan Province under Grant 2017HH0075, and in part by the Chinese Scholarship Council under Grant 20173109. Paper no. TII-18-0374. *(Corresponding author: Yi Pan.)*

X. Fang, J. Luo, and W. Wu are with the School of Computer Science and Engineering, Southeast University, Nanjing 211189, China (e-mail: xiaolin@seu.edu.cn; jluo@seu.edu.cn; weiweiwu@seu.edu.cn).

G. Luo is with the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: gcluo@uestc.edu.cn).

Z. Cai and Y. Pan are with the Department of Computer Science, Georgia State University, Atlanta, GA 30303 USA (e-mail: zcai@gsu.edu; yipan@gsu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2018.2862421

I. Introduction

The industrial Internet of Things (IoT) has received much attention since it allows objects to be sensed or controlled remotely. Based on cloud computing, big data analysis, and machine learning technologies, it creates opportunities for more direct integration of the physical world into computer-based systems, and results in improved efficiency, accuracy, and economic benefit in addition to reduced human intervention [1]–[3]. As big data collection is a precondition for such IoT systems, how to collect data from many sensors with extremely limited energy is a great challenge. For example, in a medical care system such as heart monitoring implants [4]–[6], once implanted, the biosensors are expected to work for a long time and it is inconvenient to replace the energy storage. But the limited stored energy cannot guarantee a long-term running time of the devices for data collection and analysis; therefore, we always use the energy harvesting technologies to supply energy and prolong the lifetime of an IoT system [7]–[9].

Recently, passive data collection where capacitor is employed to supply energy for passive sensors is being widely adopted [10]–[12]. Information can be collected by using a capacitor that supplies energy to a sensor, where the capacitor can receive, convert, and store radio frequency (RF) signal to energy [13]–[15]. Many problems of passive sensor data collection that use capacitor to supply energy for the sensors have been studied [16]–[19]. A fundamental problem is the efficient use of energy. Energy is a scarce resource because it must be harvested at low rates from signals transmitted meters away. Furthermore, to remain physically small and to power-up quickly, these sensors have minuscule energy stores compared with sensor network nodes. For example, the energy storage of the WISP [20] is eight orders of magnitude smaller than the battery of the popular Telos sensor mote. In the studies of the energy charging process of capacitor, it is found that the voltage increases slower when

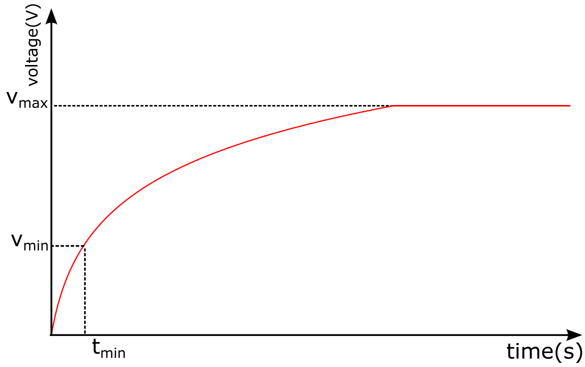


Fig. 1. Energy received over time.

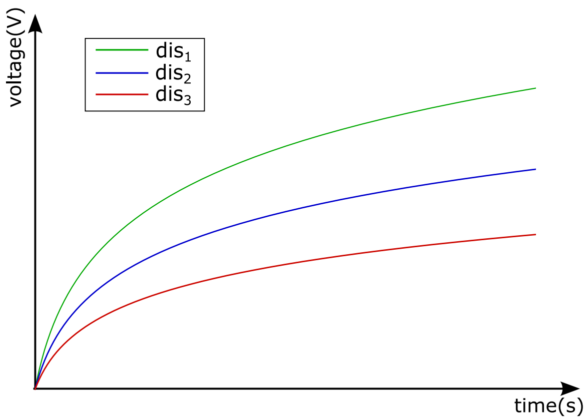


Fig. 2. Charging at different distances.

the capacitor receives more energy, that is, the energy receiving rate becomes slower throughout the charging process.

The characteristics of the usually used capacitor that supplies energy to a passive sensor can be summarized as follows. First, the energy storage is very small. Second, it needs a $v_{\min}$ to power on a sensor. Third, the energy receiving rate varies over time: when the capacitor gets full, the rate becomes slower. As shown in Fig. 1, $V_{\min}$ is the minimum voltage required to power on the circuit so as to transmit data. $V_{\max}$ is the maximum voltage that a capacitor can be charged since the capacitor has a maximum capacitance. The energy received over time looks like a logarithmic curve where the energy receiving rate is getting slower. Passive sensors receive all their energy from harvesting. Harvested energy/power is a function of distance from reader and decreases according to a power law with exponent of 2 or higher. Therefore, the energy harvesting rate varies at different distances as shown in Fig. 2.

Since data transmission is the most crucial factor for energy consumption in low power devices [21]–[24], efficient scheduling algorithms must be employed and designed. We study the energy consumption problem caused by data transmission and design efficient scheduling algorithms in this paper. A passive sensor must receive enough energy and then start to transmit data. A sensor may not receive energy continuously and stably as the RF signal transmitter may move or be out of service. Thus, given some energy receiving periods which we call charging periods in this paper, how to transmit as many packets as possible is a problem needed to be studied in the situation where the length of the charging period and charging distances vary over time. Our contributions are as follows.

1) To the best of our knowledge, this is the first work to investigate data transmission considering the varying characteristic of energy receiving rate with capacitor.
2) We study the minimum total completion time problem where there is enough energy to transmit all the packets. A $1 + \alpha$ approximation algorithm is given for the offline problem, and a $\max\{2, \beta\}$ competitive ratio algorithm is given for the online problem.
3) When there is no enough energy to transmit all the packets, we study how to transmit the maximum number of packets. An $6 + \frac{h}{b/R}$ approximation algorithm is given for the offline problem, and we prove that there does not exist a constant competitive ratio online algorithm.
4) We also study the packet transmission problem where sensors receive energy at different distances.

The rest of the paper is organized as follows. Section II states the background and motivation. Section III presents the definition of our problem. The minimum total completion time is studied in Section IV, where there is enough energy to transmit all the packets. Section V studies how to transmit the maximum number of packets when there is no enough energy and a sensor can receive energy at the same distance. Section VI studies how to transmit the maximum number of packets, and a sensor can receive energy at different distances. The performance evaluation results are presented in Section VII, and Section VIII concludes the paper.

II. Background

The variation in transmission energy requirements suggests that a better scheduling might be required to meet the different transmission needs. For example, a sensor could harvest energy until its storage is completely full before transmitting packets. In this way, it would run to avoid preventable failures and top off between transmissions. Unfortunately, storing excess energy is wasteful due to the nonlinear energy receiving characteristics of capacitor. Sensors use capacitors for energy storage as they are well suited to energy harvesting devices. They charge quickly, recharge indefinitely, are small and inexpensive, and are nontoxic. However, capacitors store energy faster when they are closer to empty than when nearly fully charged. This nonlinearity is fundamental to the way capacitors work. As the capacitor voltage, which increases with increasing charge, approaches the voltage supplied by the energy harvesting circuitry, the charging current decreases to zero. Thus, to increase the transmission rate, it makes sense to operate with a lightly charged capacitor.

Energy is wasted when a sensor starts too early and fails to complete the transmission, or waits too long and inefficiently collects excess energy. How much energy is wasted in these cases depends on how a sensor converts RF energy into harvested energy and consumes the energy.

RF power received at a sensor decreases as fast as the square of its distance from the reader. In practice, this means that the available energy varies by more than an order of magnitude over distances. Recent work on WISP shows that the voltage of the WISP capacitor [25] as it charges at different distances as illustrated in Fig. 2, where $\mathrm{dis}_1 < \mathrm{dis}_2 < \mathrm{dis}_3$, a farther distance results in low energy receiving efficiency. This is the expected behavior. A capacitor's charging rate decreases by a factor of the base of natural logarithms, and asymptotically approaches zero as the capacitor charges to the maximum voltage. This charging behavior has two implications. First, it shows the effects of distance. Far from the charger, the low received power limits the maximum energy that can be stored, while at a near distance it rises quickly to a higher energy level.

The second implication is that, even for a fixed input power, it is inefficient to charge to a higher voltage than necessary. Because the rate at which energy accumulates in a capacitor decreases exponentially as it charges, storing excess energy wastes time. There is a penalty for charging too high and leaving spare energy in the capacitor. In a sense that leftover energy was cheaper to store. This effect is magnified by the linear regulator of the WISP, which consumes more power when there is a higher charge on a capacitor.

III. Problem Definition

Based on the nonlinear but logarithmic energy receiving behavior, we now give the formal problem definition. Consider a situation in an energy harvesting system where energy can be stored in a capacitor and the energy receiving rate varies over time. Assume that we know how long to sleep to store sufficient energy. And let $H(t)$ be the energy harvesting function, and $H^{-1}(b)$ be the time it needs to receive b units of energy. We can get the $H^{-1}(b)$ for a given $H(t)$ easily, especially when we consider a discrete situation. The energy harvesting function is logarithmic with capacitor as shown in Fig. 1. We only use a function $H(t)$ to represent the energy harvesting process, but neglect how this function looks like. This is because we not only consider the logarithmic energy receiving function but also consider the general problem where given any energy harvesting function that the rate decreases over time. The analysis of our proposed algorithms is true for the general problem, that is, our algorithms not only can solve the problem where the charging curve is logarithmic, but also are suitable for the general situation that the charging rate are nonincreasing (the receiving of energy is getting slower and slower).

Because data transmission consumes the most energy [26], [27], we neglect the energy consumption for other functions, and only consider the energy consumption by data transmission in this paper. It can provide a theoretical performance bound guarantee of the energy consumption because of data transmission. This can also be regarded as the situation where a node can only cache and forward data, and data can be transmitted as long as there is enough energy.

Assume the charging periods is denoted as $E = \{E_1, E_2, \ldots, E_m\}$, $E_i = \langle s_i, l_i \rangle$, where s_i is the start time of the ith charging period and l_i is the length of the ith charging period, that is, it takes a duration of time l_i to charge a capacitor from time s_i. Given n packets $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ that should be transmitted, $P_i = \langle a_i, b_i \rangle$, where a_i is the arriving time and b_i is the packet size which is represented as the units of energy (that is, it takes b_i units of energy to transmit packet P_i). The problem is to minimize the total completion time when there is enough energy to transmit all the packets, or maximize the number of the transmitted packets when there is no enough energy to transmit all the packets. This problem is NP-hard even when we do not consider the charging periods and the decreasing energy harvesting rate characteristic [28], [29]. Therefore, we should try to design approximation algorithms for the offline case where the information of all the packets and the charging periods is known in advance, and online algorithms for the case where the information is not known in advance. Both offline and online algorithms should be evaluated through comparing the worst-case performance of the given algorithm with the optimal solution.

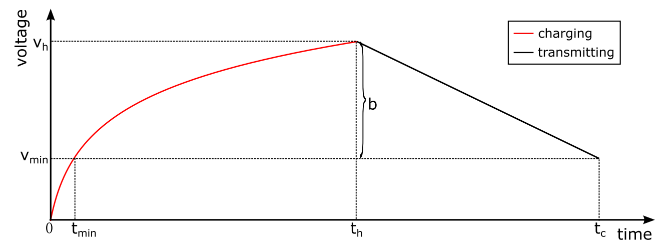


Fig. 3. Illustration of energy harvesting and packet transmission.

This paper assumes that data packets could only be transmitted after the charging process where the received energy is higher than a certain degree. This assumption is reasonable because in many cases a device can only be powered on after the device receives a minimum amount of energy, and the energy receiving rate is always lower than the energy usage rate in data transmission. If the transmission process starts after the device receiving enough amount of energy, it can guarantee the success of the packet transmission, otherwise, it may increase the failure probability of data transmission.

Let the data transmission rate be R, given a packet with size b, then the transmission time of this packet is $\frac{b}{R}$. Let the time starting to harvest energy be 0, and the energy harvesting function be $H(t)$, then the energy used to transmit data $E_{\mathrm{trans}} = V_h - V_{\min}$, and $E_{\mathrm{trans}} = b$, thus $V_h = V_{\min} + b$. Therefore, we can derive the minimum energy harvesting time t_h so that $H(t_h) = V_{\min} + b$. Fig. 3 shows an example of the energy harvesting and packet transmission process, where t_h is the minimum time it needs to receive enough energy to transmit the packet with size b. $t_{\min}$ is the minimum time it needs to receive energy to power on a device. The transmission time $\frac{b}{R} = t_c - t_h$, and the completion time is t_c in this example.

In Fig. 3, the increasing red curve represents the energy receiving process and the decreasing line represents the data transmission process. A device should receive a minimum amount of energy $v_{\min}$ so as to be powered on and start the transmission process. As the energy receiving rate decreases when the capacitor gets more energy, therefore, the red curve increases slower and slower, which makes it looks like a logarithmic function. We assume that the energy consumption is only caused by data

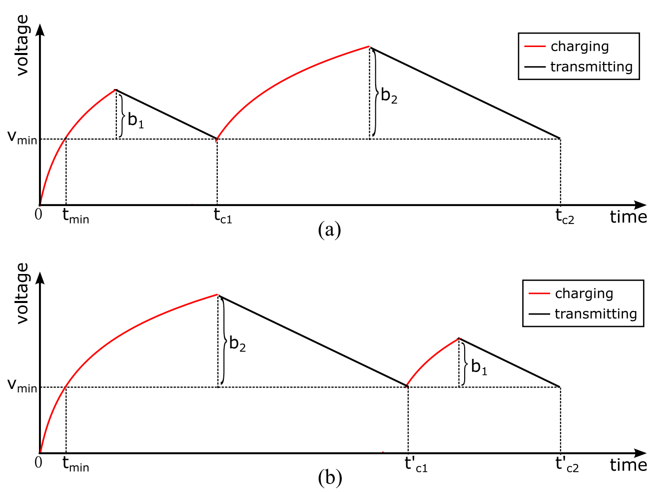


Fig. 4. Different sending order. (a) Sending order 1. (b) Sending order 2.

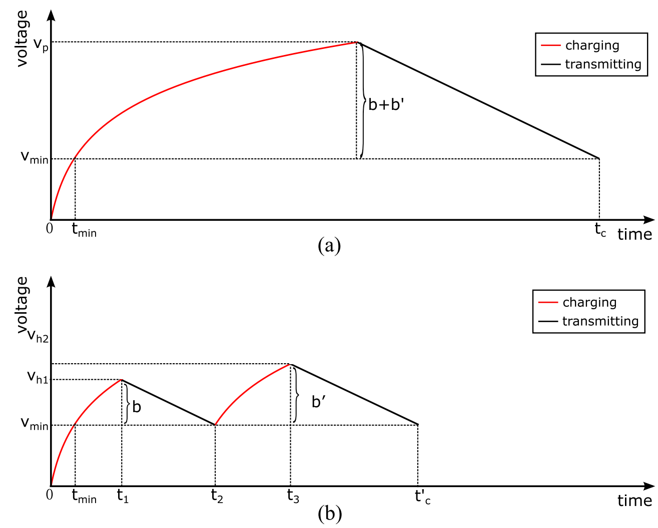


Fig. 5. Illustration for split transmission. (a) Nonsplit transmission. (b) Split transmission.

transmission, and assume that the energy consumption is linear to the size of the packet. This is reasonable because data transmission usually consumes the most energy at a sensor node. The energy consumption function is linear as shown in Fig. 3. Packets with longer size consume more energy and packets with shorter size consume less energy.

IV. Enough Harvesting Time

We start with a simple problem where there is only one charging period for the ease of understanding. We assume that this charging period is long enough so that it can receive enough energy to transmit all the data packets, i.e., $E = \{E_1\}$ and $E_1 = \langle 0, +\infty\rangle$. This is an extreme case but a real one. It can help us understand the analysis of the problem, and it is a special situation where a device stays in the charging range of the wireless energy transmitter. Because a device can receive enough energy so that all the packets can be successfully transmitted, we now consider the problem of minimizing the total completion time. The completion time of a packet is the time in which the packet is completely transmitted. The study of total completion time is useful because it is a metric to measure the minimum average time that a packet should wait so as to be transmitted.

Because data packets arrive at different time, when a packet arrives, if a device receives enough energy, it needs to decide whether to transmit the packet immediately or to wait some time to transmit the packet to improve energy utility. If it does not receive enough energy, the device needs to determine how much time is required to receive enough energy. If we use the method of first arriving first transmitting, then it may result in that a packet with long size is transmitted first while a packet with short size has to wait for a long time, increasing the total completion time. An example is shown in Fig. 4, where there are two packets to be transmitted at time 0. The packet sizes are b_1 and b_2, respectively. Let $b_1 < b_2$. If we transmit packet with size b_1 first, then the total completion time is $t_{c_1} + t_{c_2}$ as shown in Fig. 4(a). If we transmit packet with size b_2 first, then the total completion time is $t'_{c_1} + t'_{c_2}$ as shown in Fig. 4(b), which is larger than that of the first approach because $t'_{c_1} > t_{c_1}$ and $t_{c_2} = t'_{c_2}$.

Definition 1: ***Split transmission*** is defined as that a packet can be split into some smaller subpackets, and this packet is successfully transmitted only when all its subpackets are successfully transmitted.

Easily, in nonsplit transmission, a packet cannot be split into subpackets. It can only be transmitted as a whole.

Theorem 1: If the energy receiving rate decreases as it receives more energy, and the energy is 0 initially, then for any packet, the completion time of split transmission is less than that of nonsplit transmission.

Proof: Let the size of the packet P be b, the arriving time be 0, and the initial energy be 0. Let $H^{-1}(b)$ denote the charging time to receive enough energy to transmit packet P, then the completion time of the nonsplit transmission method is $C = H^{-1}(b) + \frac{b}{R}$, where R is the transmission rate for transmitting any packet. $H^{-1}(v_{\min})$ is the time to receive energy $v_{\min}$. Let $h(b) = H^{-1}(b) - H^{-1}(v_{\min})$, then it can derive $C = H^{-1}(v_{\min}) + h(b) + \frac{b}{R}$. If we use the split transmission method, without loss of generality, let packet P be split into m subpackets, the size of the ith subpacket be b_i, then $b = \sum_{i=1}^{m} b_i$. Let the energy receiving time for the ith subpacket be $H^{-1}(b_i)$, and $h(b_i) = H^{-1}(b_i) - H^{-1}(v_{\min})$, and the completion time of the split transmission method is $C' = H^{-1}(v_{\min}) + \sum_{i=1}^{m} h(b_i) + \frac{\sum_{i=1}^{m} b_i}{R} = H^{-1}(v_{\min}) + \sum_{i=1}^{m} h(b_i) + \frac{b}{R}$. Because the charging rate decreases as it receives more energy, it has $\sum_{i=1}^{m} h(b_i) \le h(b)$, thus $C' \le C$. ■

An example is demonstrated in Fig. 5, where a packet with size $b + b'$ is transmitted as a whole, the completion time is t_c in Fig. 5(a), where $t_c = H^{-1}(v_{\min}) + h(b + b') + \frac{b+b'}{R}$. In Fig. 5(a), $H^{-1}(v_{\min}) = t_{\min}$, $h(b + b') = t_p - t_{\min}$, and $\frac{b+b'}{R} = t_c - t_p$. But, if the packet is divided into two subpackets with size b and b', respectively, then the completion time of this packet is t'_c in Fig. 5(b), where $t'_c = H^{-1}(v_{\min}) + h(b) + \frac{b}{R} +$

$h(b') + \frac{b'}{R} = H^{-1}(v_{\min}) + h(b) + h(b') + \frac{b+b'}{R}$. In Fig. 5(b), $H^{-1}(v_{\min}) = t_{\min}$, $h(b) = t_1 - t_{\min}$, $h(b') = t_3 - t_2$, $\frac{b}{R} = t_2 - t_1$, and $\frac{b'}{R} = t'_c - t_3$. Because the energy receiving rate decreases, it needs a much shorter time to receive enough energy to transmit a packet if the packet is divided into multiple subpackets. Thus, $h(b) + h(b') \leq h(b + b')$. Therefore, $t'_c \leq t_c$.

From Theorem 1, the split transmission can reduce the completion time; thus, it seems that it is better to split a packet into more subpackets. However, actually, because the split transmission can result in extra transmission cost, for example, more subpackets result in more packet headers. The split transmission method can also result in extra cache cost. It is a great waste of the constraint transmission and cache resource in low-power sensor nodes. Therefore, this paper considers the problem that the packets can only be transmitted in the nonsplit transmission method. We introduce the split transmission because we will design a nonsplit transmission algorithm based on a split transmission algorithm.

Definition 2: ***Preemption*** is defined as that a packet P_1 which is being transmitted stops to wait for a new arriving packet P_2 to be successfully transmitted, and then it continues to transmit packet P_1.

In this definition, P_2 is said to preempt P_1 or P_1 is preempted by P_2. Preemption transmission has the property of split transmission. Packet P_1 is preempted by P_2, then the transmission of P_1 is a split transmission. But preemption transmission is different from split transmission in a way that preemption transmission can only happen at the time a packet is being transmitted and a new packet arrives. Preemption transmission is a type of split transmission. Also, this paper only considers the problem that the packets can be transmitted in nonpreemption transmission method.

A. Preemption Algorithm

We first give a preemption transmission method and then a nonpreemption transmission method is designed based on the preemption transmission method. For simplify, let the initial energy be 0. The preemption transmission is described in Algorithm 1, where at any time a new packet arrives or a packet is completely transmitted, it always selects the packet with the smallest completion time to transmit, and preempts if any packet is being transmitting.

Theorem 2: In the preemption transmission manner, transmitting the packet with the smallest completion time first is optimal.

Proof: Assume at time t, the packet with the smallest completion time is P, and the remaining time is $h + \frac{b}{R}$, where h is the time of receiving enough energy to transmit packet P, and $\frac{b}{R}$ is the time for transmitting packet P. If the capacitor already has enough energy to transmit the remaining packet, then $h = 0$. The completion time of packet P is $t + h + \frac{b}{R}$. Let the packets arrived before time be $t + h + \frac{b}{R}$ but have not been completely transmitted be $\mathcal{P}$. Then, it can be found that, because of transmitting packet P, it results in that the completion times of all the packets in $\mathcal{P}$ are increased by $h + \frac{b}{R}$. After the complete transmission of P, let the energy receiving time of one packet P^+ in $\mathcal{P}$ be h^+.

If there exists an optimal method that does not transmit P at time t, without loss of generality, let the optimal method transmit packet P' at time t. Then, it results in the completion times of packets in $\mathcal{P}$ being increased by $h' + \frac{b'}{R}$. After the transmission of P', let the energy receiving time of one packet P^+ in $\mathcal{P}$ be $h^{+'}$.

Because $h + \frac{b}{R}$ is the smallest remaining time, we have $h + \frac{b}{R} \leq h' + \frac{b'}{R}$. It is easy to find that the charging time is longer if the remaining time is longer, thus $\frac{b}{R} \leq \frac{b'}{R}$. Therefore, $h^+ \leq h^{+'}$, then we have $h + \frac{b}{R} + h^+ \leq h' + \frac{b'}{R} + h^{+'}$. For any packet in $\mathcal{P}$, this inequation is true. Therefore, transmitting the packet with the smallest completion time first is an optimal solution in the preemption transmission manner. ∎

Algorithm 1: Smallest Completion Time First.

Input: $[a_i, b_i]$, n_i, $1 \leq i \leq n$

Output: Schedule to minimize the total completion time.

1: At any time a packet arrives or a packet is completely transmitted,

2: Compute the completion time $\tau = h + \frac{b}{R}$ for every packet that has arrived but not completely transmitted, where $h = \max\{0, H^{-1}(b + v_{\min}) - H^{-1}(e)\}$ is the charging time the packet needs to receive enough energy to transmit the remaining amount of data b, e is the energy in the capacitor currently, and $H^{-1}(x)$ is the time to receive x units of energy. $\frac{b}{R}$ is the time it needs to completely transmit the packet.

3: Select the packet with the smallest τ to transmit.

Algorithm 2:

Transmit the packets in the same order of their completion times as that in Algorithm 1.

B. Nonpreemption Algorithm

This paper assumes that sensor nodes can only send packets in the nonsplit and nonpreemption transmission method. So, we transform Algorithm 1 into the nonpreemption algorithm. The transformation is quite simple. We send the packets in the same order of their completion times as that in Algorithm 1. That is, the packet that can be completely transmitted with the smallest completion time is transmitted first.

In Algorithm 1, the packet being transmitted can be preempted. As new arriving packet with smaller completion time can preempt the packet that is being transmitted. It should wait until the complete transmission of the new arriving packet, and then continue to transmit the preempted packet. While in Algorithm 2, the transmission cannot be preempted. A packet should be successfully transmitted and then start to transmit another packet.

Theorem 3: The approximation factor of Algorithm 2 is $1 + \alpha$, where $\alpha = \max \frac{len_x}{len'_x}$ and len_x is the time used to receive

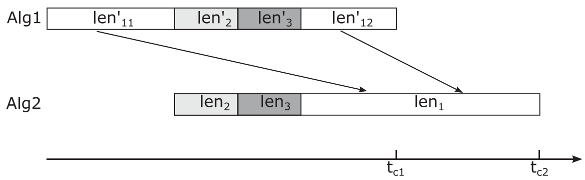


Fig. 6. Illustration of Algorithms 1 and 2.

energy and transmission for a packet P_x in Algorithm 2, and len'_x is the total time used to receive energy and transmission for a packet P_x in Algorithm 1.

Proof: let C_j^P be the completion time of packet j with preemption transmission in Algorithm 1, and C_j^N be the completion time of packet j with nonpreemption transmission in Algorithm 2. In Algorithm 2 at time C_j^N, since packet j is successfully transmitted, all the packets that completed before C_j^P are successfully transmitted. Let len_x be the sum of the energy receiving time and transmission time of packet x in nonpreemption transmission. In the worst case, packet j arrives first but is the one that is lastly successfully transmitted, then $C_j^N \leq C_j^P + \sum_{C_x^P \leq C_j^P} len_x$. Let len'_x be the sum of energy receiving time and transmission time of packet x in preemption transmission, then we have $\sum_{C_x^P \leq C_j^P} len'_x \leq C_j^P$. It can be derived that $len'_x \leq len_x$ from Theorem 1. Let $\alpha = \max \frac{len_x}{len'_x}$, then $\sum_{C_x^P \leq C_j^P} len_x \leq \alpha C_j^P$, therefore $C_j^N \leq (1+\alpha)C_j^P$. This inequation holds for every packet j, then it completes the proof. ∎

Fig. 6 illustrates an extreme case for Algorithm 2. In Fig. 6, there are three packets to be transmitted. Packet 1 arrives first, followed by packets 2 and 3. In Fig. 6, the top part is the transmission process of Algorithm 1, while the bottom part is the transmission process of Algorithm 2. Algorithm 1 would transmit packet 1 first, when packet 2 arrives, the completion time of packet 2 is earlier, then packet 2 preempts packet 1. After the transmission of packet 2, packet 3 also preempts packet 1. Finally, when packets 2 and 3 are both successfully transmitted, it continues to transmit packet 1. The completion time of packet 1 is t_{c1}. In this figure, len'_2 is the sum of energy receiving and transmission time for packet 2 and len'_3 is the sum of energy receiving and transmission time for packet 3. $len'_1 = len'_{11} + len'_{12}$ is the total energy receiving and transmission time for packet 1, as packet 1 is divided into two parts. Algorithm 2 sends packets in a nonpreemption manner and transmits the packet in the same order of completion times as that in Algorithm 1. Thus, Algorithm 2 will transmit packets 2 and 3 first, and then transmits packet 1 as a whole, so the completion time of packet 1 is t_{c2}. It is obvious that $t_{c2} \leq t_{c1} + \sum_{C_x^p \leq C_1^p} len_x$. From Theorem 1, $len'_{11} + len'_{12} \leq len_1$. Let $\alpha = \max \frac{len_x}{len'_x}$, then $t_{c2} \leq (1+\alpha)t_{c1}$.

C. Online Algorithm

In real applications, packets cannot arrive simultaneously. Therefore, we need to design an online algorithm to solve the problem. The main idea is that, for any packet, it should wait for a particular time to find whether there is a packet that arrives later but can be completed earlier. The particular time is set to $h + \frac{b}{R}$, where $h = \max\{0, H^{-1}(b + v_{\min}) - H^{-1}(e)\}$ is the charging time the packet needs to receive enough energy to transmit the packet, b is the packet size, and e is the energy in the capacitor currently. The algorithm is described in Algorithm 3.

Algorithm 3:

Input: $[a_i, b_i]$, n_i, $1 \leq i \leq n$
Output: Schedule to minimize the total completion time.
At any time t, compute $h + \frac{b}{R}$ for every not transmitted packet;
Let packet i have minimum $h + \frac{b}{R}$;
if $h + \frac{b}{R} \leq t$ **then**
    Transmit packet i immediately;
else
    Wait for next time slot;
Go to the start of the algorithm;

Theorem 4: The competitive ratio of Algorithm 3 is $\max\{2, \beta\}$, where $\beta = \max\{\frac{h+\frac{b}{R}}{\frac{b}{R}}\}$.

Proof: Let the online Algorithm 3 be ALG, and the optimal algorithm be OPT. ALG will transmit packet i with the earliest completion time at time t. Because of the transmission of packet i, it results in the packets transmitted in the period $[t, t + h + \frac{b}{R}]$ in OPT to be delayed. Let these delayed packets be $\mathcal{P}$, then the completion time of packets in $\mathcal{P}$ is all increased by $h + \frac{b}{R}$. The total increased time for the packets in $\mathcal{P}$ is at most $\sum_{P\in\mathcal{P}}(h + \frac{b}{R}) + A$, where A is the total completion time for the packets in $\mathcal{P}$ regarding time $t + h + \frac{b}{R}$ as time 0. Any packet would not be transmitted before time t, otherwise, ALG would not transmit the packet i that has the earliest completion time.

In OPT, the packets in $\mathcal{P}$ would be transmitted within period $[t, t + h + \frac{b}{R}]$. The total completion time for the packets in $\mathcal{P}$ is at least $\sum_{P\in\mathcal{P}}(t) + B$, where B is the total completion time for the packets in $\mathcal{P}$ regarding time t as time 0. If we do not take into account the energy receiving time, then at time $t + h + \frac{b}{R}$ all the packets in $\mathcal{P}$ have already arrived. Thus, ALG is optimal for the transmission of the packets in $\mathcal{P}$ after time $t + h + \frac{b}{R}$, therefore, $A \leq B$. Because $h + \frac{b}{R} \leq t$, $\sum_{P\in\mathcal{P}}(t + h + \frac{b}{R}) + A \leq 2\sum_{P\in\mathcal{P}}(t) + B \leq 2(\sum_{P\in\mathcal{P}}(t) + B)$. If we take into account the energy receiving time, then at time t the packets in $\mathcal{P}$ do not arrive, but it can begin to receive energy. In an extreme case, when the first packet in $\mathcal{P}$ arrives, it receives enough energy to transmit all the packets in $\mathcal{P}$. In this case, B does not include any energy receiving time, and it only includes the transmission time. Let $\beta = \max\{\frac{h+\frac{b}{R}}{\frac{b}{R}}\}$, then $A \leq \frac{h+\frac{b}{R}}{\frac{b}{R}}B$. We have $\sum_{P\in\mathcal{P}}(t + h + \frac{b}{R}) + A \leq 2\sum_{P\in\mathcal{P}}(t) + \frac{h+\frac{b}{R}}{\frac{b}{R}}B \leq \max\{2, \frac{h+\frac{b}{R}}{\frac{b}{R}}\}(\sum_{P\in\mathcal{P}}(t) + B)$. ∎

An extreme example is illustrated in Fig. 7. At time t, ALG transmits packet i whose sum of the energy receiving time and transmission time $h + \frac{b}{R} \leq t$ and the completion time of packet i is the earliest. Because of the transmission of packet i, it

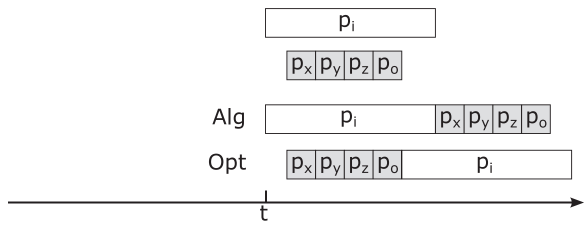


Fig. 7. Extreme example for Algorithm 3.

results in that all the completion times of packets x, y, z, o whose arriving time is a little later than packet i increased by $h + \frac{b}{R}$. In an optimal solution, it can transmit packets x, y, z, o first and then transmit packet i, and the total completion time of such a solution is less.

V. No Enough Harvesting Time

In some situation, there may not be enough energy to transmit a packet. It should wait for the next charging period so as to get enough energy. If a sensor is located in a fixed position and can be charged by a special charger, then the sensor can receive enough energy and transmit all the packets. However, under many circumstances, a sensor may be mobile, and the charger may also move over time. Both such situations result in that sensors cannot receive enough energy. Therefore, we should consider how to transmit as many packets as possible if there is no enough harvesting time.

Now, the problem can be defined as follows. Given n packets and m charging periods, each packet i is denoted as $\langle a_i, b_i\rangle$, where a_i is the arriving time and b_i is the size of packet i. Each charging period j is denoted as $\langle s_j, l_j\rangle$, where s_j is the start time and l_j is the length of the charging period j. Assume that the initial energy in the capacitor is zero, then the problem is how to select and transmit some packets so as to maximize the number of successfully transmitted packets.

A. Only One Charging Period

For ease of understanding, we begin from a simple case where there is only one charging period. In this case, it is difficult to determine whether to transmit the packet immediately when it receives enough energy, or wait for the future arriving packet because the later packet may be shorter and can save the received energy to transmit more packets. As shown in Fig. 8, different transmission strategies result in different numbers of transmitted packets. In this example, the charging period is $\langle 0, l\rangle$. Fig. 8(a) shows an example where transmitting two packets with size b' and b'' after the entire charging period. It can transmit the two packets. Fig. 8(b) shows that if it transmits the packet with size b' first and b'' later, then it can also transmit the two packets. Fig. 8(c) shows that if it transmits the packet with size b'' first, then it can only transmit one packet.

Therefore, we divide the problem into two subproblems. One is to transmit more packets that arrive before the end of the charging period. The other is to transmit more packets that arrive after the end of the charging period.

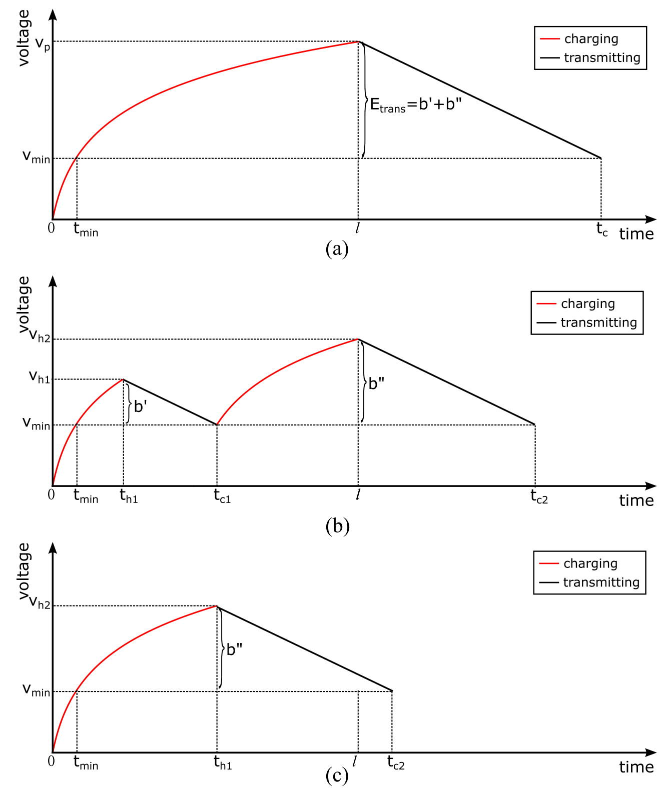


Fig. 8. Different transmission strategies result in different numbers of transmitted packets. (a) Transmission after charging period. (b) Transmission within charging period. (c) Transmission within charging period.

Algorithm 4:

For each packet, compute $\tau = h + \frac{b}{R}$, where $h = H^{-1}(b + v_{\min})$ is the time to receive energy for transmitting packet with size b, and $\frac{b}{R}$ is the transmission time. Transmit the packet with the minimum τ until there is no enough energy to transmit any packet.

For the first subproblem, the solution is described in Algorithm 4, whose idea is to always transmit the packet with the smallest size first.

Theorem 5: The approximation of Algorithm 4 is $2 + \lceil \frac{h}{b/R} \rceil$.

Proof: Let Algorithm 4 be ALG, and the optimal algorithm be OPT. Each packet i transmitted by ALG results in some packets in OPT not being transmitted. Let the time when packet i is transmitted be t. Then, at time t, packet i's $h + \frac{b}{R}$ is the minimum. It is easy to find that within time period $[t, t + h + \frac{b}{R}]$, packet i has the smallest size b, and the sizes of other packets are all larger than b. Then, within time period $[t, t + h + \frac{b}{R}]$, it can result in at most $\lceil \frac{h}{b/R} \rceil$ packets not being transmitted in OPT, where we assume it has received enough energy to transmit $\lceil \frac{h}{b/R} \rceil$ packets in the earlier time in OPT. Two packets that slightly overlap with packet i at the two ends of the time interval $[t, t + h + \frac{b}{R}]$ may not be transmitted in ALG. Therefore, the approximation factor is $2 + \lceil \frac{h}{b/R} \rceil$. ■

Fig. 9 illustrates an extreme example for Algorithm 4. In this example, any packet transmitted by ALG results in some other

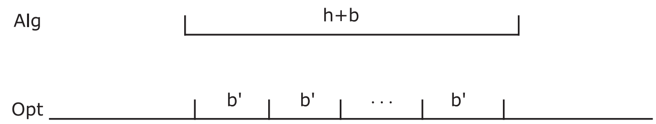


Fig. 9. Extreme example for Algorithm 4.

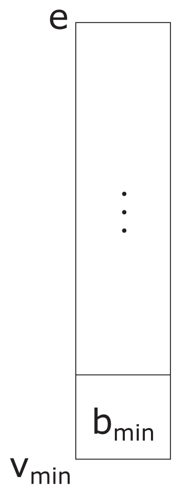


Fig. 10. Example for Algorithm 5.

Algorithm 5:

After all the packets arrive, always transmit the packet with the smallest size each time until there is no enough energy to transmit any packet.

packets that could not be transmitted while these packets could be transmitted in OPT. The number of these packets is at most $2 + \lceil \frac{h}{b/R} \rceil$ in the worst case.

Now, let us consider the second subproblem in which a sensor can receive energy within the entire charging period and does not transmit any packet. After the charging period, because there is only one charging period, it can wait until all the packets arrive and then begin to transmit the packets.

Theorem 6: Algorithm 5 transmits the most packets when it does not transmit any packet within the charging period.

The proof of this theorem is quite simple. A sensor receives energy in the entire charging period. Therefore, it receives the most energy for the packets that arrive after the end of the charging period. For such packets, always transmit the packet with the minimum size resulting in the minimum energy consumption. Thus, the algorithm can transmit the most packets. Fig. 10 shows an example for Algorithm 5, where e is the energy stored in the capacitor and $v_{\min}$ is the minimum required energy to start the transmission. Transmitting the packet with the smallest size $b_{\min}$ reserves more energy for the remaining packets. Therefore, it can transmit the most packets that arrive after the end of the charging period.

Theorem 7: The approximation factor of Algorithm 6 is $3 + \lceil \frac{h}{b/R} \rceil$.

Now, the solution for the whole problem is the better solution for the two subproblems.

Algorithm 6:

Compare the numbers of packets transmitted by Algorithms 4 and 5. Select the one that can transmit more packets. If there is any remaining energy, transmit packets in the nondecreasing order by their sizes.

Proof: Let Algorithm 6 be ALG, and the optimal algorithm be OPT. Let OPT $= n_1 + n_2$, where n_1 is the number of packets transmitted in OPT, which arrive before the end of the charging period, and n_2 is the number of packets transmitted in OPT, which arrive after the end of the charging period. From Theorem 5, the approximation ratio of Algorithm 4 is $2 + \lceil \frac{h}{b/R} \rceil$, that is, for those packets that arrive before the end of the charging period, the number of packets that can be transmitted by Algorithm 4 is at least $n'_1 \geq \frac{n_1}{2+\lceil \frac{h}{b/R} \rceil}$. From Theorem 6, Algorithm 5 transmits the most packets that arrive after the end of the charging period, that is, $n'_2 \geq n_2$. Algorithm 6 selects between Algorithms 4 and 5 the one that can transmit more packets, i.e., ALG $= \max\{n'_1, n'_2\} \geq \max\{\frac{n_1}{2+\lceil \frac{h}{b/R} \rceil}, n_2\}$. It is easy to get OPT $\leq (3 + \lceil \frac{h}{b/R} \rceil)$ALG. ■

B. m Charging Periods

We next consider the general problem where there are m charging periods. We use the similar algorithm as proposed in Algorithms 4–6.

We also divided the problem into two subproblems. The first subproblem is how to select and transmit the packets that arrive within the charging periods. For the first charging period, we also need consider the packets that arrive before the start of the charging period. For other charging periods, we only consider how to transmit the packets that arrive within the charging periods. The second subproblem is how to transmit the packets that arrive out of the charging periods. The main idea behind the two subproblems is that there may exist two extreme cases. One is that many small packets arrive within the charging periods while long packets arrive at the time out of the charging periods. In this case, transmit the small packets immediately after receiving enough energy is a better solution. The second one is that many small packets arrive at the time out of the charging periods, while long packets arrive within the charging periods. In this case, it is better to transmit as many small packets that arrive out of the charging period as possible, then receiving energy in each entire charging period can receive the most energy for transmitting packet arrive at the time out of the charging periods.

For the first subproblem, we consider how to transmit the packets that arrive within the charging period. The solution is that always transmit the packet with smallest $h + \frac{b}{R}$. In this subproblem, we also consider the packets that arrive before the first charging period.

The approximation factor of Algorithm 7 is $2 + \lceil \frac{h}{b/R} \rceil$, the same as that of Algorithm 4. The analysis is similar to the proof of Theorem 5.

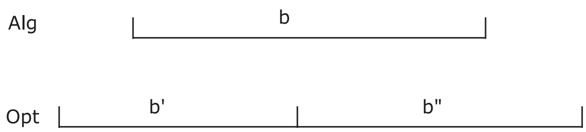


Fig. 11. Extreme example for Algorithm 8.

Algorithm 7:

Only consider the packets that arrive within the charging periods and the packets arrive before the first charging period.

In each charging period, always transmit the packet with the smallest $h + \frac{b}{R}$ first, where $h = H^{-1}(b + v_{\min})$ is the time to receive energy for transmitting packet with size b, and $\frac{b}{R}$ is the transmission time.

Algorithm 8:

Only consider the packets that arrive at the time out of the charging periods, except the packets that arrive before the first charging period. Within every charging period, it does not transmit any packet, it only receives energy.

After each charging period, use at most half of the energy to transmit the packets arrived before the next charging period in the nondecreasing order of their sizes.

For the second subproblem, we consider how to transmit the packets that arrive at the time out of the charging periods, except the packets that arrive before the first charging period. The solution is that it does not transmit any packet in every charging period. That is, it receive energy in all entire charging periods. After each charging period, it transmit the packets that arrive before the next charging period in the nondecreasing order by their sizes. After each charging period, it uses at most half of the received energy to transmit packets. The detail of this solution is described in Algorithm 8.

Theorem 8: The approximation ratio of Algorithm 8 is 4.

Proof: Let us only consider the transmission of the packets in each noncharging period. The algorithm always transmits the packet with smallest size first. Therefore, any transmitted packet by the algorithm may result at most two other packets that could be transmitted in an optimal solution. An example is shown in Fig. 11.

Let us continue to consider the energy usage. After each charging period, it use at most half of the energy to transmit the packets arrived before the next charging period. This method can transmit half of the packets in an optimal solution for the packets arrived before the next charging period.

Therefore, any optimal solution can transmit at most four times the number of packets than Algorithm 8 for the packets that arrive at the time out of the charging periods. ■

Theorem 9: The approximation ratio of Algorithm 9 is $6 + \lceil \frac{h}{b/R} \rceil$.

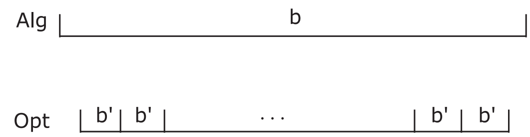


Fig. 12. Illustration for no constant online algorithm.

Algorithm 9:

Compare the number packets which transmitted by Algorithms 7 and 8. Select the one which can transmit more packets. If there is any remaining energy, transmit packets in the non-decreasing order by their sizes.

The method for the problem with m charging periods is also use a better solution which can transmit more packets between the two solutions for the subproblems, as shown in Algorithm 9.

The proof is similar to that of Theorem 7. The optimal solution cannot transmit more packets than that the two optimal solutions for the two subproblems do. The approximation ratio is the sum of the approximation ratios of Algorithms 7 and 8.

We give an approximation algorithm to solve the offline problem. However, in practical application, the packets will arrive one by one. At any time, it does not know the information of the future arriving packets and charging periods. Therefore, it needs to design online algorithms. Unfortunately, one cannot find a constant competitive ratio online algorithm.

Theorem 10: There is no constant competitive ratio online algorithm for the this problem.

Proof: It is easy to find that any online algorithm will determine to transmit packets that have arrived currently. It does not know the information of the future arriving packets. Therefore, any packet that is determined to be transmitted may result in that many smaller packets that arrive a little later cannot be transmitted. ■

Fig. 12 shows an example where at any time, any online algorithm ALG determines to transmit a packet of size b, it can result in that many very small packets of size b' cannot be transmitted.

Because there is no constant competitive ratio online algorithm, but based on our simulation, as long as the sizes of the packets are within a normal range, a first arriving first transmitting approach can give a good performance.

VI. Different Harvesting Distances

We next do study the problem where the sensor node may receive energy at different distance. As shown in Fig. 2, the energy receiving rate varies at different charging distances. And the maximum energy that can be received also varies at different charging distances.

Given n packets, each packet i can be denoted as $\langle a_i, b_i \rangle$, where a_i is the arriving time and b_i is the length of the packet, and given m charging periods, each charging period j is denoted as $\langle s_j, l_j, d_j \rangle$, where s_j is the starting time of charging period j, l_j is the length of the charging period j, and d_j is the distance between the charger and sensor node; then the problem is how to schedule and transmit the maximum number of packets.

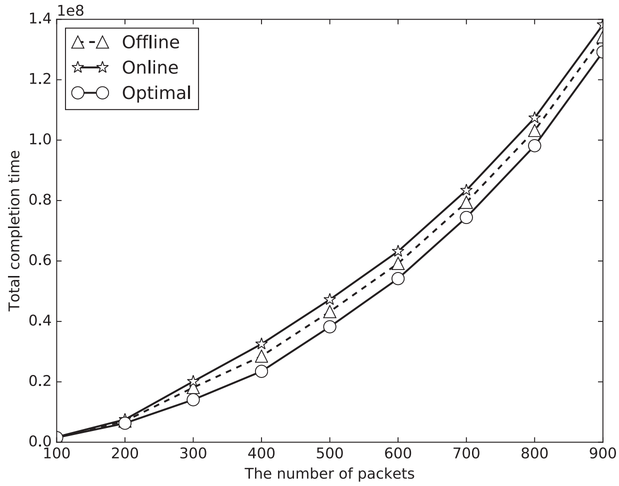


Fig. 13. Illustration for total completion time.

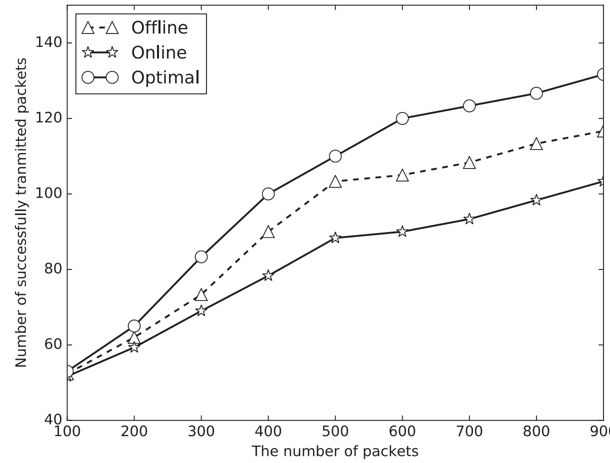


Fig. 14. Illustration for maximum transmitted packets.

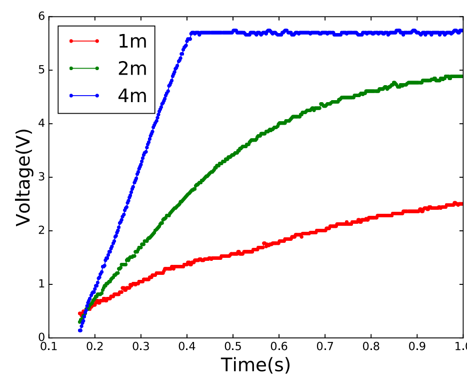


Fig. 15. Charging process.

We use the algorithm similar to Algorithm 9 to solve the problem. If we know the energy receiving function $H(t, d)$, where t is the energy receiving time and d is the energy receiving distance, then the problem is the same to Algorithm 9. And it can get the same approximation ratio as that of Algorithm 9. The approximation ratio is also $6 + \lceil \frac{h}{\frac{b}{R}} \rceil$.

For the online situation, there is still no constant competitive ratio online algorithm to solve the problem. We also use a first arriving first transmitting algorithm to transmit the packet online.

VII. Evaluations

The worst-case performance bounds of both offline and the online algorithms are analyzed theoretically in this paper. In this section, we perform simulations for the algorithm to further validate their average performances on randomly generated dataset.

The performances of the offline approximation algorithm and the online algorithm are evaluated comparing to the optimal solution. It simulate the algorithms using random generated data where the arriving times are random values within $[0, 2000]$ and the packet sizes are random values within $[1, 20]$. The charging periods are random within $[0, 2000]$, where no two charging periods overlap with each other. The energy receiving process is assumed to be logarithmic function, i.e, $H(t) = \omega \log_x(t + 1)$, where the base x is a random value from 2–5 and ω is set to 10. $V_{\max}$ and $V_{\min}$ are set to 30 and 5, respectively. The data transmission rate is set to $\frac{1}{2}$, which means that it takes two units of time to transmit one unit amount of data. Figs. 13 and 14 demonstrate the simulation results of our approximation offline algorithm and the online algorithm for the two different cases where whether the energy is enough to transmitting all the packets, respectively. In Fig. 13, Offline, Online, and Optimal, respectively, represent the total completion time of the offline approximation algorithm, the online algorithm, and a lower bound of the optimal solution. Because it is impractical to compute an optimal result, we relax the problem to be a linear programming problem and use its result as a lower bound of the optimal result, which is denoted as Optimal in Figs. 13 and 14. We evaluate the performance from 100 to 900 packets as shown in Figs. 13 and 14. Both the performances of the offline approximation algorithm and the online algorithm are very close to that of the optimal solution. The ratio between the output of online algorithm and the optimal solution is around 1.1–1.4, and the ratio between the output of offline approximation algorithm and the optimal solution is around 1.05–1.3. This validates the efficiency of our algorithms.

Next, we use the real energy receiving process data, which is extracted from Buettner's work [25, fig. 4] to further perform the simulation. In order to make the simulation simple, we remove the time it takes to power on RF, and only retain the charging process where the charging rate decreases, which is shown in Fig. 15. This charging behavior shows that farther from the energy source can result in lower maximum energy that can be stored. When the energy receiver is enough close to the energy transmitter, the energy receiving process is linear as the blue curve in Fig. 15. $V_{\min}$ is set to 0.5 V, and $V_{\max}$ are set to the maximum voltage in Fig. 15 for the three different distances, respectively. The arriving times are random values within $[0, 100]$ seconds, and the packet sizes are random values within $[0.2, 1.5]$. The charging periods are random within $[0, 100]$ seconds, and the distances are also random. The data transmission rate is still set to $\frac{1}{5}$, which means that it takes 5 units of time to transmit one unit amount of data. We do not use a mathematical function to model the charging process, but only

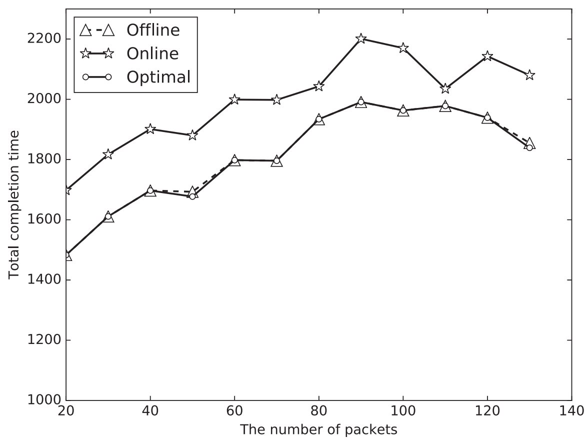


Fig. 16. Illustration for total completion time.

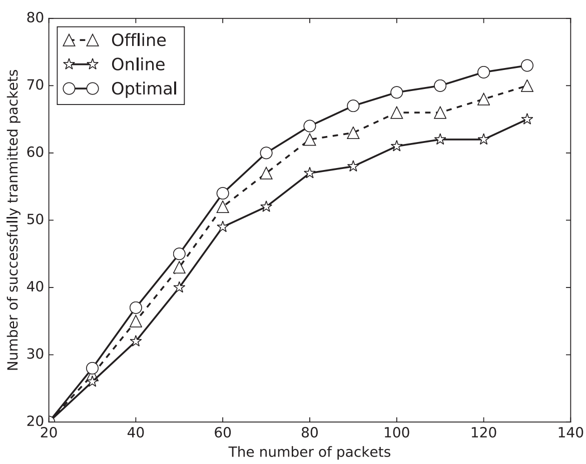


Fig. 17. Illustration for maximum transmitted packets.

use the raw curve to simulate the charging process. Our algorithms not only can solve the problem where the charging curve is logarithmic, but also are suitable for the general situation that the charging rate are nonincreasing. In this set of simulations, the blue curve is not logarithmic while it looks like a linear function whose changing rate is nonincreasing; therefore, it does not need to do any change to the algorithm for this kind of situation. Again, Offline, Online, and Optimal, respectively, represent the total completion time of the offline approximation algorithm, the online algorithm, and a lower bound of the optimal solution. We evaluate the performance from 20 to 130 packets as shown in Figs. 16 and 17. The performance of the offline approximation algorithm is very close to that of the optimal solution including the total completion time and the number of successfully transmitted packets. The online algorithm also shows good results.

VIII. Conclusion

In this paper, we theoretically study data transmission problem in passive sensors with capacitor to supply energy. The energy receiving rate with capacitor is not linear. The rate decreases as the sensor receives more energy. When there is enough energy to transmit all the packets, we give both offline and online algorithm to minimize the total completion time and analyze the approximation and competitive ratio for the proposed algorithm, respectively. When the energy is not enough to transmit all the packets, we propose an offline algorithm to transmit as more number of packets as possible, and the approximation ratio is also analyzed. For this problem, we also prove that there is no constant competitive ratio online algorithm. Simulation results further demonstrates that our algorithms can achieve good performance compared to a lower bound of the optimal solution. In the future work, we will study the weighted version problem where packets have different weights.

References

[1] S. Kumar, "Empowering low-power wide area networks in urban settings," in *Proc. ACM Conf. Special Interest Group Data Commun.*, 2017, pp. 309–321.

[2] Y. Coady, O. Hohlfeld, J. Kempf, R. Mcgeer, and S. Schmid, "Distributed cloud computing: Applications, status quo, and challenges," *ACM Sigcomm Comput. Commun. Rev.*, vol. 45, no. 2, pp. 38–43, 2015.

[3] M. Han, Z. Duan, and Y. Li, "Privacy issues for transportation cyber physical systems," in *Secure and Trustworthy Transportation Cyber-Physical Systems*. Singapore: Springer-Verlag, 2017, pp. 67–86.

[4] A. Teo, A. Mishra, I. Park, Y. J. Kim, W. T. Park, and J. Y. Yong, "Polymeric biomaterials for medical implants and devices," *ACS Biomaterials Sci. Eng.*, vol. 2, no. 4, pp. 454–472, 2016.

[5] J. Katic, "Efficient energy harvesting interface for implantable biosensors," Ph.D. dissertation, Sch. Informat. Commun. Technol., KTH Roy. Inst. Technol., Stockholm, Sweden, 2015.

[6] J. L. Miller, "Wireless power for tiny medical implants," *Physics*, vol. 67, no. 8, pp. 12–14, 2014.

[7] H. Dai *et al.*, "Safe charging for wireless power transfer," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3531–3544, Dec. 2017.

[8] J. Olivo, S. Carrara, and G. De Micheli, "Energy harvesting and remote powering for implantable biosensors," *IEEE Sensors J.*, vol. 11, no. 7, pp. 1573–1586, Jul. 2011.

[9] Xu Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: Privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1868–1878, Dec. 2017.

[10] S. Arslan, S. A. A. Shah, J. J. Lee, and H. W. Kim, "An energy efficient charging technique for switched capacitor voltage converters with low duty-ratio," *IEEE Trans. Circuits Syst. II, Express Briefs*, vol. 65, no. 6, pp. 779–783, Jun. 2018.

[11] S. Xia and L. Chen, "Theoretical and experimental investigation of optimal capacitor charging process in RC circuit," *Eur. Phys. J. Plus*, vol. 132, no. 5, pp. 1–19, May 2017.

[12] J. Li, S. Cheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Approximate holistic aggregation in wireless sensor networks," *ACM Trans. Sens. Netw.*, vol. 13, no. 2, Apr. 2017, Art. no. 11.

[13] D. Pardo *et al.*, "Design criteria for full passive long range UHF RFID sensor for human body temperature monitoring," in *Proc. IEEE Int. Conf. RFID*, Mar. 2007, pp. 141–148.

[14] S. Cheng, Z. Cai, J. Li, and X. Fang, "Drawing dominant dataset from big sensory data in wireless sensor networks," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2015, pp. 531–539.

[15] J. Li, S. Cheng, H. Gao, and Z. Cai, "Approximate physical world reconstruction algorithms in sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 12, pp. 3099–3110, Dec. 2014.

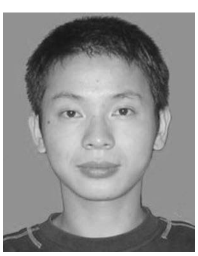
[16] J. Ou, M. Li, and Y. Zheng, "Come and be served: Parallel decoding for cots RFID tags," *IEEE/ACM Trans. Netw.*, vol. 25, no. 3, pp. 1569–1581, Jun. 2017.

[17] S. Cheng, Z. Cai, J. Li, and H. Gao, "Extracting kernel dataset from big sensory data in wireless sensor networks," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 4, pp. 813–827, Apr. 2017.

[18] S. Cheng, Z. Cai, and J. Li, "Curve query processing in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 11, pp. 5198–5209, Nov. 2015.

[19] M. Buettner, R. Prasad, M. Philipose, and D. Wetherall, "Recognizing daily activities with RFID-based sensors," in *Proc. 11th Int. Conf. Ubiquitous Comput.*, New York, NY, USA, 2009, pp. 51–60.

[20] A. P. Sample, D. J. Yeager, P. S. Powledge, A. V. Mamishev, and J. R. Smith, "Design of an RFID-based battery-free programmable sensing platform," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 11, pp. 2608–2615, Nov. 2008.
[21] D.-Y. Kim and M. Jung, "Data transmission and network architecture in long range low power sensor networks for IoT," *Wireless Pers. Commun.*, vol. 93, no. 1, pp. 119–129, Mar. 2017.
[22] J. Pagn, M. Zapater, and J. L. Ayala, "Power transmission and workload balancing policies in ehealth mobile cloud computing scenarios," *Future Gener. Comput. Syst.*, vol. 78, pp. 587–601, 2018.
[23] Z. He, Z. Cai, S. Cheng, and X. Wang, "Approximate aggregation for tracking quantiles and range countings in wireless sensor networks," *Theor. Comput. Sci.*, vol. 607, no. P3, pp. 381–390, 2015.
[24] M. Han, J. Li, Z. Cai, and Q. Han, "Privacy reserved influence maximization in GPS-enabled cyber-physical and online social networks," in *Proc. IEEE Int. Conf. Big Data Cloud Comput.*, 2016. pp. 284–292.
[25] M. Buettner, B. Greenstein, and D. Wetherall, "Dewdrop: An energy-aware runtime for computational RFID," in *Proc. 8th USENIX Conf. Networked Syst. Des. Implementation*, Berkeley, CA, USA, 2011 , pp. 197–210.
[26] M. Lundn and A. Dunkels, "The politecast communication primitive for low-power wireless," *ACM Sigcomm Comput. Commun. Rev.*, vol. 41, no. 2, pp. 31–37, 2016.
[27] H. Dai, H. Ma, A. X. Liu and G. Chen, "Radiation constrained scheduling of wireless charging tasks," in *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, 2018, pp. 314–327.
[28] O. Mashkani and G. Moslehi, "Minimising the total completion time in a single machine scheduling problem under bimodal flexible periodic availability constraints," *Int. J. Comput. Integr. Manuf.*, vol. 29, no. 3, pp. 323–341, 2016.
[29] P. Beldar and A. Costa, "Single machine batch processing problem with release dates to minimize total completion time," *Int. J. Ind. Eng. Comput.*, vol. 9, no. 3, pp. 331–348, 2018.



Xiaolin Fang received the B.S. degree from Harbin Engineering University, China, in 2007, and the M.S. and Ph.D. degrees from Harbin Institute of Technology, Harbin, China, in 2009 and 2014, respectively.

He is currently an Assistant Professor with the School of Computer Science and Engineering, Southeast University, Nanjing, China. His research interests include sensor networks, data processing, image processing, and scheduling.

Junzhou Luo (M'07) received the B.S. degree in applied mathematics and the M.S. and Ph.D. degrees in computer network from Southeast University, Nanjing, China, in 1982, 1992, and 2000, respectively.

He is a Full Professor with the School of Computer Science and Engineering, Southeast University, Nanjing, China. His current research interests include next generation network architecture, network security, cloud computing, and wireless LAN.

Dr. Luo is a member of the IEEE Computer Society and ACM, the Co-Chair of the IEEE SMC Technical Committee on Computer Supported Cooperative Work in Design, and the Chair of ACM SIGCOMM China.

Guangchun Luo received the Ph.D. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2004.

Currently, he is a Professor and the Associate Dean of computer science at the University of Electronic Science and Technology of China, Chengdu, China. His research interests include computer networks, image processing, cloud computing, and big data mining. As the Director of the Credible Cloud Computing and Big Data Laboratory, he has authored/co-authored more than 60 journal and conference papers.

Weiwei Wu (M'14) received the B.S. degree in computer science from the South China University of Technology, Guangzhou, China, and the Ph.D. degree in computer science jointly from the Department of Computer Science, City University of Hong Kong, Hong Kong, and the University of Science and Technology of China, Hefei, China, in 2011.

In 2012, he went to the Mathematical Division, Nanyang Technological University, Singapore, as a Postdoctoral Researcher. He is now an Associate Professor with Southeast University, Nanjing, China. His research interests include optimizations and algorithm analysis, wireless communications, crowd sourcing, cloud computing, reinforcement learning, game theory, and network economics.

Zhipeng Cai (M'08) received the Ph.D. and M.S. degrees from the Department of Computing Science, University of Alberta, Edmonton, AB, Canada, and the B.S. degree from the Department of Computer Science and Engineering, Beijing Institute of Technology, Beijing, China.

He is currently an Associate Professor with the Department of Computer Science, Georgia State University (GSU), Atlanta, GA, USA. Prior to joining GSU, he was a Research Faculty Member with the School of Electrical and Computer Engineering at the Georgia Institute of Technology, Atlanta. His research areas focus on networking and big data.

Dr. Cai has been an Associate Editor for several journals, including IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He is a recipient of an NSF CAREER Award.

Yi Pan (M'91) received the B.Eng. and M.Eng. degrees in computer engineering from Tsinghua University, Beijing, China, in 1982 and 1984, respectively, and the Ph.D. degree in computer science from the University of Pittsburgh, Pittsburgh, PA, USA, in 1991.

He is currently a Regents' Professor and Chair of Computer Science with Georgia State University, Atlanta, GA, USA. He was an Associate Dean and the Chair of the Biology Department from 2013 to 2017 and the Chair of Computer Science from 2006 to 2013. He is also a visiting Changjiang Chair Professor with Central South University, China. His profile has been featured as a distinguished alumnus in both Tsinghua Alumni Newsletter and University of Pittsburgh CS Alumni Newsletter. He has authored/co-authored more than 200 journal papers with over 80 papers published in various IEEE journals, more than 150 papers in refereed conferences, and also co-authored/co-edited 43 books. His work has been cited more than 9600 times. His research interests include parallel and cloud computing, wireless networks, and bioinformatics.

Dr. Pan has served as an Editor-in-Chief or Editorial Board Member for 15 journals including 7 IEEE Transactions. He is the recipient of many awards including the IEEE Transactions Best Paper Award, several other conference and journal best paper awards, four IBM Faculty Awards, two JSPS Senior Invitation Fellowships, IEEE BIBE Outstanding Achievement Award, NSF Research Opportunity Award, and AFOSR Summer Faculty Research Fellowship. He has organized many international conferences and delivered keynote speeches at more than 60 international conferences around the world.