IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 25, 2019, accepted November 25, 2019, date of publication November 28, 2019,
date of current version December 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2956483

A Two-Layer Channel Access Approach for RF-Energy Harvesting Networks

HANG YU AND KWAN-WU CHIN
School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, Wollongong, NSW 2522, Australia
Corresponding author: Hang Yu (hy472@uowmail.edu.au)

ABSTRACT We consider a Hybrid Access Point (HAP) that charges one or more energy harvesting devices via Radio Frequency (RF). These devices then transmit their data to the HAP. To date, prior works assume devices use Time Division Multiple Access (TDMA) for channel access, and these devices are able to transmit using any amounts of harvested energy. By contrast, we consider Dynamic Framed Slotted Aloha (DFSA) and devices can only transmit if they have sufficient energy. Moreover, nodes are not aware of each other's energy level, meaning the HAP and devices are unaware of the number of devices that is ready to transmit. In addition, we consider different non-linear energy conversion models. To this end, we propose a two-layer approach. At the first layer, the HAP adjusts its transmission power using a Sequential Monte Carlo (SMC) approach, and the frame size according to the Softmax function. At the second layer, devices use another Softmax function to learn the time slot that yields the highest reward for a given frame size. Our results show that throughput is affected by the minimum energy required for each transmission, the temperature of the Softmax function, transmission power used for charging devices, channel gain and network density. Our results indicate that our two-layer learning approach achieves at least 7%, 19%, 40% higher throughput than TDMA, ϵ-greedy and Aloha.

INDEX TERMS Particle filter, wireless power transfer, learning, RF charging.

I. INTRODUCTION

Radio Frequency (RF) energy harvesting techniques have recently emerged as an effective method for supplying energy to low-power devices equipped with a RF-energy harvester [1]. For example, the authors of [2] demonstrated a prototype that harvests RF signals from a television tower located 6.5 km away. Another example is a sensor device with a camera that harvests energy from the transmissions of nearby access points in a Wireless Local Area Network (WLAN) [3]. These prototypes pave the way for RF charging being used to power sensor devices in the upcoming Internet of Things (IoTs) networks [4]. Critically, in these networks, it is important that an operator collects as much sensed data from devices as possible, which can then be analyzed and processed to yield actions that affect the environment [5].

Figure 1 shows an example RF charging network. The Hybrid Access Point (HAP) coordinates energy and data transfers to/from devices. These devices employ a "harvest-then-transmit" protocol [6]; see the frame structure shown in Figure 1. The HAP first broadcasts wireless energy to all devices using the charging slot while devices transmit data packets to the HAP after receiving energy. The number of packets received by the HAP from devices is affected by the following *factors*. First, the amount of energy harvested by each device is a function of the HAP's transmission power

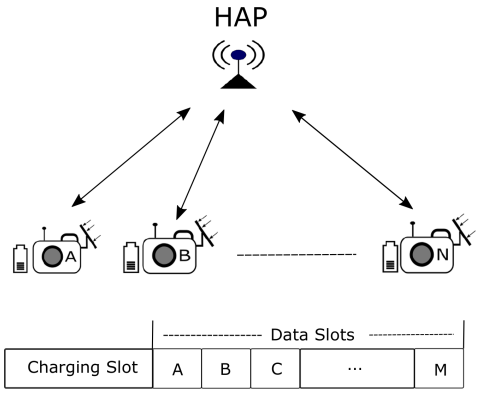


FIGURE 1. An RF charging network with a HAP and RF energy harvesting devices and an example frame structure for charging and data transmissions.

The associate editor coordinating the review of this manuscript and approving it for publication was Martin Gonzalez-Rodriguez.

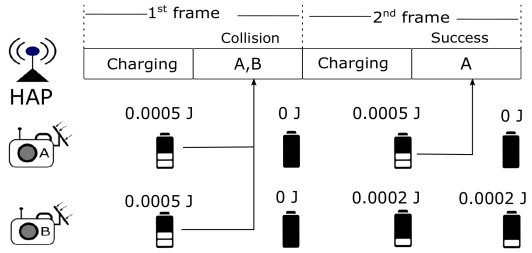


FIGURE 2. An example of channel access in a RF-charging network.

and duration, energy conversion efficiency, and channel gain. The amount of energy harvested by a device affects its transmission power. In our case, it determines whether a device has sufficient energy to transmit a packet; i.e., whether they contend for the channel in the current frame. Lastly, the channel gain between a device and the HAP determines the resulting data rate. Second, the channel access method plays a critical role in determining the number of successes and collisions experienced by devices.

To illustrate the aforementioned factors, consider Figure 2. Assume devices access the channel using framed slotted Aloha. Also, they always have data to transmit. Our goal is to maximize the number of packets that arrives at the HAP over T frames. Assume in the first frame, the HAP transmits at 0.5 Watts for a duration of 0.1 second. To simplify the example, assume that the channel gain for both devices is 0.01 and the energy conversion efficiency is fixed at 100%. Therefore, the received energy for the two devices is 0.0005 Joules. Assume the energy required to transmit one packet is 0.0005 Joules. This means both devices will contend in the upcoming data slot according to their transmission probability. Let us assume the transmission probability for the two devices is 0.5 and 0.8, respectively. Assume both device A and B decide to transmit. In this case, the two devices will experience a collision, meaning the HAP fails to receive a packet. After transmission, both A and B have no energy. For the second frame, assume the HAP transmits at a power of 0.1 Watts for 0.1 second. In this frame, the channel gain of device A and B is assumed to be 0.05 and 0.02, respectively. The amount of energy for device A and B is 0.0005 and 0.0002 Joules, respectively. In this case, only device A has sufficient energy and transmits with probability 0.8. Assume it transmits. As it is the only transmitting device, the HAP receives the packet from device A successfully, assuming no channel error. From this example, we see that our *main aim* is to maximize the number of data slots with a successful transmission and minimize the number of data slots with collisions or is idle; doing so ensures the HAP collects the maximum number of packets.

To achieve our aim, we need to consider the following issues:

- The HAP must determine an appropriate frame size. If the frame size is larger than the number of devices that has sufficient energy, some slots will be idle, which lowers throughput. On the other hand, a small frame size may increase the number of collisions. Therefore, we need to determine a frame size that allows devices to transmit frequently with minimal collisions. The challenge here is that the HAP is not aware of the number of contending devices or those with sufficient energy to transmit.
- The transmission power used by the HAP determines the number of devices that has sufficient energy to contend for a data slot. For example, a high transmission power increases channel contention because more devices will have sufficient energy to transmit. By contrast, if the HAP uses a low transmission power, a frame will have idle slots as not many devices will have sufficient energy to transmit. Consequently, the resulting throughput will be low. The challenge is to determine a transmission power that balances collisions and the number of idle slots.
- A device with sufficient energy needs to select a slot for transmission in each frame. Critically, it must avoid collisions so that it does not waste its harvested energy. The main challenge here is that a device is not aware of the number of contending devices, and a frame may have a small number of slots, causing devices to experience collision in all chosen slots.

Henceforth, to address the aforementioned issues, this paper makes the following contributions:

C1 We propose and study a novel two-layer learning strategy that allows the HAP to determine an appropriate frame size and also the transmission power used for charging. In particular, the HAP uses the Softmax function to adapt the frame size. The HAP selects a frame size according to the utility or probability of each frame size. In addition, the HAP employs a Sequential Monte Carlo (SMC) approach [7] to select the optimal transmission power over a continuous power range. We also propose a novel, distributed, learning Medium Access Control (MAC) protocol for use by devices. In particular, devices employ the Softmax function to identify the transmission slot that yields the highest reward.

C2 To the best of our knowledge, as explained in Section II, our work is novel. Specifically, past works on Wireless Powered Communication Networks (WPCNs) assume Time Division Multiple Access (TDMA) and they do not consider transmission probability or sizing the number of transmission slots. Moreover, they do not consider adjusting the transmission power of the HAP in order to reduce collisions. Also, devices are able to transmit using any amounts of harvested energy. However, in practice, devices are only able to transmit when they have sufficient energy to transmit a packet. Hence, in this paper, our devices can only transmit when they have the required energy to transmit one packet. Critically, past works assume the HAP is aware of the channel gain or current energy level of devices. We do not make such an assumption, meaning our HAP does not know exactly how many devices have sufficient energy to transmit in

a given frame. Lastly, we consider two non-linear energy conversion models; namely, quadratic and a practical model based on [8]. This is significant, as most past works assume a linear energy harvesting model, which has been shown in [9] to be inaccurate.

C3 From our experiments, we find that our two-layer learning approach achieves at least 7%, 19%, 40% higher throughput than TDMA, ϵ-greedy and Aloha. Our results show that TDMA achieves 96% and 75% throughput obtained by our two-layer learning approach at high and low traffic load respectively. When we increase the number of devices from ten to 50, the converge time of our proposed approach increased by five times, while it achieve higher throughput than TDMA for all network densities. Furthermore, the results also reveal that a severe channel reduces the harvested energy for the practical non-linear model as compared to the linear model, i.e., the gap of the harvested energy between practical and linear model decreases by 20% when the channel changes from mild to severe.

The remainder of the paper is structured as follows. In Section II we compare and contrast our works with respect to the state-of-the-art. Then we present our network model in Section III. Next, we formalize our problem in Section IV. After that, in Section V, we outline our two-layer learning approach. Our results are presented in Section VI followed by our conclusion in Section VII.

II. RELATED WORKS

Our work overlaps with those that aim to maximize the sum-rate of a WPCN; see [10] and references therein. However, in general, these works assume devices use TDMA, have no learning capability, and do not aim to vary the HAP's transmission power in order to reduce collisions nor aim to change the frame size or number of data slots in a frame. Critically, these works assume devices are already allocated a transmission slot and do not consider *random* channel access. For example, in [6] and [11], the work is focused on sizing the charging slot of the HAP, and data transmission slot of each device. The same problem is then revisited for a HAP with a full-duplex radio [12] and Multiple-Input Multiple-Output (MIMO) capability [13]. In [14], the problem is to jointly optimize downlink power allocation and uplink energy utilization at devices. The authors of [15] and [16] consider Simultaneous Wireless Information and Power Transfer (SWIPT), and jointly optimize transmit beamforming, time allocation and power splitting strategies of devices. In a recent work [17], the authors consider slotted Aloha for WPCNs. However, their devices have no learning capability and their HAP does not aim to control its transmission power to improve channel access performance. We note that in Radio Frequency IDentification (RFID) systems, many works have considered framed slotted Aloha based tag reading protocols; see [18] and references therein. However, the RFID reader transmits at a fixed power and devices/tags have no learning capability.

Channel access is also an important problem in energy harvesting systems [19], where devices rely on ambient energy sources. Example works include [20] and [21], where the transmission probability of devices is set according to their available energy. In [22], Iannello *et al.* propose an energy group-based dynamic framed Aloha protocol. Devices in the same group contend for the channel simultaneously. Therefore, the problem is to optimize the number of data slots within a group. In [23], a central node adjusts the transmission probability of devices by estimating the number of transmitting devices in previous slots. In [24], a device reserves a slot that it has transmitted successfully. In both of the previous works, the frame size is fixed and devices rely on ambient energy.

To date, many works have considered using reinforcement learning to address problems in communication systems; see [25] and references therein. These problems include determining the optimal policy given varying energy harvesting rates or Channel State Information (CSI). Example works such as [7] and [26] consider a point-to-point channel where the transmitter harvests ambient energy. The work in [27] studies a joint access control and battery prediction problem in a network with a base station and multiple energy harvesting devices. It aims to maximize the uplink transmission rate of devices, and also to minimize energy outage. For access control, the base station allocates a channel to devices. Different from these works, our aim is to *jointly* optimize the frame size and transmission power used by the HAP for charging, and also the slot selection policy of charged devices.

In summary, unlike past works in WPCNs, we consider random channel access and adopt a learning approach at *both* the HAP and devices, whereas works such as [6], [11]–[16] assume devices are already pre-assigned a transmission slot and seek to optimize charging and uplink transmission slot. These works also assume perfect CSI, while our work does not make this assumption. In addition, they assume devices are able to transmit using any amounts of harvested energy, whereas our devices can only transmit when they have sufficient energy to transmit a packet. In contrast to [7], [20]–[24] and [26] that assume devices harvest energy from ambient sources, the energy on our devices and channel access are controlled by a HAP. Specifically, our HAP is able to control the number of transmitting devices through transmit power control, and also by adjusting the frame size used by devices to select a transmission slot. Critically, these tasks are executed without CSI or energy level information at devices.

III. SYSTEM MODEL

We consider a RF-harvesting network, see Figure 1, with one HAP and a set N of RF-energy harvesting devices. The HAP is responsible for transmitting energy to the $|N|$ devices and collecting data from these devices. The energy and data is transferred over the same frequency band.

The RF-harvesting network operates over frames; each indexed by k. Each frame F^k has a fixed charging slot with

duration τ_C and M^k data slots; note $M^k \leq |N|$. The HAP is responsible for determining the transmission power used in each frame k and also the number of slots M^k, aka frame size. Slot m in frame k is denoted as s_m^k. In the charging time of each frame, the HAP broadcasts energy to devices. Specifically, it transmits at power $P^k = \Delta^k P_{max}$, where P_{max} is the maximum transmission power, and $\Delta^k \in [0, 1]$ is a proportion of its maximum power that it will use in frame k. On the other hand, the data slots are used by devices to transmit to the HAP. Each device has a transmission rate of r bps. Each data slot is sufficient to transmit one packet of size L bits; i.e., its duration is $\tau_D = L/r$ seconds. The energy cost to transmit a bit is ϵ J/bit. The minimum energy required to transmit a packet is $E_{min} = L\epsilon$ (Joules). We assume devices always have data to transmit.

We consider a block fading channel where the channel gain is fixed in each frame but varies from frame to frame. Moreover, the HAP is not aware of the channel gain to each device. This is reasonable as devices need to first harvest energy in order to participate in channel estimation. Also, it is impractical to collect CSI from a high number of devices. We write the channel gain to device i as g_{oi}^k, and from device to the AP as g_{io}^k, where 'o' denotes the HAP. We consider a log distance path loss model, thus, the channel gain is given by

$$PL_{d_0 \to d_i}(dB) = PL(d_0) + 10n log_{10}(\frac{d_i}{d_0}) + X$$
$$g^k = \frac{1}{10^{\frac{PL_{d_0 \to d_i}}{10}}} \quad (1)$$

where d_i is the distance between device i and the HAP; $PL(d_0)$ is the path loss in dB at a reference distance $d_0 \leq d_i$; n is the path loss exponent; and X denotes the shadowing effect, which is defined as a zero-mean Gaussian distributed random variable (in dB) with standard deviation μ.

The exact amount of energy harvested by device i in frame k is denoted by E_i^k. It is bounded by the battery capacity B_{max}. The battery of each device i evolves as per $B_i^{k+1} = B_i^k + E_i^k - \delta_i^k E_{min} \leq B_{max}$, where $\delta_i^k \in \{0, 1\}$ takes on a value of one whenever device i transmits in frame k. Specifically, a device only transmits if $(B_i^k + E_i^k) \geq E_{min}$; recall that E_{min} is the energy required to transmit one packet of size L at a rate of r bps. In addition, we consider the following three energy conversion efficiency models:

- *Linear* [28]. Formally, $E_i^k = \eta_1 P^k g_{oi}^k \tau_C$, where $0 \leq \eta_1 \leq 1$ is the energy conversion efficiency.
- *Quadratic* [29]. Formally, $E_i^k = (\alpha_1 (P^k g_{oi}^k)^2 + \alpha_2 P^k g_{oi}^k + \alpha_3)\tau_C$, where $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{R}$ are the parameters of the model.
- *Practical* [8]. Formally, $E_i^k = \eta_2 P^k g_{oi}^k \tau_C$, where the value of η_2 is obtained from the datasheet of the P2110B RF power harvester from Powercast [8].

These models are illustrated in Figure 3, where Figure 3a and Figure 3b show the harvested power and conversion efficiency respectively when the input power increases from 0 to 10 mW.

TABLE 1. A summary of notations.

Notations	Explanation
$\|N\|$	The number of energy harvesting devices
F^k	Frame k
τ_C	The duration of a charging slot
τ_D	The duration of a data slot
M^k	The number of data slot in frame k
s_m^k	Slot m in frame k
P^k	Transmission power of the HAP at frame k
Δ^k	A proportion of maximum power at frame k
P_{max}	Maximum transmission power
r	Uplink data rate
L	Packet length
E_{min}	Energy cost for a packet transmission
g_{oi}^k	Channel gain from the HAP to device i at frame k
g_{io}^k	Channel gain from device i to the HAP at frame k
d_i	The distance between device i and the HAP
$PL_{d_0 \to d_i}$	Path loss at distance d_i
$PL(d_0)$	Path loss at a reference distance d_0
n	Path loss exponent
X	Gaussian distributed random variable with deviation μ and zero mean
E_i^k	Amount of energy harvested by device i at frame k
B_i^k	Battery level of device i in frame k
B_{max}	Battery capacity
η_1,η_2	Energy conversion efficiency of linear and practical model
α_1, α_2, α_3	Parameters of quadratic energy conversion model

We assume channel access is carried out using Dynamic Framed Slotted Aloha (DFSA). Recall that each frame k is composed of a *variable* M^k number of data slots. If a device has sufficient energy, it selects a slot to transmit in frame k according to the method in Section V. If multiple devices select the same slot, then there is a collision and no data arrives at the HAP. Accordingly, a data slot has the following three states: I) *Success*, if only one device transmits, II) *Idle*, if no devices transmit, III) *Collision*, if more than one device transmits. Table 1 summarizes all key notations.

IV. THE PROBLEM

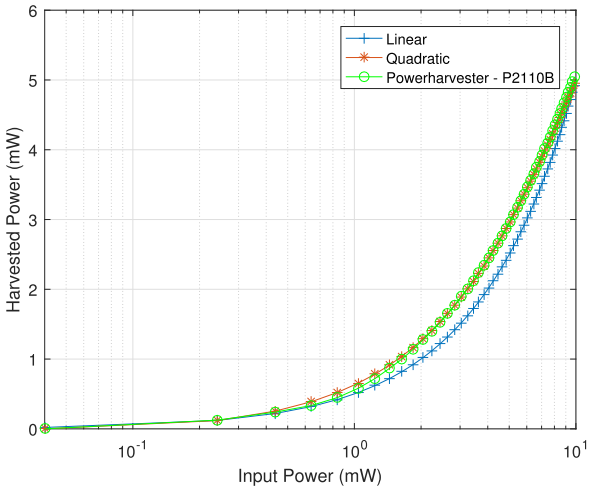
Our problem can be divided into two parts: (i) the HAP aims to determine the best transmission power P^k for use in frame k and also the number of data slots M^k, and (ii) at devices with energy, they need to determine a transmission slot in frame k. Next, we will first formalize the problem at the HAP and devices.

Let $\mathcal{S}(s_m^k) \in \{Success, Collision, Idle\}$ return the state of slot s_m^k. Define $R(s_m^k)$ as the 'reward' of slot s_m^k. Formally,

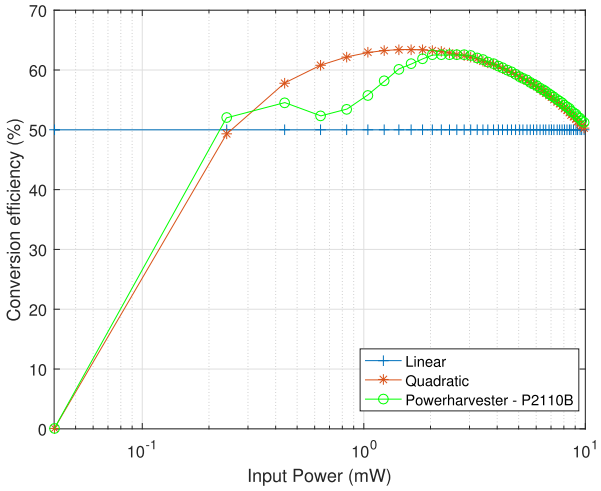
$$R(s_m^k) = \begin{cases} 0 & \mathcal{S}(s_m^k) = Collision \vee Idle \\ 1 & \mathcal{S}(s_m^k) = Success \end{cases} \quad (2)$$

The throughput of frame k is thus defined as

$$T^k = \frac{1}{\tau_C + M^k \tau_D} \sum_{m=1}^{M^k} R^+(s_m^k) \times L \quad (3)$$



(a) Harvested power versus input power.



(b) Conversion efficiency versus input power.

FIGURE 3. Comparison of the three conversion models.

The throughput at the HAP is defined as

$$\bar{T} \triangleq \lim_{k\to\infty} = \frac{1}{k}\mathbb{E}[T^k] \tag{4}$$

The expectation is with respect to the channel gains to devices.

The objective of the HAP is defined formally as

$$\begin{aligned} &\max_{\Delta^k, M^k} \ \bar{T} \\ &\text{s.t. } \Delta^k \in [0,1] \\ &\quad M^k \in \{0,1,2,\ldots,|N|\} \end{aligned} \tag{5}$$

In the foregone problem, the chief challenge is that the HAP has imperfect channel information. A high transmission power may result in more devices with E_{min} worth of energy. Consequently, multiple devices are likely to transmit and they may experience collisions if the frame size is set incorrectly. On the other hand, a lower transmission power or longer than necessary frame size may cause idle slots. Both scenarios reduce throughput.

Devices aim to select a slot that yields the most successes. Specifically, for a device i, given a frame with M^k slots, it wishes to select a slot in $\mathcal{A} = \{1,\ldots,M^k\}$ that yields the highest average number of successes. Let $p_i(s_m^k)$ denote the probability of using slot s_m^k. Define $\pi_i = [p_i(s_m^k)]_{m\in\mathcal{A}}$ as the policy used by device i for a given frame size. Here, a policy defines the strategy used by device i when selecting a slot for a given frame size. For example, if a frame has two slots and $\pi_i = [0.5, 0.5]$, then device i selects to transmit in both slots uniformly. We emphasize that devices maintain a different policy for each frame size. Let Ω be the collection of policies that satisfy $\sum_{m\in\mathcal{A}} p_i(s_m^k) = 1$; as examples, if there are two slots then both $\pi_i = [0.9, 0.1]$ and $\pi_i = [0.5, 0.5]$ belong to Ω. The average *utility* obtained by device i for each frame k when using policy π_i is thus,

$$\bar{U}^k(\pi_i) = \sum_{m\in\mathcal{A}} p_i(s_m^k) R_i(s_m^k) \tag{6}$$

where $R_i(s_m^k)$ is the reward obtained by device i.

Each device i aims to solve the following problem,

$$\begin{aligned} &\max_{\pi_i\in\Omega} \lim_{k\to\infty} = \frac{1}{k}\mathbb{E}[U^k(\pi_i)] \\ &\text{s.t. } (B_i^k + E_i^k) \geq E_{min} \end{aligned} \tag{7}$$

The challenges for devices are that they do not know how many devices have sufficient energy to transmit, i.e., the number of contending devices in each frame, as well as the strategy π_j, where $i \neq j$, used by other devices.

V. A TWO-LAYER LEARNING APPROACH

We now present an online learning approach where both the HAP and devices use the outcome of their transmissions as feedback. Our two-layer approach is depicted in Figure 4. At the first layer, the HAP is responsible for adjusting the frame size and its transmission/charging power. The second layer is located at devices, where they select a transmission slot independently. Briefly, the system operates as follows. The HAP first charges all devices at the beginning of each frame. Devices then harvest energy according to their respective received power. The HAP also informs devices the current frame size M^k. Each device i then selects a slot according to its policy and attempts transmission. If it is successful, it obtains a reward $R_i(s_m^k)$, which it then uses to adjust its slot selection policy for the given frame size. At the end of each frame, the HAP calculates a reward $R(s_m^k)$ as per the state of slots in the frame, and uses the reward to adjust its next transmission power P^{k+1} and frame size M^{k+1}. As it will become clear later, our approach is adaptive to network dynamics and it does not require devices to transmit extra messages or process large amounts of data. Both the HAP and devices only need to observe the status of their respective transmissions. The HAP then uses this information to maintain a Probability Mass Function (PMF) of its transmit power and frame sizes, whilst devices maintain a PMF over the slots of each frame size.

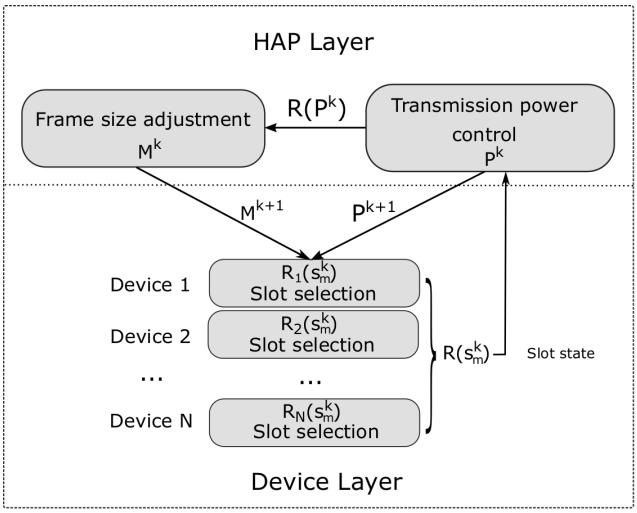


FIGURE 4. Framework of our two-layer learning approach.

A. DEVICE LAYER

The main idea is for each device to evaluate the utility or reward of each time slot in a given frame. To do this, devices calculate the probability of each slot after a transmission, and this probability corresponds to the reward obtained when it transmits in that slot. The probability of each slot forms a PMF of the time slots in each frame. Devices can then take an action or transmit in a given slot according to the computed PMF. We first introduce some notations. Recall that device i selects a slot s_m^k in $\mathcal{A}_i = \{1, \ldots, M^k\}$. Therefore, we have the following so called joint action set for all $|N|$ devices $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \ldots \times \mathcal{A}_N$. The corresponding joint reward set is $\mathcal{R} = \mathcal{R}_1 \times \mathcal{R}_2 \times \ldots \times \mathcal{R}_N$. In particular, the set $\mathcal{R}_i$ contains all reward $R_i(s_m^k)$ over the action space $s_m^k \in \mathcal{A}_i$. The reward $R_i(s_m^k)$ is defined as $C_i(s_m^k) = log_2(1 + \frac{E_{min} g_{io}^k}{\tau_D \sigma^2})$, where σ^2 is the noise power. This positive reward is only obtained by device i if $S(s_m^k) = \text{`}Success\text{'}$ for the selected slot s_m^k. Devices get a negative reward $-C_i(s_m^k)$ when they experience a collision. Let $\mathcal{I}(s_m^k)$ return the set of transmitting devices in slot s_m^k. Device i experiences no collision if the condition $\mathcal{I}(s_m^k) - \{i\} = \emptyset$ is true. Formally, we have,

$$R_i(s_m^k) = \begin{cases} -C_i(s_m^k) & \mathcal{I}(s_m^k) - \{i\} \neq \emptyset \\ C_i(s_m^k) & \text{Otherwise} \end{cases} \tag{8}$$

For a given device i, define its PMF over the slots in frame M^k as $\Gamma_i(M^k) : \mathcal{A}_i \rightarrow [0, 1]$. Let Γ_i be the space of probability distributions over the action space $\mathcal{A}_i$ of all frame sizes. The *aim* of device i is to find the PMF $\Gamma_i(M^k)^*$ that achieves the maximum reward. To this end, we employ the Softmax function to construct the PMF for a frame of size M^k. We note that the Softmax function is widely used in neural networks and reinforcement learning to represent action probabilities; interested readers are referred to [30] and [31] for more information. In our case, a device uses the Softmax function to select the slot that yields the highest reward thus far, meaning for such a slot or action s_m^k, its density or probability $P_i(s_m^k)$ will be high. Formally, we have,

$$P_i(s_m^k) = \frac{e^{R_i(s_m^k)/\tau}}{\sum_{s_m'^k \in \mathcal{A}_i} e^{R_i(s_m'^k)/\tau}} \tag{9}$$

where τ is the so called temperature that is used to control the trade-off between exploration and exploitation. Note that the probability of the action with the highest reward is much larger than other actions when we use a low τ value.

We now describe the process by which devices learn the best action. Define a learning phase ζ as consisting of F frames. In each phase ζ, each device accumulates the reward of its selected slots. At the end of a phase, devices then update their PMF $\Gamma_i(M^k)$. Algorithm 1 shows the steps for device i over one learning phase given the frame size M^k.

Initially, device i initializes its PMF $\Gamma_i(M^k)$ to the uniform distribution $\mathcal{U}$. Hence, there is equal chance of selecting any slot s_m^k in frame k. At the beginning of each frame, device i checks whether it has sufficient energy, i.e., $B_i^k \geq E_{min}$. If it does, it then selects an action or slot $m \in M^k$ according to its PMF $\Gamma_i(M^k)$. If a device has insufficient energy, then it remains idle in the current frame. At the time of their selected slot m, a device transmits; see line 4-6. If its transmission is successful, device i will receive an Acknowledgement (ACK). It then calculates its reward as per Equ. (8). Devices receive a negative reward when it experiences a collision; i.e., there is no ACK. At the end of learning phase ζ, device i calculates the average reward $\bar{R}_i(s_m^k)$ for each slot, which is then used to calculate the PMF $\Gamma_i(M^{k+1})$ as per Equ. (9), see line 13. Devices then select transmission slots according to the updated PMF in the next learning phase.

Algorithm 1 Pseudocode for Device i

```
Input: M^k
Output: Γ_i(M^{k+1})
1  Γ_i(M^k) ← U(.)
2  for each frame k in ζ do
3      if B_i^k ≥ E_min then
4          Select a slot m ∈ M^k according to Γ_i
5          WaitSlot(m)
6          Transmit()
7          Calculate R_i(s_m^k) as per Equ.(8)
8      else
9          Remain idle
10     end
11 end
12 Calculate R̄_i(s_m^k)
13 Use R̄_i(s_m^k) to update PMF Γ_i(M^{k+1}) as per Equ.(9)
14 return Γ_i(M^{k+1})
```

B. HAP LAYER

The HAP needs to learn the best transmission power for a given frame size. In addition, it also needs to determine the frame size that yields the highest throughput. We first show

how the HAP uses the SMC approach in [32] to determine the best transmission power for a given frame size. We note that SMC is a reinforcement learning approach that allows an agent or HAP to operate over a continuous action space; in our case, the transmit power range of the HAP. After discussing SMC, we present our method to adjust the frame size.

1) TRANSMISSION POWER CONTROL

For a given frame size M^k, we employ the SMC approach to learn the best transmission power that yields the highest sum rate. This is an actor-critic approach where an actor has a policy for selecting an action. This policy is then shaped by the critic, which provides the value of each action. In our case, the actor is the HAP where it needs to learn a policy that allows it to select the best action or transmission power. Let the policy of the actor or HAP be represented as a density distribution $\pi(M^k)$, which returns the probability of each action a for a given frame size M^k. Initially, $\pi(M^k)$ is set to the uniform distribution. This means the actor is equal likely to choose any actions. In particular, it will draw S samples from the distribution $\pi(M^k)$. Mathematically, we have

$$\hat{\mathcal{A}} = \{a_1, a_2, \ldots, a_S\}, \quad a_i \sim \pi(M^k) \tag{10}$$

Each sampled action $a_i \in \hat{\mathcal{A}}$ has an importance weight $w_i \in \mathcal{W}$. Mathematically, these S samples approximate the distribution $\pi(M^k)$ as follows,

$$\pi(M^k) \simeq \sum_{i=1}^{S} w_i \cdot \delta(a - a_i) \tag{11}$$

where δ is the Dirac delta measure. Initially, all samples have equal weight; i.e., $w_i = \frac{1}{S}$, where $i = 1, 2, \ldots, S$.

The next task is to determine the weight w_i of samples so that $\pi(M^k)$ better approximates the actual density function containing the most likely transmission power that yields the highest reward. To do this, the actor collects a reward $R(a_i)$ for each action and relates that to the weight of each action. This is achieved by applying the *Boltzmann* function as follows,

$$w_i = \frac{e^{\frac{R(a_i)}{\tau}}}{\sum_{j=1}^{S} e^{\frac{R(a_j)}{\tau}}} \tag{12}$$

where τ is the temperature that influences the HAP's exploration degree. The actor then draws S new samples from the updated density function $\pi(M^k)$. Over time, the density $\pi(M^k)$ converges onto actions that yield the highest reward.

As noted in [32], some actions may have a very low weight/reward/throughput. Consequently, it is a waste of time to use such actions. This problem is called *weight degeneracy*. To determine whether weight degeneracy has occurred, the HAP calculates the effective action size $\widehat{N_{eff}}$,

$$\widehat{N_{eff}} = \frac{1}{\sum_{w_i \in \mathcal{W}} w_i^2} \tag{13}$$

A low $\widehat{N_{eff}}$ value indicates high degeneracy. When this happens, the actor or HAP resamples actions as per Equ. (11) whenever the ratio between $\widehat{N_{eff}}$ and the number of actions S falls below a given threshold λ. This means actions with a high weight will be replicated many times, meaning the S samples will concentrate around high reward actions.

Algorithm 2 Pseudocode for Determining the Transmission Power Policy of the HAP

Input: M^k
Output: $\pi(M^k)$
Initialize $\pi(M^k) \leftarrow \mathcal{U}(.)$
$\hat{\mathcal{A}} = \{a_1, a_2, \ldots, a_S\}, a_i \sim \pi(M^k)$
for *each learning phase* $\kappa \in K$ **do**
  for *each transmission power* $P_i^k \in \hat{\mathcal{A}}$ **do**
    while *devices have not converged* **do**
      InformDevices(.)
      $R(P_i^k) = \text{GetReward}(P_i^k)$
      if *devices converge* **then**
        Use the next transmission power P_{i+1}^k
      end
    end
  end
  Use $R(P_i^k)$ to calculate the density function $\pi(M^k)$ as per Equ. (12)
  if $\frac{\widehat{N_{eff}}}{S} \leq \lambda$ **then**
    $\hat{\mathcal{A}} = \text{Resample}(\pi(M^k))$
    Reset probability of samples in $\hat{\mathcal{A}}$ to $\frac{1}{S}$
  end
  for *all actions* $a \in \hat{\mathcal{A}}$ **do**
    Construct the Uniform kernel $K_i(a)$ as per Equ. (14)
    Uniformly sample an action a_i from $K_i(a)$
    $\hat{\mathcal{A}} \cup a_i$
  end
end
return $\pi(M^k)$

Another problem to address is *sample impoverishment*. This problem occurs when many of the S actions are the same, meaning there is insufficient diversity in the samples or actions. In the worst case, all S samples are for the same action. Apart from that, the action space is continuous, meaning the S discrete samples may not contain the optimal action. To resolve these problems, we employ a Uniform kernel around each resampled action. Formally, we have

$$K_i(\hat{\mathcal{A}}) = U[\frac{(a_i - a_{i-1})}{2}; \frac{(a_{i+1} - a_i)}{2}]. \tag{14}$$

For samples at the boundary, i.e., a_1 and a_S, the corresponding kernel is set to $K_1(a) = U[(a_2 - a_1); \frac{(a_2 - a_1)}{2}]$ and $K_S(a) = U[\frac{(a_S - a_{S-1})}{2}; (a_S - a_{S-1})]$, respectively.

We are now ready to present how the HAP learns the best transmission power or policy $\pi(M^k)$. Algorithm 2 illustrates the steps carried out by the HAP.

Define κ and K as a learning phase of the HAP and the total number of episodes, respectively. For each learning phase κ, the HAP collects a reward of all S samples, which is used to update the density function $\pi(M^k)$. The HAP then samples a new set of actions used in the next learning phase $\kappa + 1$.

The density function is initialized to the uniform distribution, see line 1. The HAP first draws S samples from $[0, P_{max}]$ according to the density $\pi(M^k)$. We denote each transmission power as P_i^k, where $i \in 1, 2, \ldots, S$. For each learning phase κ, the HAP uses each of the S transmission power in turns. The HAP first informs devices to calculate their PMF. It then obtains a reward $R(P_i^k)$ for transmission power P_i^k by calling the function *GetReward()*. The reward $R(P_i^k)$ is defined as $\sum_{m=1}^{M^k} R^+(s_m^k)$, where $R(s_m^k)$ is calculated as per Equ. (2). Specifically, a transmission power P_i^k is used for multiple frames until the PMF of devices have converged. The HAP then switches to the next transmission power P_{i+1}^k, see line 4 - 12. After obtaining the reward of all S transmission powers, the HAP updates the density function $\pi(M^k)$ as per Equ. (12), see line 13. After that, the HAP measures the effective action size $\widehat{N_{eff}}$ as per Equ. (13), see line 14. If $\frac{\widehat{N_{eff}}}{S} \leq \lambda$, the HAP resamples actions as per Equ. (11) based on the updated density function $\pi(M^k)$. Then it sets the weight w_i of each resampled action to $\frac{1}{S}$, see line 15 - 16. Next, the HAP constructs an Uniform kernel $K_i(a)$ for each sample of a_i as per Equ. (14), and uniformly selects an action from each kernel and returns the action in the action set $\hat{\mathcal{A}}$, see line 19 - 21. The HAP then uses a new set of actions for the next learning phase.

2) FRAME SIZE ADJUSTMENT

As mentioned, the HAP also needs to determine the best frame size that yields the highest throughput. Specifically, the HAP wishes to select a frame size M^k or action $\mathcal{F} \in \{1, \ldots, |N|\}$ that yields the maximum throughput. Let $R(M^k)$ denote the weighted reward obtained for frame size M^k, which is calculated as follows,

$$R(M^k) = \sum_{i=1}^{M^k} w_i R(P_i^k) \tag{15}$$

Let the PMF over frame size M^k be defined as $\phi(M^k) : \mathbb{N}_+ \rightarrow [0, 1]$. We also use the Softmax function to calculate the probability $P(M^k)$ of frame size M^k. Formally, we have

$$P(M^k) = \frac{e^{R(M^k)/\tau}}{\sum_{j \in \mathcal{F}} e^{R(j)/\tau}} \tag{16}$$

where τ is the temperature that characterizes the frame size, which degrades over time.

We now describe how the HAP selects the optimal frame size, see Algorithm 3. Define a learning phase ξ as consisting of F frames. In each phase ξ, the HAP accumulates the reward of the selected frame size. At the end of each phase, the HAP updates the PMF $\phi(M^k)$. Initially, the PMF $\phi(M^k)$ is set to the uniform distribution, see line 1. For each frame k in the learning phase ξ, the HAP first selects a frame size M^k according to the PMF $\phi(M^k)$. It then calculates the reward $R(M^k)$ using $R(P_i^k)$ as per Equ. (15), see line 4. At the end of learning phase ξ, the HAP calculates the average reward $\bar{R}(M^k)$ of each frame size M^k. It then updates the PMF $\phi(M^k)$, see line 6 - 8. The HAP then selects a new frame size M^{k+1} based on the updated PMF $\phi(M^{k+1})$, which is then used by devices to select slots.

Algorithm 3 Pseudocode Used by the HAP to Adjust the Frame Size

Input: $R(P_i^k)$
Output: M^{k+1}
1 Initialize $\phi(M^k) \leftarrow \mathcal{U}()$
2 **for** *each frame* $k \in \xi$ **do**
3 Select a frame size M^k as per $\phi(M^k)$
4 Use $R(P_i^k)$ to calculate the reward $R(M^k)$ as per Equ. (15)
5 **end**
6 Use $R(M^k)$ to calculate the average reward $\bar{R}(M^k)$
7 Use $\bar{R}(M^k)$ to calculate PMF $\phi(M^{k+1})$
8 Select a new frame size $M^{k+1} \in \mathcal{F}$ according to $\phi(M^{k+1})$
9 **return** M^{k+1}

VI. EVALUATION

To evaluate our two layer approach, we conduct all experiments in Matlab running on a computer with an Intel Core i7 CPU@3.4GHz with 8 GB RAM. Devices are randomly deployed at a distance between 1 and 5 meter from the HAP, which ensures they meet the required receiver sensitivity to harvest RF energy. Specifically, devices are equipped with a P2110B power harvester [8] receiver that has a sensitivity of $Pr_{min} = -12$ dBm. The HAP parameters correspond to the Powercaster Transmitter TX91501 [8], which transmits with maximum power output $P_{max} = 3$W EIRP in the $f = 915$ MHz band. The HAP and devices are equipped with an antenna that has a gain of $G_t = 1$ and $G_r = 6.1$ dBi, respectively. Except for Section VI-E, all our experiments involve ten devices; this ensures multiple devices are contending for the channel. Further, devices are equipped with a battery, which is initially empty and has a capacity of 5 mJ. The linear energy conversion efficiency is set to 50%. The quadratic model uses $\alpha_1 = -0.0189$, $\alpha_2 = 0.6942$, $\alpha_3 = -0.0472$. The practical model uses the parameters from the datasheet of the P2110B RF receiver [8]. We set the path loss exponent to 2.5. The channel noise power is set to 1×10^{-12} W. Devices transmit 1500 Bytes packets. The duration of the charging slot τ_C and data slot τ_D is set to 0.5s and 0.1s, respectively. The minimum energy consumed for a packet E_{min} is 0.3 mJ. The three temperature τ values are decrease from 2 to 0.8 as the learning progresses. Table 2 summarizes all parameter settings.

TABLE 2. A summary of parameter settings.

Parameters	Value	Parameters	Value
$\|N\|$	10	d_i	1 to 5 m
P_{max}	3 W	f	915 MHz
G_t, G_r	1, 6.1 dBi	Pr_{min}	-12 dBm
B_{max}	5 mJ	η_1	50%
α_1	-0.0189	α_2	0.6942
α_3	-0.0472	n	2.5
σ^2	1×10^{-12} W	L	1500 Bytes
τ_C	0.5s	τ_D	0.1s
E_{min}	0.3 mJ	τ	Decreases from 2 to 0.8

We compare our solution against four other possible approaches. They include,

- *Offline.* All nodes have non-causal information of the energy harvesting processes. This means the HAP knows the battery level of devices. Using this information, the HAP sets the frame size to be equal to the number of devices with sufficient energy in order to eliminate idle slots. A device also knows the slot chosen by other transmitting devices to avoid collision.
- *TDMA.* Each device is assigned with a unique slot. The frame size is always set to the number of devices. In TDMA, a data slot has a successful transmission if the assigned device has sufficient energy; otherwise, the data slot is idle.
- *Aloha.* Devices attempt transmission in each slot with equal probability.
- *ϵ-greedy* [7]: Each agent selects the action with the highest reward with probability $1-\epsilon$; otherwise it selects other actions uniformly.

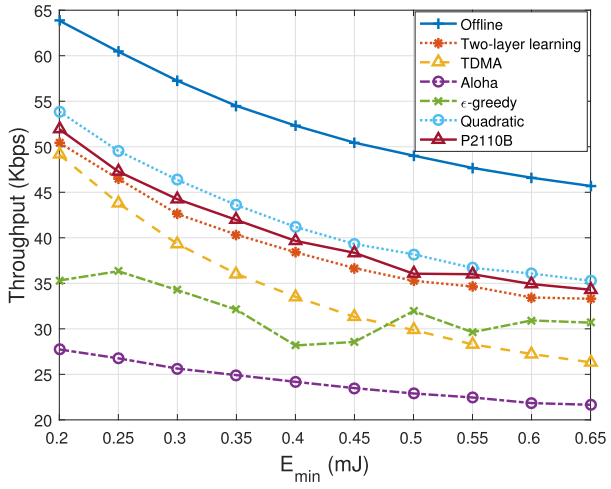
We record two metrics: (i) *Throughput*, which is calculated as per Equ. (3), and (ii) *Slot State*, where we record the average number of slots that have collisions, are idle, and contain one transmitting device; i.e., a success. We also compare the performance of three energy conversion efficiency models; see Section III. We additionally collect the average harvested energy E_{ave} in order to compare the performance of the three energy conversion models. For all the aforementioned scenarios, we study the following parameters: minimum energy consumed to transmit a packet E_{min}, temperature of the Softmax function τ, network density $|N|$ and deviation μ of the Gaussian distribution, which represents the severity of the channel. We also study the case where the HAP transmits energy at a fixed power.

A. MINIMUM ENERGY

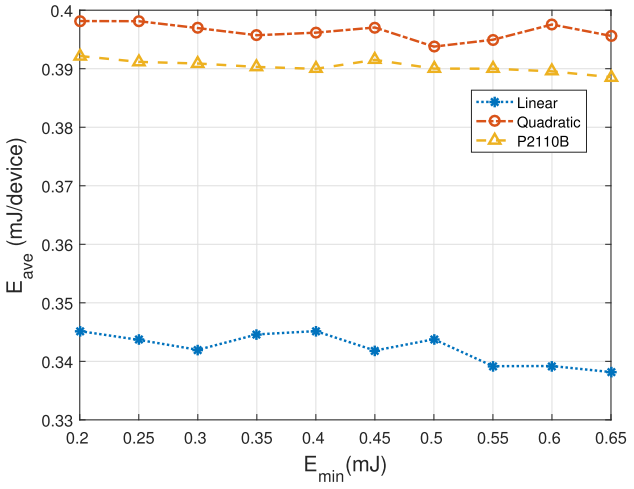
In this experiment, we study the minimum energy E_{min} required for packet transmissions. We increase the value of E_{min} from 0.2 to 0.65 mJ. In Figure 5a, our two-layer learning approach uses the linear energy harvesting model. The quadratic and P2110B model are also illustrated in Figure 5b. We see that our two-layer learning approach achieves higher throughput than ϵ-greedy, TDMA and Aloha. Specifically, TDMA only achieve 75% of the throughput attained by our approach when $E_{min} = 0.65$ mJ. This is because our approach selects smaller frame sizes than TDMA, which results in fewer idle slots. For example, Figure 5c shows the fraction of slots that have collisions, are idle and one transmission or success. We see that our proposed approach uses an average of 4.3 slots when E_{min} is set to 0.65 mJ, while the frame size of TDMA is fixed to ten. As a result, TDMA has an average of 6.8 idle slots, which is more than three times higher than our proposed approach. This explains why TDMA has a lower throughput as compared to our two-layer approach. In addition, the throughput of our approach is two times higher than that of Aloha when $E_{min} = 0.2$ mJ. This is because when devices use our approach, they experience low collisions and high number of successes; see Figure 5c. Also, as compared to the ϵ-greedy approach, our approach achieves higher throughput because of lower frame lengths and idle slots. Lastly, the Offline policy shows the maximum throughput achieved by our system setup. Referring to Figure 5a, the Offline achieves a throughput of 65 Kbps at $E_{min} = 0.2$ mJ, which is 30% higher than the proposed approach. This is because all transmission attempts are successful. For example, see Figure 5c, Offline obtains 7.1 successes per frame without collisions or idle slots when E_{min} is 0.2 mJ. However, our proposed approach only achieves 5.3 successes, which results in a lower throughput; this is expected as the Offline approach is aware of the exact number of contending devices.

We can see that the throughput reduces as the minimum required energy E_{min} increases. For example, in Figure 5a, the throughput of our two-layer learning approach decreases by 32% from 50 to 34 Kbps. TDMA decreases from 49 to 26 Kbps. This is because there are fewer devices with sufficient energy that can attempt a transmission. For example, in Figure 5c, we see that for TDMA, its average number of successes decreases from 5.4 to 1.6 per frame when E_{min} increases from 0.2 to 0.65mJ.

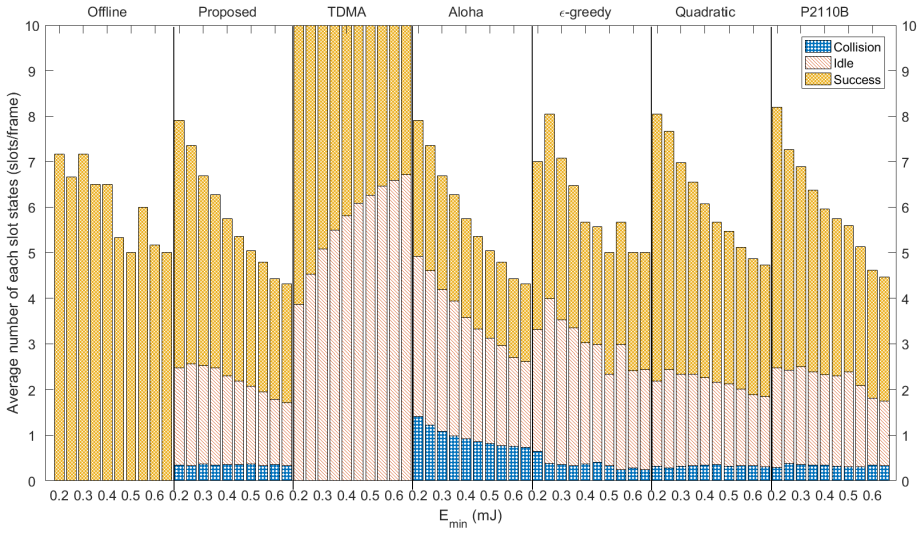
Figure 5a also illustrates the throughput attained for the three energy conversion models. We observe that the quadratic model achieves the highest performance followed by the P2110B receiver and linear model. This is because the quadratic model allows devices to harvest the highest amount of energy, see Figure 5b. For example, when devices use 0.2 mJ to send a packet, the harvested energy is 0.398, 0.392 and 0.344 mJ per device when they use the quadratic, P2110B receiver and linear model, respectively. On the other hand, the quadratic model achieves the highest number of successes. For example, from Figure 5c, we see that the quadratic model achieves three successes per frame at $E_{min} = 0.65$ mJ. However, for the P2110B receiver and linear model, there are only 2.6 and 2.5 successes, respectively. Lastly, from Figure 5a, we observe that the throughput reduces with increasing E_{min} value for all three models. We notice that the average frame size also reduces when increasing the value of E_{min} for all three models, see Figure 5c. This is because our approach reduces the frame size when the transmission attempts decrease in order to maximize throughput by reducing the number of idle slots.



(a) Throughput versus E_{min}.



(b) A comparison of harvested energy.

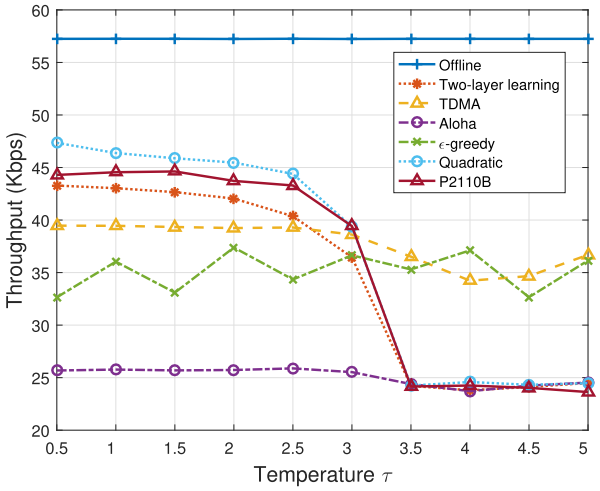


(c) Slot state comparison.

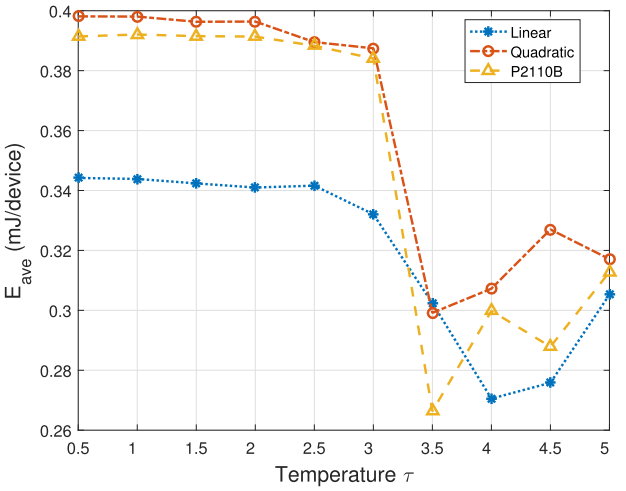
FIGURE 5. **Comparison of linear, quadratic model and P2110B receiver.**

B. TEMPERATURE

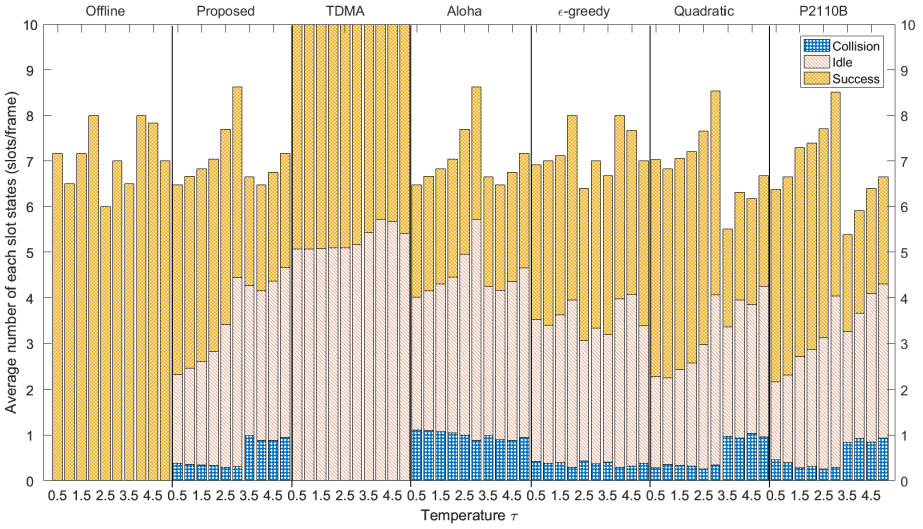
In this experiment, we fix the minimum energy E_{min} to 0.3 mJ. We vary the temperature τ of Equ. (9), Equ. (12) and Equ. (16) from 0.5 to 5. Figure 6a and Figure 6c show the effect of various τ values on the throughput and the average number of slots for each each state. Referring to Figure 6a, the throughput of the two-layer learning decreases from 44 to 25 Kbps with increasing τ values. This is because lower τ values result in lower exploration. That is, the probability of the action with the highest reward is much higher than other actions when we set τ to a low value. In this case, devices/HAP will select this action frequently over other actions, which results in a high throughput. For example, Figure 7a shows the PMF of frame sizes for $\tau = 0.5$ and 5. We see that a frame size of seven and eight are likely to be used with a frequency of 95% and 5%, respectively. Other frame sizes are not selected because they have a low reward. By contrast, when we use a high τ value, i.e., five, frame sizes or actions with a low reward obtain a higher probability as compared to when $\tau = 0.5$. For example, referring to Figure 7a, a frame size of nine slots has probability 0.19 and zero when we use $\tau = 0.5$ and 5, respectively. Referring to Figure 6a, we also notice that the throughput of our proposed approach is equal to Aloha when the temperature τ is higher than 3.5. This means devices could not find a specific slot to transmit. Referring to Figure 7b and 7c, device 1 selects slot 2 with probability one when $\tau = 0.5$, while it selects slot 5 and 7 with a probability 0.1 and 0.3, respectively when $\tau = 5$. All other slots are chosen with a probability of 0.6. This results in higher collisions. From Figure 6c, we see that the average number of collision slot per frame of our proposed approach is 0.2 and 0.9 when τ is set to 0.5 and 5, respectively. Additionally, our proposed approach and Aloha in Figure 6c have the same performance when the temperature τ is higher



(a) Throughput versus τ.

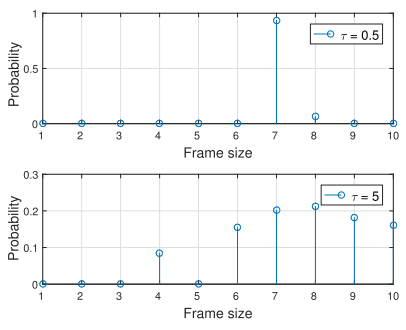


(b) A comparison of harvested energy.

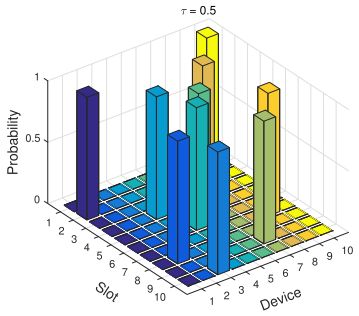


(c) Slot state comparison.

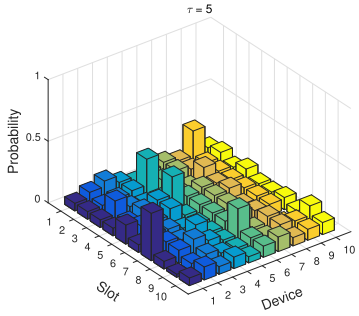
FIGURE 6. Comparison of our two-layer approach versus offline, TDMA, Aloha and ϵ-greedy, quadratic model and P2110B receiver.



(a) PMF of frame size.



(b) PMF of devices when τ = 0.5.

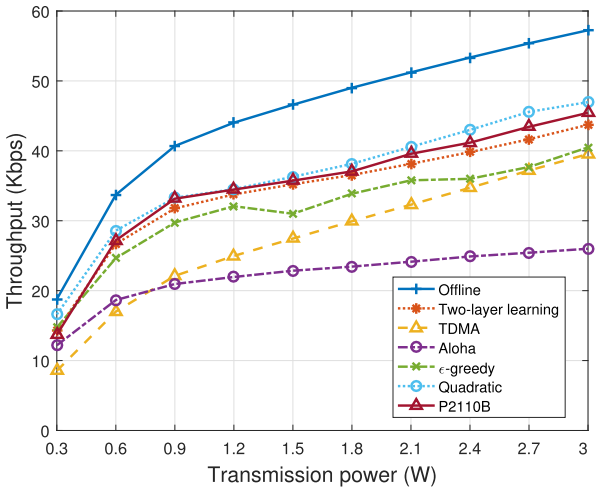


(c) PMF of devices when τ = 5.

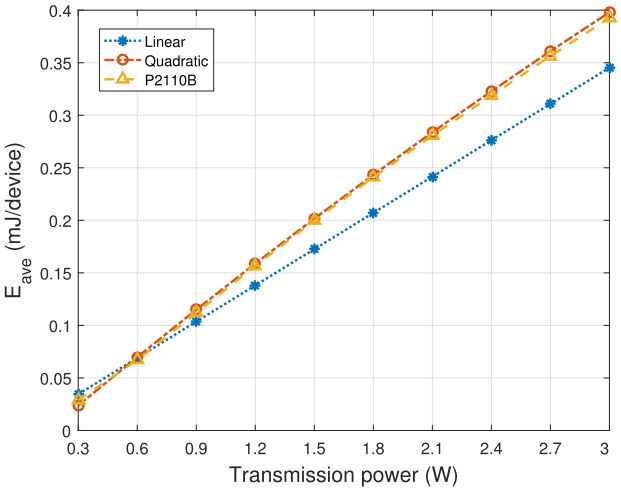
FIGURE 7. Comparison of PMF with high and low temperature τ.

than 3.5, where they have one collision slot and 3.2 idle slot on average when τ is set to 3.5. Thus, the device layer cannot converge if τ is higher or equal to 3.5. On the other hand, the parameter τ has no impact on the Offline and ϵ-greedy.

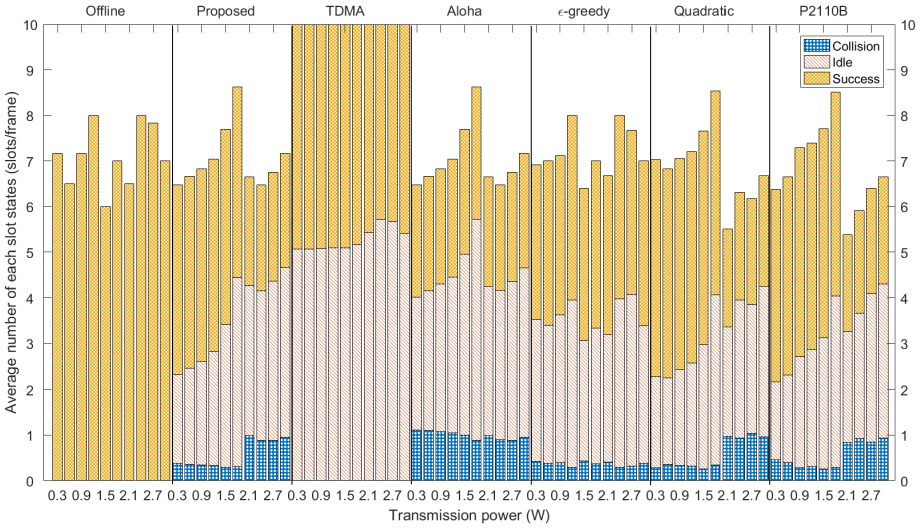
Figure 6a compares the throughput of three energy conversion models. We observe that the quadratic model obtains the highest throughput when τ is less than 3.5. This is because when devices use the quadratic model, they harvest more



(a) Throughput versus P^k.



(b) Comparison of harvested energy.



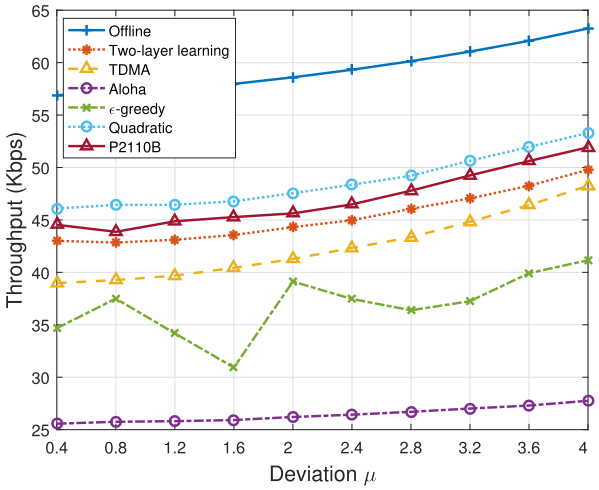
(c) Slot state comparison.

FIGURE 8. Comparison of our two-layer approach with offline, TDMA, Aloha and ϵ-greedy, quadratic model and P2110B receiver.

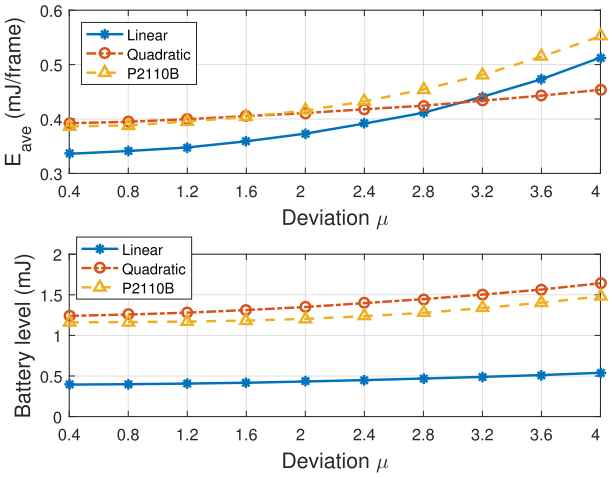
energy as compared to when they use the linear model and P2110B, see Figure 6b. However, when the temperature τ is higher than 3.5, all three models achieve the same throughput at 24 Kbps. This is because devices experience more collisions and fewer successes. For example, when devices use P2110B, they experience on average 0.2 and 0.8 collisions per frame, 4.6 and 2.2 number of successes per frame when τ is set to 3 and 3.5 in Figure 6c. Referring to Figure 6b, the average harvested energy decreases when τ increases from 2.5 to 3.5. This means the HAP fails to learn the best transmission power under these τ values. The decrease in harvested energy means devices have a lower battery level, which reduce their transmission attempts. Therefore, throughput decreases from 44 to 24 Kbps when devices use P2110B; see Figure 6a. Additionally, referring to Figure 6a, we see that the throughput of both TDMA and Aloha also decreases when τ increases from 2.5 to 3.5. Recall that devices in TDMA and Aloha harvest the same amount energy with our proposed approach. Thus, this also proves that the transmission power of the HAP does not converge when τ is higher than 2.5. To conclude, increasing the value of τ result in a longer frame and low throughput. The transmission power fails to converge when τ value is higher than 2.5. The device layer fails to converge when the value of τ is higher than 3.5.

C. FIXED TRANSMISSION POWER

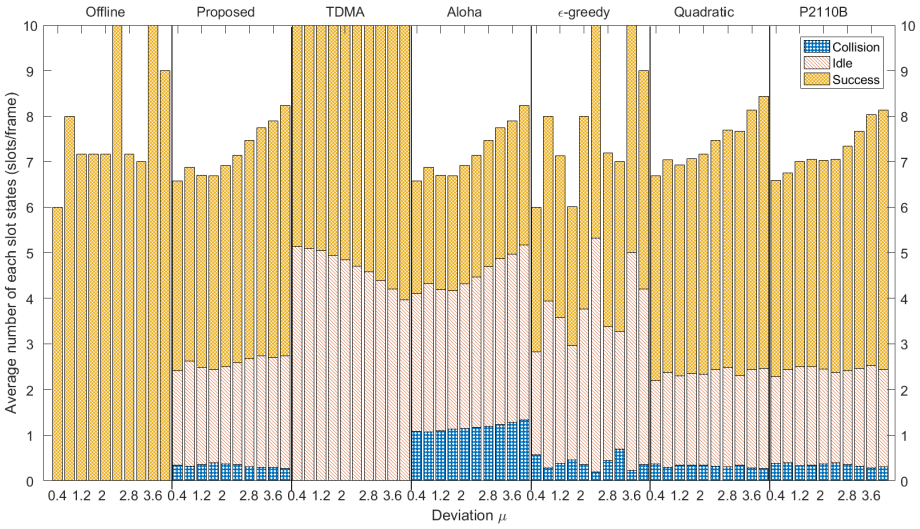
In this experiment, the HAP has a fixed transmission power that ranges from 0.3 to 3 W. The HAP only learns the best frame size. Figure 8a shows that the throughput increases with higher transmission power. For example, the throughput of our proposed approach increases from 12 to 43 Kbps when devices use the linear conversion model. The reason is that



(a) Throughput versus D.



(b) Comparison of harvested energy.



(c) Slot state comparison.

FIGURE 9. Comparison of our two-layer approach with offline, TDMA, Aloha and ϵ-greedy, quadratic model and P2110B receiver.

devices receive more energy when the HAP uses a high transmission power, which results in more transmission attempts and a higher throughput. For example, Figure 8b shows that devices receive 0.04 and 0.35 mJ when the transmission power is set to 0.3 and 3 W, respectively. Additionally, referring to Figure 8c, the HAP receives one and 4.3 packets on average when its transmission power is 0.3 and 3 W, respectively. We also notice that the two-layer learning approach outperforms TDMA for all transmission powers. For example, the throughput of the two-layer learning approach is 9 Kbps higher than TDMA when transmission power is set to 3 W, see Figure 8a. The reason is that the HAP employs a shorter frame size than TDMA in order to reduce the number of idle slots. Therefore, the slots in TDMA tend to be idle, i.e., more than four times than our proposed approach when the transmission power is 0.3 W, see Figure 8c. On the other hand, our proposed approach only achieves 4 Kbps higher throughput than TDMA when the HAP uses 0.3 W to transmit as there are fewer transmission attempts. Aloha achieved 85% and 62% of the throughput attained by our approach when the HAP uses a transmission power of 0.3 and 3 W respectively, see Figure 8a. This is because when devices use the Aloha protocol, they experience more collisions with the increase of transmission power. The ϵ-greedy learning approach achieves the same throughput with our proposed approach when the HAP uses 0.3 W to send energy, i.e., 13 Kbps. However, the performance of Aloha degrades to 93% of our proposed approach when the HAP increases its power to 3 W. This is because there are more devices attempting transmissions when the transmission power is high. The high traffic load means more collisions and low reward when the HAP employs a short frame size. Therefore, the HAP selects a larger frame size than our proposed approach. Additionally, a large number of slots are idle if the frame size is large.

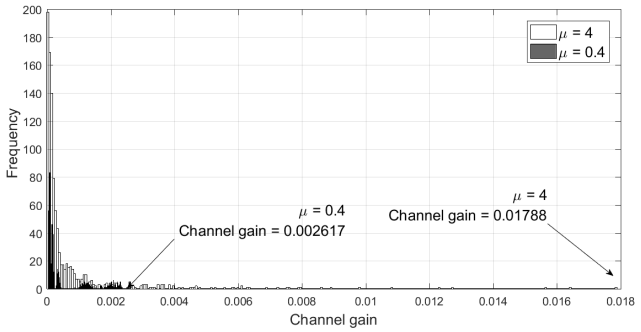


FIGURE 10. Comparison of channel gain values.

For example, referring to Figure 8c, the average frame size of ϵ-greedy is 0.9 larger than our proposed approach, which results in more than 0.5 idle slots than our proposed approach. This means the ϵ-greedy obtains better performance only at low traffic load. Referring to Figure 8b, the average harvested energy of each device grows with increasing transmission power for all three conversion models. We see that the devices that use the linear model harvest more energy if the HAP uses a low power to send energy, i.e., 0.01 mJ higher than quadratic model and P2110B receiver when transmission power is 0.3 W. The reason is that the conversion efficiency of both quadratic model and P2110B receiver is lower than the linear model if the input power is low. However, when the HAP increases its power higher than 0.6 W, the harvested energy of quadratic model and P2110B is higher than linear model. The reason is that the conversion efficiency of the former two models exceeds the linear model.

D. CHANNEL CONDITION

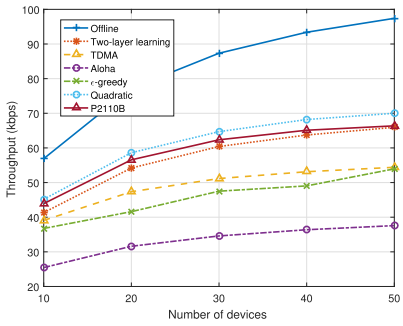
In this experiment, we vary the deviation μ of the Gaussian distribution from 0.4 to 4, with mean of zero. The goal is to study the performance of our system with mild to severe channel conditions. Figure 9a shows that the throughput grows with increasing μ values. This is because there are more success slots. For example, see the performance of our proposed approach in Figure 9c, the average number of success slots increases from 4.3 to 5.3 per frame when μ changes from 0.4 to 4. We plot the frequency distribution of channel gains when μ is set to 0.4 and 4 in Figure 10. We see that when using $\mu = 4$, the frequency of large channel gains, i.e., greater than 0.002, is much higher than when μ is equal to 0.4. A large channel gain equates to a high amount of harvested energy. This explains why the throughput grows with increasing energy conversion efficiency. In all experiments, our proposed approach achieves a higher throughput than TDMA and Aloha regardless the deviation.

From Figure 9b, we see that the harvested energy increases for all three energy conversion models. For example, when deviation is four, devices using the quadratic model harvest 31% higher energy when the μ is 0.4. We also notice that the harvested energy of the quadratic model is lower than P2110B and the linear model when μ is higher than 3.2. Recall that the conversion efficiency decreases when the input power is higher than 1.5 mW, see Figure 3b in Section III. This means a low energy conversion rate when the channel gain is high. Referring to Figure 10, when μ is set to four, the maximum channel gain is 0.018, which is 6.8 times higher than the largest channel gain generated by the Gaussian distribution with a deviation of 0.4. In this case, devices using the quadratic model harvest less energy. However, although the quadratic model achieves lower harvested energy under large μ values, it obtains a higher throughput than the other two models. We plot the average battery level in order to explain this phenomenon. From Figure 9b, we see that the quadratic model achieves the highest average battery level. Recall that the battery has a capacity of 5 mJ. Therefore, the harvested energy within a frame is limited by the battery capacity. This explains why devices using the quadratic model achieves the highest throughput even when they harvested lower energy on average than the linear and P2110B models.

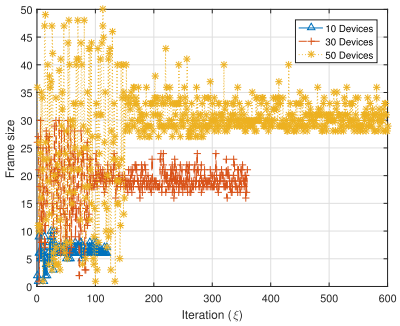
E. NETWORK DENSITY

In this experiment, we increase the number of devices, i.e., $|N|$, from ten to 50 with a step size of ten. Referring to Figure 11a, as we have more devices, throughput increases. In particular, when $|N|$ is ten, the throughput of our two-layer learning approach and TDMA is 42 and 39 Kbps, respectively. As $|N|$ increases to 50, our two-layer learning approach and TDMA improved by as much as 24 and 14 Kbps, respectively. This is because there are more transmission attempts. Referring to Figure 11d, the average number of success slots of our proposed approach increases by 15 per frame when $|N|$ rises from ten to 50. However, the average number of collision slots only increased by 2.4 per frame. This means there are more slots with a successful transmission as compared to those with collision. The results thus confirm devices are able to find the best slot for each frame size. This also explains why the throughput increases with more devices.

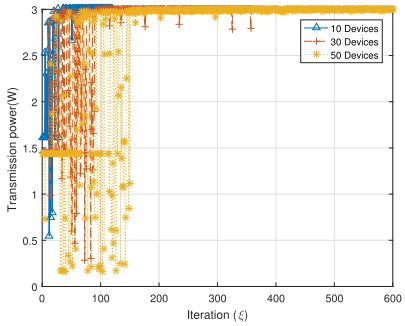
Figure 11b and 11c show the evolution of frame size and transmission power with iteration ξ, respectively. The converged frame size is correspondingly higher with increasing $|N|$ because more slots are required to ensure these devices experience minimal collision or transmit successfully. Referring to Figure 11c, the HAP uses a low power before 150 iterations when $|N|$ is 50. After that, the HAP uses the maximum transmit power. This is because collisions occur before devices are able to find the best slot. The HAP thus uses a lower power to reduce collisions or number of contending devices. After devices converge to their respective best slot, meaning devices will experience minimal collision, the HAP supplies more energy in order to maximize throughput. We also notice that with increasing $|N|$, the HAP requires a higher number of iterations before converging on the best frame size and transmission power. For example, a network with ten devices requires 125 iterations to converge. When $|N|$ increases to 50, the convergence time increased by five times. This is because the number of actions, i.e., devices and frame size, grows with increasing network density.



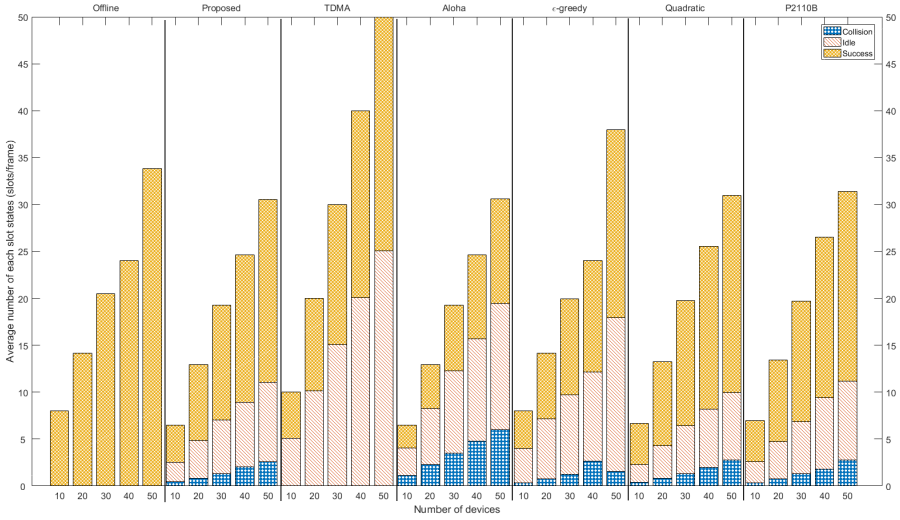
(a) Throughput versus $|N|$.



(b) Frame size versus iteration ξ.



(c) Transmission power versus iteration ξ.



(d) Slot state comparison.

FIGURE 11. **Comparison of our two-layer approach with offline, TDMA, Aloha and ϵ-greedy, quadratic model and P2110B receiver.**

VII. CONCLUSION

We have considered the channel access problem in RF-charging networks where a HAP charges a number of devices. This problem is significant as a HAP needs to determine the transmission power and frame size that yields the maximum throughput. In addition, devices need to determine the opportune time slot to transmit. Our novel two-layer learning approach allows the HAP to learn the best transmission power and frame size without any channel or energy level information. Devices are also able to independently select a transmission slot that yields the highest throughput. Simulation results show that our proposed approach is able to achieve 7% higher throughput than TDMA. Our two-layer learning approach also outperforms TDMA under high traffic load, severe channel state and high network density. An immediate future work is to consider age of information. Another interesting research direction is to incorporate full-duplex capability at the HAP.

REFERENCES

[1] X. Lu, P. Wang, D. Niyato, and Z. Han, "Resource allocation in wireless networks with RF energy harvesting and transfer," *IEEE Netw.*, vol. 29, no. 6, pp. 68–75, Nov./Dec. 2015.

[2] R. Vyas, H. Nishimoto, M. Tentzeris, Y. Kawahara, and T. Asami, "A battery-less, energy harvesting device for long range scavenging of wireless power from terrestrial TV broadcasts," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Montreal, QC, Canada, Jun. 2012, pp. 1–3.

[3] V. Talla, B. Kellogg, B. Ransford, and S. Naderiparizi, "Powering the next billion devices with Wi-Fi," in *Proc. ACM CoNEXT*, Heidelberg, Germany, Dec. 2015, p. 4.

[4] S. Gollakota, M. Reynolds, J. Smith, and D. Wetherall, "The emergence of RF-powered computing," *Computer*, vol. 47, no. 1, pp. 32–39, Jan. 2014.

[5] C. Lu, A. Saifullah, B. Li, M. Sha, H. Gonzalez, D. Gunatilaka, C. Wu, L. Nie, and Y. Chen, "Real-time wireless sensor-actuator networks for industrial cyber-physical systems," *Proc. IEEE*, vol. 104, no. 5, pp. 1013–1024, May 2016.

[6] H. Ju and R. Zhang, "Throughput maximization in wireless powered communication networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 418–428, Jan. 2014.

[7] A. Ortiz, H. Al-Shatri, X. Li, T. Weber, and A. Klein, "Reinforcement learning for energy harvesting point-to-point communications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1–6.

[8] L Powercast. (2016). *915 MHz Power Harvester Receiver*. [Online]. Available: http://www.powercast.com

[9] E. Boshkovska, D. W. K. Ng, N. Zlatanov, and R. Schober, "Practical non-linear energy harvesting model and resource allocation for SWIPT systems," *IEEE Commun. Lett.*, vol. 19, no. 12, pp. 2082–2085, Dec. 2015.

[10] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless networks with RF energy harvesting: A contemporary survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 757–789, 2nd Quart., 2015.

[11] Z. Hadzi-Velkov, I. Nikoloska, G. K. Karagiannidis, and T. Q. Duong, "Wireless networks with energy harvesting and power transfer: Joint power and time allocation," *IEEE Signal Process. Lett.*, vol. 23, no. 1, pp. 50–54, Jan. 2016.
[12] H. Ju and R. Zhang, "Optimal resource allocation in full-duplex wireless-powered communication network," *IEEE Trans. Commun.*, vol. 62, no. 10, pp. 3528–3540, Oct. 2014.
[13] E. Boshkovska, D. W. K. Ng, N. Zlatanov, A. Koelpin, and R. Schober, "Robust resource allocation for MIMO wireless powered communication networks based on a non-linear EH model," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 1984–1999, May 2017.
[14] S. Zhong and X. Wang, "Energy allocation and utilization for wirelessly powered IoT networks," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2781–2792, Aug. 2018.
[15] J. Yang, J. Hu, K. Lv, Q. Yu, and K. Yang, "Multi-dimensional resource allocation for uplink throughput maximisation in integrated data and energy communication networks," *IEEE Access*, vol. 6, pp. 47163–47180, 2018.
[16] K. Lv, J. Hu, Q. Yu, and K. Yang, "Throughput maximization and fairness assurance in data and energy integrated communication networks," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 636–644, Apr. 2017.
[17] H.-H. Choi and W. Shin, "Slotted ALOHA for wireless powered communication networks," *IEEE Access*, vol. 6, pp. 53342–53355, 2018.
[18] D. K. Klair, K.-W. Chin, and R. Raad, "A survey and tutorial of RFID anti-collision protocols," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 3, pp. 400–421, 3rd Quart., 2010.
[19] H. H. R. Sherazi, L. A. Grieco, and G. Boggia, "A comprehensive review on energy harvesting MAC protocols in WSNs: Challenges and tradeoffs," *Ad Hoc Netw.*, vol. 71, pp. 117–134, Mar. 2018.
[20] N. Michelusi and M. Levorato, "Energy-based adaptive multiple access in LPWAN IoT systems with energy harvesting," in *Proc. IEEE Int. Symp. Inf. Theory*, Aachen, Germany, Jun. 2017, pp. 1112–1116.
[21] A. Biason, S. Dey, and M. Zorzi, "Decentralized transmission policies for energy harvesting devices," in *Proc. IEEE Wireless Commun. Netw. Conf.*, San Francisco, CA, USA, Mar. 2017, pp. 1–6.
[22] F. Iannello, O. Simeone, P. Popovski, and U. Spagnolini, "Energy group-based dynamic framed ALOHA for wireless networks with energy harvesting," in *Proc. CISS*, Princeton, NJ, USA, Mar. 2012, pp. 1–6.
[23] J. H. Ahn and T.-J. Lee, "ALLYS: All you can send for energy harvesting networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 4, pp. 775–788, Apr. 2018.
[24] F. Vázquez-Gallego, J. Alonso-Zarate, and L. Alonso, "Reservation dynamic frame slotted-ALOHA for wireless M2M networks with energy harvesting," in *Proc. IEEE ICC*, London, U.K., Jun. 2015, pp. 5985–5991.
[25] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surveys Tuts.*, to be published.
[26] P. Blasco, D. Gündüz, and M. Dohler, "A learning theoretic approach to energy harvesting communication system optimization," *IEEE Trans. Wireless Commun.*, vol. 12, no. 4, pp. 1872–1882, Apr. 2013.
[27] M. Chu, H. Li, X. Liao, and S. Cui, "Reinforcement learning-based multiaccess control and battery prediction with energy harvesting in IoT systems," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2009–2020, Apr. 2018.
[28] K. Huang and E. Larsson, "Simultaneous information and power transfer for broadband wireless systems," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5972–5986, Dec. 2013.
[29] X. Xu, A. Özçelikkale, T. McKelvey, and M. Viberg, "Simultaneous information and power transfer under a non-linear RF energy harvesting model," in *Proc. IEEE ICC Workshops*, Paris, France, May 2017, pp. 179–184.
[30] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org
[31] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*. Cambridge, MA, USA: MIT Press, 1998.
[32] A. Lazaric, M. Restelli, and A. Bonarini, "Reinforcement learning in continuous action spaces through sequential Monte Carlo methods," in *Proc. 20th Int. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2008, pp. 833–840.

HANG YU received the B.E. degree from the University of Wollongong, Australia, and Zhengzhou University, China, in 2017. She is currently pursuing the Ph.D. degree with the University of Wollongong. Her current research interest includes learning methods for RF-energy harvesting networks.

KWAN-WU CHIN received the B.S. degree (Hons.) and the Ph.D. degree with the vice-chancellor commendation from Curtin University, Australia, in 1997 and 2000, respectively. He was a Senior Research Engineer with Motorola, from 2000 to 2003. In 2004, he joined the University of Wollongong as a Senior Lecturer before being promoted to the rank of Associate Professor, in 2011, where he is currently an Associate Professor. He holds four U.S. patents and has published more than 120 conference and journal articles. His current research areas include medium access control protocols for wireless networks and resource allocation algorithms/policies for communications networks.