electronics

MDPI

Article

Prioritized Uplink Resource Allocation in Smart Grid Backscatter Communication Networks via Deep Reinforcement Learning

Zhixiang Yang [1], Lei Feng [1], Zhengwei Chang [2], Jizhao Lu [3], Rongke Liu [4], Michel Kadoch [5,*] and Mohamed Cheriet [5]

1 State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and telecommunications, Beijing 100876, China; yangzx@bupt.edu.cn (Z.Y.); fenglei@bupt.edu.cn (L.F.)
2 State Grid Sichuan Electric Power Company, Chengdu 610041, China; changzw@126.com
3 State Grid Henan Electric Power Company, Zhengzhou 475005, China; sg_lujizhao@163.com
4 School of Electronic and Information Engineering, Beihang University, Beijing 100191, China; rongke_liu@buaa.edu.cn
5 École de Technologie Supérieure, University of Quebec, Montreal, QC H3C 1K3, Canada; mohamed.cheriet@etsmtl.ca
* Correspondence: michel.kadoch@etsmtl.ca

Received: 10 March 2020; Accepted: 29 March 2020; Published: 8 April 2020

Abstract: With the rapid increase in the number of wireless sensor terminals in smart grids, backscattering has become a very promising green technology. By means of backscattering, wireless sensors can either reflect energy signals in the environment to exchange information with each other or capture the energy signals to recharge their batteries. However, the changing environment around wireless sensors, limited radio frequency and various service priorities in uplink communications bring great challenges in allocation resources. In this paper, we put forward a backscatter communication model based on business priority and cognitive network. In order to achieve optimal throughput of system, an asynchronous advantage actor-critic (A3C) algorithm is designed to tackle the problem of uplink resource allocation. The experimental results indicate that the presented scheme can significantly enhance overall system performance and ensure the business requirements of high-priority users.

Keywords: backscatter; deep reinforcement learning; A3C; priority strategy; resource allocation; K-means clustering

1. Introduction

With the rapid development of smart grids, a growing number of wireless sensors are deployed in power grids to sense and monitor transmission lines, substations, homes, etc. [1]. The devices in wireless sensor networks (WSNs) are autonomous and resource-limited. To extend battery life and reduce maintenance costs, WSNs require robust and intelligent approaches to deal with data exchange, information security, resource allocation and strategy optimization. Specifically, the sensor node needs to select the optimal policy from a set of accessible policy groups. Markov decision process (MDP) is a powerful tool for designing adaptive algorithms and protocols, and thus [2] reviewed a number of instances of MDP framework applications.

However, as the scale of WSNs continues to expand, wireless powered communication networks (WPCNs) are emerging to charge energy-constrained sensor devices. The authors of [3] derived the signal-to-noise-ratio (SNR) outage zone in a energy free backscatter WPCN. Simulation results indicated that the backscatter WPCN coverage range is larger and the SNR outage zone is smaller

compared to the active WPCN. The authors of [4] maximized the throughput of backscatter and radio frequency (RF) WSNs by formulating the optimization objectives.

Harvest-the-transmit (HTT) protocol is employed in a traditional RF-powered cognitive network (CRN) where the secondary transmitter (ST) harvests the energy in primary signals and then transmits information to the receiver using the energy stored in the battery actively. Recently, ambient energy signals have been used to transmit ST information by backscattering. The overhead of transmitting data is further reduced. Therefore, it is advisable to combine ambient backscatter communication with RF-powered CRN. The authors of [5] analyzed and compared the performance and throughput of RF-powered CRN in overlay and underlay scenarios. Internet of Things (IoT) is regarded as an ultimate solution to connect everything and has aroused extensive interest in academia and industry. However, energy limitation and implementation cost are the challenges in the construction of IoT. In [6], ambient backscatter communication (AmBC) was introduced as a green communication paradigm that could exploit the ambient wireless signals to power sensors and backscatter the information.

There have been many optimization problems for RF-powered backscatter networks. The authors of [7] studied a full-duplex communication mode in the AmBC network where a full-duplex communication device transmitted the energy signal while the receiving signal was reflected back from the backscatter devices (BDs). An iterative algorithm was proposed to optimize the minimum throughput for all BDs by jointly scheduling time, power resources and adjustment reflection coefficients. The authors of [8] introduced backscatter communication into a multi-user device-to-device (D2D) network where the device could harvest energy from RF signals. A game-theoretic approach was proposed to balance relaying performance and energy harvesting.

The above algorithms need to know the complete environment information. However, it is difficult to get the dynamic characteristics of the network in time. Reinforcement learning (RL) method can solve this problem. The authors of [9] adopted a MDP framework and proposed a low-complexity online RL scheme to optimize the action decisions of RF-powered secondary users (SUs) according to energy storage, data queue and the primary channel state. The authors in [10] proposed using a Double Deep-Q Network (DDQN) to implement an optimal time scheduling strategy for managing large state and action space scenarios. It was shown by numerical results that the proposed DDQN algorithm produced significant advantages to achieve higher network throughput. In an actual smart grid cognitive network, the number of backscattering device is far more than the number of primary users in the system. However, the backscatter field of existing research does not consider device access priority, which is very important in smart grid communication. In this paper, we propose a deep reinforcement learning (DRL) algorithm based on device business priorities to search optimal strategy and realize the maximization of uplink throughput in a smart grid.

The content of this paper is devided as follows. The previous work is reviewed in Section 2. Section 3 characterizes the backscatter communication system model in a CR-based smart grid and derives an objective function. Section 4 presents the DRL algorithm for resource allocation. Simulation and performance evaluation are given in Section 5. In Section 6, we summarize the paper and indicate some future research avenue.

2. Related Work

Some studies have investigated network models combining backscatter communication. In [11], the authors integrated backscatter communication with the Non-Orthogonal Multiple Access (NOMA) system in a novel way and designed the backscatter-NOMA system based on the IoT and cellular network. Cognitive AmBC allows the BDs to transmit data information without consuming energy through backscattering. Because the primary signals in the environment are the carriers of backscatter communication, the co-channel transmission interference and channel estimation error would directly affect the performance of the backscattering system. In [12], the authors introduced a cloud wireless access network architecture into the AmBC network to realize the high-speed connection between the terminal and the processing center and reduce the influence of the primary channel on the throughput

of the backscattering system. It was demonstrated that the AmBC receiver can recover the desired information from BD. The authors of [13] made an in-depth study of RF source signals and proposed a new cooperative AmBC system in which readers could decode the desired information from RF sources. In [14], a low-complexity evolutionary game algorithm was proposed so that STs could select network access points and service patterns according to practical requirements.

The research on resource allocation of backscatter communication focuses on time and power resources. The authors of [15] introduced a backscattering auxiliary network composed of a hybrid access center and multiple BDs, and they then proposed a transmission strategy optimization problem based on working mode selection and time scheduling. In [16], the authors proposed a two-stage Stackelberg game model to deal with the time scheduling problems in the cognitive network of AmBC. In the first stage, gateway, as the leader, adjusted the price to maximize its profit. In the second stage, ST determined the backscattering time with the goal of utility maximization. The authors of [17] proposed the hybrid transmission mode of HTT and backscatter communication, and realized the optimal throughput of the secondary system by adjusting the time allocated to different modes. The authors of [18] considered power, power splitting ratio and time resources simultaneously. To ensure fairness and security, an optimization goal of maximizing the minimum throughput was proposed. The authors of [19] studied the throughput maximization problem of the full-duplex AmBC system, where the primary access point can simultaneously transmit the signal and receive the backscatter signal through full-duplex communication. An iterative method was proposed for joint time scheduling, power allocation and reflection coefficient adjustment.

Machine learning (ML) is a powerful solution to decision problems and has many applications in backscatter communication. In [20], the authors adopted a Q-learning algorithm to address the optimal strategy of an AmBC system iteratively with only partial environmental information. The authors of [21] proposed a label signal detection method for an AmBC system, which classifies the energy characteristics of received signals by an ML classification algorithm. In [22], a dynamic spectrum access framework was designed for an RF-powered backscatter system, which maximized the system throughput through an online RL algorithm. The authors of [23] proposed a Q-learning method to explore the optimal working mode of an AmBC system under the fading channel environment. Because of energy constraints, BDs sometimes require offloading computing tasks to nearby computing servers through active transmission or low-power backscatter communications. The authors of [24] proposed a deep reinforcement learning algorithm (DRL) is to implement the optimal unloading strategy in a hybrid unloading AmBC network.

However, the previous research work has not considered the priority of the BDs. The BDs are numerous and the spectrum resources and energy resources are limited. It is necessary to consider priorities of different BDs to ensure high-priority service quality in smart grid applications.

3. System Model

3.1. Multi-User Backscatter Communication Network in CR-Based Smart Grid

In this section, we consider a multi-user backscatter communication in CR-based smart grid, as shown in Figure 1. The network consists of a mobile edge computing (MEC) center, multiple primary users (PU) and multiple RF-powered SUs. The MEC holds the basic state information of the terminals and acts as the manager to make the resource allocation strategy. The PUs, such as the substations, are mainly responsible for transmitting control, management and other important information in the smart grid. A large number of sensors that sense information about the environment act as SUs, which have three working modes: backscatter, energy harvest and active transmission.

Assume that the number of PUs and SUs is M and N in the system, respectively. And the PUs are connected to the network with frequency division multiple access to avoid interference. Both PUs and SUs are in demand to transmit information to the receiver, i.e., MEC. When the channel is occupied by a PU, the SU can either conduct energy capture or backscatter communication by superimposing its

own signal on the RF signal of the PU. When the PU leaves the channel to make it free, the SU can access the channel and consume the stored energy to send signals to the MEC. For SU n, the energy captured per time slot is e_n^h units, and d_n^b packets are transmitted per time slot in the backscatter mode. When in active transmission mode, SU n transmits d_n^a packets, and e_n^a units of energy are consumed per time slot.

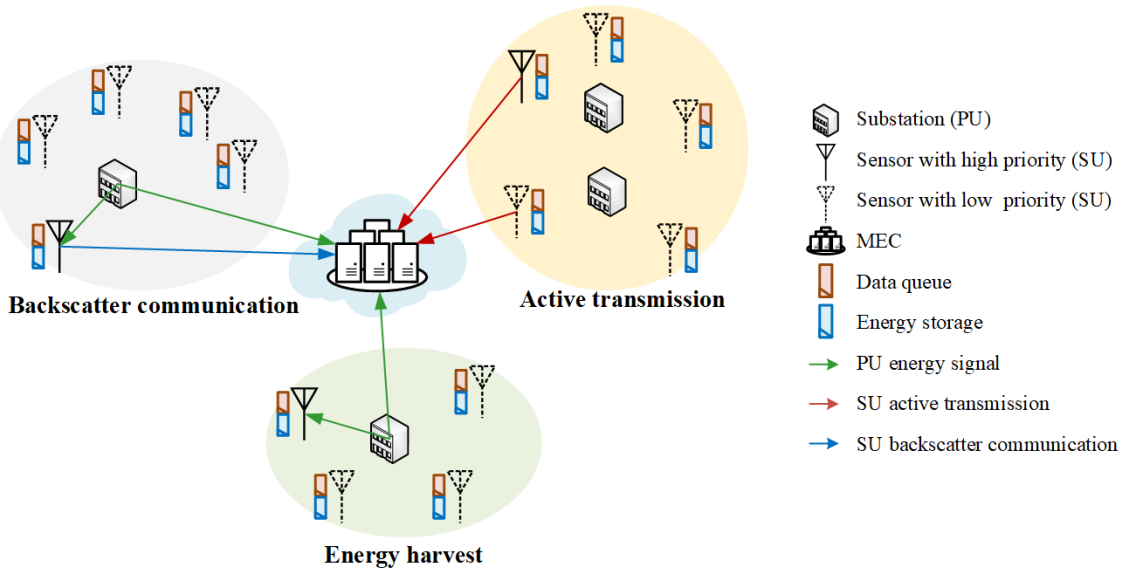

Figure 1. Backscatter communication system model.

3.2. User Scheduling Based on Priority

Because of the differences between the business activities in a smart grid, SUs are classified into different priority classes $j(j = 1, 2, ..., J)$ based on the importance of the information to be transmitted. SUs that transmit important information have a higher priority , while those that transmit information with high delay tolerance have a lower priority. By prioritizing, the information of users with high priority can be transmitted first, and the data of users with low priority can be transmited later. Therefore, the communication delay of high priority users will be greatly reduced. The initial weight value of SU n is represented by $w_{o,n}^j$ which satisfies the constraint $w_{o,n}^1 > w_{o,n}^j > ... > w_{o,n}^J$.

Moreover, when the SU senses an emergency in the smart grid, the MEC will adjust and promote its priority to send the emergency information as soon as possible. Packets of emergency messages will automatically arrive at the head of the data cache queue, that is, packets of emergency messages will be transmitted first when the user transmits data. w_e is defined to represent the emergency weight. Therefore, the real priority weight of SU n at time t, including initial weight and emergency weight, is expressed as,

$$w_n(t) = w_{o,n}^j + w_e 1_{\{E_n(t)=1\}} \tag{1}$$

where $1_{\{\bullet\}}$ is an indicator function. $E_n(t) = 1$ shows that an emergency has occurred, otherwise $E_n(t) = 0$. When the emergency packet is sent, the MEC resets the real priority weight of SUs to the original priority. Particularly, the SUs at priority class $j+1$ sending emergency information have a higher real priority weight than the SUs at class j. This priority-based user scheduling strategy ensures the priority transmission of important and urgent information in the smart grid, reduces the communication delay of these data, and is of great significance to accelerate the decision-making process of the smart grid monitoring center.

3.3. Problem Formulation

To achieve the goal of maximizing throughput, we model the communication process as a MDP, which can be represented by a tuple $< S, A, P, R >$. S is the state space of the whole network. A represents the action space of the MEC. The state transfer function and the reward function are denoted by P and R, respectively. Firstly, the network state space consists of two parts: SU state space and PU state space. We defined the SU state space as,

$$S_n = \left\{ (r_n, c_n, w_n) ; r_n \in \{0, 1, \ldots, R_n\}, c_n \in \{0, 1, \ldots, C_n\}, w_n \in \left\{ w_{c,j},\ w_{c,j} + w_e, j = 1, 2, ..., n \right\}, \right\} \quad (2)$$

where r_n represents the queue states of SU n, and R_n describes the maximum length of the data queue space for SU n. c_n represents the energy state, and C_n denotes the maximum energy storage in SU n battery. $w_n(t)$ represents the weight of SU n, and w_e represents the emergency weight. The PU state space is as follows,

$$S_m = \{ f_m ; f_m \in \{0, 1\} \} \quad (3)$$

where 0 represents the primary user m leaving the channel. Otherwise, it means that the channel is occupied. Therefore, the network state space is expressed as,

$$S = \prod_{m=1}^{M} S_m \times \prod_{n=1}^{N} S_n \quad (4)$$

The network action space can be viewed as a collection of action spaces for each SU. In each time slot, the action space of a SU n can be expressed as,

$$A_n = a_n, p_n ; a_n \in \{1, 2, 3, 4\}, p_n \in \{0, 1, 2, \ldots, M\} \quad (5)$$

where $a_n = 1, 2, 3, 4$ denotes that the SU n performs the actions of keeping waiting, energy harvest, backscattering and active transmission, respectively. p_n represents the result of channel allocation. $p_n \neq 0$ is assigned a channel, otherwise $p_n = 0$. To avoid interference, each idle PU channel can only be allocated to one SU to transmit data actively. The network action space can be expressed as,

$$A = \prod_{n=1}^{N} A_n \quad (6)$$

The transition function is a state transition probability showing that the SU takes an action and the state changes from s to s' with a certain probability at time t. The formula is expressed as,

$$p(s, a, s') = Pr(s(t+1) = s' | s(t) = s,\ a(t) = a) \quad (7)$$

In particular, the environment we have established is a deterministic environment in which the agent executes action a and then transfers it from s to s' with a probability $p(s, a, s') = 1$. Because the objective is to maximize total throughput for total SUs, the design of reward function is directly related to the number of packets sent. In addition, we introduce the real priority weight into the expression. The more important the business of SU, the greater the real priority weight. When primary channel resources are tight, the operation that allocates channel to more important users will get more rewards. The reward function is as follows,

$$R\{s, a\} = \sum_{i=1}^{N} w_n d_n^b 1_{\{a_n = 3\}} + w_n d_n^a 1_{\{a_n = 4\}} \quad (8)$$

The problem in this paper is to solve the maximum throughput of the multi-user backscatter communication system by employing an optimal resource allocation policy strategy. The formula is as follows,

$$\max_{\pi \in \Pi} R\{s, a\} \tag{9}$$

where π, Π represents a certain policy and the policy space is composed of all policies respectively.

4. Resource Allocation Policy

4.1. K-Means Clustering

4.1.1. Algorithm Description

In large-scale user scenarios, users are classified by unsupervised clustering before resource allocation. Given a set of n entities $X = \{x_1, x_2, \cdots, x_n\}$, each entity is represented by a feature vector. The goal of k-means clustering is to divide n entities into k different clusters, assuming that $k < n$. k clusters $G_1, G_2, \cdots, G_k$ form the partition of the entity set, where $G_j \cap G_j = \emptyset$, $\bigcup_{i=1}^{k} G_i = X$. Clustering is represented by function,

$$l = C(i) \tag{10}$$

where $i \in \{1, 2, \cdots, n\}$, $l \in \{1, 2, \cdots, k\}$. Squared Euclidean distance is adopted as the distance between entities,

$$\begin{aligned} d(x_i, x_j) &= \sum_{k=1}^{m} (x_{ki} - x_{kj})^2 \\ &= \|x_i - x_j\|^2 \end{aligned} \tag{11}$$

where m represents the dimension of the feature vector. Define the sum of the distance between the entity and its cluster center as the loss function,

$$W(C) = \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - \bar{x}_l\|^2 \tag{12}$$

where $\bar{x}_l = (\bar{x}_{1l}, \bar{x}_{2l}, \cdots, \bar{x}_{ml})^T$ represents the center of the cluster l. The number of entities in each cluster l is defined as,

$$n_l = \sum_{i=1}^{n} I(C(i) = l) \tag{13}$$

where $I(C(i) = l)$ is the indicator function, which is 0 or 1. K-means clustering is the optimization problem shown below,

$$\begin{aligned} C^* &= \arg\min_{C} W(C) \\ &= \arg\min_{C} \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - \bar{x}_l\|^2 \end{aligned} \tag{14}$$

In fact, the optimal solution of k-means clustering is a NP hard problem, which is solved by iterative method in reality. First, we need to select a clustering center $(m_1, m_2, \cdots, m_k)$. Entities are assigned to the clusters with the nearest centers one by one. The target of partition C is to minimize the sum of the distance between the entity and the cluster center to which it belongs. The objective function of this process is as follows,

$$\min_{C} \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - m_l\|^2 \tag{15}$$

The next step is to recalculate the mean value of the entities of each cluster as the new cluster center value according to the partition result of the previous step. The new clustering center $(m_1, m_2, \cdots, m_k)$ minimizes the sum of the distance between the entity and the cluster center to which it belongs. The loss function is as follows,

$$\min_{m_1, m_2, \cdots, m_k} \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - m_l\|^2 \tag{16}$$

The update formula of the mean m_l of cluster G_l with n_l entities is shown as follows,

$$m_j = \frac{1}{n_l} \sum_{C(i)=l} x_i,\ l = 1, 2, \cdots, k \tag{17}$$

Repeat the above two steps until the partition no longer changes, and get the result of clustering. The pseudo-code for clustering is shown in Algorithm 1.

Algorithm 1: K-Means Algorithm for BDs Clustering.

Input: The set of entities to be clustered, $X = x_1, x_2, \cdots, x_n$
The number of clusters, k
The maximum iterations, T_{max}
Output: The clustering result $C(i), i = 1, 2, \cdots, n$

```
Select K entities randomly as the initial clustering center m^(0) = (m_1^(0), m_2^(0), ..., m_k^(0));
change ← False;
t ← 0;
repeat
    Update new clustering results C^(t) according to formula (15)
    Update new clustering center m^(t+1) according to formula (16) and (17)
    if t ≥ 1 then
        if C^(t) == C^(t−1) then
            changed ← True
    t++;
until changed = True and t ≤ T_max;
```

4.1.2. Clustering Evaluation

We use silhouette coefficient to measure how appropriately the entities have been clustered through k-means. For entity x_i in the cluster c_i, the mean distance between x_i and all other entities in the same cluster is defined as,

$$a(i) = \frac{1}{|c_i| - 1} \sum_{j \in c_i, i \neq j} d(x_i, x_j) \tag{18}$$

where $d(x_i, x_j)$ is the Euclidean square distance between entities x_i and x_j in the cluster c_i. Obviously, the smaller the value $a(i)$, the better the assignment. For each entity $x_i \in c_i$, we define the smallest mean of the distance from i to all entities in C as,

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(x_i, x_j) \tag{19}$$

where $C \neq c_i$. We now define a silhouette value of one entity x_i as follows,

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases} \tag{20}$$

where $-1 \leq s(i) \leq 1$. As $a(i)$ is a measure of how dissimilar x_i is to its own cluster, a small value means it is well matched. Furthermore, a large $b(i)$ implies that x_i is badly matched to its neighboring cluster. Thus an $s(i)$ close to 1 means that the entity is appropriately clustered. If $s(i)$ is close to -1, entity i would be more appropriately clustered in its neighboring cluster. An $s(i)$ near 0 means that the entity is on the border of two natural clusters.

The mean $s(i)$ over all entities of a cluster is a measure of how tightly grouped all the points in the cluster are. Thus the mean $s(i)$ over entire entities is a measure of how appropriately the entities have been clustered. The silhouette coefficient over all the entities for a specific number of clusters k is defined as,

$$\tilde{s}(k) = \frac{1}{n} \sum_{i=1}^{n} s(i) \tag{21}$$

where $-1 \leq \tilde{s}(k) \leq 1$. The closer the $\tilde{s}(k)$ is to 1, the better the clustering. Therefore, we need to calculate the silhouette coefficient under different k values and select the k value corresponding to the maximum silhouette coefficient as the clustering number.

4.2. Deep Reinforcement Learning Algorithm

The target of our algorithm is to discover the actions that are taken to maximize the total throughput. This is because general RL algorithms, such as q-learning, are only applicable to situations in which the state action space is small. In order to obtain the mapping of large-scale state space S to action space A, we decided to adopt the A3C algorithm. A3C integrates the advantages of the policy-based method and the value-based method. In order to accelerate the convergence speed, A3C uses the skill of multi-threading experience playback. Multiple threads learn interactively with the environment simultaneously, and the learning outcomes of each thread are put together and stored in a public place. The learning results are then periodically retrieved from the public place to guide the next interaction with the environment. The A3C framework eliminates the strong correlation of the empirical playback data in an asynchronous way.

At the beginning, A3C requires maintaining a policy function $\pi(a_t|s_t;\theta)$ with the parameter θ and an value function approximation $V(s_t, \theta_v)$ with the parameter θ_v. The agent updates the policy using the estimated value function. To reduce the variance of estimate, the advantage estimate is adopted, which is shown as,

$$A(s, a; \theta, \theta_v) = U_t(\theta_v) - V(s_t; \theta_v) = \sum_{i=0}^{k-1} \beta^i R_{t+i} + \beta^k V(s_{t+k}; \theta_v) - V(s_t; \theta_v) \tag{22}$$

where $U_t(\theta_v)$ is the estimate of state action value. R is the reward in formula (8) and β means the discount coefficient which is greater than zero and less than 1. The neural network is trained to obtain approximation of policy $\pi(a_t|s_t;\theta)$ and the value function $V(s_t, \theta_v)$. At the beginning, the global network parameters are the same for each actor-learner. Afterwards, multiple actor-learners use asynchronous gradient descent to train and optimize their neural networks. The learning process is parallel and independent. Network parameters are updated by the calculated gradient, and the actor-learner sends the new parameters to the global network. Again, the global network propagates the updated weight parameters to the actor-learners to make sure that they share a common policy.

We define loss functions for policy and estimate function, respectively. The policy loss function is as below,

$$f_\pi(\theta) = \log \pi(a_i|s_i;\theta)\, A(s,a;\theta,\theta_v) \tag{23}$$

The gradient is expressed as,

$$\nabla_\theta f_\pi(\theta) = \nabla_\theta \log \pi(a_i|s_i;\theta)\, A(s,a;\theta,\theta_v) \tag{24}$$

The loss function for estimate value function is defined by,

$$f_v(\theta_v) = A(s,a;\theta,\theta_v) \tag{25}$$

Similarly, the gradient of value loss function with respect to θ_v is expressed as,

$$\nabla_{\theta_v} f_v(\theta_v) = \frac{\partial A(s,a;\theta,\theta_v)}{\partial \theta_v} \tag{26}$$

After the neural network is trained for thousands of rounds, the A3C algorithm will give the decision results. It will take seconds for the A3C algorithm to choose a best resource allocation action according to the network state dynamically. The pseudo-code for A3C is summarized in Algorithm 2.

Algorithm 2: A3C-based Resource Allocation Algorithm.

Assume global shared parameter vectors θ and θ_v and global shared counter $T = 0$
Assume thread-specific parameter vectors θ' and θ'_v
Initialize thread step counter $t \leftarrow 1$
while $T < T_{max}$ **do**
    Reset gradient: $d\theta \leftarrow 0$ and $d\theta_v \leftarrow 0$
    Synchronize thread-specific parameters $\theta' = \theta$ and $\theta'_v = \theta_v$
    $t_{start} = t$
    Get state s_t
    repeat
        Perform a_t according to policy $\pi(a_t|s_t;\theta')$
        Receive reward R_t and new state s_{t+1}
        $t \leftarrow t + 1$
        $T \leftarrow T + 1$
    until *terminal* s_t *or* $t - t_{start} == t_{max}$;
    if *terminal* s_t **then**
        $U = 0$
    else
        $U = V(s_t, \theta'_v)$
    for $i \in \{t-1, ..., t_{start}\}$ **do**
        $U \leftarrow R_i + \gamma U$
        accumulate gradients wrt θ': $d\theta \leftarrow d\theta + \nabla_{\theta'} \log \pi(a_i|s_i;\theta')\,(U - V(s_i;\theta'_v))$
        accumulate gradients wrt θ'_v: $d\theta_v \leftarrow d\theta_v + \frac{\partial (U - V(s_i;\theta'_v))}{\partial \theta'_v}$
    perform asynchronous update of θ using $d\theta$ and of θ_v using $d\theta_v$

5. Performance Evaluation

Sources [9,17,23,25] are all about resource allocation in a single user scenario. The multi-user scenario model is more complicated, for the resource allocation must consider the interaction between users. The time scheduling strategy of the multi-user AmBC system was studied in [10,19,24,26].

RL and DRL have certain advantages in dealing with resource allocation problems in time-varying communication networks because of their characteristics of learning in environment interaction. It is one of the characteristics of smart grids that different services vary greatly. The previous literature on backscattering did not consider the priority between the businesses; we introduced the priority strategy into the resource allocation problem. In this section, we evaluate the performance of the proposed DRL algorithm by simulation experiment. The backscatter scheme in [27], HTT in [17] and the random scheme are selected for comparison. Backscatter scheme includes backscattering but not active transmission, while the HTT captures the energy first and then transmits it actively. The random scheme means selecting the action in action space, which includes backscatter, energy harvest and active transmission. To verify the performance of the proposed scheme, we consider convergence, adaptability to the environment and priority policy evaluation.

5.1. Simulation Setting

We consider a network composed of a MEC and multiple users. The small-scale scenario has 3 PUs and 5 SUs, while 18 PUs and 50 SUs are in the large-scale scenario. The locations of all users are randomly distributed. There are ten time slots in a frame. The learning rate of both the action network and the critic network is 0.001. In the hardware environment, we use a laptop equipped with AMD Ryzen 5 2500U and 8G memory. The software environment is Pycharm and Tensorflow 1.8 on Windows 10. The programming language we choose is Python 3.5. RMSProp optimizer is adopted to minimize the loss function in the A3C algorithm. The simulation parameters for the proposed DRL algorithm are shown in Table 1.

Table 1. Simulation Parameters

Parameters	**Values**
The maximum length of date queue	10 packets
The maximum capacity of energy	10 units
The amount of data in each packets	1 kbit
The probability of packet arrival	0.9
The probability that the channel is idle	0.5
The probability of an emergency	0.5
Number of packets transmitted per unit time in backscatter communication (d_n^b)	1 packet
Number of packets transmitted per unit time in active transmission (d_n^a)	2 packets
Number of energy harvest per unit time (e_n^h)	1 unit
Number of energy consumption per unit time (e_n^a)	1 unit
Discount factor (β)	0.9
The maximun length of episode	1000
The length of episode to update global network parameters	10
Network learning rate	0.001
Number of hidden layers	1
Activation	Relu
Optimizer	RMSPropOptimizer

5.2. Convergence Evaluation

In the convergence evaluation experiment, the probability of packet arrival is fixed as 0.9. And the probability of channel idle and emergency are both 0.5. Throughput is an important indicator of performance evaluation. Figure 2 shows the average throughput of the A3C algorithm and the comparison algorithm. Average throughput here represents throughput of all SUs per frame time. As shown in Figure 2, the proposed algorithm converges to a much higher value than the average throughput of the comparison algorithm. In addition, the convergence speed of the A3C algorithm is fast, reaching convergence value in about 120 episodes. The backscatter scheme and the HTT scheme perform poorly because of a single transmission mode. The random scheme performs better because it adopts both backscatter and active communication as the proposed algorithm.

For large-scale user scenarios, we start with clustering. However, the number of clusters is artificially selected. To find out the best k value, we carry out the clustering evaluation experiment through computing the silhouette coefficient in a different number of clusters. As shown in Figure 3, the silhouette coefficient maximizes at $k = 9$. The maximum value of k is 18 because the number of clusters must not exceed the number of PUs to ensure at least one PU in each cluster. We know from the description of the algorithm that the larger the silhouette coefficient, the better the clustering results. Finally, $k = 9$ is selected for the next experiment.

As shown in Figure 4, we conducted comparative experiments on large-scale user scenarios. The experimental process is divided into two steps. First, K-means clustering is carried out according to the location distribution of users, then, the proposed DRL is used to find the optimal strategy in each cluster. As can be seen in the figure, the proposed DRL algorithm has obvious advantages when the convergence value of average throughput is about 15 percent higher than that of the random scheme.

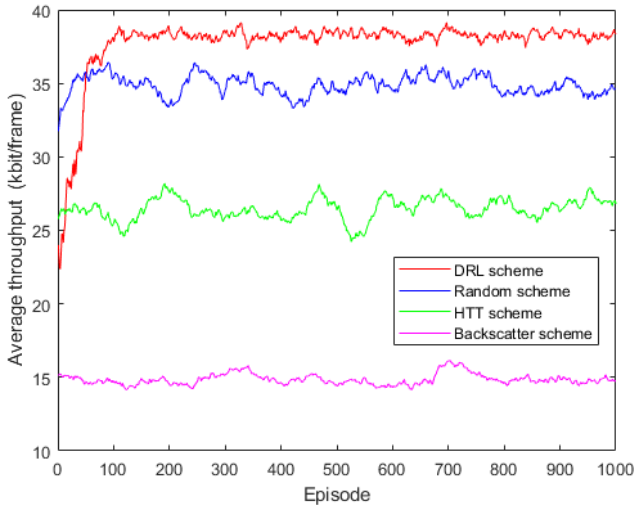

Figure 2. Convergence comparison between proposed DRL scheme and the baseline schemes in small-scale scenarios.

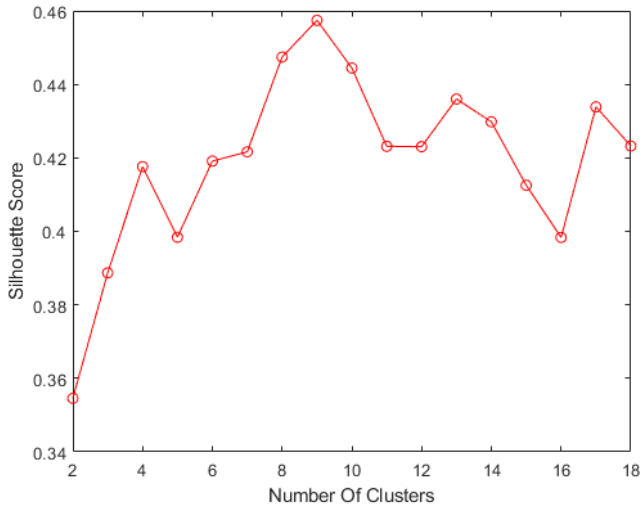

Figure 3. Silhouette coefficient versus the number of clusters.

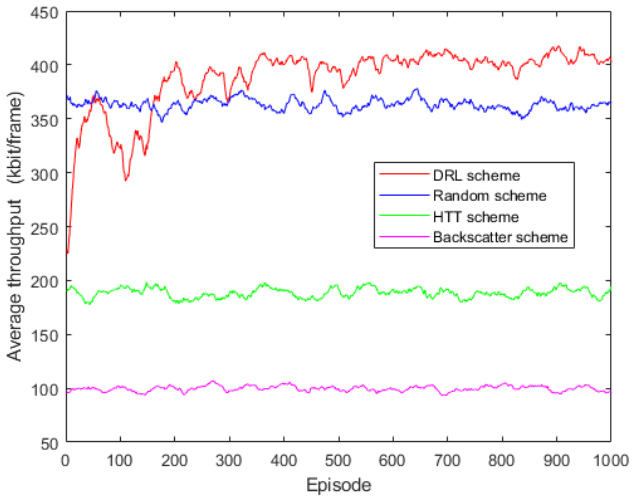

Figure 4. Convergence comparison between proposed DRL scheme and the baseline schemes in large-scale scenarios.

5.3. Adaptability to Environment

The dynamic and changeable power grid environment leads to great uncertainty in the generation time and quantity of information data. To evaluate the adaptability of the proposed DRL algorithm, we consider different scenarios by varying the probability of packet arrival and the PU idle probability. In Figure 5, with the increase of packet arrival probability, the proposed algorithm still performs better than the comparison algorithm. The proposed algorithm can allocate network resources more reasonably. In particular, when the probability of packet arrival changes from 0.4 to 0.5, the growth achieved of average throughout in the DRL scheme is approximately four times that of the HTT or backscatter schemes.

Figure 6 shows the effect of PU idle probability on average throughput. The proposed DRL algorithm can better adapt to the dynamic change of the PU channel. The single transmission mode is greatly affected by environmental changes. In particular, the backscatter scheme performance degrades as primary user idle probability decreases. This is because the backscatter scheme is heavily dependent on the presence of the primary user signal. However, the broken line of the HTT scheme shows a trend of first rising and then falling. This phenomenon is due to the requirement for the PU energy signal from energy harvesting and the requirement for an idle PU channel from the active transmission.

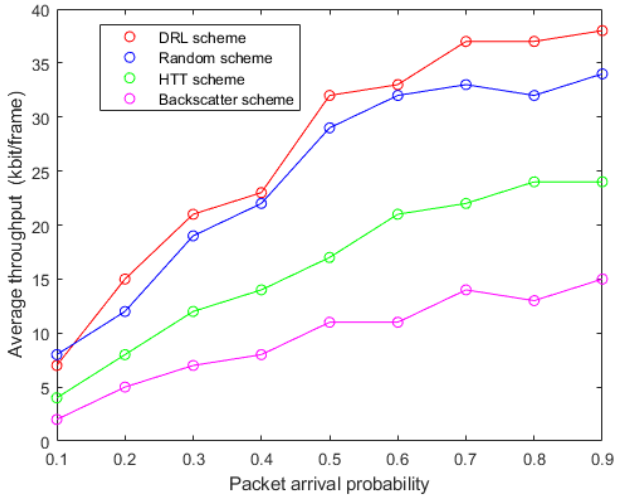

Figure 5. Average throughput of different schemes versus packet arrival probability.

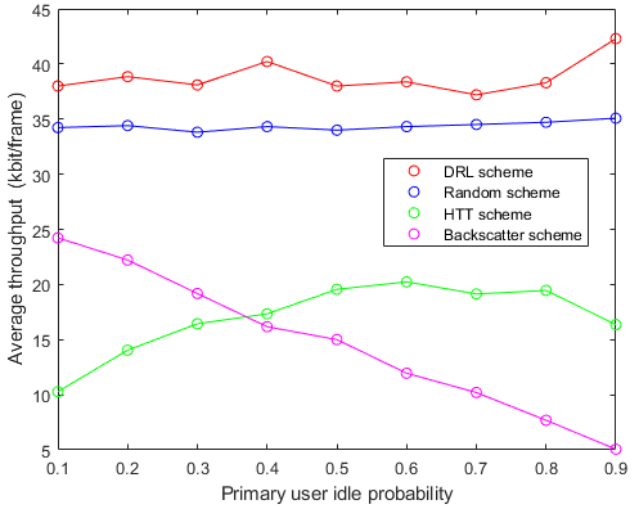

Figure 6. Average throughput of different schemes versus primary user idle probability.

5.4. Priority Policy Evaluation

In the description of the system model, a user priority policy is introduced which gives priority to ensuring the communication needs of high-priority users. The SUs are divided into two classes: with priority and with no priority. The PU idle probability is fixed as 0.5. Figure 7 shows the average throughput versus packet arrival probability for priority and non-priority SUs. As the probability of packet arrival increases, throughput for both priority and non-priority users increases. In particular, a SU with priority always maintains high throughput regardless of the packet arrival probability.

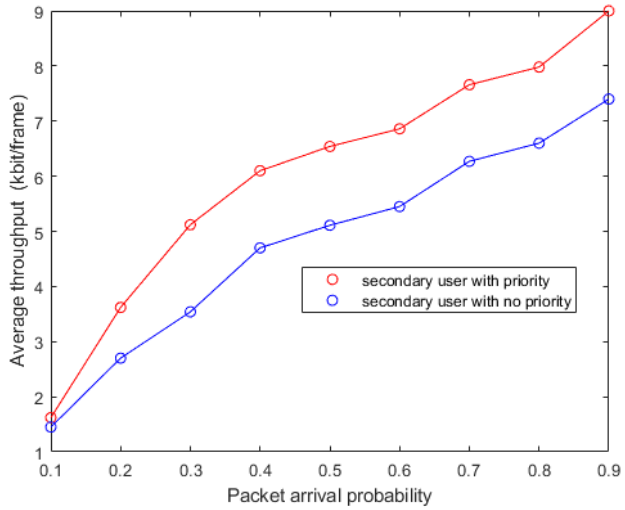

Figure 7. Average throughput of priority strategy versus packet arrival probability.

Figure 8 shows the average throughput when all SUs have a fixed packet arrival probability at 0.9. The smooth trend of the two broken lines is because the proposed algorithm can adapt well to the changes of channel environment. The priority policies ensure that channel allocation gives priority to high-priority users when channel resources are limited. Therefore, the priority user achieves higher average throughput performance.

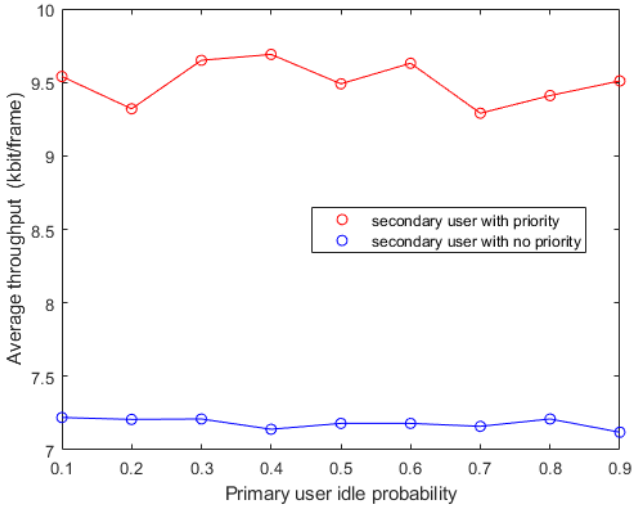

Figure 8. Average throughput of priority strategy versus primary user idle probability.

6. Conclusions and Future Research

In this paper, we propose a multi-user backscatter communication network architecture in CR-based smart grid. The network architecture contains two transmission modes: backscattering and active transmission. Backscattering requires a primary user energy signal. Active transmission requires an idle primary user channel. To improve network performance and increase system throughput, we formulate the resource allocation process as an optimization problem. In addition, we consider the business priorities of different users and introduce a priority strategy. Then, a DRL scheme is proposed to address the problem. The k-means clustering method is adopted to pre-process the application of the DRL scheme in the large-scale user scenario. Large-scale problems are transformed into small-scale problems. The numerical results verify that the proposed scheme enables greater system throughput with limited resources and gives priority to ensuring the throughput of high-priority users. In the future, we will study more flexible backscatter communication systems combined with D2D. More factors will be considered, such as smart jamming attack [20], concurrent transmission, channel conflict [28], signal power, reflection coefficient and energy conversion efficiency. As for solution methods, communication systems are matched with RL algorithms because of their interactive characteristics; therefore, we will give priority to DRL algorithms that have a lot of potential in communication scenarios.

Author Contributions: Conceptualization, Z.Y. and L.F.; methodology, Z.C.; software, J.L.; validation, R.L. and Z.Y.; formal analysis, M.K. and M.C.; investigation, L.F. and and M.C.; writing–original draft preparation, Z.Y.; writing–review and editing, L.F. and M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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

References

1. Zhang, Y.; Li, X.; Zhang, S.; Zhen, Y. Wireless sensor network in smart grid: Applications and issue. In Proceedings of the 2012 World Congress on Information and Communication Technologies, Trivandrum, India, 30 October–2 November 2012; pp. 1204–1208.
2. Abu Alsheikh, M.; Hoang, D.T.; Niyato, D.; Tan, H.; Lin, S. Markov Decision Processes with Applications in Wireless Sensor Networks: A Survey. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 1239–1267. [CrossRef]
3. Choi, S.H.; Kim, D.I. Backscatter radio communication for wireless powered communication networks. In Proceedings of the 2015 21st Asia-Pacific Conference on Communications (APCC), Kyoto, Japan, 14–15 October 2015; pp. 370–374.

4. Kwan, J.C.; Fapojuwo, A.O. Sum-Throughput Maximization in Wireless Sensor Networks with Radio Frequency Energy Harvesting and Backscatter Communication. *IEEE Sens. J.* **2018**, *18*, 7325–7339. [CrossRef]
5. Hoang, D.T.; Niyato, D.; Wang, P.; Kim, D.I.; Han, Z. Ambient Backscatter: A New Approach to Improve Network Performance for RF-Powered Cognitive Radio Networks. *IEEE Trans. Commun.* **2017**, *65*, 3659–3674. [CrossRef]
6. Zhang, W.; Qin, Y.; Zhao, W.; Jia, M.; Liu, Q.; He, R.; Ai, B. A green paradigm for Internet of Things: Ambient backscatter communications. *China Commun.* **2019**, *16*, 109–119.
7. Yang, G.; Yuan, D.; Liang, Y. Optimal Resource Allocation in Full-Duplex Ambient Backscatter Communication Networks for Green IoT. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, UAE, 9–13 December 2018.
8. Li, J.; Xu, J.; Gong, S.; Li, C.; Niyato, D. A Game Theoretic Approach for Backscatter-Aided Relay Communications in Hybrid Radio Networks. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, UAE, 9–13 December 2018.
9. Huynh, N.V.; Hoang, D.T.; Nguyen, D.N.; Dutkiewicz, E.; Niyato, D.; Wang, P. Reinforcement Learning Approach for RF-Powered Cognitive Radio Network with Ambient Backscatter. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, UAE, 9–13 December 2018.
10. Anh, T.T.; Luong, N.C.; Niyato, D.; Liang, Y.; Kim, D.I. Deep Reinforcement Learning for Time Scheduling in RF-Powered Backscatter Cognitive Radio Networks. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 6–9 April 2019.
11. Zhang, Q.; Zhang, L.; Liang, Y.; Kam, P.Y. Backscatter-NOMA: An Integrated System of Cellular and Internet-of-Things Networks. In Proceedings of the ICC 2019—2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019.
12. Darsena, D.; Gelli, G.; Verde, F. Cloud-Aided Cognitive Ambient Backscatter Wireless Sensor Networks. *IEEE Access* **2019**, *7*, 57399–57414. [CrossRef]
13. Yang, G.; Zhang, Q.; Liang, Y. Cooperative Ambient Backscatter Communications for Green Internet-of-Things. *IEEE Internet Things J.* **2018**, *5*, 1116–1130. [CrossRef]
14. Gao, X.; Feng, S.; Niyato, D.; Wang, P.; Yang, K.; Liang, Y. Dynamic Access Point and Service Selection in Backscatter-Assisted RF-Powered Cognitive Networks. *IEEE Internet Things J.* **2019**, *6*, 8270–8283. [CrossRef]
15. Lyu, B.; Yang, Z.; Gui, G.; Feng, Y. Wireless Powered Communication Networks Assisted by Backscatter Communication. *IEEE Access* **2017**, *5*, 7254–7262. [CrossRef]
16. Hoang, D.T.; Niyato, D.; Wang, P.; Kim, D.I.; Le, L.B. Overlay RF-powered backscatter cognitive radio networks: A game theoretic approach. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017.
17. Lyu, B.; Guo, H.; Yang, Z.; Gui, G. Throughput Maximization for Hybrid Backscatter Assisted Cognitive Wireless Powered Radio Networks. *IEEE Internet Things J.* **2018**, *5*, 2015–2024. [CrossRef]
18. Wang, P.; Wang, N.; Dabaghchian, M.; Zeng, K.; Yan, Z. Optimal Resource Allocation for Secure Multi-User Wireless Powered Backscatter Communication with Artificial Noise. In Proceedings of the IEEE INFOCOM 2019—IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019; pp. 460–468.
19. Xiao, S.; Guo, H.; Liang, Y. Resource Allocation for Full-Duplex-Enabled Cognitive Backscatter Networks. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 3222–3235. [CrossRef]
20. Rahmati, A.; Dai, H. Reinforcement Learning for Interference Avoidance Game in RF-Powered Backscatter Communications. In Proceedings of the ICC 2019—2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019.
21. Hu, Y.; Wang, P.; Lin, Z.; Ding, M.; Liang, Y. Machine Learning Based Signal Detection for Ambient Backscatter Communications. In Proceedings of the ICC 2019—2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019.
22. Van Huynh, N.; Hoang, D.T.; Nguyen, D.N.; Dutkiewicz, E.; Niyato, D.; Wang, P. Optimal and Low-Complexity Dynamic Spectrum Access for RF-Powered Ambient Backscatter System with Online Reinforcement Learning. *IEEE Trans. Commun.* **2019**, *67*, 5736–5752. [CrossRef]
23. Wen, X.; Bi, S.; Lin, X.; Yuan, L.; Wang, J. Throughput Maximization for Ambient Backscatter Communication: A Reinforcement Learning Approach. In Proceedings of the 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 15–17 March 2019; pp. 997–1003.

24. Xie, Y.; Xu, Z.; Zhong, Y.; Xu, J.; Gong, S.; Wang, Y. Backscatter-Assisted Computation Offloading for Energy Harvesting IoT Devices via Policy-based Deep Reinforcement Learning. In Proceedings of the 2019 IEEE/CIC International Conference on Communications Workshops in China (ICCC Workshops), Changchun, China, 11–13 August 2019; pp. 65–70.
25. Lyu, B.; You, C.; Yang, Z.; Gui, G. The Optimal Control Policy for RF-Powered Backscatter Communication Networks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 2804–2808. [CrossRef]
26. Nguyen, N.; Van Huynh, N.; Hoang, D.T.; Nguyen, D.N.; Nguyen, N.; Nguyen, Q.; Dutkiewicz, E. Energy Management and Time Scheduling for Heterogeneous IoT Wireless-Powered Backscatter Networks. In Proceedings of the ICC 2019—2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019.
27. Liu, V.; Parks, A.; Talla, V.; Gollakota, S.; Wetherall, D.; Smith, J.R. Ambient backscatter: Wireless communication out of thin air. *ACM SIGCOMM Comput. Commun. Rev.* **2013**, *43*, 39–50. [CrossRef]
28. Psomas, C.; Krikidis, I. Collision avoidance in wireless powered sensor networks with backscatter communications. In Proceedings of the 2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Sapporo, Japan, 3–6 July 2017.