Look-Ahead and Learning Approaches for Energy Harvesting Communications Systems

Ala'eddin Masadeh, Zhengdao Wang, *Fellow, IEEE*, and Ahmed E. Kamal, *Fellow, IEEE*

***Abstract*—This work investigates the performance of an energy harvesting communications system. This system consists of a transmitter and a receiver. The transmitter is equipped with an infinite buffer to store data, and energy harvesting capability to harvest renewable energy and store it in a finite battery. The goal is to maximize the expected cumulative throughput of such systems. The problem of finding an optimal power allocation policy is formulated as a Markov decision process. Two cases are considered based on the availability of statistical knowledge about the channel gain and energy harvesting processes. When this knowledge is available, an algorithm is designed to maximize the expected throughput, while reducing the complexity of traditional methods (e.g., value iteration). This algorithm exploits instant knowledge about the channel, harvested energy, and current battery level to find a near-optimal policy. For the second scenario, when the statistical knowledge is unavailable, reinforcement learning is used. Two different exploration algorithms, convergence-based and the epsilon-greedy algorithms, are used. Simulations and comparisons with conventional algorithms show the effectiveness of the look-ahead algorithm when the statistical knowledge is available, and the effectiveness of reinforcement learning in optimizing the system performance when this knowledge is unavailable.**

***Index Terms*—Energy harvesting communications, Markov decision process, reinforcement learning, exploration, exploitation.**

I. Introduction

ENERGY harvesting (EH) converts ambient energy to electric energy. It has emerged as an efficient solution for providing sustainable energy for certain systems, such as wireless communications systems [1]. EH devices are characterized by several attractive attributes, such as the ability to be deployed in hard-to-reach areas, and offering reduced carbon emissions [2].

To implement an efficient EH communications system, two important, but contradictory goals should be achieved, which are prolonging the system's lifetime and maximizing its throughput. This can be accomplished by optimizing the use of available resources, which is considered the main challenge facing EH communications [3]. This is due to the variation of the amount of energy that can be harvested over time [1]. To overcome this challenge, it is important to design power allocation policies that adapt to time-variant EH and the channel fading processes.

Designing power allocation policies depends on the available knowledge at the EH node about the environment (i.e., channel fading and EH processes). This available knowledge can be classified into three groups. The first one is the non-causal knowledge. This assumption insures an optimal allocation policy. The second group is the statistical knowledge, where the EH and the channel fading processes are stationary random processes. The last group is the causal knowledge, which is the most realistic one. This means that at every time slot, EH nodes have only information about the current and past harvested energy and channel gains [3].

A. Energy Harvesting Communications Systems With Non-Causal Knowledge

EH communications systems with non-causal knowledge have been widely discussed [4]–[9]. Management approaches in this case are called offline approaches, where the amounts of harvested energy and their arrival times are known at the beginning of the communication session [10]. Despite the difficulty of considering this assumption in reality, it is used to find the upper bound performance [4].

In [4], the problem of maximizing the throughput of EH single hop communication system with non-causal knowledge is investigated. The authors prove that this problem can be modeled as the minimization of the time required for transmitting a fixed amount of data. The problem of identifying the offline transmission policy for EH communications system with cooperative relay is discussed in [5]. The goal is to maximize the amount of data received by the destination within a given time interval. In the proposed model, both the transmitter and the relay are EH nodes. The model is investigated under two scenarios, which are half-duplex and full-duplex relaying for communications. In [9], the problem of finding the offline transmission power allocation for EH communications system with multiple half-duplex relays is studied. The problem is formulated as a convex optimization problem to find the optimal power allocation for the goal of maximizing the amount of delivered data by a deadline.

Manuscript received April 3, 2019; revised September 4, 2019; accepted November 4, 2019. Date of publication November 14, 2019; date of current version March 18, 2020. This work was supported in part by NSF under Award 1711922 and under Award 1827211. The associate editor coordinating the review of this article and approving it for publication was E. Ayanoglu. *(Corresponding author: Ala'eddin Masadeh.)*

A. Masadeh is with the Electrical and Electronics Engineering Department, Al-Balqa Applied University, Al-Salt 19117, Jordan (e-mail: amasadeh@bau.edu.jo).

Z. Wang and A. E. Kamal are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: zhengdao@iastate.edu; kamal@iastate.edu).

Digital Object Identifier 10.1109/TGCN.2019.2953644

B. Energy Harvesting Communications Systems With Statistical Knowledge

For modeling more realistic EH communications systems, it is assumed that the EH, and data generation processes are discrete Markov processes with full statistical knowledge [11], [12], [13]. A Markov decision process (MDP) is characterized by its ability to provide a suitable mathematical framework for modeling decision-making problems when the system has processes that follow the Markov property [12]. Due to its efficiency, MDP has been adopted to deal with a number of problems considering EH [11], [12], [13], [14]. In [12], a point-to-point wireless communication system is studied, where the transmitter is able to harvest energy and store it in a rechargeable battery. The goal is to maximize the expected total transmitted data during the activation time of the transmitter. The problem is formulated as a discounted MDP problem. The state space consists of the battery state, the size of the packet to be transmitted, the current channel state, and the amount of energy needed for transmitting this packet successfully. At the beginning of each time slot, the transmitter makes a binary decision, whether to drop or to transmit the packet based on the current conditions. In this work, policy iteration (PI) is employed to solve the problem. In [14], an CRN with a secondary user that is capable of harvesting RF energy is investigated. In this model, the secondary user cannot execute EH and data transmission simultaneously, since it has only one interface. As a result, at the beginning of each time slot, the secondary user has to select either harvesting or transmitting. The mode management problem is formulated as an MDP. The primary channel is modeled as a three-state Markov chain, these states are occupied, idle with bad quality, and idle with good quality. The state space of the modeled MDP is a combination of the primary channel states and the secondary user energy levels. The action space consists of two actions, which are to harvest or to transmit. Value iteration (VI) is used to find the optimal policy, and the performance is compared with the greedy policy.

While traditional methods such as VI are able to find the optimal policy for MDP problems [15], [16], the complexity to find optimal solution grows as the number of states and actions increases. The complexity of finding the optimal solution using VI is $\mathcal{O}(|\mathcal{A}|.|\mathcal{S}|^2)$, where $\mathcal{A}$ is the set of actions, $\mathcal{S}$ is the set of states for the problem [17]. This has encouraged finding alternative methods for solving MDP problems, especially in the case of having large numbers of states and actions [11], [14], [18], [19]. The authors in [11] consider a network of objects equipped with energy-harvesting active networked tags (EnHANTs). The goal is to design an optimal transmission strategy for the EnHANTs to adapt to changes in the amounts of harvested energy and the identification request. The problem is formulated as an MDP. Modified policy iteration (MPI) method [20] is employed to solve the problem and to overcome the complixity of exhaustive search. In [19], the idea of using a mobile energy gateway is investigated, which has the capability of receiving energy from a fixed charging facility, as well as transferring energy to other users. The goal is to maximize the utility by taking the optimal action of energy charging/transferring. The problem is formulated as an MDP. The authors prove that there is a threshold structure of the optimal policy with respect to the system states, which helps in obtaining the optimal policy especially for MDPs with large numbers of states. The goal of determining these thresholds is to select immediate optimal actions based on the current state instead of using the traditional methods such as VI.

C. Energy Harvesting Communications Systems With Causal Knowledge

In the previous two frameworks, a priori knowledge, either deterministic or statistical, about the EH process is required. However, in more practical scenarios, this knowledge might be unavailable, in which case reinforcement learning (RL) can be used to improve the performance of such systems [10]. RL enables an autonomous agent to select optimized actions at different states in an unknown environment [21], [22].

In [3], [10], [23], [24] the problem of optimizing EH communications systems is investigated using RL. In this context, at any time, the EH nodes have only current local knowledge of the EH process. The authors aim to find a power allocation policy that maximizes the throughput. In these two works, the RL algorithm, which is known as the state-action-reward-state-action (SARSA), is used to evaluate the taken actions. On the other hand, the ϵ-greedy exploration algorithm is used to balance between exploring and exploiting available actions.

In [12], a point-to-point communications system is investigated. The transmitter is capable of harvesting energy and storing it in a rechargeable battery. The energy and data arrivals are formulated as Markov processes. In this work, the authors use Q-learning to find the optimal transmission policy when the system does not have a priori information about the Markov processes governing the system. They use the ϵ-greedy exploration algorithm to balance between exploration and exploitation.

D. The Contribution

In this paper, our goal is to provide efficient algorithms for dealing with two scenarios. The first scenario considers the availability of the statistical knowledge about the underlying MDP model, while the second one assumes unavailability of this knowledge. For the first scenario, we designed an algorithm utilizing the available statistical knowledge to optimize the system performance. The goal of this algorithm is to maximize the expected throughput while avoiding the complexity of traditional methods such as VI and PI.

In the second scenario RL is used. SARSA is used as a learning algorithm, and two exploration algorithms are used to balance between exploration and exploitation. The first algorithm is developed in this paper and called convergence-based algorithm. The second exploration algorithm is the widely

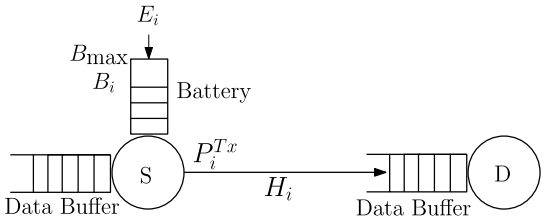


Fig. 1. Point-to-point communication system with an energy harvesting source.

used algorithm, which is called ϵ-greedy algorithm [21]. The convergence-based algorithm tries to balance between exploration and exploitation using two parameters, which are the exploration time threshold τ, and the action-value function convergence error ζ. In the first session of learning, the agent tries to evaluate available actions, and then it exploits the best resulting policy during the remaining time. On the other hand, ϵ-greedy tries to find a balance point between exploration and exploitation through the exploration probability ϵ. The motivation behind introducing convergence-based algorithm is to provide a parameter evaluating actions accurately using ζ, which is unavailable in ϵ-greedy algorithm. ζ enables systems to make correct decisions accurately, and increase the cumulative discounted return.

Finally, the performance of the proposed algorithms is compared with some algorithms, and shows the superiority of the proposed algorithms with their competitors.

E. The Paper Organization

The remainder of the paper is organized as follows. Section II describes the proposed communications system model. Then, the problem is formulated in Section III-A. The problem is reformulated as an MDP in Section III-B. Section IV presents the look-ahead algorithm, which is used when the dynamics of the underlying model are available. Section V discusses two exploration algorithms for RL to optimize the system performance when the knowledge about the underlying model is unavailable. Numerical simulation results are presented in Section VII. Finally, the paper is concluded in Section VIII.

II. System Model

In this section, a point-to-point communication system that consists of a source (S) and a destination (D) is considered. As illustrated in Fig. 1, each of S and D is equipped with an infinite buffer to store data. S has the capability of harvesting solar energy and storing it in a finite battery. We consider a time slotted system with equal length time slots, where each slot has a duration of T_c. For this system, the energy can be harvested and stored as an integer multiple of a fundamental unit. Let the maximum capacity of the battery be $B_{\max}$. B_i represents the battery level of S at the beginning of time slot *i*, where $B_i \in \mathcal{B} \triangleq \{b_1, b_2, \ldots, b_{N_b}\}$, N_b is the number of elements in $\mathcal{B}$, and $b_{N_b} = B_{\max}$.

The energy harvesting and channel gain processes are modeled as two independent Markov chains. Based on the current technologies, the amount of energy to be harvested E_i can be computed precisely [25]. During time slot *i*, the source harvests E_i units from solar sources, where $E_i \in \mathcal{E}_n \triangleq \{e_1, e_2, \ldots, e_{N_E}\}$, and N_E represents the number of elements in $\mathcal{E}_n$. $p_{\mathcal{E}_n}(e'|e)$ is the transition probability of harvested energy going from state *e* to state e' during one step transition. Let H_i be the channel state during time slot *i*, where $H_i \in \mathcal{H} \triangleq \{h_1, h_2, \ldots, h_{N_H}\}$, and N_H is the number of elements in $\mathcal{H}$. $p_{\mathcal{H}}(h'|h)$ is the transition probability for the channel going from state *h* to state h' during one time slot.

Let P_i^{Tx} be the transmission power during time slot *i*, and T_c is the transmission duration, which is constant during all time slots and equals 1 second. Since the source has causal knowledge about its environment, P_i^{Tx} is a function of E_i, B_i, and H_i. $P_i^{Tx} \in \mathcal{P}^{Tx} \triangleq \{p_1^{Tx}, p_2^{Tx}, \ldots, p_{N_p}^{Tx}\}$, where N_p is the number of elements in $\mathcal{P}^{Tx}$. In the proposed scheme, selecting P_i^{Tx} fulfills the Markov property, so the problem of optimizing the transmission power can be modeled as an MDP [21]. In this model, energy consumption is considered only due to data transmission, and it does not take into account any other energy consumption, such as processing, circuitry, etc.

The received signal at the destination D during the *i*th time slot is given as

$$y_i = \sqrt{P_i^{Tx}} H_i x_i + n_i, \qquad i = 1, \ldots, M \quad (1)$$

where x_i is the transmitted signal by S, respectively. n_i is additive Gaussian noise with zero-mean and noise variance σ_n^2.

In this context, the harvested energy is managed using harvest-store-use scheme. Using this scheme, harvested energy is stored partially or totally in a battery before using it. This scheme is characterized by its suitability for systems equipped with energy storage devices. It enables these systems to improve their performance by storing the harvested energy and using it when the channel gains are relatively good [26], [27].

III. Problem Formulation

A. Throughput Maximization Problem

In this section, we formulate the problem of maximizing the throughput by optimizing the transmission power over an infinite horizon. Two scenarios are taken into account. The first one considers the existence of statistical knowledge about the EH and channel gain processes, while the other considers the case of having only causal knowledge about these processes.

Due to lack of information about the harvestable energy and the channel gains in the future, the goal is to maximize the expected discounted return, where the discounted return following time *t*, G_t, is given by

$$G_t = \sum_{i=t}^{T-1} \gamma^{i-t} R_{i+1} \quad (2)$$

where t is the starting time for collecting a sequence of rewards, T is a final time step of an episode, and $\gamma \in (0,1)$ is the discount factor, which is used to weight the value (i.e., the importance) of the received data over time. It is a measure for the importance of transmitting data at the current time compared to transmitting the same data in the future, when it might not be important to the destination. R_{i+1} is the reward (i.e., the amount of received data) at time $i+1$ resulting from transmission using P_i^{Tx}

$$R_{i+1} = T_c \log_2\left(1 + \frac{P_i^{Tx}|H_i|^2}{\sigma_n^2}\right) \tag{3}$$

where σ_n^2 is the noise variance.

The energy causality constraints at the source, which is to ensure that the source cannot use more energy than its current battery level, and is given by

$$T_c\, P_i^{Tx} \le B_i, \quad i = t, \ldots, T-1 \tag{4}$$

Battery overflow constraint for the source, which is a rule for updating the energy level in the source's battery. It is a function of the battery level, transmission energy, harvested energy during time slot *i*, which is given by

$$B_{i+1} = \min\left\{B_i + E_i - T_c\, P_i^{Tx}, B_{\max}\right\}, \ i = t, \ldots, T-1 \tag{5}$$

Finally, P_i^{Tx}, and B_i should satisfy the following constraints

$$P_i^{Tx}, B_i \ge 0, \quad i = t, \ldots, T-1 \tag{6}$$

$$B_i \ge 0, \quad i = t, \ldots, T-1 \tag{7}$$

The optimization problem that maximizes the expected discounted return over an infinite horizon can now be formulated as

$$\max_{\{P_i^{Tx}\}} \lim_{T\to\infty} \mathbb{E}[G_t] \tag{8}$$

such that for $i = t, \ldots, T-1$,

$$\begin{aligned} P_i^{Tx}\, T_c &\le B_i, \\ B_{i+1} &= \min\{B_i + E_i - T_c\, P_i^{Tx}, B_{\max}\}, \\ P_i^{Tx} &\ge 0, \\ B_i &\ge 0. \end{aligned} \tag{9}$$

B. MDP Reformulation

Since the exact values of the harvested energy levels and channel gains are unknown in the future, this problem cannot be solved using convex optimization techniques although the problem is convex.

MDP is characterized by its ability to provide a framework for decision making problems, where outcomes are partly random and partly under control. The mathematical model of an MDP is defined by the following principles: (a) A set of states $\mathcal{S}$. (b) A set of actions $\mathcal{A}$. (c) The transition probability model $p(s'|s,a)$, which is the probability of reaching state $s' \in \mathcal{S}$ given that action $a \in \mathcal{A}$ is taken at state $s \in \mathcal{S}$. (d) The immediate reward, $r(s,a,s')$, yielded by taking action a at state s and then transiting to state s' [21].

The problem in (8) is reformulated as an MDP [28], where each state s is defined by three elements, which are the battery level, channel gain, and amount of harvested energy (i.e., $s = (b, h, e)$). The action a is defined as the selected transmission power p^{Tx}. Each state s has a subset of actions $\mathcal{P}_s^{Tx}$ such that $\mathcal{P}_s^{Tx} \in \mathcal{P}^{Tx}$. Battery levels evolve according to

$$b' = \min\{b + e - T_c\, p^{Tx}, B_{\max}\} \tag{10}$$

The transition probability $p(s'|s, p^{Tx})$ is given by

$$p\left(s'|s,p^{Tx}\right) = \begin{cases} p_{\mathcal{E}_n}(e'|e)\cdot p_{\mathcal{H}}(h'|h), & \text{if (10) is satisfied} \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

where the channel gain and EH processes are independent.

The immediate reward, which is the amount of received data resulting from taking action p^{Tx} at state s is given by

$$r\left(s,p^{Tx}\right) = T_c \log_2\left(1 + \frac{p^{Tx}|h|^2}{\sigma_n^2}\right) \tag{12}$$

In the proposed system, the immediate reward is a function of the current state s and the selected action p^{Tx} only, and it is independent of the next state s'. It is important to note the difference between (3) and (12). Equation (3) is the resulting data rate in terms of the state and action at time *i*, while (12) represents the resulting data rate in terms of the state and action spaces of the underlying MDP.

A deterministic policy π maps states into the transmission power taken at each state, $\pi(\cdot) : s \to p^{Tx}, \forall s$. The objective function is to maximize the expected cumulative throughput in (2) by finding an optimal policy π^*.

To evaluate different policies, value functions (state-value function $v_\pi(s)$ and action-value function $q_\pi(s,a)$) can be used. The optimal policy π^* has action-value function that is better than or equal to any other policy π for all states (i.e., $q_{\pi^*}(s,p^{Tx}) \ge q_\pi(s,p^{Tx}), \ \forall s \in \mathcal{S}$) [12].

In this work, two problems are studied. The first one is when the source has causal knowledge about the states (i.e., knowledge about past and current states), the available actions at the current state, the immediate reward given an action, and the transition probabilities between states. The second problem is the same as the first but when the transition probabilities are unavailable.

Two approaches are used to deal with the considered problems. The first one is the look-ahead algorithm for EH communications, which is proposed to solve the first problem. This algorithm utilizes the available statistical knowledge to maximize the objective function. It is designed to avoid the complexity of the available methods used for solving such problems, such as VI. The second approach is RL, where the transition probabilities between states are unavailable. Two exploration algorithms for RL are used to evaluate and improve the performance of the proposed system.

IV. Look-Ahead Policy for EH Communications (Known Underlying Model)

This proposed algorithm is a two-step look-ahead algorithm used when the statistical knowledge about the underlying model is available. This algorithm is broken down into a number of stages, as follows: Firstly, the overflow energy is computed. Then, the throughput using different transmission power levels are computed and compared. Finally, selecting a transmission power level based on the comparison from the previous step.

A. Two-Step Look-Ahead Throughput

The two-step look-ahead throughput is derived from the Bellman equation [21], and it is given by

$$\begin{aligned} R_1\left(s, p^{Tx}\right) &= r\left(s, p^{Tx}\right) + \gamma \sum_{s'} p\left(s'|s, p^{Tx'}\right) r\left(s', p^{Tx'}\right) \\ &= T_c \log_2\left(1 + \frac{p^{Tx}|h|^2}{\sigma_n^2}\right) + \gamma \sum_{s'} p\left(s'|s, p^{Tx'}\right) \\ &\quad \times T_c \log_2\left(1 + \frac{p^{Tx'}|h'|^2}{\sigma_n^2}\right) \end{aligned} \tag{13}$$

In this equation, the state value function *v* in the Bellman equation is replaced by the immediate reward *r* for one step only. This equation consists of two parts, the first one is the resulting throughput from using p^{Tx} in the current state *s*, while the second part is the expected throughput resulting from using $p^{Tx'}$ in the next slot s'.

B. Look-Ahead Throughput Algorithm

The overflow energy is defined as the amount of energy that could be lost due to reaching the battery's maximum capacity. This results from harvesting, not utilizing the available energy, and using a limited size battery. Overflow situations should be avoided since they are not optimal, where a higher throughput can always be achieved if the overflow energy is utilized.

Given b, e, and $B_{\max}$, the overflow energy is written as

$$e_{\text{ovf}} = \max\{b + \min\{e, B_{\max}\} - B_{\max}, 0\} \tag{14}$$

In each time slot, the goal is to use at least this amount of energy regardless of the channel state. This is because this energy will be lost if it is not utilized.

The proposed algorithm depends on computing the two-step look-ahead throughput for different energy levels at each state. These energy levels are integer multiples of a fundamental energy unit. The first step is to find the set of all possible energy levels, Λ, that can be used at each state *s*. $\Lambda = \{\lambda_1, \ldots, \lambda_{N_\Lambda}\}$, where $\lambda_1 = e_{\text{ovf}}$, $\lambda_{N_\Lambda} = b$, and N_Λ is the total number of energy levels in Λ. Λ_M is a set of *M* random energy levels selected from Λ, $1 \leq M \leq N_\Lambda$. The optimal scenario is to consider all energy levels between the maximum available energy in the battery and the overflow energy at each state, i.e., $\Lambda_M = \Lambda$. When the number of energy levels within this range is relatively small compared with the system's capability, all levels should be considered. Otherwise, the number of levels M to be evaluated should be determined based on the system's capability. The two-step look-ahead throughput for each energy level in Λ_M is computed according to

$$\begin{aligned} R_1(s, (\lambda_m/T_c)) &= r(s, (\lambda_m/T_c)) \\ &\quad + \gamma \sum_{s'} p\big(s'|s, (b'/T_c)\big) r\big(s', (b'/T_c)\big), \ m = 1, \ldots, M \end{aligned} \tag{15}$$

where b' is the battery level at the next state s', which depends on the used energy at state *s* (i.e., λ_m).

Based on the different values of $R_1(s, (\lambda_m/T_c))$ at state *s*, the energy level is selected according to

$$\lambda_{\max} \leftarrow \arg\max_{\lambda_m} [R_1(s, (\lambda_m/T_c))], \quad m = 1, \ldots, M \tag{16}$$

where the selected action (i.e., the transmission power) at state *s* is $\lambda_{\max}/T_c$. Algorithm 1 summarizes the proposed algorithm.

Algorithm 1 Look-Ahead Throughput Algorithm

```
for each s ∈ S do
    Compute the expected overflow energy e_ovf.
    Find the set of all available energy levels at s, (i.e., Λ).
    Sample M random energy levels from Λ, and assign them to Λ_M.
    for each m ∈ M do
        Compute R_l(s, (λ_m/T_c)) using (15).
    end for
    λ_max ← arg max_{λ_m} [R_l(s, (λ_m/T_c))],      m=1,...,M.
    p^{Tx} ← λ_max/T_c.
    s ← s'.
end for
```

V. Reinforcement Learning for EH Communications (Unknown Underlying Model)

This section provides a solution for the second scenario, where RL is used to handle the challenge of knowledge unavailability about the channel gain and EH processes. SARSA learning algorithm is used to evaluate different actions. The performance of the proposed model is investigated using two different exploration algorithms, which are the convergence-based algorithm, and the ϵ-greedy algorithm.

A. RL Prediction Methods

In this work, SARSA and Q-learning are used to predict the action-value function for different state-action pairs. SARSA is an on-policy updating strategy, which attempts to evaluate the policy that is used to make decisions. On the other hand, Q-learning is an off-policy method, where the action-value function is estimated for the policy that is unrelated to the policy used for evaluation [21].

Updating in SARSA works as follows. Starting from time slot *i*, let the agent be at state *s*, and the selected action according to the current policy π is *a*. Based on the selected action, it moves to the next state s' and receives a reward $r(s, a, s')$. Using a policy derived from the $Q(s, a)$ (e.g., ϵ-greedy algorithm), an action a' is selected to the next state s'. At this point, the estimate of the action-value function, $Q(s, a)$, is

updated using the gained experience. The updating equation in SARSA is given by [21]

$$Q(s,a) \leftarrow Q(s,a) + \alpha\big[r\big(s,a,s'\big) + \gamma Q(s',a') - Q(s,a)\big] \tag{17}$$

Using Q-learning, actions are assigned as follows. At the current state, actions are selected according to a policy derived from $Q(s, a)$ (e.g., ϵ-greedy algorithm), while the greedy action is assigned to the next state s'. The updating equation in Q-learning is given by [21]

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left[r(s,a,s') + \gamma \max_b Q(s',b) - Q(s,a)\right] \tag{18}$$

where $0 < \alpha < 1$ refers to the learning rate. This factor determines the amount of contribution of the newly acquired information for updating the action-value function. If $\alpha = 0$, then the agent will not learn any thing from the acquired information. On the other hand, if $\alpha = 1$, the agent will only consider the newly acquired information [29].

B. RL Exploration Algorithms

This part discusses two exploration algorithms for RL to deal with the case of knowledge unavailability about the underlaying model. The exploration algorithms play an essential role in RL. Their role appears in finding a balance between exploration and exploitation to maximize the cumulative rewards. The exploitation mode can be defined as using the current available knowledge to select the best policy to be used. On the other hand, exploration is known as investigating new policies in the hope of getting policy that is better than the current best one [21].

1) The ϵ-Greedy Algorithm: This algorithm [30] uses the exploration probability ϵ to find a balancing point between exploration and exploitation modes. This parameter changes the mode based on its value at each time slot.

In this algorithm, the current best action is selected with probability $1 - \epsilon$. On the other hand, a random non-greedy action is selected with probability ϵ. The ϵ can be either fixed [21], or with adaptive value during the learning time [10]. In the case of adaptive ϵ-greedy, ϵ takes values that changes with time. For example, in [10], ϵ is set to $e^{-0.1i}$, where i the time slot number. In this case, at the beginning of the session, the exploration probability ϵ has large values to increase the probability of exploration. As time increases, the probability of exploration decreases and the exploitation probability increases. This is to increase the opportunity of exploitation at the end of the session, where most of the policies have been explored and it is preferred to exploit the best known policy.

2) The Convergence-Based Algorithm: This part presents our exploration algorithm. It uses two parameters to balance between exploration and exploitation. The first parameter is the action-value function convergence error ζ. The same action at a state is exploited for a number of iterations until the estimated value of this state-action pair converges to a value with an error less than or equal to ζ. The second parameter is the exploration time threshold τ. This parameter controls the exploration process, where the agent can explore different actions for a τ from the total available time T, after that, the agent is forced to exploit the best available policy π_{best} during the remaining time [31], [32].

In this algorithm, the first step is to assign random feasible actions to all available states. Then, for each visited state, the same action is selected for a time until its estimated value converges to a value determined by ζ. Once the estimated value of a state-action pair converges to a value with an error less than or equal to ζ, a new random action is assigned from uniformly distributed unexplored actions to that state. This mechanism continues for all states, and stops in two cases: The first one occurs if all available actions for a states s are evaluated before reaching τ. At this time, the action with the best value $\pi_{\text{best}}(s)$ will be exploited in the future. The second case occurs when the available time reaches τ. Then, the agent suspends exploration, and starts exploiting the best available policy π_{best} regardless of exploring all available actions or not.

Using the SARSA with the convergence-based algorithm, an action for next state s' is selected according to the current policy π

$$p^{Tx'} \leftarrow \pi(s') \tag{19}$$

and for the case of integrating the Q-learning and convergence-based algorithms, an action is assigned to next state s' according to

$$p^{Tx'} \leftarrow \arg\max_a Q(s',a) \tag{20}$$

Convergence-based algorithm is characterized by a number of properties that makes it a good candidate compared with the epsilon-greedy algorithm. These properties are summarized as follows. Firstly, using convergence-based algorithm, once an action at a state has been evaluated, and its action-value function has converged to an unfavorable value, this action will not be exploited in the future. This is an important property that contributes to discarding actions that may reduce the cumulative discounted return in the future. Secondly, the convergence-based algorithm starts by evaluating most of the available actions using ζ, which enables systems to evaluate different policies accurately. This enables systems to determine and exploit policies with actual high return. On the other hand, this parameter is unavailable in epsilon-greedy algorithm, which exploits greedy actions based on their instant values. Thirdly, using the convergence error ζ, once all available actions are explored and evaluated (when the numbers of states and actions are finite and relatively small), and a suboptimal policy is determined, the exploration process is terminated. This enables systems to exploit the best resulting policy at an early time, which maximizes the cumulative discounted return significantly, where the effect of the discount factor γ is small. However, this feature is unavailable in the epsilon-greedy algorithm. Algorithm 2 summarizes the proposed algorithm.

Algorithm 2 Convergence-Based Algorithm for estimating π^*

Initialize $Q^0(s, p^{Tx})$, $\forall s \in \mathcal{S}$, $\forall p^{Tx} \in \mathcal{P}_s^{Tx}$, arbitrarily
Initialize the action-value convergence error ζ, the exploration time threshold τ, and the learning rate α
Initialize $Q_{\text{best}}(s) = -\infty$, $\forall s \in S$
Initialize the policy π and the current best policy π_{best} by random actions $\varrho \in \mathcal{P}_s^{Tx}$, $\forall s \in \mathcal{S}$
$\pi_{\text{best}}(s), \pi(s) \leftarrow \varrho, \quad \forall s \in \mathcal{S}$
$\mathcal{P}_s^{Tx} \leftarrow \mathcal{P}_s^{Tx} - \varrho\ , \quad \forall s \in \mathcal{S}$
for each step i of episode **do**
  Observe current state S
  Select action P^{Tx} to state S according to the policy π (i.e., $P^{Tx} \leftarrow \pi(S)$)
  Observe the immediate reward $r(S, P^{Tx})$, and next state S'
  Predict $Q(S, P^{Tx})$ using a prediction method (e.g., SARSA or Q-learning)
  if $|Q^i(S, P^{Tx}) - Q^{i-1}(S, P^{Tx})| \leq \zeta$ AND $i < \tau$ **then**
    if $Q^i(S, P^{Tx}) \geq Q^{\text{best}}(S)$ **then**
      $Q_{\text{best}}(S) \leftarrow Q^i(S, P^{Tx})$
      $\pi_{\text{best}}(S) \leftarrow P^{Tx}$
    end if
    if $\mathcal{P}_S^{Tx} \neq \phi$ **then**
      Update π by selecting a new random action $\varrho \in \mathcal{P}_S^{Tx}$ to state S
      $\pi(S) \leftarrow \varrho$
      $\mathcal{P}_S^{Tx} \leftarrow \mathcal{P}_S^{Tx} - \varrho$
    else
      $\pi(S) \leftarrow \pi_{\text{best}}(S)$
    end if
  else if $i \geq \tau$ **then**
    $\pi \leftarrow \pi_{\text{best}}$
  end if
  $S \leftarrow S'$
end for

VI. Complexity

Algorithm 1 aims at reducing the complexity of solving the formulated MDP problem, while approaching the optimal performance. Using the proposed algorithm, there is no need to go through all possible policies and select the optimal one, which is difficult especially when the system has a large number of actions/states combinations. For the case of using value iteration to get the optimal solution, the complexity is $\mathcal{O}(|\mathcal{A}| \cdot |\mathcal{S}|^2)$, where $\mathcal{A}$ is the set of actions, and $\mathcal{S}$ is the set of states for the problem [17]. On the other hand, the proposed algorithm has a complexity of $\mathcal{O}(|\mathcal{S}| \cdot |M|)$, where M is the number of sampled energy levels that are evaluated at each state.

For Algorithm 2, it aims at providing an efficient exploration algorithm for RL to improve the learning performance. This algorithm tries to estimate the values of different state-action pairs accurately, and then, exploit the best resulting policy. Let T is the final time step of an episode. The complexity of Algorithm 2 is $\mathcal{O}(|T|)$ when SARSA is used as a prediction method, and $\mathcal{O}(|\mathcal{A}| \cdot |T|)$ when Q-learning is used.

VII. Simulation Results

In this section, the proposed algorithms are evaluated. Then, the effects of their parameters are investigated. To evaluate the proposed algorithms, three additional approaches are considered.

- Value iteration (VI) [16].
- Hasty Policy: At each time slot, all available energy is allocated for data transmission, regardless of previous experience. The goal is to avoid energy overflow situations [10].
- Random Policy: In this case, a set of feasible random transmission power levels is considered, where all levels are uniformly distributed across their range [10].

Two types of scenarios were considered in the simulation, simple scenarios that consider small numbers of states and actions, and scenarios with large numbers of states and actions. For simple scenarios such as in [12], [33], where the optimal policy can be found easily, VI was used to evaluate the performance of the proposed algorithms. On the other hand, the proposed algorithms are compared with the hasty and random approaches only in the case of considering large number of states.

In the numerical analysis, it is assumed that each time slot is 1 second in duration. The available bandwidth BW is 1 MHz, and the noise spectral density is $N_0 = 4 \times 10^{-21}$ W/Hz.

It is also assumed that the S is equipped with solar panels with an area of 25 cm^2 and 10% harvesting efficiency. An outdoor solar panel can get the benefit of 100 mW/cm^2 solar irradiance under standard testing conditions, and harvesting efficiency between 5% and 30%, based on the used material in the panel [34]. It is assumed that the fundamental energy unit that can be harvested, stored, and transmitted is 0.05 J.

The used parameters were set as follows. The discount factor γ is set to 0.9, and the learning rate α is selected to be 0.1. Adaptive ϵ-greedy exploration algorithm is used [10]. For this algorithm, the exploration probability is set to $\epsilon = e^{-0.001i}$, where i is the time slot number in an episode. For the convergence-based exploration algorithm, ζ is set to 4, and the τ is set to be 0.8 of the total available time in an episode (i.e., $\tau = 0.8\ T$). For the throughput comparison step in the look-ahead algorithm, all possible energy levels at each state are considered, i.e., $\Lambda_M = \Lambda$.

It is also assumed that the set of harvested energy levels is $\mathcal{E}_n = \{0, 0.05, 0.1, 0.15, 0.2, 0.25\}$ J with transition probability matrix P_e

$$P_e = \begin{bmatrix} 0.4011 & 0.3673 & 0.1027 & 0.0899 & 0.0279 & 0.0111 \\ 0.4072 & 0.3441 & 0.1002 & 0.0973 & 0.0305 & 0.0207 \\ 0.3966 & 0.3239 & 0.1165 & 0.0860 & 0.0400 & 0.0370 \\ 0.3796 & 0.3272 & 0.1158 & 0.0782 & 0.0514 & 0.0478 \\ 0.3612 & 0.3451 & 0.1055 & 0.0837 & 0.0501 & 0.0544 \\ 0.3711 & 0.3341 & 0.1107 & 0.0801 & 0.0502 & 0.0538 \end{bmatrix}$$

The set of channel gains consists of 11 states that were uniformly selected between 0 and -20 dB with random transition probabilities between the states.

The used battery has a maximum capacity of 12 units. All results are averaged over 500 runs. The starting state is selected randomly, where all the states have equal probability to be the starting state. The convergence-based and ϵ-greedy algorithms starts learning from the same policy, which is the Hasty policy. All mentioned parameters were used in all experiments unless otherwise stated.

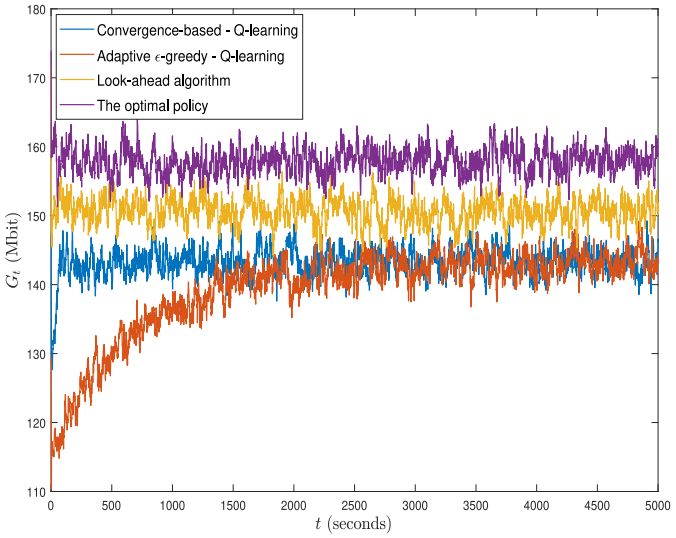


Fig. 2. The discounted return G_t versus time t for different approaches.

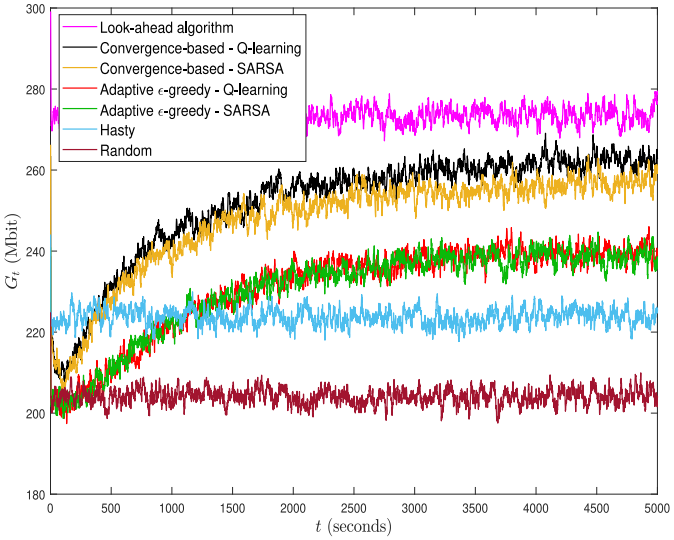


Fig. 3. The discounted return G_t versus time t for different approaches.

A. Comparison With the Upper Bound

In this part, we evaluated the proposed algorithms by comparing them with the optimal performance. The VI was used to find the optimal policy to get the upper bound performance. The VI along with the look-ahead algorithms need a priori statistical knowledge about the channel gain and EH processes. On the other hand, this knowledge is unavailable for the learning approaches.

In this scenario, the battery maximum capacity $B_{\max}$ is set to 2 units. The set of harvested energy is $\mathcal{E}_n = \{0, 0.05\}$ J with transition probability matrix P_e

$$P_e = \begin{bmatrix} 0.5050 & 0.4950 \\ 0.5215 & 0.4785 \end{bmatrix}$$

The set of channel gains $\mathcal{H}_n = \{0, -10, -20\}$ dB with transition probability matrix P_h

$$P_h = \begin{bmatrix} 0.3946 & 0.3991 & 0.2064 \\ 0.4145 & 0.3470 & 0.2385 \\ 0.5524 & 0.3637 & 0.0838 \end{bmatrix}$$

Fig. 2 shows the discounted return G_t (i.e., the cumulative discounted received data starting from time t). The cumulative discounted received data is defined as the amount of valuable data received within a given time frame. The discounted returns of the optimal policy and look-ahead algorithm take a near-constant pattern all the time. This is due to use one policy all the time, and the discount factor which bounds the discounted return to a value. For the learning approaches, in the beginning of the session, their discounted returns increase with experience, where these approaches start from hasty policy. As the time increases, they start taking a near-constant pattern, which results from learning an optimal policy that cannot be improved any more, and the discount factor that bounds the discounted return to a value.

As shown, the upper-bound on the discounted return can be achieved by exploiting the optimal policy all the time. This figure also shows that the look-ahead algorithm outperforms the remaining approaches, which is due to exploiting the statistical knowledge that is available to this approach. It can also be noticed that the convergence-based algorithm outperforms the adaptive ϵ-greedy, where the convergence-based algorithm finds an optimal policy faster than the adaptive ϵ-greedy. The superiority of the convergence-based algorithm is attributed to its approach in evaluating different actions. The convergence-based algorithm evaluates the available actions based on their convergent values. This gives the source relatively accurate indications about the values of different state-action pairs, and enables it to determine and select an optimal policy in a relatively high-precision pattern.

On the other hand, the ϵ-greedy algorithm evaluates actions based on the instant values of state-action pairs, especially in the beginning of the learning process, when the exploration probability is relatively high and the values of different state-action pairs are unable to converge. Unfortunately, these instant values might not be the actual or near actual values of these pairs, which may slow down finding an optimal policy with actual high discounted return.

B. Comparison in Large Scenario

This part considers the case of large number of states. The goal is to examine the validity of the proposed algorithms in the case of large scenarios, where the number of states used in this part is 858 states. Fig. 3 shows the performance of the proposed approaches compared with the hasty and random algorithms, where finding the optimal policy is difficult. The discounted returns of the look-ahead, hasty, and random algorithms take a near-constant pattern all the time. This is because of using one policy all the time, and the discount factor that bounds the discounted return to a certain value.

Fig. 3 shows that the look-ahead algorithm outperforms the other algorithms. This is due to having the statistical knowledge about the channel gain and EH processes, which enables the source to exploit an optimal policy from the beginning. It can also be noticed that the convergence-based algorithm

outperforms the ϵ-greedy algorithm in terms of the speed of finding an optimal policy, and the quality of learned policies by each algorithm. This superiority is due to the used approach by each algorithm for evaluating different actions as explained in the previous subsection. For the hasty and random approaches, they do not exploit the available causal knowledge in exploring and exploiting different policies, which explains the relatively poor performance of these two approaches.

Using the convergence-based algorithm, it is clear that the Q-learning outperforms the SARSA insignificantly. Q-learning learns $Q(s, a)$ by approximating the optimal action-value function q^* directly. Fortunately, approximating the optimal action-value function has improved the performance by finding an optimal policy in a shorter time compared to SARSA, even if this improvement is relatively small. On the other hand, SARSA is more conservative, it improves its performance using the estimate of the action-value function under the current policy. Although, SARSA uses a safer path, but this has slowed down finding an optimal policy and exploiting it early. Regarding to the adaptive ϵ-greedy, it can be seen that both Q-learning and SARSA have approximately the same performance, where approximating the optimal policy by Q-learning has not improved the performance.

C. RL Algorithms - Harvested Energy Levels With Equal Probabilities

This part considers another large scenario. It aims at investigating the considered RL exploration algorithms when the EH process is a process with independent and identically distributed random variables. The set of harvested energy levels is $\mathcal{E}_n = \{0, 0.05, 0.1, 0.15, 0.2, 0.25\}$ J, each with equal probability.

The considered exploration algorithms are compared using Q-learning. For the convergence-based algorithm, $\zeta = 4$ and τ has values that are changing between $0.2T$ and $0.6T$. The adaptive ϵ-greedy algorithm uses exploration probability changing between $\epsilon = \exp(0.1i)$ and $\epsilon = \exp(0.0001i)$.

Fig. 4 shows the superiority of the convergence-based algorithm over hasty and ϵ-greedy algorithms. It can also be noticed that the best performance of the adaptive ϵ-greedy approximates the performance of the hasty policy, while the hasty outperforms the adaptive ϵ-greedy for the remaining values of ϵ. The superiority of the convergence-based algorithm and the poor performance of the adaptive ϵ-greedy algorithm are explained in the previous subsections.

D. Effect of the τ in Convergence-Based Algorithm

This experiment investigates the effect of the exploration time threshold τ on the performance of the convergence-based algorithm. In this experiment, ζ is set to 4.

Fig. 5 shows the discounted return versus time. When $\tau = 0$, the performance takes a near constant pattern from the beginning, which is due to exploiting one policy all the time (i.e., there is no exploration). For the remaining values of τ, the discounted returns increase with experience. Then, they take near-constant shapes, which is due to the discount factor effect, and the inability to improve the policy any more.

Fig. 5 also shows that the discounted returns increase as τ increases up to a value, then saturation occurs. As the value of this threshold increases, the opportunity of exploring more policies increases, which also increases the opportunity of finding a good policy that increases the discounted return. After a certain time, the effect of increasing τ on the performance diminishes, which is due to assigning values for τ that are bigger than the required time for exploring all available actions. In this case, the source will be forced to exploit the best learned policy once it has evaluated all available actions regardless the value of τ.

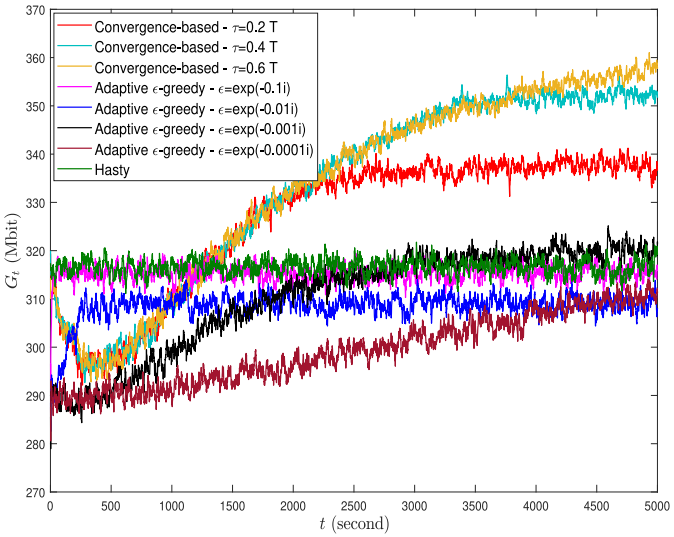


Fig. 4. The discounted return G_t versus time t for different RL exploration algorithms.

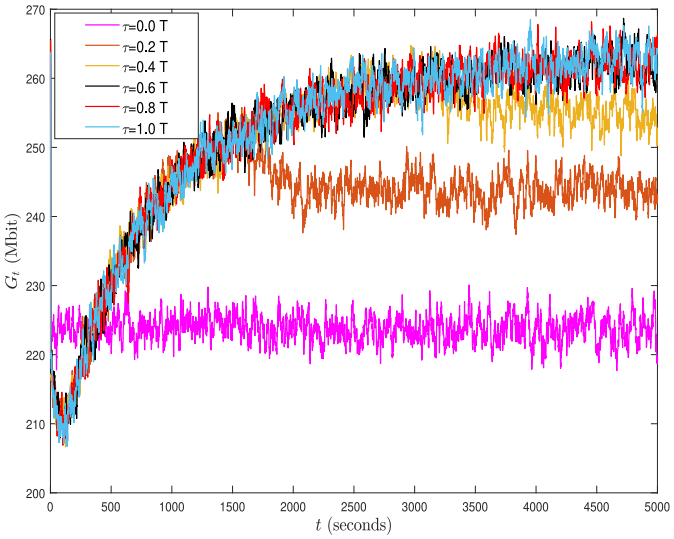


Fig. 5. The discounted return G_t versus time t for different values of the τ.

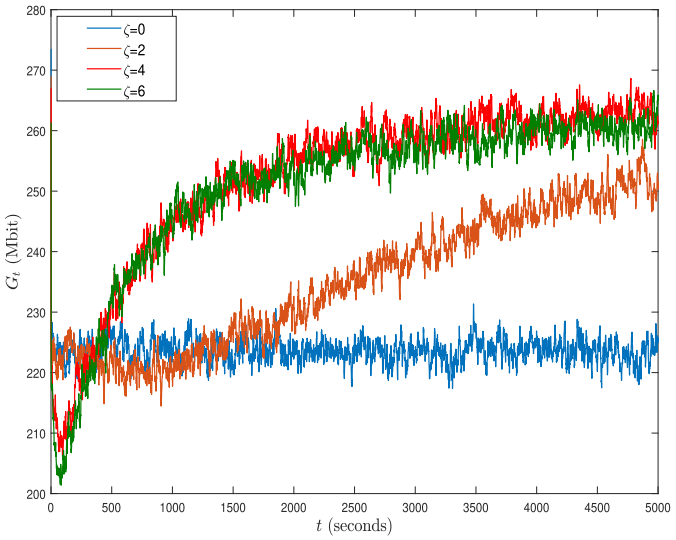


Fig. 6. The discounted return G_t versus time t for different values of the ζ.

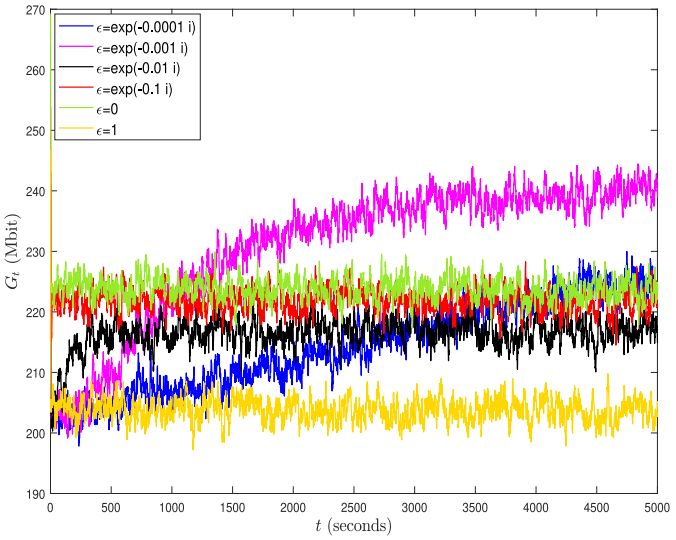


Fig. 7. The discounted return G_t versus time t for different values of the ϵ.

E. Effect of the ζ in Convergence-Based Algorithm

In this experiment, the effect of ζ on the performance of the convergence-based algorithm was studied. The value of τ is set to be $0.8T$.

Fig. 6 shows the influence of the experience on the discounted return at different values of ζ. As shown, for $\zeta = 0$, the performance has a near constant shape from the beginning, since there is no exploration. In this experiment it is difficult to achieve convergence with zero error, which prevents exploration. For the remaining values of ζ, the performance is improved with experience. Then, the discounted returns take near-constant patterns, since the source is unable to improve the policy any more, and the discount factor which bounds the return. This figure also shows that the best performance is achieved when ζ has a value of 4.

It can also be noticed that the discounted return increases as the convergence error increases up to a certain value, and then starts decreasing. This is due to the fact that increasing the convergence error increases the opportunity of exploration, which improves the performance up to a certain value of ζ. After that, the performance starts to degrade, which is due to inaccurate evaluation of various actions.

F. Effect of the ϵ in ϵ-Greedy Algorithm

This part discusses the effect of ϵ on the performance of the ϵ-greedy algorithm.

Fig. 7 investigates the performance of the adaptive ϵ-greedy algorithm using different scenarios ($\epsilon = 0$, $\epsilon = e^{-0.1i}$, $\epsilon = e^{-0.01i}$, $\epsilon = e^{-0.001i}$, $\epsilon = e^{-0.0001i}$, and $\epsilon = 1$), where i is the time slot number in an episode. This figure shows that the discounted return when $\epsilon = 0$ remains constant approximately from the beginning, since there is no exploration. For the remaining values of ϵ, the performance is enhanced with experience, then, the discounted returns maintain near-constant shapes due to the inability to improve the learned policy, and bounding the return by the discount factor. It can be noticed that the $\epsilon = e^{-0.001i}$ scenario outperforms the other scenarios.

This figure shows that slowing the decay of ϵ improves the performance up to a certain value, and then starts to degrade the performance. Decelerating decay of the ϵ means increasing the exploration probability at the beginning, which gives the source more opportunity to explore more policies and find a good policy. Increasing the exploration probability improves the performance up to a certain value, but then it starts to degrade the performance, which is due to slowing down exploiting the best learned policy from the exploration.

VIII. Conclusion

In this paper, two different scenarios for a realistic energy harvesting communication system were investigated. The first one assumes the availability of statistical knowledge about the channel gain and EH processes. On the other hand, the system in the second scenario does not have that knowledge. The source is equipped with an infinite data buffer to carry data packets and finite battery to store the harvested energy. We formulated the problem of maximizing the cumulative discounted received data as an MDP. For the first scenario, a look-ahead algorithm was designed to solve the problem efficiently by exploiting the availability of the transition probabilities between states. To optimize the performance of the system in the second scenario, RL was used. The results showed the effectiveness of the look-ahead algorithm when the statistical knowledge is available, even when the number of states and action is large. This work also showed the effectiveness of RL for optimizing the system performance in the case of unavailability of that knowledge. Two different exploration algorithms for RL were used, which are the convergence-based and ϵ-greedy algorithms. It was noticed that the convergence-based algorithm outperforms the other one. Finally, we discussed the effects of the parameters of

each algorithm on the system performance. As a future work, function approximation and neural networks can be used along with RL to consider the case of having continuous state and action spaces.

REFERENCES

[1] S. Ulukus *et al.*, "Energy harvesting wireless communications: A review of recent advances," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 3, pp. 360–381, Mar. 2015.

[2] O. Ozel, K. Tutuncuoglu, S. Ulukus, and A. Yener, "Fundamental limits of energy harvesting communications," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 126–132, Apr. 2015.

[3] A. Ortiz, H. Al-Shatri, X. Li, T. Weber, and A. Klein, "Reinforcement learning for energy harvesting decode-and-forward two-hop communications," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 3, pp. 309–319, Sep. 2017.

[4] K. Tutuncuoglu and A. Yener, "Optimum transmission policies for battery limited energy harvesting nodes," *IEEE Trans. Wireless Commun.*, vol. 11, no. 3, pp. 1180–1189, Mar. 2012.

[5] D. Gündüz and B. Devillers, "Two-hop communication with energy harvesting," in *Proc. IEEE Int. Workshop Comput. Adv. Multi Sensor Adaptive Process. (CAMSAP)*, San Juan, Puerto Rico, Dec. 2011, pp. 201–204.

[6] K. Tutuncuoglu and A. Yener, "Optimal power policy for energy harvesting transmitters with inefficient energy storage," in *Proc. Annu. Conf. Inf. Sci. Syst. (CISS)*, Princeton, NJ, USA, Mar. 2012, pp. 1–6.

[7] O. Orhan, D. Gündüz, and E. Erkip, "Throughput maximization for an energy harvesting communication system with processing cost," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Lausanne, Switzerland, Sep. 2012, pp. 84–88.

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

[9] O. Orhan and E. Erkip, "Energy harvesting two-hop communication networks," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2658–2670, Dec. 2015.

[10] 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.

[11] Z. Wang, A. Tajer, and X. Wang, "Communication of energy harvesting tags," *IEEE Trans. Commun.*, vol. 60, no. 4, pp. 1159–1166, Apr. 2012.

[12] P. Blasco, D. Gunduz, 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.

[13] H. Li, N. Jaggi, and B. Sikdar, "Relay scheduling for cooperative communications in sensor networks with energy harvesting," *IEEE Trans. Wireless Commun.*, vol. 10, no. 9, pp. 2918–2928, Sep. 2011.

[14] J. Zhao, Y. Wei, M. Song, and X. Wang, "Dynamic mode management in cognitive radio networks with RF energy harvesting," in *Proc. Int. Conf. Wireless Commun. Netw. Mobile Comput. (WiCOM)*, Shanghai, China, Sep. 2015, pp. 1–6.

[15] W. Li, M.-L. Ku, Y. Chen, and K. J. R. Liu, "On outage probability for two-way relay networks with stochastic energy harvesting," *IEEE Trans. Commun.*, vol. 64, no. 5, pp. 1901–1915, May 2016.

[16] T. Wang, C. Jiang, and Y. Ren, "Access points selection in super WiFi network powered by solar energy harvesting," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Doha, Qatar, Apr. 2016, pp. 1–5.

[17] M. L. Littman, T. L. Dean, and L. P. Kaelbling, "On the complexity of solving Markov decision problems," in *Proc. 11th Conf. Uncertainty Artif. Intell.*, Montreal, QC, Canada, Aug. 1995, pp. 394–402.

[18] V. S. Rao, R. V. Prasad, and I. G. M. M. Niemegeers, "Optimal task scheduling policy in energy harvesting wireless sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, New Orleans, LA, USA, Mar. 2015, pp. 1030–1035.

[19] Y. Zhang, D. Niyato, P. Wang, and D. I. Kim, "Optimal energy management policy of mobile energy gateway," *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 3685–3699, May 2016.

[20] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, NJ, USA: Wiley, 2005.

[21] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[22] C. Szepesvári, *Algorithms for Reinforcement Learning* (Synthesis Lectures on Artificial Intelligence and Machine Learning), vol. 4. San Rafael, CA, USA: Morgan & Claypool, 2010, pp. 1–103.

[23] F. A. Aoudia, M. Gautier, and O. Berder, "RLMan: An energy manager based on reinforcement learning for energy harvesting wireless sensor networks," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 2, pp. 408–417, Jun. 2018.

[24] H. Al-Tous and I. Barhumi, "Distributed reinforcement learning algorithm for energy harvesting sensor networks," in *Proc. IEEE Int. Black Sea Conf. Commun. Netw. (BlackSeaCom)*, Sochi, Russia, Jun. 2019, pp. 1–3.

[25] C. Bergonzini, B. Lee, J. R. Piorno, and T. S. Rosing, "Management of solar harvested energy in actuation-based and event-triggered systems," in *Proc. Energy Harvesting Workshop*, Roanoke, VA, USA, Jan. 2009.

[26] A. Alsharoa, H. Ghazzai, A. E. Kamal, and A. Kadri, "Optimization of a power splitting protocol for two-way multiple energy harvesting relay system," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 4, pp. 444–457, Dec. 2017.

[27] T. Li, P. Fan, Z. Chen, and K. B. Letaief, "Optimum transmission policies for energy harvesting sensor networks powered by a mobile control center," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6132–6145, Sep. 2016.

[28] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1. Belmont, MA, USA: Athena scientific, 1995.

[29] Y. Xu, W. Zhang, W. Liu, and F. Ferrese, "Multiagent-based reinforcement learning for optimal reactive power dispatch," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 6, pp. 1742–1751, Dec. 2012.

[30] Z. Xia and D. Zhao, "Online reinforcement learning by Bayesian inference," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Killarney, Ireland, Jul. 2015, pp. 1–6.

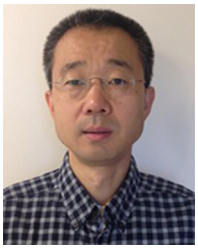
[31] A. Masadeh, Z. Wang, and A. E. Kamal, "Convergence-based exploration algorithm for reinforcement learning," Electr. Comput. Eng., Iowa State Univ. Digit. Repository, Ames, IA, USA, Rep. 1, pp. 1–11, 2018.

[32] A. Masadeh, Z. Wang, and A. E. Kamal, "Reinforcement learning exploration algorithms for energy harvesting communications systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018, pp. 1–6.

[33] W.-T. Lin, I.-W. Lai, and C.-H. Lee, "Distributed energy cooperation for energy harvesting nodes using reinforcement learning," in *Proc. IEEE Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Hong Kong, Aug./Sep. 2015, pp. 1584–1588.

[34] R. J. M. Vullers, R. V. Schaijk, H. J. Visser, J. Penders, and C. V. Hoof, "Energy harvesting for autonomous wireless sensor networks," *IEEE Solid-State Circuits Mag.*, vol. 2, no. 2, pp. 29–38, Jun. 2010.

Ala'eddin Masadeh received the B.Sc. and M.Sc. degrees in electrical engineering from the Jordan University of Science and Technology, Irbid, Jordan, in 2010 and 2013, respectively, and the Ph.D. degree in electrical engineering and computer engineering from Iowa State University, Ames, IA, USA, in 2019. He is currently an Assistant Professor with the Electrical and Electronics Engineering Department, Al-Balqa Applied University, Jordan. His research interests include wireless networks, energy harvesting communications, reinforcement learning, machine learning, and artificial intelligence.



Zhengdao Wang (S'00–M'02–SM'08–F'16) received the B.S. degree in electronic engineering and information science from the University of Science and Technology of China, Hefei, China, in 1996, the M.Sc. degree in electrical and computer engineering from the University of Virginia, Charlottesville, VA, USA, in 1999, and the Ph.D. degree in electrical and computer engineering from the University of Minnesota, Minneapolis, MN, USA, in 2002. He is currently with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA. His research interests include signal processing, communications, and information theory. From April 2004 to April 2006, he was an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE SIGNAL PROCESSING LETTERS from August 2005 to August 2008, and the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2013 to 2015. He is also an Editor of the IEEE Signal Processing Society Online Video Library and *ZTE Communications*, and an Associate Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.

Ahmed E. Kamal (S'82–M'87–SM'91–F'12) received the B.Sc. (Distinction with Hons.) and M.Sc. degrees in electrical engineering from Cairo University, Cairo, Egypt, in 1978 and 1980, respectively, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of Toronto, Toronto, ON, Canada, in 1982 and 1986, respectively. He is a Professor with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA. His research interests include wireless networks, cognitive radio networks, optical networks, wireless sensor networks, Internet of Things, and performance evaluation. He chaired or co-chaired Technical Program Committees of several IEEE sponsored conferences, including the Optical Networks and Systems Symposia of the IEEE Globecom 2007 and 2010, the Cognitive Radio and Networks Symposia of the IEEE Globecom 2012 and 2014, and the Access Systems and Networks track of the IEEE International Conference on Communications 2016. He is also the Chair of the IEEE Communications Society Technical Committee on Transmission, Access and Optical Systems for 2015 and 2016. He is on the editorial boards of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, *Computer Networks*, and the *Optical Switching and Networking*. He served as an IEEE Communications Society Distinguished Lecturer in 2013 and 2014. He is a Senior Member of the Association of Computing Machinery.