Decentralized Transmission Policies for Energy Harvesting Devices

Alessandro Biason*, Subhrakanti Dey† and Michele Zorzi*
* Department of Information Engineering, University of Padova - via Gradenigo 6b, 35131 Padova, Italy
† Department of Engineering Science, Uppsala University, Uppsala, Sweden
email: biasonal@dei.unipd.it, Subhrakanti.Dey@signal.uu.se, zorzi@dei.unipd.it

***Abstract*—The problem of finding decentralized transmission policies in a wireless communication network with energy harvesting constraints is formulated and solved using the decentralized Markov decision process framework. The proposed policy defines the transmission strategies of all devices so as to correctly balance the collision probabilities with the energy constraints. After an initial coordination phase, in which the network parameters are initialized for all devices, every node proceeds in a fully decentralized fashion. We numerically show that, unlike in the case without energy constraints where a fully orthogonal scheme can be shown to be optimal, in the presence of energy harvesting this is no longer the best choice, and the optimal strategy lies between an orthogonal and a completely symmetric system.**

I. Introduction

Energy Harvesting (EH) has been established as one of the most prominent solutions for prolonging the lifetime and enhancing the performance of Wireless Sensor Networks (WSNs). Although this topic has been widely investigated in the literature so far, finding proper energy management schemes is still an open issue in many cases of interest. In particular, using *decentralized* policies, in which every node in the network acts autonomously and independently of the others is a major problem of practical interest in WSNs where a central controller may not be used all the time. Many decentralized communication schemes (e.g., Aloha-like) can be found in the literature; however, most of them were designed without a principle of *optimality*, i.e., without explicitly trying to maximize the network performance. Instead, in this work we characterize the optimal decentralized policy in a WSN with EH constraints and describe the related computational issues. Although this approach intrinsically leads to a more complex protocol definition, it also characterizes the maximum performance a network can achieve, and may serve as a baseline for defining quasi-optimal low-complexity protocols.

Energy related problems in WSNs have been addressed by several previous works (see [2] and the references therein). Many analytical studies aimed at maximizing the performance of the network in terms of throughput [3]–[6], delay [7], quality of service [8], or other metrics. However, differently from this paper, most of the protocols proposed in the literature consider centralized policies, in which a controller coordinates all nodes and knows the global state of the system over time. [9] analyzed decentralized policies with a particular focus on symmetric systems, and proposed a game theoretic approach for solving the problem. Instead, in this paper we use a different framework based on decentralized Markov decision processes, which can also handle asymmetric scenarios.

Recently, Dibangoye *et al.* [10]–[12] derived several important results in decentralized control theory. In this paper, we apply some of their results to an energy harvesting scenario, and, specifically, we model the system using a Decentralized-Markov Decision Process (Dec-MDP), which is a particular case of Decentralized-Partially Observable Markov Decision Process (Dec-POMDP). In [12], a detailed study of the Dec-POMDPs was presented and different approaches to solve them were proposed. The notion of *occupancy state* was introduced as a fundamental building block for Dec-POMDPs, and it was shown that, differently from classic statistical descriptions (e.g., belief states), it represents a sufficient statistic for control purposes. Using the occupancy state, we can convert the Dec-POMDP to an equivalent MDP with a continuous state space, named *occupancy-MDP*. Then, standard techniques to solve POMDPs and MDPs can be applied; for example, an approach to solve a continuous state space MDP is to define a grid of points (see Lovejoy's grid approximation [13]) and solve the MDP only in a subset of states. Although several papers introduced more advanced techniques to refine the grid [14], this approach may be inefficient and difficult to apply. Instead, in this paper we use a different scheme, namely the Learning Real Time A* (LRTA*) algorithm [15], which has the key advantage of exploring only the states which are actually visited by the process, without the difficulty of defining a grid of points.

Converting the Dec-POMDP to an occupancy-MDP produces a simpler formulation of the problem, which however does not reduce its complexity. Indeed, for every occupancy state, it is still required to perform the *exhaustive backup* operation, i.e., to compute a decentralized control policy. This is the most critical operation in decentralized optimization, since it involves solving a non-convex problem with many variables. The problem can be simplified by imposing a predefined structure to the policy [16], so that only few parameters need to be optimized. While this may lead to suboptimal solutions, it greatly simplifies the numerical evaluation and, if correctly designed, produces close to optimal results.

The paper is organized as follows. Section II presents the system model and the Dec-MDP formulation. The decentralized optimization problem is described in Section III and solved in Section IV. The numerical results are shown in Section V. Finally, Section VI concludes the paper.

Notation. Throughout this paper, superscripts indicate the node indices, whereas subscripts are used for time indices. Boldface letters indicate *global* quantities (i.e., vectors referred to all users).

II. System Model

The network is composed of one Access Point (AP) and N harvesting nodes (see Figure 1 for a graphical interpretation). We focus on a large time horizon, and time slot $k = 0, 1, \dots$

An extended version of this paper can be found in [1].

978-1-5090-5908-9/17/$31.00 ©2017 IEEE

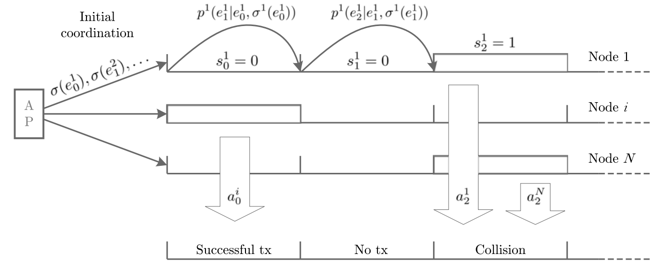

Figure 1: Time evolution of the system. After an initial coordination phase, every user acts independently of the others.

corresponds to the time interval $[kT,(k+1)T)$. During a slot, every node independently decides whether to access the uplink channel and transmit a message to AP, or to remain idle. We adopt an on/off collision model in which overlapping packet transmissions are always unrecoverable.

In slot k, node i harvests energy from the environment according to a pdf B^i_k (e.g., similarly to [17], we will use a Bernoulli energy arrival process) and we assume independent arrivals among nodes. However, the model can be easily extended to the more general, time correlated case (e.g., via an underlying common Markov model as in [3]).

Every node is equipped with a rechargeable battery, so that the energy stored in slot k can be used in a later slot. The global energy level vector in slot k is $\mathbf{e}_k=\langle e^1_k,\dots,e^N_k\rangle$, and is perfectly known by AP at $k=0$. This information is used for initializing the parameters of the whole network. After the initial coordination phase at $k=0$, every node acts independently of the others, and is not aware of the other battery levels in the network.

A. Decentralized–MDPs for EH Systems

The model presented so far can be formalized using a decentralized Markov decision process framework [11]. In our context, an N-users Dec-MDP $\mathcal{M}=(\mathbf{E},\mathbf{A},p,r,\eta_0,\beta)$ is formally defined as follows

- ***Battery Level*** $\mathbf{E}=E^1\times\cdots\times E^N$ is the set of global battery levels $\mathbf{e}=\langle e^1,\dots,e^N\rangle$, with $e^i\in E^i\triangleq\{0,\dots,e^i_{\max}\}$ (device i can store up to $e^i_{\max}$ discrete energy quanta according to Equation (2)). Throughout, the terms "battery level" or "state" will be used interchangeably;
- ***Action*** $\mathbf{A}=A^1\times\cdots\times A^N$ is the set of global actions $\mathbf{a}=\langle a^1,\dots,a^N\rangle$, where $a^i\in A^i\triangleq[0,1]$ denotes node i's transmission probability. Although a^i should assume continuous values, we quantize the interval $[0,1]$ in a_{levels} uniformly distributed levels for numerical tractability. Action a^i is chosen by user i through a function $\sigma^i:E^i\to A^i$, and depends only on the local state e^i;
- ***Transition Probability*** p is the transition probability function $p:\mathbf{E}\times\mathbf{A}\times\mathbf{E}\to[0,1]$ which defines the probability $p(\bar{\mathbf{e}}|\mathbf{e},\mathbf{a})$ of moving from a global battery level $\mathbf{e}=\langle e^1,\dots,e^N\rangle$ to a global battery level $\bar{\mathbf{e}}=\langle\bar{e}^1,\dots,\bar{e}^N\rangle$ under the global action $\mathbf{a}$. When a transmission is performed, one energy quantum is consumed;
- ***Reward*** r is the reward function $r:\mathbf{E}\times\mathbf{A}\to\mathbb{R}^+$ that maps the global action $\mathbf{a}$ to the reward $r(\mathbf{e},\mathbf{a})$ when the global state is $\mathbf{e}$;
- η_0 is the initial state distribution. In our scenario we take
$$\eta_0(\mathbf{e})=\begin{cases}1, & \text{if } \mathbf{e}=\mathbf{e}_0,\\ 0, & \text{otherwise},\end{cases}\tag{1}$$
for some $\mathbf{e}_0$, i.e., we assume perfect knowledge in the initialization phase.
- $1-\beta$ is the probability that the system stops operating in a given slot (see [18]), and will be used in Section III.

In Section III we describe the optimization problem related to $\mathcal{M}$. Its solution provides a *decentralized control policy*, which will be discussed in Section IV.

Before presenting in more detail the previous bullet points, it is important to emphasize the following key characteristics of the Dec-MDP under investigation:

- $\mathcal{M}$ is *jointly fully observable*, i.e., if all nodes collaborated, the global state would be completely known (actually, this is what differentiates Dec-MDPs and Dec-POMDPs);
- $\mathcal{M}$ is a *transition independent* Dec-MDP, i.e., the action taken by node i influences only its own battery evolution in that slot and *not* the others. Formally, the transition probability function p can be decomposed as $p(\bar{\mathbf{e}}|\mathbf{e},\mathbf{a})=\prod_{i=1}^N p^i(\bar{e}^i|e^i,a^i)$. This feature is important to develop compact representations of the transmission policies, and in particular to derive Markovian policies as discussed in our Section II-E and in [10, Theorem 1].

B. Battery Level

We adopt a discrete model, so that every battery can be referred to as an energy queue, in which arrivals coincide with the energy harvesting process, and departures with packet transmissions. In particular, the battery level of node i in slot k is e^i_k and evolves as
$$e^i_{k+1}=\min\{e^i_{\max},e^i_k-s^i_k+b^i_k\},\tag{2}$$
where the $\min$ accounts for the finite battery size, s^i_k is the energy used for transmission and b^i_k is the energy arrived in slot k. s^i_k is equal to 0 with probability $1-a^i_k$, and to 1 with probability a^i_k. This model has been widely used in the EH literature [9], and represents a good approximation of a real battery when $e^i_{\max}$ is sufficiently high.

C. Action

Node i can decide to access the channel, with probability a^i_k, or to remain idle w.p. $1-a^i_k$. When a transmission is performed, one energy quantum is drained from the battery, and a corresponding reward $g(a^i_k)$ is obtained. When $e^i_k=0$, no transmission can be performed and $a^i_k=0$.

D. Transition Probability

The transition probability function of user i, namely p^i, is defined as follows (assume $\bar{e}\neq e^i_{\max}$)
$$p^i(\bar{e}|e,a)=\begin{cases}(1-p^i_B)a, & \text{if } \bar{e}=e-1,\\ (1-p^i_B)(1-a)+p^i_Ba, & \text{if } \bar{e}=e,\\ p^i_B(1-a), & \text{if } \bar{e}=e+1,\\ 0, & \text{otherwise}.\end{cases}\tag{3}$$
p^i_B is the probability that user i harvests one energy quantum. More sophisticated models, in which multiple energy quanta can be simultaneously extracted, are described in [19], and can be integrated in our model (involving, however, higher computational costs).

E. Reward

We will use the term "global reward" to indicate the overall performance of the system, and simply "single-user reward" to refer to the performance of an individual user. We first describe the single-user reward and then extend this to the overall network.

Single-User Reward. Assume to study isolated users, which do not suffer from interference, as in [17]. Data messages are associated with a *potential reward*, described by a random variable V^i which evolves independently over time and among nodes. The realization ν_k^i is perfectly known only at a time $t \geq kT$ and only to node i whereas, for $t < kT$, only statistical knowledge is available. Every node can decide to transmit (and accrue the potential reward ν_k^i) or not in the current slot k according to its value ν_k^i. In particular, it can be shown that a threshold transmission model is optimal for this system [17]; thus, node i always transmits when $\nu_k^i \geq \nu_{\text{th}}^i(e^i)$ and does not otherwise. Note that $\nu_{\text{th}}^i(e^i)$ depends on the underlying state (battery level) of user i.

On average, the reward of user i in a single slot when the battery level is e^i will be

$$g(\nu_{\text{th}}^i(e^i)) \triangleq \mathbb{E}[\chi(V^i \geq \nu_{\text{th}}^i(e^i))V^i] = \int_{\nu_{\text{th}}^i(e^i)}^{\infty} v f_V^i(v)\, \mathrm{d}v, \quad (4)$$

where $\chi(\cdot)$ is the indicator function and $f_V^i(\cdot)$ is the pdf of the potential reward, V^i. It is now clear that the transmission probability a^i is inherently dependent on the battery level as

$$a^i = \sigma^i(e^i) = \int_{\nu_{\text{th}}^i(e^i)}^{\infty} f_V^i(v)\, \mathrm{d}v = \bar{F}_V^i(\nu_{\text{th}}^i(e^i)), \quad (5)$$

where we introduce a function $\sigma^i(e^i)$, which maps local observations (e^i) to local actions $\sigma^i(e^i) = a^i$. Note that the complementary cumulative distribution function $\bar{F}_V^i(\cdot)$ is strictly decreasing and thus can be inverted. Therefore, there exists a one-to-one mapping between the threshold values and the transmission probabilities. In the following, we will always deal with a^i, and write $g(a^i)$ with a slight abuse of notation.

It can be proved that $g(a^i)$ is increasing and concave in a^i, i.e., transmitting more often leads to higher rewards, but with diminishing returns. Finally, note that this model is quite general and, depending on the meaning of V^i, can be adapted to different scenarios. For example, in a standard communication system in which the goal is the throughput maximization, V^i can be defined as the transmission rate subject to fading fluctuations [17].

Global Reward. The global reward is zero when multiple nodes transmit simultaneously, whereas it is equal to $w^i\nu_k^i$ if only node i transmits in slot k (w^i is the weight of node i). On average, since the potential rewards are independent among nodes, we have

$$r(\nu_{\text{th}}(\mathbf{e})) = \mathbb{E}\left[\sum_{i=1}^{N} w^i V^i \chi(V^i \geq \nu_{\text{th}}^i(e^i)) \prod_{j \neq i} \chi(V^j < \nu_{\text{th}}^j(e^j))\right], \quad (6)$$

which can be rewritten as

$$r(\mathbf{a}) = r(\boldsymbol{\sigma}(\mathbf{e})) = \sum_{i=1}^{N} w^i g(a^i) \prod_{j \neq i}(1 - a^j), \quad (7)$$

where we used $\mathbf{a}$ instead of $\nu_{\text{th}}(\mathbf{e})$ for ease of notation, and we introduced the vector function $\boldsymbol{\sigma} \triangleq \langle \sigma^1, \ldots, \sigma^N \rangle$. We remark that $\boldsymbol{\sigma}$ summarizes the actions of all users given every battery level, i.e., it specifies all the following quantities

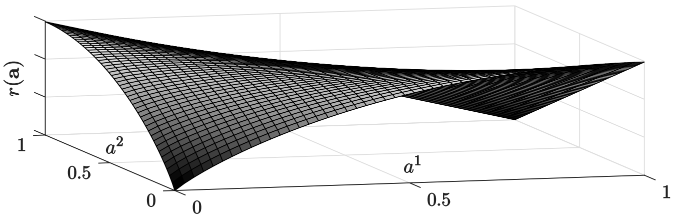

Figure 2: Global reward $r(\mathbf{a})$ when $N = 2$.

$$\begin{array}{ccc} \sigma^1(0), & \ldots & \sigma^1(e_{\max}^1), \\ \vdots & & \\ \sigma^N(0), & \ldots & \sigma^N(e_{\max}^N). \end{array} \quad (8)$$

As we will discuss later, finding $\boldsymbol{\sigma}$ represents the biggest challenge when solving a Dec-MDP.

An important observation is that the reward (7) is not necessarily increasing nor convex in $\mathbf{a}$, which significantly complicates the solution. An example of $r(\mathbf{a})$ for the two user case can be seen in Figure 2. Note that the maximum is achieved when only one device transmits with probability 1 and the other does not transmit. This implies that, when the devices are not energy constrained (i.e., they have enough energy for transmitting and the current transmission policy does not influence the future), the optimal user allocation should follow an orthogonal approach so as to avoid collisions (the corner points $\langle a^1, a^2 \rangle = \langle 1, 0 \rangle$ and $\langle a^1, a^2 \rangle = \langle 0, 1 \rangle$ achieve the maximum reward). However, as we will discuss later, this observation does not hold in EH scenarios, in which an action in the current slot influences the future energy levels and, consequently, the future rewards.

Note that, in the previous expressions, we have implicitly restricted our study to Markovian policies. A Markovian policy is a history-independent policy that maps local observations to local actions (i.e., $\sigma^i(e^i) = a^i$). In general decentralized frameworks, tracking previous observations can be used to optimally decide the current action. However, it can be proved [10] that under transition independent conditions (which hold in our case, see Section II-A), Markovian policies are optimal and thus keeping track of previous states is not necessary.

III. Optimization Problem

Ideally, the final goal of the system is to maximize the cumulative weighted discounted long-term reward, defined as

$$\bar{R}_\beta(\pi, \mathbf{e}_0) = \mathbb{E}\left[\sum_{k=0}^{\infty} \beta^k r\Big(\boldsymbol{\sigma}_k(\mathbf{e}_k)\Big)\bigg|\mathbf{e}_0, \pi\right], \quad (9)$$

where $\pi \triangleq (\boldsymbol{\sigma}_0, \boldsymbol{\sigma}_1, \ldots)$ is the *policy* and β was introduced in Section II-A as the probability that the network does not die in a given slot, which corresponds to the *discount factor* in classic MDPs [20]. Finding $\pi^\star = \arg\max_\pi \bar{R}_\beta(\pi, \mathbf{e}_0)$ corresponds to obtaining the highest reward when the state of the system $\mathbf{e}_k$ is globally known in slot k, i.e., in a *centralized*-oriented network. However, $\bar{R}_\beta(\pi, \mathbf{e}_0)$ cannot be achieved in a decentralized system, thus we must resort to a different notion of long term, which will be given in Equation (13). Nevertheless, (9) will be useful to initialize the Dec-MDP solver, since it provides an upper bound to the achievable performance, and can be easily computed using the Value Iteration Algorithm [20].

To formulate the decentralized optimization problem, we first introduce the concept of occupancy state.

A. Occupancy State

The occupancy state η_k is defined as

$$\eta_k(\bar{\mathbf{e}}) \triangleq \mathbb{P}(\mathbf{e}_k = \bar{\mathbf{e}} | \eta_0, \boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_{k-1}), \tag{10}$$

and represents a probability distribution over the battery levels given the initial distribution η_0 and all decentralized decision rules prior to k.

It can be shown that the occupancy state represents a sufficient statistic for control purposes in Dec-MDPs, and it can be easily updated at every slot of the system using old occupancy states:

$$\eta_k(\bar{\mathbf{e}}) = \omega(\eta_{k-1}, \boldsymbol{\sigma}_{k-1}) \triangleq \sum_{\mathbf{e}} p(\bar{\mathbf{e}} | \mathbf{e}, \boldsymbol{\sigma}_{k-1}(\mathbf{e})) \eta_{k-1}(\mathbf{e}), \tag{11}$$

where ω is the occupancy update function. Similarly to the reduction techniques of POMDPs, in which the belief is used as the state in an equivalent MDP, for Dec-MDPs the occupancy state will represent the building block of an equivalent MDP which can be solved using standard techniques.

B. Occupancy-MDP

Dibangoye *et al.* [12] developed a technique to solve Dec-MDPs by recasting them into equivalent continuous state MDPs. In particular, the state space of the equivalent MDP (called *occupancy-MDP*) is the occupancy simplex, the transition rule is given by (11), the action space is $\mathbf{A}$, and the instantaneous reward for taking decentralized decision rule $\boldsymbol{\sigma}_k$ is

$$\rho(\eta_k, \boldsymbol{\sigma}_k) = \sum_{\mathbf{e} \in \mathbf{E}} \eta_k(\mathbf{e}) r(\boldsymbol{\sigma}_k(\mathbf{e})), \tag{12}$$

i.e., it is the weighted sum of the rewards obtained in every battery level, where the weight is given by the occupancy state. Accordingly, the long-term reward of the occupancy-MDP is

$$R_\beta(\pi, \eta_0) = \mathbb{E}\left[\sum_{k=0}^{\infty} \beta^k \rho(\eta_k, \boldsymbol{\sigma}_k) \middle| \eta_0, \pi \right]. \tag{13}$$

Differently from $\bar{R}_\beta(\pi, \eta_0)$ in Equation (9), $R_\beta(\pi, \eta_0)$ can be actually achieved by a decentralized system.

The corresponding optimal policy is

$$\mu^\star = \arg\max_{\pi} R_\beta(\pi, \eta_0). \tag{14}$$

In the next section we discuss how to find the optimal as well as suboptimal solutions.

IV. Solution

Finding $\mu^\star$ requires solving an MDP with a continuous state space. To do that, we use techniques originally developed for POMDPs which were later used for Dec-POMDPs. In particular, the Learning Real Time A* (LRTA*) algorithm is suitable for our case, since it explores only the occupancy states which are actually visited during the planning horizon and avoids grid-based approaches (e.g., as used in [13]). In [10], the Markov Policy Search (MPS) algorithm was introduced as an adaptation of LRTA* to decentralized scenarios. In summary, MPS starts at $k = 0$ and uses η_0 as defined in (1); then, it iteratively updates upper and lower bounds to the optimal policy until they converge to the same value by using the convexity of the cost-to-go function. The solution of the fully-observable MDP of Equation (9) is used to initialize the upper bounds at the corner points of the simplex. For more details about MPS, we refer the readers to [10], [11].

A key step of MPS is the exhaustive backup, in which a new policy that maximizes the cost-to-go upper bound function $\bar{v}_{\beta,k}(\eta_k)$ is obtained. Formally, this requires to solve

$$\bar{v}_{\beta,k}(\eta_k) = \max_{\boldsymbol{\sigma}} \rho(\eta_k, \boldsymbol{\sigma}) + \beta \bar{v}_{\beta,k+1}(\omega(\eta_k, \boldsymbol{\sigma})). \tag{15}$$

In general, the exhaustive backup is critical to perform because all decentralized policies should be examined, thus the complexity would be $\mathcal{O}((a_{\text{levels}})^{e_{\max} \times N})$ if all users had the same battery size $e_{\max}$ (see the structure of $\boldsymbol{\sigma}(\mathbf{e})$ in Equation (8)). This operation is computationally infeasible when lots of possibilities are involved, thus other approaches were introduced to handle this problem. We first present some preliminary results.

A. Convexity of the Cost-to-go Function

It can be shown that, in the infinite horizon case, the optimal cost-to-go function $v^\star_{\beta,k}$ is a convex function of the occupancy states (see [11, Theorem 4.2] for a proof in the finite horizon case) and can be approximated by piecewise linear functions. Formally, $\bar{v}_{\beta,k}$ can be rewritten as

$$\bar{v}_{\beta,k}(\eta_k) = \max_{\boldsymbol{\sigma}} \rho(\eta_k, \boldsymbol{\sigma}) + \beta\, C(\Upsilon_k, \omega(\eta_k, \boldsymbol{\sigma})), \tag{16}$$

where C interpolates the occupancy state $\omega(\eta_k, \sigma)$ using the point set Υ_k, which contains the visited occupancy states along with their upper bound values. The first points to be put in Υ_k are the corners of the occupancy simplex with their values obtained solving the full knowledge MDP in Equation (9). Then, every time (16) is solved, a new point $(\eta_k, \bar{v}_{\beta,k}(\eta_k))$ is added to Υ_k.

Ideally, we could use a linear interpolation as the function C (i.e., map η_k on the convex hull of point set Υ_k), but this would incur high complexity. A faster solution, which however has shown good performance in many applications, is to replace C with the sawtooth projection:[1]

$$\begin{aligned} \textbf{sawtooth}(\Upsilon_k, \eta) &= y^0(\eta) + \min_{\ell \in \mathcal{L}} \left[(v^\ell - y^0(\eta^\ell)) \min_{\theta : \eta^\ell(\theta) > 0} \frac{\eta(\theta)}{\eta^\ell(\theta)} \right] \\ &= y^0(\eta) + \min_{\ell \in \mathcal{L}} \max_{\theta : \eta^\ell(\theta) > 0} \left[\frac{\eta(\theta)}{\eta^\ell(\theta)} (v^\ell - y^0(\eta^\ell)) \right] \\ &= \min_{\ell \in \mathcal{L}} \left[y^0(\eta) + \max_{\theta : \eta^\ell(\theta) > 0} \left[\frac{\eta(\theta)}{\eta^\ell(\theta)} (v^\ell - y^0(\eta^\ell)) \right] \right]. \end{aligned} \tag{17}$$

(η^ℓ, v^ℓ) is the ℓ-th element of Υ_k, $\mathcal{L}$ is the set of indices of Υ_k, and y^0 is the upper bound computed using the corner points of Υ_k, i.e.,

$$y^0(\eta) = \sum_{\mathbf{e} \in \mathbf{E}} \eta(\mathbf{e}) \Upsilon_k(\mathbf{e}), \tag{18}$$

where, with a slight abuse of notation, $\Upsilon_k(\mathbf{e})$ indicates the upper bound value at the corner $\mathbf{e}$ of the simplex.

The sawtooth projection produces higher (i.e., worse) upper bounds than the convex hull projection and thus MPS may require more iterations to converge (however, a single iteration can be performed much more quickly), but convergence is still guaranteed.

[1] The term "sawtooth" comes from the shape of the interpolating function in the two-dimensional case. The idea of the approach is to interpolate a point η using $|\mathbf{E}| - 1$ corner points of the simplex, and one point (the ℓ-th point) taken from Υ_k.

B. Parametric Policies

Since the main issue of the exhaustive backup is that the space of variables is exceedingly large, we aim at reducing this space, so that $\boldsymbol{\sigma}$ cannot take all possible values but is constrained to lie in a smaller subset. This will in turn lead to suboptimal solutions, which however are much faster to compute. In this paper, we use *parametric policies* and thus reduce the number of optimization variables to few parameters. In particular, we force the actions of user i to follow a predetermined structure:

$$\sigma^i(e^i) = f^i_{\text{par}}(\Theta^i, e^i) \tag{19}$$

where e^i is the independent variable and Θ^i is a set of parameters which specify the structure of f^i_{par}. For example, if we used $\Theta^i = \{\theta^i\}$, and a simple linear function $f^i_{\text{par}}(\Theta^i, e^i) = \theta^i e^i$, the only optimization variable of user i would be θ^i, and not $\sigma^i(0), \ldots, \sigma^i(e^i_{\max})$ as in the original problem. In this case, for a symmetric scenario, the complexity of the exhaustive search step goes from $\mathcal{O}((a_{\text{levels}})^{e_{\max} \times N})$ to $\mathcal{O}((\theta_{\text{levels}})^N)$, therefore it remains exponential in N but with a much smaller coefficient in the exponent. θ_{levels} is the number of values that θ^i can assume.

In our scenario we force $f^i_{\text{par}}(\Theta^i, e^i)$ to be a non-decreasing function of e^i as in [17], which implies that higher energy levels cannot correspond to lower transmission probabilities.

V. Numerical Evaluation

The numerical evaluation is performed using two nodes, since the complexity grows super-exponentially with the number of users. Indeed the size of the occupancy state evolves exponentially with N, and the exhaustive backup operation (exponential in N), or a suboptimal approach, is to be performed for every element of the occupancy state. If not otherwise stated, we adopt the following parameters: the batteries can contain up to $e^1_{\max} = e^2_{\max} = 5$ energy quanta; the probabilities of receiving an energy quantum are equal to $p^1_B = p^2_B = 0.1$ in every slot (i.i.d. energy arrival processes); when a transmission is performed a reward $V^i = \ln(1 + \Lambda^i H^i)$ is accrued, where V^i is defined as the normalized transmission rate in a slot, where H^i is an exponentially distributed random variable with mean 1 (see [17]); the average normalized SNRs are $\Lambda^1 = 6$ and $\Lambda^2 = 4$; both devices have the same weight; finally, the discount factor is $\beta = 0.9$. All the numerical evaluations were written in C++.

In Figures 3 and 4 we show the transmission probabilities of the parametric decentralized policy of Section IV-B, where f_{par} is a linear function, $\Theta^i = \{\theta^i\}$ and θ^i is such that $\theta^i e^i_{\max} \in A^i$.

From these two figures, an interesting effect can be observed in the initial transmission slots: when the available energy is scarce, then both nodes have a non-zero probability of accessing the channel; instead, if a lot of energy is available, the transmission policy almost degenerates into a pure time-orthogonal access mechanism. Also, in Figure 3, the average transmission probabilities coincide with the energy arrival rate in the long run, so as to achieve energy neutrality. In summary, this proves that when the energy resources are scarce (here we show this effect in terms of initial energy levels, but a similar behavior could be observed for low energy arrivals) then an orthogonal scheme, in which collisions are avoided, is suboptimal. The trade-off between orthogonal and random access schemes can be intuitively explained as follows. When the initial energy levels are high but the harvesting probabilities are low, both nodes know that the other device has a lot of energy available in the first slots. Thus, since there are no energy outages, they can adopt an orthogonal access policy, so that the channel is almost always used (which corresponds to the best mechanism without EH constraints). This regime almost corresponds to the full-knowledge case. Instead, as time goes on, nodes lose information about the global state of the system, thus a device does not know the energy level of the other. In this case, an orthogonal scheme might be highly inefficient, since a node may not have enough energy to transmit during its slots; here, a random access scheme provides higher performance.

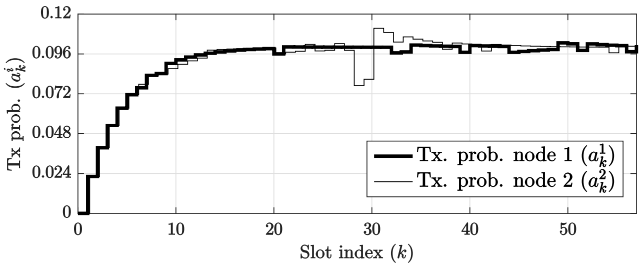

Figure 3: Transmission probabilities as a function of time for two users with batteries initially empty.

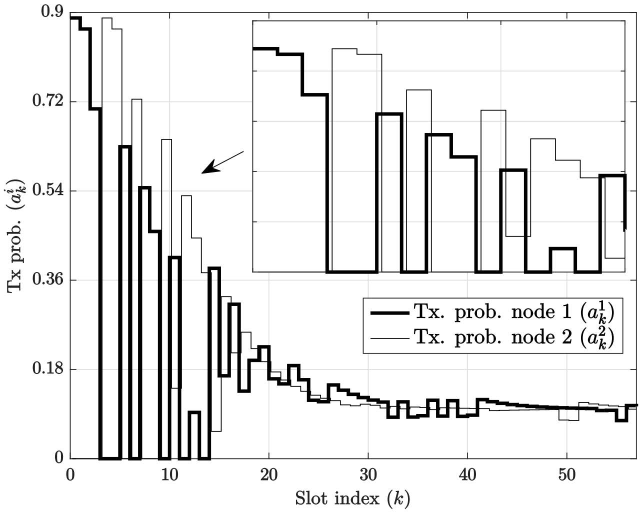

Figure 4: Transmission probabilities as a function of time for two users with batteries initially fully charged.

Figure 5 shows another interesting, although predictable, result: regardless of the initial energy level, in the long run all the energy levels degenerate to the same value. This is because all the initial fluctuations have been absorbed by the batteries. Moreover, we also remark that, after many slots, the knowledge of a user about the others coincides with their steady-state probabilities, since the global battery knowledge is not refreshed at any time after $k = 0$.

Finally, in Figure 6 we show the long-term discounted reward as a function of the energy arrival rate for the optimal centralized scheme (Equation (9)) and the decentralized parametric scheme (Equation (13)). When the initial batteries are fully charged, then centralized and decentralized schemes are much closer, whereas, for battery initially empty, the gap is much wider.

Another counter-intuitive phenomenon can be observed as

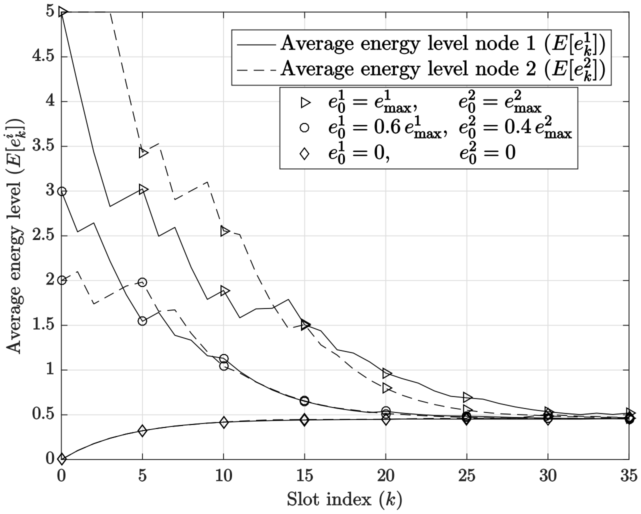

Figure 5: Battery level evolution as a function of time for two users with different initial battery levels.

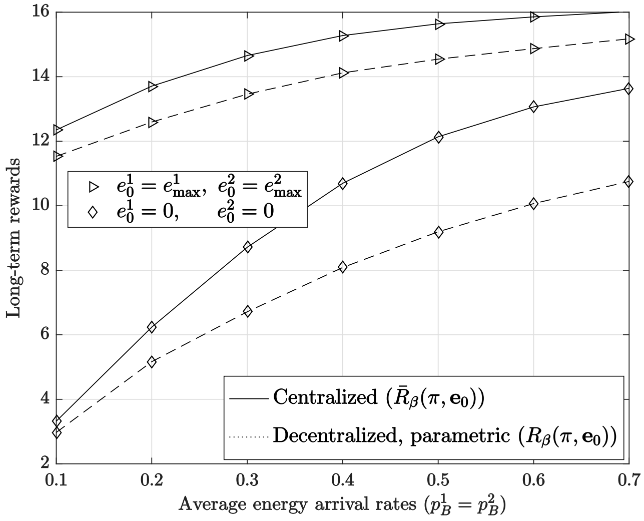

Figure 6: Centralized and decentralized long-term rewards as a function of the energy arrival rates for two users with different initial battery levels.

the average energy arrival rate grows. Indeed, as previously explained, when a lot of energy is available, an almost orthogonal scheme is optimal, thus centralized and decentralized schemes should have similar performance. However, in Figure 6 the opposite effect can be observed. This is because we are using a discounted formulation, thus the first slots are the most important ones. When a lot of energy arrives to the system, the battery fluctuations are more frequent, thus the distance between centralized and decentralized approaches becomes wider. Finally, note that the performance of the parametric policy is strongly influenced by the number of parameters Θ^i we used, and using more parameters would lead to better performance.

VI. Conclusions

We studied a decentralized optimization framework for an energy harvesting communication network with collisions. We used a decentralized Markov decision process to model the system, and described how to find the optimal policy as well as suboptimal schemes. In our numerical evaluation we describe the trade-off between accessing the channel and energy arrivals, and we showed that a pure orthogonal access mechanism is suboptimal under harvesting constraints.

Due to the super-exponential complexity of the optimal solution, future work will investigate more practical schemes which inherit the key properties of our framework while being less computational demanding.

References

[1] A. Biason, S. Dey, and M. Zorzi, “A decentralized optimization framework for energy harvesting devices,” *arXiv:1701.02081*, submitted to *IEEE Trans. on Mobile Computing*, Dec. 2016.

[2] S. Ulukus, A. Yener, E. Erkip, O. Simeone, M. Zorzi, P. Grover, and K. Huang, “Energy harvesting wireless communications: A review of recent advances,” *IEEE J. Sel. Areas in Commun.*, vol. 33, no. 3, pp. 360–381, Mar. 2015.

[3] N. Michelusi, K. Stamatiou, and M. Zorzi, “Transmission policies for energy harvesting sensors with time-correlated energy supply,” *IEEE Trans. Commun.*, vol. 61, no. 7, pp. 2988–3001, July 2013.

[4] A. Biason, D. Del Testa, and M. Zorzi, “Low-complexity policies for wireless sensor networks with two energy harvesting devices,” in *Proc. IEEE 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, June 2014, pp. 180–187.

[5] 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.

[6] O. Ozel and S. Ulukus, “Achieving AWGN capacity under stochastic energy harvesting,” *IEEE Trans. Inf. Theory*, vol. 58, no. 10, pp. 6471–6483, Oct. 2012.

[7] V. Sharma, U. Mukherji, V. Joseph, and S. Gupta, “Optimal energy management policies for energy harvesting sensor nodes,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 4, pp. 1326–1336, Apr. 2010.

[8] C. Pielli, A. Biason, A. Zanella, and M. Zorzi, “Joint optimization of energy efficiency and data compression in TDMA-based medium access control for the IoT,” in *Proc. IEEE Global Communications Conf. (GLOBECOM), IoT-LINK Workshop*, Dec. 2016.

[9] N. Michelusi and M. Zorzi, “Optimal adaptive random multiaccess in energy harvesting wireless sensor networks,” *IEEE Trans. Commun.*, vol. 63, no. 4, pp. 1355–1372, Apr. 2015.

[10] J. S. Dibangoye, C. Amato, and A. Doniec, “Scaling up decentralized MDPs through heuristic search,” in *Proc. Uncertainty in Artificial Intelligence (UAI)*, Aug. 2012.

[11] J. S. Dibangoye, C. Amato, A. Doniec, and F. Charpillet, “Producing efficient error-bounded solutions for transition independent decentralized MDPs,” in *Proc. ACM Int. Conf. Autonomous Agents and Multi-agent Systems (AAMAS)*, May 2013.

[12] J. S. Dibangoye, C. Amato, O. Buffet, and F. Charpillet, “Optimally solving Dec-POMDPs as continuous-state MDPs: theory and algorithms,” *INRIA*, Research Report, no. 8517, Apr. 2014.

[13] W. S. Lovejoy, “Computationally feasible bounds for partially observed Markov decision processes,” *Operations research*, vol. 39, no. 1, pp. 162–175, Jan.–Feb. 1991.

[14] R. Zhou and E. A. Hansen, “An improved grid-based approximation algorithm for POMDPs,” in *Proc. Int. Joint Conf. on Artificial Intelligence*, vol. 17, no. 1, Aug 2001, pp. 707–716.

[15] R. E. Korf, “Real-time heuristic search,” *Artificial intelligence*, vol. 42, no. 2-3, pp. 189–211, Mar. 1990.

[16] D. T. Hoang, D. Niyato, P. Wang, and D. I. Kim, “Optimal decentralized control policy for wireless communication systems with wireless energy transfer capability,” in *Proc. IEEE Int. Conf. Communications (ICC)*, June 2014, pp. 2835–2840.

[17] N. Michelusi, K. Stamatiou, and M. Zorzi, “On optimal transmission policies for energy harvesting devices,” in *Proc. IEEE Information Theory and Applications Workshop (ITA)*, Feb. 2012, pp. 249–254.

[18] 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.

[19] A. Biason and M. Zorzi, “Joint transmission and energy transfer policies for energy harvesting devices with finite batteries,” *IEEE J. Sel. Areas in Commun.*, vol. 33, no. 12, pp. 2626–2640, Dec. 2015.

[20] D. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific, Belmont, Massachusetts, 2005.