Reinforcement learning based transmission policies for energy harvesting powered sensors

Ruslan Seifullaev, Steffi Knorn, Anders Ahlén, and Roland Hostettler

DOI:

10.1109/TGCN.2024.3374899

Reinforcement learning based transmission policies for energy harvesting powered sensors

Ruslan Seifullaev, Steffi Knorn, *Member, IEEE*, Anders Ahlén, *Senior Member, IEEE*, and Roland Hostettler, *Member, IEEE*

***Abstract*—We consider a sampled-data control system where a wireless sensor transmits its measurements to a controller over a communication channel. We assume that the sensor has a harvesting element to extract energy from the environment and store it in a rechargeable battery for future use. The harvested energy is modelled as a first-order Markovian stochastic process conditioned on a scenario parameter describing the harvesting environment. The overall model can then be represented as a Markov decision process, and a suitable transmission policy providing both good control performance and efficient energy consumption is designed using reinforcement learning approaches. Finally, supervisory control is used to switch between trained transmission policies depending on the current scenario. Also, we provide a tool for estimating an unknown scenario parameter based on measurements of harvested energy, as well as detecting the time instants of scenario changes. The above problem is solved based on Bayesian filtering and smoothing.**

***Index Terms*—Energy-harvesting, communication networks, Bayesian filtering, reinforcement learning**

I. Introduction

This paper studies the possibility of using energy harvesting powered sensors in wireless communications and proposes a design of suitable transmission policies from sensors to a controller.

A. Motivation

Energy harvesting has gained increasing attention in recent years due to its potential to reduce reliance on non-renewable energy sources, improve the sustainability of various applications, and enable the development of self-powered devices and systems, [1]–[3]. Using harvesting-powered sensors in wireless control systems has the advantage of replenishing energy used for transmissions by extracting it from the environment. Understanding the models for energy harvesting is crucial for designing and optimizing these systems. These models can help to predict the energy harvesting potential of different sources, and can also provide insights into the energy harvesting process, ensuring efficient and reliable energy collection and storage. Moreover, knowledge of energy harvesting models is also essential to design suitable transmission policies, which can significantly reduce energy consumption and improve network efficiency in wireless control systems.

B. Energy harvesting. Background

Energy harvesting is the process of capturing and converting ambient energy from various sources into usable electrical energy. This energy can then be used to power electronic devices, e.g., wireless sensors, without the need for traditional batteries or external power sources that can be costly or impossible to replace. Among various harvesting architectures, the most attractive is the *harvest-store-used architecture*, see [4], which stores the harvested energy in a battery or capacitor for later use. This architecture is commonly used in applications that require a continuous power supply and where the energy available from the environment is intermittent or unreliable.

Energy can be harvested from various sources, including

- *Solar Energy.* This is one of the most popular energy sources for harvesting. It involves converting the light energy into electrical energy using photovoltaic (PV) cells, [5].
- *Thermal Energy.* This is the energy that results from the temperature difference between two objects. It can be harvested using thermoelectric generators (TEGs) that convert heat into electrical energy, [6].
- *Mechanical Energy.* This energy can be harvested from mechanical vibrations, such as those generated by machines or human movement. It can be converted into electrical energy using piezoelectric materials, [7], [8].
- *Radio Frequency (RF) Energy.* This energy can be harvested from ambient RF signals, such as those from Wi-Fi or cellular networks. It can be converted into electrical energy using antennas and rectifiers, [9], [10].
- *Wind Energy.* This energy can be harvested using small wind turbines that convert the kinetic energy from wind into electrical energy, [11].

Once energy is harvested, it can be stored using different storage technologies. One commonly used technology is the rechargeable battery, which can store the harvested energy and supply it when needed.

Since the energy source is unpredictable, we characterize the harvested energy as a stochastic process using Markovian processes, as traditionally done in the literature, see, e.g., [12]–

This work was supported by the Swedish Research Council (VR) under contracts Dnr: 2017-04186 and Dnr: 2022-04505.

Ruslan Seifullaev is with the Division of Systems and Control, Department of Information Technology, Uppsala University, 751 03 Uppsala, Sweden (e-mail: ruslan.seifullaev@it.uu.se).

Anders Ahlén and Roland Hostettler are with the Division of Signals and Systems, Department of Electrical Engineering, Uppsala University, 751 03 Uppsala, Sweden (e-mail: anders.ahlen@angstrom.uu.se; roland.hostettler@angstrom.uu.se).

Steffi Knorn is with the Division of Signals and Systems, Department of Electrical Engineering, Uppsala University, 751 03 Uppsala, Sweden, and also with the Institut für Prozess-und Verfahrenstechnik, Technische Universität Berlin, 10623 Berlin, Germany (e-mail: steffi.knorn@angstrom.uu.se).

[14]. In [15], additional scenario parameters[1] were introduced to cover the nonstationarity of the Markovian process. These scenario modes are modeled as another stochastic process and assumed to be slowly varying based on the harvesting environment. Sometimes, scenario parameters strongly depend on time, e.g., daylight or working hours, and can hence be interpreted as deterministic and periodic, see [16]. However, in many other cases, they are random, unknown, and have to be estimated based on the measurements of harvested energy.

C. Contribution

In this paper, we consider a controlled dynamical nonlinear system where the output is measured by a harvesting powered sensor that transmits its measurements to a controller over a fading channel [17], [18]. We assume that the sensor can send data only at discrete time instants. If the transmission occurs, then the sensor consumes a certain amount of energy depending on the channel conditions and updates the information on the controller side. If the sensor decides not to transmit, no energy is spent in that time instance, but the controller must then hold the recent output value, which may worsen the system performance. The goal of the transmission policy is to minimize the total cost, which consists of the output error penalty and the cost of energy consumption. For each fixed scenario mode, the appropriate policy can be designed based on a reinforcement learning (RL) approach that minimizes a state-value function, see e.g. [19], [20]. Then a supervisory control can be used to switch between policies based on the current mode estimate.

The main contributions of this paper are as follows.

- We use Bayesian filtering and smoothing for estimating unknown scenario parameters based on measurements of harvested energy, as well as detecting the time instants of scenario changes. For the latter, we also propose an algorithm to reduce false detections, which uses minimum mean square error (MMSE) estimates of the filter posterior probability.
- We propose a heuristic algorithm that solves the problem of adding a new mode to the scenario state space when none of the existing modes fit the measured data properly. In this case, the Jensen–Shannon divergence is used as a measure of the distance between distributions.
- We investigate and compare the use of a dynamic programming algorithm based on the solution of the Bellman equation and a Q-learning algorithm to obtain a suboptimal transmission policy by representing the complete closed-loop system as a Markov decision process (MDP) and designing an appropriate cost function.
- Finally, we illustrate the proposed approach by considering a numerical example of temperature control.

The rest of the paper is structured as follows. The problem statement is formulated in Section II, where we describe the models for the control system, the battery state, the energy-harvesting process, and transmission energy based on the channel power gain. The main results of the Bayesian estimation of unknown scenario parameters are presented in Section III. The reinforcement learning approach for transmission policy design is described in Section IV. Section V provides a numerical example demonstrating the efficiency of the proposed approaches. The conclusions are given in Section VI.

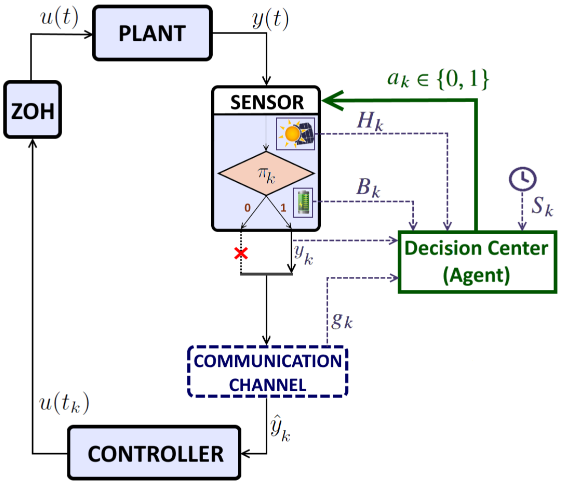


Fig. 1: Wireless Control System

II. Problem Formulation

We consider a dynamical system given by a system of ordinary differential equations (ODE)

$$\frac{dx(t)}{dt} = f(x(t), u(t)), \quad y(t) = h(x(t)), \tag{1}$$

with a state vector $x(t) \in \mathbb{R}^{n_x}$, a control input $u(t) \in \mathbb{R}^{n_u}$, and an output $y(t) \in \mathbb{R}^{n_y}$. Such a description is the most common way to model real-life processes having their own dynamics, e.g., mechanical, biological, chemical processes, etc. We assume that the output is measured by a wireless sensor that contains a rechargeable battery and an energy harvester to collect energy from the environment, see Fig. 1. Consider a periodic sequence $t_{k+1} = t_k + \bar{h}$, $k = 0, 1, 2, \ldots$, where $\bar{h} > 0$, and assume that the sensor can transmit its measurements to the controller only at time instants t_k. We also assume that an output-feedback sampled-data control is designed such that the closed-loop system is exponentially stable, i.e., there exists $\bar{\beta} \geq 1$ such that for any solution $x(t)$ with initial condition $x(t_0)$ the following inequality holds

$$||x(t)||_2^2 \leq \bar{\beta} \mathrm{e}^{-2\bar{\alpha}(t-t_0)} \, ||x(t_0)||_2^2, \; \forall t \geq t_0. \tag{2}$$

where $\bar{\alpha} > 0$ is the convergence rate[2]. However, such a transmission policy, where transmissions occur at every time instants t_k, may lead to high sensor energy consumption. In this regard, to save energy, the sensor may decide not to transmit at certain times t_k. In that case, the controller holds the most recently received measurement $\hat{y}(t_k)$, i.e.,

$$\hat{y}(t_k) = \begin{cases} y(t_k), & \text{if } \pi_k = 1, \\ \hat{y}(t_{k-1}), & \text{if } \pi_k = 0, \end{cases} \tag{3}$$

[1] We will also refer to these parameters as scenario modes or simply modes

[2] For nonlinear sampled-data control systems $\dot{x}(t) = Ax(t) + \varphi(rx(t), t) + Bu(t)$, $y(t) = Cx(t)$ with sector-bounded nonlinearities φ, the conditions guaranteeing (2) can be found in [21], [22].

where $\pi_k \in \{0,1\}$ indicates whether the transmission has occurred ($\pi_k = 1$) or not ($\pi_k = 0$), and $\hat{y}(t_{-1}) = 0$. However, the system performance can be degraded if $\pi_k = 0$ for some k. Thus, it is reasonable to have a transmission policy that takes into account both control accuracy and energy consumption. A more detailed problem statement is given below.

A. Sensor battery model

Since the transmissions over a communication channel may occur only at discrete instants t_k, we model the battery level, B_k, as a process over discrete time k given by

$$B_k = \min\left\{B_{k-1} + H_k - T_{k-1}, \bar{B}\right\}, \tag{4}$$

where H_k denotes the energy harvested by the sensor at time slot k, i.e., during the interval $[t_{k-1}, t_k)$, T_k is the energy used by the sensor at time t_k, and $\bar{B}$ is the battery capacity, see [23], [24]. The models for the harvested and transmitted energy will be described below.

B. Harvesting energy

Consider a discrete set $\mathcal{H} = \{\varkappa_1, \ldots, \varkappa_N\}$, where $\varkappa_1 \geq 0$ and $\varkappa_i - \varkappa_{i-1} = \Delta > 0$ for all $i = 2, \ldots, N$, and suppose that the values of H_k come from $\mathcal{H}$. We assume that the harvested energy H_k is a first-order Markovian stochastic process over discrete time k conditioned on a scenario mode S_k. In [15] it was proposed to interpret S_k as another stochastic process, where the values of S_k were long-term, slowly-varying parameters depending on the harvesting environment. We assume that S_k comes from a discrete set $\{1, 2, \ldots, M\}$. Then the joint probability mass function (pmf) of $H_{1:K} \triangleq [H_1, \ldots, H_K]$ and $S_{1:K} \triangleq [S_1, \ldots, S_K]$ given H_0 and S_0 can be factorized as

$$p\left(H_{1:K}, S_{1:K} \mid H_0, S_0\right) = \prod_{k=1}^{K} p\left(H_k \mid H_{k-1}, S_k\right) p\left(S_k \mid S_{k-1}\right), \tag{5}$$

where K can be sufficiently large. Suppose that n_{mij} samples are observed with $H_k = \varkappa_j$ given $H_{k-1} = \varkappa_i$ and $S_k = m$ for $k = 1, \ldots, K$. Also suppose that there are n'_{rm} samples observed where $S_k = m$ given $S_{k-1} = r$. Then the unknown model parameters $p_{mij} = \Pr(H_k = \varkappa_j \mid H_{k-1} = \varkappa_i, S_k = m)$ and $q_{rm} = \Pr(S_k = m \mid S_{k-1} = r)$ can be estimated based on empirical measurements, where the maximum likelihood estimates are $p^*_{mij} = \frac{n_{mij}}{\sum_{\mu=1}^{N} n_{mi\mu}}$ and $q^*_{rm} = \frac{n'_{rm}}{\sum_{\mu=1}^{M} n'_{r\mu}}$, which form the transition probability matrices $\mathbb{T}_m = \left[p^*_{mij}\right]$, $m = 1, \ldots, M$ and $\mathbb{T}' = \left[q^*_{rm}\right]$, see [16] for details.

As was stated in the introduction, in many cases, the modes S_k are unknown and have to be estimated based on the measurements of harvested energy. In Section III, we will use the Bayesian filtering and smoothing technique for estimating unknown scenario parameters, as well as detecting the time instants of scenario changes. Formally, we consider the following estimation problems.

Problem 1: Given the probability mass functions $p\left(H_k \mid H_{k-1}, S_k\right)$, $p\left(S_k \mid S_{k-1}\right)$, and the initial distribution $p(S_0)$. The objective is to find the estimates $\bar{S}_k$ of S_k given the measurements $H_{1:k}$ for each $k = 1, \ldots, K$, as well as to detect the switching instants n_i ($i = 1, 2, \ldots$), i.e., the times where S_k changes its value.

Problem 2: If none of the existing scenario modes $m \in \{1, \ldots, M\}$ fits the observed data properly, we have to decide whether to add a new mode $M+1$ to the scenario state space.

C. Transmission energy

Finally, we will propose a model for the transmission energy T_k that the sensor consumes to transmit a packet over the communication channel at time t_k. We consider the case when the required energy is inversely proportional to the channel gain g, and assume that g is an i.i.d. process described by a distribution $p_g(z)$, where z is the continuous received signal strength (RSS) in dBm. In industrial environments, the most suitable distribution that characterizes the radio channel power gain over long time horizons is the compound distribution

$$p_g(z \mid \tilde{\sigma}, \tilde{m}) = \int_{-\infty}^{\infty} p_1(z - \nu \mid \tilde{m})\, p_0(\nu \mid \tilde{\sigma}) d\nu, \tag{6}$$

where p_1 and p_0 are the dB-representations of the Gamma distribution with the parameter $\tilde{m}$ (Nakagami-m fading parameter) and the Lognormal distribution with standard deviation $\tilde{\sigma}$, respectively, see [17], [18]. Assume that the RSS measurements are obtained from a coarse quantizer (see [25]) with some fixed resolution, and q_g is the probability mass function corresponding to (6). Thus, we consider the following discrete model for the channel gain: $g_k \sim q_g$. Hence, the transmission energy T_k that is used by the sensor at time t_k is given by

$$T_k = \begin{cases} \frac{c_g}{g_k}, & \text{if } \pi_k = 1, \\ 0, & \text{if } \pi_k = 0, \end{cases} \tag{7}$$

where c_g is some fixed scaling parameter.

D. Transmission policy

Consider a sequence of actions (decisions)

$$\mathbf{a} = \{a_0, a_1, a_2, \ldots\}, \quad a_k \in \{0, 1\},$$

where $a_k = 1$ means that the sensor should transmit its measurements to the controller at time t_k, and $a_k = 0$ indicates that the measurements should not be transmitted such that the controller must hold the old value of the output instead. Then the sequence of actual transmissions, π_k, is defined as

$$\pi_k = \begin{cases} 1, & \text{if } a_k = 1 \text{ and } B_k \geq \frac{c_g}{g_k}, \\ 0, & \text{if } a_k = 0 \text{ or } B_k < \frac{c_g}{g_k}. \end{cases} \tag{8}$$

In other words, the transmission occurs if there is a decision to transmit and the battery has enough energy for it.

The main goal of this paper is to find a suitable transmission policy providing good control quality and, at the same time, efficient energy consumption. More formally, we consider the following cost function

$$l_k = e_k^{\mathrm{T}} \Lambda e_k + c_e T_k \left(1 - \frac{B_k}{\bar{B}}\right), \tag{9}$$

where $e_k = (y_k - \hat{y}_k)$, $y_k \triangleq y(t_k)$, $\hat{y}_k \triangleq \hat{y}(t_k)$, $c_e > 0$ and Λ is a positive definite weighting matrix. We can see that the cost l_k consists of two terms: the output error penalty and the cost for energy consumption. If $\pi_k = 1$, i.e., the transmission occurs, the information on the controller side is updated, which means that $\hat{y}_k = y_k$, and hence the output error e_k is zero[3]. Then the cost comes only from energy usage[4]. And vice versa, if $\pi_k = 0$, then $T_k = 0$, see (7), and we get only the output error penalty. The coefficient c_e is a parameter defining the weight between control accuracy and energy consumption. Therefore, the problem can be formulated as follows.

Problem 3: For each fixed mode $S_k \equiv m$ $(m = 1, \ldots, M)$, we have to find a policy $\mathbf{a}_m$ that minimizes the total cost function $\mathbb{E}\left[\sum_{k=1}^{\infty} \delta^k l_k\right]$, where $\delta \in (0,1)$ is a discount factor.
This problem can be addressed using reinforcement learning (RL) approaches, as demonstrated in Section IV. Once Problems 1–3 are solved, a supervisory control can be used to switch between the designed policies depending on the scenario mode changes.

III. Scenario Parameter Estimation

To implement the supervisory control, i.e., switching between the policies, the mode S_k has to be known for each k. However, in many cases, S_k is unknown and has to be estimated based on the measurements of harvested energy. In this section, to address Problems 1 and 2, we will use the Bayesian filtering and smoothing technique for estimating unknown scenario modes, as well as detecting the time instants of scenario changes.

The purpose of Bayesian filtering is to compute the posterior probability mass function (pmf) of S_k given $H_{1:k}$. Consider the probability density functions $p_c\left(S_k \,|\, S_{k-1}\right)$ and $p_c\left(H_k \,|\, H_{k-1}, S_k\right)$ obtained from the given pmfs $p\left(S_k \,|\, S_{k-1}\right)$ and $p\left(H_k \,|\, H_{k-1}, S_k\right)$, respectively, using zero-order hold interpolation, i.e.,

$$p_c\left(S_k \,|\, S_{k-1}\right) = p\left(m \,|\, r\right) = q^*_{rm}, \tag{10}$$

where the indexes $m, r \in \{1, \ldots, M\}$ are chosen from

$$S_k \in [m - 1/2, m + 1/2], \quad S_{k-1} \in [r - 1/2, r + 1/2].$$

If S_k or S_{k-1} is outside the interval $[1/2, M + 1/2]$, then $p_c\left(S_k \,|\, S_{k-1}\right) = 0$. Note that the arguments of distributions $p_c\left(S_k \,|\, S_{k-1}\right)$ and $p_c\left(H_k \,|\, H_{k-1}, S_k\right)$ are considered as continuous random variables. Similarly

$$p_c\left(H_k \,|\, H_{k-1}, S_k\right) = \frac{p\left(\varkappa_j \,|\, \varkappa_i, m\right)}{\Delta} = \frac{p^*_{mij}}{\Delta}, \tag{11}$$

where the indexes $i, j \in \{1, \ldots, N\}$, $m \in \{1, \ldots, M\}$ are chosen from $H_k \in \left[\varkappa_j - \Delta/2, \varkappa_j + \Delta/2\right]$, $H_{k-1} \in \left[\varkappa_i - \Delta/2, \varkappa_i + \Delta/2\right]$, $S_k \in [m - 1/2, m + 1/2]$. Then the posterior distribution can be calculated using the Bayesian recursion [26]:

- *prediction step:*

$$p_c\left(S_k \,|\, H_{1:k-1}\right) = \int p_c\left(S_k \,|\, S_{k-1}\right) p_c\left(S_{k-1} \,|\, H_{1:k-1}\right) dS_{k-1}, \tag{12}$$

- *update step:*

$$p_c\left(S_k \,|\, H_{1:k}\right) \propto p_c\left(H_k \,|\, H_{k-1}, S_k\right) p_c\left(S_k \,|\, H_{1:k-1}\right). \tag{13}$$

Taking into account zero-order hold interpolation (10), (11), we obtain that the corresponding posterior pmf is $p(S_k \,|\, H_{1:k}) = p_c\left(S_k \,|\, H_{1:k}\right)$, for all $S_k = 1, \ldots, M$. Then the maximum a posteriori (MAP) and minimum mean squared error (MMSE) estimates are

$$\hat{S}_k^{MAP} = \max_{m=1,\ldots,M} p(m \,|\, H_{1:k}), \quad \hat{S}_k^{MMSE} = \sum_{m=1}^{M} m p(m \,|\, H_{1:k}).$$

The simplest solution to Problem 1 is $\bar{S}_k = \hat{S}_k^{MAP}$ and $n_{i+1} = \min\left\{k > n_i : \hat{S}_k^{MAP} - \bar{S}_{n_i} \neq 0\right\}$. To reduce the number of false detections (or false positives, i.e., switches to the wrong mode, when the estimated mode differs from the current one) we introduce the positive lag N_d and consider the following rule:

$$n_{i+1} = \min\Bigg\{k > n_i + N_d : \left[\hat{S}_{k-N_d}^{MAP} - \bar{S}_{n_i} \neq 0\right] \&$$
$$\left[\operatorname{sign}\left(\hat{S}_{k-N_d}^{MAP} - \bar{S}_{n_i}\right) \frac{1}{N_d + 1} \sum_{j=k-N_d}^{k} \left(\hat{S}_{k-N_d}^{MAP} - \hat{S}_j^{MMSE}\right)\right] < \varepsilon_{tr}\Bigg\}, \tag{14}$$

where $\varepsilon_{tr} > 0$ is a threshold parameter. Then

$$\begin{cases} \bar{S}_{n_{i+1}} = \hat{S}_{k-N_d}^{MAP}, \\ \bar{S}_j = \bar{S}_{n_i} \quad \text{for} \quad j = n_i + 1, \ldots, n_{i+1} - 1. \end{cases} \tag{15}$$

The idea is as follows: if the estimate obtained at $k - N_d$ differs from the current estimated mode $\bar{S}_{n_i}$, then the point $k - N_d$ becomes a candidate to be a switching point with the mode $\hat{S}_{k-N_d}^{MAP}$. Next, we consider the following N_d points and calculate the average distance between the MMSE estimates at these points and the candidate $\hat{S}_{k-N_d}^{MAP}$. If this distance is small, i.e., at the next points the estimate remains sufficiently close to $\hat{S}_{k-N_d}^{MAP}$, we accept it as a new mode estimate. Otherwise, we ignore it and continue detection. The parameters N_d and ε_{tr} are tuning parameters. The complete algorithm for the solution to Problem 1 is summarized in Algorithm 1.

The Bayesian filter takes into account the measurements acquired until the current step and is suitable for online estimation. However, in certain cases, such as post-processing, it may be feasible to leverage future measurements to obtain more precise estimates through Bayesian smoothing. The backward recursive equations for computing the smoothed distributions

[3] Note that in traditional approaches to learning dynamical systems, there is no assumption of nominal stability, and the learning goal includes designing a stabilizing controller. As a result, the cost function typically includes a standard quadratic term $x_k^{\mathrm{T}} \Lambda x_k$ that needs to be minimized. However, this approach can significantly increase the size of the quantized state space and lead to much higher computational complexity. Instead, in (9) we use the quadratic difference between the output values on the sensor and controller sides, i.e., $e_k^{\mathrm{T}} \Lambda \mathrm{e}_k = \left(y_k - \hat{y}_k\right)^{\mathrm{T}} \Lambda \left(y_k - \hat{y}_k\right)$. Since the closed-loop dynamical system is nominally stable, i.e., stable for $\pi_k \equiv 1$, the reduction of e_k leads to a reduction of $x_k^{\mathrm{T}} \Lambda x_k$ as well. In particular, from (2) we can conclude that from $e_k = 0$ it follows that $||x(t)||_2^2 \leq \bar{\beta} \mathrm{e}^{-2\bar{\alpha}(t-t_k)} \, \|x_k\|_2^2$ for all $t \in [t_k, t_{k+1}]$.

[4] Note that the term $\left(1 - \frac{B_k}{\bar{B}}\right)$ characterizes the ''price'' of energy, which depends on the state of the battery: if the battery level is low, then the price is higher, and vice versa.

Algorithm 1 Scenario estimation

```
Input: pmfs $p\left(H_k \,|\, H_{k-1}, S_k\right)$, $p\left(S_k \,|\, S_{k-1}\right)$, and $p(S_0)$; tuning parameters $N_d$ and $\varepsilon_{tr}$;
Output: the estimates $\bar{S}_k$ and the switching instants $n_i$;
    Initialization : $\bar{S}_0 \sim p(S_0)$;
for $k = 1$ to $K$ do
    obtain $\hat{S}_k^{MAP}$ and $\hat{S}_k^{MMSE}$ from the Bayesian filter;
    if the conditions in (14) are fulfilled then
        update $n_{i+1}$ according to (14);
        $\bar{S}_k = \hat{S}_{k-N_d}^{MAP}$;
    else
        $\bar{S}_k = \bar{S}_{n_i}$;
    end if
end for
```

$p(S_k \,|\, H_{1:k+N_s})$ for all $k = K - N_s, \ldots, 1$ are given by the following Bayesian fixed-lag smoothing equations:

$$\begin{aligned} p_c\left(S_k \,|\, H_{1:k+N_s}\right) &= p_c\left(S_k \,|\, H_{1:k}\right) \\ &\times \int \frac{p_c\left(S_{k+1} \,|\, S_k\right)}{p_c\left(S_{k+1} \,|\, H_{1:k}\right)} p_c\left(S_{k+1} \,|\, H_{1:k+N_s}\right) dS_{k+1}. \end{aligned} \tag{16}$$

The corresponding MAP and MMSE estimates can be obtained from the posterior $p_c\left(S_k \,|\, H_{1:k+N_s}\right)$ similarly to the filter.

Now, we consider Problem 2. Without loss of generality, assume that the scenario modes $\{1, \ldots, M\}$ are ordered in order of increasing average harvested energy per slot. Since immediate decisions of adding a new mode are not strictly required, we can use smoother estimates to address this problem. The following heuristic procedure can be proposed. Starting from time $k \geq N_s + T_s$, where $T_s > 0$, we analyze the measurements $H_{k-N_s-T_s:k-N_s}$ and the corresponding scenario estimates $\bar{S}_{k-N_s-T_s:k-N_s}$ obtained from the smoother with the lag N_s. Denote by $\rho_{k-N_s-T_s:k-N_s}$ the energy distribution obtained from the data $H_{k-N_s-T_s:k-N_s}$. If there were more than $\bar{N}$ switches of $\bar{S}_k$ between consecutive modes m and $m+1$ on the interval $[k - N_s - T_s, k - N_s]$, then we assume that $\rho_{k-N_s-T_s:k-N_s}$ corresponds to a new scenario (between m and $m+1$). We add this new scenario to the existing set $\{1, \ldots, M\}$ by assigning it the index $m+1$, and update the state space accordingly by recoding any state that was previously associated with mode $m+1$ or higher and increment the mode index by 1. On the other hand, if $\bar{S}_j = M$ for almost all $j \in [k - N_s - T_s, k - N_s]$, this may potentially mean that $\rho_{k-N_s-T_s:k-N_s}$ corresponds to a new mode higher than M. This can be verified by calculating a statistical distance[5] from $\rho_{k-N_s-T_s:k-N_s}$ to the distribution corresponding to mode M. The above can be summarized in the following Algorithm 2 as a solution to Problem 2.

Algorithm 2 Learning a new scenario

```
Input: pmfs $p\left(H_k \,|\, H_{k-1}, S_k\right)$, $p\left(S_k \,|\, S_{k-1}\right)$, and $p(S_0)$; tuning parameters $N_s$, $T_s$, $\bar{N}$, $\gamma$;
Output: updated scenario state space $\{1, \ldots, M_{new}\}$;
    Initialization : $\bar{S}_0 \sim p(S_0)$;
for $k = N_s + T_s$ to $K$ do
    obtain the estimates $\bar{S}_{k-N_s-T_s:k-N_s}^{MAP}$ from the Bayesian smoother with the lag $N_s$;
    if number of switches $> \bar{N}$ then
        add a new mode;
        continue with $k \leftarrow k + T_s$
    else
        calculate the energy distribution $\rho_{k-N_s-T_s:k-N_s}$ and distances to all known modes
        if minimal distance $> \gamma$ then
            add a new mode; train a new policy;
            continue with $k \leftarrow k + T_s$
        end if
    end if
end for
```

IV. Reinforcement Learning Based Transmission Policies

In this section, we will apply dynamic programming for transmission policy design. We can represent the complete model as a Markov decision process (MDP), where an agent interacts with the environment. At every time slot k, the agent generates an action a_k based on the current state $\mathbf{s}_k$ received from the environment. For our problem, the state $\mathbf{s}_k$ can be defined as follows:

$$\mathbf{s}_k = \left[y_k^{\mathrm{T}}, \hat{y}_{k-1}^{\mathrm{T}}, B_k, H_k, g_k\right]^{\mathrm{T}}.$$

For each fixed mode $S_k \equiv m$ ($m = 1, \ldots, M$), the transition probability matrices $p_m(\mathbf{s}_{k+1} \,|\, \mathbf{s}_k, a_k)$ from $\mathbf{s}_k$ to $\mathbf{s}_{k+1}$ under action a_k can be derived[6] from the models described in Section II. Then the agent observes a cost l_k (a reward in the traditional formulation), which is defined by (9). Note that $\mathbf{s}_k$ contains the term $\hat{y}_{k-1}$ instead of $\hat{y}_k$ since the latter depends on a_k. To overcome this issue, we can rewrite the first term in (9) as $e_k^{\mathrm{T}} \Lambda e_k = (y_k - \hat{y}_k)^{\mathrm{T}} \Lambda (y_k - \hat{y}_k) = (1 - \pi_k)(y_k - \hat{y}_{k-1})^{\mathrm{T}} \Lambda (y_k - \hat{y}_{k-1})$. Thus, the cost l_k is properly defined as a function of $\mathbf{s}_k$ and a_k, i.e.,

$$\begin{aligned} l_k = L(\mathbf{s}_k, a_k) &\triangleq (1 - \pi_k)(y_k - \hat{y}_{k-1})^{\mathrm{T}} \Lambda (y_k - \hat{y}_{k-1}) \\ &+ c_e T_k \left(1 - \frac{B_k}{\bar{B}}\right). \end{aligned} \tag{17}$$

Note that T_k and π_k are defined by (7) and (8), respectively, and depend on $\mathbf{s}_k$ and a_k.

The goal of the RL agent is to find a policy a_k that minimizes the total cost function $\mathbb{E}\left[\sum_{k=1}^{\infty} \delta^k L(\mathbf{s}_k, a_k)\right]$, i.e., solves Problem 3. Then a supervisory control can be used to switch between the policies depending on the current scenario mode, see Fig. 2.

[5] We consider Jensen–Shannon divergence as a statistical distance between two distributions $P_1(x)$ and $P_2(x)$ defined as $\mathrm{JSD}(P_1 || P_2) = \frac{1}{2} D_{\mathrm{KL}}(P_1 || P_3) + \frac{1}{2} D_{\mathrm{KL}}(P_2 || P_3)$, where $D_{\mathrm{KL}}(P_1 || P_2) = \sum P_1(x) \log \frac{P_1(x)}{P_2(x)}$ is Kullback–Leibler divergence and $P_3(x) = \frac{1}{2}(P_1(x) + P_2(x))$.

[6] In order to facilitate numerical computation, we quantize the state space of the dynamical system and the battery level state space, resulting in a final MDP.

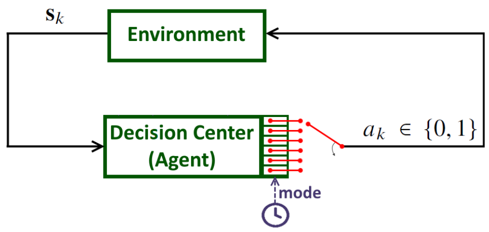


Fig. 2: Markov Decision Process

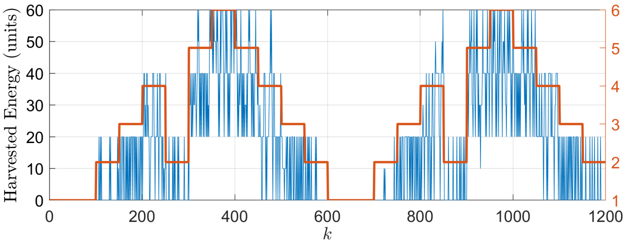


Fig. 3: The energy distribution depending on the mode

A. DP policy

Under the assumption that the transition probabilities $p_m(\mathbf{s}_{k+1} | \mathbf{s}_k, a_k)$ are known, a stationary optimal transmission policy $\mathbf{a}_m$ can be computed offline using exact dynamic programming and the Bellman optimality equation [27]–[29]

$$V_m(\mathbf{s}) = \min_{a\in\{0,1\}} \left[L(\mathbf{s}, a) + \delta \sum_{\mathbf{s}'} p_m(\mathbf{s}' | \mathbf{s}, a) V_m(\mathbf{s}') \right], \quad (18)$$

where V_m is the state-value function. A suboptimal solution of (18) can be numerically found using dynamic programming (DP), e.g., value iteration or policy iteration algorithms, see ch. 4.4–4.5 in [20]. The corresponding stationary policy can then be designed as

$$\mathbf{a}_m(\mathbf{s}) = \operatorname*{argmin}_{a\in\{0,1\}} \left[L(\mathbf{s}, a) + \delta \sum_{\mathbf{s}'} p_m(\mathbf{s}' | \mathbf{s}, a) V_m(\mathbf{s}') \right], \quad (19)$$

where V_m constitutes the solution of (18).

B. Q-Learning based policy

Sometimes, it may not be possible to completely know the transition probability matrix due to factors such as unavailable or inaccurate system parameters, or unknown disturbances in the models. In these cases, the optimal policy can be determined by minimizing a state-action value function, $Q(\mathbf{s}_k, a_k)$. The Q-function is estimated through online experimentation in the environment, using a trial-and-error method. In such cases, the Q-function can be learned using the iterative algorithm referred as Q-learning [19], [20]

$$\begin{aligned} &Q(\mathbf{s}_k, a_k) \leftarrow Q(\mathbf{s}_k, a_k) \\ &+ \alpha_k(\mathbf{s}_k, a_k) \left(l_k + \delta \min_{a\in\{0,1\}} Q(\mathbf{s}_{k+1}, a) - Q(\mathbf{s}_k, a_k) \right). \end{aligned} \quad (20)$$

We will choose the action a_k based on the epsilon-greedy policy, which is a commonly used strategy in Q-learning. It is used to balance exploration (trying new actions) and exploitation (using known actions). In the epsilon-greedy policy, an agent selects the best known action with probability $1-\epsilon$, and selects a random action with probability ϵ, i.e.,

$$a_k = \begin{cases} \text{random} \in \{0,1\}, & \text{with probability } \epsilon, \\ \operatorname*{argmin}_{a\in\{0,1\}} Q(\mathbf{s}_k, a), & \text{with probability } 1-\epsilon, \end{cases} \quad (21)$$

where $\epsilon \in [0,1]$ determines the degree to which the agent will explore the environment rather than relying on its current knowledge. To guarantee convergence of the algorithm, the learning rate $\alpha_k(\mathbf{s}_k, a_k)$ should be chosen such that $\sum_k \alpha_k(\mathbf{s}, a) = \infty$ and $\sum_k \alpha_k^2(\mathbf{s}, a) < \infty$ for all possible $\mathbf{s}$ and a, [20], [30]. The latter guarantees that all state-action pairs are visited infinitely often.

V. Numerical Example

In this section, we illustrate the obtained results by a numerical example. We consider a simplified model of temperature control in a room

$$\dot{T}(t) = \eta_1(T_e - T(t)) + \eta_2(T_g - T(t)) + u(t), \quad (22)$$

where $T(t)$ is the temperature of the room, T_e and T_g are the outside and ground temperatures, respectively, η_1 and η_2 are the thermal conductivity coefficients, $u(t)$ is the control action from the heat source. We consider the following sampled-data feedback law

$$u(t) = \kappa \hat{y}(t_k) + \eta_1(T_d - T_e) + \eta_2(T_d - T_g), \quad (23)$$

where $\hat{y}(t_k) = y(t_k)\pi_k + \hat{y}(t_{k-1})(1-\pi_k)$ is defined from (3) and (8), $y(t) = T(t) - T_d$, and T_d is the desired room temperature. We assume that the sensor sends its measurements to the controller at time instants t_k, and the control gain κ and the sampling step $\bar{h} = t_{k+1} - t_k$ are chosen such that the closed-loop system (22), (23) is exponentially stable for $\pi_k \equiv 1$.

Denote $Y_k = [y_k, \hat{y}_{k-1}]^{\mathrm{T}}$. Then after calculating the solutions of (22), (23) at the sampling points, the resulting system can be rewritten as follows

$$Y_{k+1} = A_1 Y_k + A_2 Y_k \pi_k + [\omega_k, 0]^{\mathrm{T}}, \quad (24)$$

where the additive Gaussian noise $\omega_k \sim \mathcal{N}(0, \bar{\sigma})$ was added to the model, and $A_1 = \begin{bmatrix} \eta & \bar{\eta} \\ 0 & 1 \end{bmatrix}$, $A_2 = \begin{bmatrix} \bar{\eta} & -\bar{\eta} \\ 1 & -1 \end{bmatrix}$, $\eta = \mathrm{e}^{-(\eta_1+\eta_2)h}$, $\bar{\eta} = \frac{1-\eta}{\eta_1+\eta_2}$. Thus, after quantization of the state space, together with (5)–(8) the model (24) can be represented in terms of an MDP

$$\mathbf{s}_{k+1} \sim p_m(\mathbf{s}_{k+1} | \mathbf{s}_k, a_k) \quad (25)$$

for a given mode $m \in \{1, \ldots, M\}$, where $\mathbf{s}_k = [Y_k^{\mathrm{T}}, B_k, H_k, g_k]^{\mathrm{T}}$.

A. Scenario parameter estimation

As was stated in Section IV, the scenario parameters S_k are unknown and have to be estimated based on the measurements of harvested energy. We design the energy distribution (5),

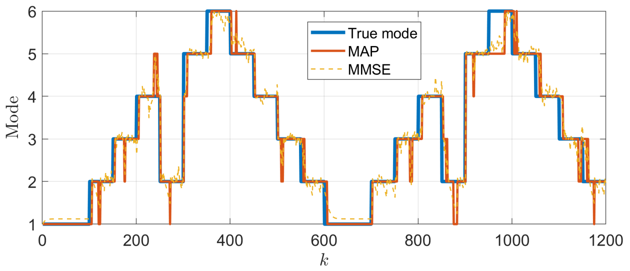


Fig. 4: The estimates of the Bayesian filter

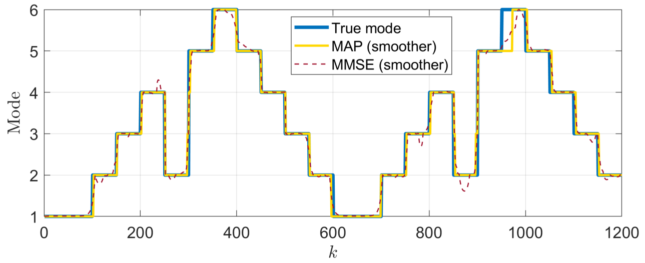


Fig. 5: The estimates of the Bayesian smoother

i.e., the parameters $\left[p^*_{mij}\right]$, using the empirical measurements of a real temperature sensor with a solar cell allowing harvesting of light energy[7]. We consider the following state space $\mathcal{H} = \{0, 10, \dots, 60\}$ (units of energy). Without loss of generality, we assume that we have six different scenarios, i.e., $S_k \in \{1, \dots, 6\}$, where the mode 1 corresponds to the night period (no energy income) and the mode 6 corresponds to a sunny day with the maximum harvesting. The energy distribution depending on the mode is shown in Fig. 3. Since the scenario parameters are slowly varying, we assume that S_k has a quantized and truncated (on the interval $[1, 6]$) Gaussian distribution with the mean S_{k-1} and standard deviation $\sigma_q = 0.23$ (Gaussian random walk). The estimates obtained with the Bayesian filter and smoother (using all the data $H_{1:K}$) are illustrated in Figs. 4 and 5, respectively. We can see that the smoother gives considerably better results, hence, it is more reasonable to use it for post-processing.

Next, we illustrate the performance of Algorithm 1. In Fig. 4 we can see that the filter estimates, $\hat{S}^{MAP}_k$ and $\hat{S}^{MMSE}_k$ contain false detections. Then we apply Algorithm 1 to obtain $\bar{S}_k$ with $\varepsilon_{tr} = 0.3$ and $N_d = 1$ (Fig. 6) and $N_d = 5$ (Fig. 7). Consider the RMSE between the true mode S_k and the estimates $\bar{S}_k$, i.e., $e_{RMS} = \sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(S_k - \bar{S}_k\right)^2}$, and the number of switches $|n_i|$. We see that large N_d reduces the number of switches but increases the RMSE due to the lag. Thus, the choice of the parameters N_d and ε_{tr} in (14) is a trade-off between the estimation error and the number of switches.

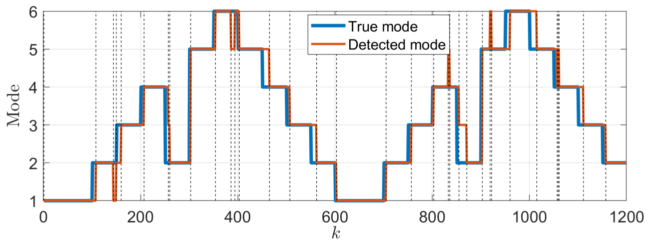


Fig. 6: Algorithm 1 with $\varepsilon_{tr} = 0.3$ and $N_d = 1$, cf. (14). Dashed vertical lines denote switching instants n_i. The RMSE $e_{RMS} = 0.4116$, and the number of switches $|n_i| = 34$

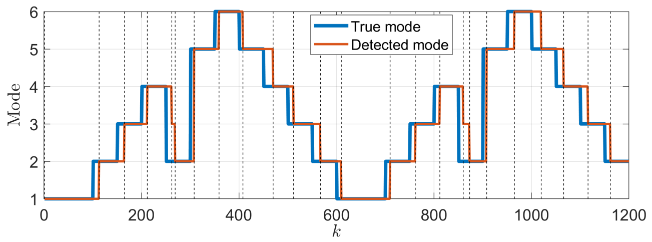


Fig. 7: Algorithm 1 with $\varepsilon_{tr} = 0.3$ and $N_d = 5$, cf. (14). The RMSE $e_{RMS} = 0.6042$, and the number of switches $|n_i| = 24$

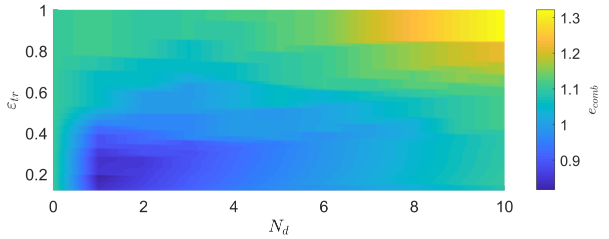


Fig. 8: The dependence of e_{comb} on N_d and ε_{tr} for $c = 0.013$

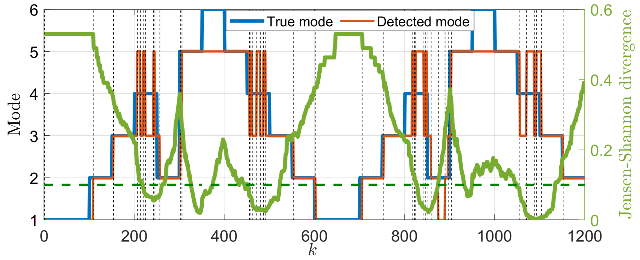


Fig. 9: Algorithm 2 with $T_s = 50$ and the threshold $\gamma = 0.1$. Unknown modes 4 and 6. The algorithm detects multiple switches between modes 3 and 5, indicating that mode 4 needs to be added. To learn mode 6, we use the Jensen–Shannon divergence between $\rho_{k-50,k}$ and the distribution corresponding to mode 5 on intervals where the estimated mode is maximum, i.e., on $[300, 450]$ and $[900, 1050]$. The peak values at $k = 400$ and $k = 1000$ indicate that there is a higher mode (mode 6) on $[350, 400]$ and $[950, 1000]$

[7] It was deployed over three consecutive working days in a typical office building, where energy was harvested both from the sun and fluorescent light, see [16]. During night periods, the sensor spends more energy than it harvests since there is no solar light as well as no fluorescent light (the working day has not yet started). During the day, the sensor can harvest light energy and the battery is charging.

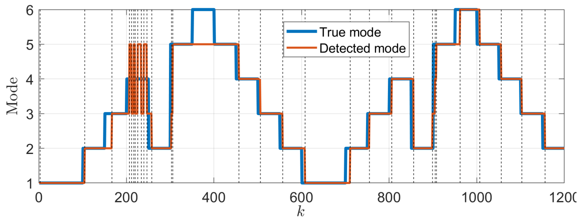


Fig. 10: Algorithms 1 and 2 with unknown modes 4 and 6. At $k = 300$ we detect multiple switches between modes 3 and 5 on $k \in [200, 250]$ and decide to learn mode 4. At $k = 450$ we observe that on $k \in [350, 400]$ the estimated mode poorly fits the observed data implying a large Jensen–Shannon divergence. We then learn mode 6 and add it to the state space. We see that for $k > 450$ all the modes are estimated properly

We then introduce the following combined cost function: $e_{comb} = e_{RMS} + c\,|n_i|$, where a balance constant c defines the trade-off. The dependence of e_{comb} on the parameters N_d and ε_{tr} is illustrated in Fig. 8 for $c = 0.013$, where the optimal values are $N_d = 1$ and $\varepsilon_{tr} = 0.3$.

Finally, we consider Problem 2, which involves learning a new mode, and show the performance of Algorithm 2. We now assume that modes 4 and 6 are unknown, limiting the scenario parameter estimate to values from the set $\{1, 2, 3, 5\}$. In Fig. 9 we can see many consecutive switches between modes 3 and 5, suggesting the presence of a new mode in between. Furthermore, in intervals where the actual scenario is identified as $S_k = 6$, the estimated mode is $\hat{S}_k = 5$ (the highest known). To address this issue, we compute the Jensen–Shannon divergence between $\rho_{k-50,k}$, i.e., the energy distribution obtained from the data $H_{k-50:k}$, and the distribution corresponding to mode 5. We observe that there is an interval (around $k = 400$) where the distance exceeds the threshold parameter $\gamma = 0.1$, with the peak value occurring at $k = 400$. This indicates that the measurements $H_{350:400}$ correspond to a larger mode that must be learned. We illustrate the performance of Algorithms 1 and 2 in Fig. 10.

B. Transmission policies design

We consider the system (24) with the following parameters: $\eta_1 = 0.1, \eta_2 = 0.2$, $T_e = T_g = 0°C$, $T_d = 22°C$, $\kappa = -1.7$, $\bar{h} = 10$ min, $\bar{\sigma} = 0.01$. To represent the model in terms of MDP, we quantize the temperature state space as it is shown in Fig. 11, where a more fine grained quantizer is used for values close to the desired temperature T_d. Also, we assume that $\bar{B} = 1000$ and the quantization step of the battery state space is 20. Suppose that the values of transmission energy comes from the set $\{5, 10, 15, 20, 25\}$ (units of energy) with the probabilities $\mathrm{p} = [0.0855, 0.6180, 0.2592, 0.0338, 0.0035]$, corresponding to the quantized compound distribution (6) with $\tilde{m} = 1$, $\tilde{\sigma} = 4$, and a resolution 10 dBm, see [16]. For each mode $r = 1, \dots, 6$, we implement the value iteration algorithm to find a solution to the Bellman equation (18). We will refer to this policy as the DP policy. In the algorithm, we used $\delta = 0.95$, $c_e = 0.3$, and the number of iterations was chosen such that the updated values of V_r differed from the previous ones by no more than 10^{-3}. The projections of the obtained DP policy to the Y_k-axis are illustrated in Fig. 11 for different modes r. We can see that the transmission policy is more energy-saving for lower modes.

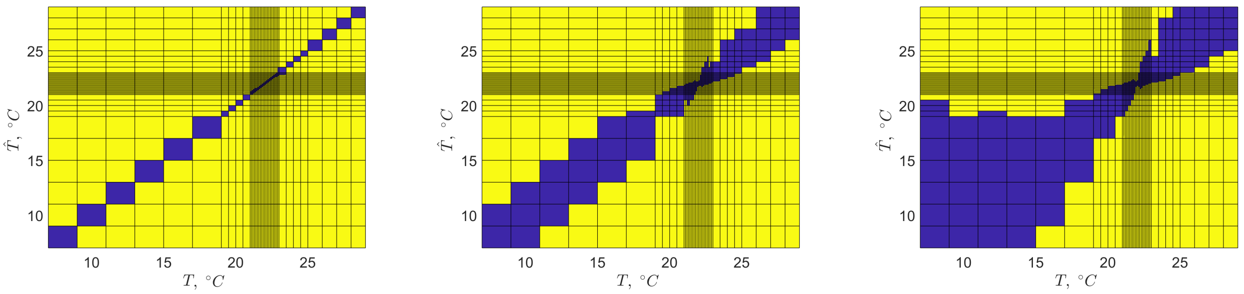
In Fig. 12, we compare the DP policy with random transmissions. The blue trajectories illustrate the temperature and battery behavior for $\beta = 0.5$, where β is the transmission probability. We can see that the control performance is appropriate when the battery is not empty. However, due to high energy consumption, there are periods when the battery is discharged, and the controller information cannot be updated. If we decrease β to 0.1 (the purple trajectories), this leads to fewer transmissions and good battery behavior but simultaneously to poor temperature control. The red trajectories illustrate the DP policy. We can note that the battery usage is similar to $\beta = 0.1$, which also means a similar number of total transmissions. However, since these transmissions are used in an optimal way, the temperature control performance remains suitable.

When implementing Q-learning, i.e., online learning, we do not train the policy separately for each mode m, since a majority of states $\mathbf{s}$ with low probability cannot be visited in practice with a finite number of iterations. Instead, we allow the algorithm to learn the system behavior under changing modes (scenarios), where the main objective is to understand how to conserve energy when harvesting is high in order to use the saved energy for lower modes. In the algorithm (20)–(21), we used 200000 iterations, $\epsilon = 0.02$, $\alpha_k(\mathbf{s}, a) = \frac{1}{1+N_{(\mathbf{s},a)}}$, where $N_{(\mathbf{s},a)}$ is a number of visits of the state-action pair $(\mathbf{s}, a)$. The results of Q-learning are illustrated in Fig. 13. We can see that the battery behavior is similar or even slightly better than for the DP policy[8] in Fig. 12. However, if we change the switching algorithm of scenario modes or move the system to another state which is unlearned by the Q-policy, then the system behavior becomes poor and the learning process starts almost from the beginning.

VI. Conclusions

We have applied the reinforcement learning approach to design suitable transmission policies providing both good control performance and efficient energy consumption. The DP policy based on the solution of the Bellman equation requires knowledge of the system model and a large number of iterations since it consequently covers all elements from the state space. However, it allows to train the policy offline for all possible states and scenario modes, and then use a supervisor to switch between the sub-policies according to the mode change, making the DP policy more flexible and versatile. If the system model is unknown, then the Q-learning approach can be used, where we estimate the Q-function through experimentation in the environment, i.e., by the trial-and-error method. The Q-policy shows good results when the system is run under normal circumstances. At the same time, if the system state is placed in conditions under which the algorithm has not been trained by experimentation, its

[8] The reason is that a finite number of iteration for the DP algorithm was used as well as it was trained for each mode separately and did not take into account the mode change model, which is unknown.



(a) Mode 6 (high energy income)

(b) Mode 3 (medium energy income)

(c) Mode 1 (low energy income)

Fig. 11: Projections of DP policies to Y_k-axis for different modes. Yellow color: $a_k = 1$ (transmit), blue color: $a_k = 0$ (not transmit). The desired temperature T_d is 22°C. The transmission policy is more energy-saving for lower modes

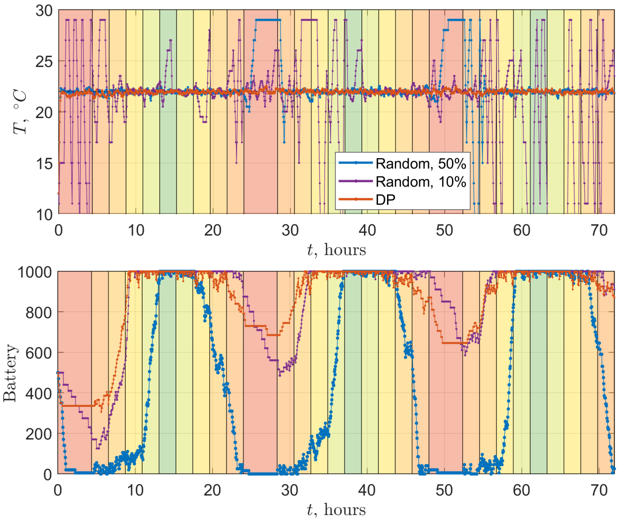


Fig. 12: DP policy versus random policies. The colored areas represent different modes, where a change from red to green corresponds to a mode change from $m = 1$ to $m = 6$. Blue: random policy with 50% transmission probability. Temperature variance is 1.06. High energy consumption discharging the battery (battery mean is 392). Purple: random policy with 10% transmission probability. Good battery behavior (battery mean is 904) and poor control accuracy (temperature variance is 4.26). Red: DP transmission policy. Both good battery behavior (battery mean is 946) and control accuracy (temperature variance is 0.28).

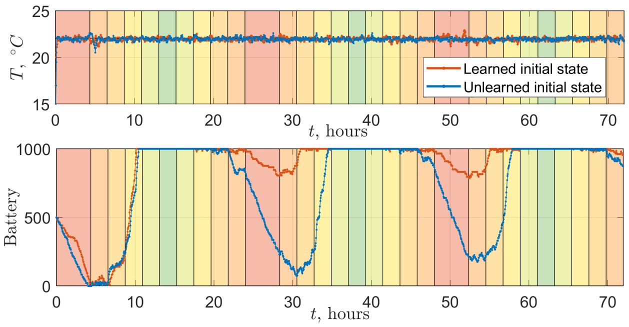


Fig. 13: Q-learning based transmission policy. Red: the system is iterated from a learned state. The performance is similar to one when using the DP policy. Blue: the system is iterated from an unlearned state, resulting in performance degradation

performance becomes poor, requiring a new learning process. Therefore, the final choice of the appropriate policy depends on the capabilities and requirements of the system.

We have also considered the Bayesian filter and smoother technique for estimating unknown scenario parameters based on measurements of harvested energy, as well as detecting the time instants of scenario changes. We have introduced Algorithm 1 reducing the number of false switches, which uses MMSE estimates of the filter posterior probability. Finally, we have also proposed a heuristic Algorithm 2 that solves the problem of adding a new mode to the scenario state space when none of the existing modes fit the measured data properly. For the latter, the Jensen–Shannon divergence has been used as a measure of the distance between distributions.

Future research directions may include investigating combinations of the proposed transmission policies with other control-oriented policies that consider various network constraints and imperfections. Moreover, with the increasing adoption of smart grids and Internet of Things devices, the risk of cyber attacks on the power grid is becoming more significant. Thus, addressing cybersecurity issues in energy-based transmission policies and developing solutions to mitigate potential threats could also be a promising avenue for future research.

References

[1] V. Sharma, U. Mukherji, V. Joseph, and S. Gupta, "Optimal energy management policies for energy harvesting sensor nodes," *IEEE Trans. Wireless Commun.*, vol. 9, pp. 1326–1336, 2010.

[2] C. Ho and R. Zhang, "Optimal energy allocation for wireless communications with energy harvesting constraints," *IEEE Trans. Signal Process.*, vol. 60, pp. 4808–4818, 2012.

[3] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 4, pp. 32–es, 2007.

[4] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: Survey and implications," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 3, pp. 443–461, 2011.

[5] J. Taneja, J. Jeong, and D. Culler, "Design, modeling, and capacity planning for micro-solar power sensor networks," in *International Conference on Information Processing in Sensor Networks*, 2008, pp. 407–418.

[6] V. Leonov, "Thermoelectric energy harvesting of human body heat for wearable sensors," *IEEE Sensors Journal*, vol. 13, no. 6, pp. 2284–2291, 2013.

[7] H. Kulah and K. Najafi, "Energy scavenging from low-frequency vibrations by using frequency up-conversion for wireless sensor applications," *IEEE Sensors Journal*, vol. 8, no. 3, pp. 261–268, 2008.

[8] H. S. Kim, J.-H. Kim, and J. Kim, "A review of piezoelectric energy harvesting based on vibration," *International Journal of Precision Engineering and Manufacturing*, vol. 12, 12 2011.

[9] S. Kim, R. Vyas, J. Bito, K. Niotaki, A. Collado, A. Georgiadis, and M. M. Tentzeris, "Ambient RF energy-harvesting technologies for self-sustainable standalone wireless sensor platforms," *Proceedings of the IEEE*, vol. 102, no. 11, pp. 1649–1666, 2014.

[10] Y. H. Bae and J. W. Baek, "Sensing strategy exploiting channel memory in CR network with RF energy harvesting," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2539–2542, 2018.

[11] C. Park and P. H. Chou, "AmbiMax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes," in *3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, vol. 1, 2006, pp. 168–177.

[12] M. Ku, Y. Chen, and K. J. R. Liu, "Data-driven stochastic models and policies for energy harvesting sensor communications," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 8, pp. 1505–1520, 2015.

[13] N. Dang, R. Valentini, E. Bozorgzadeh, M. Levorato, and N. Venkatasubramanian, "A unified stochastic model for energy management in solar-powered embedded systems," in *2015 IEEE/ACM Int. Conf. Computer-Aided Design (ICCAD)*, 2015, pp. 621–626.

[14] M. Y. Naderi, S. Basagni, and K. R. Chowdhury, "Modeling the residual energy and lifetime of energy harvesting sensor nodes," in *2012 IEEE Global Commun. Conf. (GLOBECOM)*, 2012, pp. 3394–3400.

[15] C. K. Ho, P. D. Khoa, and P. C. Ming, "Markovian models for harvested energy in wireless communications," in *12th IEEE Int.l Conf. Commun. Syst.*, Singapore, 2010, pp. 311–315.

[16] R. Seifullaev, S. Knorn, and A. Ahlén, "Event-triggered transmission policies for harvesting powered sensors with time-varying models," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 4, pp. 2139–2149, 2021.

[17] P. Agrawal, A. Ahlén, T. Olofsson, and M. Gidlund, "Long term channel characterization for energy efficient transmission in industrial environments," *IEEE Trans. Commun.*, vol. 62, no. 8, pp. 3004–3014, 2014.

[18] T. Olofsson, A. Ahlén, and M. Gidlund, "Modeling of the fading statistics of wireless sensor network channels in industrial environments," *IEEE Trans. Signal Process.*, vol. 64, no. 12, pp. 3021–3034, 2016.

[19] C. Watkins, "Learning from delayed rewards," Ph.D. dissertation, University of Cambridge, Cambridge, UK, 1989.

[20] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[21] R. Seifullaev and A. Fradkov, "Robust nonlinear sampled-data system analysis based on Fridman's method and S-procedure," *International Journal of Robust Nonl. Control*, vol. 26, pp. 201–217, 2016.

[22] R. Seifullaev, S. Knorn, and A. Ahlén, "Event-triggered control of systems with sector-bounded nonlinearities and intermittent packet transmissions," *Automatica*, vol. 146, p. 110651, 2022.

[23] A. Ahlén, J. Åkerberg, M. Eriksson, A. J. Isaksson, T. Iwaki, K. H. Johansson, S. Knorn, T. Lindh, and H. Sandberg, "Toward wireless control in industrial process automation: A case study at a paper mill," *IEEE Control Syst. Mag.*, vol. 39, pp. 36–57, 2019.

[24] S. Knorn, S. Dey, A. Ahlén, and D. Quevedo, "Optimal energy allocation in multisensor estimation over wireless channels using energy harvesting and sharing," *IEEE Trans. Autom. Control*, vol. 64, pp. 4337–4344, 2019.

[25] R. Seifullaev, S. Knorn, and A. Ahlén, "The effect of uniform quantization on parameter estimation of compound distributions," *IEEE Control Syst. Lett.*, vol. 3, no. 4, pp. 1032–1037, Oct 2019.

[26] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

[27] R. Bellman, *Dynamic Programming*. Princeton Univ. Press, 1957.

[28] D. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 1995, vol. 2.

[29] ——, *Reinforcement Learning and Optimal Control*. Athena Scientific, 2019.

[30] K. Prabuchandran, S. Meena, and S. Bhatnagar, "Q-learning based energy management policies for a single sensor node with finite buffer," *IEEE Wireless Commun. Lett.*, vol. 2, no. 1, pp. 82–85, 2013.

Ruslan Seifullaev received the Diploma (M.Sc.) degree in theoretical cybernetics and the Candidate of Science (Ph.D.) degree in discrete mathematics and mathematical cybernetics from St. Petersburg University in 2012 and 2016, respectively. From 2014 to 2018, he was employed as a Research Engineer at the Institute of Problems in Mechanical Engineering of Russian Academy of Sciences. In 2023, he received the Ph.D. degree in signal processing from the Division of Signals and Systems, Uppsala University, Sweden. He is currently a Postdoctoral Researcher at the Division of Systems and Control, Uppsala University. His research interests include nonlinear control theory, networked control systems, time-delay systems, security of cyber-physical systems, energy harvesting.

Steffi Knorn received the Dipl.Ing. in 2008 from the University of Magdeburg, Germany, and the Ph.D. from the Hamilton Institute at the National University of Ireland Maynooth, Ireland, in 2013. In 2013 she was a research academic at the Centre for Complex Dynamic Systems and Control at the University of Newcastle, Australia. Since 2014 she is with the Signals and Systems Division at Uppsala University, Sweden. Between 2019 and 2021 she was an assistant professor at Otto von Guericke University Magdeburg. Since 2021 she is a full professor for control at Technische Universitä Berlin, Germany. Dr. Knorn's research interests include stability analysis and controller design for marginally stable two-dimensional systems, port-Hamiltonian systems, string stability and scalability of vehicle platoons, distributed control, networked control, multi-sensor estimation, and energy harvesting and energy sharing in wireless networks.

Anders Ahlén is senior professor in Signal Processing at Uppsala University. Previously, from July 1996 - May 2022, he was the head of the Signals and Systems Division and held the chair in Signal Processing at the same university. He was born in Kalmar, Sweden, and received the PhD degree in Automatic Control from Uppsala University. He was with the Systems and Control Group, Uppsala University from 1984-1992 as an Assistant and Associate Professor in Automatic Control. During 1991 he was a visiting researcher at the Department of Electrical and Computer Engineering, The University of Newcastle, Australia. He has been a visiting professor at the same university several times since 2008. In 1992 he was appointed Associate Professor of Signal Processing at Uppsala University. During 2001-2004 he was the CEO of Dirac Research AB, a company offering state-of-the-art audio signal processing solutions. He has been a member of the Board-of-Directors since 2005 and during 2005-2020 he was the chairman of the same company. His research interest, which includes Signal Processing, Communications and Control, is currently focused on Signal Processing and Machine Learning, Wireless Sensor Networks, Wireless Control, Security and Privacy of Cyber Physical Systems, and Audio Signal Processing. From 1998 to 2004 he was the Editor of Signal and Modulation Design for the IEEE Transactions on Communications.

Roland Hostettler received the Dipl. Ing. degree in electrical and communication engineering from Bern University of Applied Sciences, Switzerland, in 2007, the MSc degree in electrical engineering and the PhD degree in automatic control from Luleå University of Technology, Sweden, in 2009 and 2014, respectively. He has held postdoctoral researcher positions at Luleå University of Technology, Sweden and Aalto University, Finland. Currently, he is associate professor with the Department of Electrical Engineering, Uppsala University, Sweden. He is a member of the IEEE machine learning for signal processing technical committee. His research interests include statistical signal processing and sensor fusion and their applications.