# Reward Function Evaluation in a Reinforcement Learning Approach for Energy Management

Yohann Rioual, Yannick Le Moullec, Johann Laurent, Muhidul Islam Khan, Jean-Philippe Diguet

# Reward Function Evaluation in a Reinforcement Learning Approach for Energy Management

Yohann Rioual[*+], Yannick Le Moullec[*], Johann Laurent[+], Muhidul Islam Khan[*] and Jean-Philippe Diguet[+]
[+] CNRS UMR6285 - Lab-STICC, Univ. Bretagne Sud, F-56100 Lorient, France
Email: {firstname.lastname}@univ-ubs.fr
[*] Thomas Johann Seebeck Department of Electronics - Tallinn University of Technology, Tallinn, Estonia
Email: yannick.lemoullec@ttu.ee, mdkhan@ttu.ee

***Abstract*—In the past decade, the energy needs in WBANs have increased due to more information to be processed, more data to be transmitted and longer operational periods. On the other hand, battery technologies have not improved fast enough to cope with these needs. Thus, miniaturized energy harvesting technologies are increasingly used to complement the batteries in WBANs. However, this brings uncertainties in the system since the harvested energy varies a lot during the node operation. It has been shown that reinforcement learning algorithms can be used to manage the energy in the nodes since they are able to make decisions under uncertainty. But the efficiency of these algorithms depend on their reward function. In this paper we explore different reward functions and seek to identify the most suitable variables to use in such functions to obtain the expected behavior. Experimental results with four different reward functions illustrate how the choice thereof impacts the energy consumption of the nodes.**

***Index Terms*—reinforcement learning, WBAN, reward function, Q-learning, energy management**

## I. Introduction

The rapid growth of interest in physiological sensors, low-power integrated circuits, and wireless communication has enabled a new generation of wireless sensor networks, called Wireless Body Area Networks (WBANs). WBANs consist of a number of intelligent nodes fitted on the body to monitor physiological parameters of the wearer. Advances in microelectronics lead to significant miniaturization of sensors; however, battery technology has not improved at the same rate [1].

To minimize the battery's size, an increasingly popular approach is to harvest energy directly from the environment [2]. Miniaturized energy harvesting technologies can not harvest a lot of energy (see Table I), but they can be used as complement to the battery. However, such energy sources vary greatly over time and bring uncertainties in the system. Reinforcement learning (RL) algorithms have acquired a certain popularity in recent years ( [3], [4] ...) because they can handle such uncertainty in energy sources and appear to be a valid solution for energy management. They adapt the node's behavior by rewarding good decision using a reward function. However, it can be difficult to choose the most suitable reward function; since this function determines the behavior of the system, choosing it is an essential task for the system designer. However, the literature on this topic rarely discusses the choice of the reward function.

TABLE I
Power density of energy harvesting technologies

| Harvesting technologies | Power density |
|---|---|
| Solar cell (outdoors at noon) | 15 $\mathrm{mW/cm^2}$ |
| Wind flow (at 5 $\mathrm{m/s}$) | 16.2 $\mu\mathrm{W/cm^3}$ |
| Vibration (Piezoelectric – shoe insert) | 330 $\mu\mathrm{W/cm^3}$ |
| Vibration (electromagnetic conversion at 52 Hz) | 306 $\mu\mathrm{W/cm^3}$ |
| Thermoelectric (5 °C gradient) | 40 $\mu\mathrm{W/cm^3}$ |
| Acoustic noise (100 dB) | 960 $\mathrm{nW/cm^3}$ |

Thus, in this paper, we explore the influence of different reward functions used in a popular RL algorithm, i.e. Q-learning. We also propose an approach for deciding on the appropriate reward function and parameters in order to maximize the battery's autonomy.

The remainder of the paper is structured as follows. Section 2 discusses the related work and Section 3 presents our use case. Section 4 introduces the RL mechanism and presents the proposed decision approach. Section 5 presents our experimental results wherein four reward functions are evaluated and compared. Lastly, section 6 summarises and concludes the paper.

## II. Related work

Works on energy management typically focus on the consumption of the radio, often disregarding the power consumption of other parts of the system. In cases where energy harvesting technologies are employed, related works have proposed adaptive protocols that deal with the challenge of providing the required quality of service under uncertainty in the energy input [5].

To extend a sensor node lifespan in WBANs, there exist different energy management methods. The work presented in [6] adapts the node's consumption according to the activity of the wearer. They propose a new classifier that detects the person's activity and adapts the operating policy accordingly.

RL is an approach to take decisions under uncertainty, which makes it suitable for energy management in systems where energy harvesting technologies are used. [7] have developed a power control approach in WBANs based on RL. This approach provides a substantial saving in energy consumption per bit. However, they only focus on the wireless communication.

There are few papers about WBANs that deal with harvesting energy and RL for the energy management of the entire node. Most of these papers do not explain the process for choosing the reward function; for example, [7] shows good results with one reward function but without explaining if they tried different reward functions.

## III. Use Case

In this work, we want to manage the energy consumption of a sensor node fitted on the chest to monitor the cardiac activity for non medical application using an ECG sensor. Each measurement lasts 10 seconds, and data is sent to a smartphone used as base station using Bluetooth Low Energy (BLE) transmitter. The node does not continuously monitor the cardiac activity; after each measurement it enters a sleep mode to minimize energy consumption. The period between each measurement is variable and lasts from 10 to 60 minutes.

TABLE II
Node components and respective current consumption

| Component | Active mode | Sleep mode |
|---|---|---|
| ECG sensor | 1.6 mA | 0.12 mA |
| Microcontroller | 129 µA/MHz | 0.78 µA |
| BLE transmitter | 21.1 mA | 0 A |

Our node features a kinetic motion energy harvester used as a complement to the battery. This energy harvester is presented in [8]. The harvested energy is low but it still can extend the node lifespan; Table III shows how much energy can be harvested according to the activity of the wearer. These data were extracted from [8].

TABLE III
Kinetic motion's harvested energy for three different activities

| Activity | Power generation |
|---|---|
| relaxing | 2.4 µW |
| walk | 180.3 µW |
| run | 678.3 µW |

We use the dominant frequency of motion, $F_m$, to determine which activity is performed by the wearer. We obtain $F_m$ by determining the maximum spectral component of the Fourier Transform of the acceleration $a(t)$. Since the harvested energy is uncertain, we use an RL approach to manage the node's consumption by adjusting its sleep duration.

## IV. Reinforcement Learning

In this section, we give an overview of RL, we present the selected Markov decision process used to deal with the energy management process and we introduce the selected RL algorithm, i.e. Q-learning.

### A. Overview of Reinforcement Learning

RL is a formal framework that models sequential decision problems [9], in which an agent learns to make better decisions by interacting with the environment (fig. 1). When the agent performs an action, the state changes and the agent receives a reinforcement called a reward, which indicates the quality of the transition. The agent's goal is to maximize its total reward over the long term.

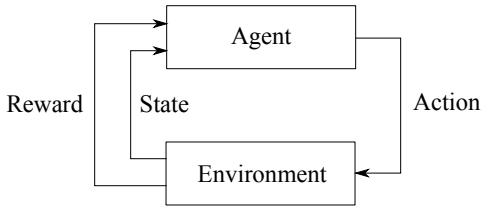


Fig. 1. Interaction between an agent and its environment

There is a trade-off between exploration and exploitation in RL. Exploration chooses an action randomly in the system to find out the utility of that action. Exploitation deals with the actions which have been chosen based on the previously learned utility of the actions. A heuristic is used where the exploration probability at any point of time is given in [10]:

$$\epsilon = min(\epsilon_{max}, \epsilon_{min} + k \times (S_{max} - S)/S_{max}) \quad (1)$$

where $\epsilon_{max}$ and $\epsilon_{min}$ denote upper and lower boundaries for the exploration factor, respectively. $S_{max}$ represents the maximum number of states which is three in our work and $S$ represents the current number of states already known. At each time step, the system calculates $\epsilon$ and generates a random number in the interval $[0, 1]$. If the selected random number is less than or equal to $\epsilon$, the system chooses a uniformly random task (exploration), otherwise it chooses the best task using $Q$-values (exploitation).

### B. Markov Decision Process for Energy Management

We can model the energy management problem as a Markov decision process (MDP). An MDP provides a mathematical framework for modeling decision making in situations where outcomes are partly random and partly under the control of a decision maker. An MDP is formally defined as a n-uplet $< S, A, T, R >$ where $S$ is a state space, $A$ a set of possible actions, $T : S \times A \times S \rightarrow [0, 1]$ are the transition's probability between states ($T(s, a, s') = p(s'|a, s)$ is the probability to reach the state $s'$ starting from $s$ after taking the action $a$) and $R : S \times A \rightarrow R$ is a reward signal.

In this work, we define a set of actions with different processor frequencies ($F_p$) and periods between each measurement ($P_s$) (Table IV). For instance, action 1 had a processor frequency of 80MHz and a measurement every minute, whereas action 3 has a processor frequency of 10MHz and a measurement every 5 minutes. All these actions have different energy consumption levels since they depend on the processor's frequency in active mode and its consumption in sleep mode (see the second row in Table II).

Our state space is divided into three different states. We use $F_m$ which is correlated with the energy we harvest to consider our state; a high value of $F_m$ corresponds to more energy being harvested and a low value of $F_m$ correspond to less energy

TABLE IV
SET OF ACTIONS WITH DIFFERENT PROCESSOR FREQUENCIES ($F_p$) AND PERIODS BETWEEN EACH MEASUREMENT ($P_s$)

| Action | $F_p$ | $P_s$ |
|---|---|---|
| 1 | 80 MHz | 1 min |
| 2 | 10 MHz | 1 min |
| 3 | 10 MHz | 5 min |
| 4 | 10 MHz | 20 min |
| 5 | 5 MHz | 60 min |

being harvested. The considered state uses the value of $F_m$ and corresponds to an activity. The activity can be considered high (i.e. running) if $F_m > 2\text{Hz}$ , moderate (i.e. walking) if $2\text{Hz} \geq F_m > 1\text{Hz}$ or low (i.e. relaxing) if $1\text{Hz} \geq F_m$.

We do not have a transition's probability between the states $T$, so we use a model-free algorithm. Model-free algorithms works even when we do not have a precise model of the environment; these algorithms primarily rely on learning algorithms such as Q-learning which is described in the next section.

### C. Q-learning algorithm

In this section we present the Q-Learning algorithm [11]. The Q-learning algorithm is widely used since it is simple to implement yet effective and its convergence is proven. So we use this algorithm for the energy management of a sensor node in combination with the MDP presented above.

**Algorithm 1** Q-learning algorithm

Initialize $Q(s, a)$ arbitrarily
The agent observes the initial state $s_0$
Initialize s
**for** each decision epochs **do**
  Choose $a$ from $s$ using policy derived from Q
  Take action $a$, observe the new state $s'$ and the associated reward $r$
  $Q(s, a) \leftarrow Q(s, a) + \alpha(r + \gamma \max_{a'} Q(s', a') - Q(s, a))$
  $s \leftarrow s'$
**end for**

*Learning rate $\alpha$:* The learning rate $\alpha$ determines how fast the new information will surpass the old one. A factor of 0 would not teach the agent in question anything, whereas a factor of 1 would only teach the agent the latest information. In our work, we decrease slowly the learning rate $\alpha$ in such a way that it reflects the degree to which a state-action pair has been chosen in the recent past. It is calculated as:

$$\alpha = \frac{\zeta}{visited(s, a)} \tag{2}$$

where $\zeta$ is a positive constant and $visited(s, a)$ represents the visited state-action pairs so far [12].

*Discount factor $\gamma$:* The discount factor $\gamma$ determines the importance of future rewards. A factor of 0 would make the agent myopic by considering only current rewards, while a factor close to 1 would also involve more distant rewards. If the discount factor is close or equal to 1, the value of Q may never converge.

Using the MDP, we try to identify the best parameters to use as variables in the reward function to adapt the energy consumption according to the energy we can harvest. In the following section, we present some results and identify those parameters.

## V. EXPERIMENTAL RESULTS

First of all, it should be noted that the harvesting capabilities are not sufficient to recharge the sensor node's battery. So we seek and expect to reduce the node's consumption when the harvested energy is low. We test four different reward functions to identify which parameters influence correctly our system's behavior.

There are different constraints when designing the system and most of them are conflicting; for example, keeping the sleep period at a minimum while also reducing energy consumption. The main purpose of the RL algorithm is to find the balance point to respect the constraints. For the first two reward functions, we use a parameter $\beta$ to balance the equilibrium point according to what is considered most important [10].

The first reward function tries to balance the conflicting objectives between the sleep duration $P_s$ and the energy consumption of the system. $B_r(t)$ is the residual energy in the battery's node at time $t$.

$$R = \beta * \frac{\min(P_s)}{P_s} + (1 - \beta) * (B_r(t) - B_r(t-1)) \tag{R1}$$

The second reward function is similar to the first one but instead of using the energy consumption, it only uses the residual energy of the battery's node at time $t$.

$$R = \beta * \frac{\min(P_s)}{P_s} + (1 - \beta) * \frac{B_r(t)}{B_{max}} \tag{R2}$$

The third reward function does not consider the sleep duration $P_s$ but only the energy consumption. Indeed, the objective is to find the less consuming functioning mode according the energy we can harvest.

$$R = B_r(t) - B_r(t-1) \tag{R3}$$

The last reward function tries to balance $P_s$ and the residual energy but this function do not have a parameter $\beta$ to balance the objectives. Instead it is a product of both parameters, sleep duration $P_s$ and residual energy $B_r(t)$.

$$R = P_s \times B_r(t) \tag{R4}$$

We simulate the evolution of the battery's charge level. Each 30 minutes, the activity changes and each 20 minutes the algorithm chooses an action. Figure 2 shows the average action taken by the algorithm (i.e. across the values given in Table IV) according to the activity identified with the dominant frequency of motion, $F_m$. Higher average values correspond to less consuming actions and lower average values correspond to more consuming actions. Moreover, $\beta$ is fixed at 0.3 since

our primary goal is to adapt the node's consumption, i.e. we give more importance to the energy factor. The results show that the choice of the reward function has a significant impact on the results; while some reward functions yield the expected behavior, others adapt poorly to the activity and others do not yield the correct behavior at all, as discussed in what follows.


Fig. 2. Average functioning state according to the activity and the reward function. Higher average values correspond to less consuming actions

Reward functions (R1) and (R2) increase the node's consumption when the harvested energy increases whereas reward function (R4) increases the node's consumption when it harvests a moderate amount of energy but decreases the node's consumption when it harvests a lot of energy. This is due to the fact that reward function (R4) is more influenced by the sleep time $P_s$ than by the consumption of the sensor node. Reward function (R3) does not make any difference between the activities. Reward functions (R3) and (R4) are not suitable to manage correctly the energy in a sensor node. The best reward function are those that use a parameter $\beta$.

With the same value for the parameter $\beta = 0.3$, the results are very similar. However, the residual energy achieves the lowest energy consumption. These reward functions allow us to manage the importance given to the energy consumption according to the application requirements by increasing or decreasing the value of $\beta$.

## VI. Conclusions and Perspectives

The development of harvesting technologies has led to new energy management algorithms. The RL approach is a valuable solution for this kind of problem. However, to apply it, we need to tune several aspects such as the trade-off between exploration and exploitation, the value of the learning rate, and the definition of the reward function.

The choice of a reward function is important since it influences the behavior of the node. Its choice must be justified and not just stated as an experimental parameter. We conducted a series of simulations to identify the best reward function, and we found out that the proposed reward functions (R1) and (R2) that include an balancing parameter are better able to find the balance to give between performance and consumption in the context of energy management.

On the other hand, the reward functions (R3) and (R4) did not allow a good energy management. Reward function (R3) did not make a connection between the node's consumption and the energy harvested. Reward function (R4) did a connection but failed to choose less consuming actions when the harvested energy is low.

Reward functions (R1) and (R2) succeeded to choose less consuming actions when the harvested energy is low. The choice between the battery's residual energy or the consumption does not seem to make any particular difference, the choice must be made according to the context of the application, e.g. if we want to reduce consumption at certain times (no recharging capabilities) or if we want to control the charge and discharge of the battery.

Future work includes the validation of the simulated approach in practice. However, the sensor nodes have more tunable knobs and can present more parameters to balance (e.g. several sensors with different importance, several concurrent applications running on the same node), which might require adjustment to the reward functions.

## Acknowledgement

This work was funded in part by CNRS research group ISIS, in part by Digital Division of Université Bretagne Sud and in part by a DoraPlus grant. This project has received funding from the European Unions Horizon 2020 research and innovation program under grant agreement No 668995.

## References

[1] X. Yuan, H. Liu, and J. Zhang, *Lithium-ion batteries: advanced materials and technologies*. CRC press, 2011.

[2] G. Zhou, L. Huang, W. Li, and Z. Zhu, "Harvesting ambient environmental energy for wireless sensor networks: a survey," *Journal of Sensors*, vol. 2014, 2014.

[3] S. Shresthamali, M. Kondo, and H. Nakamura, "Adaptive power management in solar energy harvesting sensor node using reinforcement learning," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 16, no. 5s, p. 181, 2017.

[4] T. Ahmed and Y. Le Moullec, "A qos optimization approach in cognitive body area networks for healthcare applications," *Sensors*, vol. 17, no. 4, p. 780, 2017.

[5] E. Ibarra, A. Antonopoulos, E. Kartsakli, J. J. Rodrigues, and C. Verikoukis, "Qos-aware energy management in body sensor nodes powered by human energy harvesting," *IEEE Sensors Journal*, vol. 16, no. 2, pp. 542–549, 2016.

[6] F. Casamassima, E. Farella, and L. Benini, "Context aware power management for motion-sensing body area network nodes," in *Proceedings of the conference on Design, Automation & Test in Europe*. European Design and Automation Association, 2014, p. 170.

[7] R. Kazemi, R. Vesilo, and E. Dutkiewicz, "Dynamic power control in Wireless Body Area Networks using reinforcement learning with approximation," *2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 2203–2208, 2011.

[8] M. Gorlatova, J. Sarik, G. Grebla, M. Cong, I. Kymissis, and G. Zussman, "Movers and shakers: Kinetic energy harvesting for the internet of things," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 1. ACM, 2014, pp. 407–419.

[9] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[10] M. I. Khan and B. Rinner, "Energy-aware task scheduling in wireless sensor networks based on cooperative reinforcement learning," in *Communications Workshops (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 871–877.

[11] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[12] R. Bianchi, C. Ribeiro, and A. Costa, "Advances in artificial intelligence," 2004.