This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

LoRa-RL: Deep Reinforcement Learning for Resource Management in Hybrid Energy LoRa Wireless Networks

Rami Hamdi, *Member, IEEE,* Emna Baccour, Aiman Erbad, *Senior Member, IEEE,* Marwa Qaraqe, *Member, IEEE,* and Mounir Hamdi, *Fellow, IEEE*

***Abstract*—LoRa wireless networks are considered as a key enabling technology for next generation internet of things (IoT) systems. New IoT deployments (e.g., smart city scenarios) can have thousands of devices per square kilometer leading to huge amount of power consumption to provide connectivity. In this paper, we investigate green LoRa wireless networks powered by a hybrid of the grid and renewable energy sources, which can benefit from harvested energy while dealing with the intermittent supply. This paper proposes resource management schemes of the limited number of channels and spreading factors (SFs) with the objective of improving the LoRa gateway energy efficiency. First, the problem of grid power consumption minimization while satisfying the system's quality of service demands is formulated. Specifically, both scenarios the uncorrelated and time-correlated channels are investigated. The optimal resource management problem is solved by decoupling the formulated problem into two sub-problems: channel and SF assignment problem and energy management problem. Since the optimal solution is obtained with high complexity, online resource management heuristic algorithms that minimize the grid energy consumption are proposed. Finally, taking into account the channel and energy correlation, adaptable resource management schemes based on Reinforcement Learning (RL), are developed. Simulations results show that the proposed resource management schemes offer efficient use of renewable energy in LoRa wireless networks.**

***Index Terms*—LoRa, energy harvesting, resource management, reinforcement learning.**

I. Introduction

LoRa wireless networks are considered as a key technology for next generation of internet of things (IoT) wireless networks [1]. These systems are based on the deployment of a large number of low-powered connected devices. Indeed, innovative wireless network, such as LoRa, enables the exponential growth connected devices, robust operations, wider coverage, and higher energy efficiency [1]. Hence, LoRa may provide sustainable connectivity to low-powered devices distributed over very large geographical areas [2], [3]. LoRa which operates in the unlicensed bands [4] provides also adaptive transmission rates and coverage for low-powered devices. LoRa enables long range transfer of information with a low transfer rate. [6]. The chirp spreading modulation (CSM) was adopted as the modulation technique for LoRa transmission [7]. This scheme is based on coding the information in the frequency shift at the beginning of the symbol. The chirp is assumed to be as a kind of carrier and the modulated signal is a chirp waveform which its behaviour depends on the SF. LoRa signals with different SFs are quasi-orthogonal [7]. However, LoRa signals with the same SF exhibit cross-correlation properties that could make them vulnerable to interference. The performance of CSM was theoretically investigated in [7]. The performance analysis of this modulation scheme was extended by considering various fading channels in [8] and by considering interference in [9]. The scalability of LoRa networks was investigated in [10] by proposing a stochastic geometry framework. This framework supports the exponential growth of connected devices. Furthermore, adequate and intelligent resource management strategies may be adopted in LoRa networks to enhance the system performance.

RL approaches have become increasingly popular, particularly for systems with complex and dynamic problem spaces [12]–[14]. These approaches are apt to act under unforeseen environments by making decisions, receiving rewards and penalties, and learning policies based on the system conditions. Unlike supervised learning, RL pursues the optimal solution by interacting with the environment parameters (e.g., the total required energy, and the current energy price). In particular, the RL approach adopts a trial-and-error search method to discover the network environment and learn the resource management policy without labeling the data at each time step. Learning the statistical distributions of the environment parameters produces the most effective action policy that adapts to changes over time and leads to the maximum reward. Hence, RL is powerful tool that can be applied in the resource management problem in LoRa wireless networks.

For greening and improving energy efficiency of LoRa networks, the devices may be powered by renewable energy sources [15], [16]. Energy harvesting allows wireless systems to continually acquire energy from nature or man-made phenomena (solar, wind, electromagnetic, ...). It provides wireless devices self-sustainability and virtually perpetual operation. Indeed, it allows reducing the use of conventional energy and accompanying carbon footprint. Hence, we propose to investigate in this work a resource management problem in hybrid

The authors are with the Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University, Qatar Foundation, Doha, Qatar (email: hrami@hbku.edu.qa; ebaccourepbesaid@hbku.edu.qa; aerbad@ieee.org; mqaraqe@hbku.edu.qa; mhamdi@hbku.edu.qa).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

energy LoRa wireless networks. We propose various resource management schemes considering both scenarios uncorrelated and time-correlated channels. An optimal resource management solution with high complexity is proposed as benchmark. Low complexity heuristic resource management schemes are proposed for the case of real-time application. Moreover, smart and adaptable resource management approaches based on RL were developed. Hence, the key contributions are summarized as follows:

- We formulated the problem of grid energy cost minimization in LoRa wireless networks while the LoRa gateway (LG) is powered by both an energy harvesting source and the grid.
- We solved the optimal offline resource management problem by decoupling the formulated problem into two subproblems. The first one is a channel and SF assignment problem and the second one is an energy management problem.
- We developed an optimal SF assignment scheme that minimizes the grid energy cost.
- We investigated the online resource management problem by proposing efficient heuristic channel, SF assignment, and energy management algorithms for both scenarios uncorrelated and time-correlated channels.
- We proposed an efficient heuristic channel, SF assignment, and energy management algorithm for hybrid energy powered LoRa networks considering *time-correlated* channels.
- We developed our resource management algorithm using deep reinforcement learning to reduce the complexity of the NP-hard optimization and implement an adaptive online energy assignment.
- We performed extensive evaluation of the proposed resource management schemes under different scenarios to illustrate the system performance in terms of grid energy cost.

The remainder of the paper is organized as follows: The system model is presented in Section II. The related works are discussed in Section III. The resource management problem in hybrid energy LoRa networks is formulated in Section IV. The optimal offline resource management problem is investigated in Section V. The online resource management algorithms are developed in Section VI. The evaluation results are presented and discussed in Section VII. Finally, conclusions are provided in Section VIII.

II. Related Work

Various works tackled the resource management problem under different LoRa network architectures and assumptions. Specifically, SF assignment, sub-band selection, user scheduling, and power allocation in LoRa networks were the focus of [17]–[42]. An efficient SF assignment scheme was developed in [17] based on instantaneous channel realizations to enhance the symbol error rate. Also, the authors of [18] proposed a novel SF allocation strategy based on matching theory to optimize the LoRa network throughput. Moreover, the number of connected devices was optimized in [19] by proposing an

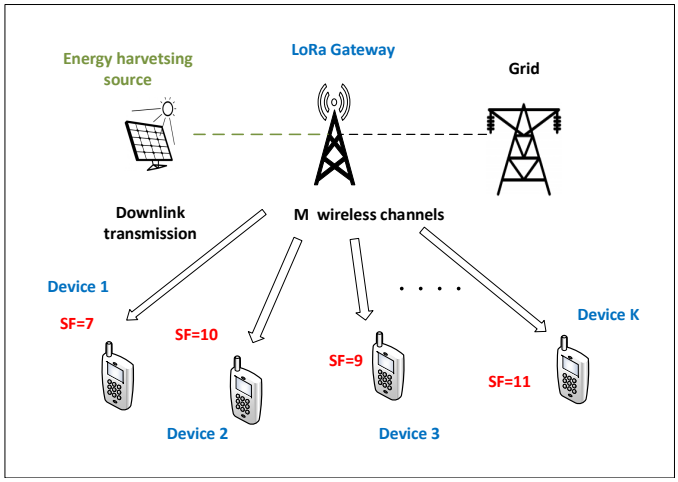


Fig. 1: Hybrid energy powered LoRa wireless networks.

efficient SF assignment scheme. A power allocation and SF assignment solution was proposed in [20] to reduce energy consumption in LoRa networks with imperfect SF orthogonality. An energy-efficient user scheduling, SF assignment, and power allocation scheme was proposed in [21]. In [22], [23], the authors proposed efficient SF assignment schemes to maximize the packet success probability. The authors of [24] proposed an efficient interference-aware SF assignment. An adequate SF allocation strategy was proposed in [25] to reduce the convergence period in LoRa networks with an adaptive data rate mechanism. In [26], a multi hop LoRa system was investigated in order to enable energy-efficient connectivity in smart city applications and the system performance was evaluated based on a experimental case study. This system was investigated further in [27] by proposing an efficient clustering algorithm. A capacity maximization problem in LoRa networks was studied in [28]. User scheduling was incorporated to a multi channel LoRa network in [29] to improve the synchronization packet length. The average number of decoded LoRa frames was investigated in [30] by taking into account physical layer and medium access control. In [32], the devices are powered by energy harvesting sources for uplink transmission considering only one channel. The optimal energy management and SF assignment algorithm was devised. In [31], the authors proposed a low complexity energy-efficient rate control scheme for LoRa uplink transmission based on Markov chain. In [33], the authors investigated the performance of LoRa networks in term of latency by proposing a sub-band selection scheme. The authors of [34] provided a theoretical analysis of the achievable LoRa throughput in uplink considering imperfect SF orthogonality. In [35], the trade-off relation between the waiting time and the energy consumption in LoRa networks was optimized by deriving the optimal number of ping slots. In [36], the authors proposed an appropriate radio configuration scheme based on integer linear programming, which takes into account the scalability of the network in order to enhance the reliability of the LDs. A dynamic LoRa transmission control system was proposed in [37] to improve the energy efficiency. In [38], the LoRa

capacity was enhanced by proposing an interleaved chirp spreading scheme. Moreover, the LoRa goodput was improved in [39] by controlling the receiver window size. In [40], the coverage in hybrid LoRa networks was enhanced by developing an optimal planning scheme. The energy efficiency of LoRa networks was improved in [41] by proposing a tree network adopted for LoRa to mitigate the energy consumption constraints. In [42], the authors proposed a novel fair and scalable relay control scheme for LoRa wireless networks to improve the success probability.

Different from the existing works, we proposed in this paper to investigate a novel system model for LoRa wireless networks powered by both an energy harvesting source and the grid, which can benefit from harvested energy while dealing with the intermittent supply. The design of energy-efficient LoRa networks is challenging due to the intermittency of renewable energy sources. Since most of the existing works are dealing with the coverage and the throughput, we formulated in this paper a novel challenging optimization problem with the objective of improving the LoRa gateway energy efficiency. Furthermore, we developed smart and adaptive resource management schemes for green wireless networks based on deep reinforcement learning.

III. System Model

A. Channel and Signal Model

In this work, we consider a typical downlink LoRa wireless network (as shown in Fig. 1) that includes a LG serving K arbitrarily distributed LoRa devices (LDs) through M channels. Let B_m denote the bandwidth of channel m. A given time interval is partitioned into L frames with duration T_{out}. The channel coefficients between the gateway and LD k through channel m at frame i is given by $g_{k,m}(i) = \beta_k(i)h_{k,m}(i)$, where $\beta_k(i)$ represents the path loss and $h_{k,m}(i)$ represents the small-scale fading channel coefficient. The important notations are summarized in Table I.

The LoRa modulation known as CSM is introduced in [7], where the modulated signal is a chirp waveform and the frequency increases linearly with the time index. The LG sends a symbol $s_k(i)$ to LD k at frame i with duration $2^{\alpha_k(i)}T$, where $\alpha_k(i)$ is the SF taking values in $\Gamma = \{7, 8, 9, 10, 11, 12\}$ and $T = \frac{T_{out}}{2^{12}}$ is the duration of a sample transmission [6]. The LG sends $\alpha_k(i)$ bits to device k at frame i. The symbol $s_k(i)$ takes values in $\left\{0, 1, 2, \ldots, 2^{\alpha_k(i)} - 1\right\}$ [6]. The LDs adopt different SF for transmission in order to ensure orthogonality and enable multi user transmission [7]. Hence, the transmitted waveform vector for device k at frame i is given by:

$$\mathbf{x}_k(i) = \begin{bmatrix} \left[\frac{1}{\sqrt{2^{\alpha_k(i)}}} e^{\jmath 2\pi \left[(s_k(i)+f) \bmod 2^{\alpha_k(i)} \right] \frac{f}{2^{\alpha_k(i)}}} \right]_{f=0..2^{\alpha_k(i)}-1} \\ \mathbf{0}_{2^{12}-2^{\alpha_k(i)}} \end{bmatrix} \quad (1)$$

A zero padding with length $2^{12} - 2^{\alpha_k(i)}$ is added for each vector in order to ensure the same vector length for all devices. The possible waveforms of CSM modulation are shown to be orthogonal [8]. Hence, the inner product receiver may be applied [7]. It consists of projecting the received vector $\mathbf{y}(i)$ onto the different signals given by:

$$\mathbf{c}_{|s_k(i)} = \left[g^*_{k,m}(i) \frac{1}{\sqrt{2^{\alpha_k(i)}}} e^{\jmath 2\pi \left[(s_k(i)+f) \bmod 2^{\alpha_k(i)} \right] \frac{f}{2^{\alpha_k(i)}}} \right]^T_{f=0..2^{\alpha_k(i)}-1} \quad (2)$$

and choosing the one with maximal square modulus projection. Hence, the best estimate of the transmitted signal $\hat{s}_k(i)$ by device k at frame i is given by:

$$\hat{s}_k(i) = \underset{0..2^{\alpha_k(i)}-1}{\operatorname{argmax}} |\langle \mathbf{y}(i), \mathbf{c}_{|s_k(i)} \rangle|^2. \quad (3)$$

It is worth mentioning that only 6 LDs can be served simultaneously in one channel, since there are six available SFs range from 7 to 12 [6]. Also, each LD can access at most one channel. Let $\chi_{k,m}(i)$ be a Boolean parameter that is set to 1 if LD k at frame i is assigned to channel m and to 0 otherwise. Let $\psi_m(i)$ denote the set of devices assigned to channel m at frame i. The vector of received signals through channel m at frame i is expressed as:

$$\mathbf{y}_m(i) = \sum_{k=1}^{K} \chi_{k,m}(i) p_k(i) g_{k,m}(i) \mathbf{x}_k(i) + \mathbf{w}_m(i), \quad (4)$$

where $p_k(i)$ is the power allocated for LD k at frame i and $\mathbf{w}_m(i)$ is assumed to be additive white Gaussian noise (AWGN) with zero mean and variance σ_m^2. Hence, the downlink signal-to-noise ratio (SNR) of LD k through channel m at frame i is expressed as:

$$\gamma_{k,m}(i) = \frac{\chi_{k,m}(i) p_k(i) \mid \mathbf{g}_{k,m}(i) \mid^2}{\sigma_m^2}. \quad (5)$$

The rate for LD k through channel m at frame i is given by:

$$R_{k,m}(i) = B_m \log_2 \left(1 + \frac{\chi_{k,m}(i) p_k(i) \mid \mathbf{g}_{k,m}(i) \mid^2}{\sigma_m^2} \right). \quad (6)$$

An LD k is scheduled at frame i, if it is assigned to one of the channels $\sum_{m=1}^{M} \chi_{k,m}(i) = 1$, otherwise it is not scheduled and $\sum_{m=1}^{M} \chi_{k,m}(i) = 0$.

B. Energy Model

The LG is powered by both an energy harvesting source and the grid. The grid energy source compensates for the randomness and intermittency of the harvested energy. The harvested energy $E(i)$ is first stored in a battery with maximal capacity B_{max}. It is modeled as a correlated time process following a discrete-time Markov model as in [44], [45], $E(i) \in \Omega \triangleq \{\omega_1, \omega_2, ..., \omega_M\}$ where Ω is the set of possible amount of harvested energy and $Q(\omega_m, \omega_j) = Pr(E(i+1) = \omega_m \mid E(i) = \omega_j)$ is the state transition probability. Let $B(i)$ denote the battery level at frame i. The required energy consumed at frame i is given by:

$$\begin{aligned} X(i) &= X^h(i) + X^g(i) \\ &= E_c + \sum_{k=1}^{K} p_k(i) 2^{\alpha_k(i)} T, \end{aligned} \quad (7)$$

where E_c is a fixed energy consumed by the circuit which includes the amounts of power consumed by digital to analog

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

TABLE I: Summary of Important Notations.

Symbol	Description	Symbol	Description
K	Number of LDs	$p_k(i)$	Power allocated for LD k
B_m	Bandwidth of channel m	σ_m^2	Noise variance
L	Number of frames	$\gamma_{k,m}(i)$	SNR of LD k through channel m at frame i
T_{out}	Duration of a frame	B_{max}	Capacity of the battery
$g_{k,m}(i)$	Channel coefficient of LD k through channel m	$E(i)$	Amount of harvested energy at frame i
$\beta_k(i)$	Path loss	$B(i)$	Battery level at frame i
$h_{k,m}(i)$	Small-scale fading channel coefficient	E_c	Fixed energy consumed by the circuit
$\alpha_k(i)$	Spreading factor	W_i	Grid's weight
Γ	Set of SFs	$X^g(i)$	Energy drawn from the power grid
T	Duration of a sample transmission	γ_{th}	Minimum received SNR
$\chi_{k,m}(i)$	Channel assignment index	$\psi_m(i)$	Set of devices assigned to channel m
S	Set of states	A	Set of possible actions
R	Immediate reward	P	State transition probability
γ	Discount factor	θ	Neural network weights
Π	RL policy	D	Experience memory
λ	Adjusting factor of PPO	ϵ	Cliprange of PPO
ρ	Soft update of DDPG	N^s	Noise process of DDPG
α	Learning rate of the DNN		

converters (DACs), mixers and filters, and $X^h(i)$ and $X^g(i)$ denote the energy drawn from the energy harvesting source and the power grid respectively. The consumed energy from the energy harvesting source cannot exceed the battery level. Hence, the energy causality constraint is given by:

$$X^h(i) \leq B(i). \tag{8}$$

The battery level update is expressed as:

$$B(i+1) = \min\left(B_{\max}, B(i) - X^h(i) + E(i)\right). \tag{9}$$

We consider that the consumed grid energy at frame i is weighted by a factor W_i [46]. Hence, the grid energy cost which is considered in this paper as an objective function, is expressed as:

$$\Delta = \sum_{i=1}^{L} W_i X^g(i). \tag{10}$$

In Fig. 2, we illustrate the proposed hybrid energy LoRa framework.

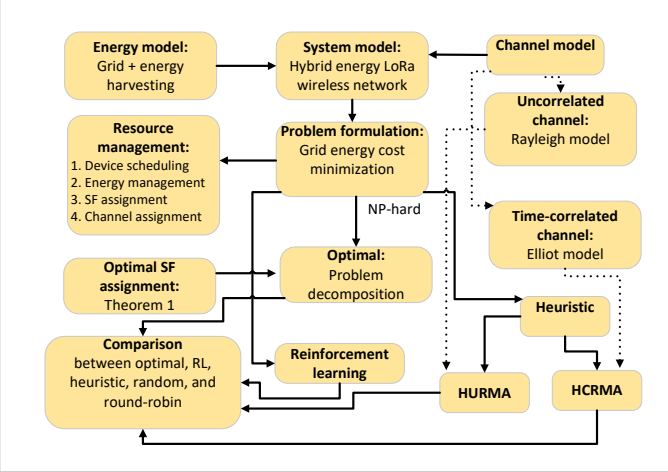


Fig. 2: Illustration of research design and different steps of the work.

IV. Problem Formulation

The objective is to minimize the grid energy cost while using the available harvested energy. Each device requires a minimum received SNR to be satisfied. The energy management should take advantage of the grid's power weight variations by consuming more grid power when the associated weight is low while storing the renewable energy for future use, and by consuming less grid power when its weight is high. Moreover, the channels and SFs may be optimally assigned to the devices. Only 6 LDs per channel can be served simultaneously at each frame. The aim will be to schedule the devices with the minimal grid energy cost while making use of the available harvested energy and ensuring a minimum SNR to each scheduled LD. The main problem can be formulated as:

$$\min_{\substack{\{\chi_{k,m}(i),\alpha_k(i),p_k(i)\}\\ k=1,\dots,K,m=1,\dots,M,i=1,\dots,L}} \sum_{i=1}^{L} W_i X^g(i)$$

subject to

$$\begin{aligned}
&(11.a): \gamma_{k,m}(i) \geq \chi_{k,m}(i)\gamma_{th},\ \forall k=1,\dots,K, i=1,\dots,L,\\
&(11.b): \sum_{i=1}^{l} X^h(i) \leq \sum_{i=1}^{l} E(i),\ \forall l=1,\dots,L\\
&(11.c): \sum_{i=1}^{l} E(i) - \sum_{i=1}^{l-1} X^h(i) \leq B_{max},\ \forall l=2,\dots,L,\\
&(11.d): X^g(i) + X^h(i) = E_c + \sum_{k=1}^{K} p_k(i) 2^{\alpha_k(i)} T,\\
&\qquad \forall i =,\dots,L,\\
&(11.e): \sum_{m=1}^{M} \chi_{k,m}(i) \leq 1,\ \forall k=1,\dots,K, i=1\dots L,\\
&(11.f): \sum_{k=1}^{K} \chi_{k,m}(i) = 6,\ \forall m=1,\dots,M, i=1\dots L,\\
&(11.g): \alpha_k(i) \neq \alpha_p(i),\ \forall k,p \in \psi_m(i), k \neq p,\\
&\qquad m=1,\dots,M,, i=1,\dots,L,\\
&(11.h): p_k(i) \geq 0,\ \forall k=1,\dots,K, i=1,\dots,L,\\
&(11.i): \alpha_k(i) \in \Gamma,\ \forall k=1,\dots,K, i=1,\dots,L\\
&(11.j): \chi_{k,m}(i) \in \{0,1\},\ \forall k=1,\dots,K, m=1,\dots,M,\\
&\qquad i=1,\dots,L.
\end{aligned} \tag{11}$$

Constraint $(11.a)$ ensures a minimum received SNR, denoted γ_{th}, to each LD. Constraint $(11.b)$ is related to the energy causality, i.e. the consumed harvested energy cannot exceed the available energy at the battery. Additionally, constraint $(11.c)$ implies that the harvested energy at the current frame cannot exceed the maximum battery capacity. Constraint $(11.d)$ specifies that the required consumed energy is drawn from the grid and the energy harvesting source. Constraint $(11.e)$ imposes that only one channel at most can be assigned to each LD. Constraint $(11.f)$ specifies the maximal number of LDs within same channel, which cannot exceed the number of SFs. Constraint $(11.g)$ imposes that the LDs within same channel should be assigned different SFs. Constraint $(11.h)$ ensures the non-negativity of the allocated amounts of power. Constraint $(11.i)$ specifies the set of available SFs. Finally, constraint $(11.j)$ specifies the channel assignment index.

V. Optimal Offline Resource Management

In this section, the optimal channel and SF assignment, as well as the energy management are investigated. The main formulated problem (11) is a mixed integer non-linear program because of its combinatorial nature and the non-linearity of the constraints. Meanwhile, to solve (11), the problem may be decoupled into two sub-problems. Since the goal is to minimize the total consumed energy at each frame, the channels and SFs may be optimally assigned among the LDs at each frame. Hence, the optimal total required consumed energy for network operation at each frame could be determined. Next, the optimal energy drawn from the energy harvesting source over time may be optimally derived based on the grid's wight. First, the required transmit power to meet the SNR constraint of LD k using channel m at frame i following (5) is given by:

$$p_k(i) = \chi_{k,m}(i) \frac{\gamma_{th}\sigma_m^2}{\mid \mathbf{g}_{k,m}(i) \mid^2}. \tag{12}$$

Hence, the required consumed energy at frame i is given by:

$$X(i) = E_c + \sum_{k=1}^{K} \left(\sum_{m=1}^{M} \frac{\gamma_{th}\sigma_m^2}{\mid \mathbf{g}_{k,m}(i) \mid^2} \chi_{k,m}(i) \right) 2^{\alpha_k(i)} T. \tag{13}$$

Hence, replacing $p_k(i)$ and $X(i)$ by their expression in (12) and (13), the channel and SF assignment problem at frame i can be formulated as:

$$\min_{\substack{\{\chi_{k,m}(i),\alpha_k(i)\}\\ k=1,\dots,K,m=1,\dots,M}} \sum_{k=1}^{K} \left(\sum_{m=1}^{M} \frac{\gamma_{th}\sigma_m^2}{\mid \mathbf{g}_{k,m}(i) \mid^2} \chi_{k,m}(i) \right) 2^{\alpha_k(i)}$$

subject to

$$\begin{aligned}
&(14.a): \sum_{m=1}^{M} \chi_{k,m}(i) \leq 1,\ \forall k=1,\dots,K,\\
&(14.b): \sum_{k=1}^{K} \chi_{k,m}(i) = 6,\ \forall m=1,\dots,M,\\
&(14.c): \alpha_k(i) \neq \alpha_p(i),\ \forall k,p \in \psi_m(i), k \neq p, m=1,\dots,M,\\
&(14.d): \alpha_k(i) \in \Gamma,\ \forall k=1,\dots,K,\\
&(14.e): \chi_{k,m}(i) \in \{0,1\},\ \forall k=1,\dots,K, m=1,\dots,M.
\end{aligned} \tag{14}$$

The problem (14) is combinatorial and non-linear; and thus is a non-linear integer problem. Consequently, the problem is NP-hard [47] and could be solved by brute-force search with exponential complexity growth.

The required consumed energy for the network operation could be determined at each frame after deriving the optimal channel and SF assignment. Since the goal is to minimize the grid energy cost, the energy drawn from the energy harvesting source may be optimally managed over time. Hence, the energy management problem can be formulated as:

$$\min_{\substack{\{X^h(i)\}\\ i=1,\dots,L}} -\sum_{i=1}^{L} W_i X^h(i)$$

subject to

$$\begin{aligned}
&(15.a): \sum_{i=1}^{l} X^h(i) \leq \sum_{i=1}^{l} E(i),\ \forall l=1,\dots,L,\\
&(15.b): \sum_{i=1}^{l} E(i) - \sum_{i=1}^{l-1} X^h(i) \leq B_{max},\ \forall l=2,\dots,L,\\
&(15.c): X^h(i) \leq X(i),\ \forall i=1,\dots,L,\\
&(15.d): X^h(i) \geq 0,\ \forall i=1,\dots,L.
\end{aligned} \tag{15}$$

The objective function and the constraints of problem (15) are clearly linear. Hence, the optimal energy management is obtained by solving a linear program using interior-point method implemented in numerical tools such as CVX [48].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

VI. Online Resource Management

In this section, online resource management is investigated for both scenarios uncorrelated and time-correlated channels by proposing low complexity heuristic algorithms. The LG is assumed to know the channel coefficients, the harvested energy and the grid's weight only at the current frame i.

A. Heuristic Approach

1) Optimal SF Assignment: The optimal SF assignment can be derived using the following theorem:

Theorem 1. Let consider K LDs, where their coefficients v_k verify $v_1 < v_2 < \ldots < v_K$. The optimal SF assignment that minimizes the objective function $\sum_{k=1}^{K} v_k 2^{\alpha_k}$ is given by assigning the lowest SF $\alpha^{\min}$ to the LD with biggest coefficient v_k until assigning the biggest SF $\alpha^{\max}$ to the LD with lowest coefficient v_k.

Proof. Let consider two LDs with $v_1 < v_2$. We have

$$\begin{aligned} v_1 2^7 + v_2 2^8 - (v_1 2^8 + v_2 2^7) &= -v_1 2^7 + v_2 2^7 \\ &= 2^7 (v_2 - v_1) > 0. \end{aligned} \tag{16}$$

Hence, the optimal SF assignment is given by assigning 8 to LD 1 and 7 to LD 2. This relation can be extended recursively to the case of K LDs. □

2) Uncorrelated Channel: Let consider a quasi-static Gaussian independent and identically distributed (i.i.d.) slow fading channel. A heuristic low complexity algorithm is proposed to solve channel and SF assignment, and energy management problem in LoRa network powered by energy harvesting. Only NM LDs can be scheduled at each frame. The proposed algorithm starts by scheduling the LDs one by one based on their channel coefficient. The best channel is assigned to the LD with the highest channel coefficient modulus in order to save the required consumed energy. This procedure is repeated until all the channels become full.

Next, the SF assignment are performed following **Theorem 1** by assigning the lowest SF $\alpha^{\min}$ to the user with biggest coefficient p_k until assigning the biggest SF $\alpha^{\max}$ to the LD with lowest coefficient p_k. Finally, after assigning the channels and SFs, the required consumed energy could be computed. The proposed algorithm uses the maximum available harvested energy at the battery at each frame. The proposed Heuristic Uncorrelated Resource Management Algorithm (HURMA) is described in **Algorithm 1**.

The computational complexity of HURMA is derived as follows. For the SF assignment, an array with N elements is sorted M times in the **for** loop with a complexity order $O(MN\log(N))$. The channel assignment in the **while** loop has a complexity order equal to $O(MN\log(MN))$ because it is similar to sort an array with MN elements. Hence, the computational complexity of HURMA is given by:

$$\begin{aligned} C^{\text{HURMA}} &= O(MN\log(N) + MN\log(MN)) \\ &= O(MN\log(MN)). \end{aligned} \tag{17}$$

Hence, the proposed low complexity algorithm can be executed in polynomial time.

Algorithm 1 Heuristic Uncorrelated Resource Management Algorithm (HURMA)

$B(1) \leftarrow E(1)$, *// battery initialization*
for $i = 1 : L$ **do**
  $\chi_{k,m}(i) \leftarrow 0$, *// initialization*
  $\boldsymbol{c} \leftarrow \mathbf{0}_{K\times 1}$, *// initialize the vector that contains the channel index for each LD*
  $\boldsymbol{c}_f \leftarrow \mathbf{0}_{M\times 1}$, *// initialize the vector that indicates the number of scheduled LD for each channel*
  $\boldsymbol{V}$ is a matrix that contains the modulus of the channels coefficients for all LDs
  while $\sum_{m=1}^{M} \boldsymbol{c}_f[m] \neq MN$ **do**
    $(k_{\max}, m_{\max}) \leftarrow \operatorname{argmax} \boldsymbol{V}$, *// select the LD with higher channel coefficient*
    if $\boldsymbol{c}_f[m_{\max}] < N$ **then**
      $\boldsymbol{c}[k_{\max}] \leftarrow m_{\max}$, *// assign channel $m_{\max}$ to LD $k_{\max}$*
      $\boldsymbol{c}_f[m_{\max}] \leftarrow \boldsymbol{c}_f[m_{\max}] + 1$
      $\chi_{k_{\max},m_{\max}}(i) \leftarrow 1$, *// LD selection*
      $\boldsymbol{V}[k_{\max}, :] \leftarrow \mathbf{0}_{1\times M}$
    else
      $\boldsymbol{V}[:, m_{\max}] \leftarrow \mathbf{0}_{K\times 1}$
    end if
  end while
  for $m = 1 : M$ **do**
    $\boldsymbol{d}_m \leftarrow$ indices of LDs assigned to m
    $v_n \leftarrow \frac{\sigma_m^2}{|\mathbf{g}_{\boldsymbol{d}_m(n),m}(i)|^2}, n = 1, \ldots, N$
    $\boldsymbol{s}_m \leftarrow \boldsymbol{d}_m$ sorted in ascending order based on p_n
    $\alpha_{\boldsymbol{s}_m(n)}(i) \leftarrow 13 - n, n = 1, \ldots, N$, *// SF assignment*
  end for
  $X(i) \leftarrow E_c + \sum_{k=1}^{K} \left(\sum_{m=1}^{M} \frac{\gamma_{th}\sigma_m^2}{|\mathbf{g}_{k,m}(i)|^2} \chi_{k,m}(i) \right) 2^{\alpha_k(i)} T$, *// compute the required consumed energy*
  if $X(i) \leq B(i)$ **then**
    $X^h(i) \leftarrow X(i)$
    $X^g(i) \leftarrow 0$
  else
    $X^h(i) \leftarrow B(i)$
    $X^g(i) \leftarrow X(i) - X^h(i)$
  end if
  $B(i) \leftarrow \min\left(B_{\max}, B(i) - X^h(i) + E(i)\right)$, *// battery update*
end for

3) Time-correlated Channel: Let us consider a time-correlated channel which is modeled by Gilbert Elliot channel model [43]. The state of the channel at frame i is modeled as a one-dimensional Markov chain with two states: a good state denoted by $\boldsymbol{G}$, and a bad state denoted by $\boldsymbol{B}$. Channel transitions occur at the beginning of each frame. The transition probabilities are given by:

$$\mathbb{P}\left[h_{k,m}(i) = \boldsymbol{G} \mid h_{k,m}(i-1) = \boldsymbol{G}\right] = \lambda_1, \tag{18}$$

and

$$\mathbb{P}\left[h_{k,m}(i) = \boldsymbol{G} \mid h_{k,m}(i-1) = \boldsymbol{B}\right] = \lambda_0. \tag{19}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

A second heuristic resource management algorithm is proposed taking into account the channel correlation. The channel assignment is performed as follows. First, the NM LDs with highest path loss coefficient are selected. Then, the proposed algorithm starts by assigning the channels to the LDs one by one based on their path loss coefficient starting from the most distant LD. The corresponding LD chooses the best available channel, i.e., the one with highest channel coefficient modulus. We start by the most distant LD by assigning it the best channel in order to compensate their SNR and save the required consumed energy. This procedure is repeated until all the channels become full. Next, the SF assignment and energy management are performed similar to the algorithm HURMA. The proposed Heuristic Correlated Resource Management Algorithm (HCRMA) is described in **Algorithm 2**.

The computational complexity of HCRMA is derived as follows. The SF assignment for HCRMA is similar to HURMA and is done with a complexity order $O(MN\log(N))$. The channel assignment in the **for** loop has a complexity in the order $O(M^2N)$. The sort of the array $\boldsymbol{w}$ is done with a complexity $O(K\log(K))$. Hence, the computational complexity of HCRMA is given by:

$$C^{\text{HCRMA}} = O(MN\log(N) + M^2N + K\log(K)). \quad (20)$$

The proposed low complexity algorithm can be executed in polynomial time.

Algorithm 2 Heuristic Correlated Resource Management Algorithm (HCRMA)

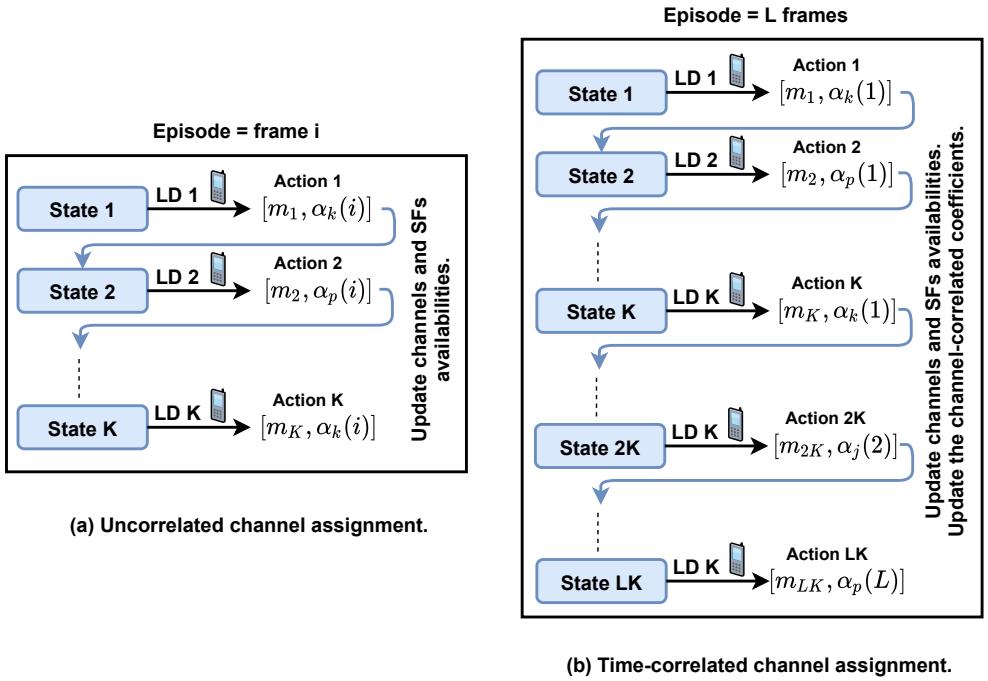
$B(1) \leftarrow E(1)$, *// battery initialization*
for $i = 1 : L$ **do**
  $\chi_{k,m}(i) \leftarrow 0$, *// initialization*
  $\boldsymbol{c} \leftarrow \mathbf{0}_{K\times 1}$, *// initialize the vector that contains the channel index for each LD*
  $\boldsymbol{c}_f \leftarrow \mathbf{0}_{M\times 1}$, *// initialize the vector that indicates the number of scheduled LD for each channel*
  $\boldsymbol{w}$ contains the ascending sort of the indices of the LDs based on path loss
  for $k = K - MN + 1 : K$ **do**
    $k_{\max} \leftarrow \boldsymbol{w}[k]$
    $m_{\max} \leftarrow$ the index of the channel with highest coefficient for LD $k_{\max}$
    $\boldsymbol{c}[k_{\max}] \leftarrow m_{\max}$, *// assign channel $m_{\max}$ to LD $k_{\max}$*
    $\chi_{k_{\max},m_{\max}}(i) \leftarrow 1$, *// LD selection*
    $\boldsymbol{c}_f[m_{\max}] \leftarrow \boldsymbol{c}_f[m_{\max}] + 1$
    if $\boldsymbol{c}_f[m_{\max}] = N$ **then**
      remove channel $m_{\max}$ from the set of available channels
    end if
  end for
  for $m = 1 : M$ **do**
    $\boldsymbol{d}_m \leftarrow$ indices of LDs assigned to m
    $v_n \leftarrow \frac{\sigma_m^2}{|\mathbf{g}_{\boldsymbol{d}_m(n),m}(i)|^2}, n = 1,\ldots,N$
    $\boldsymbol{s}_m \leftarrow \boldsymbol{d}_m$ sorted in ascending order based on p_n
    $\alpha_{\boldsymbol{s}_m(n)}(i) \leftarrow 13-n, n = 1,\ldots,N$, *// SF assignment*
  end for
  $X(i) \leftarrow E_c + \sum_{k=1}^{K}\left(\sum_{m=1}^{M}\frac{\gamma_{th}\sigma_m^2}{|\mathbf{g}_{k,m}(i)|^2}\chi_{k,m}(i)\right)2^{\alpha_k(i)}T$, *// compute the required consumed energy*
  if $X(i) \le B(i)$ **then**
    $X^h(i) \leftarrow X(i)$
    $X^g(i) \leftarrow 0$
  else
    $X^h(i) \leftarrow B(i)$
    $X^g(i) \leftarrow X(i) - X^h(i)$
  end if
  $B(i) \leftarrow \min\left(B_{\max}, B(i) - X^h(i) + E(i)\right)$, *// battery update*
end for

B. Reinforcement Learning Approach

The problem in (11) is NP-hard, which makes deriving the optimal solution extremely complex and time consuming. Moreover, in the optimization, we suppose that we have a full overview about the future harvested energy, which is not realistic. On the other hand, the online heuristic presents a low-complexity solution. However, allocation decisions are taken greedily, without any insight about the harvested energy in the next frames. Recently, reinforcement learning techniques have become highly adopted for applications characterised by dynamic and complex problem spaces. More specifically, RL is considered as one of the important paradigms of machine learning, in addition to supervised and unsupervised learning [11], [13]. The advantage of this technique is that it approaches the optimal solution by interacting with the environment parameters (e.g., the harvested energy and the energy prices), learning the statistical distributions of these features, and determining the most efficient policy that takes the actions based on the current status of the system and the learned insights about the future. Therefore, the reinforcement learning can be the most adequate solution to reduce the complexity of the NP-hard problem, while taking sub-optimal decisions owing to the knowledge learned about the system.

On these bases, we propose to investigate an online resource management solution based on RL. In the previous section, we decomposed the optimal solution (11) into two sub-problems, namely channel assignment and energy management problems. Accordingly, we design two RL systems, responsible to solve the above-mentioned sub-problems. In particular, in each time step of the resource management process, the RL agents should make a decision on the channels to allocate, the assigned SFs, and the optimal amount of energy to draw from the energy harvesting source. The agents ultimately aim to minimize the grid energy cost, while respecting the battery capacity and the constraints on the available resources. During the learning process, the energy harvesting system receives rewards and experiences penalties for each allocation decision it makes until it reaches the convergence to the optimal policy. The defined sub-problems can be abstracted as two Markov

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal



(a) Uncorrelated channel assignment.

(b) Time-correlated channel assignment.

Fig. 3: Episode design of uncorrelated and time-correlated channel assignments.

Decision Process (MDP) frameworks; each one is presented by the five-tuple (S, A, P, R, γ) [11]. S presents the set of states for each framework, A denotes the set of potential actions, P is the state transition probability, R is defined as the immediate reward gained for each action, and γ denotes the discount factor. These elements are discussed for both sub-problems, in the following sub-sections.

1) Channel assignment problem:

- **MDP environment design:** In our paper context, the MDP environment represents the LoRa wireless network with which the agent interacts. We design this environment to receive the action A_t generated by the agent at a step t, assign a reward R_t, and introduce the next state S_{t+1}. The finite sequence of such steps is called an episode. By experiencing various episodes, the agent is trained to learn from historical actions and their associated rewards. Therefore, in the first RL framework, we define each step as the channel and spreading factor allocation to one of the LoRa devices, as illustrated in the sub-problem (14). We underline that different episodes of experiences are independent, which means that at the beginning of each episode, the cumulative reward is initiated to 0. Therefore, since the channels can be time-correlated or totally uncorrelated, the episode is defined for each network configuration differently. More specifically, if channels are uncorrelated among different L frames, we define each episode as one frame i where channels and SFs are assigned to the existing LDs. This way, the episode length is equal to the number of devices K. In case the channels are correlated, the resource management of all frames should be accomplished in one episode, implying that the episode length is equal to $K \times L$. The illustration of both episode designs is presented in Fig. 3. It is worth mentioning that the agent does not have an overview of the environment design. Instead, the optimal policy $\Pi : S \rightarrow A$ is built by observing the surrounding environment, selecting actions, and gaining rewards.
- **States and actions:** The set of states S is composed of all possible environment circumstances and conditions at each step. We set $S = \{assignedSF(i), g_t(i)\}$, where $assignedSF(i)$ is an $M \times 6$ matrix of assigned SFs in the frame i. $assignedSF(i)_{m,j}$ is equal to 1, if the spreading factor j of the channel m is assigned to one of the LDs, 0 otherwise. $assignedSF(i)$ is initiated to a null matrix at each new frame. g_t is the matrix of the channels' coefficients between the gateway and the LD related to the current step t and frame i. Depending on the system state and based on the policy Π, the agent takes an action A_t. This action includes selecting the appropriate channel m and the related spreading factor α_k. Thus, the action can be expressed as $A_t = [m, \alpha_k]$. Note that $m \in \{0..M\}$ and $\alpha_k \in \Gamma$, as indicated by the constraints (14.d) and (14.e). We note that m equal to 0 means that the LD will not be assigned a channel. The constraint (14.a) is respected by design as we assign at most one channel m to each LD at each step. Once the decision is made, the LD related to the current step is served through the chosen resources. After each step, the $assignedSF$ matrix is updated according to the predicted action. Subsequently, the new matrix is fed as an input to the next step as illustrated in Fig. 3.
- **Reward function:** The reward function is formulated to match the high-level objective of the optimization problem (14), aiming at minimizing the required energy at each frame, while respecting the availability of channels and spreading factors. To ensure respecting such constraints, rewards and penalties are assigned. Specifically, when a state S_t is received, a decision A_t should be taken while meeting

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

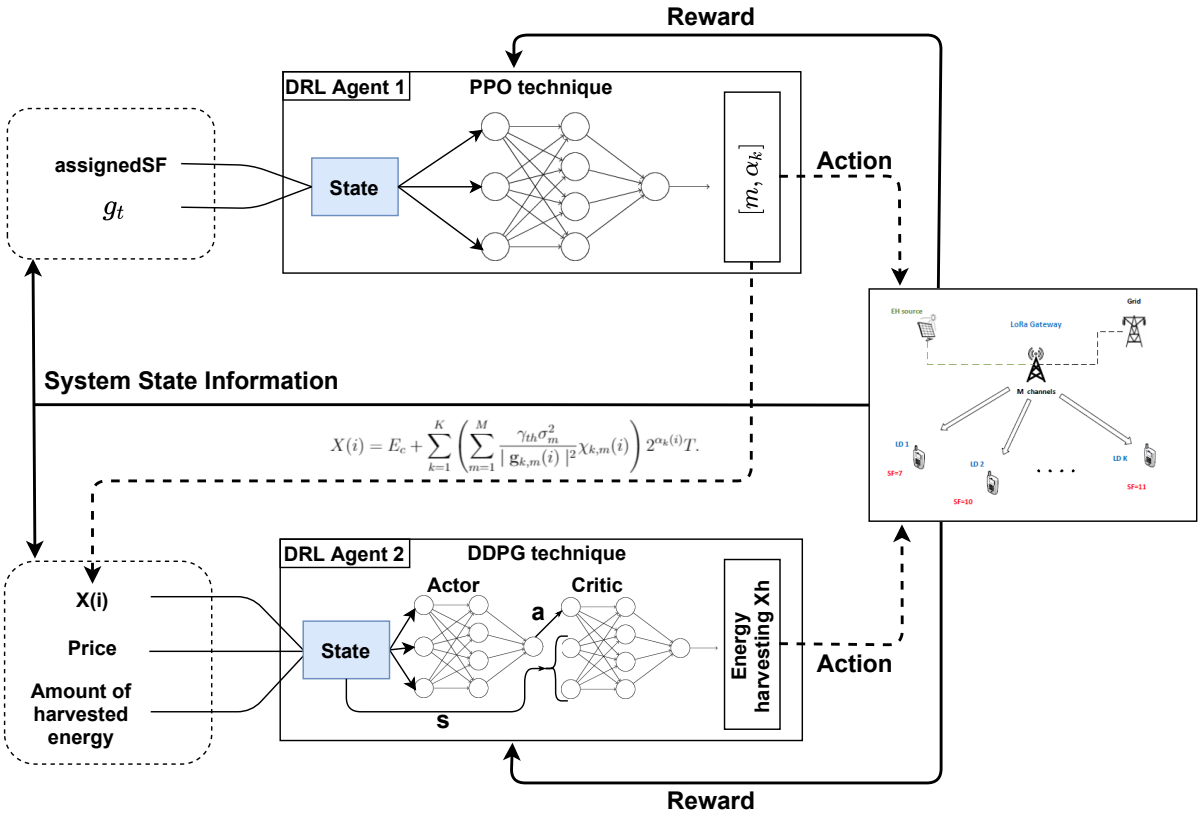


Fig. 4: Deep reinforcement learning architecture.

the following requirements:

$$\begin{cases} C_1: & \sum_{j=(i.K+1)}^{t} (A_j(1) == A_t(1)) \leq 6 & \text{constraint (14. b)} \\ C_2: & assignedSF_{A_t(1),A_t(2)}(i) = 0 & \text{constraint (14. c)} \end{cases} \tag{21}$$

C_1 indicates that the maximal number of LDs within the chosen channel cannot exceed the number of SFs, which matches the constraint (14.b). C_2 shows that the chosen spreading factor associated to the allocated channel is not assigned to another LD, which is equivalent to the constraint (14.c). We note that i denotes the current frame, which is equal to $\lfloor \frac{t-1}{K} \rfloor$. To this end, we define the immediate reward as follows:

$$R_t = C_1 * C_2 - \frac{\gamma_{th}\sigma^2_{A_t(1)}}{|\mathbf{g}_{t,A_t(1)}(i)|^2} 2^{A_t(2)}. \tag{22}$$

If one of the constraints C_1 or C_2 is not respected ($C_1 = 0$ or $C_2 = 0$), invalid situations can occur. Hence, we attribute 0 as a reward. The maximum direct reward is only received, when all requirements are met. In this way, the RL system tries to meet the scenarios that respect the constraints, in order to maximize the received bonuses and avoid null rewards. Additionally to the reward assigned for respecting the constraints of the LoRa system, the RL agent is charged for inaccurate resource allocation. Since the performance of the network is quantified by minimizing the required energy at each frame i, we use this indicator to define the penalty added to the reward function, as shown in Eq. (22). Accordingly, the RL agent selects optimal allocations by maximizing the cumulative rewards and minimizing the penalties.

- **Agent design:** The goal of the agent is to learn how to minimize the required energy throughout different episodes. To achieve this goal in the long run, the agent needs to build an optimal policy Π that maximizes the future expected reward approximated by the action-value function Q^{Π} expressed by the classical Bellman equation:

$$Q^{\Pi}(s,a) = \mathbb{E}[\sum_{k=0}^{T} \gamma^k R_{t+k} | \Pi, S_t = s, A_t = a], \tag{23}$$

where $0 \leq \gamma \leq 1$ denotes the discount parameter, that reflects the importance of the direct reward compared to long-term reward received at the end of the episode. Setting γ to be small implies that the agent is designed to be shortsighted, and only the step rewards are considered. A bigger γ indicates that the agent is farseeing and gives higher weights to future rewards. The Q-value $Q(S_t, A_t)$ serves to assess the accuracy of the decision A_t for a given state S_t. After determining the optimal policy, the agent selects its actions as follows:

$$\Pi^*(s) = argmax_a Q^{\Pi}(s,a). \tag{24}$$

Since our LoRa system is dynamic and the action space is dimensional and depends on the number of channels and spreading factors, it is challenging to save all Q-values in a Q-table and use traditional RL methods [12]. Consequently, we propose to adopt Deep Reinforcement Learning (DRL)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

using DNNs to approximate the action-value function. DRL approaches can be classified into two categories: the value-based and policy-based methods. Particularly, the value-based method adopts deep learning to estimate the value function (e.g., DQN [13]), whereas the policy-based DRL uses DNNs for approximating the parameterized policy (e.g., REINFORCE [14]). We opt for the latter approach as it achieves better performance for stochastic policies. This approach works by computing an estimator of the policy gradient:

$$\nabla L(\theta) = \mathbb{E}[\nabla_\theta log \Pi(S_t|A_t,\theta)\hat{E_t^s}], \tag{25}$$

where Π is the parametrized policy, θ represents the weight of the DNN, and $\hat{E_t^s}$ is the function estimator at the step t. $\hat{E_t^s}$ is calculated as follows:

$$\hat{E_t^s} = \sum_{i=0}^{\infty} (\gamma\lambda)^i \delta_{t+i}^s, \tag{26}$$

$$\delta_t^s = R_t + \gamma V(S_{t+1},\theta) - V(S_t,\theta), \tag{27}$$

where λ is used to adjust the bias–variance trade-off and $V(S_t,\theta)$ is the state-value defined as the expected return when being in state S_t and parametrized by θ.

To enhance the exploration ability of the policy-based approaches, model-free and on-policy learning is introduced, where the learning is based on historical actions and the current policy, without any knowledge about the environment. One of the most known on-policy algorithms proposed by OpenAI is Proximal Policy Optimization (PPO) [49]. The objective function of PPO is presented by:

$$L^{CLIP}(\theta) = \mathbb{E}[(p_t(\theta)\hat{E_t^s}, clip(p_t(\theta), 1-\epsilon, 1+\epsilon)\hat{E_t^s})], \tag{28}$$

where the policy probability ratio $p_t(\theta)$ is defined as:

$$p_t(\theta) = \frac{\Pi(A_t|S_t,\theta)}{\Pi(A_t|S_t,\theta_{old})}. \tag{29}$$

The clip function is responsible to constraint p_t between $(1-\epsilon)$ and $(1+\epsilon)$, which prevents the system from moving outside this interval. In this way, the PPO objective is limited to the lower bound of the unclipped part. Owing to these advantages, we adopt the PPO method to design our channel assignment RL system.

- **DRL algorithm:** To stabilize the training and ensure the convergence of the learning process, multiple steps must be followed, which we illustrate in Algorithm 3. First, two networks are initialized with the same weights, from which two PPO policies are established. The training process starts by generating samples from the policy with fixed parameters θ_{old}, for different episodes. These samples are saved in a replay memory D (lines 16-18). More specifically, the agent receives the set of observations from the environment and takes an action using the policy $\Pi_{\theta_{old}}$ aiming at being highly rewarded (lines 11-15). After each episode, the estimator is computed and a random mini-batch from D is sampled. Next, the main policy Π_θ is updated by calculating the gradient of θ for each sample of the batch (lines 21-25). This mechanism, known as experience replay, is very important to learn from past experience and to reach the convergence and stabilization of the learning. In the end, we synchronize both policies by replacing θ_{old} with θ (line 29).

Algorithm 3 Channel assignment RL system

1: **Initialization:**
2: - Randomly set the parameters θ of the DNN to get Π_θ.
3: - Set the sampling policy $\Pi_{\theta_{old}}$ with $\theta_{old} \leftarrow \theta$.
4: **if** channel is uncorrelated **then**
5: F=1
6: **else** F=L
7: **end if**
8: **DRL Learning:**
9: **for** each episode e **do**
10: $i = 0$
11: **for** each frame $f = 1..F$ **do**
12: $assignedSF(f) = zeros(M, 6)$
13: **for** each step $t = i+1..i+K$ **do**
14: $S_t = \{assignedSF(f), g_t\}$
15: Select A_t based on $\Pi_{\theta_{old}}$
16: $R_t = C_1 * C_2 - \frac{\gamma_{th}\sigma^2_{A_t(1)}}{|\mathbf{g}_{t,A_t(1)}(i)|^2} 2^{A_t(2)}$
17: Update $assignedSF(f)$
18: Observe R_t and the next state S_{t+1}.
19: Save (S_t, A_t, R_t, S_{t+1}) in the experience
20: memory D.
21: **end for**
22: $i = i + K$
23: **end for**
24: - Compute the estimator $\hat{E_t^s}$ according to (26).
25: - Sample a mini-batch of (S_j, A_j, R_j, S_{j+1})
26: from the memory D.
27: - Update θ by maximizing the objective
28: function (28) and using the sampled data.
29: - Update the old policy: $\theta_{old} \leftarrow \theta$
30: **end for**

2) Energy management problem:

- **MDP environment design:** Similarly to the first problem, the second RL agent responsible for the energy management interacts with the LoRa wireless network. However, in this problem, an episode denotes the management of energy among L frames, while each step throughout the episode denotes the allocation of the energy X^h at a frame t. Therefore, the episode length will be equal to L.
- **States and actions:** At each time step t, the set of states received by the agent comprises the following components: $E(t)$ which is the amount of harvested energy, $X(t)$ which denotes the required energy at the frame t, and W_t presenting the energy cost. We remind that $X(t)$ is derived from the decision of the first RL agent, as depicted in Fig. 4. The system state S_t is, therefore, a vector defined as: $S_t = \{E(t), X(t), W_t\}$. Next, by observing the state, the agent decides the amount of energy $X^h(t) \geq 0$ to be harvested from the energy source. Finally, after each step, the set of states S_{t+1} is generated.
- **Reward function:** The reward function in the second RL system should match the objective of the optimization

problem (15) aiming to minimize the grid energy cost, which is equivalent to maximizing $W_t X_h$. In addition, this function should guarantee that the action $A_t = X^h(t)$ is taken while meeting the following requirements:

$$\begin{cases} C_3: & \sum_{i=1}^{t} A_i \leq \sum_{i=1}^{t} E(i) & \text{constraint (15.a)} \\ C_4: & \sum_{i=1}^{t-1} E(i) - \sum_{i=1}^{t} A_i \leq B_{max} & \text{constraint (15.b)} \\ C_5: & A_t \leq X(t) & \text{constraint (15.c)} \end{cases} \tag{30}$$

The constraints C_3, C_4, and C_5 in (30) are set to match the equations (14a), (14b), and (14c), respectively. Accordingly, the immediate reward is defined as follows:

$$R_t = C_3 * C_4 * C_5 + (W_t * A_t) * \mathbb{1}_{(C_3 * C_4 * C_5 = 1)} \tag{31}$$

As in the first RL model, the non-respect of one of the constraints (set to $C_3 * C_4 * C_5 = -1$ in this section) incurs invalid scenarios. This means that maximizing the rewards involves respecting the constraints to avoid the negative penalties. Moreover, in the second model, the high-level goal is to maximize the harvested energy. Thus, the weight of energy $W_t * A_t$ is added to the reward function, if all constraints are respected. In this way, maximizing the cumulative rewards throughout the episodes implies maximizing the amount of harvested energy and consequently reducing the usage of grid energy.

- **Agent design:** The energy management RL model is characterized by a dynamic state space and a non-discrete action space. Based on these factors, we select a Deep Deterministic Policy Gradient (DDPG)-based method to optimize the decision-making policy. DDPG is an actor-critic, off-policy, and model-free algorithm known for its high performance on continuous action and state spaces [50]. The actor-critic algorithms are generally composed of a policy and an action-value functions, where the policy function plays the role of the actor that takes decisions and interacts with the environment, whereas the action-value function is called a critic that is responsible to evaluate the performance of the actor.
- **DRL algorithm:** Two deep neural networks are adopted to build the approximation functions of the DDPG algorithm. The first DNN is used to train the actor and it is defined by the policy function $\mu(s|\theta^\mu)$ and the weights θ^μ. The second network, which corresponds to the critic, is described by the action-value function $Q(s, a|\theta^Q)$ with the related weights equal to θ^Q. The actor-critic networks are illustrated in Fig. 4. As done in the first system, steps to follow are illustrated in Algorithm 4. First, copies of the main networks, namely target networks, are created with the same NN parameters, i.e., $\theta'^\mu = \theta^\mu$ and $\theta'^Q = \theta^Q$ (lines 2-4). Next, the learning process is performed by generating episodes of experiences repeatedly (lines 6-31) and storing the generated MDP tuples in a replay buffer D (line 14). To ensure that the agent discovers different possible actions, an exploration policy is constructed by adding a noise sample through a noise process N^s. The selected action under the state S_t is defined by $A_t = \mu(s|\theta^\mu) + N_t^s$ (line 9). Once the decision is made, the direct reward is assigned and a new state S_{t+1} is given. To improve the critic policy, the agent samples each step a random mini-batch from the replay buffer D (lines 17-18) and calculates the target values $y(j)$ for each sample using the critic target network (lines 19-20). Meanwhile, the actor target network generates an action $\mu_{target}(S_{j+1})$ and feeds it to the critic target network. The DDPG critic policy is presented by the classical Bellman equation illustrated in (24). This critic network is updated by reducing the loss illustrated in (line 20) and expressed as:

$$L = \frac{1}{B} \sum_j (y(j) - Q(S_j, A_j|\theta^Q))^2. \tag{32}$$

Algorithm 4 Energy management RL system

1: **Initialization:**
2: - Randomly set the parameters θ^μ of the actor network
3: and θ^Q of the critic networks.
4: - Set weights of target networks: $\theta'^\mu \leftarrow \theta^\mu$ and $\theta'^Q \leftarrow \theta^Q$.
5: **DDPG Learning:**
6: **for** each episode e **do**
7: **for** each step $t = 1..L$ **do**
8: $S_t = \{E(t), X(t), W_t\}$
9: Select $A_t = \mu(S_t|\theta^\mu) + N_t^s$
10: **if** $C_3 * C_4 * C_5 = 1$ **then**
11: $R_t = W_t * A_t$
12: **else** $R_t = -1$
13: **end if**
14: - Observe R_t and the next state S_{t+1}.
15: - Save (S_t, A_t, R_t, S_{t+1}) in an experience
16: memory D.
17: - Sample a mini-batch of (S_j, A_j, R_j, S_{j+1}) of
18: size B from the memory D.
19: - Find target Q-value $y(j)$ from target Q-network:
20: $y(j) = R_j + \gamma Q_{target}(S_{j+1}, \mu_{target}(S_{j+1}|\theta'^\mu)|\theta'^Q)$
21: - Update the weights θ^Q of the critic network by
22: reducing the loss:
23: $L = \frac{1}{B} \sum_j (y(j) - Q(S_j, A_j|\theta^Q))^2$
24: - Update the weights θ^μ of the actor network using
25: the gradient policy:
26: $\nabla_{\theta^\mu} J \simeq \frac{1}{B} \sum_j \nabla_a Q(s, a|\theta^Q)|_{s=S_j, a=\mu(S_j)} \nabla_{\theta^\mu} \mu(s|\theta^\mu)|_{S_j}$
27: - Update the target networks:
28: $\theta'^Q \leftarrow \rho\theta^Q + (1-\rho)\theta'^Q$
29: $\theta'^\mu \leftarrow \rho\theta^\mu + (1-\rho)\theta'^\mu$
30: **end for**
31: **end for**

On the other hand, the actor network μ is updated using the gradient policy:

$$\nabla_{\theta^\mu} J \simeq E[\nabla_{\theta^\mu} Q(s, a|\theta^Q)|_{s=S_t, a=\mu(S_t|\theta^\mu)}]. \tag{33}$$

Similarly to the target network, the main actor generates an action $\mu(S_j)$ and inputs it to the critic network. However, in this process, the action and weights gradients, namely $\nabla_{\theta^\mu} Q(S_j, A_j|\theta^Q)$ and $\nabla_{\theta^\mu} \mu(S_j|\theta^\mu)$, are calculated using the automatic differentiation technique. These gradients al-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

low the approximation of the global policy gradient:

$$\nabla_{\theta^\mu} J \simeq \frac{1}{B}\sum_j \nabla_a Q(s,a|\theta^Q)|_{s=S_j,a=\mu(S_j)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{S_j}. \tag{34}$$

Finally, the target networks are softly updated using a small rate ρ, to always represent the most recent learning (lines 28-29):

$$\theta'^Q \leftarrow \rho\theta^Q + (1-\rho)\theta'^Q, \quad \theta'^\mu \leftarrow \rho\theta^\mu + (1-\rho)\theta'^\mu. \tag{35}$$

The theoretical complexity of our DRL resource management approach in hybrid energy LoRa wireless networks is based on the complexity of both RL systems. More specifically, the complexity of Algorithm 3 to accomplish one episode is determined by the loops between lines 11 and 21, where the number of iterations is equal to LK. Meanwhile, the complexity of Algorithm 4 is determined by one loop performing L iterations. Therefore, the complexity of the whole system can be expressed as $O(L(K+1))$. Furthermore, the complexity of the DNN decisions inside these loops depends on the neural network. In our case, we adopt an MLP deep neural network comprising 2 layers of 64 neurons, which is a light-weight network.

VII. Evaluation Results

In this section, Monte Carlo and reinforcement learning simulations are done to evaluate the proposed resource management schemes in LoRa wireless networks by averaging up to 10000 realizations. Monte Carlo simulations are based on repeating random sampling to obtain numerical results. The underlying concept is to use randomness to solve problems having a probabilistic interpretation. In our work, we use Monte Carlo simulation to simulate the random variables that represent the channel, energy, and device distribution. The users are uniformly distributed within a circular cell. The grid energy consumption weights W_i are randomly generated according to a standard uniform distribution. The simulation parameters used in this section are summarized in Table II.

The RL algorithms are validated based to the parameters

TABLE II: Simulation Parameters.

Symbol	Description	Value
B_{max}	max battery capacity	200 J
K	number of devices	35
L	number of frames	50
M	number of channels	5
	path loss exponent	3.7 [43]
	noise PSD	-174 dBm/Hz [43]
	circuit power	30 dBm [51]
	cell radius	500 m

defined in Tables III and IV. These parameters are empirically adopted and we expect that, using the same values, similar architectures perform identically.

A. Uncorrelated Channel

First, we investigate the performance of the resource management schemes in LoRa networks considering uncorrelated channels. Particularly, we will start by examining the performance of the RL approaches in terms of convergence and ability to respect the system constraints. Fig. 5 illustrates the variation of the cumulative rewards over the training episodes, for the channel allocation RL system.

TABLE III: Hyper-parameters of the channel assignment RL system.

Parameter	Description	Value
γ	Gamma	0.99
α	Learning rate	0.0001
Policy	DNN policy	MLP, 2 layers of 64
ϵ	cliprange	0.2
λ	Adjusting factor	0.01
	PPO epoch	4

TABLE IV: Hyper-parameters of the energy management RL system.

Parameter	Description	Value
γ	Gamma	0.99
α_a	Learning rate of actor	0.0001
α_c	Learning rate of critic	0.001
Policy	DNN policy	MLP, 2 layers of 64
bz	Buffer size	25000
N_t^s	Noise parameter	None
ρ	soft update	0.001

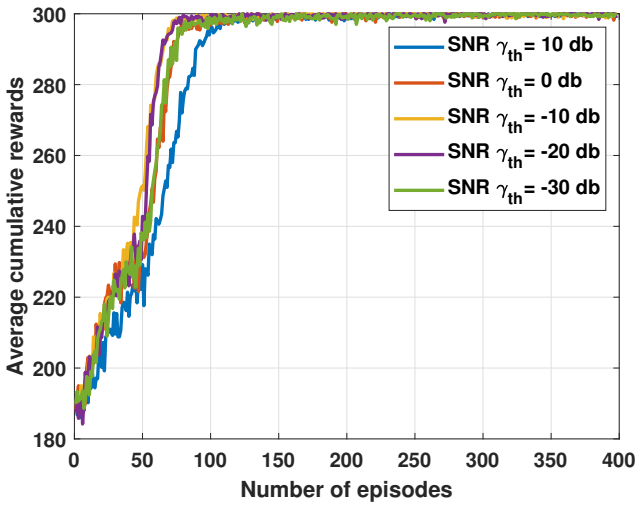


Fig. 5: Average cumulative rewards vs. training episodes of the channel allocation RL system ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

We note that the presented rewards are averaged over a window of 5 for all SNR levels to see the behavior of the system during training. In the beginning of the learning process, all decisions are taken randomly in order to have

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

an initial estimation of the RL policy. At this stage, we can notice that the reward is low, which means that the constraints described in eq. (21) are not respected. However, as the number of episodes increases, the system starts to learn how to respect different constraints and assign optimal allocations. After the learning process, the stability is reached, which confirms the convergence of the channel allocation RL system.

Fig. 6 shows the accuracy of both RL systems, namely channel assignment and energy management. We define the accuracy of the RL as its ability to respect different constraints. More specifically, the accuracy is equal to the percentage of episodes where all system requirements are satisfied, after the convergence. We can see that the accuracy of both RL models is very high reaching 80% and more for most of the SNR levels, which means more than 80% of episodes respect the defined constraints (e.g. channels, SFs, and energy constraints).

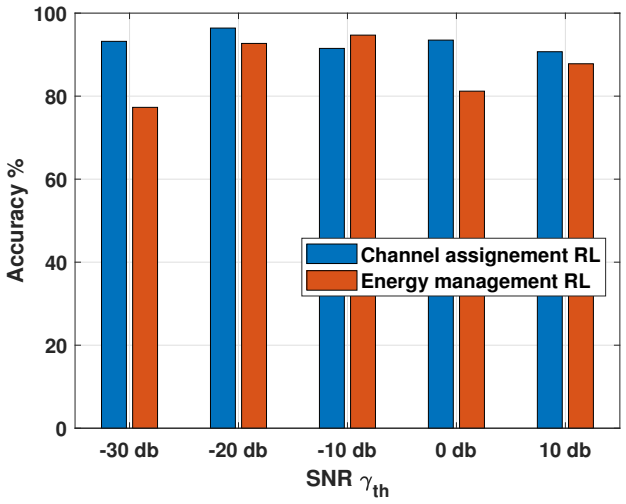


Fig. 6: Accuracy in terms of meeting constraints for both RL systems ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

Next, we will present the performance of the proposed HURMA algorithm and energy management RL system compared to the optimal solution. Fig. 7 plots the Grid energy cost of the proposed low complexity algorithm HURMA and the RL system as a function of the SNR. This figure is drawn for only limited number of devices $K = 6$, number of channels $M = 2$ and a set of SFs $\Gamma = \{7, 8, 9\}$ due to the high complexity of the exhaustive search optimal algorithm. It is clear that the HURMA and the RL approach significantly outperform the random scheme (random channel and SF assignment) thanks to the adequate channel and SF assignment, which significantly saves the transmitted power. Also, they achieve a performance near to the optimal specifically in low SNR region in which LoRa operates. However, we can notice that the HURMA heuristic presents a better performance compared to the RL approach. This can be explained by the fact that no MDP process is set, when channels are uncorrelated. This means that the episode steps are independent and the state transition

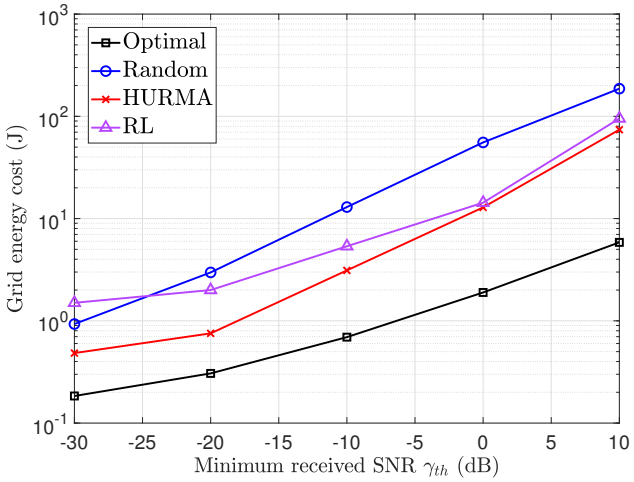


Fig. 7: Optimal grid energy cost versus SNR target ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

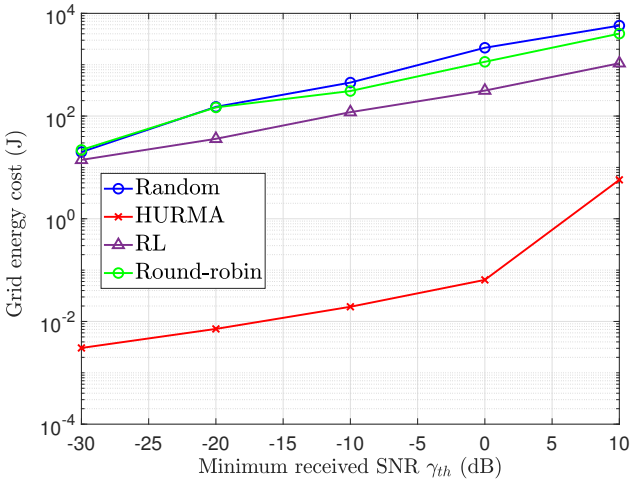


Fig. 8: Grid energy cost versus SNR target with heuristic resource management schemes ($M = 5, K = 35, B_{max} = 200$ J, $\Gamma = \{7, 8, 9, 10, 11, 12\}$).

follows a uniform distribution. Therefore, the RL policy only learns how to respect the constraints to maximize the reward. Additionally, it captures the channels and SFs that have higher probabilities to minimize the required energy, owing to the experience memory D storing past allocation decisions.

To summarize, when channels are uncorrelated, HURMA outperforms the RL approaches due to its efficient energy management design. Still, the RL shows a good performance, while presenting a lower complexity. Particularly, our RL system uses an MLP predictive network composed of 2 layers of 64 neurons, which is a light-weight DNN model with a negligible complexity. Moreover, the edge of the RL over heuristic based approaches is its run-time ability to adapt to the environment changes (e.g., SNR, number of channels, and number of users.), thanks to its continual online learning.

In Fig. 8, the performance of HURMA is shown as a function of the SNR for higher number of number of devices

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

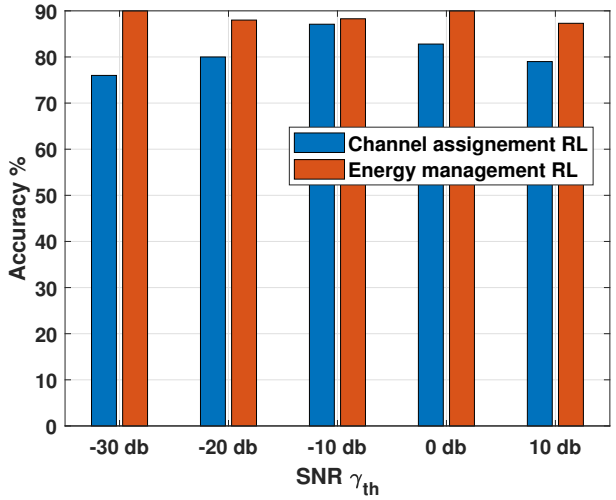


Fig. 9: Accuracy in terms of meeting constraints for both RL systems ($M = 5, K = 35, B_{max} = 200$ J, $\Gamma = \{7, 8, 9, 10, 11, 12\}$).

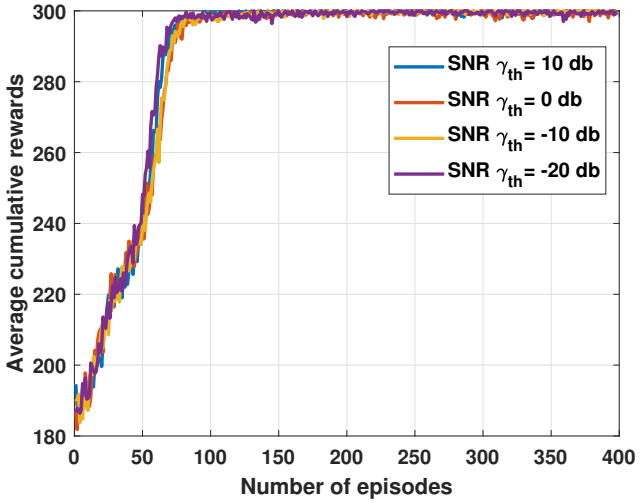


Fig. 11: Average cumulative rewards vs. training episodes of the channel allocation RL system ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

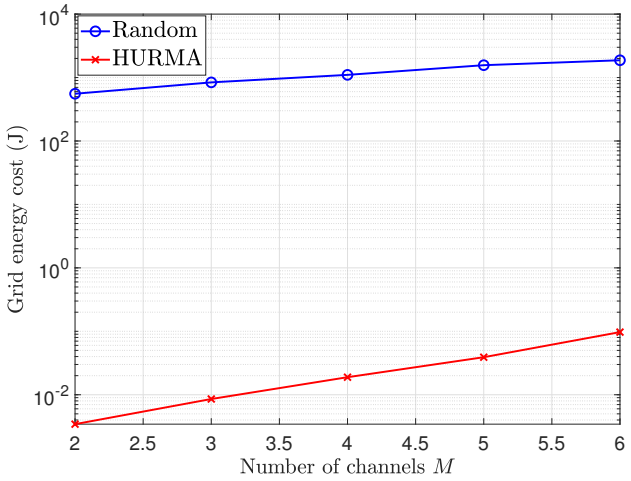


Fig. 10: Grid energy cost versus number of channels with heuristic resource management schemes ($K = 40, B_{max} = 200$ J, $\Gamma = \{7, 8, 9, 10, 11, 12\}, \gamma_{th} = 0$ dB).

$K = 35$, number of channels $M = 5$ and a set of SFs $\Gamma = \{7, 8, 9, 10, 11, 12\}$. It is clear that the proposed algorithm HURMA significantly saves grid energy cost compared to the RL approach and round-robin scheduling, which confirms the performance of the heuristic over the online learning when channels are uncorrelated. We note that the accuracy of both RL models leading to energy saving is verified in Fig. 9.

Fig. 10 shows the performance of the proposed scheme HURMA as a function of the number of channels K. The increase of the number of channels allows to schedules more devices, which increases the total grid energy cost. Moreover, HURMA performs very well and the performance gap with the random scheme keeps almost unchanged when the number of channels increases.

B. Time-correlated Channel

Now, the performance of the resource management schemes in LoRa networks is investigated considering time-correlated channels. Fig. 11 depicts the average cumulative rewards of the channel allocation RL over the training episodes, smoothed over a window of 5. Similarly to the uncorrelated scenario, the system applies the trial and error process, until reaching the convergence phase. Fig. 12 depicts the convergence of

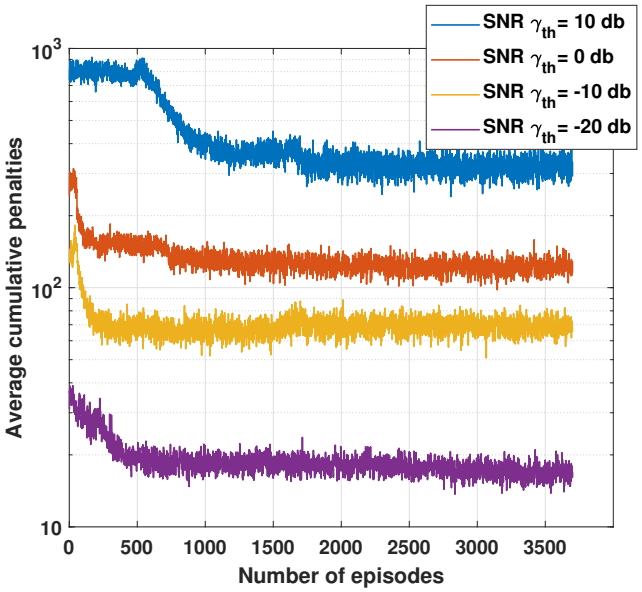


Fig. 12: Average cumulative penalties vs. training episodes of the channel allocation RL system ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

the penalties used to learn the optimal channel assignment strategy. In fact, the RL penalties match the objective function of the optimization problem (14), which is added to the reward function as illustrated in eq. (22).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

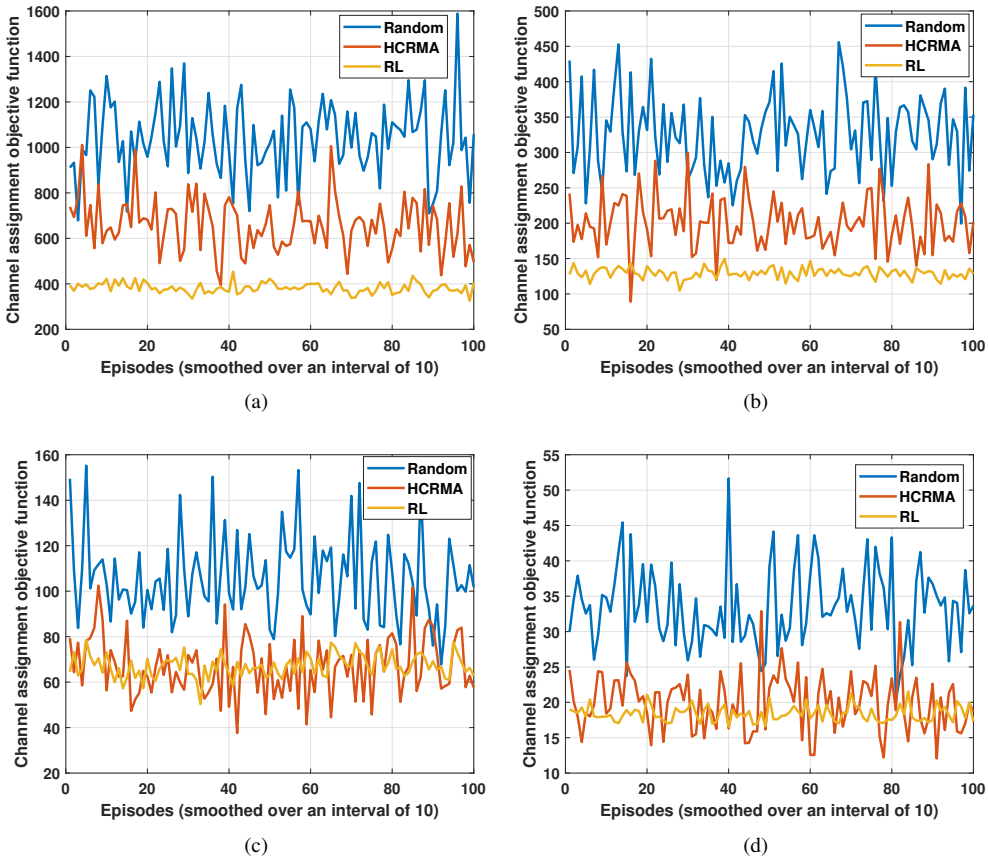


Fig. 13: Average penalties (channel assignment objective function) vs. smoothed episodes ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

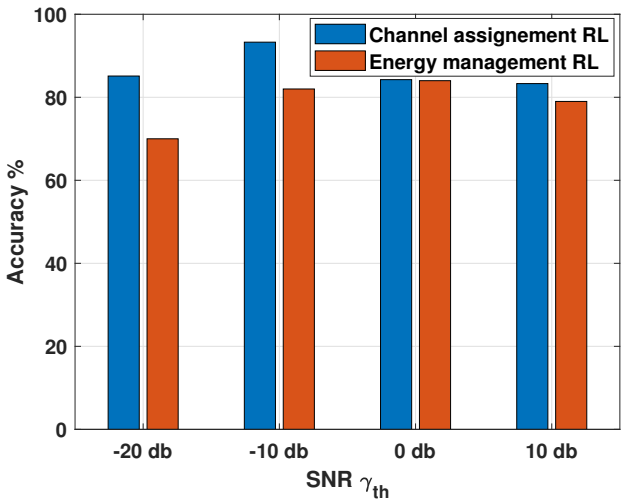


Fig. 14: Accuracy in terms of meeting constraints for both RL systems ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

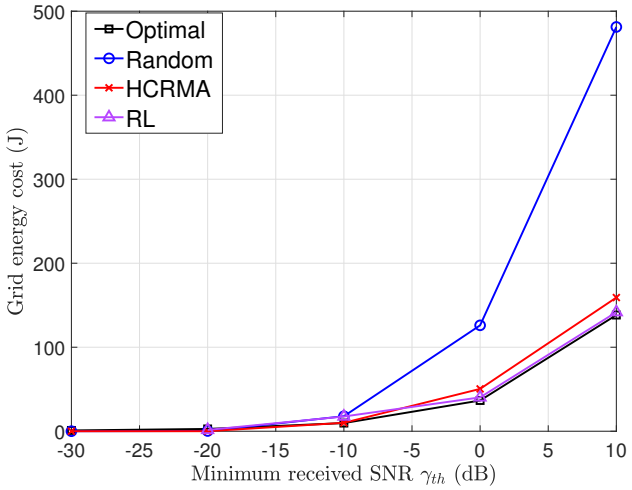


Fig. 15: Optimal grid energy cost versus SNR target ($M = 2, K = 6, B_{max} = 10$ J, $\Gamma = \{7, 8, 9\}$).

According to the latter equation, the RL agent selects the optimal allocations by minimizing the penalties over the training episodes. We note that the cumulative penalties are smoothed over a window of 10. On these bases, we notice that the convergence is reached after 5000 episodes, which is not the case of the cumulative rewards that converge faster. This can be explained by the fact that the exploration space of channel allocation tasks is large. However, after learning the optimal actions related to the given states, reasonable decisions will be taken on the fly. It is worth to mention that this convergence pattern is not observed, when channels are uncorrelated due to the insignificance of the MDP process.

To further evaluate the performance of the channel assignment

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

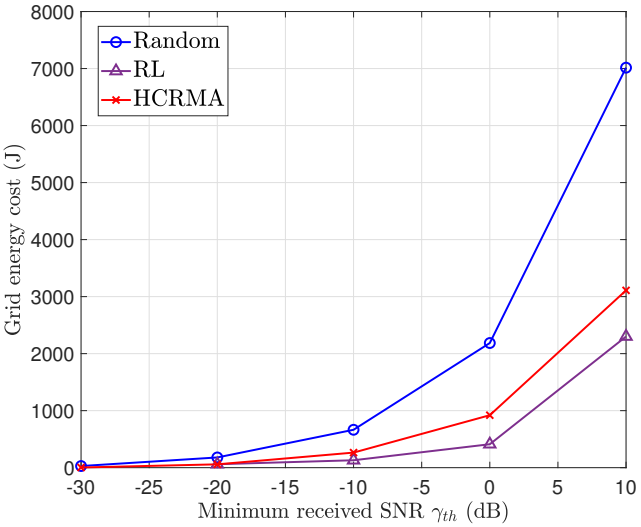


Fig. 16: Grid energy cost versus SNR target with heuristic resource management schemes ($M = 3, K = 20, B_{max} = 200$ J, $\Gamma = \{7, 8, 9, 10, 11, 12\}$).

RL system over episodes, we compared it to the HCRMA heuristic and the Random resource allocation. Fig. 13 presents the average cumulative penalties (i.e., objective function of the channel allocation problem) over 1000 episodes for different approaches. We can see that the RL approach outperforms the heuristic in terms of allocation decisions and stability, when the SNR is equal to 10 or 0. In low SNR region, the RL presents a similar or slightly better performance.

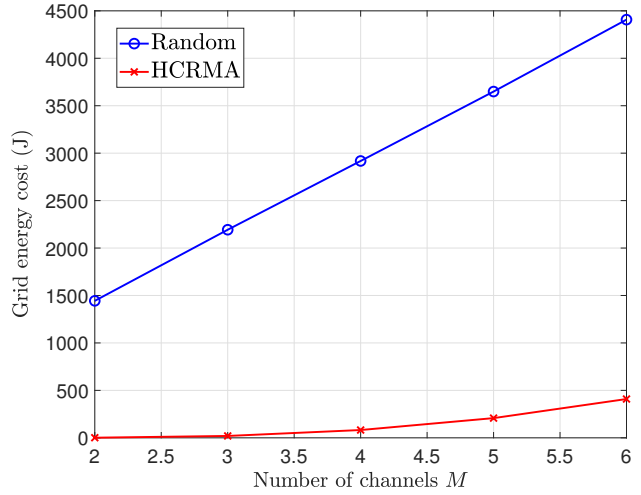


Fig. 18: Grid energy cost versus number of channels with heuristic resource management schemes ($K = 40, B_{max} = 200$ J, $\Gamma = \{7, 8, 9, 10, 11, 12\}, \gamma_{th} = 0$ dB).

Next, the performance of the second RL responsible for energy management is examined through evaluating its accuracy first (see Fig. 14), and then benchmarking it against the low complexity algorithm HCRMA and the optimal solution (see Fig. 15). Fig. 14 shows that both RLs have a high ability to respect different requirements and constraints of the system. Fig. 15 presents the grid energy cost as a function of the SNR, for different resource management approaches. Similarly to uncorrelated channel scenario, the optimal resource management solution is derived with high computational complexity. The proposed algorithm HCRMA and the RL system achieve high system performance thanks to their adequate SF and channel assignment methods. Indeed, the performance gap with the optimal is tight, for both approaches. However, in the correlated channel scenario, the RL approach outperforms the heuristic-based strategy thanks to its capacity to learn a sub-optimal allocation strategy, predict the required energy for the next frames, and manage the existing resources accordingly. This is not the case of HCRMA that executes greedy decisions.

We plot the performance of HCRMA and the corresponding RL system for higher number of devices $K = 20$. Fig. 16 presents the grid energy cost versus the SNR target, for different resource management schemes. Clearly, the HCRMA heuristic and the RL approach outperform the random scheme in terms of grid power consumption cost for wide range of SNR. Additionally, we can see that the RL gives better results compared to the heuristic, owing to its online learning ability. The convergence and accuracy performance of the channel assignment and energy management RL models are confirmed in Fig. 17, where we present the average cumulative rewards and penalties, and the accuracy in terms of respecting the constraints.

The performance of the proposed scheme HCRMA is shown in Fig. 18 as a function of the number of channels M. The increase of the number of channels allows to us schedule more devices, which increases the total grid energy cost. HCRMA performs very well in terms of grid power consumption cost and the performance gap with the random scheme keeps increasing when the number of channels increases.

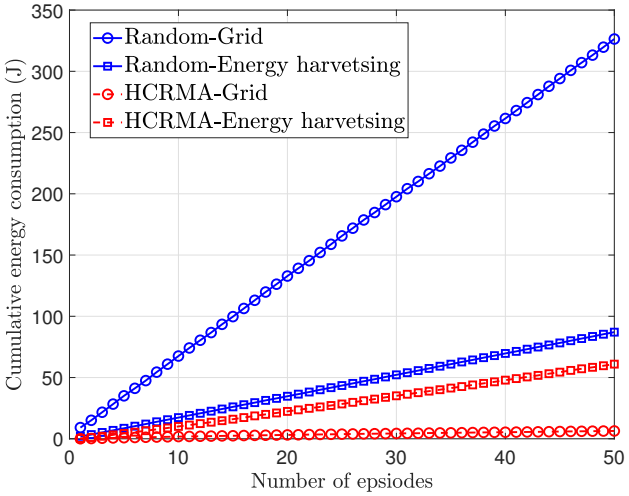


Fig. 19: Cumulative consumed grid energy and harvested energy versus number of episodes with heuristic resource management schemes. ($M = 5, K = 35, \gamma_{th} = -20dB, B_{max} = 200$ J, $\Gamma = \{7, 8, 9, 10, 11, 12\}$).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3110996, IEEE Internet of Things Journal

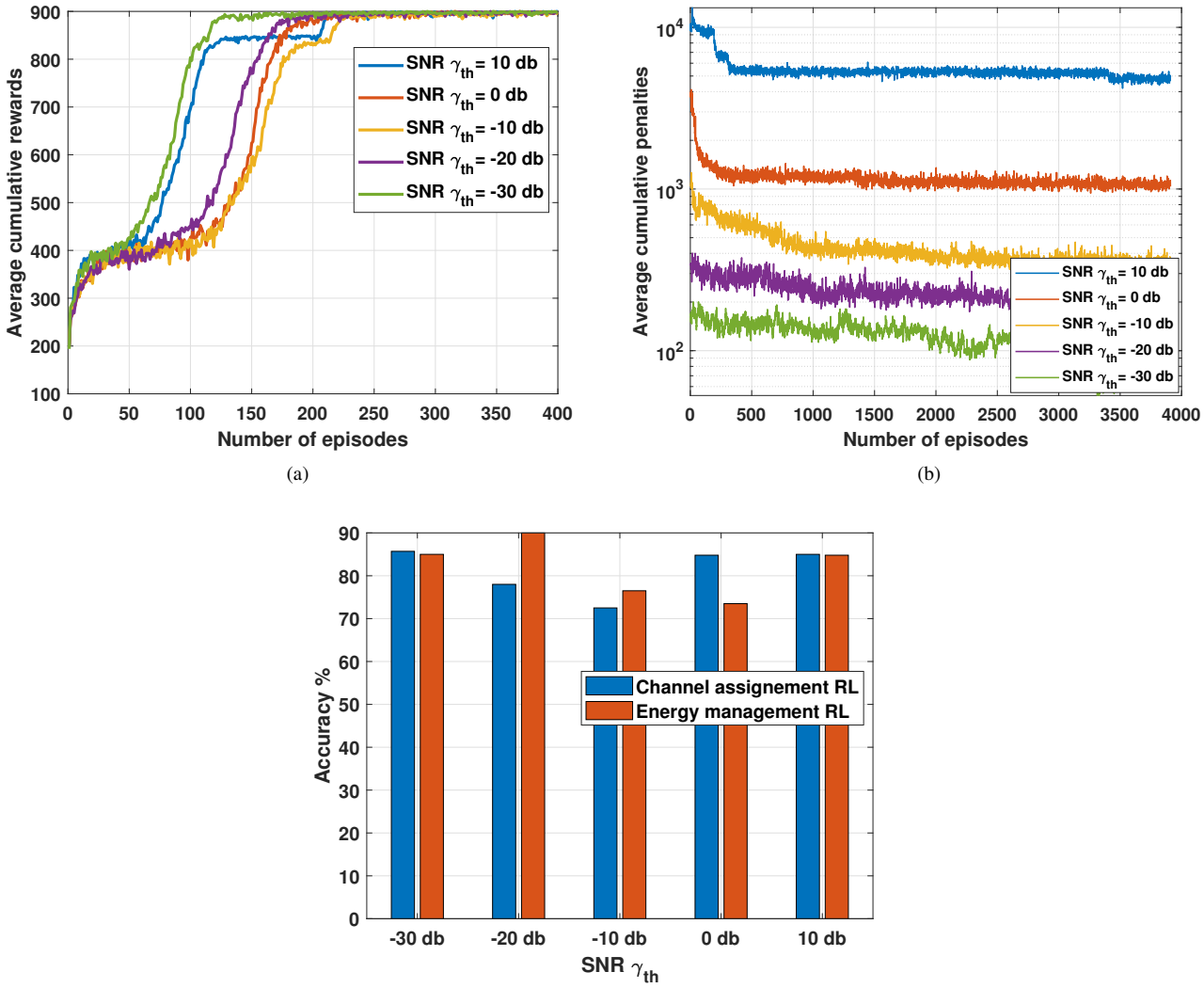


Fig. 17: Performance of the RL models ($M = 3, K = 20, B_{max} = 200$ J, $\Gamma = \{7, 8, 9, 10, 11, 12\}$): (a) Average cumulative rewards vs. training episodes of the channel assignment RL system; (b) Average cumulative penalties vs. training episodes of the channel assignment RL system; (c) Accuracy in terms of meeting constraints for both RL systems.

We show in Fig. 19 the cumulative consumed grid energy and harvested energy over time. It can be seen that HCRMA allows to reduce both the consumed energy from the grid and from the battery. Moreover, HCRMA optimizes the use of the renewable energy which allows to minimize the grid energy cost.

VIII. Conclusion

This paper has investigated energy-efficient resource management in green LoRa wireless network powered by both a renewable energy source and the conventional grid. A grid power consumption minimization problem subject to the devices' quality of service demands, has been formulated. The optimal energy management solution which consists of device scheduling, SF and channel assignment, and energy management, has been solved. The online resource management has been also investigated considering both scenarios uncorrelated and time-correlated channels by developing low complexity heuristic algorithms. Moreover, smart and adaptable resource management schemes based on RL have been developed taking into account the channel and energy correlation with the objective to improve the power consumption in LoRa wireless networks. Simulation results show that the proposed resource management approaches allow efficient use of renewable energy in LoRa wireless networks.

Future works may cover the study of model-based RL frameworks and their performance on LoRa systems compared to our proposed model-free RL approach. We will also focus on developing efficient resource management schemes for federated learning over LoRa wireless networks. Moreover, SWIPT system could be incorporated in LoRa wireless networks, and efficient resource management schemes may be developed.

Acknowledgment

This work was made possible by NPRP-Standard (NPRP-S) Thirteen (13th) Cycle grant # NPRP13S-0205-200265 from

the Qatar National Research Fund (QNRF) (a member of Qatar Foundation) and the TÜBITAK—QNRF Joint Funding Program grant (AICC03-0324-200005) from the Scientific and Technological Research Council of Turkey and QNRF. The findings herein reflect the work, and are solely the responsibility, of the authors.

REFERENCES

[1] J. P. S. Sundaram, W. Du, and Z. Zhao, "A survey on LoRa networking: Research problems, current solutions, and open issues," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 371-388, First quarter 2020.

[2] M. Alenezi, K. K. Chai, Y. Chen, S. Jimaa, "Ultra-dense LoRaWAN: Reviews and challenges," *IET Commun.*, vol. 14, no. 9, pp. 1361-1371, Apr. 2020.

[3] A. Lavric, and V. Popa, "Internet of things and LoRa low-power wide-area networks: a survey," *in Proc. IEEE Int. Symp. Signals Circuits Syst. (ISSCS)*, Lasi, Romania, Jul. 2017, pp. 1-5.

[4] Au. Ikpehai, B. Adebisi, K. M. Rabie, K. Anoh, R. E. Ande, M. Hammoudeh, H. Gacanin, and U. M. Mbanaso, "Low-power wide area network technologies for Internet-of-things: A comparative review," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2225-2240, Apr. 2019.

[5] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, "Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios," *IEEE Wireless Commun.*, vol. 23, no. 5, pp. 60-67, Oct. 2016.

[6] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of LoRa: Long range & low power networks for the internet of things," *Sensors*, vol. 16, no. 9, pp. 1-18, Sep. 2016.

[7] L. Vangelista, "Frequency shift chirp modulation: The LoRa modulation," *IEEE Signal Process. Lett.*, vol. 24, no. 12, pp. 1818-1821, Dec. 2017.

[8] M. Alsharef, A. M. Hamed, and R. K. Rao, 'Error rate performance of digital chirp communication system over fading channels," *in Proc. World Congress Eng. Comput. Sci. (WCECS)*, San Francisco, USA, Oct. 2015, pp. 727-732.

[9] O. Afisiadis, M. Cotting, A. Burg, and A. Balatsoukas-Stimming, "On the error rate of the LoRa modulation with interference," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 1292-1304, Feb. 2020.

[10] O. Georgiou and U. Raza, "Low power wide area Network analysis: Can LoRa scale?," *IEEE Wireless Commun. Lett.*, vol. 6, no. 2, pp. 162-165, Apr. 2017.

[11] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT press, 2018.

[12] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol.8, no.3, pp. 279-292, May 1992.

[13] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K.Fidjeland, G. Ostrovski, and S. Petersen, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[14] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, no. 3, pp. 229–256, May 1992.

[15] D. Ma, G. Lan, M. Hassan, W. Hu, and S. K. Da, "Sensing, computing, and communications for energy harvesting IoTs: A survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1222-1250, May. 2020.

[16] W. Zeng, Y. R. Zheng, and R. Schober, "Online resource allocation for energy harvesting downlink multiuser systems: Precoding with modulation, coding rate, and subchannel selection," *IEEE Trans. Wireless Commun.*, vol. 14, no. 10, pp. 5780-5794, Oct. 2015.

[17] R. Hamdi, M. Qaraqe, and S. Althunibat, 'Dynamic spreading factor assignment in LoRa wireless networks," *in Proc. IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, Jun. 2020, pp. 1-5.

[18] L. Amichi, M. Kaneko, N. El Rachkidy, and A. Guitton, 'Spreading factor allocation strategy for LoRa networks under imperfect orthogonality," *in Proc. IEEE Int. Conf. Commun. (ICC)*, Shanghai, China, May 2019, pp. 1-7.

[19] F. Cuomo, M. Campo, A. Caponi, G. Bianchi, G. Rossini, and P. Pisani, 'EXPLoRa: Extending the performance of LoRa by suitable spreading factor allocations," *in Proc. IEEE Int. Wireless Mobile Comput. Netw. Commun. (WiMob)*, Rome, Italy, Oct. 2017, pp. 1-8.

[20] L. Amichi, M. Kaneko, E. H. Fukuda, N. El Rachkidy, and A. Guitton, "Joint allocation strategies of power and spreading factors with imperfect orthogonality in LoRa networks," *IEEE Trans. Commun.*, vol. 68, no. 6, pp. 3750-3765, Jun. 2020.

[21] B. Su, Z. Qin, and Q. Ni, "Energy efficient uplink transmissions in LoRa networks," *IEEE Trans. Commun.*, vol. 68, no. 8, pp. 4960-4972, Aug. 2020.

[22] J. T. Lim and Y. Han, "Spreading factor allocation for massive connectivity in LoRa systems," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 800-803, Apr. 2018.

[23] A. Farhad, D. H. Kim, and J. Y. Pyun, "Resource allocation to massive internet of things in LoRaWANs," *Sensors*, vol. 20, no. 9, pp. 1-20, May 2020.

[24] A. Farhad, D. H. Kim, P. Sthapit, and J. Y. Pyun, 'Interference-aware spreading factor assignment scheme for the massive LoRaWAN network," *in Proc. IEEE Int. Conf. Electron. Inf. Commun. (ICEIC)*, Auckland, New Zealand, Jan. 2019, pp. 1-2.

[25] A. Farhad, D. H. Kim, S. Subedi, and J. Y. Pyun, "Enhanced LoRaWAN adaptive data rate for mobile internet of things devices," *Sensors*, vol. 20, no. 22, pp. 1-21, Nov. 2020.

[26] M. S. Aslam, A. Khan, A. Atif, S. A. Hassan, A. Mahmood, H. K. Qureshi, and M. Gidlund, "Exploring multi-hop LoRa for green smart cities," *IEEE Netw.*, vol. 34, no. 2, pp. 225-231, Apr. 2020.

[27] G. Zhu, C. H. Liao, T. Sakdejayont, I. W. Lai, Y. Narusue, and H. Morikawa, 'Improving the capacity of a mesh LoRa network by spreading-factor-based network clustering," *IEEE Access*, vol. 7, pp. 21584-21596, Feb. 2019.

[28] D. Zorbas, G. Z. Papadopoulosy, P. Mailley, N. Montavonty, and C. Douligeris, 'Improving LoRa network capacity using multiple spreading factor configurations," *in Proc. IEEE Int. Conf. Telecommun. (ICT)*, Saint Malo, France, Jun. 2018, pp. 516-520.

[29] J. Haxhibeqiri, I. Moerman, and J. Hoebeke, "Low overhead scheduling of LoRa transmissions for improved scalability," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3097-3109, Apr. 2019.

[30] A. Furtado, J. Pacheco, and R. Oliveira, "PHY/MAC uplink performance of class A LoRa networks," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6528-6538, Jul. 2020.

[31] W. S. Jeon, and D. G. Jeong, "Adaptive uplink rate control for confirmed class A transmission in LoRa networks," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 10361-10374, Oct. 2020.

[32] R. Hamdi and M. Qaraqe, 'Resource management in energy harvesting powered LoRa wireless networks," *in Proc. IEEE Int. Conf. Commun. (ICC)*, Montreal, Canada, Jun. 2021, pp. 1-6.

[33] R. B. Sørensen, D. Min Kim, J. J. Nielsen, and P. Popovski, "Analysis of latency and MAC-Layer performance for class A LoRaWAN," *IEEE Wireless Commun. Lett.*, vol. 6, no. 5, pp. 566-569, Oct. 2017.

[34] A. Waret, M. Kaneko, A. Guitton, and N. El Rachkidy, "LoRa throughput analysis with imperfect spreading factor orthogonality," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 408-411, Apr. 2019.

[35] D. Ron, C. J. Lee, K. Lee, H. H. Choi, and J. R. Lee, "Performance analysis and optimization of downlink transmission in LoRaWAN class B mode," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7836-7847, Aug. 2020.

[36] G. Premsankar, B. Ghaddar, M. Slabicki, and M. D. Francesco, "Optimal configuration of LoRa networks in smart cities," *IEEE Trans. Ind. Inform.*, vol. 16, no. 12, pp. 7243-7254, Dec. 2020.

[37] Y. Li, J. Yang, and J. Wang, "DyLoRa: Towards energy efficient dynamic LoRa transmission control," *in Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Virtual Conference, Jul. 2020, pp. 2312-2320.

[38] P. Edward, M. El-Aasser, M. Ashour, and T. Elshabrawy, "Interleaved chirp spreading LoRa as a parallel network to enhance LoRa capacity," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3864-3874, Mar. 2021.

[39] D. Xu, X. Chen, N. Zhang, N. Ding, J. Zhang, D. Fang, and T. Gu, "Cantor: Improving goodput in LoRa concurrent transmission," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1519-1532, Feb. 2021.

[40] Y. Sun, J. Chen, S. He, and Z. Shi, "High-confidence gateway planning and performance evaluation of a hybrid LoRa network," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1071-1081, Jan. 2021.

[41] Y. H. Tehrani, A. Amini, and S. M. Atarodi, "A tree-structured LoRa network for energy efficiency," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 6002-6011, Apr. 2021.

[42] S. Lee, J. Lee, H.-S. Park, and J. K. Cho, "A novel fair and scalable relay control scheme for internet of things in LoRa-based low-power wide-area networks," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5985-6001, Apr. 2021.

[43] M. S. H. Abad, O. Ercetin, and D. Gündüz, "Channel sensing and communication over a time-correlated channel with an energy harvesting transmitter," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 1, pp. 114-126, Mar. 2018.

[44] D Niyato, E Hossain, and A Fallahi, "Sleep and wakeup strategies in solar-powered wireless sensor/mesh networks: Performance Analysis and Optimization," *IEEE Trans. Mobile Comput.*, vol. 6, no. 2, pp. 221-236, Feb. 2007.

[45] 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.

[46] Y. Che, L. Duan and R. Zhang, "Dynamic base station operation in large-scale green cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3127-3141, Dec. 2016.

[47] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: complexity and duality," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, Feb. 2008.

[48] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," 2009.

[49] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, Aug. 2017.

[50] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, Jul. 2019.

[51] R. Kumar and J. Gurugubelli, "How green the LTE technology can be?," *in Proc. Int. Conf. Wireless Commun. Veh. Technol. Inf. Theory Aerosp. Electron. Syst. Technol.*, Chennai, India, Jul. 2011, pp. 1-5.