Improvement of the LPWAN AMI Backhaul's Latency thanks to Reinforcement Learning Algorithms

Rémi Bonnefoi, Christophe Moy, Jacques Palicot

RESEARCH　　　　　　　　　　　　　　　　　　　　　　　　　　**Open Access**

CrossMark

Improvement of the LPWAN AMI backhaul's latency thanks to reinforcement learning algorithms

Rémi Bonnefoi[1]* ⓘ, Christophe Moy[2] and Jacques Palicot[1]

Abstract

Low power wide area networks (LPWANs) have been recently deployed for long-range machine-to-machine (M2M) communications. These networks have been proposed for many applications and in particular for the communications of the advanced metering infrastructure (AMI) backhaul of the smart grid. However, they rely on simple access schemes that may suffer from important latency, which is one of the main performance indicators in smart grid communications. In this article, we apply reinforcement learning (RL) algorithms to reduce the latency of AMI communications in LPWANs. For that purpose, we first study the collision probability in an unslotted ALOHA-based LPWAN AMI backhaul which uses the LoRaWAN acknowledgement procedure. Then, we analyse the effect of collisions on the latency for different frequency access schemes. We finally show that RL algorithms can be used for the purpose of frequency selection in these networks and reduce the latency of the AMI backhaul in LPWANs. Numerical results show that non-coordinated algorithms featuring a very low complexity reduce the collision probability by 14% and the mean latency by 40%.

Keywords: Advanced metering infrastructure (AMI), Low power wide area networks (LPWANs), Internet of Things (IoT), LoRaWAN, ALOHA, Smart grid communications

1 Introduction

The increasing development of renewable energy production and the high cost associated with power failures have been driving electricity operators towards the development of new functions enabling the real-time management of the electrical grid. Thanks to these improvements, traditional electrical grids have morphed progressively into the so-called smart grids.

The transformation of the electrical grid into the smart grid mainly impacts the distribution grid. Three functions are necessary to manage the smart distribution grid: the advanced metering infrastructure (AMI), the distribution automation (DA) and the management of distributed energy resources (DER) [1]. Furthermore, the management of a smart grid relies on a network of smart sensors and actuators deployed all along the grid. One of the main

roles of these devices is to provide an overall view of the state of the grid, in a way that must be as continuous as possible. This cannot be done without an efficient communication system. Each of the functions developed for the management of the grid has its own constraints in term of throughput, latency, security and reliability [2]. As a consequence, the design of an efficient smart grid communication infrastructure is one of the key challenges in the smart grid deployment.

In AMI, smart meters measure and report the electricity consumption to a control center. The information received by the control center is then used to manage both the electricity production and consumption. In particular, the control center is in charge of computing the new electricity price which is applied to consumers.

Many communication standards and protocols are envisioned for the smart grid and in particular for AMI communications [3], and the use of both wired and wireless technologies have been investigated. AMI communications are done through two networks: the neighbourhood network, which links smart meters and local aggregators,

*Correspondence: remi.bonnefoi@centralesupelec.fr
[1]CentraleSupélec/IETR, CentraleSupélec campus de Rennes, Avenue de la boulaie, 35510, Cesson-Sévigné, France
Full list of author information is available at the end of the article

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking* (2018) 2018:34

Page 2 of 18

and the AMI backhaul linking aggregators and the control center [4]. As an example, in France, power-line communications (PLCs) are used for the neighbourhood network and the General Packet Radio Service (GPRS) network is used for the AMI backhaul [5].

Besides, LPWANs rely on wireless telecommunication standards recently designed to handle a large number of long-range uplink communications and have been identifed as potential networks for AMI communications [6, 7]. In these networks, a large number of low power end-devices send short packets to a base station or gateway. Moreover, in LPWANs, the band is divided in narrowband channels, which are continuously monitored by the base station in order to collect all the uplink packets sent by end-devices.

A wide range of LPWAN standards have recently been proposed [8]. These standards can be sorted in two categories. On the one hand, there are slotted protocols such as the NarrowBand IoT (NB-IoT) standard [9], designed by the 3rd Generation Partnership Project (3GPP) and the Weightless standard [10]. On the other hand, there are unslotted (or pure) ALOHA-based protocols [11] such as the LoRaWAN standard [12] and the protocol used by Sigfox[1], which is based on ultra narrow band (UNB) [13] communications. In these unslotted protocols, the signalling is reduced so as to mitigate the end-device energy consumption, and transmissions are asynchronous and event-driven. Moreover, in some of these LPWANs, an acknowledgement is used to avoid unnecessary retransmissions. Furthermore, in order to limit the impact of the acknowledgement on the end-device's energy consumption, the receive window, during which the end-device waits for an acknowledgement, is shortened. In order to do that, the acknowledgement is always sent at the same time. In other words, the receive delay between the end of the uplink packet, and the transmission of the acknowledgement in the downlink is constant. Thanks to this simple mechanism, the end-device is able to sense the channel during a very short time and detect the preamble of the downlink packet. This solution is used in the LoRaWAN standard [12, 14]. More precisely, in this standard, and that is the case in several regions (e.g. Europe, China) [15], an acknowledgement is sent into the channel being used for the uplink transmission 1 s after the end of the reception of the uplink packet by the base station [12].

In [16], the authors analyse the capacity limits of LoRaWAN in an AMI scenario. In the present article, we also consider a LoRaWAN AMI backhaul but we focus our analysis on latency. LPWANs operate in unlicensed bands which are shared by many end-devices which use different standards and have different behaviours or capabilities, which depend on their manufacturers and on the requirements of their applications (temperature sensing, smart grid monitoring, etc.). These heterogeneous devices can create a heavy traffic which is unevenly distributed in channels. This causes packet collisions and improves, consequently, the latency which is one of the key performance indicators of AMI communications [2]. In this article, we show that reinforcement learning algorithms and more precisely multi-armed bandit (MAB) learning reduce the latency of the AMI backhaul in LPWANs.

In the first part, we propose to analyse the collision probability in a LPWAN which uses the acknowledgement mechanism of the LoRaWAN standard and the effect of collisions on the latency for several access schemes. The analysis of collisions in ALOHA networks is an old topic [11, 17]. However, recent LPWAN standards implement new solutions which have not been previously considered. As an example, the protocol used by Sigfox is the first time/frequency unslotted standard and its performance has been recently evaluated in [13, 18]. Moreover, one of the specificities of the LoRaWAN standard is its acknowledgement mechanism. Indeed, in this standard, an acknowledgement can be sent into the channel used for the purpose of uplink communications after a fixed receive delay. The probability of collisions and other performance indicators of the LoRaWAN standard can be evaluated using either numerical simulations or analytical derivations. Numerical simulations have been used in [19, 20] to evaluate the capacity and coverage of the LoRaWAN standard. Moreover, analytical derivations of the throughput in a LoRaWAN network have been conducted in [21, 22]. However, in these two papers, the acknowledgement is not considered. In the present article, we derive a closed-form expression for the probability of collision in a LoRaWAN-like network, in which uplink packets are acknowledged. To the best of the authors' knowledge, the collision probability in a pure ALOHA-based protocol, in which the acknowledgement is sent after a fixed receive delay in the same channel, as in the LoRaWAN standard, has never been analysed in the literature.

In the second part, we will show that the channel selection in LPWANs can be modeled as a MAB problem [23] and that this problem can be solved using simple learning algorithms such as the upper confidence bound algorithm (UCB) [24] or the Thompson sampling (TS)[25]. These algorithms have already been proposed for dynamic spectrum access (DSA) [26, 27] in a cognitive radio (CR) [28] context. In [29, 30], the authors propose to use MAB learning algorithms in a time-slotted IoT network and in [31] these algorithms have been proposed for Wifi networks. In the present article, we introduce MAB learning algorithms for LPWANs and in particular for unslotted LPWANs in unlicensed bands.

The main contributions of this article are summarised as follows:

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking*   (2018) 2018:34

Page 3 of 18

- We first derive closed-form expressions for the probability of a successful transmission into one channel, in an LPWAN featuring a simple acknowledgement, which is similar to the one used by the LoRaWAN standard in Europe.
- Then, we use these probabilities to derive the expression of the latency of AMI communications for different frequency access schemes.
- Finally, we show that the channel selection in an LPWAN can be modeled as a MAB problem and that learning algorithms such as the UCB and TS algorithms can be used by aggregators and provide an efficient channel access scheme and reduce the collision probability and the latency in ALOHA-based LPWANs.

The rest of this paper is organized as follows: the system model is introduced in Section 2. The probability of a successful transmission in a channel is calculated in Section 3. The average latency of the AMI backhaul for different access schemes is analysed in Section 4. The multi-armed bandit theory and various learning algorithms are introduced in Section 5. In Section 6, numerical simulations are used to assess the performance of the UCB algorithm in the proposed LPWAN, and Section 7 concludes this paper.

2 System model

In this article, as illustrated in Fig. 1, we suppose a LoRaWAN-like network composed of one base station which is shared by many end-devices. This base station is used by aggregators for the AMI backhaul. In this network, the available bandwidth is divided into N_c channels

that feature a large number of end-devices, which have different RF capabilities and send packets to a base station. We assume that the number of devices that use each channel is large enough to allow us to consider that a transmission in a channel does not affect the probability that a second transmission occurs. In this case, we can suppose that the uplink traffic in each channel follows a Poisson distribution [11, 32]. We denote λ_T^j the intensity of the uplink traffic in channel j. Furthermore, we suppose that the traffic generated by end-devices is unevenly distributed in channels and does not vary in time.

For the analysis of collisions between packets, we assume that all uplink packets have the same duration denoted T_m. Furthermore, as illustrated in Fig. 2, we assume that a collision occurs in a channel when, at one given moment, at least two packets (uplink or downlink) superpose on each other even partially in time in the channel. Moreover, we suppose that the received power is almost the same for all packets and consequently that there is no capture effect [33].

This hypothesis is valid in a LoRaWAN network, indeed, in this standard, end-devices can use several spreading factors (SF) depending on their path loss [19]. Furthermore, two packets that use different SF are orthogonal [22] and cannot collide. Consequently, if all the devices use the LoRaWAN standard, a packet sent by an end-device only interferes with packets that use the same spreading factor. These packets have consequently the same length and a comparable received power. Please note that in the LoRaWAN standard, only 6 SF are available (the value of the SF is an integer between 7 and 12). As a consequence,
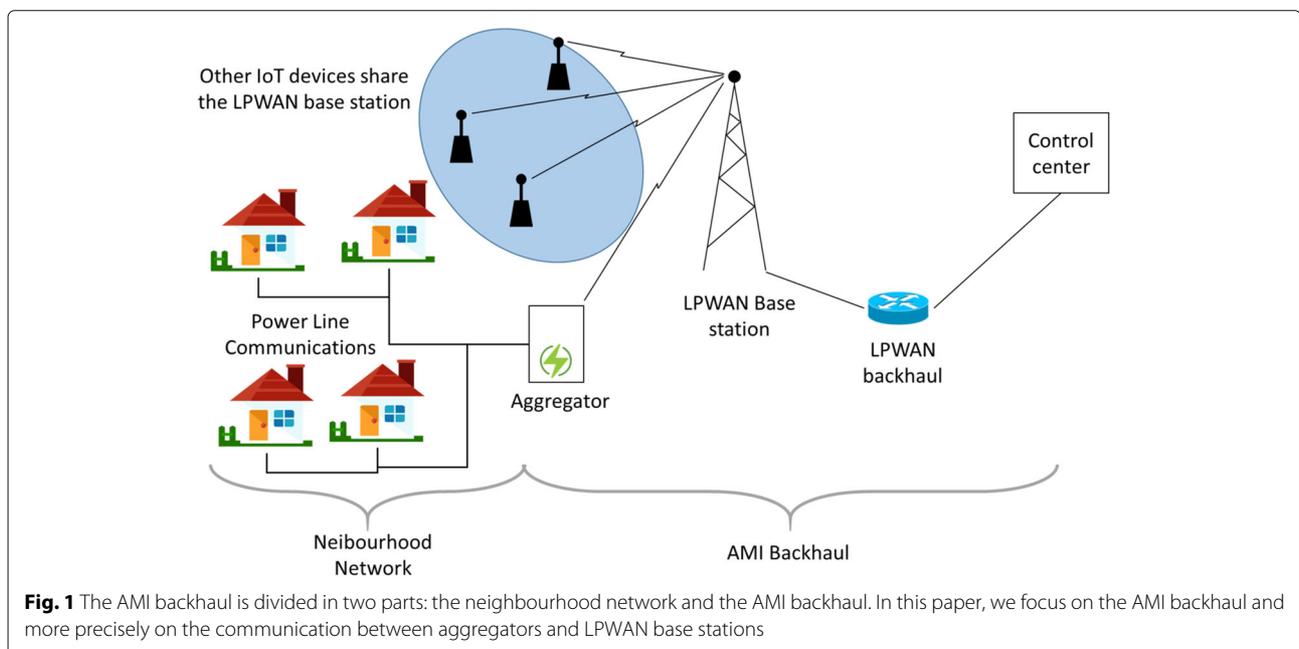

Fig. 1 The AMI backhaul is divided in two parts: the neighbourhood network and the AMI backhaul. In this paper, we focus on the AMI backhaul and more precisely on the communication between aggregators and LPWAN base stations
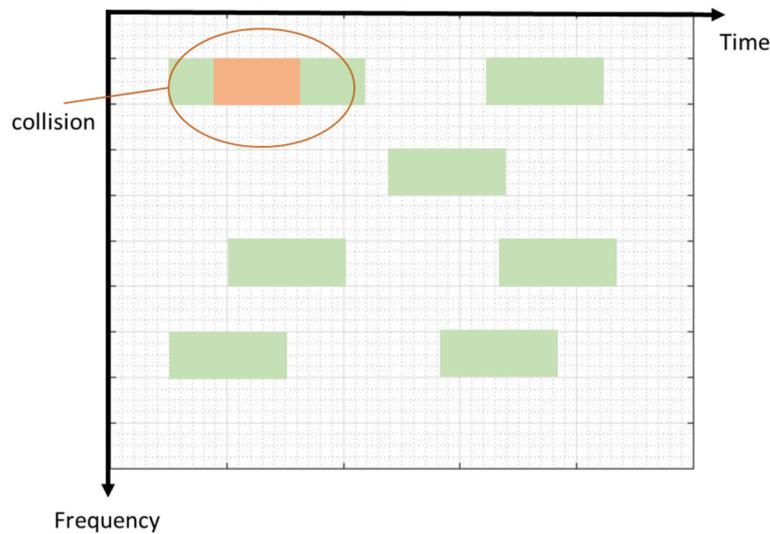
Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking* (2018) 2018:34

Page 4 of 18

Fig. 2 Collision between two packets in the same frequency channel

a large number of end-devices use the same SF. This can cause a large number of collisions in the network.

As we focus on the problem of collisions in LPWAN, the fading of wireless channel is not considered in this article. In unlicensed bands, the interfering traffic of the AMI backhaul (set of packets for which collisions may occur) can be generated by devices that use the same base station, use the same standard but transmit their data to another base station or use another standard. In all this article except Section 6.3, we only consider interferers which use the same standard. In Section 6.3, we extend the use of learning algorithms in case where the interfering traffic is generated by different packet sizes. Besides, as in the LoRaWAN specifications [12], after sending a packet, an end-device waits for an acknowledgement in the channel used for the purpose of uplink communications. As illustrated in Fig. 3, this acknowledgement is sent, by the

base station, after a fixed receive delay denoted T_d. We denote T_a the duration of the acknowledgement. This duration is shorter than the message duration T_m. When a packet does not receive an acknowledgement, the packet is retransmitted. This occurs either if the uplink packet collided with another packet or with an acknowledgement, or if the acknowledgement collided with an uplink packet. Please note that in the LoRaWAN standard case, if the base station cannot send the acknowledgement after the first receive delay, an acknowledgement can be sent after a second receive delay into another channel reserved for downlink communications. This second receive window is not considered in this article.

To resend a packet, as in a LoRaWAN network, a device computes a random retransmission time T_r uniformly distributed during a fixed backoff interval T_{bo}. Then, the RF chain of the device switches to sleep mode and is turned
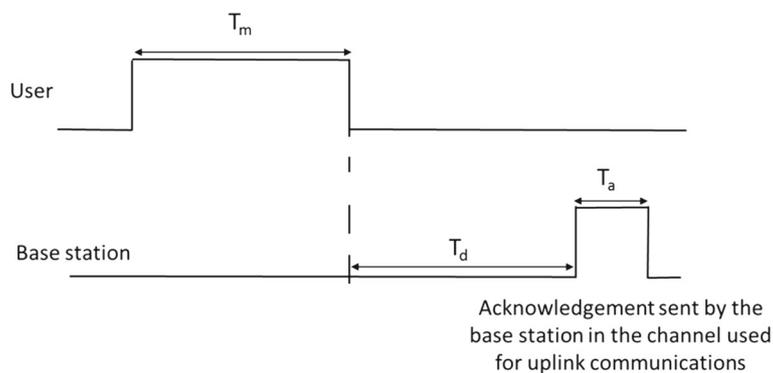

Fig. 3 In this paper, we suppose that the acknowledgement is sent, by the base station, in the channel used for the last uplink communication. This acknowledgement is sent after a time interval denoted T_d

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking* (2018) 2018:34

Page 5 of 18

on for the retransmission. The replica can either be sent in the same channel or into another channel. The selection of this channel is done by the end-device. Each packet is sent no more than M times. If an acknowledgement has not been received after M transmissions, the end-device stops the retransmission process and the packet is lost. Figure 4 shows the operation of an end-device. We suppose that the number of devices in the network is large and that end-devices retransmit their packets after a long back-off interval. In this case, we can consider that the probability of a successful transmission does not depend on the index of retransmission.

When a base station successfully receives a packet, it waits for T_d and, if the channel is free, sends the acknowledgement to the end-device. Since the base station can analyse the presence of a packet in the channel, we can suppose that the base station has a perfect knowledge of the state of the channel (busy or free) and we can thus neglect the sensing time.

3 Probability of successful transmission

In this section, we derive two probabilities, which allow to assess the performance of a LoRaWAN-like LPWAN. The first one is the probability of a successful uplink transmission. It is the probability that a packet sent by an end-device into a channel is received by the base station, i.e. the sent packet did not collide with another packet. This probability is denoted $P(su)$ in this section. The second probability is the probability of a successful

transmission, which is the probability that the end-device receives the acknowledgement. We denote $P(sd)$ this probability. Please note that when a packet is sent by an end-device, it can collide either with an acknowledgement sent by the base station or with an uplink packet sent by another end-device.

In order to compute these two probabilities, we assume that a packet is sent into a channel by an end-device (e.g. by an aggregator), and we denote packet 1 this packet and analyse the probability of a successful transmission. In this section, we make our analysis channel by channel. We denote λ_T the intensity of the uplink traffic in a channel. Moreover, all the events used in this section for the computation of the two probabilities are described in Table 1.

As a first step, a successful downlink transmission happens if the acknowledgement is successfully received after a successful uplink. The following formula makes the link between $P(su)$ and $P(sd)$:

$$P(sd) = P(su)P(sa|su). \qquad (1)$$

Where $P(sa)$ is the probability of having a successful transmission of the acknowledgement. Furthermore, $P(su)$ and $P(sd)$ depend on the value of T_d and T_m. Indeed, these two probabilities do not have the same expression if $T_d \leq T_m$ or where $T_d \geq T_m$.

Moreover, in order to compute the probabilities of a successful transmission, we have to note that the base station sends an acknowledgement only if the channel is free. As a
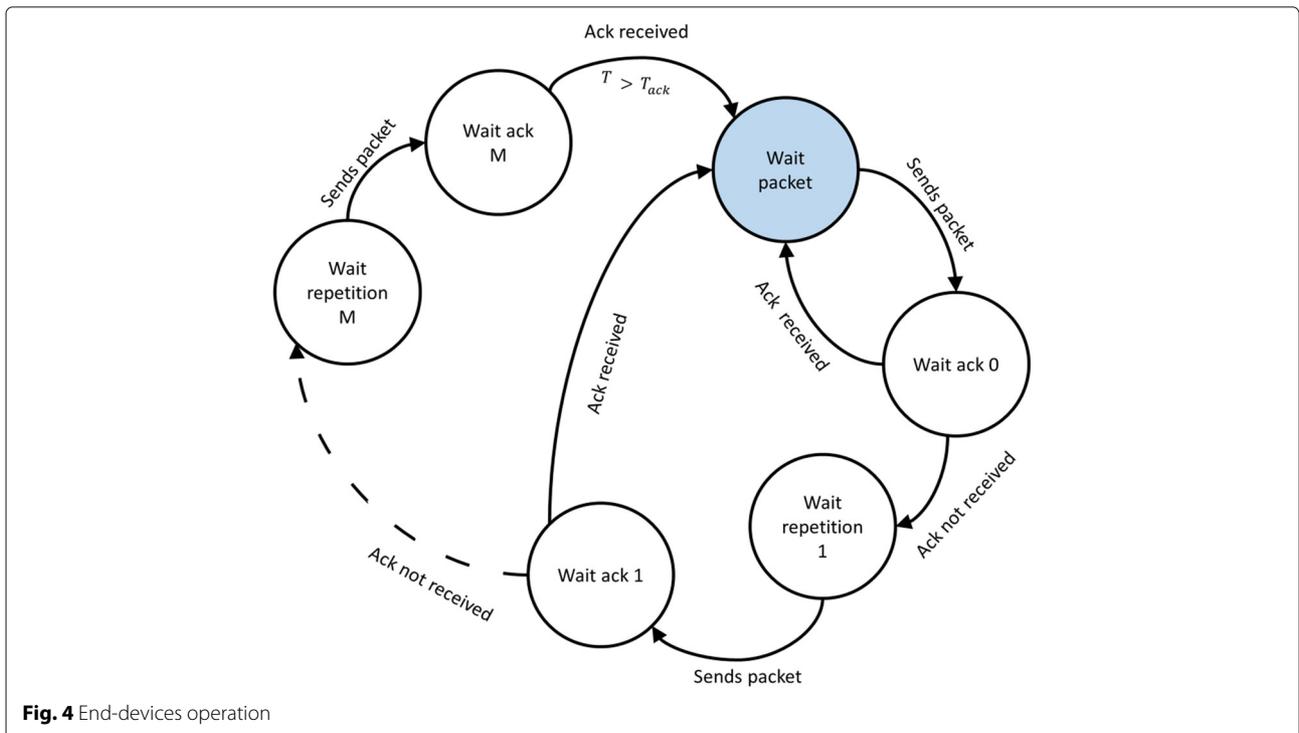

Fig. 4 End-devices operation

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking* (2018) 2018:34

Page 6 of 18

Table 1 Description of the events considered for the computation of the probability of successful transmission

Notation	Event
su	Successful uplink: there is no collision between the uplink packet and another packet. The packet has been received by the base station.
sd	Successful downlink: the end-device receives the acknowledgement
sa	Successful acknowledgement: the downlink transmission is successful, the acknowledgement is sent and does not collide with another packet.
ca	Collision after: the uplink packet has a collision with another uplink packet sent after it.
cb	Colision before: the uplink packet has a collision with an uplink packet sent before it or with an acknowledgement.
cub	Collision uplink before: the uplink packet has a collision with an uplink packet sent before it.
cd	Collision downlink: the uplink packet has a collision with an acknowledgement.
pss	Packet successfully sent: a packet is successfully sent in the interval $[-T_d - T_m - T_a; -T_d - T_m]$.
pb	Packet between: there are packets between the considered packet and its acknowledgement. These packets do not collide with the considered packet or prevent the transmission of the acknowledgement.

consequence, an uplink packet can collide with a downlink packet (acknowledgement) only if the acknowledgement is sent before the packet. This occurs if another packet has been successfully sent in the interval $I_a = [-T_d - T_m - T_a; -T_d - T_m]$ and if the channel is free at the end of the receive delay T_d as illustrated in Fig. 5.

3.1 Case 1: $T_d \leq T_m$

We start by calculating $P(\text{su})$. Which is the probability of having no collision with other packets:

$$P(\text{su}) = P(\overline{\text{cb}} \cap \overline{\text{ca}}) = P(\overline{\text{cb}})P(\overline{\text{ca}}). \qquad (2)$$

Where $P(\text{cb})$ and $P(\text{ca})$ are respectively the probabilities of having a collision with a packet sent before and after

packet 1. As the uplink traffic follows a Poisson process, the events cb and ca are independent.

In order to compute $P(\text{cb})$, we use the law of total probability to decompose it in two terms. The first one is the probability of having a collision with an uplink packet sent before packet 1 and is denoted $P(\text{cub})$. The second one is the probability of having a collision with a downlink packet sent before packet 1 knowing that we do not have a collision with an uplink packet:

$$P(\text{cb}) = P(\text{cb}|\text{cub})P(\text{cub}) + P(\text{cb}|\overline{\text{cub}})P(\overline{\text{cub}})$$
$$= P(\text{cub}) + P(\text{cd}, \overline{\text{cub}}). \qquad (3)$$

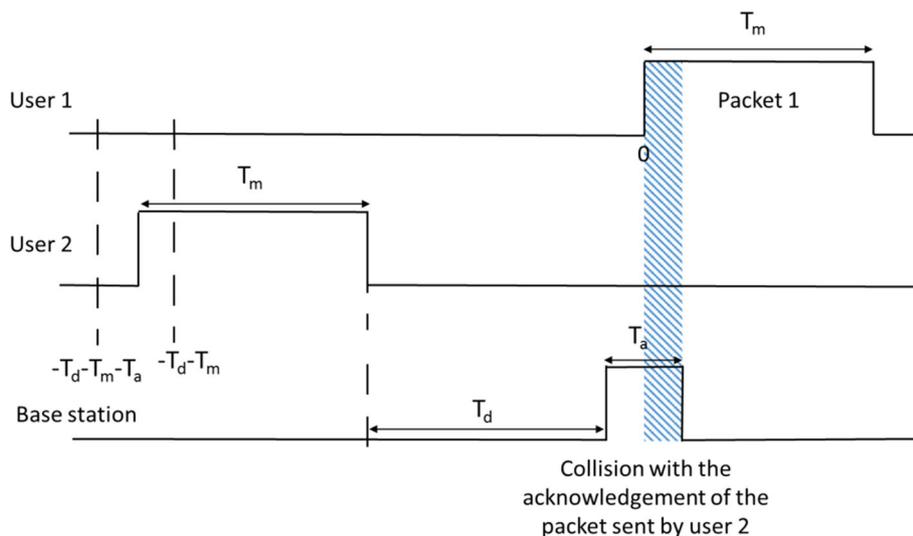Where, $P(\text{cd})$ is the probability of having a collision with an acknowledgement.

Fig. 5 Example of collision between an uplink packet and an acknowledgement sent by the base station

If $T_d \leq T_m$, there exists a collision with a downlink packet, without collision with an uplink packet sent before packet 1 (cd $\cap$ $\overline{\text{cub}}$), if and only if the last packet transmitted before packet 1 is sent in I_a and does not collide with a packet sent before it. Indeed, if there is another packet between packet 1 and a packet sent in I_a, then this packet will either collide with one of the two packets or hinder the transmission of the acknowledgement by the base station. Moreover, the inter-arrival time between two packets follows an exponential distribution with a rate parameter λ_T. This allows to compute $P(\text{cd}, \overline{\text{cub}})$, which is the probability that the inter-arrival time is between $T_m + T_d$ and $T_m + T_d + T_a$ and that the packet sent in I_a does not collide with a packet sent before it:

$$P(\text{cd},\overline{\text{cub}}) = \underbrace{\left(e^{-\lambda_T(T_d+T_m)} - e^{-\lambda_T(T_d+T_m+T_a)} \right)}_{\substack{\text{Probability that the time interval between} \\ \text{two packets is in } [T_m+T_d; T_m+T_d+T_a]}} \times \underbrace{P(\overline{\text{cb}})}_{\substack{\text{Probability that the packet sent} \\ \text{in the interval } I_a \text{ did not collide} \\ \text{with a packet sent before it}}}.$$

$$(4)$$

Furthermore, the probability to have a collision with an uplink packet sent before packet 1 is the probability that at least one packet is sent in the interval $[-T_m; 0]$:

$$P(\text{cub}) = 1 - e^{-\lambda_T T_m}. \tag{5}$$

By replacing, $P(\text{cub})$ and $P(\text{cd}, \overline{\text{cub}})$ by their expressions in (3), we can compute the probability of having no collision with a packet sent before packet 1:

$$P(\overline{\text{cb}}) = 1 - P(\text{cb})$$
$$= \frac{e^{-\lambda_T T_m}}{1 + e^{-\lambda_T(T_d+T_m)} - e^{-\lambda_T(T_d+T_m+T_a)}}. \tag{6}$$

We, now, express the probability of having no collision with a packet sent after packet 1. As illustrated in Fig. 6, packet 1 collides with a packet sent after it, if the interval between its transmission and the transmission of the next packet is shorter than T_m. We can deduce the expression of $P(\overline{\text{ca}})$ from this observation:

$$P(\overline{\text{ca}}) = e^{-\lambda_T T_m}. \tag{7}$$

We can finally compute the probability of having no collision:

Proposition 1 *If $T_d \leq T_m$, the probability of a successful uplink transmission is given by:*

$$P(\text{su}) = P(\overline{\text{cb}})P(\overline{\text{ca}})$$
$$= \frac{e^{-2\lambda_T T_m}}{1 + e^{-\lambda_T(T_d+T_m)} - e^{-\lambda_T(T_d+T_m+T_a)}}. \tag{8}$$

Furthermore, since $T_a < T_m$, if an uplink packet is sent just after the end of packet 1, in the interval $[T_m; T_m + T_d + T_a]$, then either the acknowledgement of packet 1 will not be sent or it will collide with the uplink packet. Consequently, $T_d \leq T_m$, the probability $P(\text{sa}|\text{su})$ that the acknowledgement is received is the probability of having no uplink packet in an interval of length $T_d + T_a$. Consequently,

$$P(\text{sa}|\text{su}) = e^{-\lambda_T(T_d+T_a)}. \tag{9}$$

And $P(\text{sd})$ can be computed with Eq. (1):

Proposition 2 *If $T_d \leq T_m$, the probability of a successful transmission (uplink and downlink) is given by*

$$P(\text{sd}) = P(\text{su})P(\text{sa}|\text{su})$$
$$= \frac{e^{-\lambda_T(2T_m+T_d+T_a)}}{1 + e^{-\lambda_T(T_d+T_m)} - e^{-\lambda_T(T_d+T_m+T_a)}}. \tag{10}$$

We use numerical simulations to verify the proposed formula. We suppose that N_d devices transmit packets into a channel following a Poisson distribution of parameter $\lambda = \frac{10^{-4}}{T_m} s^{-1}$. With this assumption, the intensity of the traffic in the channel is $\lambda_T = N_d \lambda$. As in the LoRaWAN standard, we suppose that $T_d = 1s$, we consider two different values for T_m: $T_m = 1.6$ s and $T_m = 2.8$ s which are the maximum uplink packet length respectively for
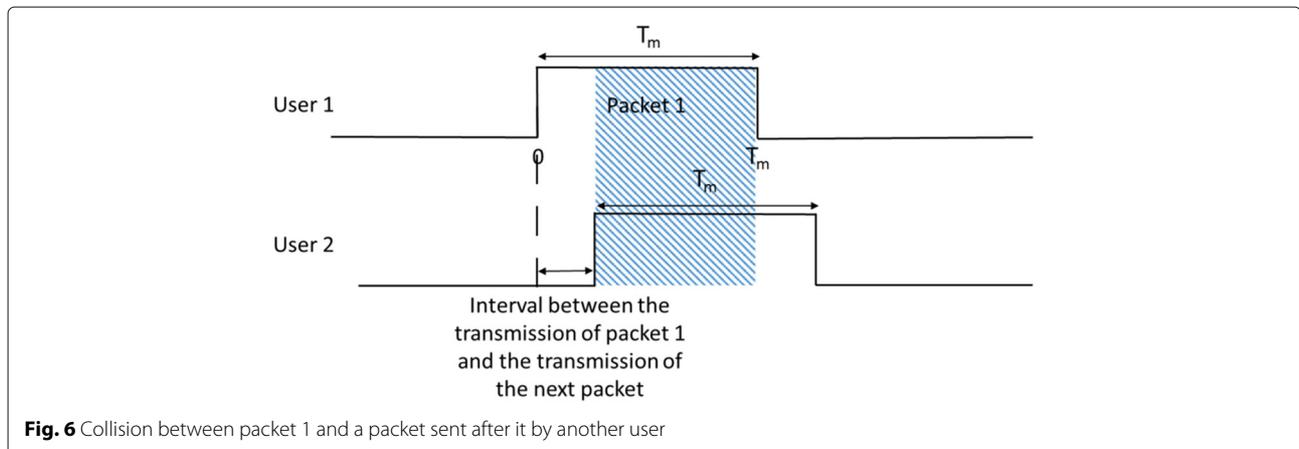

Fig. 6 Collision between packet 1 and a packet sent after it by another user

SF 11 and 12 in the LoRaWAN standard [19]. We display our results for different values of T_a which are compliant with the LoRaWAN standard. Figure 7 shows the evolution of the probability of a successful transmission $P(\text{sd})$ versus $\lambda_T T_m$ (the channel load). As expected, the probability of success decreases as the load increases and the proposed analytical formula and our simulations give the same results.

3.2 Case 2: $T_d \geq T_m$

We also base the computation of $P(\text{su})$ in case where $T_d \geq T_m$ on Eqs. (2) and (3). We begin with the computation of $P(\text{cd}, \overline{\text{cub}})$, the probability of having a collision with a downlink packet without any collision with an uplink packet. The event $\text{cd} \cap \overline{\text{cub}}$ occurs only if a packet has been successfully sent in the interval I_a. In the following, in order to ease the understanding, we denote packet 2 this packet. As illustrated in Fig. 8, we have to consider two incompatible situations for the calculation of $P(\text{cd}, \overline{\text{cub}})$:

- Packet 2 is the last uplink packet sent before packet 1 and does not collide with a packet sent before it (this is the situation studied where $T_d \leq T_m$).
- Packet 2 is successfully sent in I_a, and other uplink packets are transmitted between this packet and its acknowledgement but do not prevent the transmission of the acknowledgement.

In other words, we have to consider two different cases depending on the presence of absence of packets between packet 2 and its acknowledgement. As these two cases are incompatible, we can rewrite the probability $P(\text{cd}, \overline{\text{cub}})$ as the sum of the probabilities of the following two events:

$$P(\text{cd}, \overline{\text{cub}}) = P(\text{cd}, \overline{\text{cub}}, \overline{\text{pb}}) + P(\text{cd}, \overline{\text{cub}}, \text{pb}). \quad (11)$$
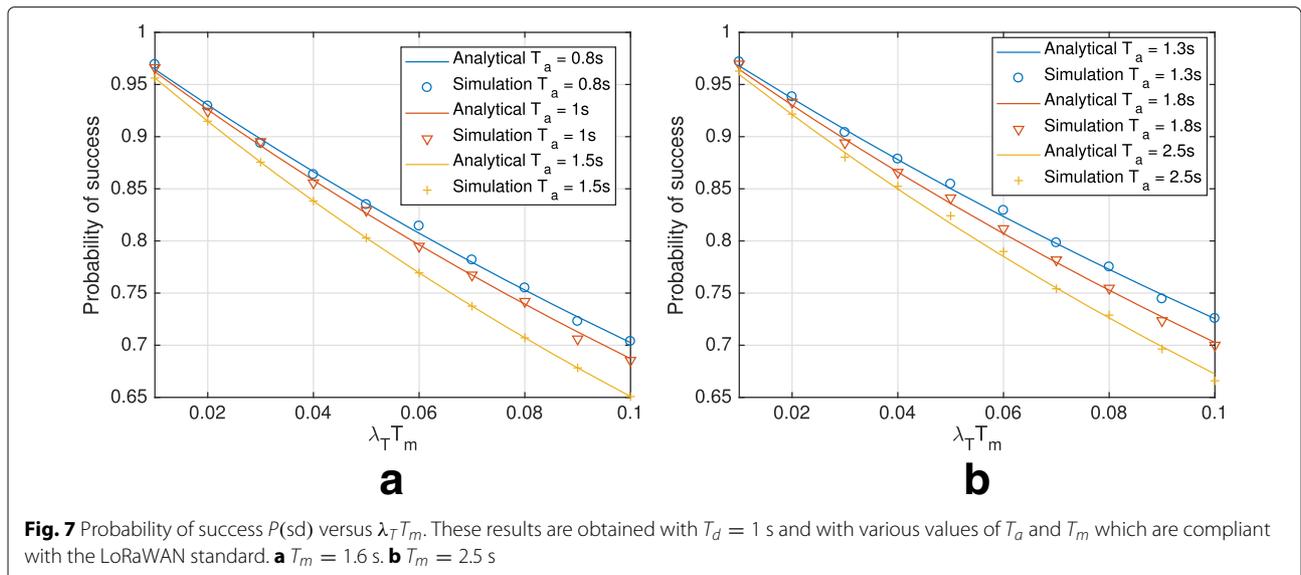
Where $P(\text{pb})$ is the probability to have at least one packet between a given packet (e.g. packet 2) and its acknowledgement. The first term of this expression has been previously computed. If we do not have any packet between packet 2 and its acknowledgement, this packet is the last uplink packet transmitted before packet 1. We are, consequently, in the case previously studied. Strictly speaking, the expression of $P(\text{cd}, \overline{\text{cub}}, \overline{\text{pb}})$ if $T_d \geq T_m$ is equal to the expression of $P(\text{cd}, \overline{\text{cub}})$ if $T_d \leq T_m$. As a consequence, the expression of $P(\text{cd}, \overline{\text{cub}}, \overline{\text{pb}})$ is given in Eq. (4).
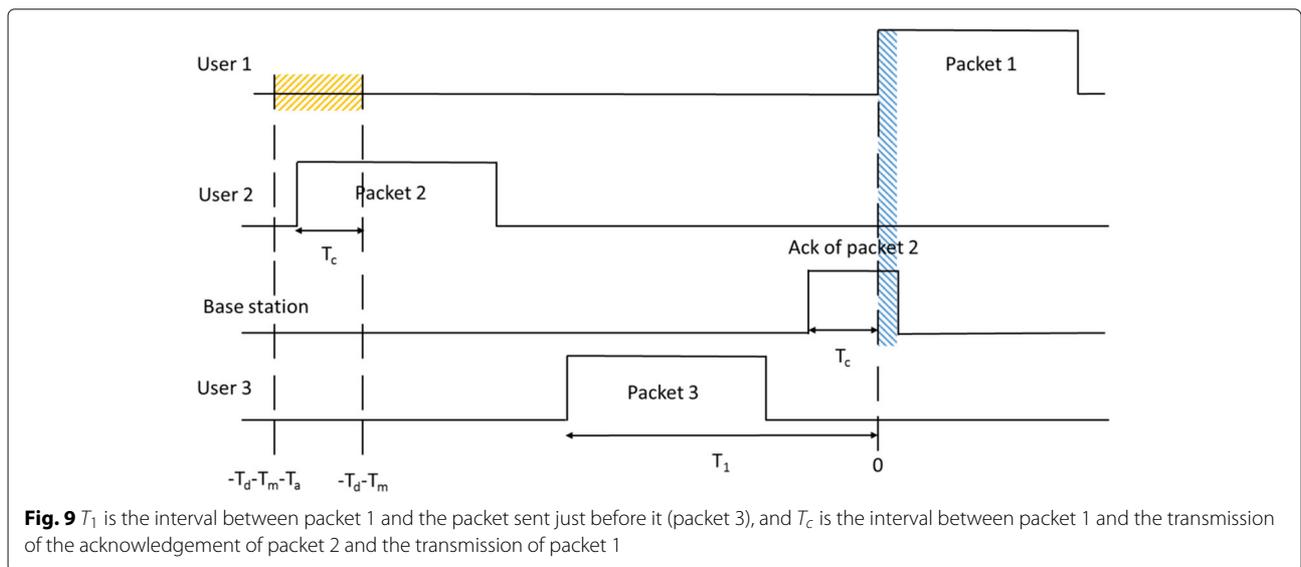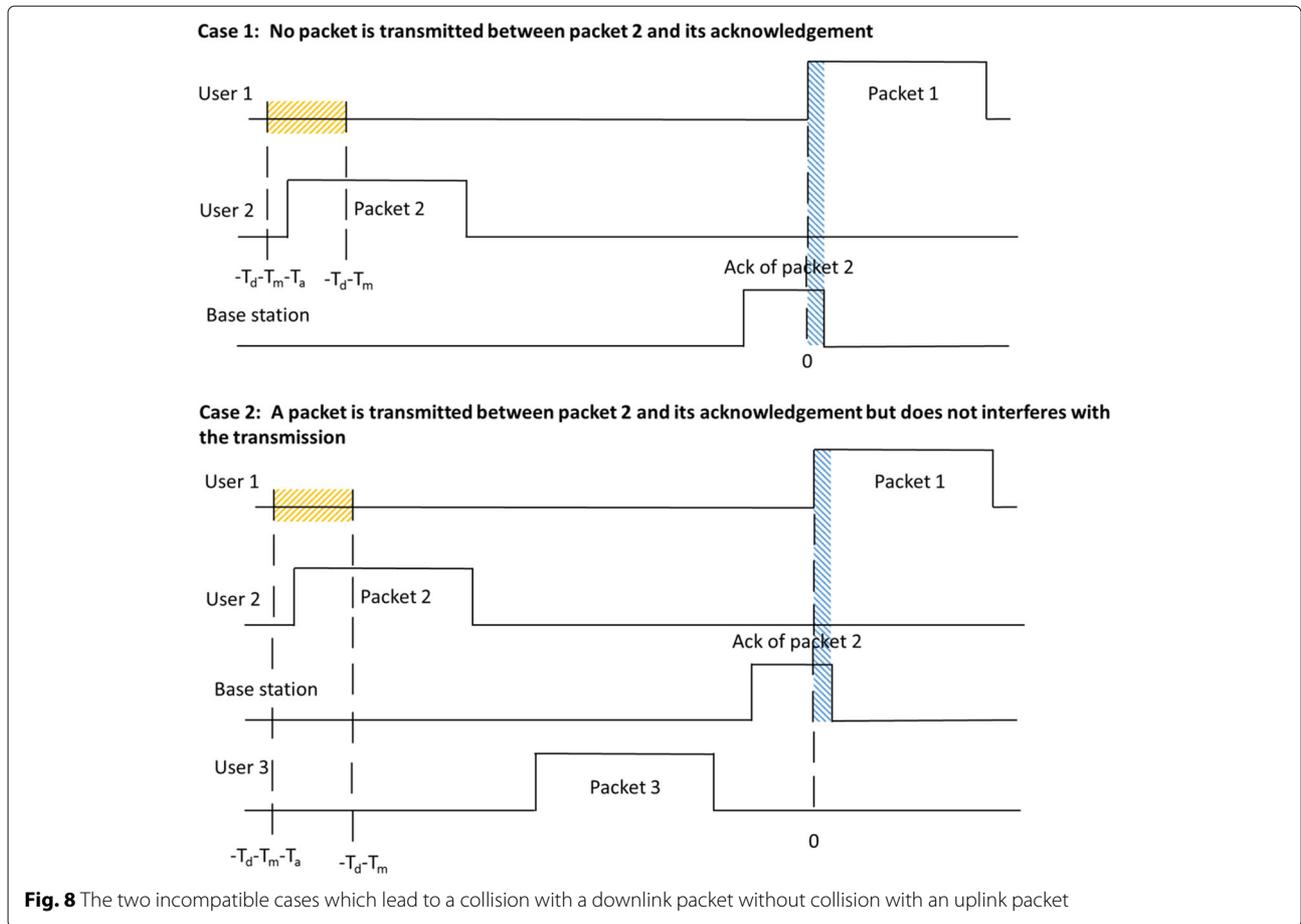
We now consider the second term of Eq. (11). To ease the understanding, in the following, we denote packet 3 the last packet sent before packet 1. Furthermore, the event $\text{cd} \cap \overline{\text{cub}} \cap \text{pb}$ occurs if and only if

- A packet is successfully sent in I_a (packet 2 is successfully sent).
- The last packet transmitted before packet 1 is sent between the packet successfully sent in I_a and its acknowledgement. In other words, packet 3 is sent between packet 2 and its acknowledgement.

These two events are independent. As a consequence, $P(\text{cd}, \overline{\text{cub}}, \text{pb})$ is the product of two probabilities. The first one is the probability that packet 2 is successfully sent in the interval I_a. This probability is denoted $P(\text{pss})$ and can be expressed as

$$P(\text{pss}) = \underbrace{(1 - e^{-\lambda_T T_a})}_{\substack{\text{Proba. that a packet} \\ \text{is sent in } I_a}} \times \underbrace{e^{-\lambda_T T_m}}_{\substack{\text{Proba. of having no} \\ \text{collision with a packet} \\ \text{sent after it}}} \times \underbrace{(1 - P(\text{cb}))}_{\substack{\text{Proba. of having no} \\ \text{collision with a packet} \\ \text{sent before it}}}$$

$$= (e^{-\lambda_T T_m} - e^{-\lambda_T (T_m + T_a)})(1 - P(\text{cb})). \quad (12)$$

Fig. 7 Probability of success $P(\text{sd})$ versus $\lambda_T T_m$. These results are obtained with $T_d = 1$ s and with various values of T_a and T_m which are compliant with the LoRaWAN standard. **a** $T_m = 1.6$ s. **b** $T_m = 2.5$ s

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking* (2018) 2018:34

Page 9 of 18

Fig. 8 The two incompatible cases which lead to a collision with a downlink packet without collision with an uplink packet

Fig. 9 T_1 is the interval between packet 1 and the packet sent just before it (packet 3), and T_c is the interval between packet 1 and the transmission of the acknowledgement of packet 2 and the transmission of packet 1

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking* (2018) 2018:34

Page 10 of 18

In order to compute the second probability, we have to analyse the interval T_1 between packets 1 and 3 and the interval T_c and the acknowledgement of packet 2. As illustrated in Fig. 9, the probability that packet 3 is sent between packet 2 and its acknowledgement is $P(T_c + T_m \leq T_1 \leq T_c + T_d) = P(T_m \leq T_1 - T_c \leq T_d)$.

Since packet 2 has been successfully received, we know that there is only one packet in I_a. As a consequence, T_c follows a uniform distribution on $[0; T_a]$ [34]. Moreover, T_1 follows an exponential distribution. In order to compute the probability density function (pdf) $f_{T_1-T_c}$ of $T_1 - T_c$, we have to compute the convolution of the probability density functions of T_1 and $-T_c$. After some mathematical derivations:

$$f_{T_1-T_c}(\tau) = \begin{cases} 0 & \text{if } \tau < -T_a \\ \frac{1}{T_a}\left(1 - e^{-\lambda_T(\tau+T_a)}\right) & \text{if } \tau \in [-T_a; 0] \\ \frac{1}{T_a}\left(e^{-\lambda_T\tau} - e^{-\lambda_T(\tau+T_a)}\right) & \text{if } 0 < \tau \end{cases}$$

(13)

This allows to conclude that

$$P(T_m \leq T_1 - T_c \leq T_d) = \frac{1}{\lambda_T T_a} \times$$
$$\left(e^{-\lambda_T T_m} - e^{-\lambda_T(T_m+T_a)} - e^{-\lambda_T T_d} + e^{-\lambda_T(T_d+T_a)}\right).$$

(14)

Then, we can calculate $P(\text{cd}], \overline{\text{cub}}, \overline{\text{pb}})$ as

$$P(\text{cd}, \overline{\text{cub}}, \overline{\text{pb}}) = P(\text{pss})P(T_m \leq T_1 - T_c \leq T_d). \quad (15)$$

Moreover, we can rewrite the expression of $P(\text{cb})$, the probability of having a collision with a packet sent before packet 1, thanks to Eqs. (3), (11) and (15):

$$P(\text{cb}) = P(\text{cub}) + P(\text{cd}, \overline{\text{cub}}, \overline{\text{pb}})$$
$$+ P(\text{pss})P(T_m \leq T_1 - T_c \leq T_d).$$

(16)

All the terms of Eq. (16) can be expressed as functions of $P(\text{cb})$, λ_T, T_d, T_m and T_a. We can consequently derive $P(\overline{\text{cb}})$:

$$P(\overline{\text{cb}}) = 1 - P(\text{cb}) = \frac{e^{-\lambda_T T_m}}{1 + f(\lambda_T, T_m, T_d, T_a)}. \quad (17)$$

Where $f(\lambda_T, T_m, T_d, T_a)$ denotes

$$f(\lambda_T, T_m, T_d, T_a) = \left(e^{-\lambda_T T_m} - e^{-\lambda_T(T_m+T_a)}\right)$$
$$\times\left[e^{-\lambda_T T_d} + \frac{1}{\lambda_T T_a}\left(e^{-\lambda_T T_m} - e^{-\lambda_T(T_m+T_a)} - e^{-\lambda_T T_d} + e^{-\lambda_T(T_d+T_a)}\right)\right].$$

(18)

We finally derive $P(\text{su})$ from Eq. (7).

Proposition 3 *If $T_d \geq T_m$, the probability of a successful uplink transmission is given by:*

$$P(\text{su}) = \frac{e^{-2\lambda_T T_m}}{1 + f(\lambda_T, T_m, T_d, T_a)}. \quad (19)$$

Where $f(\lambda_T, T_m, T_d, T_a)$ is defined in Eq. (18).

Moreover, if $T_d \geq T_m$,

$$P(\text{sa}|\text{su}) = e^{-\lambda_T(T_m+T_a)}. \quad (20)$$

Eq. 20 allows to derive the expression of $P(\text{sd})$.

Proposition 4 *For $T_d \geq T_m$, the expression of the probability of successful transmission (uplink and downlink) is given by*

$$P(\text{sd}) = \frac{e^{-\lambda_T(3T_m+T_a)}}{1 + f(\lambda_T, T_m, T_d, T_a)}. \quad (21)$$

Where $f(\lambda_T, T_m, T_d, T_a)$ is defined in Eq. (18).

For numerical simulations, as in the LoRaWAN standard, we set $T_d = 1$ s. We suppose two uplink packet lengths: $T_m = 0.4$ s and $T_m = 0.7$ s, these values respectively correspond to the longest uplink frames for SF 7 and 8. Moreover, we consider different values for T_a which are compliant with the LoRaWAN standard. The evolution of the probability of collision versus the load $\lambda_T T_m$ is displayed in Fig. 10. As expected, the proposed formula fits the numerical simulation.
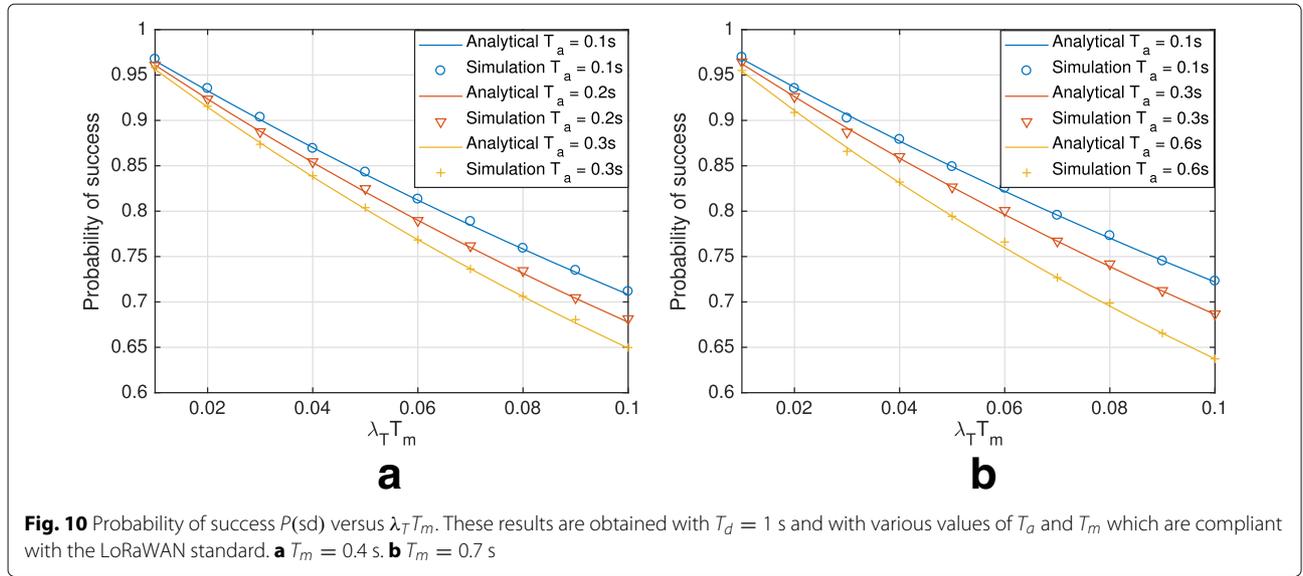
3.3 Analysis of the probability of success

We now analyse the evolution of $P(\text{sd})$ as a function of T_d. An analysis of the sign of the derivative of Eqs. (21) and (10) shows that $P(\text{sd})$ decreases if $T_d \leq T_m$ and increases if $T_d \geq T_m$. The evolution of $P(\text{sd})$ versus T_d is displayed in Fig. 11 for different values of T_m and T_a which are compliant with the LoRaWAN standard. In each pair (T_m, T_a), T_m is the maximum uplink packet length for the corresponding SF and T_a is compliant with the standard [19]. As expected, we can see in this figure that the longer is T_m, the lower is the probability of success. The probability of a successful transmission decreases over $[0; T_m]$ and, if T_d is longer than T_m, $P(\text{sd})$ is almost constant and only slightly increases with T_d. As a consequence, if $T_d \geq T_m$, the probabilities $P(\text{sd})$ and $P(\text{su})$ can be approximated by their values on $T_d = T_m$ and $T_d \to +\infty$.

Proposition 5 *For $T_d \geq T_m$, the expression of the probability of a successful transmission can be approximated by*

$$P(\text{sd}) \approx \frac{e^{-\lambda_T(3T_m+T_a)}}{1 + e^{-2\lambda_T T_m} - e^{-\lambda_T(2T_m+T_a)}}$$
$$\approx \frac{e^{-\lambda_T(3T_m+T_a)}}{1 + \frac{\left(e^{-\lambda_T T_m} - e^{-\lambda_T(T_m+T_a)}\right)^2}{\lambda_T T_a}}. \quad (22)$$

Proof For this proof, we denote respectively $P_{T_m}(\text{sd})$ and $P_\infty(\text{sd})$ the first and the second proposed approximations.

Fig. 10 Probability of success $P(\mathrm{sd})$ versus $\lambda_T T_m$. These results are obtained with $T_d = 1$ s and with various values of T_a and T_m which are compliant with the LoRaWAN standard. **a** $T_m = 0.4$ s. **b** $T_m = 0.7$ s

First of all,

$$\frac{e^{-\lambda_T T_m} - e^{-\lambda_T (T_m + T_a)}}{\lambda_T T_a} = e^{-\lambda_T T_m} \frac{\left(1 - e^{-\lambda_T T_a}\right)}{\lambda_T T_a}. \quad (23)$$

Moreover, in the studied network, we can assume that $\lambda_T T_a << 1$. As a consequence,

$$1 - e^{-\lambda_T T_a} \approx \lambda_T T_a. \quad (24)$$

And therefore,

$$\frac{e^{-\lambda_T T_m} - e^{-\lambda_T (T_m + T_a)}}{\lambda_T T_a} \approx e^{-\lambda_T T_m}. \quad (25)$$

Eq. 25 allows us to prove that $P_{T_m}(\mathrm{sd}) \approx P_\infty(\mathrm{sd})$. Furthermore, $P(\mathrm{sd})$ is an increasing function of T_d over

$[T_m; +\infty]$. As a consequence,

$$P_{T_m}(\mathrm{sd}) \leq P(\mathrm{sd}) \leq P_\infty(\mathrm{sd}). \quad (26)$$

Which proves that $P(\mathrm{sd}) \approx P_{T_m}(\mathrm{sd}) \approx P_\infty(\mathrm{sd})$. This finally proves proposition 5.

□

We have computed the expression of the probability of a successful transmission in a LoRaWAN-like LPWAN. In the following, we analyse the latency of AMI communications in this network for different access schemes as a function of the probability of successful transmission $P(\mathrm{su})$.

4   Latency in an LPWAN
We now consider an aggregator that wants to send a packet to a LPWAN base station. In order to send this packet, this aggregator can use one of the N_c available channels. In each channel, the uplink communication can either be successful or the transmitted packet can collide with the interfering traffic. The probability of having a successful uplink transmission in channel j is denoted $P^j(\mathrm{su})$. As detailed in the previous section, this probability depends on λ_T^j the intensity of the traffic in the channel. In this section, we analyse the latency of the communications of the AMI backhaul as being a function of $P^j(\mathrm{su})$ $\forall j \in [\![1; N_c]\!]$ for the two following different frequency access schemes:

1. The aggregator randomly selects the channel for each transmission.
2. The aggregator uses the channel with the highest probability of successful transmission for all its transmissions. Please note that this policy requires

Fig. 11 Probability of successful transmission for $\lambda_T = 0.2$ s^{-1} and for different values of T_m and T_a
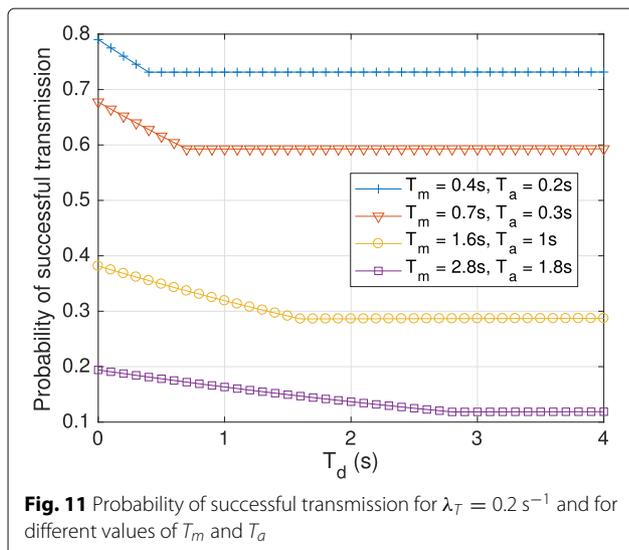
the aggregator to have perfect knowledge of the probability of success in the channels. We present some learning algorithms which allow to acquire this knowledge in Section 5.

4.1 Case 1: random channel selection

The expected latency $\mathbb{E}[\mathcal{L}]$ is defined as the mean time between the first transmission of a packet and the first reception of the packet by the base station. According to the law of total expectation, the average latency is

$$\mathbb{E}[\mathcal{L}] = \sum_{i=1}^{M} P(N_{\text{ret}} = i)\mathbb{E}[\mathcal{L}|N_{\text{ret}} = i]. \tag{27}$$

Where N_{ret} is the number of retransmissions. Please note that the expression of $\mathbb{E}[\mathcal{L}|N_{\text{ret}} = i]$ does not depend on the frequency access scheme. Actually, in Eq. (27), only $P(N_{\text{ret}} = i)$ is dependent on the access scheme. Moreover, given the specific studied acknowledgement mechanism, the expected latency for N_{ret} retransmissions is

$$\mathbb{E}[\mathcal{L}|N_{\text{ret}} = i] = (i-1)\left(T_m + T_d + T_s + \mathbb{E}[T_r]\right) + T_m \tag{28}$$

$$= (i-1)\left(T_m + T_s + T_d + \frac{T_{\text{bo}}}{2}\right) + T_m. \tag{29}$$

Where T_s is the time during which the end-device senses the channel so as to detect the preamble of the acknowledgement. This time is short in the LoRaWAN standard. Please note that, after a failed transmission, the acknowledgement is not transmitted by the base station. In that case, in the LoRaWAN standard, the device does not wait for the acknowledgement during T_a but during T_s, a shorter time which is long enough to detect the presence or absence of acknowledgement in the channel [14]. In the following, we will denote $T_l = T_m + T_d + T_s \approx T_m + T_d$.

We now have to compute the expression of $P(N_{\text{ret}} = i)$ which can be expressed as

$$P(N_{\text{ret}} = i) = P(\text{su trans. i})\prod_{k=1}^{i-1}\left(1 - P(\text{su trans. k})\right). \tag{30}$$

Where $P(\text{su trans. i})$ is the probability of having a successful i-th transmission of the packet. As the probability of success is the same for all retransmissions, the expression of $P(\text{su trans. k})$ is:

$$P(\text{su trans. k}) = \frac{1}{N_c}\sum_{l=1}^{N_c} P^l(\text{su}) = P_m(\text{su}) \tag{31}$$

Where $P_m(\text{su})$ is the average probability of a successful transmission in the network. We finally derive the expression of the average latency:

$$\mathbb{E}[\mathcal{L}] = P_m(\text{su})T_m\sum_{i=1}^{M}(1 - P_m(\text{su}))^{i-1}$$
$$+ P_m(\text{su})\left(T_l + \frac{T_{\text{bo}}}{2}\right)\sum_{i=2}^{M}(i-1)(1 - P_m(\text{su}))^{i-1}. \tag{32}$$

We now employ the expression of the derivative of the geometric series so as to obtain the expression of the latency for an infinite number of repetitions:

$$\mathbb{E}[\mathcal{L}] \xrightarrow[M\to\infty]{} \left(T_l + \frac{T_{\text{bo}}}{2}\right)\frac{1 - P_m(\text{su})}{P_m(\text{su})} + T_m. \tag{33}$$

4.2 Case 2: best channel selection

In this section, we denote $P^{j*}(\text{su})$ the probability of having a successful transmission in the best channel. In case, where the aggregator uses the least loaded channel for all its transmission, $P(\text{su trans.k}) = P^{j*}(\text{su})$ and we can derive the expression of the latency with this access scheme:

$$\mathbb{E}[\mathcal{L}] = P^{j*}(\text{su})T_m\sum_{i=1}^{M}(1 - P^{j*}(\text{su}))^{i-1}$$
$$+ P^{j*}(\text{su})\left(T_l + \frac{T_{\text{bo}}}{2}\right)\sum_{i=2}^{M}(i-1)(1 - P^{j*}(\text{su}))^{i-1}. \tag{34}$$

As for the random channel selection, we use the derivative of a geometric series to get the expression of $\mathbb{E}[\mathcal{L}]$ for an infinite number of retransmissions:

$$\mathbb{E}[\mathcal{L}] \xrightarrow[M\to\infty]{} \left(T_l + \frac{T_{\text{bo}}}{2}\right)\frac{1 - P^{j*}(\text{su})}{P^{j*}(\text{su})} + T_m. \tag{35}$$

4.3 Comparison of the two strategies

When comparing Eqs. (33) and (35), we can see that the latency always decreases as the best channel is chosen for the first transmission. Furthermore, if we compute the difference between the latency of Eqs. (33) and (35):

$$\mathbb{E}[\mathcal{L}]_{\text{rand}} - \mathbb{E}[\mathcal{L}]_{\text{BC}} = \left(T_l + \frac{T_{\text{bo}}}{2}\right)\left(\frac{1}{P_m(\text{su})} - \frac{1}{P^{j*}(\text{su})}\right) \tag{36}$$

Where $\mathbb{E}[\mathcal{L}]_{\text{rand}}$ is the expected latency with a random channel selection and $\mathbb{E}[\mathcal{L}]_{\text{BC}}$ is the expected latency with a best channel selection. Eq. 36 shows that the gain in latency provided by the selection of the best channel, only depends on the difference between the inverse of the average probability of a successful transmission in the random channel selection case and the inverse of this probability in the best channel case.

The selection of the best channel requires the knowledge of the probability of collision in the channels. In

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking* (2018) 2018:34

Page 13 of 18

the following, we introduce two reinforcement learning algorithms to acquire this knowledge.

5 Reinforcement learning algorithms in LPWAN

5.1 MAB learning

The equations derived in the previous section show that the selection of the best channel can significantly reduce the latency of AMI communications when the traffic is unevenly distributed in the channels. This can occur either if some devices use another LPWAN or base station or if all the devices do not use the same set of channels. In this section, we will show that the channel selection can be viewed as a multi-armed bandit (MAB) problem [23], which can be solved thanks to simple reinforcement learning algorithms. This modelling has already been used in dynamic spectrum access (DSA) [26, 27]. In such a scenario, spectrum sensing is used as a feedback for channel selection. However, spectrum sensing has a poor performance in LPWANs [6]. That is why we use the acknowledgement as a reward for learning. With this acknowledgement, machine learning algorithms can be used by end-devices for the purpose of channel selection.

Please note that, with the proposed MAB learning algorithms, each end-device optimises its own energy consumption without exchanging information with other end-devices. This solution is, consequently, a non-coordinated solution. One of the main advantages of such a solution is its energy consumption. Indeed, the algorithms proposed here have a low complexity. They consume, consequently, few energy. This energy is negligible compared to the energy that would be consumed to exchange information between end-devices.

If we now consider the problem as a MAB problem, each channel is viewed as a gambling machine (bandit). All bandits lead to the same reward (a successful transmission) but with different probabilities. Indeed, $P^j(\text{su})$ and $P^j(\text{sd})$ change from one channel to another. We denote t the number of transmissions realised by the aggregator, where $T_j(t)$ denotes the number of selections of channel j.

In order to select the best channel, which features the highest probability of a successful transmission, aggregators have to learn about the quality of the channels. This learning is based on the reward obtained after the previous transmissions. We define the reward of the data transmission in channel j at time t as

$$r_t(j) = \begin{cases} 1 & \text{if the transmission is successful,} \\ 0 & \text{else.} \end{cases} \quad (37)$$

In LPWAN, the reward can be provided by the acknowledgement, and an end-device considers that the reward is 1 if the acknowledgement is received, and 0 otherwise. With this solution, the proposed algorithms do not require any extra signalling. In the studied problem, an aggregator that uses a reinforcement learning algorithm

begins without any information about the probabilities of successful transmission in the N_c channels. The device first explores all the channels and uses the reward to learn about the channels' probability of successful transmission. On the basis of the acquired knowledge, the device uses more and more the channels that provided the highest reward. It improves consequently its probability of having a successful transmission. After several transmissions, the end-device has enough knowledge to send almost all its packets into the channel featuring the highest probability of successful transmission and consequently the lowest latency.

Furthermore, two types of reinforcement learning algorithms have been proposed to solve MAB problems: frequentist algorithms where the channel is deterministically chosen on the basis of past experience, and Bayesian algorithms where the decision is drawn from a prior distribution [35]. In this paper, with no loss of generality, we analyse the performance of two algorithms, the upper confidence bound (UCB) algorithm [26] which is frequentist and the Thompson sampling (TS) algorithms [25] which is Bayesian. The main advantages of these two algorithms are their low computational complexity and their low memory requirements, which allow them to be implemented in any end-device and in particular in aggregators.

5.2 UCB$_1$ algorithm

The UCB$_1$ algorithm is proven to be asymptotically order optimal where the interfering traffic generated by other end-devices follows a Bernoulli distribution [24]. Moreover, it requires little processing resources and memory. In the UCB algorithm case, we use the sample mean of the reward to assess the probability of collision in channel j:

$$\overline{X}_j(t) = \frac{1}{T_j(t)} \sum_{l=0}^{t-1} r_l(j) \mathbb{1}(a_l = j). \quad (38)$$

Where $\mathbb{1}(a_l = j)$ denotes the indicator function. This function is equal to 1 if the device made its l-th transmission in channel j and 0 elsewhere. We define the upper confidence bound algorithm indexes in each channel as [24]

$$B_j(t) = \overline{X}_j(t) + A_j(t). \quad (39)$$

Where $A_j(t)$ is an upper confidence bias. In the UCB algorithm case, the selected channel features the highest upper confidence bound:

$$a_t = \underset{j}{\text{argmax}}(B_j(t)). \quad (40)$$

The bias of the UCB$_1$ algorithm is [24]

$$A_j(t) = \sqrt{\frac{\alpha \ln t}{T_j(t)}}. \quad (41)$$

In Eq. (41), α is the exploration coefficient. The UCB$_1$ is proven to be order optimal for $\alpha > 0.5$ [24] and has good performance for lower values of $\alpha > 0$ [36]. The larger this coefficient is, the longer the exploration is. During the initial transmissions, the empirical mean is low compared to the bias and the aggregator explores all the channels. Progressively, the value of the bias decreases and the empirical mean becomes predominant. With this algorithm, the aggregator learns at each transmission. Once it has learned enough, it starts mostly using a single channel, the one that guarantees the higher empirical mean for the reward. Consequently, in the UCB$_1$ algorithm case, and after exploration, the latency of AMI communications will be equal to the one studied in Section 4.2.

In the UCB$_1$ algorithm, the computation of indexes is deterministic. It is, consequently, a frequentist algorithm. In the following section, we introduce the Thompson sampling algorithm which is a Bayesian algorithm. With this algorithm, the indexes are sampled from a random distribution.

5.3 Thompson sampling

In the case of the Thompson sampling algorithm [25], the channel index is computed thanks to a beta distribution whose parameters depend on prior experience. In the following, we denote:

$$S_j(t) = \sum_{l=0}^{t-1} r_l(j)\mathbb{1}_{\{a_l=j\}}, \tag{42}$$

the sum of the reward in channel j at instant t, and

$$F_j(t) = T_j(t) - S_j(t). \tag{43}$$

The number of unsuccessful transmissions in channel j. For each of its transmissions, the channel index in channel j at a given time t is sampled from the beta distribution:

$$B_j(t) \sim \beta(1 + S_j(t), 1 + F_j(t)). \tag{44}$$

As for the UCB$_1$ the channel featuring the higher index is chosen for the t-th transmission. With this algorithm, at the beginning, all the indexes are uniformly distributed in $[0;1]$ (i.e. flat prior $\beta(1,1)$). When the algorithm learns about channel j, the distribution becomes squeezed and centered around $P_j(\text{sd})$. As for UCB$_1$, after a sufficient learning period, when the distributions are squeezed and the expectations have been well estimated, the end-device will use the most vacant channel for most of its transmissions. In order to better understand the behaviour of the algorithm, we compute the expectation of the index $B_j(t)$:

$$\mathbb{E}\{B_j(t)\} = \frac{1 + S_j(t)}{2 + T_j(t)} \underset{T_j(t) \to \infty}{\sim} \frac{S_j(t)}{T_j(t)} = \overline{X}_j(t). \tag{45}$$

$$V\{B_j(t)\} = \frac{(1 + S_j(t))(1 + F_j(t))}{(2 + T_j(t))^2(3 + T_j(t))} \tag{46}$$

We can see in Eq. (46) that the higher $T_j(t)$ is, the lower the variance of the distribution of $B_j(t)$. Furthermore, as shown in Eq. (45), the expectation of the index $B_j(t)$ tends towards $P_j(\text{sd})$ when $T_j(t)$ tends to infinity.

Please note that, for each transmission, the TS algorithm only requires to compute N_c values from beta distributions.

6 Numerical evaluation of MAB learning in LPWANs

In this section, we use numerical simulations to assess the performance of the MAB-learning algorithms, introduced in the previous section, in an pure ALOHA-based LPWAN.

6.1 Simulation scenario

For simulations, we consider an LPWAN comprising $N_c = 10$ channels. All the devices in the network use the same SF and transmit an uplink packet during $T_m = 0.7$ s (this corresponds to SF 8 in a LoRaWAN network [19]). Moreover, we suppose that $T_d = 1$ s and $T_a = 0.1$ s. We suppose that T_s is short enough to be neglected. When a device does not receive an acknowledgment, it selects a random time T_r between 0 and $T_{\text{bo}} = 10$ s. Then, it waits for T_r and resends the packet. The maximum number of repetitions is equal to 5 in all this section.

In order to generate the interfering traffic, we consider a set of non-intelligent devices that use the network. Each of these devices (e.g. temperature sensors, humidity sensors or smart appliances) uses only one channel. The traffic generated by these non-intelligent devices is an interfering traffic for the AMI backhaul. In this article, we suppose that interfering end-devices and aggregators use the same standard; however, similar performance can be obtained when the interfering traffic is generated by devices using different standards. Each of these devices sends a packet following a Poisson distribution. The intensity of the Poisson process verifies $\lambda_s T_m = 10^{-4}$ for all non-intelligent devices. This intensity does not take into account the traffic generated by retransmissions. With this intensity, each device sends approximately one packet every 2 h.

We suppose that there are 1000 non-intelligent end-devices in the first channel, 900 in the second one, 800 in the third one, and so on until 100 in the tenth channel. We simulate the network made of non-intelligent devices so as to estimate the probabilities of a successful transmission in each channel. With this distribution of non-intelligent devices, these probabilities are equal to (0.45, 0.53, 0.57, 0.64, 0.70, 0.77, 0.82, 0.87, 0.92, 0.96).

We suppose that 50 aggregators that have learning capabilities begin to use the LPWAN. These aggregators have the same characteristics than those of other devices, but can use channel selection algorithms. We suppose that each aggregator transmits its packets following a Poisson

process whose intensity verifies $\lambda_a T_m = 4 \times 10^{-4}$ (on average an aggregator sends a packet every 30 min). We simulate the network during 14 days, and we analyse the evolution of the probability of a successful transmission $P(\text{sd})$ and that of the mean latency.

6.2   Simulation results in a LoRaWAN network

In the studied network, we evaluate the performance of several learning algorithms, we consider that either UCB_1 or Thompson sampling algorithms are implemented in aggregators.

We first analyse the number of transmissions in each channel after 14 days of learning with the UCB_1 algorithm. On average, during these 14 days, each aggregator transmits 672 times. As we ranked channels by vacancy rate probability, with no loss of generality, we can see in Fig. 12 that aggregators mostly transmit in channels with the lowest probability of collision. Moreover, aggregators transmit more than 25% of their packets in the less loaded channel and around 20% in the second one. Furthermore, after 14 days, less than 20% of the packets transmitted by aggregators are transmitted in the five most loaded channels.
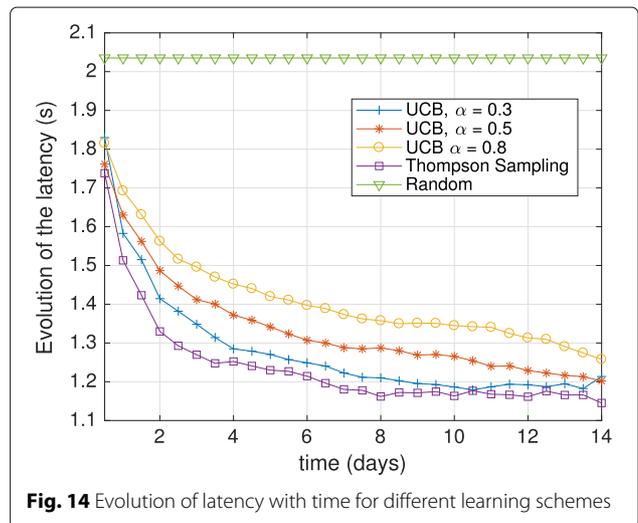
We now analyse the evolution of the probability of successful transmissions $P(\text{sd})$ for aggregators featuring intelligent capabilities. We then compare the results obtained in this case with those of a scenario in which aggregators randomly select the channel for each of their transmissions. This random selection is currently employed in the LoRaWAN standard. The results are displayed in Fig. 13, as for the probability of successful transmissions and Fig. 14 as for the evolution of the latency. At the beginning, aggregators explore all the channels. The probability of a successful transmission and the latency of

Fig. 13 Evolution of the number of successful transmissions with time for different learning schemes
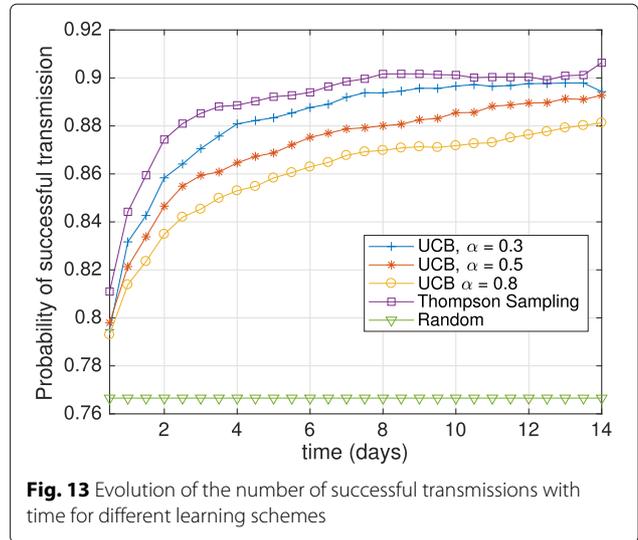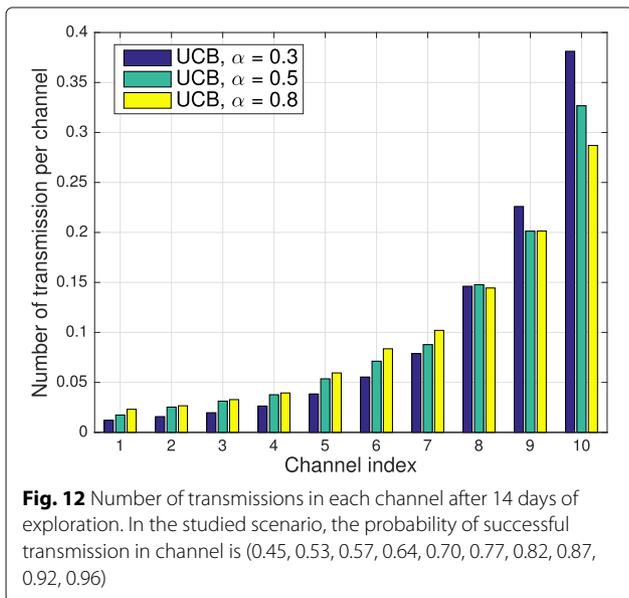
AMI communications featuring learning algorithms are only slightly better than those experienced where using a random allocation. However, after some transmissions, aggregators learn about the occupancy in channels and the probability of successful transmission increases. This probability is of 76.5% for a random allocation and reaches 90% after a few days of exploration. This represents an increase of 14% in the probability of successful transmission (uplink and downlink).

An increase in the probability of successful transmission is beneficial for the latency of AMI communications. As seen in Fig. 14, learning algorithms reduce by 0.8 *s* the latency of aggregators' communications. This represent a 40% gain compared to the random channel selection.

We now compare the performance of the studied learning algorithms. We can see in Fig. 14 that the Thompson sampling algorithm reduces latency more quickly. This

Fig. 12 Number of transmissions in each channel after 14 days of exploration. In the studied scenario, the probability of successful transmission in channel is (0.45, 0.53, 0.57, 0.64, 0.70, 0.77, 0.82, 0.87, 0.92, 0.96)

Fig. 14 Evolution of latency with time for different learning schemes

result is in line with the theoretical studies. Indeed, the Thompson sampling has been proven to converge more quickly than the UCB algorithm in case where the interfering traffic follows a Bernoulli process [35]. However, the computation of the TS indexes requires a little bit more computation than the UCB ones. It is important to note that, in the present article, the interfering traffic is generated by both the static interfering traffic and the traffic generated by other aggregators. The static interfering traffic follows a Bernoulli process. However, other aggregators also use learning algorithms and the traffic they generate is not stochastic [30]. In the simulated scenarios, the traffic generated by other aggregators is small compared to the traffic generated by static devices. The interfering traffic can, consequently, be approximated by a Bernoulli process.

Furthermore, the TS and the UCB$_1$ algorithm with $\alpha = 0.3$ provide similar results after 14 days of exploration. For such low value of α (i.e. below $\alpha = 0.5$), we do not have any theoretical proof of convergence. However, the algorithm has good performance in our simulation scenarios. On the basis of the comparison of the performances of the UCB$_1$ algorithm for different values of α, we can see that the reduction of the latency is faster with a small α (e.g. $\alpha = 0.3$). Figure 14 shows that, in the proposed scenario, the reduction of the latency is increasingly slowly as the α coefficient increases. The analysis of the α coefficient is done here empirically. A comprehensive empirical study of the impact of the α coefficient in the MAB problem has been conducted in [37].

6.3 Extension to different packet sizes

In the previous section, we analysed the performance of MAB learning algorithms in a network in which all devices use the same standard, and in particular the same SF in a LoRaWAN network. In this section, we confirm the ability of MAB algorithms to reduce the latency of communications and we highlight the ability of the proposed algorithms to cope with different packet sizes.

For that purpose, we consider that the 50 aggregators previously introduced communicate with the same LoRaWAN base station. In this section, the interfering traffic is generated by end-devices which transmit packets of different sizes. Each of these static end-devices transmits following a Poisson process with, on average, one packet every 2 h. Moreover, the packet size is a multiple of 100 ms uniformly distributed between 0.1 and 2 s. The packets transmitted by static devices are neither acknowledged nor retransmitted. We suppose that the number of devices in each channel is the following: [750, 1000, 650, 600, 450, 300, 500, 700, 850, 1050]. With this distribution of static end-devices, we have the following probability of a successful transmission in channels: (0.59, 0.51, 0.64, 0.65, 0.74, 0.78, 0.72, 0.59, 0.54, 0.50). In this second
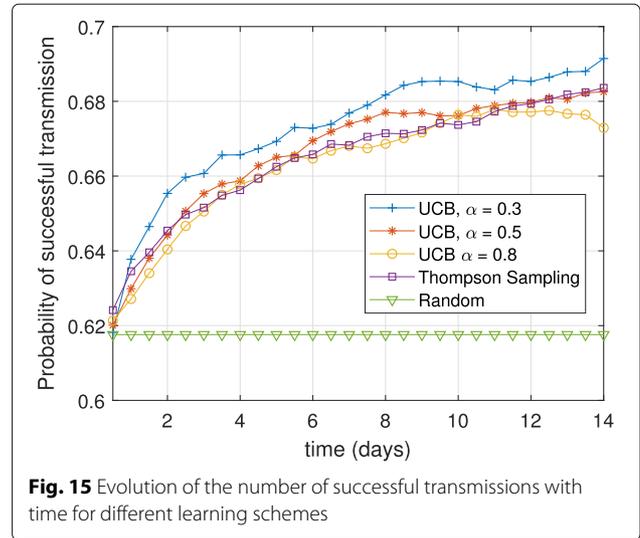

Fig. 15 Evolution of the number of successful transmissions with time for different learning schemes

scenario, we have less difference between the channels. As a consequence, according to Eq. (36), the gain that learning can bring is less important in this scenario. We display the obtained simulation results in Figs. 15 and 16.

In this second scenario, after 14 days of transmission, reinforcement learning algorithms provide a gain of 8 to 11% in probability of successful transmission. This reduction in the probability of successful transmission allows to reduce the average latency from 1.95 to around 1.65 s, i.e. a decrease in latency of 15%. These results show that learning algorithms can reduce the latency of communications even when the interfering traffic is generated by devices which use dissimilar packet sizes, i.e. different standards.

7 Conclusions

Unslotted ALOHA-based LPWAN standards such as LoRaWAN are perfect candidates for AMI backhaul. In this paper, we first derive closed-form and analyse the
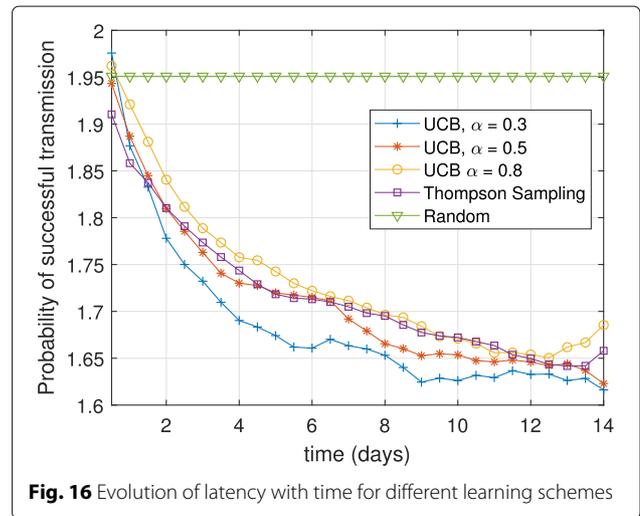

Fig. 16 Evolution of latency with time for different learning schemes

probability of successful transmission in a LoRaWAN-like LPWAN with acknowledgement in a channel. Then, we use these probabilities to analyse the latency in the network. Furthermore, we propose to use MAB learning algorithms as simple and efficient solutions to tackle the spectrum contention issue in unlicensed bands. We use the acknowledgement as a reward for online learning algorithms. The UCB1 and TS algorithms have a low cost in processing and energy consumption and do not require any extra signalling. Furthermore, in the studied scenario, these algorithms allow to increase by 14% the probability of successful transmission and to reduce by 40% the latency in the network. In our future work, we will either analyse other learning algorithms to tackle spectrum contention issues in IoT networks or consider a more realistic model, e.g. by considering the fading of wireless communications. We can also analyse the potential of MAB learning algorithms in different standards.

Endnote

[1] Sigfox is a French LPWAN operator whose network covers a large part of western Europe and is under deployment in the US. – www.sigfox.com.

Abbreviations

3GPP: 3rd Generation Partnership Project; AMI: Advanced metering infrastructure; CR: Cognitive radio; DA: Distribution automation; DER: Distributed energy resources; DSA: Dynamic spectrum acces; GPRS: General packet radio service ; IoT: Internet of Things; LPWAN: Low power wide area network; M2M: Machine-to-machine; MAB: Multi-armed bandit; NB-IoT: NarrowBand-Internet of things; PLC: Power line communications; RL: Reinforcement learning; SF: Spreading factor; TS: Thompson sampling; UCB: Upper confidence bound; UNB: Ultra narrow band

Availability of data and materials
The matlab code used for generating our simulation results can be found at https://bitbucket.org/scee_ietr/reinforcement-learning-in-unslotted-lpwan/.

Authors' contributions
RB is the main author of the current paper. CM and JP contributed to the conception and design of the study and to the structuring and reviewing of the manuscript. All authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details
[1]CentraleSupélec/IETR, CentraleSupélec campus de Rennes, Avenue de la boulaie, 35510, Cesson-Sévigné, France. [2]Univ Rennes, CNRS, IETR - UMR 6164, F-35000, Rennes, France.

References
1. RE Brown, in *Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*. Impact of Smart Grid on distribution system design (IEEE, 2008), pp. 1–4
2. U.S. Department of Energy, Communications requirements of smart grid technologies (2010)
3. J Gao, Y Xiao, J Liu, W Liang, CLP Chen, A survey of communication/networking in smart grids. Futur. Gener. Comput. Syst. **28**(2), 391–404 (2012)
4. Smart grid reference architecture, CEN-CENELEC-ETSI Smart Grid Coordination Group. Technical report (2012)
5. X Mamo, S Mallet, T Coste, S Grenard, in *2009 IEEE Power Energy Society General Meeting*. Distribution automation: the cornerstone for smart grid development strategy (IEEE, 2009), pp. 1–6
6. X Xiong, K Zheng, R Xu, W Xiang, P Chatzimisios, Low power wide area machine-to-machine networks: key techniques and prototype. IEEE Commun. Mag. **53**(9), 64–71 (2015)
7. M Centenaro, L Vangelista, A Zanella, M Zorzi, Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios. IEEE Wirel. Commun. **23**, 60–67 (2016)
8. U Raza, P Kulkarni, M Sooriyabandara, Low power wide area networks: an overview. IEEE Commun. Surv. Tutor. **PP**(99), 1–1 (2017)
9. R Ratasuk, B Vejlgaard, N Mangalvedhe, A Ghosh, in *2016 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. NB-IoT system for M2M communication (IEEE, 2016), pp. 428–432
10. W Webb, *Understanding weightless: technology, equipment, and network deployment for M2M communications in white space*, 1st. (Cambridge University Press, New York, 2012)
11. N Abramson, in *Proceedings of the November 17-19, 1970, Fall Joint Computer Conference. AFIPS '70 (Fall)*. THE ALOHA SYSTEM: Another Alternative for Computer Communications (ACM, New York, 1970), pp. 281–285
12. N Sornin, M Luis, T Eirich, T Kramp, O Hersent, LoRaWAN Specification. Technical report, LoRa Alliance, Inc. (2016)
13. M Anteur, V Deslandes, N Thomas, AL Beylot, in *2015 IEEE Global Communications Conference (GLOBECOM)*. Ultra Narrow Band Technique for Low Power Wide Area Communications (IEEE, 2015), pp. 1–6
14. Recommended SX1272 settings for EU868 LoRaWAN network operation. Technical report, Semtech (2015)
15. LoRa Alliance Technical committee, LoRaWAN regional parameters. Technical report, LoRa Alliance, Inc. (2016)
16. N Varsier, J Schwoerer, in *Communications (ICC), 2017 IEEE International Conference On*. Capacity Limits of LoRaWAN Technology for Smart Metering Applications (IEEE, 2017), pp. 1–6
17. PC Pinto, MZ Win, in *MILCOM 2008 - 2008 IEEE Military Communications Conference*. A unified analysis of connectivity and throughput in packet radio networks (IEEE, 2008), pp. 1–7
18. MT Do, C Goursaud, JM Gorce, in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2014 12th International Symposium On*. On the benefits of random FDMA schemes in ultra narrow band networks (IEEE, 2014), pp. 672–677
19. K Mikhaylov, J Petaejaejaervi, T Haenninen, in *European Wireless 2016; 22th European Wireless Conference*. Analysis of capacity and scalability of the LoRa low power wide area network technology (IEEE, 2016), pp. 1–6
20. B Vejlgaard, M Lauridsen, H Nguyen, IZ Kovacs, P Mogensen, M Sorensen, in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*. Coverage and capacity analysis of Sigfox, LoRa, GPRS, and NB-IoT (IEEE, 2017), pp. 1–5
21. D Magrin, M Centenaro, L Vangelista, in *Communications (ICC), 2017 IEEE International Conference On*. Performance Evaluation of LoRa Networks in a Smart City Scenario (IEEE, 2017), pp. 1–6
22. O Georgiou, U Raza, Low power wide area network analysis: can lora scale? IEEE Wirel. Commun. Lett. **PP**(99), 1–1 (2017)
23. TL Lai, H Robbins, Asymptotically efficient adaptive allocation rules. Adv. Appl. Math. **6**(1), 4–22 (1985)
24. P Auer, Using confidence bounds for exploitation-exploration trade-offs. J. Mach. Learn. Res. **3**, 397–422 (2003)
25. S Agrawal, N Goyal, Analysis of thompson sampling for the multi-armed bandit problem. CoRR. **abs/1111.1797**, 39.1–39.26 (2011)

Bonnefoi *et al. EURASIP Journal on Wireless Communications and Networking*   (2018) 2018:34

Page 18 of 18

26. W Jouini, D Ernst, C Moy, J Palicot, in *Communications (ICC), 2010 IEEE International Conference On*. Upper Confidence Bound Based Decision Making Strategies and Dynamic Spectrum Access (IEEE, 2010), pp. 1–5
27. Q Zhao, BM Sadler, A survey of dynamic spectrum access. IEEE Signal Proc. Mag. **24**(3), 79–89 (2007)
28. J Mitola, GQ Maguire, Cognitive radio: making software radios more personal. IEEE Pers. Commun. **6**(4), 13–18 (1999)
29. R Bonnefoi, C Moy, J Palicot, in *Smart Grid Communications (SmartGridComm), 2016 IEEE International Conference On*. Advanced metering infrastructure backhaul reliability improvement with cognitive radio, (2016)
30. R Bonnefoi, L Besson, C Moy, E Kaufmann, J Palicot, in *CROWNCOM 2017*. Multi-armed bandit learning in IoT networks: learning helps even in non-stationary settings (Springer, 2017)
31. V Toldov, L Clavier, V Loscri, N Mitton, in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. A Thompson sampling approach to channel exploration-exploitation problem in multihop cognitive radio networks (IEEE, 2016), pp. 1–6
32. C Goursaud, Y Mo, in *2016 23rd International Conference on Telecommunications (ICT)*. Random unslotted time-frequency ALOHA: theory and application to IoT UNB networks (IEEE, 2016), pp. 1–5
33. DJ Goodman, AAM Saleh, The near/far effect in local ALOHA radio communications. IEEE Trans. Veh. Technol. **36**(1), 19–27 (1987)
34. RG Gallager, *Discrete Stochastic Processes*. (Springer, Berlin, 1996)
35. E Kaufmann, O Cappe, A Garivier, in *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics (AISTATS-12). ND Lawrence, MA Girolami*. On Bayesian upper confidence bounds for bandit problems, vol. 22 (PMLR (Proceedings of Machine Learning), 2012), pp. 592–600. Journal of Machine Learning Research - Workshop and Conference Proceedings
36. L Melian-Gutierrez, N Modi, C Moy, F Bader, I Perez-Alvarez, S Zazo, Hybrid ucb-hmm: A machine learning strategy for cognitive radio in hf band. IEEE Trans. Cogn. Commun. Netw. **1**(3), 347–358 (2015)
37. N Modi, C Moy, P Mary, J Palicot, in *Cognitive Radio Oriented Wireless Networks: 11th International Conference, CROWNCOM 2016, Grenoble, France, May 30 - June 1, 2016, Proceedings*. A New Evaluation Criteria for Learning Capability in OSA Context (IEEE, 2016)