Robust CSMA for Long-Range LoRa Transmissions with Image Sensing Devices

Congduc Pham
Univ. Pau, LIUPPA Laboratory
congduc.pham@univ-pau.fr

***Abstract*—As long-range technologies allow for simpler connectivity of remote devices, a tremendous increase of the device density proposing innovative services is expected in the next years. In this article, we consider a dense deployment of IoT devices generating a wide range of message sizes: short messages from traditional telemetry devices, medium-size message from multi-sensors devices and large message from image sensors. We investigate how a Carrier Sense mechanism can be adapted for LoRa networks to decrease collisions, taking into account this variety of message sizes and the difficulty to get a reliable free channel indication. We show experimental results on a large-scale IoT LoRa test-bed implementing various use-cases from the EU H2020 WAZIUP project targeting IoT deployment in developing countries.**

I. Introduction

Low-Power Wide Area Networks (LPWAN) introduced both by Sigfox and Semtech's (i.e. LoRa$^{\text{TM}}$) are currently gaining incredible interest to connect so-called Internet-of-Things (IoT) devices. These low-power and long-range technologies have definitely contributed to the recent incredible uptake of small IoT devices in a large variety of applications as deployment, based on 1-hop connectivity, can be made much simpler while preserving battery lifetime. While the maturity of IoT devices providing simple physical measures, such as temperature, is demonstrated by the availability of a tremendous number of products on the market, more complex devices such as those allowing multimedia information to be sensed and delivered by resource-constrained devices are still in their very early stage of deployment. These multimedia IoT devices open a lot of new perspectives to a number of surveillance applications, one example being visual information for large-scale situation awareness in many application domains.

When considering cost, network availability and versatility, LoRa technology [1], which can be privately deployed in a given area without any service subscription, has a clear advantage over Sigfox which coverage is entirely operator-managed. However, in both technologies, the high receiver's sensibility that allows long-range transmissions is realized at the cost of a much lower throughput making the transmission of images a real challenge. In [2], we built our first image sensor prototype from off-the-shelves low-cost components by promoting maximum flexibility and modularity. Our motivations for the work described in [2] were: (1) to use only off-the-shelf components in order to provide maximum flexibility, evolutivity and reproducibility; and (2) to provide an efficient image compression algorithm to provide high compression ratio while producing a packet stream tolerant to packet losses.

We are proposing in this article a Carrier Sense Multiple Access (CSMA) mechanism adapted to LoRa physical layer to improve the robustness of long-range transmissions. This issue becomes more important when higher amount of data need to be transmitted and when the transmission time of a packet is increased. While the LoRaWAN specifications [3] may ease the deployment of LoRa networks by proposing some mitigation mechanisms to allow for several LoRa networks and thousands of nodes to coexist (such as multiple channels, orthogonal spreading factors, dynamic channel discrimination) a LoRa network working in a given set of parameters still remains similar to a simple ALOHA system, which performance limitations are well-known [4]. Due to the extremely low throughput of these long-range technologies (100bps-30kbps), the time-on-air (ToA) of message can be very large, typically in the range of several seconds, thus dramatically increasing the probability of collisions despite the limitation on the duty-cycle imposed by regulations. Figure 1 shows for various combinations of bandwidth (BW) and spreading factor (SF) the ToA of a LoRa packet as a function of the payload size in bytes. The maximum throughput is shown in the last column with a 255B-payload packet. Modes 4 to 6 provide quite interesting trade-offs for longer range, higher data rate and immunity to interferences but in practice, when maximum range is needed, mode 1 will be the de facto standard (these are actually the default parameters in LoRaWAN). In a recent article [5], the authors have studied the scalability of LoRa networks and they confirmed the low Data Extraction Rate when the number of nodes increases.

			time on air in second for payload size of						
LoRa mode	BW	SF	5 bytes	55 bytes	105 bytes	155 Bytes	205 Bytes	255 Bytes	max thoughput (255B packet) in bps
1	125	12	0.958	2.597	4.235	5.874	7.512	9.150	223
2	250	12	0.479	1.217	1.872	2.527	3.265	3.920	520
3	125	10	0.281	0.690	1.100	1.509	1.919	2.329	876
4	500	12	0.240	0.608	0.936	1.264	1.632	1.960	1041
5	250	10	0.140	0.345	0.550	0.755	0.959	1.164	1752
6	500	11	0.120	0.304	0.509	0.693	0.878	1.062	1921
7	250	9	0.070	0.183	0.295	0.408	0.521	0.633	3221
8	500	9	0.035	0.091	0.148	0.204	0.260	0.317	6442
9	500	8	0.018	0.051	0.082	0.115	0.146	0.179	11408
10	500	7	0.009	0.028	0.046	0.064	0.083	0.101	20212

Fig. 1. Time on air for various LoRa modes as payload size is varied

To the best of our knowledge, there is limited published works discussing channel access methods for LoRa. There are mostly contributions on limitations of current LoRa technology [5], [6], [7] rather than on proposing enhancements. In this article, we investigate how a Carrier Sense (CS) mechanism can be adapted to decrease collisions in LoRa transmissions and show experimental results on a large scale LoRa testbed that includes image IoT devices.

The rest of the article is organized as follows. Section II presents our low-cost IoT platform and the large scale test-bed used for all the experiments. In Section III, we review the main CSMA methods found in wireless networks such as IEEE 802.11 (WiFi) and IEEE 802.15.4 and present the steps leading to a CSMA mechanism adapted to the specific case of LoRa technology and capable of handling both short and long LoRa messages in real-world deployment scenarios. Results and discussion will be presented. We conclude in Section IV.

II. Low-cost & long-range IoT platform

A. Low-cost, DIY IoT

Our IoT platform is developed in the context of the EU H2020 WAZIUP project. It fully takes the ”Arduino” philosophy of low-cost, simple-to-program yet efficient hardware platforms, that is ideally well-suited for do-it-yourself (DIY) IoT, especially in WAZIUP that addresses rural applications in developing countries [8]. The Arduino-compatible ecosystem is large and proposes various board models, from powerful prototyping boards to smaller and less energy-consuming boards for final integration purposes. For instance, the small form factor Arduino Pro Mini board that is available in the 3.3v & 8MHz version for much lower power consumption can definitely be used to provide a generic low-cost IoT platform as it can be purchased for less than 2 euro from Chinese manufacturers.

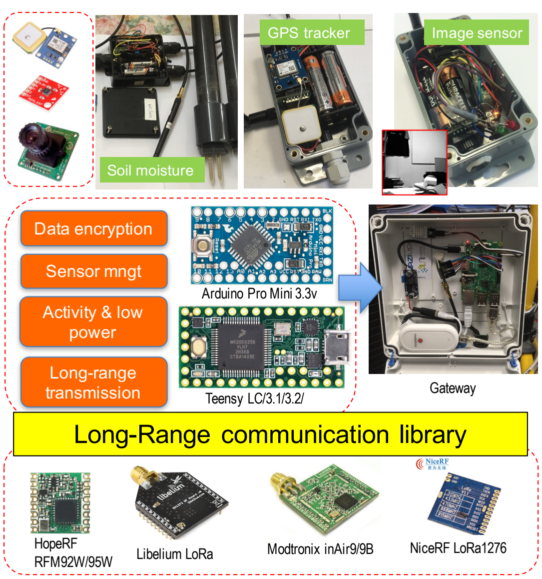

Fig. 2. Generic IoT platform and software building blocks

For more demanding IoT applications, such as image sensing, we use the Teensy family boards (LC/31/32) that offer state-of-the-art micro-controllers with more memory and advanced power management features at a very reasonnable cost (about 10 euro for the LC). The generic platform integrates software building blocks in ready-to-use templates for quick and easy customization, see Fig. 2.

This generic platform is used in WAZIUP to propose 4 Minimum Viable Product (MVP): Cattle Rustling, AGRI, Water-Fish Farming, and Waste Mgnt. Significant real-world deployment have already been realized in Senegal (Cattle Rustling), Ghana (Fish Farming, AGRI-Weather) and Pakistan (AGRI-Soil with multi-level soil moisture for crop irrigation). The latter was done in collaboration with the Nestlé's WaterSense project. With efficient power management, the generic device offers several years of autonomy with simple AA batteries on the base of 1 measure/hour. Although not presented in this paper but illustrated in Fig. 2, our platform also includes a low-cost LoRa gateway to receive, manage and present data from end-devices in a very flexible manner. The gateway is built on the well-known Raspberry PI – all models are supported – and the cost of the entire gateway can be less than 45 euro. More details and all software can be found in [9].

B. Test-bed

The test-bed used for all the experiments presented in this article consists in a large variety of sensor devices and one gateway deployed at the Gaston Berger University in Saint-Louis, Senegal, which serves as test site for all WAZIUP's MVP pilots. Figure 3 shows several areas in UGB that host the MVP pilots: an experimental farm hosts the AGRI-Soil MVP, fish ponds host the Water MVP, the CIMEL center for cattle research hosts the Cattle Rustling MVP, a weather station is deployed for the AGRI-Weather MVP. As can be seen in the figure, 3 images sensors are deployed: 2 for situation-awareness (indoor for the moment) and 1 as part of a test of AGRI MVP consisting in plant monitoring.

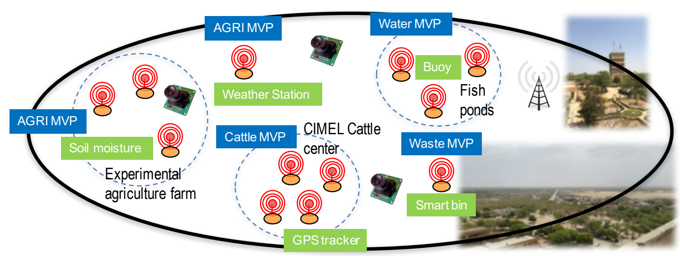

Fig. 3. Deployment at University Gaston Berger

These various MVPs will generate different LoRa message sizes: (i) small messages, typically under 20 bytes, for simple single-sensor devices such as the GPS tracker collars (Cattle Rustling MVP), soil moisture sensors (AGRI MVP) and smart bin (Waste MVP) ; (ii) medium-size messages, between 20 and 60 bytes, for simple multi-sensor devices (combined air temperature, air humidity, water temperature, dissolved oxygen level,...) such as the low-cost buoy (Water MVP) and Weather Station (AGRI MVP); and (iii) long messages, typically above 100 bytes, for image sensors. Fig. 4 shows the devices from the various MVPs deployed in the test-bed. A dedicated node will constantly monitor the radio channel activity performing Channel Activity Detection (CAD) procedure of LoRa radio chip. As CAD is an important component used for performing Carrier Sense we will present this feature in more details later on. This device is attached to a computer to plot the observed channel activity.

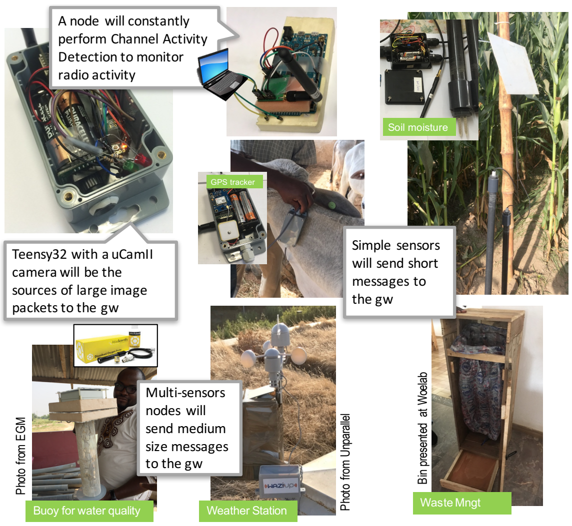

Fig. 4. Various devices of the test-bed

Our image sensor is based on a Teensy32 board and a 4D System uCamII camera configured for 8bpp gray-scale and 128x128 images. The image sensor runs on 4 AA batteries and is fully autonomous with low-power features. The image encoding scheme is adapted for low-resource devices, supports high packet-loss rates and features an image quality factor parameter to ajust the compression ratio. The control software periodically takes a snapshot (one per hour for instance) and transmit the encoded image to the gateway (which will decode the image and make it available through an embedded web page). As can be seen in Fig. 5, using a quality factor of 10 offers a high trade-off between image size (compression ratio of 18) and visual quality.

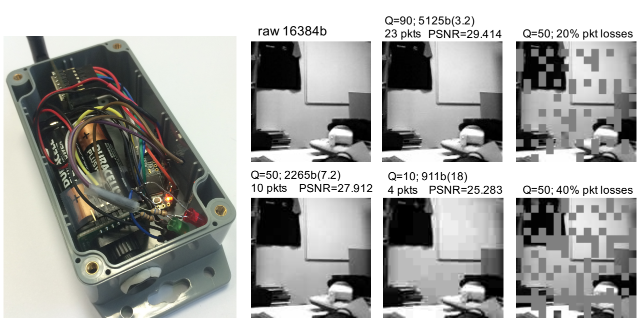

Fig. 5. The image sensor device

A typical generated image with a quality factor of 10 is typically 900-1200 bytes (i.e. between 4 and 5 packets when maximum packet size if set to 235 bytes) and can be transmitted within the ETSI limit of 36s of radio time allowed per hour in Europe. If larger size images are necessary, they can be transmitted on 2 successive cycles giving an image rate of 1 image/2 hours. More detail on our long-range image sensor device for situation-awareness scenarios can be found in [10].

A summary of IoT traffic on the test-bed is presented in Figure 6. At the UGB test-bed, the image sensors send an image every 15 minutes as there is no duty-cucle regulations in Senegal. Doing so emulates a larger number of image devices.

Device	QT	Message type	Traffic profile
GPS Tracker	5	small	1 message every 10mins
Soil Moisture	10	small	1 message every 60mins
Smart bin	2	small	1 message every 60mins
Weather Station	1	medium	1 message every 15mins
Buoy	2	medium	1 message every 30mins
Image sensor	3	long	1 image (4-5 packets) every 15 mins

Fig. 6. Summary of test-bed traffic

III. Channel access for LoRa networks

A. Review of CSMA and IEEE 802.11 CSMA/CA

As stated in the introduction, the scalability of LoRa networks can be a serious issue as there is almost no channel access mechanism defined leading to the so-called ALOHA access with poor performance. There has been a notable amount of research done on the performance of ALOHA and CSMA in wireless networks. It is beyond the scope of this paper to go through all these contributions but interested readers can start with [11], [12], [13]. Among many CSMA variants, the one implemented in the IEEE 802.11 (WiFi) is quite representative of the approach taken by most of random access protocols with so-called backoff procedure. Fig. 7 illustrates the IEEE 802.11 CSMA mechanism used in the basic Distributed Coordinated Function (DCF) mode which is the common operation mode of WiFi networks with a base station. In this basic mode, the optional RTS/CTS mode is not used. The basic DCF IEEE 802.11 CSMA/CA (Collision Avoidance) works as follows:

- A node senses the channel to determine whether another node is transmitting before initiating a transmission
- If the medium is free for a DCF inter-frame space ($DIFS$) the transmission will proceed (green $DIFS$)
- If the medium is busy (red $DIFS$), the node defers its transmission until the end of the current transmission and waits an additional $DIFS$ before generating a random number of backoff slot time in the range $[0, W-1]$.
- The backoff timer is decreased as long as the medium is sensed to be idle, and frozen when a transmission is detected on the medium, and resumed when the channel is detected as idle again for more than DIFS
- When the backoff reaches 0, the node transmits its packet
- The initial W is set to 1. W is doubled for each retry (exponential backoff) until it reaches a maximum value

The random backoff timer is applied after a busy channel because it is exactly in that case that the probability of a collision is at its highest value. This is because several users could have been waiting for the medium to be available again.

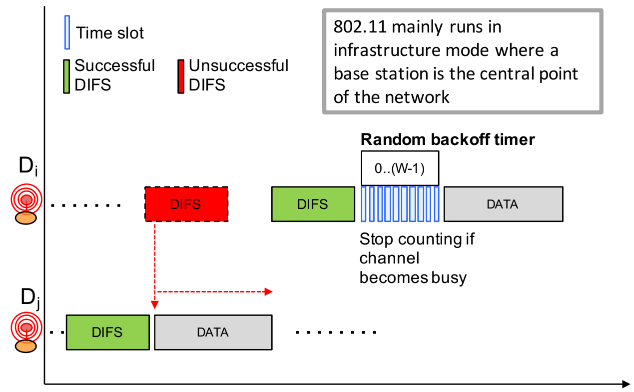

Fig. 7. IEEE 802.11 DCF CSMA/CA

B. What can be done for LoRa?

1) LoRa's channel activity detection (CAD): Before investigating what CSMA approach can be adapted for LoRa, it is necessary to know how a LoRa channel can be defined busy or idle to implement a CS mechanism. As LoRa reception can be done below the noise floor the use of the RSSI is not reliable enough. For clear channel assessement, there is a special Channel Activity Detection (CAD) procedure that can be realized by a LoRa chip. We use the dedicated Arduino Due device to constantly perform CAD procedure and a dedicated interactive device to send periodic messages (see previous Fig. 5). Fig. 8 shows 2 cases: (i) 44 byte message (40 bytes payload + 4 byte header) every 15s with a CAD procedure every 100ms and (ii) 244 byte message (240+4) every 15s with a CAD procedure every 1000ms. As can be seen in Fig. 8 the LoRa CAD procedure can correctly detect all the LoRa transmission, and not only the preamble.

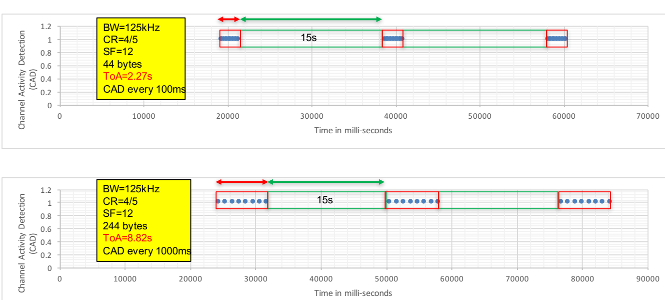

Fig. 8. Test of the LoRa CAD mechanism

2) Adaptation from 802.11: As a first attempt towards a CSMA protocol for LoRa, we start by adapting the previously shown 802.11 CSMA protocol and not the 802.15.4 one, although 802.15.4 is widely used in WSN and early IoT implementation, for 2 reasons. The first reason is that LoRa network architecture is mainly a single-hop star topology from devices to gateway, which is very similar to the WiFi topology with a base station. Therefore, the concept and the management of the 802.11's random backoff timer after a busy channel looks efficient for such environment. The second reason for not starting from 802.15.4 comes from its initial random waiting without channel sensing method that is more suitable for low density networks than for high density networks that will definitely be the case for LoRa networks.

To adapt the 802.11 CSMA protocol, we first need to define how the $DIFS$ operation can be implemented. Usually, IFS should be related somehow to the symbol period T_{sym}. For LoRa, T_{sym} depends on BW and SF as follows: $T_{sym} = 2^{SF}/BW$. For instance, LoRa mode 1 use BW=125kHz and SF=12 therefore $T^{mode\_1}_{sym} = 2^{12}/125000 = 0.032768$. In [14], it is reported that the CAD duration is between $1.75T_{sym}$ and $2.25T_{sym}$ depending on the spreading factor, see Fig. 9. We performed some experimental tests to verify the real CAD duration against what is given in [14]: Fig. 9 also shows the minimum and the maximum values measured with a 1ms-accuracy clock (the Arduino `millis()` function). We can see that the measured CAD durations are quite consistent.

LoRa mode	Tsym (ms)	CAD duration (Tsym)	CAD duration (ms)	Experimental measures min value	max value
1	32.768	1.86	60.948	60	62
2	16.384	1.86	30.474	29	31
3	8.192	1.77	14.500	14	16
4	8.192	1.86	15.237	15	16
5	4.096	1.77	7.250	7	8
6	4.096	1.81	7.414	7	9
7	2.048	1.75	3.584	3	5
8	1.024	1.75	1.792	1	3
9	0.512	1.79	0.916	1	1
10	0.256	1.92	0.492	0	1

Fig. 9. Theoretical CAD duration and experimental measures

In our current implementation $DIFS$ does not depend directly on T_{sym} but on the duration of the CAD mechanism therefore we assign an integer number of CAD to $DIFS$. Our communication library provides a low-level `doCAD(counter)` function that takes an integer number of CAD, i.e. `counter`, performs sequentially the requested number of CAD and returns 0 if all CAD have been successful (no channel activity). If one CAD detects activity the function exits with value greater then 0. The $DIFS$ procedure shown in Fig. 10 works that way and once a failed CAD has been observed the node exits the $DIFS$ procedure and continuously checks for a free channel.

In Fig. 10, $DIFS$ is assigned 9 CAD which gives a duration of about $9 \times 61ms = 549ms$ for LoRa mode 1. At this point of the study, the duration of $DIFS$ is not really important as we only need to be able to assert a free channel for a given duration. The value of 9 CAD provides enough time to detect channel activity and also provides the possibility to define a much shorter timer (using 3 CAD for instance), such as the 802.11's $SIFS$, to implement priority schemes is needed, and still be able to detect channel activity. Then the random backoff timer is also defined as a number of CAD because the channel should be checked in order to froze or continue the decrease of the backoff timer. The upper bound, W, of the random backoff timer can be set in relation to the number of CAD defined for $DIFS$. For instance, if $DIFS = 9$ CAD then W can be defined as $n \times DIFS$. For instance, if $n = 2$ then $W = 2 \times 9 = 18$ CAD.

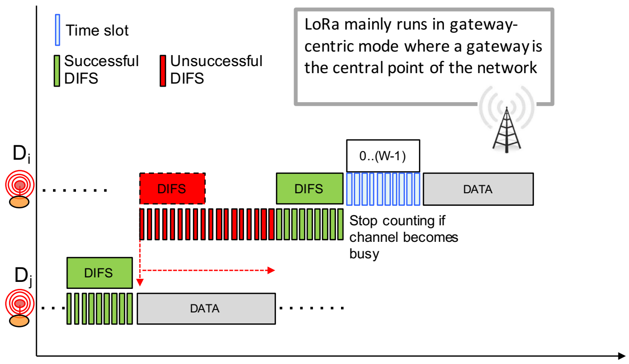

Fig. 10. CSMA mechanism adapted from IEEE 802.11

It is also possible to double W for each retry (exponential backoff) until it reaches a maximum value. However, while 802.11 initiates a retry when no ACK is received after a given time, the usage of acknowledgement is not common in LoRa as it is very costly for the gateway (the gateway is considered as a normal node and therefore its radio duty-cycle can be limited by regulations). Therefore there is no such retry concept with unacknowledged transmissions. Nevertheless, when 802.11 doubles W for each retry the underlying assumption for the transmission errors is a denser channel. Here, we can follow the same guideline and double W each time the channel cannot be found free for an entire $DIFS$, starting from the second $DIFS$ attempt. In the current implementation we set $W = 18$ CAD initially and we can double it 3 times so the maximum value is $W = 144$ CAD which will give a maximum wait timer of 8784ms for LoRa mode 1. If we add the value of the successful $DIFS$ which is 9 CAD, i.e. 549ms, then the maximum total wait timer after a busy channel is about 9333ms which correspond roughly to the ToA of the maximum LoRa packet size. This property remains roughly true for all the defined LoRa modes and therefore can avoid waiting longer than necessary.

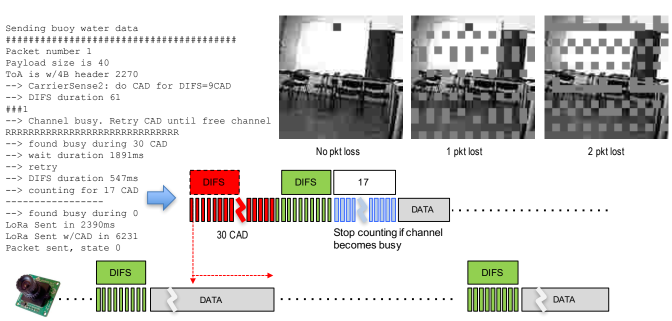

Fig. 11. Experimental test of the proposed CSMA adaptation

Fig. 11 shows an experiment with an image sensor sending 4 image packets (about 240 bytes per packet) while some nodes are sending medium-size messages of 40 bytes. The text output is from a buoy node and it can be seen that the adapted CSMA protocol can nicely avoid the collision by deferring the transmission of the buoy's message. In the illustrated experiment, transmission is deferred only once before transmission succeeds as the time between 2 image packets is greater than a $DIFS$ plus the random backoff timer of 17 CAD. Fig. 11 also shows the received image without any packet loss and 2 examples of received images when there is no channel access mechanism (pure ALOHA). It all our tests, the proposed CSMA protocol adapted from 802.11, and further referred to as $CSMA_{802.11}^{LoRa}$, totally avoids packet losses for both the image sensor and the other devices.

C. CAD reliability issues

By testing further the CSMA mechanism in various long-range deployment, we observed a fast decrease of the CAD's reliability when distance increases: although a transmission can be successful at several kilometers, CAD starts to not reliably detect the whole transmission when the distance to the sender is about 1km (with dense vegetation, CAD reliability can start to decrease even at 400m). Fig. 12 shows CAD reliability with the same traffic pattern previously shown in Fig. 8 but with the sender and the Arduino Due device performing CAD separated by 400m with some trees between them. As can be seen, the CAD procedure fails to detect channel activity many times during an on-going transmission.

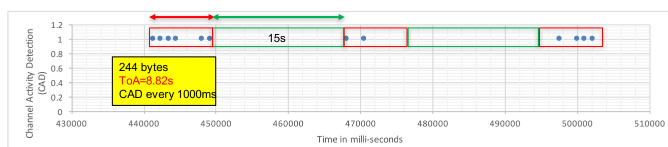

Fig. 12. CAD fails to detect activity of on-going transmissions

This CAD unreliability issue in real-world deployment scenario has a huge negative impact on the CS mechanism. For instance, in the previous proposed CSMA adaptation from 802.11, it is not possible anymore to rely on CAD to detect when the channel will become really free after a busy state nor to rely on a successful $DIFS$ as a free channel indication to start transmission. However, what can be observed in Fig. 8 and verified by the tests that we performed, is that during a long transmission the probability that all CAD attempts fail is quite low. In all our tests, and up to 1km in NLOS conditions, there have always been some successful CAD during any transmission.

D. Proposed CSMA mechanism

The CAD reliability issue raised previously calls for a different approach to prevent collisions. First, the previous $DIFS$ is extended to the ToA of the longest LoRa packet in a given LoRa mode, e.g. 9150ms for 255 bytes in LoRa mode 1 (see

Fig. 1). During this extended $DIFS(ToA_{max})$, CAD procedure is performed periodically (for instance every 1000ms as in Fig. 8–bottom). The purpose of $DIFS(ToA_{max})$ is to maximize the probability to detect an on-going transmission which can possibly be a long message with many unsuccessful CADs, thus appearing by mistake as a short message.

Then, when a CAD fails during a $DIFS(ToA_{max})$, instead of continuously waiting for a free channel followed by a $DIFS$+random backoff timer where CAD is checked constantly; here, there is a simple constant waiting period (pure delay) of ToA_{max}. Again, the purpose of the constant delay of ToA_{max} is to avoid performing CAD and transmission retries during the transmission of a possible long message, as a successful CAD does not guarantee a free channel. After the delay, the transmitter will try again to see a free channel for at least a $DIFS(ToA_{max})$ and the process continues until a maximum number of retries have been performed. The new CSMA proposal is illustrated in Fig. 13.

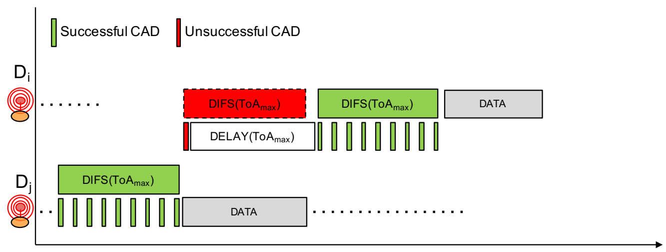

Fig. 13. New CSMA proposition

It all our tests with the new proposed CSMA protocol, noted $CSMA^{LoRa}_{new}$, we totally avoids packet losses for both the image sensor and the other devices even when the nodes are hundredth of meters away from each others.

E. Discussions

1) CAD frequency during $DIFS(ToA_{max})$: A CAD procedure takes between 0.5ms and 61ms, from mode 10 down to mode 1, as shown in Fig. 9 while the ToA of the longest LoRa packet, ToA_{max}, is respectively between 100ms and 9150ms as shown in Fig. 1. Therefore, depending on the CAD failure probability (not detecting an on-going transmission) it is possible to increase or decrease the number of CAD during a $DIFS(ToA_{max})$ to ensure at least 1 successful CAD to detect an on-going transmission. In our tests, we set the number of CAD to 9, similar to the number of CAD defined for a $DIFS$ in section III-B. Therefore the time between 2 CAD is $ToA_{max}/(9-1)$. For instance, in LoRa mode 1 where ToA_{max} =9150ms, there will be one CAD every 1143ms.

2) Energy considerations: We can compare the energy consumption between $CSMA^{LoRa}_{802.11}$ and $CSMA^{LoRa}_{new}$ with the scenario depicted in Fig. 14: a long packet is transmitted by device j after a successful $DIFS$ and there is an attempt from device i right at the beginning of this transmission. In Fig. 14 there are 2 lines for each device, the first line shows $CSMA^{LoRa}_{802.11}$ while the second line shows $CSMA^{LoRa}_{new}$.

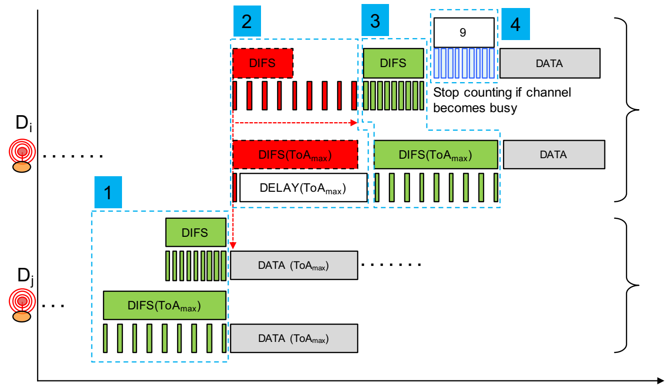

Fig. 14. Scenario for comparing $CSMA^{LoRa}_{802.11}$ and $CSMA^{LoRa}_{new}$

To perform the energy comparison, we measured for the Arduino Pro Mini and the Teensy32 the drawn current when performing CAD, when waiting using the `delay()` function and when waiting using deep sleep (DS) mode.

- Arduino Pro Mini
  - CAD: 12mA; `delay()`: 5.7mA; DS: 5uA
- Teensy32
  - CAD: 36mA; `delay()`: 29.5mA; DS: 110uA

As expected, deep sleep mode provides a very low energy consumption compared to the `delay()` function and CAD operation. Therefore it is possible to state that $E_{DIFS} = E_{DIFS(ToA_{max})} = 9 \times E_{CAD}$. With this approximation, sensing for a free channel before transmission at device j – block 1 – has comparable energy consumption level in $CSMA^{LoRa}_{802.11}$ and $CSMA^{LoRa}_{new}$.

Then, for device i, with $CSMA^{LoRa}_{802.11}$, checking until the end of the transmission – block 2 – can be comparable a $DIFS(ToA_{max})$ with periodic CAD performed 9 times. Therefore the energy consumption can be approximated again to $9 \times E_{CAD}$. With $CSMA^{LoRa}_{new}$, with the example depicted in Fig. 14, $DIFS(ToA_{max})$ fails at the first CAD to continue with $DELAY(ToA_{max})$ which has negligible energy consumption when using deep sleep mode for the waiting. Therefore, block 2 for $CSMA^{LoRa}_{new}$ has an energy consumption of $1 \times E_{CAD}$.

Block 3 for both CSMA protocols is comparable to block 1. Then, for device i with $CSMA^{LoRa}_{802.11}$ there is the random backoff timer – block 4. Assuming that the channel is always free for the pending transmission then the mean timer value is $W/2$. As W is initially set to 18 CAD then the random backoff timer has a mean duration of 9 CAD, thus an energy consumption of $9 \times E_{CAD}$.

Finally, for the scenario depicted in Fig. 14, $CSMA^{LoRa}_{802.11}$ has a total energy consumption of $4 \times [9 \times E_{CAD}]$ while $CSMA^{LoRa}_{new}$ has an energy consumption of $2 \times [9 \times E_{CAD}] + 1 \times E_{CAD}$ which is about half the energy consumption of $CSMA^{LoRa}_{802.11}$ – exactly $36/19$ time less. If the channel is found busy in block 3, then block 2 is repeated N times with an energy consumption ratio of 1:9 for $CSMA^{LoRa}_{new}$. Thus, in "heavy" traffic load, $CSMA^{LoRa}_{new}$ definitely shows a much lower energy consumption than $CSMA_{802.11}^{LoRa}$: $(3+N) \times [9 \times E_{CAD}]$ for $CSMA_{802.11}^{LoRa}$ and $2 \times [9 \times E_{CAD}] + N \times E_{CAD}$ for $CSMA_{new}^{LoRa}$. With $N = 2$ for instance, the ratio becomes $45/20$ which is now more than half.

If we take into account the CAD success probability (detecting an on-going transmission), noted $P_{CAD} =]0, 1]$, then the total energy consumption of for $CSMA_{new}^{LoRa}$ increases to $2 \times [9 \times E_{CAD}] + N \times \frac{1}{P_{CAD}} \times E_{CAD}$. Fig. 15 shows the energy consumption when varying N and P_{CAD} in number of CAD. To get the real energy consumption, we have to multiply by the duration of a CAD in a given LoRa mode, see Fig. 9.

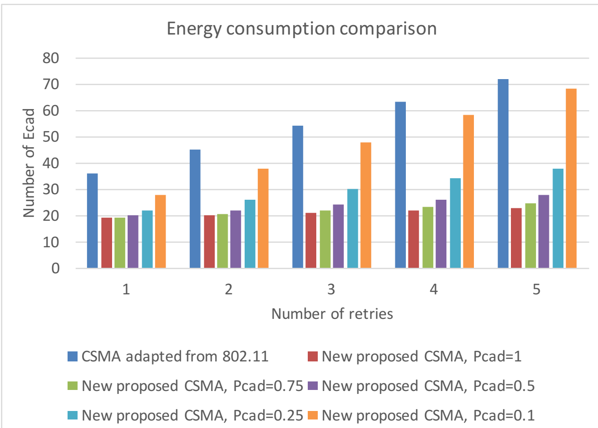

Fig. 15. Energy comparison of $CSMA_{802.11}^{LoRa}$ and $CSMA_{new}^{LoRa}$

Now, if we compare $CSMA_{new}^{LoRa}$ to a raw LoRa transmission without carrier sense then the additional cost of performing a carrier sense mechanism is simply $9 \times E_{CAD}$ when assuming that the channel is free (case of device j in Fig. 14). If the channel is not free then the raw LoRa transmission would create a packet collision and a comparison would be unfair.

3) Latency: $CSMA_{new}^{LoRa}$ obviously increases the sending latency because $DIFS(ToA_{max})$ is much larger than $DIFS$ (9150ms compared to 549ms for LoRa mode 1 and 255 bytes messages). Also, instead of continuously checks for a free channel in block 2, the node attempting to transmit always waits for $DELAY(ToA_{max})$. However, it is also possible to set the maximum packet size to a smaller value, i.e. 150 bytes, even for image packets, thus reducing $DIFS(ToA_{max})$, i.e. from 9150ms to 5874ms. When doing so, the number of image packets per image will increase and the additional overhead would only consist in the 4-byte header per packet.

IV. Conclusions

In this article, we investigated how a Carrier Sense mechanism can be adapted to decrease collisions in LoRa transmissions. We proposed a CSMA protocol adapted to LoRa networks, capable of handling both short and long messages. Experimental tests with image sensor nodes for innovative long-range image transmission showed very promising results where long on-going transmissions can be secured to avoid collisions even when the nodes are hundredth of meters away from each others.

Acknowledgments

This work is supported by the WAZIUP project with funding from the EU's Horizon 2020 research and innovation programme under grant agreement No 687607.

References

[1] Semtech, "Lora modulation basics. rev.2-05/2015," 2015.

[2] C. Pham, "Low cost wireless image sensor networks for visual surveillance and intrusion detection applications," in *12th IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Apr 2015.

[3] L. Alliance, "Lorawan specification, v1.0," 2015.

[4] R. Nelson and L. Kleinrock, "The spatial capacity of a slotted aloha multi-hop packet radio network with capture," *IEEE Trans. Comm.*, vol. 32, 1984.

[5] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do lora low-power wide-area networks scale?" in *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '16. New York, NY, USA: ACM, 2016, pp. 59–67. [Online]. Available: http://doi.acm.org/10.1145/2988287.2989163

[6] D. Bankov, E. Khorov, and A. Lyakhov, "On the limits of lorawan channel access," in *Proceedings of International Conference on Engineering and Telecommunication (EnT)*, 2016.

[7] K. Mikhaylov, J. Petaejaejaervi, and T. Haenninen, "Analysis of capacity and scalability of the lora low power wide area network technology," in *Proceedings of the 22th European Wireless Conference*, 2016.

[8] C. Pham, A. Rahim, and P. Cousin, "Low-cost, long-range open iot for smarter rural african villages," in *Proceedings of the IEEE International Smart Cities Conference (ISC2)*, 2016.

[9] C. Pham, "A diy low-cost lora gateway. http://cpham.perso.univ-pau.fr/lora/rpigateway.html and https://github.com/congducpham/lowcostloragw," accessed June 30th, 2017.

[10] ——, "Low-cost, low-power and long-range image sensor for visual surveillance," in *Proceedings of the 2nd Workshop on Experiences with Design and Implementation of Smart Objects (SMARTOBJECTS'16). Co-located with ACM MobiCom'2016.*, 2016.

[11] M. Kaynia and N. Jindal, "Performance of aloha and csma in spatially distributed wireless networks," in *Proceedings of IEEE International Conference on Communications (ICC).*, 2008.

[12] Y. Yang and T.-S. P. Yum, "Delay distributions of slotted aloha and csma," *IEEE Trans. Comm.*, vol. 51, 2003.

[13] F. A. Tobagi, "Distribution of packet delay and interdeparture time in slotted aloha and carrier sense multiple access," *J. Assoc. Comput. Mach.*, vol. 29, 1982.

[14] Semtech, "Sx1272/73 - 860 mhz to 1020 mhz low power long range transceiver. rev.2-07/2014," 2014.