Enabling and deploying long-range IoT image sensors with LoRa technology

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

***Abstract*—Recent Low-Power Wide Area Networks (LPWAN) networks, e.g. Lora, introduce a high level of flexibility when deploying IoT devices. However, this flexibility also raises some performance issues that can become critical when introducing innovative IoT devices such as image sensor devices. This paper presents 2 control mechanisms to enable the deployment of such image sensor devices with LPWAN technologies: (i) an adapted Carrier Sense Multiple Access (CSMA) mechanism to avoid costly packet collision and (ii) an activity time sharing mechanism to mitigate the issue of duty-cycle. Both mechanisms show their efficiency in real experiments.**

***Index Terms*—IoT, LPWAN, MAC layer, duty-cycle.**

I. Introduction

Internet of Things (IoT), and previously Wireless Sensors Networks (WSN), allows for close interaction between the physical world and digital systems. One of the earliest applications of IoT/WSN consists in building and deploying monitoring and surveillance systems. There are now a growing interest to go beyond simple measures such as temperature in order to add rich visual information for deploying innovative monitoring applications.

However, it was not long ago when deploying even simple IoT node in a large-scale was already a challenge. Telecommunication cellular solutions such as GSM consume a high amount of energy and are hardly suitable for battery-operated devices that must offer several years of autonomy. Shorter range radio devices such as those using IEEE 802.15.4 can use multi-hop routing to offer longer range but this is realized at the cost of a much higher level of complexity, in addition to require very high node density. While these techniques can be somehow deployed in a smart-city scenario, they can hardly be considered in remote/rural areas.

Recently, Low-Power Wide Area Networks (LPWAN) concept based on ultra-narrow band modulation (UNB) – for Sigfox – or Chirp Spread Spectrum modulation (CSS) – for LoRa [1] – has attracted attention with their capability to provide long range communication with a much lower power consumption to enable several years of operations on batteries. These technologies can achieve more than 15km in LOS condition and they definitely provide a better connectivity answer for battery-operated IoT by avoiding complex and costly relay nodes as a star topology with a central gateway can be deployed similar to cellular network topology. Therefore, even though these technologies are not yet standards endorsed by recognized standardization bodies, they can be considered as de-facto standards in the emerging LPWAN ecosystem. While both LoRa and Sigfox are competing for the LPWAN market, LoRa technology can be privately deployed when Sigfox adopts the operator-based approach. Therefore, when considering flexibility and versatility, LoRa is more suitable for ad-hoc deployment scenario in remote and rural areas.

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 which produces a packet stream tolerant to packet losses. There have been many research on image platforms such as SeedEyes [3], Panoptes [4], Cyclops [5] and iMote2/IMB400 [6] to name a few but most of these contributions develop ad-hoc visual hardware or use powerful Linux-based platforms. As a consequence, very few of them actually address the issues raised by very low-bandwidth radio environments.

The long-range version of our image sensor using LoRa radio technology has been described in [7]. We demonstrated that the off-the-shelves component approach for our image sensor node allows for fast and efficient integration of these new technologies. We proposed the long-range version of our image IoT platform to (1) avoid relying on operator-based communications; (2) remove the complexity and cost of deploying a multi-hop short range infrastructure; and (3) offer out-of-the-box long-range visual surveillance facilities.

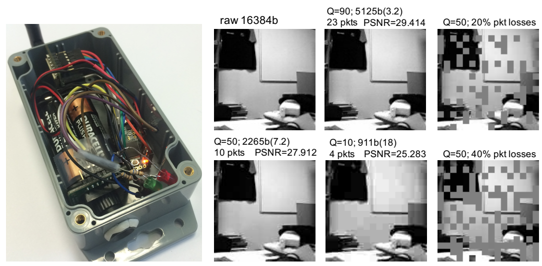

Fig. 1. Long-range image sensor device

Figure 1 shows the image node based on a Teensy3.2 board that drives the CMOS uCamII/III camera. 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 adjust the compression ratio. As can be seen in Fig. 1, using a quality factor of 10 offers a high trade-off between image size (compression ratio of 18) and visual quality. Using a quality factor of 10 gives an image size of 900-1200 bytes that can be packed in 4 to 5 LoRa packets. The image sensor is programmed to be completely autonomous once powered on: it can either send image in a periodic manner (situation awareness scenario) or can run an image change detection mechanism to only send an image when there are significant changes in the scene (event detection scenario). When taking 1 image/hour, the image sensor can run for more than a year on batteries.

There are however some constraints to the flexibility of long-range transmissions. For instance, using current unlicensed bands impose strict limitation on the radio duty-cycle. Another issue resides in the access to the radio medium which is currently lacking in current technologies. In this paper, we present 2 control mechanisms that are contributing to enable the deployment of long-range image sensors with LoRa technology. The rest of the paper is organized as follows. Section II reviews the LPWAN LoRa technology and especially focuses on its main identified constraints and limitations for supporting image IoT devices. Then, Section III presents the 2 proposed control mechanisms: (i) an adapted Carrier Sense Multiple Access (CSMA) mechanism to avoid costly packet collision and (ii) an activity time sharing mechanism to mitigate the issue of duty-cycle. We conclude in Section IV.

II. LPWAN LIMITATIONS AND CONSTRAINTS

A. *Inefficient radio medium access control*

Long-range transmissions has the obvious side effect of increasing the number of devices competing for the radio medium. 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 [8] 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 [9]. 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.

Figure 2 shows for various combinations of bandwidth (BW) and spreading factor (SF) the ToA (at the physical layer) of a LoRa packet as a function of the payload size in bytes. Transmissions with different SF do normally not interfere each other but in practice, when maximum range is needed, using SF=12 is de facto standard and is actually the default SF value in LoRaWAN networks. In a recent article [10], 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 (kHz)	SF	5 bytes	55 bytes	105 bytes	155 Bytes	205 Bytes	255 Bytes	max thoughput in bps
1	125	12	0.9585	2.5969	4.2353	5.8737	7.5121	9.1505	223
2	250	12	0.4792	1.2165	1.8719	2.5272	3.2645	3.9199	520
3	125	10	0.2806	0.6902	1.0998	1.5094	1.919	2.3286	876
4	500	12	0.2396	0.6083	0.9359	1.2636	1.6323	1.9599	1041
5	250	10	0.1403	0.3451	0.5499	0.7547	0.9595	1.1643	1752
6	500	11	0.1198	0.3041	0.5089	0.6932	0.8776	1.0619	1921
7	250	9	0.0701	0.1828	0.2954	0.4081	0.5207	0.6333	3221
8	500	9	0.0351	0.0914	0.1477	0.204	0.2604	0.3167	6442
9	500	8	0.0175	0.0508	0.0815	0.1148	0.1455	0.1788	11408
10	500	7	0.0088	0.028	0.0459	0.0638	0.083	0.1009	20212

Fig. 2. Time on air for various LoRa modes as payload size is varied

Carrier Sense Multiple Access (CSMA) method is the base to most wireless channel sharing mechanism. It combines clear channel assessment (CCA) with random waiting. However, 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 the CS mechanism. As LoRa reception can be done below the noise floor the use of the RSSI is not reliable enough. For CCA, there is a special Channel Activity Detection (CAD) procedure that can be realized by the LoRa chip (i.e. Semtech's SX127x chip). We tested the LoRa CAD feature and use the dedicated device to constantly perform CAD procedure and an other dedicated device to periodic send messages of various size. Fig. 3 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. In Fig. 3 the red rectangle and green rectangle denote channel active duration and inactive duration respectively, and a blue spot denotes a successful CAD. As can be seen in Fig. 3 the LoRa CAD procedure can correctly detect all the LoRa transmission, and not only the preamble.

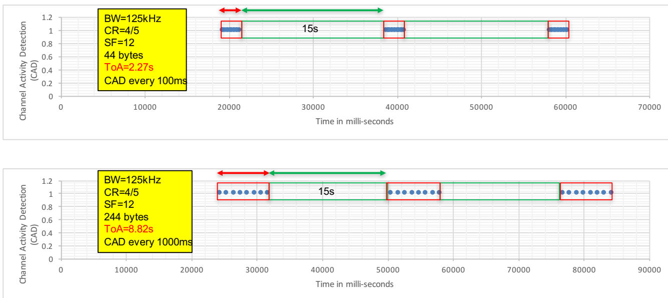

Fig. 3. Test of the LoRa CAD mechanism

Unfortunately, the LoRa CAD procedure's reliability decreases when distance increases. For instance, in our field tests, 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. 4 shows CAD reliability with the same traffic pattern previously shown in Fig. 3 but with the sender device and the device performing CAD separated by 400m with some trees on the line of sight. As can be seen, the CAD procedure fails to detect channel activity many times during a long on-going transmission.

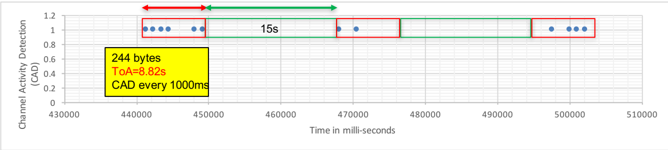

Fig. 4. CAD fails to detect activity of on-going transmissions

B. Limited radio duty-cycle

While the transmission in unlicensed frequency bands is not mandatory nor comes from a technical limitation, both Sigfox and LoRa networks are currently deployed in these unlicensed bands and this situation is most likely not going to change, at least in the next few years – as working in the unlicensed band allows for a much quicker uptake of the technology – although licensed band version may be also deployed in parallel by some operators. However, the flexibility of long-range transmission in the unlicensed bands comes at the cost of stricter legal regulations such as limited duty-cycling, e.g. maximum transmission time per hour.

In Europe, LoRa transmissions is considered as Short Range Devices (SRD) transmission and the ETSI EN300-220-1 regulation [11] applies: transmitters are constrained to 1% duty-cycle (i.e., 36s/hour) total transmission time, regardless of the frequency channel. While this 36s duty-cycle may be large enough for most of deployed IoT applications, monitoring and surveillance applications can have critical information to transmit in a timely manner. After applying all possible optimization mechanisms (e.g. data aggregation, adaptive data rate,...) a device can not simply delay the transmission of critical information nor deliberately violate the regulation.

III. Enabling Image Sensors on LoRa

A. CSMA mechanism for LoRa networks

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 [12], [13], [14]. 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. 5 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. The basic DCF IEEE 802.11 CSMA/CA (Collision Avoidance) works as follows:

- Prior to a transmission, a node determines if there is an on-going transmission by sensing the wireless channel
- After an idle medium during at least a DCF inter-frame space ($DIFS$) the transmission can proceed
- Otherwise, on a busy medium (red $DIFS$), the transmission is deferred and the node waits until the channel is detected idle. Then, after a successful $DIFS$ where channel is idle, the node waits for a random number of backoff time slots in the range $[0, W-1]$.
- As long as the wireless channel is sensed idle, the backoff timer is decreased. It is frozen when a transmission is detected and will resume when the channel is detected as idle again for a $DIFS$
- The node can transmit its packet when timer reaches 0
- Initially W is set to 1. W is doubled for each retry (exponential backoff) until it reaches a maximum value

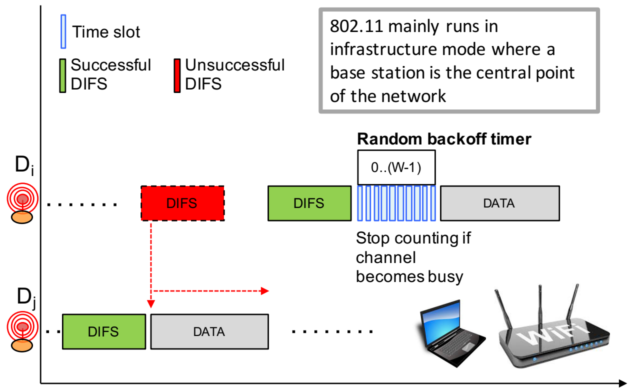

Fig. 5. IEEE 802.11 DCF CSMA/CA

A direct adaptation of this 802.11 CSMA mechanism is illustrated in Fig. 6 where the duration of the LoRa CAD procedure is used to define the time-slot duration and then the value of the $DIFS$ timer.

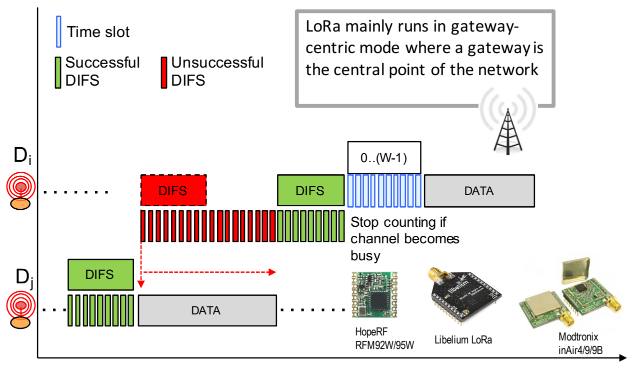

Fig. 6. CSMA mechanism adapted from IEEE 802.11

Here, $DIFS$ is assigned 9 CAD which gives a duration of about $9 \times 61ms = 549ms$ for LoRa mode 1. 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 if 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.

However, the CAD reliability issue raised previously calls for a different approach to prevent collisions because performing CCA during a $DIFS$ can definitely not guarantee that there is not an on-going long transmission in the background. Therefore, the $DIFS$ must be 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. 2). During this extended $DIFS(ToA_{max})$, CAD procedure is performed periodically (for instance every 1000ms as in Fig. 3–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.

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. This new CSMA mechanism adapted to LoRa networks is depicted in Fig. 7.

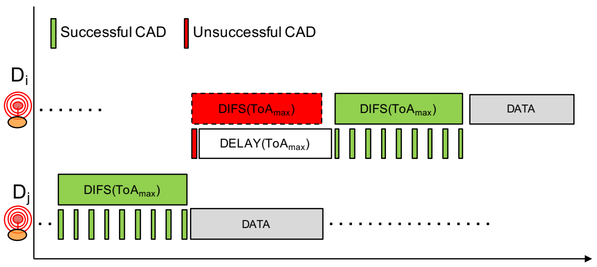

Fig. 7. New CSMA proposition

As the pure delay of ToA_{max} can be implemented by putting the board in deep sleep mode, the proposed CSMA approach is also much more energy efficient compared to the case when the CAD procedure is called constantly to detect the end of the current transmission. The proposed CSMA mechanism has been implemented in our long-range communication library and in all our tests we totally avoids packet losses for both the image sensors and the other devices even when the nodes are hundredth of meters away from each others.

B. Long-range Activity Sharing (LAS)

In countries under duty-cycle regulations for LoRa transmissions, our second control mechanism proposes an advanced radio activity-sharing mechanism allowing some devices (for instance an image sensor that needs to send an image on event detection) to borrow activity time from other devices in order to globally satisfy the duty-cycle requirements. The gateway will manage the total activity time allowed per 1 hour cycle so that each device knows the potential activity time that it can use in this cycle. In this approach, a pool of n devices can use up to a Global Activity Time $G_{AT} = n \times D_{AT}$ per hour, where $D_{AT} = 36000ms$ (according to ETSI regulation of 1% duty-cycle per hour, the value of D_{AT} can be adapted to other duty-cycle regulations). Each individual device can then use up to G_{AT} and will be informed of any changes over the 1-hour period.

1) Packet format: We uses 3 control packet types between the gateway and the end-devices: REG (register), INIT (initialization) and UPDT (update). The first byte, DSP, contains two 4-bit fields for flag indicators and packet type. We illustrate in Fig. 8 the packet format. l^i_{RAT} is device i's (noted D_i) local Remaining Activity Time while l^i_{RAT0} in a REG message is the initial local Remaining Activity Time announced by D_i to the gateway (most of the case $l^i_{RAT0} = D_{AT} = 36000ms$. r^i_{ATU} is D_i's local Remote Activity Time Usage. $|AT^i|$ and $E\{D_k\}$ in an UPDT message are respectively the Activity Time of D_i computed by the gateway and a list of device's id (address).

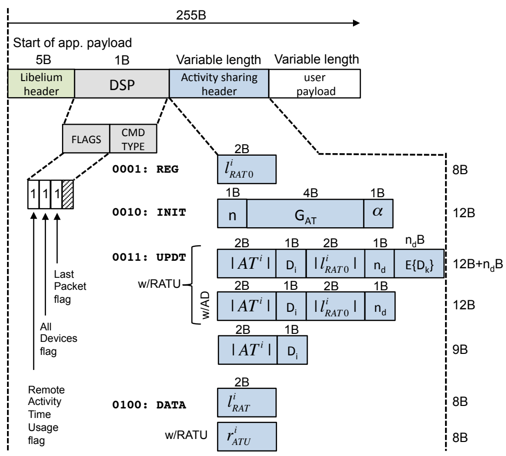

Fig. 8. Packet format

RATU flag can be set in a DATA packet to determine if the packet carries an l^i_{RAT} field or an r^i_{ATU} field. In case the transmission of several packets should be considered as one transaction, the LP flag can be used to indicate that it is the last packet of the serie.

2) Description of the activity time sharing mechanism: Our approach is centralized on the gateway which will update G_{AT} after each packet reception from remote devices and will then broadcast G_{AT} new values. We describe hereafter the various steps of our approach:

1) **Initialization**

   1a) all deployed devices D_i that can share their activity time register (REG packet) to the gateway and indicate their local Remaining Activity Time l^i_{RAT0}. The gateway stores all l^i_{RAT0} in a table (the last l^i_{RAT0} value, noted $lastl^i_{RAT0}$, will also be saved; initially $lastl^i_{RAT0} = l^i_{RAT0}$), computes G_{AT} and broadcasts (INIT packet) both n (the number of devices) and G_{AT}, see Fig. 9(left). This step is realized every hour.

1b) on reception of n and G_{AT} from INIT message each device D_i can consider an initial (and locally managed) $G^i_{AT} = l^i_{RAT0} + \sum_{j=1, j\neq i}^{n} l^j_{RAT0}$, as shown in Fig. 9(right)(a). D_i also sets its local Remaining Activity Time, l^i_{RAT} (the green bar), to l^i_{RAT0} and both its local Total Activity Time, l^i_{TAT}, and its remote Activity Time Usage, r^i_{ATU}, to 0.

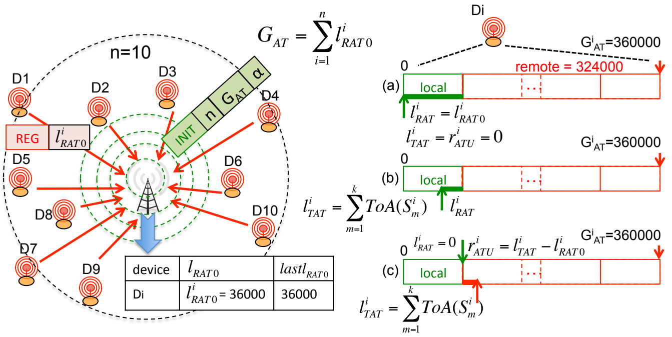

Fig. 9. Left: initialization. Right: device's local and remote activity time

2) **Device D_i wants to send a DATA packet k of size S^i_k**

2a) D_i computes $ToA(S^i_k)$.

2b) if $l^i_{TAT} + ToA(S^i_k) > G^i_{AT}$ then ABORT.

2c) D_i updates $l^i_{TAT} = l^i_{TAT} + ToA(S^i_k)$ and $l^i_{RAT} = l^i_{RAT} - ToA(S^i_k)$, see Fig. 9(right)(b).

2d) if $l^i_{TAT} > l^i_{RAT0}$ then D_i sets $l^i_{RAT} = 0$ and $r^i_{ATU} = l^i_{TAT} - l^i_{RAT0}$ (the red bar), see Fig. 9(right)(c).

2e) if $r^i_{ATU} > 0$ puts r^i_{ATU} in data packet and sets the Remote Activity Time Usage (RATU) flag; otherwise, puts l^i_{RAT} in data packet.

3) **Gateway receives a DATA packet k from D_i of size S^i_k**

3a) gateway computes $ToA(S^i_k)$ and updates for device D_i $l^i_{RAT0} = l^i_{RAT0} - ToA(S^i_k)$.

3b) when last packet or timeout from D_i computes $AT^i = l^i_{RAT0} - lastl^i_{RAT0}$.

3b.1) if $l^i_{RAT0} > 0$, broadcasts an UPDT message indicating $|AT^i|$ and D_i's id.

3b.2) if $l^i_{RAT0} < 0$, determines how many devices, n_d, should be used to distribute the extra activity time consumed by device D_i. Then broadcasts an UPDT message with Remote Activity Time Usage (RATU) flag and advertise $|AT^i|$, D_i, $|l^i_{RAT0}|$, n_d and a list of device's id. If $lastl^i_{RAT0} < 0$ then $|AT^i|$ is replicated in the $|l^i_{RAT0}|$ field as D_i had already consumed all its local activity time. For the selected devices j, the gateway updates their l^j_{RAT0} (stored in the table) accordingly, $l^j_{RAT0} = l^j_{RAT0} - |l^i_{RAT0}|/n_d$, and sets $lastl^j_{RAT0} = l^j_{RAT0}$.

3b.3) if an UPDT message has been sent, the current value of l^i_{RAT0} is saved into $lastl^i_{RAT0}$.

4) **Device D_j receiving an UPDT from gateway**

4a) if $j \neq i$ then D_j updates $G^j_{AT} = G^j_{AT} - |AT^i|$.

5) **Device D_j receiving an UPDT w/RATU from gateway**

5a) if $D_j \in E\{D_k\}$, takes the advertised $|l^i_{RAT0}|$ and updates $l^j_{TAT} = l^j_{TAT} + |l^i_{RAT0}|/n_d$, $l^j_{RAT} = l^j_{RAT} - |l^i_{RAT0}|/n_d$ and $G^j_{AT} = G^j_{AT} - |AT^i| + |l^i_{RAT0}|$ because all D_j in the list of devices contribute to $|l^i_{RAT0}|$.

5b) if $D_{j\neq i} \notin E\{D_k\}$, updates $G^j_{AT} = G^j_{AT} - |AT^i|$ to remove what has been consumed by D_i.

The main work is done by the gateway with action 3.b which determines the Activity Time consumed by a device D_i. While $l^i_{RAT0} > 0$ only local activity time is used so UPDT messages only trigger at $D_{j\neq i}$ action 4.a which only decreases G^j_{AT}. This is illustrated in Fig. 10 where D_4 uses 20896ms of its local activity time. For any device D_k, the amount of time between the green arrow (l^k_{RAT}) and the red arrow (local G^k_{AT}) is the total allowed activity time for that device.

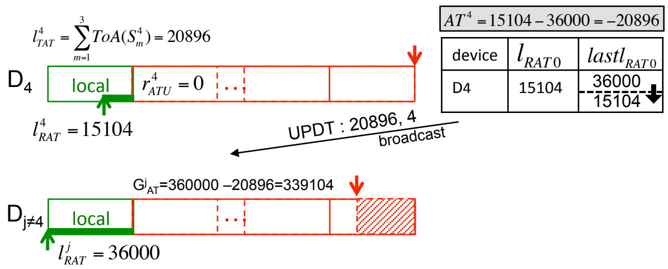

Fig. 10. Local activity time consumption

Upon reception of a packet from a device D_i the gateway keeps track of the consumed activity time. However, the data packet header also includes for device D_i the value of either l^i_{RAT} or r^i_{ATU}, depending on the RATU flag in the packet header, to make the system more robust to packet losses. This is the purpose of action 2e. Although not shown, action 3a also compares l^i_{RAT} indicated in the packet to l^i_{RAT0} stored by the gateway: packet losses would make the 2 values to differ.

In Fig. 11, we illustrate the case where device D_4 continues transmitting, uses all his allowed local activity time, i.e. 36000ms, and also used $r^4_{ATU} = 14942ms$ from the remote activity time pool. Therefore, when the gateway received the last packet from D_4, $AT^4 = -30046ms$ according to action 3b. In the example, the gateway decides to assign to devices D_5 and D_6 the task of taking charge of the extra activity time consumed by D_4. To do so, the UPDT message with the RATU flag starts with the value of $|AT^4| = 30046ms$ followed by D_4's id, $n_d = 2$, $|l^4_{RAT0}| = 14942$ from the table and finally D_5 and D_6 ids. D_5 and D_6 will each remove 7471ms from l^5_{RAT} and l^6_{RAT} respectively. If we assume that both devices did not send any message, then $l^5_{RAT} = l^6_{RAT} = 28529ms$.

They then update their local value of $G_{AT}^{5,6}$ by removing AT^4, but adding $|l_{RAT0}^4|$ because both of them already contributed previously to $|l_{RAT0}^4|$, action 5a. Therefore, at the end, they both have their $G_{AT}^{5,6}$ decreased by D_4's whole allowed local duty-cycle. A device $D_{j \neq i}$ and not appearing in the selected device list will have to remove the totality of what has been consumed from its local value of G_{AT}^j to have a consistent view for G_{AT}, see action 5b.

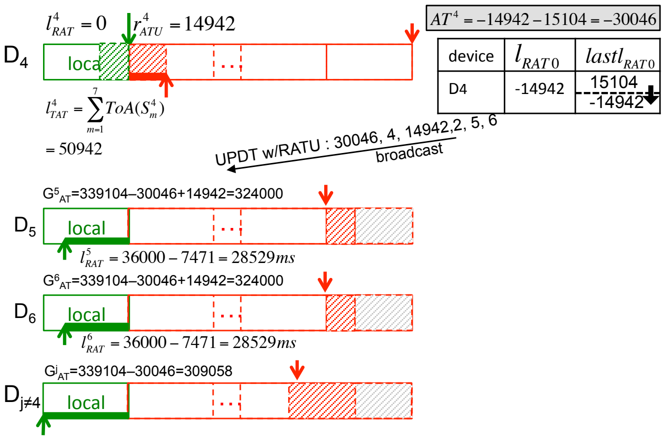

Fig. 11. Remote activity time consumption

3) Implementation of the LAS mechanism: The described LAS mechanism has again been implemented into our communication library used for both Arduino-based end-devices and Linux-based gateways. Real experiments show that when an end-device needs to go beyond the limitation imposed by duty-cycle regulations, the activity-time sharing mechanism can provide an appropriate solution to allow the transmission of critical data.

IV. Conclusions

In this paper we presented 2 control mechanisms to enable the deployment of innovative image sensor devices with long-range LoRa technology: (i) an adapted CSMA mechanism to avoid costly packet collision and (ii) an activity time sharing mechanism to mitigate the issue of duty-cycle.

The first contribution proposes a new CSMA mechanism that can leverage the unreliable Channel Activity Detection mechanism of LoRa. Real experiments on a test-bed with several image sensors and traditional sensors confirm the efficiency of the mechanism to avoid costly packet collisions.

In the second contribution, we proposed an activity time sharing mechanism to provide QoS in the context of radio duty-cycle limitation in scenarios where the devices are deployed by a single organization. The proposed mechanism has been implemented in our long-range communication library and, again, real experiments show the efficiency of the proposed mechanism while keeping the level of signaling trafic low.

Acknowledgments

WAZIUP project has received funding from the European Union'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 ICNSC*, 2015.

[3] Evidence Embedding Technology, "Seed-eye board, a multimedia wsn device", http://rtn.sssup.it/index.php/hardware/seed-eye. Accessed March 8, 2018.

[4] W.-C. Feng, E. Kaiser, W. C. Feng, and M. L. Baillif, "Panoptes: Scalable low-power video sensor networking technologies," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 1, no. 2, pp. 151–167, May 2005.

[5] M. Rahimi et al., "Cyclops: In situ image sensing and interpretation in wireless sensor networks," in *ACM SenSys*, 2005.

[6] S. Paniga, L. Borsani, A. Redondi, M. Tagliasacchi, and M. Cesana, "Experimental evaluation of a video streaming system for wireless multimedia sensor networks," in *10th IEEE/IFIP Med-Hoc-Net*, 2011.

[7] C. Pham, "Low-cost, low-power and long-range image sensor for visual surveillance," in *SMARTOBJECTS'16. Co-located with ACM MobiCom'2016*, 2016.

[8] L. Alliance, "LoRaWAN specification, v1.0.2," 2016.

[9] 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.

[10] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do lora low-power wide-area networks scale?" in *19th ACM MSWiM*, 2016.

[11] ETSI, "Electromagnetic compatibility and radio spectrum matters (ERM); short range devices (SRD); radio equipment to be used in the 25MHz to 1000MHz frequency range with power levels ranging up to 500 mW; part 1: Technical characteristics and test methods," 2012.

[12] M. Kaynia and N. Jindal, "Performance of ALOHA and CSMA in spatially distributed wireless networks," in *IEEE ICC*, 2008.

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

[14] 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.