

Article

# An Orthogonal Air Pollution Monitoring Method (OAPM) Based on LoRaWAN

Rahim Haiahem <sup>1</sup>, Pascale Minet <sup>2</sup> , Selma Boumerdassi <sup>3,\*</sup>  and Leila Azouz Saidane <sup>1</sup>

<sup>1</sup> RAMSIS Team, CRISTAL Laboratory, 2010 Campus University, 2010 Manouba, Tunisia; haiahem.rahim@gmail.com (R.H.); leila.saidane@ensi-uma.tn (L.A.S.)

<sup>2</sup> EVA Project, Inria-Paris, 75012 Paris, France; pascale.minet@inria.fr

<sup>3</sup> CEDRIC/CNAM, 292, Rue Saint Martin, 75003 Paris, France

\* Correspondence: selma.boumerdassi@cnam.fr

Received: 20 July 2020; Accepted: 27 August 2020; Published: 9 September 2020



**Abstract:** High accuracy air pollution monitoring in a smart city requires the deployment of a huge number of sensors in this city. One of the most appropriate wireless technologies expected to support high density deployment is LoRaWAN which belongs to the Low Power Wide Area Network (LPWAN) family and offers long communication range, multi-year battery lifetime and low cost end devices. It has been designed for End Devices (EDs) and applications that need to send small amounts of data a few times per hour. However, a high number of end devices breaks the orthogonality of LoRaWAN transmissions, which was one of the main advantages of LoRaWAN. Hence, network performances are strongly impacted. To solve this problem, we propose a solution called OAPM (Orthogonal Air Pollution Monitoring) which ensures the orthogonality of LoRaWAN transmissions and provides accurate air pollution monitoring. In this paper, we show how to organize EDs into clusters and sub-clusters, assign transmission times to EDs, configure and synchronize them, taking into account the specificities of LoRaWAN and the features of the air pollution monitoring application. Simulation results corroborate the very good behavior of OAPM.

**Keywords:** air pollution monitoring; LoRaWAN; LoRaWAN scalability; LoRaWAN reliability; network synchronization

## 1. Introduction

The Earth's atmosphere contains an ever-growing amount of polluting particles and gases. Human activities are largely responsible for this pollution, in particular automobile traffic, which alone causes a third of all polluting gas emissions [1]. The dispersion and the increase in the concentration of pollutant levels in the air have harmful effects on the environment by increasing crop diseases, soil nutrient depletion, acidification of mineral soil and ground waters, etc. They also have adverse effects on human health by causing cardiovascular diseases, degradation of lung function, systematic inflammation and oxidative stress, asthma, cancer [2–4], etc. Since 1948, the World Health Organization (WHO) has put in place many initiatives to monitor atmospheric pollution. More than 7000 people in 150 international offices work to identify and publish manuals for the regulation and limitation of pollutant emissions [5]. To evaluate the pollution level, the U.S. Environmental Protection Agency (EPA) defined the Air Quality Index (AQI) [6]. The AQI has six levels of air pollution, each level has a dedicated color and an impact on environment and human health. Table 1 gives for each AQI level the health concern represented by a color.

In this paper, we develop a new LoRaWAN-based air pollution monitoring system. LoRaWAN was specified in January 2015 and developed to facilitate Internet-of-Things (IoT) applications. It has been widely adopted and deployed by telecommunications providers (Dutch telecommunications

operator KPN, The Things Network (TTN), etc.), companies, entrepreneurs, and research institutes [7]. This suggests that LoRaWAN has become a strong contender among the set of LPWAN protocols. However, it suffers from a scalability problem as observed for instance by the authors of [8] who measured a drastic decrease of 50% in the Packet Delivery Ratio when the number of devices range over 900. Our purpose is to provide a LoRaWAN-based solution with a measured Packet Delivery Ratio (PDR) supporting a number of IoT devices up to 2000 or 5000 depending on the Monitoring Period value. This solution is based on the scheduling of ED transmissions which considerably limits the collisions which usually cause the PDR degradation.

**Table 1.** Health effects of the Air Quality Index (AQI).

Air Quality Index	Levels of Health Concern	Colors
up to 50	Good	Green
51 to 100	Moderate	Yellow
101 to 150	Unhealthy for Sensitive groups	Orange
151 to 200	Unhealthy	Red
201 to 300	Very Unhealthy	Purple
301 to 500	Hazardous	Brown

Before detailing our solution called OAPM (Orthogonal Air Pollution Monitoring), we first present some background about air pollution monitoring in Section 2 and then some related work in Section 3. Then we provide the necessary background on LoRa and LoRaWAN, justify this choice and present the scalability problem in Section 4. The architecture of OAPM is described in Section 5. The medium activity over time and the computation of transmission times are detailed in Section 6. Section 7 presents the behavior of the Network Server and the End Devices (EDs) in charge of air pollutant monitoring, including the configuration and synchronization of EDs. Simulation results are reported in Section 8 and compared with those of LoRaWAN and ADG—another state-of-the-art solution. Finally, we conclude in Section 9.

## 2. Background

### Background on Air Pollution Monitoring

The control and the monitoring of major air pollutants ( $CO$ ,  $NO_2$ ,  $SO_2$ ,  $O_3$ ,  $PM_{2.5}$ ,  $PM_{10}$ ) has now become feasible at low cost and in real-time. Instead of using expensive static monitoring stations that provide limited spatio-temporal air pollution information, tiny and low-cost sensors are deployed everywhere in the urban environment to sense and transmit air pollution concentrations to the back-end system using wireless network technologies.

Six major pollutants have been identified as the causes of air pollution. They are: Carbon Monoxide  $CO$ , Nitrogen Dioxide  $NO_2$ , Sulfur Dioxide  $SO_2$ , Ozone  $O_3$ , Particulate Matter  $PM_{2.5}$  and Particulate Matter  $PM_{10}$ . The observation period and the unit of measure for each of these pollutants are given in Table 2. Note that ppm and ppb stand for part per million and part per billion, respectively.

For each pollutant  $p$  among the six major pollutants identified, the associated index  $I_p$  is computed according to Equation (1).

$$I_p = \frac{I_{Hi} - I_{Lo}}{BP_{Hi} - BP_{Lo}} (C_p - BP_{Lo}) + I_{Lo} \tag{1}$$

where  $C_p$  is the observed concentration of pollutant  $p$  in an observation period whose length is given in Table 2.

$BP_{Lo}$  and  $BP_{Hi}$  denote the lowest breakpoint and the highest breakpoint defined in the Guidance on the Air Quality Index produced by the EPA [9] such that  $BP_{Lo} \leq C_p \leq BP_{Hi}$ . These breakpoints are given in Table 2 for each major air pollutant.

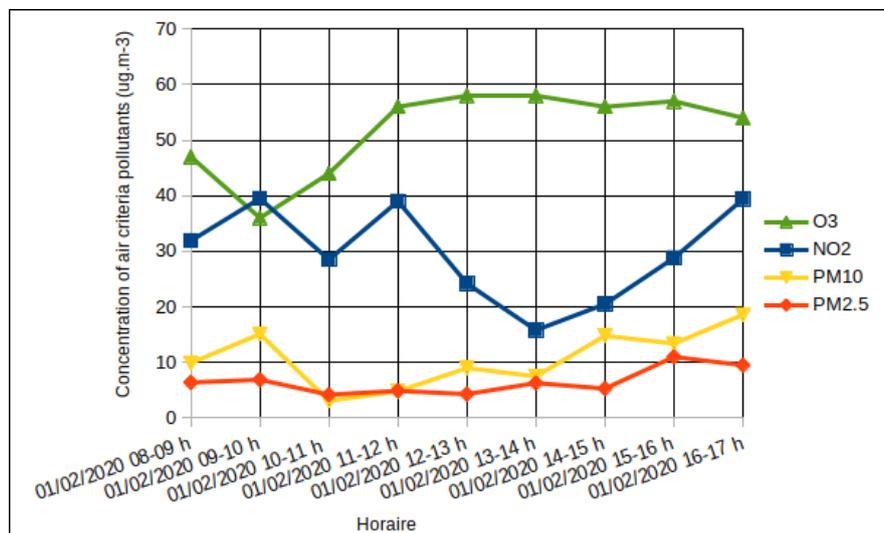
$I_{Hi}$  denotes the AQI value for  $BP_{Hi}$  and  $I_{Lo}$  the AQI value for  $BP_{Lo}$ : see the standardized values of these parameters in the first column of Table 2.

**Table 2.** Unit of measure, observation period and breakpoints for major air pollutants used in AQI computation.

AQI	O <sub>3</sub>		PM <sub>2.5</sub>	PM <sub>10</sub>	CO	SO <sub>2</sub>	NO <sub>2</sub>
	ppm 8 h	ppm 1 h	µg/m <sup>3</sup> 24 h	µg/m <sup>3</sup> 24 h	ppm 8 h	ppb 1 h	ppb 1 h
Good (≤50)	0–0.059		0–12.0	0–54	0–4.4	0–35	0–53
Moderate (51–100)	0.060–0.075		12.1–35.4	55–154	4.5–9.4	36–75	54–100
Unhealthy for Sensitive Groups (101–150)	0.076–0.095	0.125–0.164	35.5–55.4	155–254	9.5–12.4	76–185	101–360
Unhealthy (151–200)	0.096–0.115	0.165–0.204	55.5–150.4	255–354	12.5–15.4	186–304	361–649
Very Unhealthy (201–300)	0.116–0.374	0.205–0.404	150.5–250.4	355–424	15.5–30.4	305–604 [24 h]	650–1249
Hazardous (301–500)	-	0.405–0.604	250.5–500.4	425–604	30.5–50.4	604–1004 [24 h]	1250–2049

Finally, the AQI takes the maximum value of  $I_p$  among all pollutants and the pollutant  $p$  providing this maximum value is considered to be responsible for air pollution.

An example of the evolution of air pollutant concentration over eight consecutive hours in a day in a big city is depicted in Figure 1 according to the Airparif measurement [10] performed in Paris and its suburb the first February 2020. We can observe that the air pollutant concentration increases at 10:00 a.m., 12:00 a.m., 5:00 p.m. and decreases or stabilizes in the rest of the observed time. This fact confirms that urban traffic (i.e., motor vehicle emissions) in peak hours is one of the major sources of air pollution.



**Figure 1.** Evolution of pollutant concentration over time.

### 3. Related Work

In this section, different air monitoring systems are presented first. We then describe more particularly LoRa based solutions ensuring monitoring.

#### 3.1. Air Pollution Monitoring Solutions

In [11], the authors present a customized design of an IoT enabled environment monitoring system to monitor CO<sub>2</sub>, temperature, humidity. The system consists of a receiver node which forwards the wirelessly received data sent from a transmitter node to a personal computer. Data is depicted graphically through a customized graphical user interface and the PHP API execution on the Internet

enables the transfer of this data to an Android-based smartphone. The experimentation was deployed at several places at the institute (DA-IICT) and around the Gandhinagar city in India under varying environmental conditions.

In [12], the authors propose an IoT-based atmospheric environment monitoring system which can effectively observe air pollution information without the restriction of place or space. The system consists of atmospheric environment measurement devices, an atmospheric environment analyzer, and a user application. The devices send TCP packets containing air pollution information using the LTE network. The analyzer performs error verification and the user application provides the user with the observation results. The system provides a large coverage of the environment, but adopting a cellular network is expensive for air pollution monitoring. Unlike [11], where the system focused only on measurements of one air pollutant ( $CO_2$ ), this system measures various types of air environment information including fine dust, ozone,  $SO_2$ , etc.

In [13], the authors propose a real-time air pollution index measurement platform using a 5G wireless network and blockchain technology. The overall architecture is composed of five layers consisting of the perception layer, data processing layer, data preservation layer, application layer, and 5G wireless network layer. These layers are responsible for processing, preservation, and management of the data collected via IoT sensors and transmitted to the edge computing nodes through a 5G wireless network. To prevent forgery and tampering, the extracted results are transferred to the cloud and the application uses blockchain encryption technology.

In [14], an IoT kit for air contaminant measurement is developed. This kit consists of gas sensors, Arduino IDE (Integrated Development Environment), and WiFi module. The gas sensors gather data from the air and forward the data to the Arduino IDE. The latter transmits the data to the cloud via the WiFi module. Furthermore, an Android application, called IoT-Mobair, has been developed which enables users to access relevant air quality data. In addition, it predicts and displays on Google maps the pollution level of the entire route if a user is traveling to some specific location.

In [15], each LoRaWAN end device in charge of monitoring the air in a smart city is equipped with a battery and a solar panel for energy harvesting. To save energy consumption, the activity of the LoRaWAN network is limited to some predefined time slots per day.

All these papers agree on the fact that for a more accurate air pollution monitoring, many IoT devices sensing the air pollution are preferred to a single sophisticated monitoring station. Furthermore, many of them aimed at demonstrating the feasibility of IoT-based solutions by means of a prototype. The prototype was made up of a limited number of IoT devices. Our purpose is to provide a solution where the reliability of LoRaWAN, measured by the Packet Delivery Ratio (PDR), does not fall while supporting a number of devices up to 2000 or 5000 depending on the value of the Monitoring Period.

### 3.2. LoRa-Based Solutions

We now focus more particularly on LoRa based solutions ensuring monitoring. Many authors observed that the Packet Delivery Ratio (PDR) of LoRaWAN collapses, when the number of End Devices (EDs) increases. This is due to a high number of collisions that the Aloha medium access protocol neither prevents nor avoids. Some authors try to make these collisions constructive instead of destructive, like for instance Choir [16] or QuAiL [17] that rely on the linear addition of powers of phase-asynchronous channels in the air. Others use specific hardware to successfully decode weak transmissions at the GW, like Charm [18]. Our paper does not belong to this category. It aims at proposing a solution to strongly limit or avoid collisions.

Since the clocks of devices drift over time, all the solutions avoiding collisions require synchronized devices to work correctly. In [19], the authors introduce a time synchronization service for low-cost IoT devices connected to a gateway that uses the LoRaWAN protocol. The service is scheduled when any ED transmits a confirmed uplink frame and registers the timestamp at the end of its transmission. The GW considers its own timestamp as the reference and sends a timestamped acknowledgement in the RX1 receive window. Thus, the ED has all the information needed for clock

re-alignment. To verify the overall functionality, authors developed and deployed a LoRaWAN testbed in real-life conditions. Furthermore, the obtained results demonstrate that regulating the access to the medium within slots (Slotted Aloha) provides a better reliability than the standard Aloha MAC in real-life deployments.

LongShoT [20] provides on-demand time synchronization in one-shot for devices close to the GW and in two-shot for long range devices. The authors recommend the use of a 5-min periodic synchronization to keep the devices synchronized within  $\pm 100 \mu\text{s}$  of the reference time. LongShoT uses less energy than GPS for synchronization intervals longer than 50 s between synchronization requests.

In [21], the authors designed a low-overhead fine-grained synchronization and scheduling scheme for LoRaWAN networks, where time slots are assigned to End Devices (EDs) based on their traffic needs. In this system, EDs infrequently trigger the synchronization and scheduling method by sending a signaling message to the Network Synchronization and Scheduling Entity (NSSE) within the Network Server. This entity encodes time slot indexes when the EDs are allowed to transmit and inserts them in a probabilistic data structure using Bloom filters. This reduces the size of the messages that are needed to perform the synchronization and scheduling. The authors demonstrate that the proposed algorithm outperforms standard Aloha. Furthermore, synchronized LoRaWAN networks ensure better reliability than the un-synchronized ones. However, the per node basis synchronization and scheduling scheme consumes duty cycle limitations of the gateway in downlink and does not scale for a high number of devices, or even a moderate number of devices but with high SFs.

In [22], a time slotted LoRaWAN solution is proposed, where each End Device deduces its time slot from its own address and the frame length. The GW uses the last slot of the frame to synchronize the EDs and to group the acknowledgments of all the transmissions sent in this frame. The frame length is adjusted to take into account all the EDs having joined the LoRaWAN network.

In [23], to allow the use of LoRaWAN in machine vibration monitoring, a high-accuracy synchronization of devices is provided. Each End Device computes its clock drift and the average clock drift on each channel taking into account the last two synchronization messages received from the GW. Before transmitting, each ED applies a compensation value proportional to the sampling period and the difference between its clock drift and the average value. In addition, NB-IoT is used to transmit the monitoring reports collected by the GW to the cloud. This solution ensures that two devices are synchronized within  $5 \mu\text{s}$ .

None of the aforementioned synchronization algorithm meets all our requirements which are:

- A simple algorithm, where the processing performed by the EDs is kept minimum.
- A periodic algorithm, at the initiative of the GW. No ED has to take the initiative of requesting to be synchronized with the GW's clock.
- An efficient algorithm able to synchronize all EDs at the same time, even if they use different spreading factors.

In [24], the authors propose an Alternative Data Gathering (ADG) of air pollutants using LoRaWAN. ADG divides the area of interest into subareas and assigns to each subarea a Time Window (TW). Each sensor transmits its air pollution reports using a Spreading Factor (SF) depending on its distance to the gateway, using a random frequency channel and a random transmission time chosen within TW. Simulation results obtained by ADG show that when the number of nodes increases or the TW duration becomes shorter, the probability that a high number of devices schedule their transmissions simultaneously on the same frequency channel and using the same SF increases. This fact leads to possible intra-SF collisions, leading to a strong degradation of the network performances.

To avoid a strong decrease of reliability when the number of EDs increases, we propose a new solution that aims at ensuring a total orthogonality between transmissions and avoiding collisions. Our algorithm is implemented over the NS3 simulation tool and the results obtained are compared with those found in ADG and regular LoRaWAN based on pure Aloha.

#### 4. LoRa and LoRaWAN

Before recalling the main features of LoRa and LoRaWAN, we justify the choice of this network technology for Air Pollution Monitoring. Indeed, in air pollution monitoring, the choice of the network technology is a very important issue, especially with regard to (i) coverage area and (ii) network lifetime. Short communication range technologies (e.g., ZigBee) applied to air pollution monitoring require multi-hop communications in order to increase the coverage area. However, this is obtained at the cost of a decrease in network lifetime [25]. Cellular networks such as GSM, LTE, and 3G provide a very good coverage area [25], but due to the licensed frequency band and the transmission rate of pollution reports, every message sent must be paid for, which may result in an expensive monitoring system [26]. Notice that the cost should be taken into account for any operator-based network solution.

Today, a new family of networks called Low-Power WANs (LPWANs) is gaining great importance. This family is located between (i) short-range and high-bandwidth networks and (ii) cellular networks that improve large coverage, but also high power consumption [27,28]. LoRaWAN, Sigfox, and NB-IoT are examples of LPWANs that enable long communication range and operate for long periods [28]. Furthermore, as the Internet-of-Things will include a huge number of operated devices, an additional requirement for air pollution monitoring is the use of low-cost wireless transceivers.

Table 3 compares different network technologies in terms of throughput, communication range [24,29], topology, battery lifetime, where a higher number of ‘+’ denotes a higher lifetime of battery-operated end devices. Taking into account the features of the air pollution monitoring application, namely its data rate, its monitoring area that ranges over 5 km and a high network lifetime requested, the LoRaWAN solution is the most attracting.

**Table 3.** Comparison of different network technologies.

	Battery Lifetime	Throughput	Com. Range	Topology	Operator/Autonomic
Wi-Fi	+	1.3 Gbits/s	<100 m	Star	Autonomic
Zigbee	++++	250 kbit/s	<300 m	Star, Mesh, Point-to-Point	Autonomic
NB-IoT	+++	250 kbit/s	>30 km	Cellular	Operator
Sigfox	++++	800 bits/s	>10 km	Cellular	Operator
LoRaWAN	++++	250 to 5470 bit/s	>10 km	Star	Both

##### 4.1. LoRa

LoRa is the physical layer used to create a long-range communication link [30]. It is based on the chirp spread spectrum (CSS) modulation [31], which has been used in military and space communication for decades due to the technology’s long-range capability and robustness to interference. The Spreading Factor (SF) determines the chirp duration with  $2^{SF} = B \times T$ , where  $B$  is the spread bandwidth expressed in kHz and  $T$  the chirp duration in seconds. In LoRa, SF7 gives the shortest time on air, and SF12 the longest. According to this formula, an increase of one in the spreading factor doubles the message time on air, which also depends on packet encoding. As a consequence, the higher the SF is, the longer the message time on air, the longer the communication range and also the more reliable and the more energy consuming the message reception is. The bitrates associated with these spreading factors [32] for the 125 kHz mode are given in Table 4. The values given in Table 4 come from the datasheet of the Semtech SX1301 product [32].

**Table 4.** Spreading factors and bitrates.

	SF7	SF8	SF9	SF10	SF11	SF12
Bitrate (kbps)	5.469	3.125	1.758	0.977	0.537	0.293

The main advantage in the CSS modulation lies in the possibility of correct demodulation of overlapping transmissions [33].

#### 4.2. LoRaWAN

LoRaWAN™ defines the communication protocol and system architecture for a network based on LoRa. LoRaWAN specifications allow network characteristics such as battery lifetime of a node, network capacity, quality of service and security [34] to be determined. The architecture distinguishes three main types of devices: Network Server (NS), Gateways (GWs) and End Devices (EDs). Furthermore, it defines three different classes of EDs with the following features:

- Class A devices, with the basic set of features that all devices must implement. To listen to the downlink messages, EDs open two receive windows (RX1, RX2) at a predefined time after the end of transmission of an uplink message. EDs are prohibited from starting a new uplink transmission before the end of the second receiving window. Class A is the lowest power consumption class [35]. It is also called the default class. Using LoRaWAN MAC commands, an ED of Class A can change its status to Class B.
- Class B devices implement the functionalities of Class A. In addition, these devices are also accessible at time slots defined by Beacon messages multicast by the gateway periodically every 128 s [36].
- Class C devices implement the functionalities of Class A. Class C is dedicated to Continuously listening end devices which are less power-constrained devices. Gateways can, therefore, send any downlink transmission at any time. EDs of this class are accessible with low-latency but consume more energy than EDs of other classes [37].

EDs communicate in asynchronous mode and transmit their messages according to the Aloha method. In LoRaWAN, any ED has a direct link with the GW. This strongly contributes to reducing node energy consumption [38]. Packets transmitted by EDs are typically received by one or multiple GWs and the latter forwards the received packet to the cloud-based network server via standard IP connections (Cellular, Ethernet, or Wi-Fi). As the Internet of Things is expected to involve a very high number of end devices, GWs must have the capability to receive messages from all the operated end nodes and provide them with a good quality of service. This is achieved in the LoRaWAN network by utilizing, on the one hand, adaptive data rates (ADR) and, on the other hand, a multi-channel transceiver in the gateway able to receive simultaneous messages on multiple channels. Commercial LoRa radio chipsets allow eight parallel receive paths for GWs. Each receive path is assigned to a frequency channel and is able to lock on one signal and decode it, whatever its SF. As a consequence, two transmissions with different SFs on the same channel are orthogonal provided that there are at least two receive paths assigned to this channel [33,39].

Finally, the NS is in charge of the complex tasks such as filtering redundant received packets, performing security checks, scheduling acknowledgments and downlink traffic through the optimal gateway, as well as adapting data rates, etc. [34].

#### 4.3. Frequency Bands and European Regulations

LoRaWANs operate on the Industrial, Scientific, and Medical (ISM) frequency band. In Europe, the LoRa network uses the 863–870 MHz band which is divided into three different categories:

- Sub-band 1: h1 (867–868 MHz), h1.1 (868–868.6 MHz), h1.4 (869.7–870 MHz).
- Sub-band 2: h1.2 (868.7–869.2 MHz).
- Sub-band 3: h1.3: (869.4–869.65 MHz).

The ETSI specified regulations and limitations that must be respected by each transmission in any sub-band. Indeed, sub-band 1 has a 1% duty cycle to be shared between all sub-channels and an Effective Radiated Power (ERP) limited to 14 dBm [34,40]. The regulations provided by [41] are summarized in Table 5.

**Table 5.** LoRaWAN default channels and duty cycle limitations.

Channel	Central Frequency (MHz)	Bandwidth	% of Time on Air	Max ERP	Regulatory Regime
1	868.1	125 kHz	1%	14 dBm	h1.1
2	868.3	125 kHz	1%	14 dBm	h1.1
3	868.5	125 kHz	1%	14 dBm	h1.1
4	868.85	125 kHz	0.1%	14 dBm	h1.2
5	869.05	125 kHz	0.1%	14 dBm	h1.2
6	869.525	125 kHz	10%	27 dBm	h1.3

#### 4.4. LoRaWAN Collisions

LoRaWAN packets can be lost due to the propagation loss when devices are far from the gateway. Packet losses may also be caused by collisions. There are two main reasons for collisions:

1. Intra-SF collisions: Whenever several packets arrive at a gateway within the same frequency and time. If these packets were sent using the same SF, all of them will be rejected if one is not received with a signal strength 6 dBm higher than the concurrent packets, taking into account the capture effect.
2. Collisions because of unavailability of a Receive Path: Whenever the number of packets simultaneously arriving at a gateway with different SFs on the same frequency, is strictly higher than the number of Receive Paths available for uplink traffic, only the number of packets corresponding to the number of Receive Paths available will be correctly received.

The collision phenomenon will occur more frequently when the packet transmission rate or the number of end devices increases, leading to the scalability problem of LoRaWAN, that is addressed in this paper. Moreover, as LoRaWAN gateways work in half-duplex, no frame can be received when gateways either transmit a high number of acknowledgments (ACKs) requested by a high number of nodes, or forward a high number of downlink messages sent from the server. In this paper, we consider only Unconfirmed uplink messages which do not require ACK from the server.

## 5. Proposed Architecture

The coverage and monitoring of a urban environment implies deploying thousands of end devices in the region of interest. When the number of EDs grows per gateway, the LoRaWAN Medium Access Control method (i.e., Aloha) cannot efficiently solve the channel access contention. This causes different types of packet losses either due to Intra-SF collision or to the unavailability of GW Receive Paths. This study aims at (i) avoiding collisions between ED transmissions and (ii) ensuring network scalability. The idea behind our solution consists in (1) dividing the zone of interest into clusters and then dividing these clusters into sub-clusters and (2) assigning transmission times to end devices in such a way that they avoid collisions.

The solution proposed is an alternative medium access strategy for LoRa-based devices. It includes an additional synchronization mechanism, different from that specified in the standard. The system architecture we propose is based on the Network Server, Gateways and EDs of LoRaWAN. The air pollution monitoring application determines the number and the location of all the end devices used to monitor the air pollutants. We assume that these EDs are LoRaWAN Class A devices, which have by far the longest battery lifetime and are the most commonly deployed today [42]. We also assume that the GW is installed at the center of the monitoring zone. The Network Server assigns to each ED its Spreading Factor (SF). This SF value is chosen to ensure the best reconstruction of the received signal. It depends on the distance of the ED considered to the GW. Unlike LoRaWAN, EDs are grouped into sub-clusters, which are themselves grouped into clusters, as depicted in Figure 2. For example, Figure 2 shows a monitoring area consisting of four clusters. Devices that are nearer to the GW,

represented in blue, communicate using SF7, whereas more distant devices, represented in brown, use SF12.

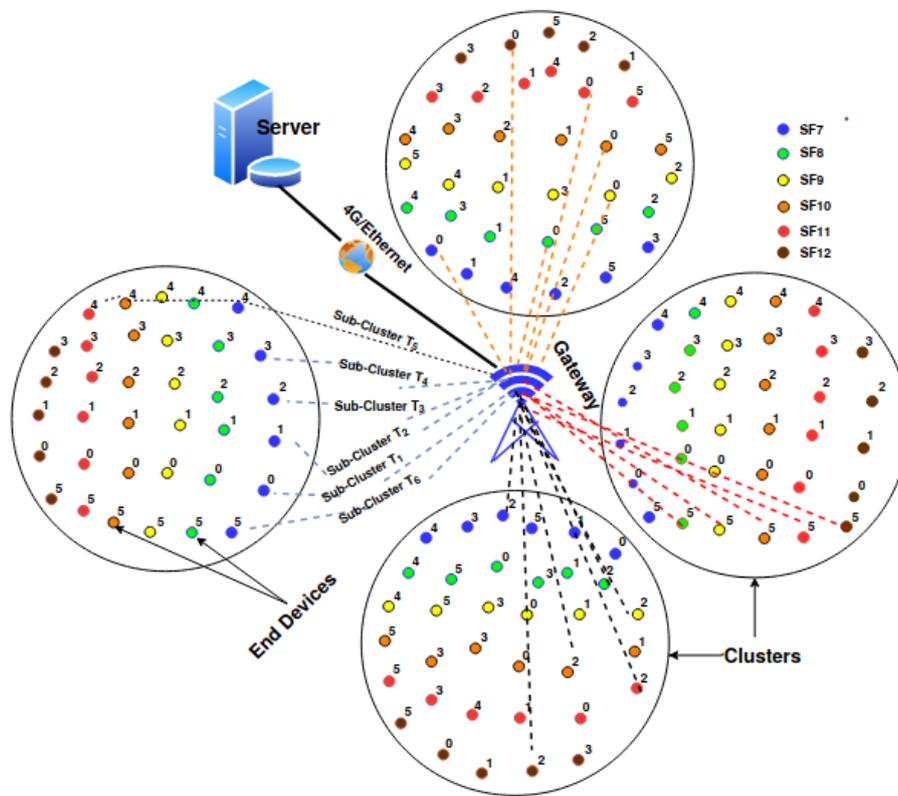


Figure 2. Example of OAPM architecture with 4 clusters in the monitoring area.

### 5.1. System Components

#### 5.1.1. Cluster Formation

Clustering has been widely applied in wireless networks, taking into account various constraints such as maximum number of members per cluster, maximum distance to the cluster head, residual energy of nodes, connectivity degree of nodes, etc. However, the purpose here is different. Cluster formation should be centralized and with no complexity in the EDs. In addition, all the clusters should have the same distribution of spreading factors used by the EDs. Since the spreading factor used by an ED is determined by its distance to the GW, we opted for an implicit cluster formation, which is a direct result from the deployment of EDs: the cluster to which an ED belongs is deduced from its geographic coordinates obtained during its deployment. The monitoring area is divided into angular sectors centered at the GW and with equivalent density (i.e., about the same number of EDs). Each sector contains EDs with different SFs, because they are at different distances to the GW. Each sector is called a cluster. As a result, each cluster consists of EDs that meet the following requirements:

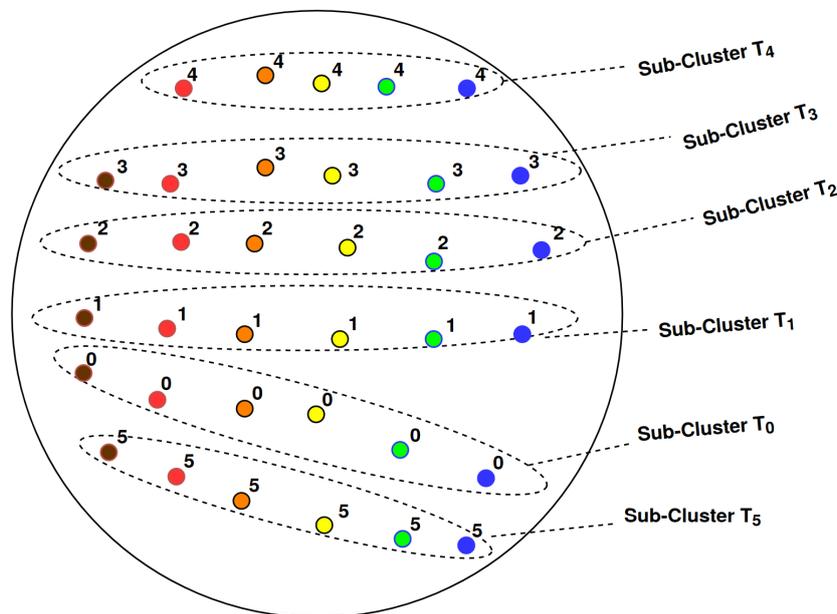
- All clusters have a similar number of EDs.
- Each cluster tries to maximize the number of different SFs present. This requirement is ensured by the breakdown into sectors.
- Each cluster  $j$  is assigned a Time Window  $TW_j$  such that all EDs belonging to cluster  $j$  can transmit their monitoring report to the GW in  $TW_j$ . In each Monitoring Period  $MP$ , there exists one Time Window assigned to each cluster, and the Time Windows assigned to two different clusters do not overlap.

### 5.1.2. Transmission Index

Any ED  $i$  belonging to any cluster  $j$  is assigned a transmission index that gives the rank of transmission of  $i$  within its cluster. The role of this index is to avoid the case where two EDs with the same SF transmit simultaneously. The number shown on each ED in Figure 2 is its transmission index.

### 5.1.3. Sub-Cluster Formation

A sub-cluster contains a subset of nodes of the same cluster with different SFs. As depicted in Figure 3, EDs having the same index  $i$  within a Cluster belong to the same sub-cluster  $i$ . Each sub-cluster consists of at most six EDs with different SFs (i.e., different colors in Figure 2). Since there are no two EDs located in the same sub-cluster with the same SF parameter value, all the EDs within the same sub-cluster are allowed by OAPM to transmit simultaneously. Furthermore, the well-established rank (Index) between sub-clusters of a same cluster decreases the probability of the non-availability of Receive Paths on the gateway.



**Figure 3.** Decomposition of a cluster into sub-clusters.

## 5.2. Time Constraints

We now derive the time constraints from the time requirements of the air pollution monitoring application. The US EPA defined the necessary observation periods to measure the AQI for each pollutant. As presented in Table 2, measuring the AQI of  $SO_2$  and  $NO_2$  for example, requires having the average concentration observed in one hour (the lowest observation period). To calculate the average concentration of a pollutant, the server needs to know several values measured in the hour considered. For this reason, we split the time (e.g., each hour) into several Monitoring Periods ( $MPs$ ) and each  $MP$  into multiple Time Windows ( $TWs$ ), and associate one  $TW$  per cluster. All the EDs of the same cluster transmit within the  $TW$  of their cluster. These different splittings are depicted in Figure 4.

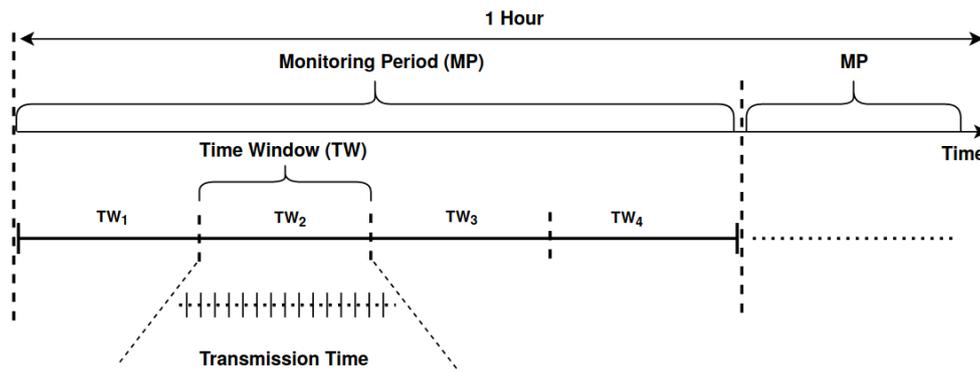


Figure 4. Monitoring period composition.

## 6. System Behavior

In OAPM, we distinguish the following four phases in the life of any ED:

- *Joining phase* during which the ED joins the LoRaWAN Network and gets a short configuration index on two bytes, the channel to listen for the configuration parameters and the SF to use for uplink transmission, which is also the SF used to receive the configuration parameters. The ED also gets the reference time given by the GW.
- *Configuration phase* during which the GW configures all the EDs using their configuration index, telling each ED when to transmit its monitoring report and when to receive the synchronization message from the GW, on which frequency channel and with which SF: see Algorithm 1 in Section 7. To reduce the size of configuration messages and the duration of the configuration phase, the GW multicasts its configuration messages to all EDs using the same SF, configuring at once as many EDs simultaneously configured is computed from the maximum payload allowed with the SF used.
- *Synchronization phase* during which the GW synchronizes all the EDs. Only one synchronization message is multicast to all EDs per Synchronization Period. This message uses the highest SF used by some EDs in the network considered.
- *Monitoring phase* during which each ED monitors air pollutants and transmits its monitoring report to the gateway once per Monitoring Period: see Algorithm 2 in Section 7.

We assume the existence of a function in LoRaWAN allowing any ED to enter the listening mode on a given channel with a given SF and at a given time. This can be seen as an evolution of the second reception time window opening for class A devices, but at a time predefined in the ED configuration, and without wasting energy by listening in the first reception window and then in the second one after each uplink transmission. Hence, this extension is less energy consuming than using class B or class C devices.

### 6.1. Medium Activity over Time

The medium activity over time is ruled by Figure 5. The fields Synchronization Reserved and Monitoring Reserved denote the time reserved for the transmission of the Synchronization message and the Monitoring message, respectively, whereas three guard times are introduced to avoid overlapping transmissions, assuming that the synchronization algorithm maintains the clock offset of any ED  $\leq \Delta$  with regard to the GW clock.

The three guard times introduced are:

- MG1 the monitoring guard that avoids overlapping between a previous downlink synchronization message and an uplink transmission.

$$MG1 = \Delta + \max\_downpropagation\_delay \tag{2}$$

- MG2 the monitoring guard that avoids overlapping between two successive uplink transmissions.

$$MG2 = 2\Delta + \max\_uppropagation\_delay \tag{3}$$

- SG the synchronization guard that avoids overlapping between a previous uplink transmission and the downlink synchronization message.

$$SG = \Delta + \max\_uppropagation\_delay \tag{4}$$

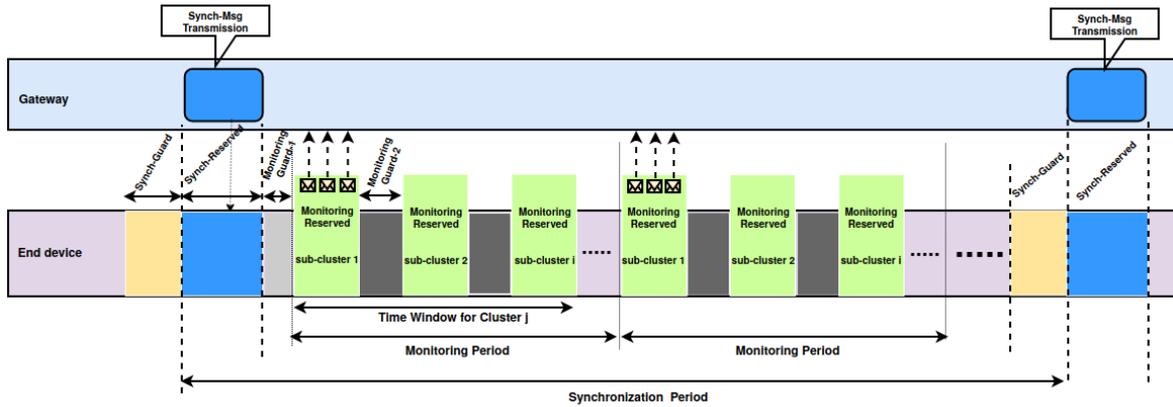


Figure 5. Medium activity over time.

### 6.2. Transmission Time Computation

We focus on any Synchronization Period delimited by two successive synchronization messages and express all the times relatively to the beginning of the entity including it. The including entities are from the greatest to the smallest: the Synchronization Period  $SP$ , the Monitoring Period  $MP$  and the Time Window  $TW$ , as depicted in Figure 5.

Semtech Corporation is the proprietary of the LoRa technology and the only responsible for developing chips implementing the patented LoRa modulation. In order to understand and evaluate the performances of the SX1272, 1273, 1276, 1277 chips, Semtech developed a free program called LoRa calculator. The program allows us to evaluate basic performances, including link budget, time on air and data rate ...etc, according to the selected modulation and packet parameters. These values are compliant with the Standard and the SX1272 datasheet.

The  $i$ th Synchronization Period starts at time  $SP_i = SP\_Start + (i - 1)SP$ .

Within this Synchronization Period, the  $j$ th Monitoring Period starts at time  $MP_j = Sync + MG1 + (j - 1)MP$ , expressed relatively to  $SP_i$ .

The  $k$ th time window in this Monitoring Period starts at time  $TW_k = TW_{k-1} + \sum_{h=1}^{nsub_{k-1}} ToA_s + MG2 * nsub_{k-1}$ , expressed relatively to  $MP_j$ , where  $nsub_{k-1}$  denotes the number of sub-clusters in cluster  $k - 1$ , and  $ToA_h$  is the total Time on Air used by sub-cluster  $h$  to transmit the Monitoring Reports of its devices.

The  $h$ th sub-cluster in the cluster associated with this time window transmits at time  $TT_h = \sum_{s=1}^{h-1} ToA_s + (h - 1) * MG2$ , expressed relatively to  $TW_k$ .

The last Monitoring Period is followed by  $SG$  which precedes the transmission of the synchronization message.

We now express the constraints related to the transmissions. The reader is invited to refer to Figure 5 for an illustration of the following equations. In each Monitoring Period, the last transmission

of the last sub-cluster of the last cluster must be received before the first transmission of the next Monitoring Period starts, leading to:

$$\sum_{h=1}^{nsub} ToA_h + nsub MG2 \leq MP, \quad (5)$$

where  $nsub$  denotes the total number of sub-clusters in the monitoring area and  $ToA_h$  is the maximum transmission duration on air of sub-cluster  $h$ .

In each Synchronization Period, the last transmission of the last sub-cluster of the last cluster in the last Monitoring Period must be received before the transmission of the next synchronization message starts.

$$MP_1 + (nMPperSP - 1)MP + \sum_{h=1}^{nsub} ToA_h + (nsub - 1)MG2 + SG \leq SP, \quad (6)$$

where  $nMPperSP$  denotes the number of Monitoring Periods per Synchronization Period.

### 6.3. Scalability

To evaluate the scalability of OAPM, we compute the maximum number of EDs supported by OAPM. For that purpose, we adopt some additional assumptions:

**Assumption 1.** All SFs in the interval  $[minSF, maxSF]$  are present in the monitoring area, and these SFs are uniformly distributed over all the EDs.

With Assumption 1, the maximum number of EDs supported by OAPM is computed from Equations (5) and (6) as follows:

$$MED = (maxSF - minSF + 1) \min \left( \left\lfloor \frac{MP}{ToA(Rep_{max}) + MG2} \right\rfloor, \left\lfloor \frac{SP + MG2 - SG - Sync_{max} - MG1 - (nMPperSP - 1)MP}{ToA(Rep_{max}) + MG2} \right\rfloor \right) \quad (7)$$

where  $Sync_{max}$  and  $Rep_{max}$  denotes the Time on Air of the Synchronization message and the Monitoring message, respectively; both are transmitted with  $SF = maxSF$ .

**Assumption 2.** All clusters have the same Time Window size, which is enforced by the application.

In the specific case of Assumptions 1 and 2, the maximum number of EDs supported must also ensure that the last transmission of the last sub-cluster of any cluster different from the last one must be received before the first transmission of the next cluster starts, leading to:

$$\sum_{h=1}^{nsubpercluster} ToA_h + nsubpercluster MG2 \leq TW, \quad (8)$$

where  $nsubpercluster$  denotes the number of sub-clusters in a cluster.

With Assumptions 1 and 2, the maximum number EDs must meet the following equation, in addition to Equation (7):

$$MED \leq \left\lfloor \frac{MP}{TW} \right\rfloor (maxSF - minSF + 1) \left\lfloor \frac{TW}{ToA_{max} + MG2} \right\rfloor, \quad (9)$$

where  $\lfloor \frac{MP}{TW} \rfloor$  denotes the maximum number of clusters supported, and  $ToA_{max}$  is the transmission time of a monitoring message using  $SF = maxSF$ .

Table 6 gives the maximum number of EDs supported by OAP for different values of  $MP$  and different values of  $SF$ , assuming that each ED remains synchronized  $\pm\Delta = 1$  ms to the GW’s clock,  $SP = 1602$  s, the maximum propagation delay is 18  $\mu$ s corresponding to a range of 6 km around the GW,  $minSF = 7$  and  $maxSF$  ranges from 7 to 12. We consider two different cases:

- The transmission window size is free: Assumption 1 is met.
- The transmission window size is enforced by the application: Assumptions 1 and 2 are met. We take the example of a time window whose size is set to 100 s, 200 s, 300 s and 400 s.

**Table 6.** Maximum number of EDs supported by OAPM.

$maxSF$	$MP(s)$	Case 1	Case 2	
		Free TW	TW Enforced	
		$MED$	$TW(s)$	$MED$
$SF12$	400	1812	100	1800
	800	3630	200	3624
	1200	5448	300	5448
	1600	7266	400	7248
$SF11$	400	3020	100	3020
	800	6045	200	6040
	1200	9070	300	9060
	1600	12,090	400	12,080
$SF10$	400	4292	100	4288
	800	8584	200	8576
	1200	12,876	300	12,864
	1600	17,168	400	17,168
$SF9$	400	6402	100	6396
	800	12,807	200	12,804
	1200	19,212	300	19,212
	1600	25,617	400	25,608
$SF8$	400	7772	100	7768
	800	15,546	200	15,544
	1200	23,320	300	23,320
	1600	31,094	400	31,088
$SF7$	400	6826	100	6824
	800	13,653	200	13,652
	1200	20,479	300	20,476
	1600	27,306	400	27,304

As expected, the number of EDs that can be deployed increases when  $MP$  increases or  $maxSF$  decreases. For instance, taking  $MP = 1600$  s, with  $maxSF = SF12$  the maximum number of deployed EDs becomes 7266 instead of 1812 with  $MP = 400$  s. With  $maxSF = SF10$ , the maximum number of EDs in the monitored area becomes 4292 instead of 1812 with  $SF12$ .

However, there is an exception, when there is only a single SF in the monitoring area (e.g.,  $minSF = maxSF = SF7$  in Table 6). This can be explained by the fact that in this case there is only one ED per sub-cluster (instead of two EDs in the case of  $maxSF = 8$ , for instance). When all EDs have the same SF, OAPM does not allow any parallel transmissions, because each sub-cluster is reduced to a single ED.

It is also worth noting that the maximum number of EDs supported with free TW size is always greater than or equal to the number of EDs with a TW size enforced by the application. The difference between both is  $\leq 18$  in all the cases tested.

## 7. Configuration, Synchronization and Monitoring

To be able to correctly schedule its transmissions, each ED must know:

- The following parameters common to all EDs:  $SP$ ,  $MP$ ,  $SP\_Start$ ,  $MP_1$ ,  $nMPperSP = \lfloor \frac{SP - MP_1 - SG}{MP} \rfloor$ .
- The parameters specific to each ED:  $TW_j$  its cluster transmission starting time expressed relatively to the beginning of the current Monitoring Period, and  $TT_h$  its own transmission time expressed relatively to  $TW_j$ .

These parameters are given to each ED during the configuration.

### 7.1. Message Format

The format of the Configuration messages, Synchronization messages and Monitoring messages are depicted by Figures 6–8, respectively.

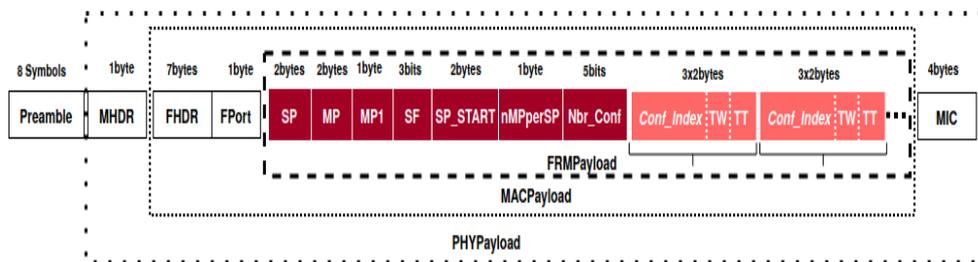


Figure 6. Format of a Configuration message.

For the configuration of EDs, the GW multicasts a message containing the value of:

- Common parameters: the Synchronization Period (2 bytes), the Monitoring Period (2 bytes), the Starting of the Synchronization Period (2 bytes), the number of MPs per SP (1 byte), the Starting time of the first Monitoring Period expressed relatively to the starting of the Synchronization Period (1 byte) as well as the highest SF used by some EDs (3 bits) (this SF is used by the GW to transmit the synchronization), and the number of EDs configured by this message (5 bits). This gives a total of 9 bytes. These parameters are depicted in dark red in Figure 6.
- Parameters specific to each ED: the configuration index (2 bytes), the starting of the transmission window (2 bytes) within the current Monitoring Period and the Transmission Time within the Transmission Window (2 bytes), leading to 6 bytes per ED configured. These parameters are depicted in light red in Figure 6.

The configuration message has a size close to the maximum size allowed by the SF used, as shown in the following example. As an example, let us consider the air pollution monitoring of a disk ranging 6km around the GW with a density of 10 EDs deployed per km<sup>2</sup> inspired from the West Oakland deployment in California [43]. Since this area can be divided into 132 square cells of 1 km<sup>2</sup>, 1320 EDs are deployed. In this case, the distribution of EDs in the different SFs is the following: 240 EDs in each  $SF \in [7, 10]$ , 200 EDs in  $SF_{11}$  and 160 EDs in  $SF_{12}$ . With a maximum payload of 59 bytes for  $SF \geq 10$ , up to eight EDs are configured simultaneously. This number is equal to 17 EDs for a maximum payload of 115 bytes for  $SF = 9$ , and 32 EDs for a maximum payload of 222 bytes for  $SF = 7$  or 8. Hence, the GW has to multicast 7 messages in  $SF_7$ , 7 in  $SF_8$ , 15 in  $SF_9$ , 30 in  $SF_{10}$ , 25 in  $SF_{11}$  and 20 in  $SF_{12}$ .

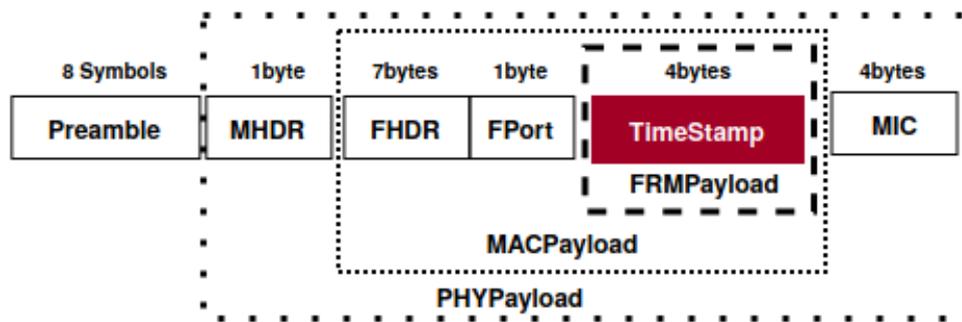


Figure 7. Format of a Synchronization message.

The Synchronization message contains the timestamp over 4 bytes, leading to a total size of 17 bytes.

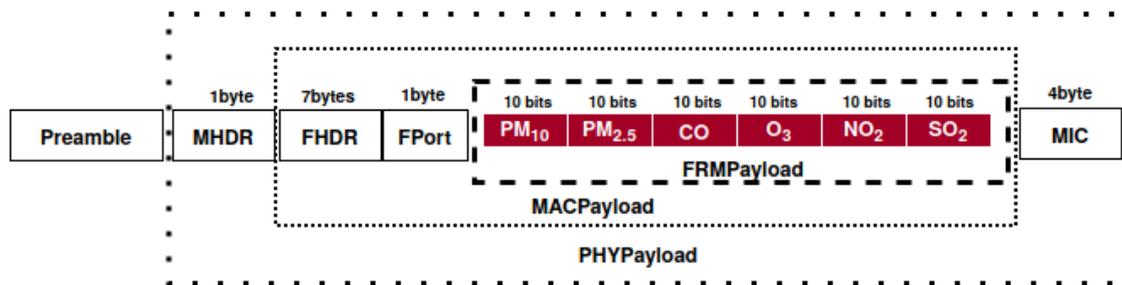


Figure 8. Format of a Monitoring message.

The index of each of the six main air pollutants is coded on 10 bits, leading to 8 bytes of payload and a total of 21 bytes for the Monitoring message.

### 7.2. Configuration of End Devices

All the configuration parameters are computed by the Server for each ED, and are then multicast to as many EDs using the SF of the configuration message as possible, as shown in Algorithm 1. Since the server knows the locations and SFs of each ED from the ED deployment, it deduces the ED membership to the clusters. It computes  $maxSF$  and  $minSF$  which are the maximum and minimum SF used in the network considered. It initializes all parameters that are common to all EDs such as the values of the Synchronization Period, the Monitoring Period, the time where the synchronization starts, the number of Monitoring Periods per Synchronization Period, etc. Then it computes the parameters specific to each ED. It assigns the transmission index to each ED belonging to Cluster  $j$ , ensuring that two EDs configured to use the same SF and belonging to the same Cluster  $j$  do not have the same transmission index  $Index$ . This is achieved by keeping the last index already assigned to the SF value in this cluster from Algorithm 1  $I(SF)$ . Then the server assigns  $TW_j$ —the starting time of the time window associated with cluster  $j$  expressed relatively to the current Monitoring Period.

After having computed the parameters of all EDs, the GW transmits these parameters to the EDs. To limit the overhead induced, the GW uses multicasts to configure at once as many EDs as possible, these EDs have the same SF as this used by the GW in its Configuration message. At the end of the Configuration phase, all EDs are configured with the necessary parameters required to ensure a correct transmission scheduling and total orthogonality of transmissions. More details are shown in Algorithm 1.

**Algorithm 1** End Devices Configuration

---

```

/* Run by the server to configure all the End Devices */
/* Initializations */
for each ED in the monitored area do
  /* Cluster assignment */
  Assign ED to a cluster according to its geographic coordinates obtained during deployment
  Compute  $minSF$  and  $maxSF$  the minimum and maximum SF used in the network
end for
/* Compute the parameters common to all EDs */
Initialize  $SP, MP, SP\_Start, MP_1, nMPperSP, maxSF$ 
/* Compute the transmission and receive times of all EDs */
for each cluster in the monitored area do
  if cluster=1 then
     $TW_1 \leftarrow 0$ 
  else
     $TW_{cluster} \leftarrow TW_{cluster-1} + \sum_{h=1}^{nsub_{cluster-1}} ToA_h + MG2 * nsub_{cluster-1}$ 
  end if
  for  $SF = minSF$  to  $maxSF$  do
    /* Initialize Transmit index per SF in cluster */
     $I(SF) \leftarrow 1$ 
  end for
  for each  $ED \in cluster$  do
    /* Assign Index=subcluster */
     $ED.Index \leftarrow I(ED.SF);$ 
     $I(ED.SF) ++$ 
    /* Assign Time Window */
     $ED.TW \leftarrow TW_{cluster}$ 
    /* Assign Transmission Time within TW */
     $ED.TT \leftarrow \sum_{k=1}^{Index-1} ToA_k + (Index - 1)MG2$ 
  end for /* End Device */
   $nsub_{cluster} \leftarrow Index$ 
end for /* Cluster */
/* Send configuration parameters to all EDs */
for  $SF = minSF$  to  $maxSF$  do
   $Unconfigured \leftarrow$  all the EDs using  $SF$ 
  while  $Unconfigured \neq empty$  do
    multicast configuration parameters to a maximum number of  $EDs \in Unconfigured$  using  $SF$ 
    Remove these EDs from  $Unconfigured$ 
  end while
end for

```

---

**7.3. Synchronization of EDs**

Since the clocks of EDs derive over time, it is necessary to resynchronize them periodically to avoid collisions between two successive transmissions made by two different EDs. All End Devices are periodically synchronized on the time reference provided by the GW. The main originality of this algorithm is to use one-shot synchronization with period  $SP$  and to apply clock drift compensation with a period  $MP$  much shorter than the Synchronization Period  $SP$ . Figure 9 depicts the time diagram of the periodic synchronization with period  $SP$  and the periodic monitoring with period  $MP \leq SP$ .

This lightweight synchronization algorithm takes advantage of multicast messages to synchronize all EDs at once. It uses two successive synchronization messages separated by one Synchronization Period to allow each ED to compute its clock drift with regard to the GW's clock which is the time reference. Each ED computes a compensation value applied before any transmission and any reception. As a consequence, any end device is kept synchronized within  $\pm\Delta \mu s$  to the GW time. More precisely,

the synchronization algorithm proceeds as follows. When the GW multicasts its  $i$ th synchronization message, it timestamps it with its local clock value  $T_{G,i}$ . When the End Device receives this message, it reads the value of its local clock denoted  $T_{E,i}$ , as depicted in Figure 9. These two values meet the following equation:

$$T_{E,i} = T_{G,i} + propagation\_delay + clock\_offset \tag{10}$$

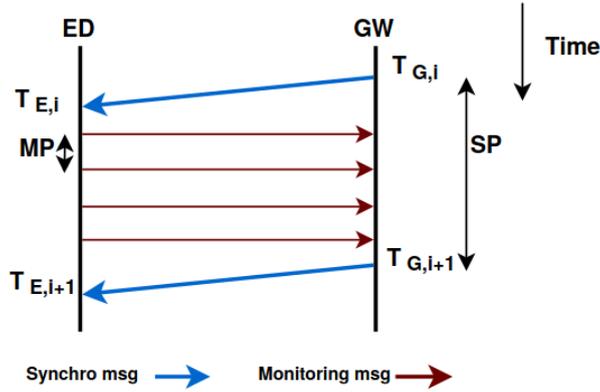


Figure 9. Time diagram of synchronization and monitoring.

Similarly for the  $(i + 1)$ th synchronization message, we get:

$$T_{E,i+1} = T_{G,i+1} + propagation\_delay + clock\_offset + clock\_drift * SP \tag{11}$$

By subtracting Equation (10) from Equation (11) and assuming the same propagation delay for the  $i$ th and the  $(i + 1)$ th synchronization messages:

$$T_{E,i+1} - T_{E,i} = T_{G,i+1} - T_{G,i} + clock\_drift * SP \tag{12}$$

Since  $T_{G,i+1} - T_{G,i} = SP$ , we get:

$T_{E,i+1} - T_{E,i} = SP + clock\_drift * SP$ . Hence the value of  $clock\_drift$ :

$$clock\_drift = \frac{T_{E,i+1} - T_{E,i} - SP}{SP} \tag{13}$$

To reduce the uncertainty and provide a better accuracy [44], the insertion of the timestamp in the synchronization message should be done at the physical layer. A clock drift compensation is computed by the ED, each time it has to transmit or receive a message on the medium. This computation relies on two assumptions:

- The clock drift is linear during the Synchronization Period. This assumption has been checked by many authors such as [23] for instance.
- The clock drift in the next Synchronisation Period will be very close to the one observed in the previous Synchronisation Period.

With these assumptions, the compensation, denoted  $Comp$ , to apply after a time duration  $\delta$  elapsed after the last synchronization message, is computed as:

$$Comp = clock\_drift * \delta. \tag{14}$$

This compensation is positive if the ED clock is going slowly and negative if it is going fast. As previously said, this compensation is applied by ED before transmitting its monitoring report and before receiving the next synchronization message to account for this clock drift. This compensation aims at allowing a less frequent synchronization (i.e.,  $MP \leq SP$ ).

#### 7.4. Air Pollution Monitoring by End Devices

Before participating in the data gathering process, each ED receives the configuration parameters from the server. After the reception of the parameters, each ED initializes its local parameters and computes the compensations to apply before any monitoring report transmission and before any synchronization message reception. Then, it sleeps until time *NextAwake*, which is set to the beginning of the next Synchronization Period, where it is ready to receive the Synchronization message. It updates its clock and computes the new compensations to apply before any monitoring report transmission and before any synchronization message reception. It sleeps again until time  $NextAwake = SP\_Start + MP_1 + ED.TW + ED.TT + Comp$  to transmit its Monitoring message.

When it awakes, the ED builds its air pollution report. Then, it transmits the report to the GW and sleeps until the time to transmit its next report in the next Monitoring Period. The ED repeats these same operations until the last Monitoring Period in the current Synchronization Period. It sleeps again until the end of the Synchronization Period where it is ready to receive the next synchronization message. The ED repeats the same behavior for the next Synchronization Period. More details are shown in Algorithm 2.

---

#### Algorithm 2 Monitoring step

---

```

/* Run by any End Device ED */
Receive (ConfigurationParameters)
Initialize its local parameters
NextAwake ← SP_Start
NbS = 0 /* Number of the Synchronization Period */
repeat
  /* Behavior of any ED during a synchro period */
  Sleep until NextAwake to receive next synchro msg
  Process the synchronization message,
  Update the clock
  Comp ← the compensation before next transmit
  NbS ++
  NextAwake ← SP_Start + MP1 + ED.TW + ED.TT + Comp
  for NbM = 1 to nMPperSP do
    /* Transmit its monitoring report once per MP
    during nMPperSP successive periods */
    Sleep until NextAwake to transmit its monitoring msg
    Build the air pollutant report
    Transmit the air pollutant report to the GW
    Comp ← the compensation before next transmit
    NextAwake ← SP_Start + MP1 + NbM * MP + ED.TW + ED.TT + Comp
  end for
  Comp ← the compensation before next receipt
  NextAwake ← SP_Start + NbS * SP + Comp
until forever

```

---

#### 7.5. Duty Cycle and Energy Consumption

The duty cycle of any ED can be computed per Synchronization Period as follows:

$$DutyCycle = \frac{nMPperSP \times Report_{SF} + Sync_{max}}{SP}, \quad (15)$$

where *nMPperSP* is the number of Monitoring Periods per Synchronization Period, *Report<sub>SF</sub>* denotes the transmission time on air of the monitoring report using the spreading factor *SF* of the ED considered, and *Sync<sub>max</sub>* is the transmission time of the synchronization message using the highest SF used by some EDs in the monitoring area.

To express the energy consumption of any ED, we introduce some additional notations. Let  $P_{sleep}$ ,  $P_{tx}$  and  $P_{rx}$  denote the power consumed in sleep mode, in transmit mode and in receive mode, respectively. The energy consumed by any ED during a Synchronization Period is equal to the energy to transmit  $nMPperSP$  monitoring messages, plus the energy to receive one synchronization message and plus the energy to sleep the remaining time, leading to:

$$Energy = nMPperSP \times Report_{SF} \times P_{tx} + (Sync_{max} + \Delta) \times P_{rx} + (SP - nMPperSP \times Report_{SF} - Sync_{max} - \Delta) \times P_{sleep}. \tag{16}$$

### 7.6. Average Message Delivery Latency

The average message delivery latency is defined as the average time elapsed from message generation on an ED to its delivery to the GW. It consists of:

- The waiting time before transmission. Since the monitoring report can be generated at any time in the Monitoring Period, this waiting time depends on the generation time. It ranges from 0 to the next Monitoring Period. We distinguish two cases:
  - First case: the current Monitoring Period is directly followed by another Monitoring Period. This occurs  $nMPperSP - 1$  times per  $nMPperSP$  Monitoring Periods.
  - Second case: the current Monitoring Period is followed by a synchronization. This occurs once per  $nMPperSP$  Monitoring Periods.
- The Time on Air of the monitoring report, which is fixed for a given ED.

Let  $SyncTime$  denote the Time on Air of the synchronization message plus two synchronization guard times. As a consequence, the average latency is equal to the half of the Monitoring Period, plus  $\frac{SyncTime}{2nMPperSP}$ , plus the Time on Air of the monitoring report.

## 8. Simulation Results

To evaluate the performances of OAPM, simulations were performed using the NS3.29 LoRaWAN module [27,45]. This module allows the simulation of the eight parallel receive paths of commercial GWs. Furthermore, it takes into account the capture effect, which allows to decode one among two colliding messages, if one message is received 6 dB higher. If a collision happens between two packets of the same SF, one can survive if it is 6 dB stronger than the other one. The implemented architecture consists of one gateway and a variable number of end devices deployed in a circular area of 6000 m radius around the gateway. The end devices are class A devices. They adopt the LoRaWAN default channels (868.1, 868.2, and 868.3) with a 125 kHz bandwidth to communicate with the gateway.

The gateway must listen to these three default channels. The number of Receive Paths of the gateway is 3, 6 or 8. Six receive paths correspond to the number of EDs per sub-cluster, whereas eight is the maximum number of receive paths that a commercial GW can listen to. The receive paths, denoted RP, are assigned to the three default channels according to Table 7.

**Table 7.** Receive paths assigned to the 3 default channels.

Total Number of Receive Paths	Channel 1	Channel 2	Channel 3
3 RPs	1 RP	1 RP	1 RP
6 RPs	2 RPs	2 RPs	2 RPs
8 RPs	3 RPs	3 RPs	2 RPs

We assume that each ED transmits its air pollution report once per Monitoring Period and this report contains the index values of the major six air pollutants, leading to a data payload of 21 bytes. The value of the Monitoring Period varies from 400 s to 1600 s, corresponding to a Time Window

ranging from 100 s to 400 s, because of the existence of four clusters. A large number of Monitoring Periods is simulated. The simulation parameters used in our experiments are summarized in Table 8. Note that the simulations are performed with the same conditions as those taken for the computation of results given in Table 6 of Section 6.3.

**Table 8.** Simulation Parameters.

Parameter	Value
Channels	Default Channels 868.1, 868.2, 868.3 MHz
Bandwidth	125 kHz
Receive Paths	3, 6, 8
Spreading Factors	7, 8, 9, 10, 11, 12
TX Power	14 dBm corresponding to $Power_{Tx} = 28$ mA
RX Current	$Power_{Rx} = 11.2$ mA
Idle Current	$Power_{Idle} = 1.4$ mA
Sleep Current	$Power_{Sleep} = 15$ $\mu$ A
Number of EDs	[0...5000]
Area radius	6000 m
Data payload	21 bytes
Message Type	Unconfirmed
Time Window	100 s, 200 s, 300 s, 400 s
Simulation Time	32,000 s

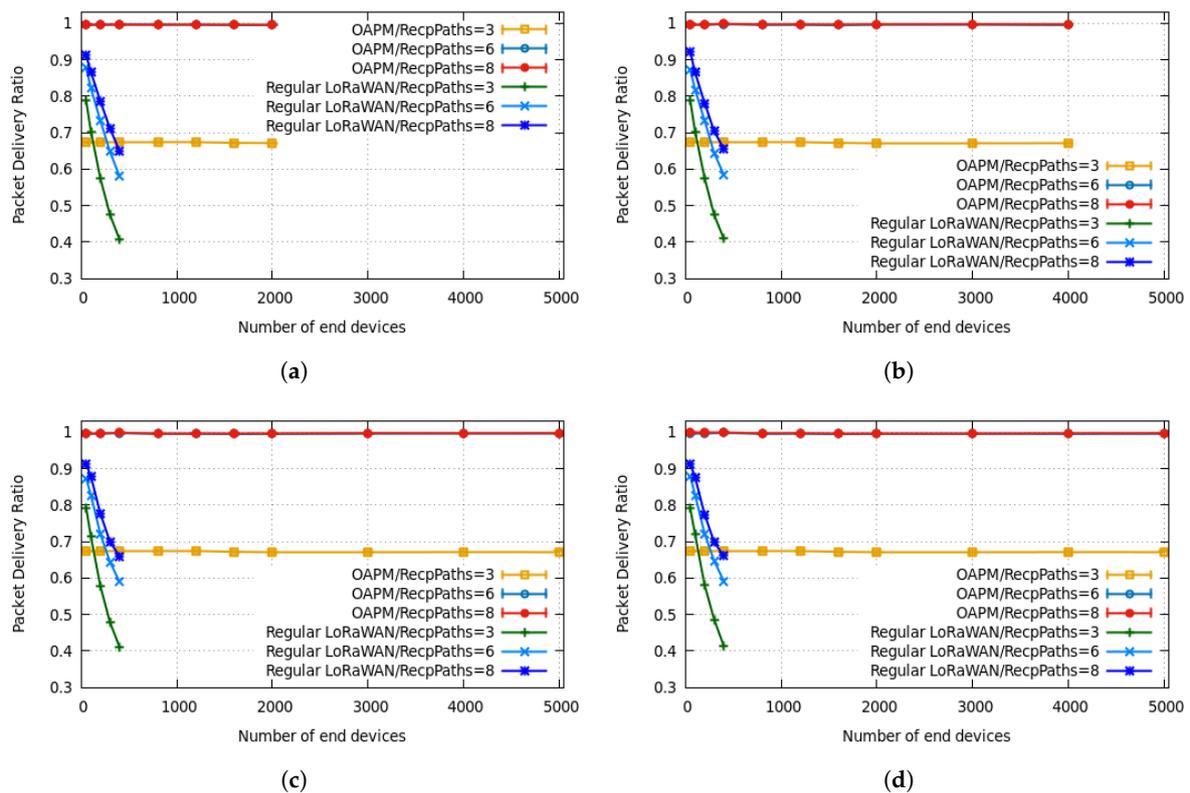
### 8.1. Nominal Behavior

We first study the OAPM behavior when the number of EDs meets Equations (7) and (9) given in Section 6.3, which corresponds to its nominal behavior. According to Table 6 and assuming a Monitoring Period of 400 s, corresponding to a Time Window  $TW = 100$  s for each of the four clusters considered, and the existence of EDs with SF12, the maximum number of EDs is equal to 1812 for a free time window and is equal to 1800 if the time window is enforced by the application. Similarly, it is limited to 3630 for  $TW = 200$  s. This is the reason why the simulations are not run for a higher number of EDs for  $TW = 100$  s and  $TW = 200$  s in Figure 10a,b, respectively, as well as for Figure 11. For higher values of  $TW$ , the number of EDs supported exceeds 5000.

As depicted in Figure 10a–d, three conclusions can be drawn for both OAPM and regular LoRaWAN:

- The implementation of only three receive paths results in the worst packet delivery ratio (PDR).
- The implementation of eight receive paths instead of six slightly improves the PDR.
- A smaller Monitoring Period leads to a smaller number of EDs supported while ensuring an acceptable PDR (i.e.,  $\geq 0.7$ ).

In OAPM, each sub-cluster generally schedules six simultaneous transmissions. If there are only three Receive Paths and six transmissions are simultaneously received by the GW, only 50% of them are demodulated correctly and the rest of them are rejected. Simulation results show a PDR of 0.67 which can be explained by the fact that all sub-clusters may not contain the six spreading factors. By implementing six Receive Paths, each packet of the six simultaneously generated can be demodulated on one available path which results in a packet delivery ratio higher than 98%. The full capacity configuration offers more Receive Paths (i.e., eight paths) than the number of simultaneously generated packets which results in a better PDR up to 99.7%. With eight Receive Paths, the probability of selecting a busy channel is smaller than for six Receive Paths. This explains the higher PDR for eight Receive Paths.



**Figure 10.** Packet delivery ratio versus number of end devices, for different numbers of gateway Receive Paths and different time window durations. (a) TW = 100 s; (b) TW = 200 s; (c) TW = 300 s; (d) TW = 400 s.

Furthermore, it is worth noting the excellent behavior of OAPM, materialized by a quasi-constant value of the PDR as long as the number of EDs meets Equations (7) and (9). However, for a better PDR a number of receive paths  $\geq 6$  is required.

Regular LoRaWAN based on pure Aloha is unable to support more than 100 EDs if a  $PDR \geq 0.7$  is required, as illustrated in Figure 10a–d. It is worth noting that for small values of EDs (i.e., less than 100), pure Aloha outperforms OAPM with three receive paths, which can be explained by the distribution of ED transmission times all over the Monitoring Period with pure Aloha, whereas the ED transmission times are concentrated at the beginning of each Monitoring Period with OAPM. As a conclusion of Figure 10a–d, the number of available paths if strictly less than 6, has a very strong impact on the PDR ensured by OAPM. It is strongly recommended to use OAPM with at least six receive paths, which considerably outperforms pure Aloha, while supporting a very higher number of EDs.

Figure 11 compares the PDR obtained for different time window durations using full capacity Receive Paths (i.e., eight paths) with OAPM and regular LoRaWAN. With OAPM, as long as the number of EDs meets the Equations (7) and (9), the duration of the time window has no impact on the PDR. For example, for a TW duration of 300 s and 400 s, the PDR remains stable, even when the number of EDs reaches 5000. Since the communications of EDs belonging to the same sub-cluster are orthogonal, and the number of packets lost due to under sensitivity factor of the GW is null, simulation results confirm that the decrease of PDR is due to the non-availability of Receive Paths. For eight Receive Paths, three are assigned to the first frequency channel, three to the second channel and two to the third channel. A non-availability of Receive Paths occurs when on a same channel, a number of EDs higher than the number of receive paths assigned to this channel transmit simultaneously. Consequently, the number of available paths has a strong impact on the PDR, when the number of EDs meets Equations (7) and (9).

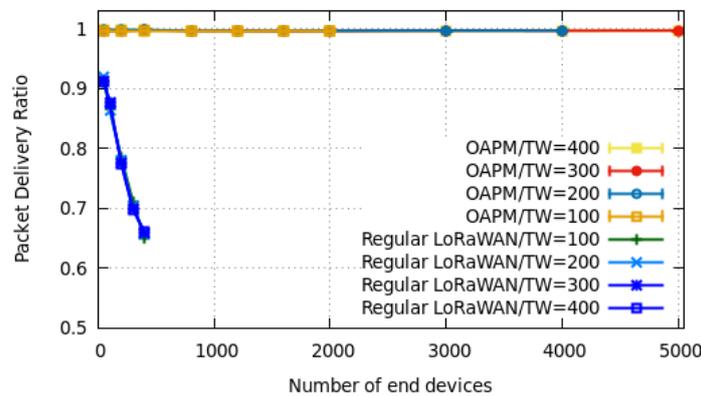


Figure 11. Impact of time window duration on PDR using 8 Receive Paths.

We now study the behavior of OAPM outside the limits enforced by Equations (7) and (9) and evaluate the reliability degradation, when the number of receive paths is equal to 3, 6 or 8. For example, taking the number of EDs equal to 5000 and TW equal to 100 s, 6723 packets are lost when the three default Receive Paths are implemented as depicted in Figure 12a. If six Receive Paths are used, 1274 packets are lost, as shown in Figure 12b. Only the number of packets corresponding to the number of available paths are demodulated correctly and the rest is rejected due to the non-availability of Receive Paths. In addition, when Equations (7) and (9) are no longer met, EDs belonging to two different sub-clusters may transmit simultaneously, which increases the probability of unavailability of a receive path leading to a message loss.

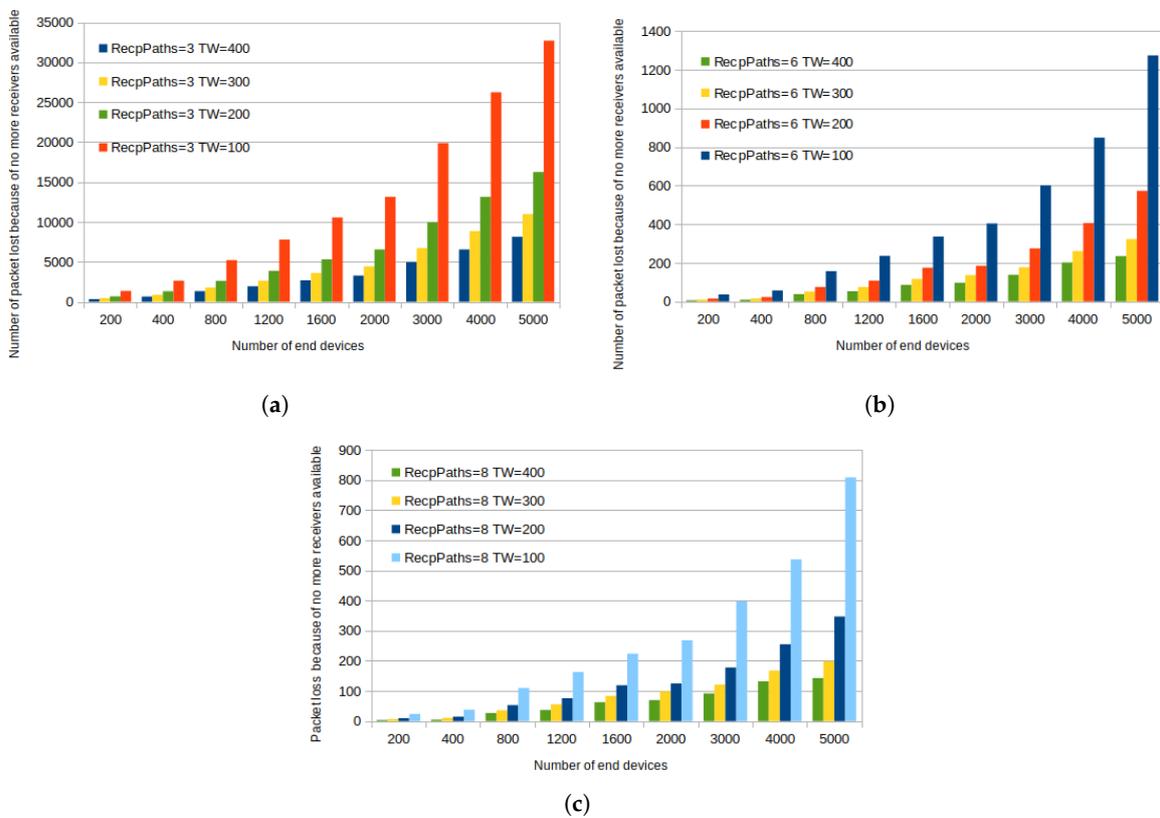


Figure 12. Number of lost packets versus number of EDs. (a) Receive Paths = 3; (b) Receive Paths = 6; (c) Receive Paths = 8.

### 8.2. Non-Nominal Behavior

On the other hand, when Equations (7) and (9) are met, the only cause of message loss is due to the fact that a number of EDs, in the same sub-cluster, higher than the number of Receive Paths assigned to this channel submit their packets on the same frequency channel. This is due to the random selection of the frequency channel by each ED.

In any case, as shown in Figure 12a–c, the number of packets lost increases when the number of Receive Paths decreases.

Simulation results depicted in Figures 10a–12c corroborate the theoretical results presented in Table 6 of Section 6.3. Finally, we compare OAPM with Alternative Data Gathering [24] and regular LoRaWAN where the EDs access the medium according to pure Aloha. We evaluate the PDR when the number of end devices ranges from 200 to 5000, assuming the GW implements the three default channels, eight receive paths and the Monitoring Period is equal to 400 s, corresponding to  $TW = 100$  s. Figure 13 shows that OAPM outperforms ADG which outperforms LoRaWAN in terms of PDR. Furthermore, the gain brought by OAPM increases with the number of end devices. For example, in the case of 1200 devices, the PDR of OAPM is equal to 0.972 against 0.799 in ADG and 0.49 in regular LoRaWAN. When the number of devices increases, the number of simultaneous transmissions also increases which leads to a massive drop in PDR values for both ADG and regular LoRaWAN. OAPM, by scheduling ED transmissions, provides more stable values of PDR up to 5000 EDs, which confirms the very good behavior of OAPM. All the simulation results reported in this paper show that OAPM considerably improves the scalability of LoRaWAN.

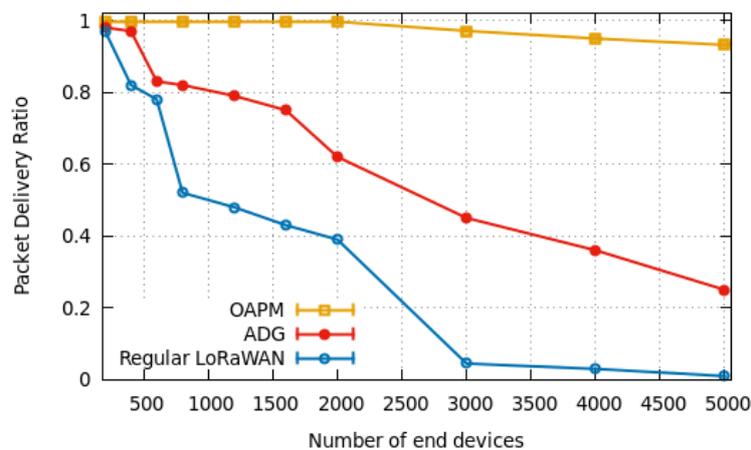


Figure 13. Packet Delivery Ratio of OAPM compared with ADG and regular LoRaWAN.

### 8.3. Energy Consumption, Duty Cycle and Average Latency

With a Monitoring Period of 400 s and a Synchronization Period of 1602 s and assuming a battery capacity of 1000 mAh, OAPM ensures an ED energy consumption per Synchronization Period, lifetime, duty cycle and average message delivery latency which depend on the spreading factor used by the ED considered. Simulation results are given in Table 9 for different values of the spreading factor. They corroborate the theoretical results given in Sections 7.5 and 7.6. Notice that these values are perfectly compliant with a duty cycle  $\leq 1\%$  required by ETSI. This evaluation does not take into account the power consumed to read sensors and process the reading. An evaluation of this consumption on a real prototype is left for further work.

**Table 9.** Energy consumption, duty cycle and average latency for OAPM.

Parameter	ED Spreading Factor		
	SF7	SF10	SF12
Energy consumption (J)	0.0052	0.0342	0.1218
Lifetime (year)	8.83	3.27	1.13
Duty cycle (%)	0.08	0.15	0.39
Average latency (s)	200.58	200.80	201.84

As a consequence, there is no accuracy problem with OAPM, a TDMA-based solution for periodic data gathering. Notice that the average latency does not depend on the number of EDs, provided that this number meets Equations (7) and (9) given in Section 6.3.

## 9. Conclusions

This paper presents OAPM, a LoRaWAN-based architecture for monitoring air pollution. The system's behavior includes four phases after the deployment: (1) the Joining phase where the ED joins the LoRaWAN network, (2) the Configuration phase where the ED is assigned its configuration parameters (e.g., Synchronization Period, Monitoring Period, number of Monitoring Periods per Synchronization Period, Transmission Window, Transmission Time), (3) the Synchronization phase where the ED is synchronized to the clock of the gateway, which is the reference time in LoRaWAN, and (4) the Monitoring phase where the EDs send their pollutant report to the gateway and sleep until their next transmission period or the next Synchronization Period, when they repeat the transmission process. Potential applications are various including air pollution monitoring, smart farming, environment monitoring. They would benefit from OAPM advantages which are the following:

- It supports of a high number of EDs while maintaining a PDR close to 1, provided a number of receive paths  $\geq 6$ .
- It has a low energy consumption, because all transmissions are scheduled by the network server.
- It provides a high network lifetime.
- It is fully compliant with the requirements expressed by the World Health Organization and the US Environmental Agency for air pollution monitoring.
- It is a simple solution based on an implicit clustering of EDs according to their geographic coordinates and the computation of their transmission times by the Network Server.

OAPM has been implemented and simulations were conducted using the NS3 simulator. By comparing OAPM with other proposed strategies in the literature, we found that OAPM ensures orthogonality between transmissions while supporting large-scale networks. However, the non-deterministic selection of the frequency channels introduces a slight drop in the performances. This happens when several EDs belonging to the same sub-cluster share the same frequency channel.

As a further work, we intend to introduce a new solution enabling a collision-free sharing of frequencies between EDs. This amendment will be combined with the proposed OAPM to ensure high-level performances and create a scalable air pollution monitoring solution. We will also study how to extend this solution to support multiple GWs.

**Author Contributions:** R.H. designed the software, run the simulations and contributed to the methodology, writing and editing of the paper. P.M. was in charge of the conceptualization, formal analysis and actively contributed to the writing, review and editing of the paper. S.B. was in charge of the investigation and supervision. She also participated to the writing, review and editing of the paper. L.A.S. was in charge of the supervision. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Notations

The following notations are used in this manuscript:

ED	End Device
GW	Gateway
NS	Network Server
DOAJ	Directory of open access journals
TLA	Three letter acronym
LD	linear dichroism
SF	Spreading Factor
RP	Receive Path
MP	Monitoring Period
SP	Synchronization Period
TW	Transmission Window
TT	Transmission Time
MG1	Monitoring Guard Time 1
MG2	Monitoring Guard Time 2
SG	Synchronization Guard Time
$\Delta$	Synchronization of any ED with regard to the GW ensured by the synchronization algorithm
<i>Sync</i>	Time on Air of the Synchronization message transmitted with the maximum SF used in the network
<i>Rep</i>	Time on Air of the Monitoring Report message
<i>SP_Start</i>	Starting time of the first Synchronization Period
<i>ToA<sub>h</sub></i>	Total Time on Air used by sub-cluster <i>h</i> to transmit the Monitoring Reports of its devices
<i>nMPperSP</i>	Number of Monitoring Periods per Synchronization Period
PDR	Packet Delivery Ratio
MED	Maximum number of EDs supported by OAPM

## References

1. World Health Organization. *Ambient Air Pollution: A Global Assessment of Exposure and Burden of Disease*; World Health Organization: Geneva, Switzerland, 2016.
2. World Health Organization. *WHO Guidelines for Indoor Air Quality: Household Fuel Combustion*; World Health Organization: Geneva, Switzerland, 2014.
3. Mokrani, H.; Lounas, R.; Bennai, M.T.; Salhi, D.E.; Djerbi, R. Air quality monitoring using iot: A survey. In Proceedings of the 2019 IEEE International Conference on Smart Internet of Things (SmartIoT), Tianjin, China, 9–11 August 2019; pp. 127–134.
4. Wesseling, J.; de Ruiter, H.; Blokhuis, C.; Drukker, D.; Weijers, E.; Volten, H.; Vonk, J.; Gast, L.; Voogt, M.; Zandveld, P.; et al. Development and implementation of a platform for public information on air quality, sensor measurements, and citizen science. *Atmosphere* **2019**, *10*, 445. [CrossRef]
5. World Health Organization. World Health Organization, Who We Are. Available online: <https://www.who.int/about/who-we-are> (accessed on 30 March 2020).
6. U.S. Environmental Protection Agency Office of Air Quality Planning. *Air Quality Index: A Guide to Air Quality and Your Health*; EPA-456/F-14-002; U.S. Environmental Protection Agency Office: Research Triangle Park, NC, USA, 2014.
7. Blenn, N.; Kuipers, F. LoRaWAN in the wild: Measurements from the things network. *arXiv* **2017**, arXiv:1706.03086.
8. Bor, M.C.; Roedig, U.; Voigt, T.; Alonso, J.M. 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, Malta, Malta, 13–17 November 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 59–67. [CrossRef]
9. Mintz, D. *Technical Assistance Document for the Reporting of Daily Air Quality—The Air Quality Index (aqi): US Environmental Protection Agency*; Office of Air Quality Planning and Standards: Washington, DC, USA, 2013; pp. 1–26.

10. Airparif Association. Air Quality in Ile-de-France. 2020. Available online: <https://data-airparif-asso.opendata.arcgis.com/search?collection=Dataset> (accessed on 20 March 2020).
11. Shah, J.; Mishra, B. IoT enabled environmental monitoring system for smart cities. In Proceedings of the 2016 international conference on internet of things and applications (IOTA), Pune, India, 22–24 January 2016; pp. 383–388.
12. Kim, S.H.; Jeong, J.M.; Hwang, M.T.; Kang, C.S. Development of an IoT-based atmospheric environment monitoring system. In Proceedings of the 2017 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 18–20 October 2017; pp. 861–863.
13. Han, Y.; Park, B.; Jeong, J. A Novel Architecture of Air Pollution Measurement Platform Using 5G and Blockchain for Industrial IoT Applications. *Procedia Comput. Sci.* **2019**, *155*, 728–733. [[CrossRef](#)]
14. Dhingra, S.; Madda, R.B.; Gandomi, A.H.; Patan, R.; Daneshmand, M. Internet of Things Mobile–Air Pollution Monitoring System (IoT–Mobair). *IEEE Internet Things J.* **2019**, *6*, 5577–5584. [[CrossRef](#)]
15. Tzortzakakis, K.; Papafotis, K.; Sotiriadis, P.P. Wireless self powered environmental monitoring system for smart cities based on LoRa. In Proceedings of the 2017 Panhellenic Conference on Electronics and Telecommunications (PACET), Xanthi, Greece, 17–18 November 2017; pp. 1–4.
16. Eletreby, R.; Zhang, D.; Kumar, S.; Yağan, O. Empowering low-power wide area networks in urban settings. In Proceedings of the Conference of the ACM Special Interest Group on Data Communication, Los Angeles, CA, USA, 21–25 August 2017; pp. 309–321.
17. Gadre, A.; Yi, F.; Rowe, A.; Iannucci, B.; Kumar, S. Quick (and Dirty) Aggregate Queries on Low-Power WANs. In Proceedings of the 2020 The 19th ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN), Sydney, Australia, 21–24 April 2020.
18. Dongare, A.; Narayanan, R.; Gadre, A.; Luong, A.; Balanuta, A.; Kumar, S.; Iannucci, B.; Rowe, A. Charm: Exploiting geographical diversity through coherent combining in low-power wide-area networks. In Proceedings of the 2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), Porto, Portugal, 11–13 April 2018; pp. 60–71.
19. Polonelli, T.; Brunelli, D.; Benini, L. Slotted aloha overlay on lorawan—a distributed synchronization approach. In Proceedings of the 2018 IEEE 16th International Conference on Embedded and Ubiquitous Computing (EUC), Bucharest, Romania, 29–31 October 2018; pp. 129–132.
20. Ramirez, C.G.; Sergeev, A.; Dyussenova, A.; Iannucci, B. LongShoT: Long-range synchronization of time. In Proceedings of the 18th International Conference on Information Processing in Sensor Networks, Montreal, QC, Canada, 16–18 April 2019; pp. 289–300.
21. Haxhibeqiri, J.; Moerman, I.; Hoebeke, J. Low overhead scheduling of lora transmissions for improved scalability. *IEEE Internet Things J.* **2018**, *6*, 3097–3109. [[CrossRef](#)]
22. Zorbas, D.; Abdelfadeel, K.; Kotzanikolaou, P.; Pesch, D. TS-LoRa: Time-slotted LoRaWAN for the Industrial Internet of Things. *Comput. Commun.* **2020**, *153*, 1–10. [[CrossRef](#)]
23. Gao, S.; Zhang, X.; Du, C.; Ji, Q. A Multichannel Low-Power Wide-Area Network with High-Accuracy Synchronization Ability for Machine Vibration Monitoring. *IEEE Internet Things J.* **2019**, *6*, 5040–5047. [[CrossRef](#)]
24. Haihem, R.; Ghazel, C.; Azouz Saidane, L. An alternative data gathering of the air pollutants in the urban environment using lora and lorawan. In Proceedings of the 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), Limassol, Cyprus, 25–29 June 2018; pp. 1237–1242.
25. Raza, U.; Kulkarni, P.; Sooriyabandara, M. Low power wide area networks: An overview. *IEEE Commun. Surv. Tutorials* **2017**, *19*, 855–873. [[CrossRef](#)]
26. Wang, C.H.; Huang, Y.K.; Zheng, X.Y.; Lin, T.S.; Chuang, C.L.; Jiang, J.A. A self sustainable air quality monitoring system using WSN. In Proceedings of the 2012 Fifth IEEE International Conference on Service-Oriented Computing and Applications (SOCA), Taipei, Taiwan, 17–19 December 2012; pp. 1–6.
27. Magrin, D.; Centenaro, M.; Vangelista, L. Performance evaluation of LoRa networks in a smart city scenario. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–7.
28. Addabbo, T.; Fort, A.; Mugnaini, M.; Parri, L.; Parrino, S.; Pozzebon, A.; Vignoli, V. A low power IoT architecture for the monitoring of chemical emissions. *ACTA IMEKO* **2019**, *8*, 53–61. [[CrossRef](#)]
29. Mekki, K.; Bajic, E.; Chaxel, F.; Meyer, F. A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express* **2019**, *5*, 1–7. [[CrossRef](#)]

30. Wg, T. *Technical Overview of LoRa and LoRaWAN*. Lora-Alliance; LoRa® Alliance 2400 Camino Ramon: San Ramon, CA, USA, 2015.
31. Chiani, M.; Elzanaty, A. On the LoRa Modulation for IoT: Waveform Properties and Spectral Analysis. *IEEE Internet Things J.* **2019**, *6*, 8463–8470. [[CrossRef](#)]
32. Semtech. Wireless and Sensing Products Datasheet. 2017. Available online: <https://www.semtech.com/products/wireless-rf/lora-gateways/sx1301/#download-resources> (accessed on 30 April 2020).
33. Capuzzo, M.; Magrin, D.; Zanella, A. Confirmed traffic in LoRaWAN: Pitfalls and countermeasures. In Proceedings of the 2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), Capri, Italy, 20–22 June 2018; pp. 1–7. [[CrossRef](#)]
34. Alliance, L. LoRaWAN 1.0. 3 specification. *Lora-Alliance. Org* **2018**, *1*, 1–72.
35. Queralta, J.; Gia, T.; Zou, Z.; Tenhunen, H.; Westerlund, T. Comparative study of LPWAN technologies on unlicensed bands for M2M communication in the IoT: Beyond LoRa and LoRaWAN. *Procedia Comput. Sci.* **2019**, *155*, 343–350. [[CrossRef](#)]
36. Adelantado, F.; Vilajosana, X.; Tuset-Peiro, P.; Martinez, B.; Melia-Segui, J.; Watteyne, T. Understanding the limits of LoRaWAN. *IEEE Commun. Mag.* **2017**, *55*, 34–40. [[CrossRef](#)]
37. Polonelli, T.; Brunelli, D.; Marzocchi, A.; Benini, L. Slotted aloha on lorawan-design, analysis, and deployment. *Sensors* **2019**, *19*, 838. [[CrossRef](#)] [[PubMed](#)]
38. Duda, A.; Heusse, M. Spatial issues in modeling lorawan capacity. In Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Miami, FL, USA, 25–29 November 2019; pp. 191–198.
39. Goursaud, C.; Gorce, J.M. Dedicated networks for IoT: PHY/MAC state of the art and challenges. In *EAI Endorsed Transactions on Internet of Things*; European Alliance for Innovation: Ghent, Belgium, 2015; doi:10.4108/eai.26-10-2015.150597. [[CrossRef](#)]
40. The Things Network. Regional Parameters (Frequency Plans & Duty Cycle). 2020. Available online: <https://www.thethingsnetwork.org/docs/lorawan/frequency-plans.html> (accessed on 25 June 2020).
41. CEPT, Electronic Communications Committee. ERC Recommendation 70-03. 2020. Available online: <https://www.ecodocdb.dk/download/25c41779-cd6e/Rec7003e.pdf> (accessed on 25 June 2020).
42. Yu, F.; Zhu, Z.; Fan, Z. Study on the feasibility of LoRaWAN for smart city applications. In Proceedings of the 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Rome, Italy, 9–11 October 2017; pp. 334–340.
43. Caubel, J.J.; Cados, T.E.; Preble, C.V.; Kirchstetter, T.W. A Distributed Network of 100 Black Carbon Sensors for 100 Days of Air Quality Monitoring in West Oakland, California. *Environ. Sci. Technol.* **2019**, *53*, 7564–7573. [[CrossRef](#)] [[PubMed](#)]
44. Rizzi, M.; Depari, A.; Ferrari, P.; Flammini, A.; Rinaldi, S.; Sisinni, E. Synchronization Uncertainty Versus Power Efficiency in LoRaWAN Networks. *IEEE Trans. Instrum. Meas.* **2019**, *68*, 1101–1111. [[CrossRef](#)]
45. Haiahem, R.; Boumerdassi, S.; Azouz Saidane, L. A New Dynamic Urban Environment Air Pollution Monitoring Protocol using WAVE (DAP-MP). In Proceedings of the 2020 International Conference on Information Networking (ICOIN), Barcelona, Spain, 7–10 January 2020; pp. 622–627.

