Collection of Data with Drones in Precision Agriculture: Analytical Model and LoRa Case Study

Antonio Caruso*, Stefano Chessa†, Soledad Escolar‡ Jesús Barba‡, Juan Carlos López‡
*Department of Mathematics and Physics "Ennio De Giorgi", University of Salento, Collegio Fiorini, Lecce, Italy.
†Department of Computer Science, University of Pisa, Pisa, Italy.
‡School of Computer Science, University of Castilla-La Mancha, Ciudad Real, Spain.

***Abstract*—Unmanned Aerial Vehicles (UAVs) are autonomous devices employed as data collectors in precision agriculture to support a large number of applications. These UAVs gather data from on-the-ground wireless sensor networks, especially in scenarios that lack any kind of fixed communication infrastructure or where the available infrastructure does not fit the application requirements. Sensors on the ground can store sensing data, and in scenarios that do not require a real-time observation and analysis of the data, like in smart farming, a drone can be used maybe once or two times each day to collect and report the data to a command-and-control center that use them directly, without any other infrastructure (cloud or edges). In this paper, we study analytically how close the drone, that uses a LoRa radio, needs to fly over the sensors to collect data with a given quality of data collection. This can be used to properly spacing the sensor on the field at deployment time, to select among different type of drones, and to properly solve some tradeoff related to field size vs autonomy of the drones and the path used by the latter when collecting the data.**

***Index Terms*—Internet of Things; Precision Agriculture; Optimal Data Collection; Drones; LoRa.**

I. Introduction

The adoption of Internet of Things (IoT) technologies [1] is a recent trend in precision agriculture [2], [3]. In fact, due to the ability to combine sensing, actuation, processing and Internet connectivity into low-cost devices, it is possible to monitor the crops, the soil and the environment where they grow with an unprecedented granularity. The consequence is however the continuous streaming of sensor data that need to be collected and analyzed, in order to be transformed into high-level information that can be interpreted by the agronomists and that can be used to activate the automatisms that feed and assist the crops. On the side of the data analysis requirements, the recent developments in artificial intelligence and data analytics are already being explored by researchers to produce novel models suitable to interpret such data and relate them to key parameters of the crops [4]–[6]. On the side of the data collection requirement however, the connectivity to the Internet of IoT devices, which is provided by 4G/5G networks in populated areas, cannot be given for granted in rural and remote areas, which is the typical case in large-scale, precision agriculture sites. A study on the coverage vs. costs provided by different communication technologies can be found in [7].

Also for this reason, new connectivity services based on long-range, low power technologies (such as LoRa [8]) have appeared on the market[1].

Typically, they exploit a fixed infrastructure of base stations able to provide connectivity to IoT devices with very low bandwidth requirements over a range much larger than usual Wi-Fi, ZigBee or Bluetooth technologies (it can even reach a few kilometers, depending on the shape and conditions of the ground). An alternative approach is to replace the fixed infrastructure of base stations with mobile base stations that are carried by Unmanned Aerial Vehicles (UAV), hereafter called "drones". These vehicles have now reached a level of reliability at affordable costs that make them appealing for civil applications, and their market is growing fast, with exponential growth expectations [9]. By means of drones, a single mobile base station may provide connectivity to IoT devices over an area that may otherwise require even several fixed base stations to be covered.

In this paper we consider this latter approach, and we study how an architecture comprising drones and sensors in the field can be configured in order to reach minimum connectivity constraints. We focus in particular on large row crop farming, which are densely and regularly planted areas that can be managed by machines across the entire field and that are widely studied in the literature [10], [11].

In this context we consider a regular deployment of sensors in the field (namely in a grid), which maps well over the regular structure of the crops. We also assume that a drone travels periodically over the field carrying a base station to communicate with the sensors in the ground, and that it moves along a predefined path to ensure uniform coverage of all the sensors in the field. Under these assumptions we provide an analytic model that can be used to configure the path of the drone. The model can be used to determine the *minimum-maximum distance* that should be kept between each sensor and the drone, to guarantee a given probability of collecting the sensor data. The analytic model is general and can be adapted to different wireless technologies with different ranges and throughputs. As a case study we analyze, by means of simulation, the use of the model in a LoRa scenario, considering different specifications of commercial drones and field sizes.

The rest of the paper is organized as follows. After reviewing the related work in Section II, in Section III we dip into LoRa to review the set of low-level parameters that an application

[1] https://www.semtech.com/lora/ecosystem/networks, https://www.a2asmartcity.it/utilty-week-paris/

designer needs to know as they impact significantly on the communication performance. In Section IV we present a data collection model to compute a regular deployment of sensors on the target field that guarantees a given probability of successful data communication between the drone and the sensors through point-to-point communications. We study the minimum-maximum distance for small and large deployments and discuss the analytical results and their impact on the drones autonomy and field size. In Section V we discuss the conditions for deployments admissibility and evaluate the feasibility of using LoRa as communication protocol for the model described, as well as its performance in terms of collisions and retransmissions. Finally, in Section VI we draw the main conclusions and suggestions for further research.

II. Related Work

In a typical organization, wireless sensor networks (WSN) are organized as a multi-hop tree structure rooted in a device called sink [12]. Each sensing device (node) thus occupies a position in this tree, and acts as a router by forwarding towards the sink all the data produced by the nodes in its subtree. This network organization can be managed by simple protocols, but it does not scale well because the nodes closer to the sink in the network become overburdened by the flow of data coming from the rest of the network and thus deplete faster their batteries, thus implicitly producing a disconnection of the network from the sink and thus making the WSN unavailable [13]. A solution to overcome this problem consists in the use of a controlled mobile sink (in practice a drone) that may travel across the network to collect the data from the nodes [14], [15]. A good survey of the literature on these approaches can be found in [16] and [17], [18]. In the context of precision agriculture [19], [2], the use of aerial observation, to better understand and monitor large fields from altitude, is widely practiced. In recent years, there has been an increasing interest in employing UAVs instead of ground mobile sinks. In the following we cover some works that use UAVs as a mobile data sink.

In Wang et. al [20] an entire framework called FPPWR (Fast Path Planning with Rules) is proposed, it solves the problems of nodes positioning, anchor points searching, path planning of the UAV, and data collection. They evaluated the performance of their proposal by studying the time required to collect all data, the flight path distance, and the volume of data collected. In [21] the authors consider how to optimize the data collection time for multiple drones, when data transmission time should be respected. They propose a predefined route scheme and formulate it into the problem of *route selection and communication association*, that is solved with a $\log(m)$-approximation ratio using a greedy algorithm, where m is the number of cluster-heads. They show that the proposed greedy heuristic is 154% at most and 145% on average of optimal solution. In [22], it is assumed that the UAV flies horizontally with a fixed altitude, and their authors study energy-efficient UAV communication with a ground terminal, optimizing the UAV's trajectory, the communication throughput and the UAV's energy consumption.

In [23] the authors study the problem of maximizing the throughput of the down-link channel (so the focus is more related to different application scenarios than data collection), by proper scheduling and UAV's trajectory planning. They formulated a mixed integer non-convex optimization problem and solved it efficiently. In [24] the authors propose the use of drones and LoRa to collect data from sensors, they change the default Aloha MAC in LoRaWan with a time-scheduled policy to reduce the probability of packet collisions. Their simulations show that a drone can collect the data produced by 80 sensors in a day (5760 bytes) in an area of 1.5km^2 without packet collisions. In [25], [26] the authors assume that sensors are deployed randomly on a field and the data are preliminary collected by a set of cluster heads, it is also assumed that the communication area of different cluster heads do not overlap. The problem of computing the optimal (minimum latency) trajectory for a drone through all cluster heads, in order to collect all data from them, has been studied. The drone could move freely in the area, the number and locations of the cluster heads, and the amount of data to be transferred, from them to the drone, are known *ex ante*. The problem of minimizing the latency from the start of the drone to its return to the base reduces to two sub-problems: an extension of the TSP to obtain the optimal circuit through the cluster heads, and the computation of the minimum hovering time for each cluster head so that all data can be transmitted to the drone; since the problem is clearly NP-Complete, two different approximation algorithms are presented. In [27] the authors study how to dynamically control the speed of the drone in order to maximize the data collection efficiency, while reducing the access congestion for the UAV-based base stations.

Another relevant aspects, that also concerns the use of drones, is related to cyber attacks and security. For these aspects we refer the reader to the literature [28], [29].

The problem of data collection using LoRa has been faced up in several works. In a typical deployment, the LoRa gateway is a fixed device installed on a radio mast or antenna tower connected to the power line, and the sensors transmit data towards the gateway in one hop, thus implementing a simple tree topology of depth 1 (also known as a start topology), that may cover distances of up to around 15 km in open areas as reported in [30], and which is sufficient for most large deployments. as reported in [30], which is sufficient for most large deployments. However, even these long distances could not be enough to interconnect end devices-to-gateways at rural areas that lack any network infrastructure, at outdoor areas with obstacles between the end devices and gateways (where the wireless signal strength may be reduced resulting into data losses and communication errors) and at indoor scenarios. In such scenarios two main alternatives have been studied: mesh networks as opposite to star topologies and mobile gateways, typically based on UAVs. In [30] the authors address real experiments for three mobile scenarios, where a LoRa device is installed on a rotary lathe, on a dash board of a car and on a boat, respectively, to measure the LoRa/LoRaWAN performance (packet delivery ratio and coverage) under different effects and velocities. Their results conclude that for speeds larger than 40 km/h, the LoRa

performance deteriorates while for low-speed mobility (lower than 25 km/h), the communication is still sufficiently reliable.

Many other works in the literature have analyzed the LoRa/LoRaWAN performance on different real and simulated scenarios in terms of packet delivery ratio, maximum number of sensors supported by a gateway (scalability), coverage range, network density and collisions. For example, in [31] the authors question the scalability provided by LoRa and study its behavior by means of a customized simulator LoRaSim when different network settings are considered. They report that, for a typical smart city deployment (with a selected conservative transmission settings and one only gateway) with sensors sending 20 bytes packets each 16 minutes, a gateway can support 120 sensors per 3.8 ha, which is insufficient for future IoT deployments. These results are, however, very far from the theoretical results provided in [32]–[34].

In most of work above, we noted an emphasis in optimizing the communication from the sensors to the drone, in the scenario that the sensors are limited in power. In our scenario, if the sensors are equipped with an energy-harvesting device they are not battery limited, hence it is the drone that becomes the bottleneck from the energy point of view. Moreover, the application of WSNs in precision agriculture requires that the deployment is planned beforehand with an agronomist that is the main responsible to set the best position of the sensor in the field. As we see in Section IV we assume that sensors are deployed regularly in a grid, since it is a common and quite practical assumption for this scenario.

III. LoRa Specifications and Constraints

LoRa (Long Range) [8] is one of the enabling technologies for Low-Power Wide Area Networks (LP-WANs) [35], which are characterized by a long range communication, a low data rate and a low energy consumption. LoRa is especially targeted for connecting devices in applications that need to send small amounts of data a few times per day over long distances (in the order of kilometers), which fits well with very diverse outdoor monitoring scenarios, as for instance, precision agriculture. In a typical precision agriculture setting, a single device may embeds several different transducers for the monitoring of the crops and of the soil, and it may transmit all the sampled data into one single LoRa frame, provided it does not exceeds the maximum allowed payload. Note that the actual payload size depends on the LoRa configuration and may range from a few tens to a few hundreds of bytes, and may thus contain several sensed data. The reader can consult the work in [36] for a comparison between LoRa and other communications technologies for LP-WANs as are SigFox and NB-IoT. The next subsections describe briefly the LoRa specifications and its major constraints. Note that we do not cover the details of LoRaWan, a network specification that use the physical layer, in the rest of the paper we refer only to LoRa and do not use the LoRaWAN layers.

A. The LoRa physical layer

LoRa uses a wide range of license-free sub-gigahertz radio frequency bands and operates in different bands in different regions depending on the local regulations. For example, in Europe, LoRa operates in the 863 to 870 Mhz band and in United States it operates in the 902 to 928 Mhz frequency band. The frequency band is divided in channels, their number depends in turn on the specific local regulations. In Europe, for example, there exist a total of 10 channels, three of which must be implemented by the LoRa end-devices. These channels are 868.10, 868.30 and 868.50 Mhz, each one with a bandwidth of 125 Khz and a variable bitrate between 0.3 and 5 Kbps.

LoRa uses a proprietary spread spectrum modulation named *Chirp Spread Spectrum* (CSS), where each bit of payload is represented by multiple *chirps*. In LoRa terms, the amount of spreading code applied to the original data signal is called the spreading factor (SF). LoRa modulation has a total of six spreading factors (SF7 to SF12). The larger the spreading factor used, the farther the signal will be able to travel and still be received without errors by the RF receiver, but also the lower bit rate and the larger energy consumption.

Another parameter is the *coding rate* (CR), which is the error correction coding given as A/B, where A is the data block length (A=4) and B is the codeword length; available rates are 4/5, 4/6, 4/7 and 4/8. The data transmission rate (DR) is the amount of data transmitted by unit of time (bps) and it depends on the SF, bandwidth, and CR as expressed by DR = SF$\times\frac{BW}{2^{SF}}\times$ CR, and it ranges between 0.3 and 27 Kbps.

The transmission power ranges between 5 and 23 dBm, where higher power increases the energy consumption. Finally, LoRa specifies the packet format, which is composed of three elements: a preamble (8 symbols by default), an optional header, a variable-length payload and a CRC (1 byte). The preamble enables the receiver to synchronize with the incoming data; the header if exists, includes metadata to inform the CR, the payload length and payload CRC presence. The payload includes the data coded at the CR specified. Note that varying the spreading factor (SF) changes the range of communication between two devices; therefore, in rest of the paper, any change in the communication range between the drone and a sensor deployed on the field, directly expressed as a distance in km, can be interpreted as a proper choice of SF in the LoRa radio of both devices, that is able to support that minimum range, if the value exists. The Table below (from https://lora-developers.semtech.com/) provides some combination of SF, Bit Rate, Range, and Time on air of a packet.

Spreading Factor (For UL at 125 KHz)	Bit Rate	Range (Depends on Terrain)	Time on Air for an 11-byte payload
SF10	980 bps	8 km	371 ms
SF9	1760 bps	6 km	185 ms
SF8	3125 bps	4 km	103 ms
SF7	5470 bps	2 km	61 ms

B. Limitations on the duty cycle

Theoretically, hundreds of devices (sensors and gateways) could use the same channel for data transmission. In order to make a equitable sharing of the channel between the radio-based devices, the governments have regulated, among other

TABLE I
MOST IMPORTANT SYMBOLS USED IN THE PAPER WITH THEIR UNITS.

N	number of sensors	-
g	distance between sensors	meters
x	field width (g multiple)	meters
y	field heigth (g multiple)	meters
r	range of wireless comm.	meters
Δ	transmission period	seconds
A	drone autonomy	seconds
D	path-length of the drone	meters
$v = D/A$	drone's velocity	m/s
d	in-range distance	meters
$T = d/v$	drone time of flight over the sensor	seconds
p_k	probability of receiving a packet	(0,1)
k	drone-sensors distance	meters

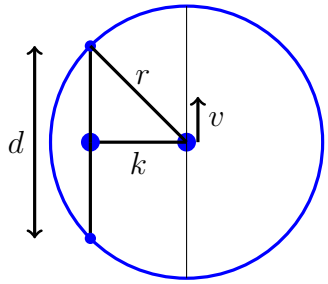

Fig. 1. The picture is centered on the drone that moves vertically with constant speed v. A sensor remains into the communication range for time $T = d/v$. The distance $0 < k < r$ from sensors to drone is the optimal distance that maximize the packet reception probability.

parameters, the maximum *duty cycle* (DC), i.e. the fraction of time during which a device using unlicensed bands can occupy a channel [37]. In Europe, the ETSI EN300.220 standard defines the maximum limits of DC for each sub-band in the spectrum. For example, in the 863 to 870 Mhz frequency band, these limits range between 0.1% and 10%. The duty cycle limit applies to the total transmission time in the period of reference of one hour. For example, with a DC of 1% a LoRa device (both a sensor and a gateway) can transmit up to 36 seconds per hour in each sub-band for each end-device.

In practice, each application may specify the transmission mode by defining the most adequate SF, bandwidth and CR and compute, in advance, the maximum time of transmission for their messages (airtime) based on the calculations of LoRa modulation [38]. This limitation of the duty cycle may reduce not only the suitability of LoRa for some IoT applications (e.g. the ones that require real-time data transmission at high frequencies or the transmission of huge amounts of data) but it may also reduce the amount of sensors that a gateway can successfully support. In Section V, we will use also a LoRa simulator to test the feasibility of using LoRa radios for the data transmission following the model described in the next section.

IV. THEORETICAL MODEL AND ANALYSIS

Let us consider a field of size $x \cdot y$, where a set of N sensors are deployed in a grid, at a distance g from each other (we assume that x and y are both multiple of g), and $N = \left(\frac{x}{g} + 1\right)\left(\frac{y}{g} + 1\right)$. The sensors collect micro-climatic data and data about the state of the soil and surrounding plants. We assume that sensors are wireless-connected, battery-powered, and (possibly) energy harvesting [39], so that they do not require deployment of wired infrastructures as they are completely autonomous. The data collection is performed by a drone that acts as data mule and gateway of the network. It swipes the field according to a predetermined flight plan and it collects the available data from the sensors that are into its communication range. We assume that this range is such that the drone can communicate with sensors placed up to distance r from its projection on the ground. A summary of the symbols and parameters used in this work is presented in Table I

Since the drone passes occasionally over the sensors, each sensor is programmed to log the sensed data in its memory, to aggregate and compress these data, and to transmit the data when the link with the drone is available. We assume that the sensors and the drone are only weakly synchronized, hence the sensors have knowledge of the approximate time of the next flight of the drone, so that they can activate their wireless interfaces when the drone is expected to come. Once activated, they start transmitting periodically their data (according to the duty cycle they can sustain with their radio interface) and stop transmitting either if the drone acknowledges their transmission or when the time frame in which the drone is flying over them ends. Since the sensors are not synchronized among themselves, collisions in transmission are also possible. If at the end of the transmission time frame a sensor was unable to send its data to the drone, it will try to send the data again at the next flight of the drone, along with the new data sensed meanwhile.

Usually the sensors have a low duty cycle, within which each sensor transmits its data with a period Δ. In order to transmit successfully data, the sensor must be in the range of the drone flying over it. The drone has an autonomy of flight of A hours; hence, letting D be the distance covered in its flight plan its average speed is $v = \frac{D}{A}$. Hereafter, we assume the drone always moves at constant speed v. Note that v, D and A are not free variables but are, in fact, constrained. For example, D depends on the size of the field and on the flight plan (in turn also dependent on the wireless coverage of the drone), the autonomy depends on the size of the battery that can be carried by the drone and on its consumption, while the speed depends on the technical specifications of the drone, but it is also limited by the maximum speed that allows the wireless interface to successfully communicate.

While swiping the field, a radio link of length r does not guarantee that we can successfully receive a packet from all sensors in that range, since the delay and the probability of collisions between packets of different sensors must be taken into account. In the following, we study the upper bound k ($k < r$) to the minimum distance between the drone and a sensor such that the probability of receiving a packet p_k can be maximized (we also call k the *maximum minimum distance* of the drone from the sensors). In particular, we analyze the relationship between k and the other parameters in the model, by providing proper bounds to the probability p_k with respect

to k. Figure 1 shows the relationship between k and the time of flight of the drone over a sensor, for a given $k < r$ the drone flies over such a sensor for a time $T = \frac{d}{v}$ which is the time elapsed from when the drone gets in range with the sensor, to the time in which it gets out of range. Since the distance[2] d is $2\sqrt{r^2 - k^2}$ we have that the time of flight over the sensor is $T = \frac{d}{v} = \frac{2A\sqrt{r^2-k^2}}{D}$ and during this period the sensor transmits up to t_x times, where $t_x \geq \lfloor \frac{T}{\Delta} \rfloor$ times, which we lower bound with $t_x \geq \frac{T}{\Delta} - 1$.

Considering that the sensors that transmit the data to the drone are not synchronized, it is possible that some communications collide. This is especially true when the transmission range is very large (this is the case if LoRa is used for example), and thus the number of sensors within the transmission range of the drone is also very large. Furthermore, packet loss is possible due to environmental conditions; we model packet loss with the parameter p_c that expresses the probability that a packet transmitted by a sensor is not received by the drone (see Figure 13 for plots of p_c in LoRa). Consequently, the probability p_k can be estimated as:

$$p_k = 1 - p_c^{T/\Delta - 1}. \tag{1}$$

In general, to assess the value of parameter k, we need to meet the requirement that the probability of successful transmission is at least[3] p', i.e. $p_k \geq p'$. Recalling that $t_x \geq \frac{T}{\Delta} - 1$, follows that $1 - p_c^{T/\Delta - 1} \geq p' \Rightarrow p_c^{T/\Delta - 1} \leq 1 - p'$. Introducing $T = \frac{2A\sqrt{r^2-k^2}}{D}$, under the constraint $p_k \geq p'$ follows:

$$\frac{T}{\Delta} = \frac{2A\sqrt{r^2 - k^2}}{D\Delta} \geq \frac{\log(1-p')}{\log p_c} + 1 \tag{2}$$

Note that if k is very close to r then the time T (numerator above) decreases. Hence, as k approaches r, T goes to 0 and so does p_k. On the other hand, when k is small and v is limited, the length of the path D increases (it is inversely proportional to k intuitively, if the drone flights closer to the sensors, the path length increases) and, based on the autonomy of the drone, it could be infeasible to cover all its length. So, the goals of maximizing p_k and minimizing the path length D are mutually contrasting, and we need to study an optimal trade-off among the two. From inequality (2) follows that:

$$\sqrt{r^2 - k^2} \geq v \cdot \Delta \left(\frac{\log(1-p')}{2\log p_c} + \frac{1}{2} \right)$$

and solving for k we obtain:

$$k \leq \sqrt{r^2 - \left(v \cdot \Delta \left(\frac{\log(1-p')}{2\log p_c} + \frac{1}{2} \right) \right)^2} \tag{3}$$

The maximum of k is obtained from the bound above with $v = v_{max}$, with v_{max} be the maximum speed determined both by the drone specs or by the specs of the communication interfaces that may not work well at higher speeds. Figure 2 shows the maximum k as a percentage of r when: $v_{max} = 11.11$ m/s = 40 km/h, $\Delta = 60$s, $p = 0.98$ and p_c equals to $1\%, 5\%, 10\%, 15\%$ for different curves. We see that when r is small, in particular below 1km, the maximum value of k could be even lower than half the range. The value of k is also a decreasing function of p_c, on the other hand, when r increases over 2 km, k is over 80% of r since the negative part inside the square root is a constant, and it is negligible as r increases. The gap in the lower left part of some curves is where k is not defined: when r is too small, the expression inside the root is negative. Until this point we do not have specified the flight path followed by the drone. In the next subsections, we improve the details of our model. To explicitly introduce the relation between the path length D and k, we define two simple flight plans, one for very large fields and one for medium/small fields.

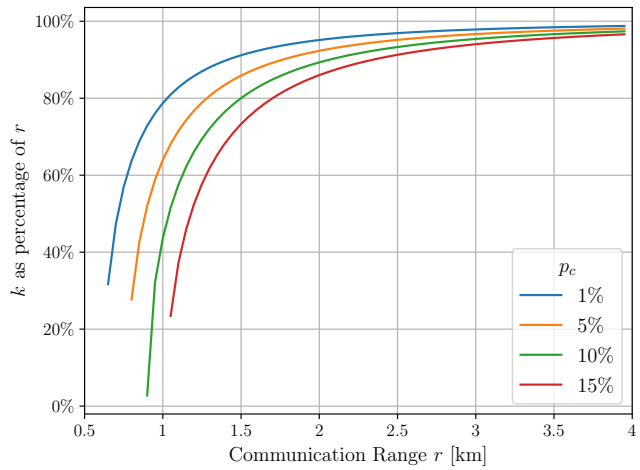

Fig. 2. Maximum value of k as a percentage of the communication range r.

A. A model for large fields

In large fields we assume that $g < r \ll x$ and $g < r \ll y$, which can be obtained if the field is big as compared to the transmission range. This condition in practice can be obtained also if the field is not so big but the transmission range is small (for example when WiFi is used to communicate between the drone and the sensors). The assumed flight plan follows a conventional plan suitable to cover a regular distribution of the sensors in the field. It has the same starting and end point, so that the drone can return to the base after the flight. An example of such a flight plan is shown in Figure 3, it is a *back-and-forth* or *zig-zag* path, studied in several papers (see [40], [41]).

Note that, for the sake of simplicity, in the rest of the section we assume that $\frac{x}{2k}$ is integer and odd, so that the flight plan of the drone is a straightforward generalization of that shown in Figure 3. Note also that the flight plan can be easily generalized to the case in which $\frac{x}{2k}$ is even. If the drone follows the path in Figure 3, the distance traveled by the drone can be computed as follows: The number of movements of the drone along the y-axis are: $\frac{x-2k}{2k} + 1 = \frac{x}{2k}$, The total distance covered

[2]We assume that the drone flights linearly over short distances so we can always approximate d with a linear segment, so we are disregarding turns off the drone.

[3]In a single pass of the drone over the sensor, as in the flight plan the drone may pass over the same sensor more than once.

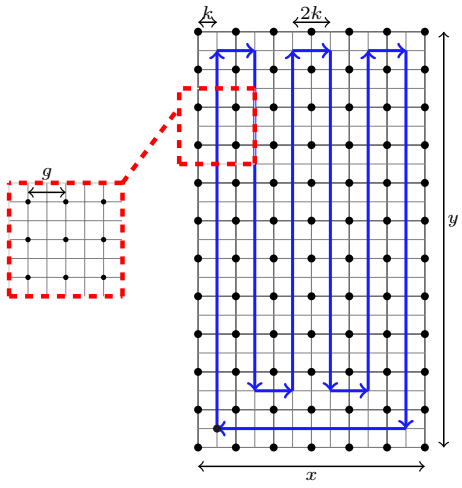

Fig. 3. An example of a *zig-zag* or serpentine path followed by the drone. The sensors are spaced regularly in the field every g meters, the domain is oriented such that $x \leq y$ and the path follow a direction along the y.

by the drone along the y-axis (parallel to the y-axes) is thus $\frac{x}{2k}(y-4k)+4k$; The distance covered by the drone along the x-axis (parallel to the x axis) is: $2(x-2k) = 2x-4k$, hence: $D = \frac{x}{2k}(y-4k)+4k+2x-4k = \frac{xy}{2k}$.

Taking the distance as a constant, we can express the speed of the drone as $v = \frac{xy}{2kA}$, and introducing $T = \frac{d}{v}$, from (2) we obtain:

$$\frac{4kA\sqrt{r^2-k^2}}{xy} \geq \Delta\left(\frac{\log(1-p')}{\log p_c}+1\right) \Rightarrow$$
$$k\sqrt{r^2-k^2} \geq \frac{xy\Delta}{4A}\left(\frac{\log(1-p')}{\log p_c}+1\right)$$

Letting $h = \frac{xy\Delta}{4A}\left(\frac{\log(1-p')}{\log p_c}+1\right)$, follows $k^4-k^2r^2+h^2 \leq 0$. Considering the constraint $r > k > 0$, must be:

$$\sqrt{\frac{r^2-\sqrt{r^4-4h^2}}{2}} \leq k \leq \sqrt{\frac{r^2+\sqrt{r^4-4h^2}}{2}} \quad (4)$$

Now, to maximize p_k we need to take the value of k that maximizes T. The first derivative of T is $\frac{4A(r^2-2k^2)}{xy\sqrt{r^2-k^2}}$, which is null for $k = \pm\frac{r}{\sqrt{2}}$ (under the constraint $r > k > 0$). Since T grows in the range $[0, \frac{r}{\sqrt{2}}]$, $k = \frac{r}{\sqrt{2}}$ corresponds to a maximum of T, and the value of k that maximizes p_k is hence:

$$k = \min\left(\frac{r}{\sqrt{2}}, \sqrt{\frac{r^2+\sqrt{r^4-4h^2}}{2}}\right) = \frac{r}{\sqrt{2}} \quad (5)$$

So, the optimum value of k is a very simple function of r, and insensitive to other parameters of the model. There are some additional considerations. This model determines the value of k aiming at maximizing the probability that the drone receives the transmission from each sensor with a single pass. Actually, for many sensors the drone actually does two passes (remember that the actual transmission range is $r > k$), which means that the probability of receiving the data will be, in general higher. Secondly, the model disregards the fact that the time of flight above the sensors that are at the turns of the drone (the corners of the flight plan) will be smaller than for the other sensors (this means that these sensors will have a smaller probability of successful transmission). Under the assumption that the field is big as compared to the transmission range, the number of sensors in this situation is small. Furthermore, this can be dealt with by letting the drone slow down its speed during the turns, so to give more time to these sensors to communicate. Again, under the assumption that $r \ll x, y$ the effect of these slow-downs would be negligible on the model. In the next model, for small fields, we instead analyze this effect in more detail.

B. A model for small fields

When the field is small, for example, when the communication range r between the sensors on the ground and the drone is very large and is close to at least one of the two dimensions x or y, i.e. x and y are comparable to $k \leq r$, a more accurate model is possible. In this model, the drone performs a rectangular path, as shown in Figure 4 above, and we consider the case $y = \min(x, y) \approx r$, the path length is $2 * (x-2k)+4k$ for a total distance $D = 2x$. Note that in this case we bound the height of the domain by $y \leq 4k$ to ensure an appropriate coverage also of the central part of the field. Similar to the previous model we have the problem of the sensors that are at the corner of the field, that can be at distances greater than k. We solve this problem altering the time velocity of the drone.

For what concerns the time of flight of the drone on the sensors, if we exclude the sensors that are close to the corners of the field, we have that $T = 2\sqrt{r^2-k^2}/v$. For the sensors that are closer to the corners instead, and especially for those that are exactly at the four corners, the time of flight is shorter to compensate to the larger distance to the drone. The situation of these sensors is shown in the bottom part of Figure 4.

Considering the sensor at the corner, the distance covered by the drone while in range of this sensor will be $d' = 2\sqrt{r^2-k^2}-2k$, and the time of flight $T' = \frac{2\sqrt{r^2-k^2}-2k}{v} = T - \frac{2k}{v}$. In order to provide the same coverage of these sensors as well, we assume that the drone stops in each corner for a time equal to $2k/v$, so to ensure a time of flight of those sensors of at least T. To consider these delays into the drone flight autonomy, assuming the drone moves at the slowest possible speed v to cover the entire distance within A (note that this is desirable because a lower speed favors a longer time of flight over the sensors and thus a higher probability of receiving the transmissions), we have: $A = \frac{2x}{v} + 4\frac{2k}{v}$ and then $v = \frac{2x+8k}{A}$. In summary we have:

- $D = 2x$
- $v = \frac{2x+8k}{A}$
- $T = 2\sqrt{r^2-k^2}\frac{A}{2x+8k}$
- $p_c^{T/\Delta-1} \leq 1-p'$ and then $\frac{T}{\Delta} \geq \frac{\log(1-p')}{\log p_c}+1$

From which, introducing T, we obtain:

$$\frac{\sqrt{r^2-k^2}}{(x+4k)} \geq \frac{\Delta}{A}\left(\frac{\log(1-p')}{\log p_c}+1\right) \quad (6)$$

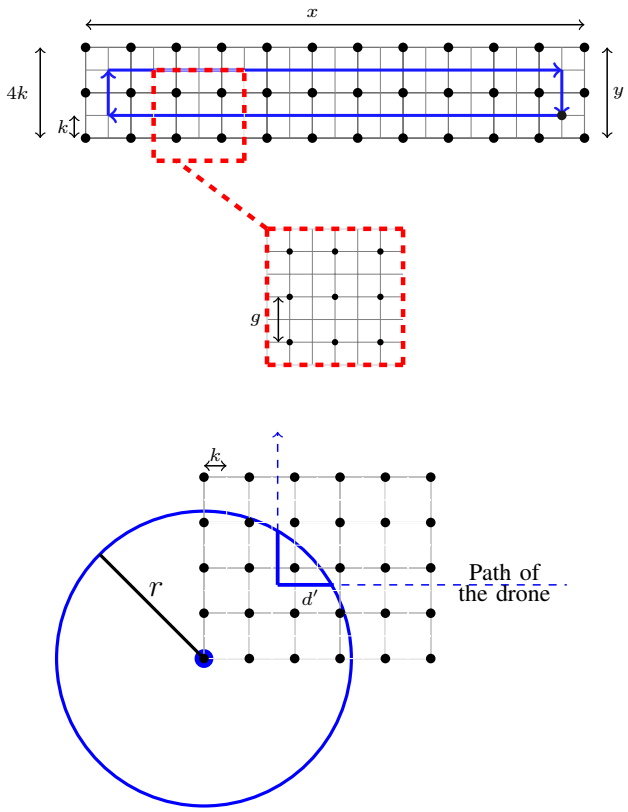

Fig. 4. top) Example of a reduced field to be covered by a drone. The rectangle is created with parameters: $x = 22k$, and $y = 4k$ so $N = 36$. bottom) Distance d' inside the range of a sensor at the corner of the field.

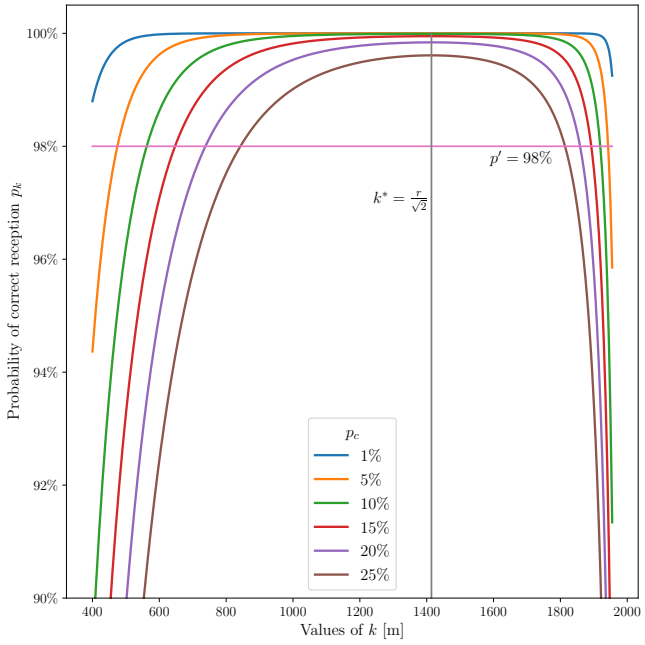

Fig. 5. Large Fields: Probability p_k as a function of k plotted for different values of p_c. Others parameters: $r = 2$km, $k^* = 1414.2$km, $x = 8484$m, $y = 11312$m, $A = 1$ hour, $vmax = 40$km/h.

Letting $h = \frac{\Delta}{A}\left(\frac{\log(1-p')}{A \log p_c} + 1\right)$, follows $(16h^2+1)k^2 + 8xh^2k + h^2x^2 - r^2 \leq 0$. Considering that $r > k > 0$ and that $h > 0$, the latter equation has two real solutions (k_1, k_2) and the admissible values of k are in $[k_1, k_2]$. Also in this case, the value of k that maximizes p_k is the one that maximizes T, considering that the first derivative of T is $-\frac{A(kx+4r^2)}{\sqrt{r^2-k^2}(4k+x)^2}$, by means of a simple analysis (and assuming $x > 4k$), we observe that T is decreasing for $k \in [0, r]$. Hence the maximum of probability p_k is reached when $k = k_1$:

$$k_1 = \frac{-4xh^2 + \sqrt{(16h^2+1)r^2 - x^2h^2}}{16h^2+1} \tag{7}$$

V. Simulations and Evaluation

We validate our approach and the model by means of simulation. Firstly, in Section V-A we evaluate analytically the two models for large and small fields previously described in order to determine how the parameters of the models can be optimized. Secondly, in Section V-B we look for commercial drones to validate their suitability as data collectors in our scenario, and then we face up the usage of LoRa as physical layer technology to evaluate by simulation its performance in terms of collisions and successful reception probability for several deployments and LoRa configurations.

A. Analysis of Cases and Application of the Model

In the following we report some analysis of the two models described in Section IV-A and IV-B for different scenarios. We start with large fields, then move to small ones. In all the analysis the range of communication is considered a constant and set to $r = 2$km, a conservative value for technologies like LoRa. Enlarging the communication range does not alter the qualitative analysis that is obtained here. The parameter p' that affects the expected bound on the probability p_k of receiving a packet from the sensors to the drone, is always set to 0.98.

The plot in Figure 5 represents the values of p_k as a function of k obtained with different values of p_c. The values of x, y are set to be multiple of $2k$, the optimal value of k is equal to 1414.2km that is 70% of r, and represented in the figure as k^*. The others two solutions computed in (4) are shown at the intersection of the curves with the horizontal line with $p_k = p' * 100 = 98\%$, we call these two values k_{min}, and k_{max} in the following. We note from the plots that increasing p_c decreases the values of p_k as expected, but only marginally at k^*, this effect is more emphasized for k_{max} that even if suboptimal will play an important role in practical application as we will see. Choosing $k = k^*$ guarantees that p_k is maximized but it also affects the path length of the drone, and thus the required flight autonomy to cover the field.

Figure 6 shows the minimum flight autonomy A as a function of k (note that for each value of x in the legend, we set $y = 2x$). The figure shows that the autonomy of the drone increases when k decrease and for $k = k^*$ it exceeds 300 minutes (5 hours), a value that is far greater than the higher autonomy of most commercial drones (see Table II). Note also that, in practical applications, it is important to distinguish

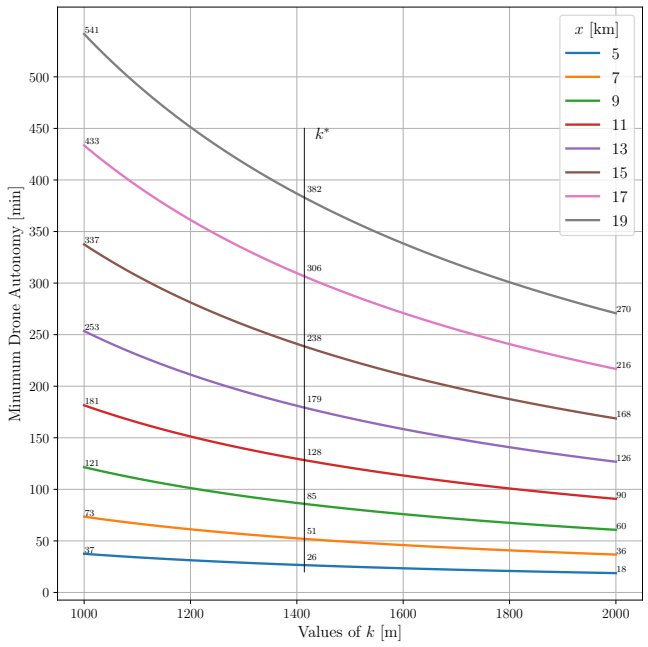

Fig. 6. Analysis of the minimum autonomy of the drone, as a function of k and p_c.

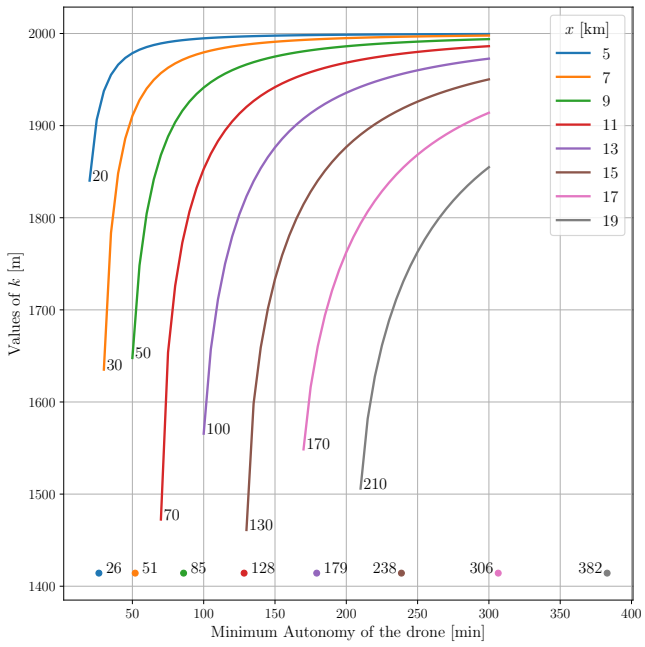

Fig. 7. Analysis of the value of k with different field side x with $y = 2x$ as a function of the autonomy of the drone.

between k and r. In fact, the minimum autonomy that could be obtained by setting $k = r$ would be much lower, but it would not take into account the real path that must be followed by the drone in order to guarantee the desired probability of collecting the data from the sensors. This effect is particular noticeable for large fields, since the slope of the curve for larger values of x is steeper.

From all the analysis done so far, we see that setting $k = k^*$ has several limitation in practice, specifically, if we fix the domain size, the minimum autonomy required for the drone is too high, this can be approached in different ways. A possibility is the use of multiple drones, splitting the domain in smaller parts, another possibility is to consider the data collection phase to be composed of several flight missions that alternate between different part of the domain, still using a single drone. However, a simple solution is also to increase the value of k over k^* as much as possible, i.e. taking $k = k_{\max}$. In Figure 7, we plotted the values of $k_{\max}$ as a function of the autonomy of the drones (in minutes), for different field size, moreover, we can compare the values in the curves, with the values required if we used $k = k^*$ represented as dots at $k^* = 1414.2$. From the plot, we see clearly that using the maximum value for k, reduces the autonomy required to cover a field of 19km×38km from 382 minutes to 210 minutes, a gain of more than 54%.

Now we study the case of small fields, using Equation 7. In this case we set the autonomy of the drone to $A = 40$ minutes and $\Delta = 60$ sec. We set $p' = 0,98$ and $r = 2$ km as in the previous case. Figure 8 shows k as a function of x. Specifically, the curves represent k as a percentage of r for values of p_c ranging from 1% up to 25%. As expected, the plot shows that if we consider x constant, when p_k increases, k decreases (so increasing the time spent in the range of a sensor); the same

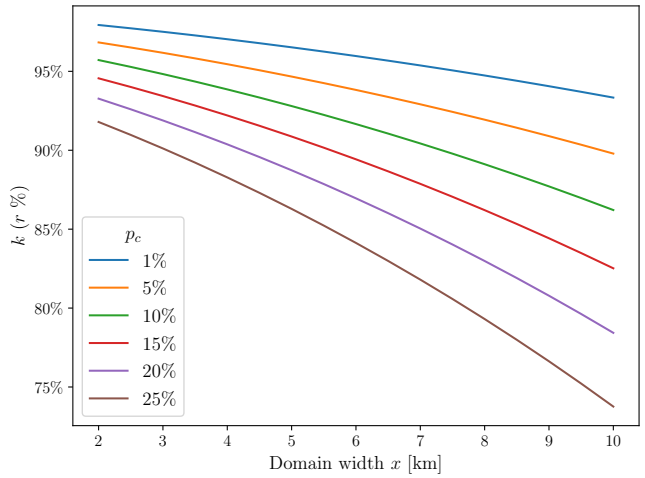

Fig. 8. Values of the maximum-minimum distance k as a function of the field side length x.

effect can be seen if we increase the side of the domain x. The plot in Figure 9 shows that the time T that the drone spends in the range of a single sensor decreases as k increases, but as we see in the next section, it is still sufficiently large with respect to the time required by a packet to be transmitted in LoRa. The time is also a decreasing function of x, since a bigger domain requires an increased velocity of the drone to complete the path.

Finally Figure 10 plots p_k for different values of k and for different packet collision probabilities p_c. The values of the optimal values for k for different values of p_c, obtained using equation 7 are annotated in the plot. As we see from the plot, p_k decreases with increasing p_c and with increasing k, here

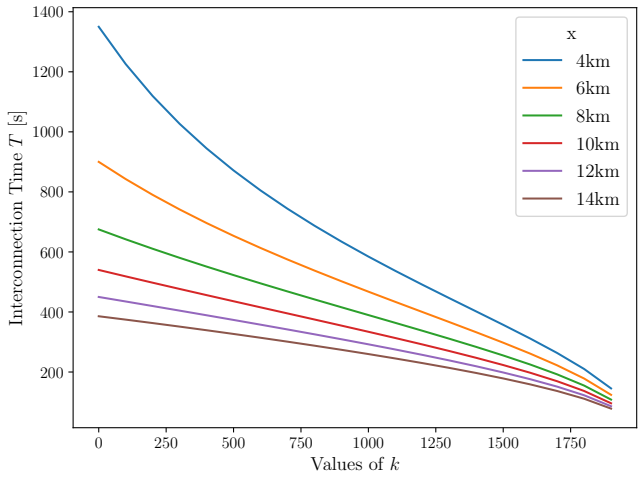

Fig. 9. The time T in second, available for communication between a drone flying over the field and a sensor on the ground as a function of the maximum-minimum distance k.

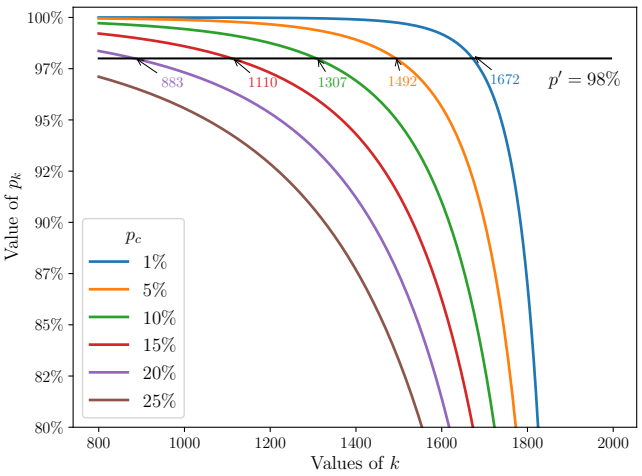

Fig. 10. Value of p_k as a function of different values of k. The optimal value of k are annotated in the plot for different values of p_c

the different model shows a higher sensibility to the value of k, and the optimal values of k ranges from $883 = 44\%$ of r to $1672 = 83\%$ of r.

To conclude, we observe that in small fields we can use lower values of k depending on the values of p_c, in order to optimize p_k, the reduction in k, increases the path length, but since here we are not particularly limited by the drone's autonomy, there is much more flexibility in the choice of k.

B. Case Studies Simulation

We now assess the suitability of the specifications of typical commercial drones and of LoRa radio interfaces for the data collection process described in Section IV. To this purpose, we analytically determine the packet transmission times and the maximum density of the sensors (that is, the maximum number of sensors in a transmission range) that can be served by a drone. Finally, we evaluate by simulation the performance of LoRa in terms of collisions and successful reception probability for several deployments and LoRa configurations.

1) Admissibility Conditions: As described in Section IV, given a field and a drone defined by the tuple $\langle x{\cdot}y, r, A, \Delta\rangle$, we may compute the corresponding optimal deployment defined as $\langle k, N, v, D\rangle$. Such a deployment is admissible, that is, the sensors in the field are all covered by a drone, if two conditions hold: 1) $v \leq v_{max}$, i.e the velocity must be in the range of velocities supported by the drone, and the maximum speed allowed by the wireless interface to successfully communicate (in LoRA, for example, and according to [34] this value is 40km/h); and 2) the time required to cross the path should be lower then A (the autonomy of the drone, as provided in the drone specifications), that is, $\frac{D}{v} \leq A$. Table II reports the specifications of some current commercial drones. Note that the size of the field is, therefore, the key factor that determines the admissibility condition, since it affects directly the parameters v and D.

2) LoRa Feasibility and Performance for Admissible Deployments : Let us assume the drone collects data from N sensors in a field, by using a LoRa communication interface, and that the sensors have a maximum duty cycle of 1%. Concerning the feasibility of the LoRa communication interface to our purposes, we assess two conditions: 1) the time for a packet transmission is lower than the time in which a sensor remains into the communication range of the drone, i.e., $T_{packet} \leq T$; and 2) $T_{packet} \cdot n_{packet} \leq T_{airtime}$ (remind that $T_{airtime}$ is limited to 36 seconds per hour with a DC = 1%). Note that we assume that each sensor performs data aggregation and, thus transmits one only packet of a certain payload during the flight of the drone, i.e. $n_{packet} = 1$.

Figure 11 presents the transmission times T_{packet} for packets with payloads between 10 and 100 bytes (amount of data that a node transmits in an only packet), different SFs and a CR fixed to 4/5. As expected, these times grow with the payload and SF, since the data size to transmit is larger. This coincides with the numbers already reported in many papers that analytically compute the LoRa performance [32], [42]: a larger SF implies a longer transmission time and a longer communication range. In fact, an SF i allows to send 2 times more bytes than an SF $i+1$ in the same time or, alternatively, allows to reduce the time approximately to the half for a same payload. A higher CR implies both a larger overhead and a higher reliability of communications. Note that the times T_{packet} shown in the figure, hold the two conditions that ensure the LoRa feasibility: 1) T_{packet} is always lower than the times T provided in Figure 9 on the right, even for the smallest times that correspond to the largest fields evaluated, with the largest values of x, y and k; and 2) since we assume $n_{packet} = 1$ the second condition also holds. Note, however, that if the sensor tries the continuous transmission of packets within the same transmission period Δ and on the same channel (i.e., $n_{packet} > 1$), for instance due to retries as consequence of collisions or packet losses, the maximum number of packets transmitted should not exceed the upper bound defined in Section III-B. We can conclude here, therefore, that LoRa may be successfully adopted as a communication protocol for the data collection process described in this work.

TABLE II
SPECIFICATIONS OF COMMERCIAL DRONES. **LEGEND:** [1]: DIAGONAL WEELBASE OR LXWXH DIMENSIONS; [2]: WEIGHT INCLUDES PAYLOAD. **NA:** NOT AVAILABLE.

Name	Manufacturer	Type Controller	Dimensions[1] (mm)	Battery (mAh)	Max Speed (km/h)	Weight[2] (Kg)	Max. Flight Time (min)	Price
eBee SQ	senseFly	Fixed Wing	1100	4900 LiPo 3S	110	1.25	55	12000$
Lancaster 5	PrecisionHawk	Fixed Wing	1500	NA	79	3.55	45	25,000+$
Firefly6	BirdsEyeView Aerobotics	Fixed Wing	1520	NA	64	4.5	50-59	8000$
AgDrone System	HoneyComb	Fixed Wing	1245	8000 3S LiPo	82	2.15	55	>10000$
Mavic 2 Pro	DJI	Quadcopter	322x242x84	3950 LiPo 4S	72	1.1	31	1499€
Phantom 4	DJI	Quadcopter	370x325x220	6000 LiPo 2S	72	1.58	28	1153€
3D IRIS+	3D Robotics	Quadcopter	550(L)x100(H)	5100 3S	39.6	1.48	20	750$
Aero Ready to Fly	Intel	Quadcopter	360(L)x222(H)	4000 Li-Po 4S	53.76	2.765	20	1099$
Autel EVO	Autel Robotics	Quadcopter	338	4300 Li-Po	72	1	30	1049$
Anafi	Parrot	Quadcopter	175x240x65	2700 2 cells	55	0.32	25	699$
HYBRiX-2.1	Quaternium	Quadcopter	1630x913x509	Petrol/Battery	80	5	240	NA
H_2Quad	EnergyOR	Quadcopter	1200 (diagonal)	Fuel Cell/Battery	NA	10	120	NA
MATRICE 600 PRO	DJI	Heptacopter	1668x1518x759	6 45 00 LiPo 6S	65	15.1	35	5699$
Xena	OnyxStar	Octocopter	NA	10000/6S	50	5.6	37	NA

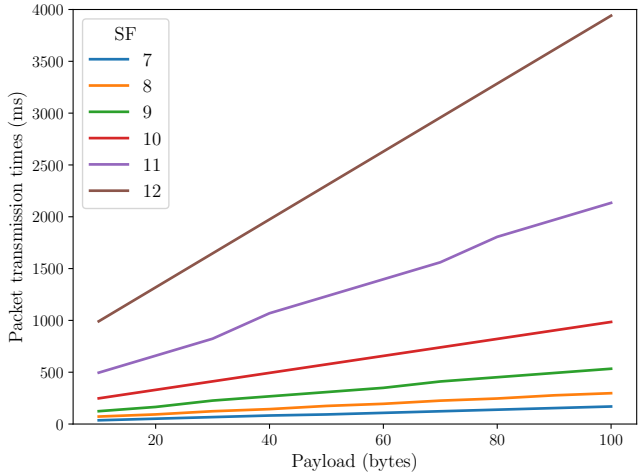

Fig. 11. Packet transmission times for different payloads and SFs, taking CR=4/5.

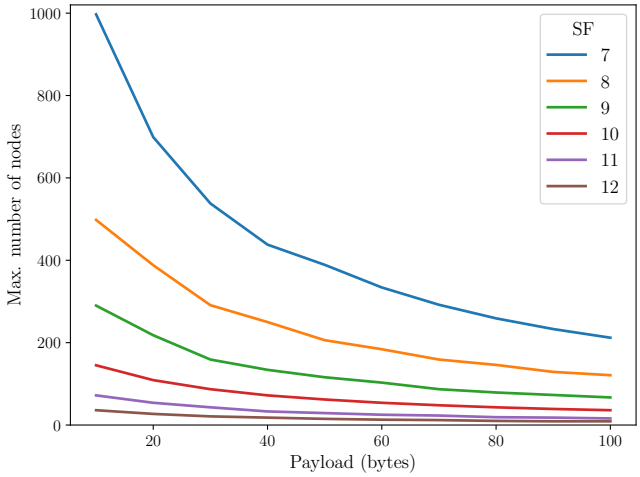

Fig. 12. Maximum number of sensors that can be covered by a single drone for different payloads and SFs, taking CR=4/5.

Let us denote with N_r the maximum admissible number of sensors per transmission range r and let us assume, for the sake of simplicity, that the drone uses for data collection a single channel of 125 kbps. Concerning the maximum density of sensors that the drone can serve, we estimate the maximum amount of packets of a certain payload that the drone is able to receive during its activity cycle (<1%) under perfect synchronization, i.e. without overlapping among packets. Figure 12 shows the maximum number of sensors that can theoretically communicate with the drone, where each sensor transmits exactly a data packet with different size data using a CR=4/5. As expected, the widest coverage is achieved when the payload is minimum and the SF is the lowest. On the other hand, the coverage is minimum, with only 9 sensors, for the maximum payload and the highest SF=12. It is clear, therefore, that an application designer should keep the payload into the limits that ensure that N_r sensors can be heard simultaneously by a single drone

3) Collision Probability: To analyze the probability of collision p_c we have used LoRaFREE[4] [43], a comprehensive LoRa simulator written in Python that supports packet error model, the orthogonality of SFs, the fading impact, the duty cycle limitation at both the sensors and the gateway, retransmisions and acknowledgments, and energy consumption profiling. For our evaluation purposes, we modify this simulator to have an adapted version that simplifies the process of data transmission in order to transmit exactly one packet of data. We have also adapted the target scenarios to simulate a square field of area $x \cdot y$ where the sensors are regularly distributed on a square, keeping a distance between each pair of sensors of $g = 200$ meters. In our experiments we assume that the drone is located at the center of the square (0,0) and it has a radius of r. Table III shows the number of sensors that drop within the radius r,

[4]LoRaFREE: https://github.com/kqorany/FREE

TABLE III
SCENARIOS SIMULATED WITH LORAFREE.

x, y[km]	1	3	4	5	6	7	8	9
r[km]	0.5	1.5	2	2.5	3	3.5	4	4.5
N	36	49	81	121	169	225	289	361
N_r	16	29	49	81	113	149	197	253

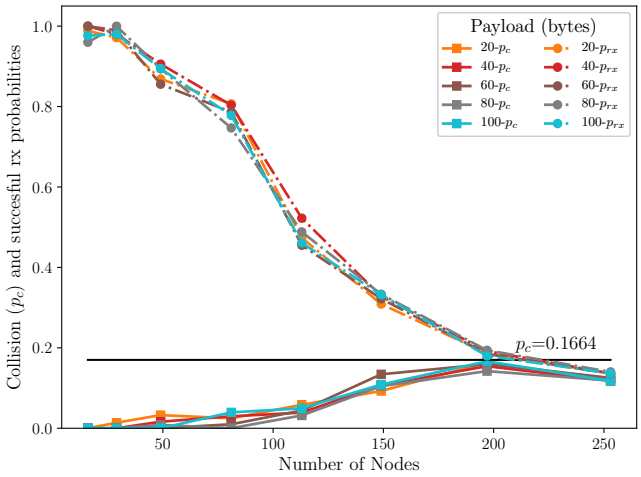

Fig. 13. p_c (lower curves) and probability of successful reception (higher curves) for the deployments in Table III.

denoted as N_r for different fields. Note, therefore, that in our simulation scenario not all sensors are covered by the drone with a radius r, but the optimal paths for large and small fields described in Section IV ensures a complete coverage of sensors. Note, also, that this simulation scenario represents a small fragment of the flight of the drone and, with its movement, the drone enters similar fragments (squares).

In our simulations each sensor transmits one only packet in each transmission period Δ, directly to the drone and in a single hop, using an optimal LoRa radio configuration that selects the lowest SF that guarantees the sensor has a higher received signal strength indicator (RSSI) than the receiver sensitivity, a CR=4/5 and a channel of 125 kbps. After transmitting the packet, each sensor waits a time for an acknowledgment from the drone; if ACK is received, the sensor waits for the next transmission period; otherwise, it retransmits the packet.

A collision occurs at the receiver when two or more LoRa packets overlap at time. We compute the probability of collision p_c as the quotient between the number of packets collided and the total number of packets sent. We also compute the *probability of successful reception*, that is computed as the quotient between the number of successful received packets and the total number of packets transmitted.

The number of packets successful received is affected both by the number of collisions and the unconfirmed traffic (bad formed packets that need to be disregarded, lost packets and lost ack) that generate a new transmission of the original packet, so the probability of successful reception is always lower than $(1 - p_c)$. All simulations were repeated 5 times and, in the following, average values are shown.

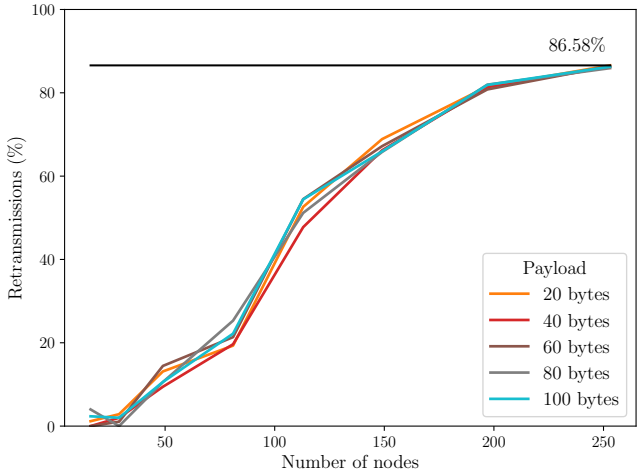

Fig. 14. Percentage of retransmissions for the deployments in Table III.

Figure 13 shows the collision probability p_c (lower curves) and probability of successful reception (upper curves) reported by the simulator for the deployments in Table III. As shown, p_c keeps always under 0.2, it increases with the number of sensors until a certain point when this probability reduces slightly due to the ratio between the number of collisions and the total number of packets transmitted starts to decrease (number of collisions does not increase significantly but the number of packets transmitted is much higher). The probability of successful reception drops (below 0.2 in the worst case) when the number of sensors increases because the load of the network grows, either with more packets and longer packets, and therefore the probability of errors also increases (e.g. due to not only collisions but packet losses or bad formed packets that need to be disregarded). Note that for the smallest deployments this probability is close to 1.0.

Since the probability of successful reception lower than $(1 - p_c)$, this means that, with the current network load managed, the number of collisions is small and it does not represent the main reason for packet retransmissions, which may be due mainly to other causes as previously mentioned. This effect can be observed in Figure 14, where we show the percentage of retransmissions, which increases with the number of sensors, computed as the quotient between the total number of packets transmitted, i.e., original plus retransmissions, and the total number of original packets. For a coverage of $N_r > 100$ the percentage of retransmissions is above 50% and for $N_r > 250$ this percentage achieves 80%, which makes inefficient this network configuration and suggests the need of adding one or more gateways to balance the load network.

VI. CONCLUSIONS AND FUTURE WORK

We address the problem of data collection from sensors for precision agriculture by means of drones acting as data mules. We consider a scenario in which the sensors are deployed in a regular pattern, and the drone, that acts as a mobile gateway, follows a pre-determined path to provide wireless coverage

of the field. This scenario gives rise to a complex trade-off that we solve by introducing an analytic model, according to which it is possible to relate the field parameters (e.g. the field size) with the specs of the communication technology and of the drone (e.g. radius, velocity, autonomy), in order to determine the parameters of the drone path that achieve a desired target probability of successful data collection. The output of our model includes the optimal distance between nodes, the time in which a sensor is in the range of the drone for a successful communication, and the velocity that needs to keep the drone to complete its flight plan. The analytic model is general and can be adapted to different sort/long range communication technologies and to different drone specifications. In particular, it is built over few parameters that are technology-dependent, namely the maximum drone speed and autonomy, the duty cycle of the sensors and the range and packet loss of the communications. These parameters can then be set and analyzed one for all, as we showed in Section V, for a given technology, to adapt the model. We assess the use of the model in a scenario where the drone employs a LoRa communication technology to interact with the sensors in the field and show how the parameters can be optimized in this case. As a final remark, note that our approach is not limited to precision agriculture applications, but may be applied also to other scenarios in in which sensors are deployed in a grid and the data collection is executed by means of a drone that moves along a pre-defined path on the field (similar requirements may be found in the monitoring of large polluted areas for example). However, if the sensors are deployed with a regular structure different than a grid, and the drone moves along a different path, the general methodology of our work remains valid, although the equations would need to be adapted to the specific case.

As further research we plan to analyze scenario with different field shapes, flight plans for the drone, and different deployment of the sensors. We also plan to investigate the feasibility of combining the use of the drone with a classical scheme *sense-store-forward* in the sensors. Other promising future works concern the study of flight strategies for the drone in the context of hybrid data collection architectures that combine drones and fixed base stations, and also the study of flight plan strategies that may meet time constraints in the data collection procedure, so to enable not only off-line data analysis but also timely actuation in response to the field conditions.

ACKNOWLEDGMENT

This work has been partly funded by the EU's Horizon 2020 programme under project SHAPES (GA Nº 857159), by the Spanish Ministry of Economy and Competitiveness under project PLATINO (TEC2017-86722-C4-4-R) and by the Regional Government of Castilla-La Mancha under project SymbIoT (SBPLY/17/180501/000334).

REFERENCES

[1] C. Gomez, S. Chessa, A. Fleury, *et al.*, "Internet of Things for enabling smart environments: A technology-centric perspective," *JAISE - Journal of Ambient Intelligence and Smart Environments*, vol. 11, no. 1, pp. 23–43, Jan. 2019.

[2] S. E. Dìaz, J. C. Pérez, A. C. Mateos, *et al.*, "A novel methodology for the monitoring of the agricultural production process based on wireless sensor networks," *Computers and Electronics in Agriculture*, vol. 76, no. 2, pp. 252–265, 2011.

[3] I. Becker-Reshef, C. Justice, M. Sullivan, *et al.*, "Monitoring Global Croplands with Coarse Resolution Earth Observations: The Global Agriculture Monitoring (GLAM) Project," *Remote Sensing*, vol. 2, no. 6, pp. 1589–1609, 2010.

[4] A. Kocian, D. Massa, S. Cannazzaro, *et al.*, "Dynamic Bayesian network for crop growth prediction in greenhouses," *Computers and Electronics in Agriculture*, vol. 160, p. 105 167, Jan. 2020.

[5] R. Pydipati, T. Burks, and W. Lee, "Identification of citrus disease using color texture features and discriminant analysis," *Computers and Electronics in Agriculture*, vol. 52, no. 1-2, pp. 49–59, 2006.

[6] W. Shen, M. Zhou, F. Yang, *et al.*, "Multi-crop convolutional neural networks for lung nodule malignancy suspiciousness classification," *Pattern Recognition*, vol. 61, pp. 663–673, 2017.

[7] M. Lauridsen, H. Nguyen, B. Vejlgaard, *et al.*, "Coverage comparison of gprs, nb-iot, lora, and sigfox in a 7800 km area," vol. 2017-June, 2017.

[8] LoRa, *LoRa*, https://www.lora-alliance.org, May 2019.

[9] M. Boyle, "The race for drones," *Orbis*, vol. 59, Dec. 2015.

[10] A. Ahmadi, L. Nardi, N. Chebrolu, and C. Stachniss, "Visual servoing-based navigation for monitoring row-crop fields," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 4920–4926.

[11] B. Zhang, W. Fan, X. Xu, and Y. Liu, "FAPAR and BRDF simulation for row crop using Monte Carlo method," in *2014 IEEE Geoscience and Remote Sensing Symposium*, 2014, pp. 812–815.

[12] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '00, Boston, Massachusetts, USA: Association for Computing Machinery, 2000, pp. 56–67.

[13] S. Basagni, A. Carosi, C. Petrioli, and C. A. Phillips, "Coordinated and controlled mobility of multiple sinks for maximizing the lifetime of wireless sensor networks," *Wireless Networks*, vol. 17, no. 3, pp. 759–778, 2011.

[14] A. Chakrabarti, A. Sabharwal, and B. Aazhang, "Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks," in *Information Processing in Sensor Networks*, F. Zhao and L. Guibas, Eds., ser. IPSN'03, Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 129–145.

[15] A. A. Somasundara, A. Ramamoorthy, and M. B. Srivastava, "Mobile Element Scheduling for Efficient Data Collection in Wireless Sensor Networks with Dynamic

Deadlines," in *25th IEEE International Real-Time Systems Symposium*, Dec. 2004, pp. 296–305.

[16] M. Di Francesco, S. K. Das, and G. Anastasi, "Data Collection in Wireless Sensor Networks with Mobile Elements: A Survey," *ACM Transactions on Sensor Networks*, vol. 8, no. 1, pp. 1–31, Aug. 2011.

[17] A. W. Khan, A. H. Abdullah, M. H. Anisi, and J. I. Bangash, "A Comprehensive Study of Data Collection Schemes Using Mobile Sinks in Wireless Sensor Networks," *Sensors*, vol. 14, no. 2, pp. 2510–2548, 2014.

[18] H. Huang, A. V. Savkin, M. Ding, and C. Huang, "Mobile robots in wireless sensor networks: A survey on tasks," *Computer Networks*, vol. 148, pp. 1–19, 2019.

[19] J. Burrell, T. Brooke, and R. Beckwith, "Vineyard Computing: Sensor Networks in Agricultural Production," *IEEE Pervasive Computing*, vol. 3, no. 1, pp. 38–45, Jan. 2004.

[20] C. Wang, F. Ma, J. Yan, *et al.*, "Efficient Aerial Data Collection with UAV in Large-Scale Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 11, 2015.

[21] T. Wu, P. Yang, Y. Yan, *et al.*, "ORSCA: Optimal Route Selection and Communication Association for Drones in WSNs," in *Proceedings - 5th International Conference on Advanced Cloud and Big Data, CBD*, 2017, pp. 420–424.

[22] Y. Zeng and R. Zhang, "Energy-Efficient UAV Communication with Trajectory Optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, 2017.

[23] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 2109–2121, 2018.

[24] D. Zorbas and B. O'Flynn, "Collision-Free Sensor Data Collection using LoRaWAN and Drones," in *Global Information Infrastructure and Networking Symposium (GIIS)*, Oct. 2018, pp. 1–5.

[25] C. Luo, L. Wu, W. Chen, *et al.*, "Trajectory Optimization of UAV for Efficient Data Collection from Wireless Sensor Networks," in *International Conference on Algorithmic Applications in Management*, Springer US, 2019, pp. 223–235.

[26] C. Luo, Y. Wang, Y. Hong, *et al.*, "Minimizing data collection latency with unmanned aerial vehicle in wireless sensor networks," *Journal of Combinatorial Optimization*, pp. 1–24, 2019.

[27] Q. Pan, X. Wen, Z. Lu, *et al.*, "Dynamic speed control of unmanned aerial vehicles for data collection under internet of things," *Sensors*, vol. 18, no. 11, 2018.

[28] S. Sontowski, M. Gupta, S. S. L. Chukkapalli, *et al.*, "Cyber attacks on smart farming infrastructure," in *6th IEEE International Conference on Collaboration and Internet Computing (IEEE CIC 2020)*, 2020.

[29] M. Gupta, M. Abdelsalam, S. Khorsandroo, and S. Mittal, "Security and privacy in smart farming: Challenges and opportunities," *IEEE Access*, vol. 8, pp. 34 564–34 584, 2020.

[30] J. Petäjäjärvi, K. Mikhaylov, M. Pettissalo, *et al.*, "Performance of a low-power wide-area network based on LoRa technology: Doppler robustness, scalability, and coverage," *International Journal of Distributed Sensor Networks*, vol. 13, no. 3, p. 1 550 147 717 699 412, 2017. eprint: https://doi.org/10.1177/1550147717699412.

[31] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do LoRa Low-Power Wide-Area Networks Scale?" In *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '16, Malta, Malta: ACM, 2016, pp. 59–67.

[32] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, *et al.*, "Understanding the Limits of LoRaWAN," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, Sep. 2017.

[33] M. Capuzzo, D. Magrin, and A. Zanella, "Confirmed traffic in LoRaWAN: Pitfalls and countermeasures," in *17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2018, pp. 1–7.

[34] H.-C. Lee and K.-H. Ke, "Monitoring of Large-Area IoT Sensors Using a LoRa Wireless Mesh Network System: Design and Evaluation," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, pp. 2177–2187, 2018.

[35] S. Farrell, "Low-Power Wide Area Network (LPWAN) Overview," RFC Editor, RFC 8376, May 2018, pp. 1–43.

[36] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," *ICT Express*, vol. 5, no. 1, pp. 1–7, 2019.

[37] T. T. Network, *Duty Cycle*, https://www.thethingsnetwork.org/docs/lorawan/duty-cycle.html, Mar. 2021.

[38] SEMTECH, *LoRa Modem Design Guide*, https://www.semtech.com/uploads/documents/LoraDesignGuide_STD.pdf, Jul. 2013.

[39] A. Caruso, S. Chessa, S. Escolar, *et al.*, "A Dynamic Programming Algorithm for High-Level Task Scheduling in Energy Harvesting IoT," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2234–2248, Jun. 2018.

[40] T. M. Cabreira, L. B. Brisolara, and P. R. Ferreira Jr, "Survey on coverage path planning with unmanned aerial vehicles," *Drones*, vol. 3, no. 1, p. 4, 2019.

[41] J. I. Vasquez-Gomez, M. Marciano-Melchor, L. Valentin, and J. C. Herrera-Lozada, "Coverage path planning for 2d convex regions," *Journal of Intelligent & Robotic Systems*, vol. 97, no. 1, pp. 81–94, 2020.

[42] S. Escolar, F. Rincón, X. D. Toro, *et al.*, "The PLATINO Experience: A LoRa-based Network of Energy-Harvesting Devices for Smart Farming," in *Design of Circuits and Integrated Systems, (DCIS). Bilbao (Spain)*, Nov. 2019, pp. 1–6.

[43] K. Q. Abdelfadeel, D. Zorbas, V. Cionca, and D. Pesch, "FREE —fine-grained scheduling for reliable and energy-efficient data collection in lorawan," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 669–683, 2020.