UAV Location Broadcasting with Wi-Fi SSID

Franco Minucci, Evgenii Vinogradov, Hazem Sallouha, Sofie Pollin
ESAT - Telemic
KU Leuven
Email: {name.surname}@esat.kuleuven.be

***Abstract*—To enable widespread and safe use of drone technologies, it is fundamental to design sense and avoid solutions with low-cost and reliable performance. To this end, broadcasting the drone location and speed is an essential safety requirement. This paper presents a method to broadcast short messages within the SSID of a Wi-Fi network. As a relevant example, we propose to use such messages to broadcast the position and speed of UAVs in a fashion that resembles the ADS-B used in Airplanes. Unlike the existing sense and avoid methods, our approach provides two main advantages. Firstly, it operates in a broadcast manner, meaning that no delay is imposed by establishing a connection. Secondly, no custom hardware is required. Nonetheless, a tradeoff must be made between spending time broadcasting messages and data communication using an actual Wi-Fi connection. Therefore, we introduce a practical broadcast and receive protocol which improves receiving reliability and robustness against jamming. To validate our approach, extensive measurements of broadcast delivery rates have been conducted for multiple broadcasting strategies and the results are presented.**

***Index Terms*—Unmanned aerial vehicle (UAV), Wi-Fi, sense, SSID, location broadcasting**

I. Introduction

Unmanned Aerial Vehicles (UAVs), a.k.a drones, have recently been recognized as a favorable solution for daily tasks such as structural monitoring, buildings surveillance and package delivery[1]–[3]. Consequently, UAVs traffic is set to rise introducing serious safety concerns about possible collisions with air-crafts or other UAVs. Air-crafts tackle such a problem by constantly broadcasting their GPS positions using ADS-B. However, ADS-B transceivers are expensive and complex to integrate in UAVs. Furthermore, the spectrum used for ADS-B is already congested around busy airports. This urgent need for an alternative solution to have broadcast and sense functionality in UAVs motivates our study in this article.

Integrating new transceivers in small UAVs is not straightforward due to the various kinds of hardware configurations used. In particular, it is arduous to find a common interface to exchange GPS information. Nonetheless, the vast majority of UAVs uses Wi-Fi for control and multimedia transmission making it a common module. In this work, we propose a simple solution for using the Wi-Fi module to emulate the ADS-B functionality. In fact, each Wi-Fi interface offers a standard mechanism to broadcast information, namely beacon messages [4]. Accordingly, positional information, such as GPS coordinates and speed, can be embedded in the Service Set Identifier (SSID) field of beacon packets.

A vital advantage of using the SSID is the intrinsic simplicity of implementation which requires only a firmware update to get the best performance, it is possible on all Wi-Fi modules on the market and, it works also without optimizing the firmware but with reduced performance. Moreover, broadcasting GPS coordinates using SSID (GPS-SSID) does not require any connection or setup time which typically takes around 10 seconds [5]. Instead, it only requires other UAVs to periodically listen for beacons and broadcast their owns. The GPS typically updates the position once per second. Therefore, we target broadcasting the new position in the same rate.

The performance of Wi-Fi is well investigated in the literature. In [6], [7] and [8], the authors presented an experimental analysis of Wi-Fi with UAVs. It has been shown that the throughput of Wi-Fi drops significantly for distance longer than 500 m. However, looking at the results, one sees that at longer distances the throughput stabilizes around 20Mbps which is more than sufficient to exchange GPS coordinates. Also, mutual interference between ground devices and aerial users may affect the performance of aerial network. However, it can be concluded that Wi-Fi has the sufficient characteristics to broadcast and sense GPS coordinates in terms of range and throughput.

In order to have sense functionality on-board, technologies based on sound [9], radar [10], lidar [11], infrared and vision[12] are used. All these non-cooperative technologies rely on various kinds of sensors to estimate the position of objects in the 3D space and are designed to work as emergency measures [13]. On the contrary, the cooperative avoidance scheme introduced in this paper is more suitable for providing air separation between UAVs and thus preventing the emergency situation. Particularly, in this paper, multiple GPS-SSID broadcasting strategies are presented and characterized considering both half-duplex and full-duplex scenarios. At first, a mathematical model of all strategies is elaborated and secondly the model is validated by experimental results.

The rest of the paper is organized as follows. In

Section II we present the structure of the beacon frames used to broadcast UAV's position and speed. The system model for full-duplex and half-duplex is introduced in Section III and IV, respectively. In Section V we present the experimental results of our approach. Finally, we conclude our paper in Section VI.

II. Background

In Wi-Fi, beacon frames are used by the MAC layer to broadcast management information. Most of the fields in the frame body, can be modified to contain user's data without invalidating the frame. In this work we are interested in embedding the location information in the SSID field because the standard does not pose any limitations on its content. However, since other fields have a specific meaning, putting incorrect data in them may result in unpredictable behavior.

The full beacon frame, including the physical layer fields, is 672 bits long, 480 of which compose the MAC-PDU. The MAC-PDU contains multiple management fields including the SSID. Our payload, the UAV's position and velocity, is encoded to fit in the SSID consisting of 32 bytes (256 bits). To reduce the bandwidth occupied by the GPS messages, the protocol is designed to use 802.11b at 11Mbit/s. Consequently, each beacon frame is $61.1 \times 10^{-5} s$ long (i.e., 0.0611ms).

The ISM band at 2.4 GHz is subdivided by the Wi-Fi standard into 14 channels, only 13 of which are usable in Europe. When using multiple channels it is important to take into account that, every time a receiver hops between channels, some time is lost for tuning the RF chain, elaborate, and decode the received data. The lost time is strictly module dependent. Nevertheless, it does not affect the global system behavior.

III. System Model - Full-Duplex

This section presents the mathematical model upon which the GPS-SSID Broadcasting Protocol is based. In particular, we analyze the situation in which the RF chains of transmit and receive are duplicated and work independently. Subsequently, we introduce the four strategies used to broadcast the position of the UAV within the Wi-Fi SSID in a full-duplex manner.

A summary of all the symbols used in our description and a short explanation of their meaning can be found in Tab. I.

A. Protocol Definition

In standard Wi-Fi, each station uses a single channel but channel hopping schemes can be used to improve reliability and to avoid interference. Accordingly, one defines N_{ch}^{Tx} and N_{ch}^{Rx} to be the number of transmit and receive channels, respectively. Fig. 1 illustrates the timing diagram used to describe the GPS-SSID messaging protocol. Let T_{Rx} denote the time the receiver needs to complete a full scan cycle over all channels.

TABLE I: Summary of the parameters used in the system models

Parameter	Description
N_{ch}^{Tx};N_{ch}^{Rx}	Number of channels used by transmitter and receiver, respectively
N_{Rep}	Number of times a unique message is transmitted on the same channel
P_{Tx}	Duration of a transmission
Δ_{Tx}	Interval between two consecutive transmissions
T_{Tx}	Transmission period of N_{Rep} copies of the same message over N_{ch} channels: P_{Tx} + Δ_{Tx}
P_{Rx}	Time interval in which the receiver is actively listening
T_{Rx}	Time needed to scan all the channels involved in the protocol. Also indicated as scan period: $P_{Rx} + \Delta_{Rx}$
$\Delta_{Rx,proc}$;$\Delta_{Rx,ch}$	Portions of the scan interval in which the receiver is not actively listening but processing packets or handling channel switching respectively
$\Delta_{Tx,ch}$	Minimum possible time in between two consecutive transmissions on different channels
D_{Rx}; D_{Tx}	Respective duty cycle of receiver and transmitter
$\mathcal{P}$	Probability of reception of at least one instance of a repeated message
T_{gps}	GPS coordinates update period
N_{Ts}	Number of slots in which T_{gps} is divided
N_{Tx};N_{Rx}	Number of slots used by transmitter and received

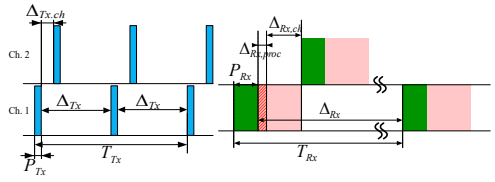


Fig. 1: Visual description of the timing symbols. T_{Rx} is the receiver period, which is the time needed by the receiver to scan all the channels. P_{Rx} is time during which the receiver listens to the channel. Δ_{Rx} is the time in between two consecutive scan cycles. $\Delta_{Rx,proc}$ and $\Delta_{Rx,ch}$ are the time to process the incoming data and the time needed to switch channel respectively. The transmit time T_{Tx} can include multiple transmission cycles. P_{Tx} is the duration of a beacon and $\Delta Tx, ch$ is the time needed to change channel. Δ_{Tx} is the time in between two transmissions cycles.

Within the scan period, only a limited portion of time, denoted by P_{Rx}, is dedicated to listen to the active channel. The remaining time naming the blind time is indicated by Δ_{Rx}. Subsequently, one can write

$$T_{Rx} = P_{Rx} + \Delta_{Rx}, \tag{1}$$

where Δ_{Rx} is defined as:

$$\Delta_{Rx} = (N_{ch}^{Rx} - 1)P_{Rx} + N_{ch}^{Rx}(\Delta_{Rx,ch} + \Delta_{Rx,proc}). \tag{2}$$

In (2), $\Delta_{Rx,ch}$, represents the time needed to switch between two channels and, $\Delta_{Rx,proc}$ is the time spent to process the incoming data. It is worth noting that $(N_{ch}^{Rx} - 1)P_{Rx}$ in (2) represents the listening intervals spent on channels other than the active one.

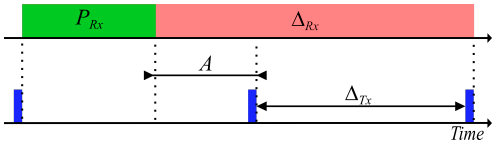


Fig. 2: Rx/Tx timing when $\Delta_{Tx} \geq P_{Rx} - P_{Tx}$. The probability of missing a transmit event can be computed as the probability that a the event happens just outside the receive interval P_{Rx}. This happens when (9) is invalid.

Referring to the timing diagram illustrated in Fig. 1 the receiver duty cycle can be defined as:

$$D_{Rx} = N_{ch}^{Rx} \frac{P_{Rx}}{T_{Rx}}, \tag{3}$$

as a receiver is scanning N_{ch} times during the full scan cycle T_{Rx}.

On the transmitter side, given that each message can be repeated N_{Rep} times, one can also define T_{Tx} that represents the duration of a full transmission cycle. An analogous set of equations can then be written for the transmitter as:

$$T_{Tx} = N_{Rep}(P_{Tx} + \Delta_{Tx}); \tag{4}$$

$$\Delta_{Tx} = (N_{ch}^{Tx} - 1)P_{Tx} + N_{ch}^{Tx}\Delta_{Tx,ch}; \tag{5}$$

$$D_{Tx} = N_{Rep} N_{ch}^{Tx} \frac{P_{Tx}}{T_{Tx}}, \tag{6}$$

where $\Delta_{Tx,ch}$ is the time needed by the transmitter to change channel, P_{Tx} is the duration of a packet and Δ_{Tx} is the minimum interval between two consecutive transmissions.

Now, assuming that transmitter and receiver operate on the same channels, one derives the number of repetitions as:

$$N_{Rep} = \left\lfloor \frac{P_{Rx} + \Delta_{Rx}}{P_{Tx} + \Delta_{Tx}} \right\rfloor, \tag{7}$$

where $\lfloor \cdot \rfloor$ denotes a floor operation.

B. Reception probability analysis

To guarantee that a message is received successfully within one scan period, T_{Rx}, a complete transmission must occur during a listening interval P_{Rx} which implies that:

$$T_{Tx} = T_{Rx}; \tag{8}$$

$$\Delta_{Tx} = \frac{T_{Rx}}{N_{Rep}} \leq P_{Rx} - P_{Tx}. \tag{9}$$

In case conditions (8) and (9) are not satisfied, SSID reception within an Rx cycle is uncertain. This particular situation is described in Fig. 2.

The probability of reception $\mathcal{P}$ is the probability that at least one of the messages transmitted N_{Rep} times is correctly received at a given channel. As shown in Fig. 2, the transmitted signal cannot be received in case it appears during the interval A. Considering the fact that $T_{Rx} = T_{Tx}$ and the natural random shift between the beginning of the intervals, the probability of unsuccessful signal reception is calculated as $\frac{A}{P_{Tx}+\Delta_{Tx}}$. Consequently, the probability of success is then:

$$\mathcal{P} = 1 - \frac{\Delta_{Rx} - (N_{Rep} - 1)(\Delta_{Tx} - P_{Tx})}{P_{Tx} + \Delta_{Tx}}. \tag{10}$$

From (10), we can immediately see that increasing N_{Rep} increases the probability of successful transmission. Note that (10) is valid only when (9) is not satisfied, otherwise $\mathcal{P} = 1$.

C. Strategies

Given the above premises, we can define four different broadcasting strategies (Fig. 3):

- *Strategy A*: Transmitter and receiver operate on the same fixed channel.
- *Strategy B*: Receiver stays on a fixed channel, transmitter spans over all the available Wi-Fi channels.
- *Strategy C*: Transmitter operates on a single channel, receiver scans all the available channels.
- *Strategy D*: Both transmitter and receiver operate on multiple channels.

In the following subsections we detail the four communication strategies implemented in this work.

1) Strategy A: Tx fixed, Rx fixed: In this scenario, both transmitter and receiver operate on the same channel. Since there is no channel switching, $N_{ch}^{Rx} = N_{ch}^{Tx} = 1$ and the minimum Δ_{Tx} is zero while the maximum can be calculated according to (9):

$$0 \leq \Delta_{Tx}^{A} \leq P_{Rx} - P_{Tx}. \tag{11}$$

Considering (9) and (11), we can define the range of N_{Rep} ensuring the successful reception of the transmitted signal as

$$\left\lfloor \frac{T_{Rx}}{P_{Rx}} \right\rfloor \leq N_{Rep}^{A} \leq \left\lfloor \frac{T_{Rx}}{P_{Tx}} \right\rfloor. \tag{12}$$

Inserting in (6) the maximum and minimum N_{Rep} and Δ_{Tx} we obtain the upper and lower bound for the transmitter duty cycle:

$$\frac{P_{Tx}}{P_{Rx}} \leq D_{Tx}^{A} \leq 1. \tag{13}$$

The receiver duty cycle D_{Rx} is the same as in (3). Note that when $N_{ch}^{Rx} = 1$, the interval Δ_{Rx} between two consecutive listening intervals P_{Rx} becomes equal $\Delta_{rx} = \Delta_{rx.proc}$ since the receiver does not have to change the channel. By using the amount of repetitions stated in (12), it is possible to achieve a probability of successful reception $\mathcal{P} = 1$.

While this strategy is the simplest, it implies coordination between transmitter and receiver because they both need to know on which channel the other is operating. Moreover, it is not possible just to fix a channel by design because it may not be usable due to congestion or interference or it may be forbidden in some countries. This coordination may not always be possible, (e.g., if

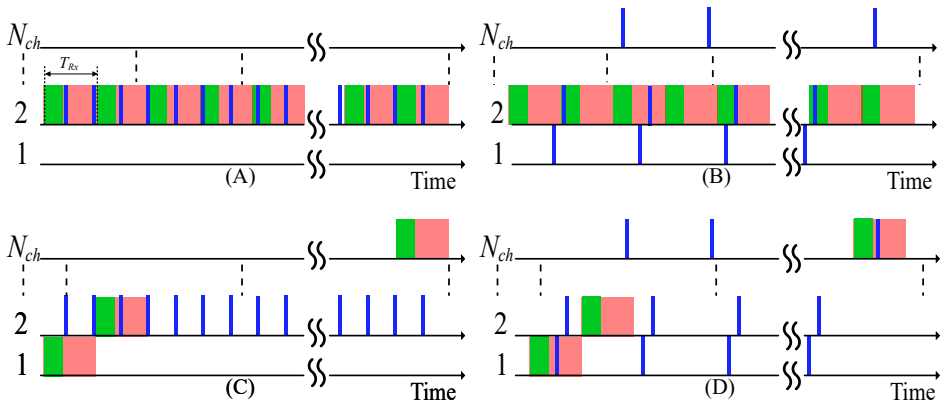


Fig. 3: Possible strategies for scanning and broadcasting: (A) Tx and Rx operate on the same channel; (B) Tx hopping, Rx on a fixed channel; (C) Tx on a fixed, Rx hopping; (D) Tx and Rx both hopping.The red and green square represent the time spent by the receiver on each channel. The green part is the interval P_{Rx} when the receiver is actually listening to the channel. The red squares represent the portion of scan interval Δ_{Rx} not used for listening. This time is used to tune the receiver (e.g. PLL, filters) and to process the incoming packets. The vertical lines represent transmission events P_{Rx}.

the two UAVs belong to different owners) reducing the probability of a successful reception from 1 to $\mathcal{P} = \frac{1}{N_{ch}}$. Following strategies overcome this disadvantage.

2) Strategy B: Tx hopping, Rx fixed: In this case, the transmitter is hopping from one channel to the other sequentially, while the receiver stays fixed on the same channel. As shown in Fig. 1, transmissions on consecutive channels overlap with the Δ_{Tx} interval of other channels. When two consecutive transmissions of the same message are on two different channels, there must be a short period of time $\Delta_{Tx,ch}$ between them, in order to properly tune the transmitter. The expression for the transmit time T_{Tx} per channel does not differ from (4). The minimum Δ_{Tx} can be derived from (6) by posing $N_{ch}^{Tx} > 1$, depending on how many channels are used as:

$$\Delta_{Tx} = N_{Rep}[(N_{ch}^{Tx} - 1)P_{Tx} + N_{ch}^{Tx}\Delta_{Tx,ch}]. \quad (14)$$

If all possible Wi-Fi channels are used, then $N_{ch} = 13$ (14 in USA). Substituting the maximum and minimum Δ_{Tx}, we can get the upper and lower bound for N_{Rep}

$$\left\lfloor \frac{T_{Rx}}{P_{Rx}} \right\rfloor \leq N_{Rep}^{B} \leq \left\lfloor \frac{T_{Rx}}{N_{ch}^{Tx}(P_{Tx} + \Delta_{Tx,ch})} \right\rfloor. \quad (15)$$

Beware though that, in our discussion, repetitions are defined per channel. This means that transmitting the same message once on every Wi-Fi channel, still counts as one repetition. In (15), we are stating that $\Delta_{Tx,ch}$ must be kept as small as possible for the inequality to be verified. The maximum and minimum transmitter and receiver duty cycles (D_{Tx} and D_{Rx}) can be derived from (6) and (3) respectively. In particular D_{Tx} we get:

$$\frac{N_{Rep}N_{ch}^{Tx}P_{Tx}}{P_{Rx}} \leq D_{Tx}^{B} \leq \frac{P_{Tx}}{P_{Tx} + \Delta_{Tx,ch}}, \quad (16)$$

$$D_{Rx}^{B} = \frac{P_{Rx}}{P_{Rx} + \Delta_{Rx,proc}}. \quad (17)$$

This strategy overcomes the problem of coordinating receiver and transmitter. The receiver can autonomously decide on which channel it is better to listen (e.g., to avoid interference) but the transmitter need more energy because it needs to re-transmit the same message multiple times on different channels.

3) Strategy C: Tx fixed, Rx hopping: Another way to avoid coordination between transmitter and receiver is by having the transmitter choosing a single channel but having the receiver to perform a scan of all channels. This is analogous to what is performed in standard Wi-Fi passive scan operation to search for all visible networks.

In this strategy, two consecutive listening intervals P_{Rx} are separated by Δ_{Rx} which is significantly longer than in the case of only one receiving channel as it can be seen from (2). First, unlike in the case of $N_{ch}^{Rx} = 1$, the time needed to tune the receiver for receiving on another channel cannot be ignored. Second, $(N_{ch}^{Rx} - 1)$ listening intervals P_{Rx} are contributing in Δ_{Rx}. Expression (2) quantifies the increase of Δ_{Rx}.

In absence of coordination between receiver and transmitter, each message need to be repeated many times, such that it can cover a full scan period, for $T_{Tx} = T_{Rx}$ to be true. For this strategy, $0 \leq \Delta_{Tx} \leq P_{Rx} - P_{Tx}$, which is the same as (11) for Strategy A.

The required number of repetitions N_{Rep}^{C} can be calculated as:

$$\left\lfloor N_{ch}^{Rx} \frac{P_{Rx} + \Delta_{Rx,proc} + \Delta_{Rx,ch}}{P_{Rx}} \right\rfloor \leq N_{Rep}^{C} \leq \left\lfloor N_{ch}^{Rx} \frac{P_{Rx} + \Delta_{Rx,proc} + \Delta_{Rx,ch}}{P_{Tx} + \Delta_{Tx}} \right\rfloor, \quad (18)$$

where N_{ch}^{Rx} is the number of channels scanned by the receiver. With the constraint over Δ_{Tx} expressed by (9). The meaning of (18) is that the transmitter needs to repeat the same message at least N_{ch}^{Rx} more times than for strategies A and B. From (2) and (18), is evident

that $\mathcal{P}$ can still be 1 if the transmitter sends enough repetitions to compensate for the large Δ_{Rx}. The duty cycle of the transmitter D_{Tx} is the same as (13). The duty cycle of the receiver D_{Rx}, on each channel, is smaller than what we find in (3) and can be calculated as:

$$D^{C}_{Rx,ch} = \frac{P_{Rx}}{N^{Rx}_{ch}(P_{Rx} + \Delta_{Rx,proc} + \Delta_{Rx,ch})}. \quad (19)$$

The global duty cycle, that takes into account all the receive cycles on all the channels, is correctly expressed by (3).

The main drawbacks of this strategy are thus the high amount of repetitions needed and the latency, because the delay between two coordinate updates is significantly higher than for one receive channel is lower due to the additional delay $\Delta_{rx.ch}$ introduced by channel hopping.

4) Strategy D: Tx hopping, Rx hopping: This strategy overcomes the main weakness of *Strategy C*, which is the very long time between two consecutive scan intervals on each channel. If the receiver scans more than one channel, it is wise for the transmitter to also transmit on more than one channel and to make use of the otherwise lost scan intervals. This reduces the amount of repetitions compared to *Strategy C* because in (7), $N^{Tx}_{ch} = N^{Rx}_{ch} = N_{ch}$.

Assuming that receiver and transmitter use the same number of channels:

$$\Delta_{Rx} = N_{ch}(\Delta_{Rx,ch} + \Delta_{Rx,proc}); \quad (20)$$

$$\Delta_{Tx} \geq N_{ch}\Delta_{Tx,ch}; \quad (21)$$

$$\left\lfloor \frac{P_{Rx} + N_{ch}\Delta_{Rx}}{P_{Rx}} \right\rfloor \leq N_{Rep} \leq \left\lfloor \frac{P_{Rx} + N_{ch}\Delta_{Rx}}{P_{Tx} + N_{ch}\Delta_{Tx,ch}} \right\rfloor. \quad (22)$$

The respective duty cycles do not change from (3) and (6) thanks to the fact that receiver and transmitter use the same channels. The main drawback of this strategy is that, like for *Strategy C*, the received messages are only available for the software to be processed when the scan cycle on all channels is complete, which is much slower than strategies *A* and *B*. The probability of receiving a message, expressed in (10), needs to be adapted to take into account that transmit and receive events occur on multiple channels.

$$\mathcal{P}_D = 1 - \frac{N_{ch}(\Delta_{Rx,ch} + \Delta_{Rx,proc})}{P_{Tx} + \Delta_{Tx}} - \frac{\frac{N_{ch}(N_{ch}+1)}{2}(P_{Tx} + \Delta_{Tx,ch}) - \Delta_{Tx}(N_{Rep} - 1)}{P_{Tx} + \Delta_{Tx}}. \quad (23)$$

IV. System Model - Half Duplex

Practical embedded Wi-Fi modules used for UAVs do not offer separate RF chains to receive and transmit, meaning that transmit and receive operations cannot be performed concurrently. Therefore, it is not possible to directly rely on the four strategies described in the previous sections in practical scenarios where a node should both transmit GPS updates and receive updates from other nodes. In this section we describe a new asynchronous messaging protocol based on the schemes presented before.

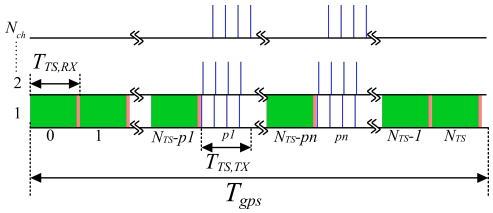


Fig. 4: TDD-mode: T_{gps} is split into N_{TS} time slots.

The main goal is to exchange GPS messages between UAVs, so one can start by defining a basic period T_{gps}, which is the period at which the GPS provides new coordinates. For typical GPS modules available on the market, $T_{gps} = 1$ s, and a protocol needs to be designed so that each GPS update can be transmitted to neighboring receiving nodes within such period T_{gps}.

From the analysis in Section III, *Strategy B* is the best candidate for building the Time Division Duplex (TDD) scheme because it does not require coordination between transmitter and receiver and offers low latency combined with the maximum D_{Rx}.

The T_{gps} can be split into N_{TS} time-slots that are allocated to listening or transmitting operations.

For simplifying the analysis, the duration of a transmit slot $T_{TS,Tx}$ is kept the same as the duration of a receive slot $T_{TS,Rx}$ ($T_{TS,Rx} = T_{TS,Tx} = T_{TS}$).

T_{gps} can then be written as:

$$T_{gps} = (N_{TS,Rx} + N_{TS,Tx})T_{TS}; \quad (24)$$

where $N_{TS,Rx}$ and $N_{TS,Tx}$ are the number of receiving and transmitting slots respectively and, $T_{TS,Rx}$ and $T_{TS,Tx}$ are defined as:

$$T_{TS,Rx} = P_{Rx} + \Delta_{Rx,proc}; \quad (25)$$

$$T_{TS,Tx} = N_{Rep}(P_{Tx} + \Delta_{Tx}). \quad (26)$$

Within T_{gps}, $N_{TS,Tx}$ and $N_{TS,Rx} = N_{TS} - N_{TS,Tx}$ time slots are assigned to transmit and receive operations, respectively. The transmitting slots are randomly interleaved with receiving slots within the period T_{gps}.

Once the number of time slots is fixed, one needs to maximize the number of repetitions of each message, without exceeding the duration of a transmit slot. Also, to guarantee the correct reception of each transmission, (9) still holds. Thus, Δ_{Tx} needs to be the minimum possible, which, according to (5), is $N_{ch}\Delta_{Tx,ch}$.

During a T_{gps} period, each message is repeated N_{Rep} times and the repetitions are equally distributed among

the transmit time slots (e.g. for $N_{Rep} = 8$ and $N_{TS,Tx} = 2$, each message is repeated 4 times in the first time slot and 4 times in the second time slot). The number of repetitions in each time slot is then:

$$N_{Rep}^{Ts} = \left\lfloor \frac{N_{Rep}}{N_{TS,Tx}} \right\rfloor. \quad (27)$$

Keeping in mind that transmit and receive slots have the same duration, the maximum N_{Rep} can be calculated as the ratio between the time spent listening and the duration of one transmission:

$$N_{Rep} = \left\lfloor \frac{T_{gps} - N_{TS,Rx}(P_{Rx} + \Delta_{Rx})}{N_{TS,Tx} N_{ch}^{Tx}(P_{Tx} + \Delta_{Tx,ch})} \right\rfloor. \quad (28)$$

Since the module cannot continuously broadcast, but it also needs to listen for incoming messages, it is important to evaluate the transmit duty cycle and its boundaries. The duty cycle is the ratio between the time spent transmitting and the total period:

$$D_{Tx} = \frac{T_{Tx}}{T_{gps}} = \frac{N_{TS,Tx} N_{Rep} N_{ch} P_{Tx}}{(N_{TS,Rx} + N_{TS,Tx}) T_{TS}}. \quad (29)$$

The transmit duty cycle is then minimum when $N_{TS,Tx} = 1$ and maximum when $N_{TS,Tx} = 15$, since N_{Rep} is fixed by (28). The two extremes, namely $N_{TS,Tx} = N_{TS}$ and $N_{TS,Tx} = 0$ do not make much sense.

$$\frac{P_{Tx} N_{Rep} N_{ch}}{T_{gps}} \leq D_{Tx} \leq \frac{(N_{TS} - 1) N_{Rep} N_{ch} P_{Tx}}{T_{gps}}. \quad (30)$$

It is important to keep in mind that the Wi-Fi modules are not synchronized among each other nor they implement any feedback mechanism. The unlikely scenario in which two modules are perfectly aligned in time is the worst case scenario, since if the two modules use the same combination of time slots, the probability of missing is 1.

The probability that a module uses a particular pattern ('certain positions of the transmitting and receiving time slots 'or' certain combination of the transmitting and receiving time slots positions') can be calculated as:

$$\mathcal{P}_{Ts} = \frac{1}{C}, \quad (31)$$

where C, which is the total number of possible combinations of Tx and Rx time slots, is:

$$C = \frac{N_{TS}!}{N_{TS,Tx}!(N_{TS} - N_{TS,Tx}!)}. \quad (32)$$

When two modules are time-aligned, in order to receive a message, the two modules must use different patterns. Since modules are independent, the probability of them using the same pattern is:

$$\mathcal{P}_{Ts,2} = \mathcal{P}_{Ts} \cdot \mathcal{P}_{Ts} = \mathcal{P}_{Ts}^2. \quad (33)$$

So the probability that the two modules are using different combinations and thus the message is received is:

$$\mathcal{P}_2 = 1 - P_{Ts}^2. \quad (34)$$

Because nodes are not synchronized among each other, it can happen that a transmit slot of one module is only partially overlapping to a receive slot of another module. In this case, the message can still be received correctly, as long as at least one transmit event of the first module happens during a receive interval of the second module.

In the next section we provide an experimental validation of our analysis with non synchronized nodes.

V. GPS-SSID EXCHANGE EXPERIMENTAL VALIDATION

In this section, we describe the experiments we performed in order to validate the analysis carried on in section III-A.

A. Hardware and Software

The Wi-Fi modules used for our experiments are ESP8266 by Espressif [14]. To validate the different strategies *A* to *D*, two modules are used, one configured only as receiver and the other only as transmitter. The ESP8266 are programmed using the Arduino IDE and the ESP8266WiFi framework [15]. The modules report the results of their scan to a computer using a UART over USB port. To measure the packet loss in our lab experiments, we do not exchange GPS coordinates but messages with fixed format, containing consecutive numbers. The modules are configured to use 802.11b at 11MB/s.In Tab. II, we present the system parameters we measured for the ESP8266. Each strategy has been tested with 10000 unique messages.

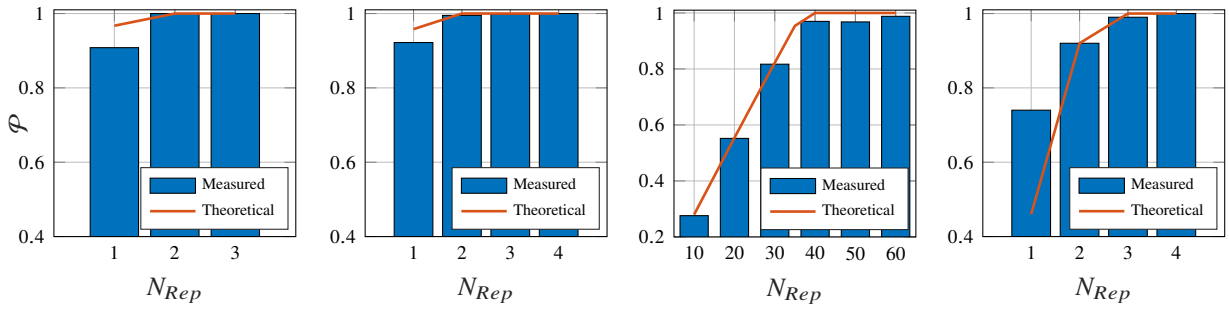
All the measurements have been performed in our offices and labs where other Wi-Fi networks are present. The modules where placed in multiple positions from 30 cm up to 2 m apart.

B. Experimental results

It is evident how D_{Tx} is quite different from the the theoretical maximum expressed by (5). This is due to the practical limitations of the physical devices. In fact, there should always be at least a delay of 1 ms between two transmissions, which translates into a minimum $\Delta_{Tx} = 1\ ms$. Each call to the delay function, allows the processor to pause the user task to execute systems tasks like the Wi-Fi stack. It is also worth noting that the "delay()" function provided by the API accepts only integer values in milliseconds.

The values reported are an average over multiple measurements.

1) Strategy A: Tx fixed, Rx fixed: According to the theoretical model, this strategy is the most efficient since with only two repetitions, it can already deliver $\mathcal{P} = 1$. The experimental results follow the theoretical model even if they are affected by the imperfections of the physical device. The Δ_{Rx} has been measured to be only approximately 2 ms, resulting in a receiver duty cycle $D_{Rx} = 0.967$. The experimental results and the



(a) Strategy A (b) Strategy B (c) Strategy C (d) Strategy D

Fig. 5: The probability of reception $\mathcal{P}$ vs different number of repetitions N_{Rep} for the four different strategies. As predicted by the theoretical model, *Strategy (A)* and *Strategy (B)* offer the best performance while *Strategy (C)* the worst. In red, the result for the theoretical model is shown to match very closely the measurement results.

TABLE II: Experimental Wi-Fi module parameters. The maximum and minimum duty cycles (D_{Rx} and D_{Tx} are measured respectively for the minimum and maximum Δ_{Rx} and Δ_{Tx}).

Parameter	Str. A	Str. B	Str. C	Str. D
N_{ch}^{Rx}	1		14	
N_{ch}^{Tx}	1	14	1	14
$\Delta_{Tx,ch}$	0 ms	1 ms	0 ms	1 ms
min(Δ_{Tx})	1 ms	14.74 ms	1 ms	14.74 ms
T_{Rx}	62 ms		2124 ms	
$\Delta_{Rx,proc}$	2 ms			
$\Delta_{Rx,ch}$	0 ms		90 ms	
P_{Tx}	0.061 ms			
P_{Rx}	60 ms			
max(D_{Tx})	0.0575			
min(D_{Tx})	$5.74 \cdot 10^{-5}$	$8.04 \cdot 10^{-4}$	$1.14 \cdot 10^{-3}$	$1.2 \cdot 10^{-3}$
D_{Rx}	0.967		0.395	

theoretical success probability against N_{Rep} are shown in Fig. 5 (a).

The first point for $N_{Rep} = 1$ is the only one that shows a significant difference from the theoretical prediction ($\mathcal{P} = 0.91$ instead of 0.967) because, being each message transmitted only once, timing jitter and inaccuracies, and interference have a higher influence on the measurement. With $N_{Rep} > 1$, these effects are smeared over multiple points reducing their effect. This is valid also for all the other strategies.

2) Strategy B: Tx hopping, Rx fixed: The receiver duty cycle and theoretical success probability remains the same as for Strategy A. However, the transmitter duty cycle changes due to channel hopping (see Tab. II).

Experimental results show that $N_{Rep} = 4$ is necessary for $\mathcal{P} = 1$, while the theory predicts only $N_{Rep} = 2$. (Fig. 5). However, for $N_{Rep} = 2$ and 3, the measured $\mathcal{P}$ is 0.995 and 0.998 respectively.

The minimum Δ_{Tx} measured on the ESP8266 is 14.74 ms, as shown in Tab. II.

3) Strategy C: Tx fixed, Rx hopping: Strategy C is the least efficient, because the transmitter needs to repeat the same message many times to compensate for the very long Δ_{Rx} which has been measured to be approximately 2062 ms. Furthermore, since the receiver makes the result of a scan available only when all channels are scanned, the communication latency is equal to the scan time for all the channels. For the ESP8266, the full scan time is approximately 2124 ms as reported in Tab. II, which is too long compared with the GPS period T_{gps} of 1 s..

As shown by Fig. 5, between 40 and 45 repetitions are needed to get a theoretical $\mathcal{P} = 1$ (measured $\mathcal{P} = 0.989$).

The measured $\Delta_{Rx,ch}$ is 90 ms. What happens in this period is not disclosed in the documentation from Espressif. However, it is reasonable to say that the ESP's CPU may use this time for running system tasks, like memory management and data processing. However, this strategy is inherently inefficient because scanning all the available channels will always be slower than scanning a single one, independently from which hardware is used.

4) Strategy D: Tx hopping, Rx hopping:: When both Tx and Rx are hopping, the performance is better than for *Strategy C*. With $N_{Rep} = 4$, it is possible to obtain a measured $\mathcal{P} = 1$. Since the transmitter uses multiple channels, the probability of a transmission event happening during a listening interval is higher than it is for *Strategy C*. However, this strategy still suffers from the same latency observed in *Strategy C*, which is its main weakness.

C. Shared RF chain - TDD mode

In Tab. III, the results obtained are reported. The parameters of this method were set as follows: $N_{TS} = 16$, $N_{TS,Tx} = 2$, $N_{TS,Rx} = 14$. For each transmit slot $N_{Rep}^{Ts} = 4$, which is the maximum amount of repetitions that fits in 60 ms, $\Delta_{Tx} = 14.854$ ms, and $\Delta_{Tx,ch} = 1$ ms.

While it is true that $N_{TS,Tx} = 2$ is not optimal, it still provides a high theoretical probability of success

P_{Ts} = 0.9961 with a the transmit duty cycle is kept very low (D_{Tx} = 0.0142) reducing the probability of interfering with other transmitters and saving energy.

In the experiments performed, More than 99% of transmitted messages are constantly received. Even when three modules are involved in the exchange, the percentage of received messages is consistently above 99%. TDD mode is the best candidate to be used with smaller,

TABLE III: Results obtained with 2 and 3 nodes in TDD-mode

Results for 2 nodes	
Node A	Node B
99.04%	99.33%
Results for 3 nodes	
Node A	
Node B	Node C
99.48%	99.45%
Node B	
Node A	Node C
99.88%	99.43%
Node C	
Node A	Node B
99.80%	99.80%

less capable modules that cannot make use of full featured Wi-Fi modules.

VI. Conclusions and Future Work

From our results we can conclude that SSID messaging can be a viable solution for low latency coordinate exchange for collision avoidance. While this messages system is not reliable for high risk missions (class 1), it makes a lot of sense for low risk missions performed by class 2 UAV, where limited hardware capabilities and battery size make impossible to adopt higher end solutions like ADS-B. The main advantage of SSID messaging is that it relies on simple hardware that is already present in many small UAVs thus being very inexpensive to implement and deploy. The main disadvantage is that the performance are limited compared to a more sophisticated technology such as 802.11p which in turn requires special hardware and software.

Among the different strategies that we tested, *Strategy B*, in which messages are broadcasted on all the available channels while the receiver sits on a single channel, resulted to be the best when the Wi-Fi adapter can use two separate receive and transmit chains. This strategy can be easily adapted to the case in which receiver and transmitter must share the RF Chain. While the results obtained in our labs are very encouraging, further research is needed to evaluate the outdoor performance of the GPS-SSID-Broadcast Protocol and its interoperability with normal Wi-Fi communication.

Acknowledgment

This work is part of a project that has received funding from the SESAR Joint Undertaking (JU) under grant agreement No. 763702. The JU receives support from the European Unions Horizon 2020 research and innovation programme and the SESAR JU members other than the Union.

References

[1] P. Liu, A. Y. Chen, Y.-N. Huang, J.-Y. Han, J.-S. Lai, S.-C. Kang, T. Wu, M.-C. Wen, M. Tsai, *et al.*, "A review of rotorcraft unmanned aerial vehicle (UAV) developments and applications in civil engineering," *Smart Struct. Syst*, vol. 13, no. 6, pp. 1065–1094, 2014.

[2] S. Keyworth and S. Wolfe, "UAVS for land use applications: UAVs in the civilian airspace institution of engineering and technology," in *IET Seminar on UAVs in the Civilian Airspace*, pp. 1–13, March 2013.

[3] H. Sallouha, M. Mahdi Azari, and S. Pollin, "Energy-constrained uav trajectory design for ground node localization," in *2018 IEEE GLOBECOM*, Dec 2018.

[4] "IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pp. 1–3534, Dec 2016.

[5] C. Pei, Z. Wang, Y. Zhao, Z. Wang, Y. Meng, D. Pei, Y. Peng, W. Tang, and X. Qu, "Why it takes so long to connect to a WiFi access point," in *IEEE INFOCOM 2017*, pp. 1–9, May 2017.

[6] B. Van den Bergh, T. Vermeulen, and S. Pollin, "Analysis of Harmful Interference to and from Aerial IEEE 802.11 Systems.," in *Proc. of DroNet*, DroNet '15, ACM, 2015.

[7] S. Hayat, E. Yanmaz, and C. Bettstetter, "Experimental analysis of multipoint-to-point UAV communications with IEEE 802.11n and 802.11ac," in *IEEE 26th PIMRC*, pp. 1991–1996, Aug 2015.

[8] M. Asadpour, D. Giustiniano, and K. A. Hummel, "From Ground to Aerial Communication: Dissecting WLAN 802.11N for the Drones," in *Proc. of the 8th WiNTECH workshops*, ACM, 2013.

[9] A. Finn and S. Franklin, "Acoustic sense and avoid for UAV's," in *2011 Seventh ICISSNIP*, pp. 586–589, Dec 2011.

[10] E. Vinogradov, D. Kovalev, and S. Pollin, "Simulation and detection performance evaluation of a UAV-mounted passive radar," in *2018 IEEE PIMRC*, September 2018.

[11] M. U. de Haag, C. G. Bartone, and M. S. Braasch, "Flight-test evaluation of small form-factor LiDAR and radar sensors for sUAS detect-and-avoid applications," in *2016 IEEE/AIAA 35th DASC*, pp. 1–11, Sept 2016.

[12] C. D. Wagter, S. Tijmons, B. D. W. Remes, and G. C. H. E. de Croon, "Autonomous flight of a 20-gram Flapping Wing MAV with a 4-gram onboard stereo vision system," in *2014 IEEE ICRA*, pp. 4982–4987, May 2014.

[13] M. M. Azari, H. Sallouha, A. Chiumento, S. Rajendran, E. Vinogradov, and S. Pollin, "Key technologies and system trade-offs for detection and localization of amateur drones," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 51–57, 2018.

[14] "Espressif website: Esp8266." https://www.espressif.com/en/products/hardware/esp8266ex/overview. Accessed: 26-10-2018.

[15] "ESP8266 Arduino Core." https://arduino-esp8266.readthedocs.io/en/latest/index.html, 2018. [Online; accessed 20-August-2018].