# Learning from Errors: Detecting Cross-Technology Interference in WiFi Networks

Daniele Croce, Domenico Garlisi, Fabrizio Giuliano, Ilenia Tinnirello
Department of Electrical Engineering, Università di Palermo, Italy
Email: *name.surname*@unipa.it

***Abstract*—In this work we show how to detect and identify cross-technology interference on commodity WiFi cards by monitoring the reception errors, such as synchronization errors, invalid header formats, too long frames, etc. Indeed, in presence of non-WiFi modulated signals, the occurrence of these types of errors follows statistics that can be easily recognized. Moreover, the duration of the error bursts depends on the transmission interval of the interference source, while the error spacing depends on the receiver implementation.**

**On the basis of these considerations, we propose the adoption of hidden Markov chains for characterizing the behavior of WiFi receivers in presence of controlled interference sources (*training phase*) and then run-time recognizing the most likely cause of error sequences. Experimental results prove the effectiveness of our approach for detecting ZigBee, Microwave and LTE (in unlicensed spectrum) interference.**

***Index Terms*—Wireless LAN, Interference, Hidden Markov models.**

## I. Introduction

Nowadays, we are witnessing an impressive success of IEEE 802.11 technology, better known as WiFi, for supporting the growing demand of wireless broadband connectivity. Public WiFi networks are deployed worldwide, with more than 50% of the total mobile traffic carried by WiFi. The availability of WiFi networks is often considered as a commodity service driving immense economic value, and the unlicensed spectrum is becoming one of society's most valuable resources. Although WiFi is a dominant communication technology in this spectrum, many other low range technologies coexist in unlicensed ISM bands for supporting several vertical applications, such as house and building automation, smart metering systems, surveillance systems, health care monitoring, game remote controllers and so on. Moreover, cellular technologies are trying to extend their operation to ISM bands for increasing their capacity. Two different solutions have been envisioned by 3GPP in ISM bands, referred to as Licensed Assisted Access (LAA) [1] and LTE-Unlicensed (LTE-U) [2], which work respectively, with and without the listen-before-talk mechanism.

Despite the fact that many mechanisms have been included in the WiFi standard to cope with interference (e.g. carrier sense, adaptive modulation and coding), it has been shown that serious performance impairments can arise in presence of exogenous interfering signals due to different technologies. For example, in [3] it is shown that the capacity of a good WiFi link can be reduced to zero in presence of analog phones, video cameras, or sensors based on IEEE 802.15.4 technology, while other devices such as a Xbox controller and a microwave oven can half the throughput. The effect of sensors' interference on the WiFi link is impressive if we consider that the 802.11 and 802.15.4 technologies are pretty heterogeneous in terms of bandwidth (2 MHz for 802.15.4, and 20 MHz for 802.11, hereafter referred as ZigBee and WiFi) and transmission power (e.g. 0 dBm for ZigBee and 20 dBm for WiFi). Moreover, ZigBee applications are typically low rate, while WiFi networks exhibit abundant channel idle space in time domain [4]. About the interference with cellular technologies, several research studies are trying to characterize the impact of LTE transmissions on WiFi performance. In [5] it is demonstrated that even when utilizing the listen-before-talk principle, LAA-LTE heavily impacts WiFi performance, and that WiFi with MIMO performs worse than WiFi without MIMO when LTE interference is strong. Additionally, increasing distance between LTE and WiFi links does not necessarily decrease the impact of interference in indoor environments.

In this scenario, we argue that a critical aspect for WiFi networks is enabling the correct identification of coexistence problems with other technologies, which in turn can serve as basis for some cross-technology coordination mechanisms. While state-of-the-art solutions for detecting coexistence problems in WiFi networks have mainly worked on the characterization of RSSI samples observed at different frequencies and with varying temporal gaps, in this work we propose to simply monitor the reception errors of commodity WiFi cards, and then apply hidden Markov chains in order to identify and characterize cross-technology interference. Our mechanism is based on the analysis of the *error domain*, i.e. on the classification of error events and on the time intervals between their occurrence. Statistics of these errors are widely available on many WiFi *commodity* cards and can be easily exploited to improve interference detection and troubleshooting algorithms of wireless networks. Although in this work we focus on three interference sources, namely ZigBee, LTE and microwave ovens, our solution does not depend on the type of technology, but only requires a training phase based on the events generated in presence of a controlled source of interference. We then employ a hidden Markov chain to identify the interference technology from the occurrence of the generated error events. The idea is that the proposed approach

This work has been partially supported by EU funded research projects symbIoTe, H2020-ICT-2015 grant agreement 688156, and Flex5Gware, H2020-ICT-2014-2 grant agreement 671563.

could be easily extended to any other type of interference.

After a brief review of some literature solutions (section II), we provide necessary background information on the competing technologies (section III) and we analyze the theoretical and experimental error rates caused by this interference (sections IV-B and IV-C). The interference detection model is introduced in section V, where we also present our implementation choices. Experimental results show that the approach is promising and suitable for further extensions as described in the concluding remarks.

## II. Related Work

*Effects of cross-technology interference.* Performance degradation of WiFi networks in presence of cross-technology interference has been widely studied in recent literature. Indeed, since each technology implements different mechanisms and protocols for reacting to interference, it is not obvious to predict WiFi performance in case of coexistence with other technologies. Several analytic and simulation models, as well as experimental studies, have been proposed for characterizing the cross-technology interference in ZigBee and WiFi networks [6], [7]. While early studies mostly focus on the analysis of ZigBee performance degradation in presence of WiFi interference, it has been shown that significant throughput reductions can also be observed in WiFi networks [6], [8]. Surprisingly, WiFi vulnerabilities arise despite the fact that many mechanisms have been included at the MAC and PHY layer for guaranteeing robustness to interference. This phenomenon has been justified by considering the higher time resolution needed by ZigBee for detecting channel activity and preventing collisions [9], [10].

LTE transmissions in unlicensed bands can have a deep impact on WiFi performance, even when the listen-before-talk mechanism is adopted [5]. Although most of the current studies are based on simulations (see for example [11]), preliminary empirical results show that WiFi performance can be critically affected even when LTE links operate at the minimum bandwidth of 1.4 MHz. This is due to the fact that WiFi nodes are generally able to sense LTE nodes operating in ISM bands and therefore are prevented from accessing the medium in case of LTE transmissions. Solutions based on duty-cycle muting or blank subframes [12] can be effective for increasing WiFi throughput, byt they are unilaterally controlled by LTE nodes. Advanced PHY solutions can also be envisioned for improving coexistence. For example, in [13] a mechanism to decode WiFi MIMO transmissions under strong LTE interference is proposed using a GNU Radio testbed with USRP devices.

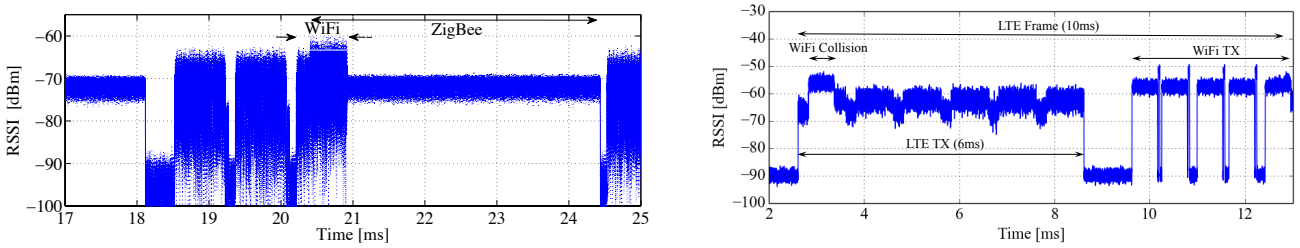
*Coordination strategies.* A simple solution for improving coexistence is introducing some forms of coordination mechanisms among technologies. Early solutions which detect interference and simply choose a better channel to transmit are becoming not viable because of the increasing number of technologies and applications in the market. Other solutions rely on complex and expensive radio transceivers to communicate with multiple protocols and different technologies [14], or increase the robustness of the transmission with use of error correction codes or multiple antennas [15]. Different approaches have considered the possibility to introduce some indirect forms of coordination between WiFi and ZigBee, based on opportunistic exploitation of WiFi temporal spaces [16], channel reservations [9] based on an additional ZigBee channel for making the channel busy for WiFi stations, or by means of simple forms of adaptive redundancy [10]. Regarding LTE, it has been proposed to improve coexistence with WiFi by introducing a centralized controller and tune LTE parameters based on WiFi traffic conditions [17], [18]. However, this requires a global authority which is difficult to implement in practice.

*Detection of cross-technology interference.* Obviously, an important component of any coordination strategy is detecting the coexistence problem, i.e. identifying the presence of two coexisting technologies. The monitoring of heterogeneous RF signals on ISM bands has been specifically addressed in [19], where it is proposed a design of a monitoring module for GNU radio implementing multiple receivers able to quickly identify the transmitting technology by simultaneously demodulating the received signal according to different PHY specifications. Other approaches which do not implement a complete per-technology demodulator are based on cyclostationary signal analysis and blind signal detection [20] or other spectrum sensing techniques [21]. Although these approach are very effective, they require specialized hardware (basically, a spectrum and signal analyzer). The possibility to detect ZigBee and other interference sources by means of WiFi commodity cards is explored in [3] by using an 802.11n PHY able to read RSSI values at different sub-carriers. Complex feature extraction algorithms are applied to the RSSI samples for characterizing spectral, energy and pulse signals that are mapped into a technology classification scheme. The approach is very promising, although the extraction of some features requires to monitor the interfering signals for some seconds. Similarly to this solution, our solution is purely software and works on measurements provided by commercial WiFi cards. However, we propose a completely different (and complementary) approach: rather than characterizing the frequency and time signatures of *external* interfering signals, we model the *internal* behavior of the WiFi receiver under different interfering sources on the basis of the error traces experienced in presence of interference. The receiver model is then exploited for classifying *each* interfering transmission. A preliminary version of this work was first presented in [22], although with a simplified model focused only on the identification of interference bursts caused by ZigBee or microwave ovens. In this paper, we extend the model to include inter-arrival timings and for the detection of LTE-U interference.

## III. Background

In this section we briefly recall some key aspects of the MAC/PHY layers in WiFi, ZigBee and LTE that affect the power of cross-technology interference and the typical timings of transmissions and channel idle intervals.

*Interference power.* Although all the technologies considered in this study work in the ISM bands, they differ in terms




(a) (b)

Fig. 1. Interference between technologies: temporal trace (RSSI samples) of WiFi-ZigBee (a) and WiFi-LTE collisions (b).

of available channels, operating bandwidth and transmission power. While WiFi can work on both the 2.4 GHz and 5 GHz bands of unlicensed spectrum, ZigBee works on the 2.4 GHz spectrum only and LTE will likely operate on the less crowded 5 GHz spectrum. The interference power experienced by a generic WiFi node depends on the transmission power of the interference source, but also on the overlapping between the bandwidth used by the interferer and the WiFi node.

WiFi and LTE transmissions are typically performed at a maximum power of 15 or $20dBm$, while ZigBee transmissions can span in the range $[-25, 0]dBm$. LTE transmission power is modulated because of power control mechanisms, which are usually not implemented in WiFi and ZigBee. The interfering power is a portion of the total transmission power roughly given by the portion of transmission bandwidth which overlaps with the WiFi receiver bandwidth. For identifying such a portion, we remind that each WiFi channel is 20 MHz wide and is spaced of 5 MHz from the adjacent ones. ZigBee channels have only 2 MHz of bandwidth with 3 MHz of inter-channel gap bands (i.e. the center frequencies maintain the spacing of 5 MHz from the adjacent channels). It follows that four ZigBee channels are entirely included in a WiFi channel. LTE center frequencies in ISM bands coincide with WiFi ones, with a typical bandwidth of 5 MHz (but bandwidths as smaller as 1.4 MHz are possible).

*Transmission times.* Since the three technologies have been defined for different applications, the frame size, the data rates and the channel access units considered by the standards are quite different.

For WiFi and ZigBee, channel access is performed on a per-packet basis, i.e. transmission times correspond to the time required for completing the transmission of a packet (or an aggregation/fragmentation of packets). ZigBee packets are small, with a maximum payload of only 127 bytes. Bytes are organized into 4-bit symbols that are mapped into 16 pseudo-random sequences of 32-chip transmitted at 2 Mchip/s (i.e. 250 Kbps), which correspond to a frame transmission interval of about 4 ms for the maximum frame size. WiFi frames are much longer, with a maximum frame size of 2358 bytes and multiple OFDM modulations and coding schemes available (from 6 Mbps up to 54 Mbps, which lead respectively to a maximum transmission time of about 3.2 $ms$ and 0.37 $ms$). For LTE, the channel access is performed on the basis of resource block allocations, which are organized into sub-intervals lasting a fixed time of 1 $ms$ within a frame of 10 $ms$. Packet transmissions are achieved by scheduling a given set of resource blocks in one or multiple consecutive frames. Although the total number of resource blocks used for each packet depends on the employed data rate and multiple rates are available (up to 25.2 Mbps for 5 MHz of bandwidth with 300 sub-carriers, 64-QAM modulation, and a symbol time of 71.4 $\mu s$), the channel occupancy time in each channel access is fixed according to the LTE frame structure.

*Intervals between transmissions.* Different channel access schemes are employed in WiFi, ZigBee and LTE for unlicensed bands. WiFi and ZigBee are mostly based on random access: each node senses the channel before transmitting and randomly defers its transmission in case the channel is sensed as busy. Although the rules for managing the deferral time and sensing the medium are technology-specific, for both the technologies the random access scheme implies a random variability of the time between consecutive transmissions. The deferral time unit, called backoff slot, is set to 320 $\mu s$ in ZigBee and 9 $\mu s$ in WiFi. The difference is due to the technology-specific granularity at which the channel sensing is performed. During a backoff slot, ZigBee spends 128 $\mu s$ for detecting the channel activity and 192 $\mu s$ to switch from reception to transmission mode. If a WiFi transmission is originated during this switching time, it cannot be detected by the ZigBee node. Figure 1-a shows a channel occupancy trace acquired by means of a USRP node in a network in which a WiFi node coexist with a ZigBee one. In the figure we clearly observe that each transmitter is characterized by a specific RSSI value and frame transmission time: WiFi frames occupy the channel for less than 1 $ms$ with a RSSI value of -65 dBm, while ZigBee frames last 4 $ms$ with a RSSI value of -72 dBm. The figure also shows that a ZigBee transmission can overlap with WiFi, in case a WiFi frame is transmitted during the time spend by ZigBee for switching from sensing to transmission mode.

LTE transmissions in licensed bands are organized into frames of 10 $ms$ that start at regular time intervals. For operating in unlicensed bands, two different adaptations have been envisioned: employing duty cycles for periodically suspending frame transmissions, while keeping the synchronization of time instants at which frame transmissions can start (LTE-U); employing listen-before-talk before transmitting each frame (LTE-LAA). In this second case, when the medium is sensed as busy, the deferral time is given by a fixed time of 10 $ms$ for maintaining the synchronization of frame starting

times (with the so called FBE mechanism) or it is given by a random slotted deferral time compensated by a varying channel occupancy time (with the so called LBE mechanism). In our work, we emulate both the LTE-U and LTE-LAA approach, by assuming that LTE frame transmissions can start only at regular time intervals. Figure 1-b gives an example of the interaction between an LTE-U transmission with 6 active and 4 silent subframes (i.e. 6 $ms$ on and 4 $ms$ off) and a WiFi station which tries to access the same channel: the figure shows that WiFi packets can collide with LTE and that part of the channel time is wasted due to the consequent backoff.

## IV. Error Analysis in WiFi Receivers

Our work is motivated by the observation that, in WiFi receivers, the errors generated by exogenous RF signals (i.e. non-WiFi modulated signals) exhibit significant differences, in terms of occurrence probability and time intervals between consecutive errors, from the ones generated by collisions with other WiFi transmissions. Obviously, the receiver errors are triggered only when the external RF signal is able to activate the receiver. The receiver activation depends on the interfering power and on the receiver sensitivity and settings (e.g. the AGC gain). In some cases, even the background noise can stimulate the receiver to perform a synchronization trial.

### A. Error Occurrence Probability

In case of wide-band noise and exogenous interference signals, WiFi receivers demodulate a sequence of completely random bits and tries to interpret these bits according to the format of WiFi frames. Being all the bits random, the probability of having a specific error heavily depends on the format of the expected frame.

Figure 2 summarizes the error probability observed when an 802.11g receiver is triggered by non-WiFi modulated signals. Since the PLCP header has one bit only for parity checks, on average one half of the frames should be classified as frames with *Bad PLCP*. However, the receiver can rely also on the RATE field of the header for detecting *Bad PLCP* errors: since the RATE field is 4 bits long while only 8 modulation rates are admitted (out of the 16 possible values), one half of frames which randomly results in a correct parity check will contain a wrong RATE value, thus increasing the *Bad PCLP* error probability to 3/4.

When a *Bad PLCP* is not detected (25% of the times), the receiver will leave the transceiver on and will continue demodulating until another error is reached, i.e. *Too Long*, *Too Short* or *Bad FCS*. In particular, the LENGTH field in the PLCP header is 12 bits long (values between 0 and 4095), while the length of a WiFi frame is generally between 14 and 2346 Bytes. Therefore, the frame will be considered *Too Long* with probability $1 - 2346/4096 \approx 0.43$ and *Too Short* with probability $14/4096$. The FCS is 32 bits long which means that the probability of having a random sequence with good FCS is only $2^{-32}$ and therefore it is almost certain that a *Bad FCS* error will appear when the frame is not *Too Short* or *Too Long* ($\approx 0.57$). Finally, an *Invalid MAC Header* error occurs when the 2 bits of the VERSION field in the MAC header are

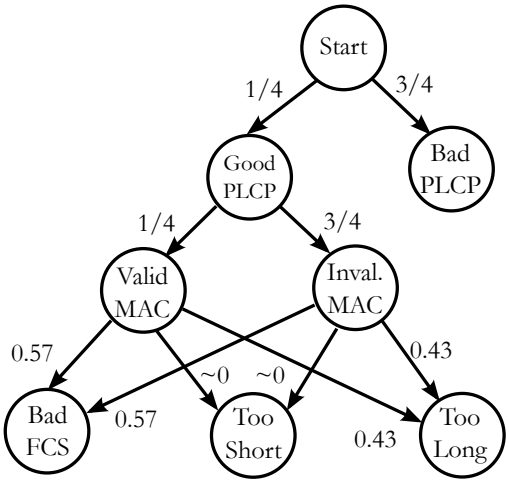



Fig. 2. Receiver events and relevant probabilities during cross-technology interference.

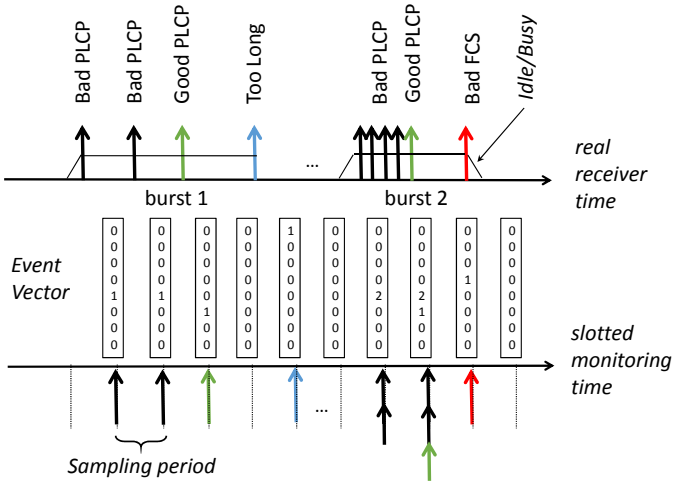



Fig. 3. Mapping between a real trace of receiver events and the time-slotted vectors generated by the monitoring process.

not 0, which correspond to 3 configurations out of 4 possible ones, i.e. to an occurrence probability of 3/4. In this case the transceiver does not suspend the reception but continues until another error is encountered.

When the errors detected by a WiFi station closely follow these statistics, it is very likely that interference is generated by non-WiFi modulated signals. For WiFi modulated signals error statistics are very different from the previous ones and vary during the frame reception as a function of the field length and rate. For example, PLCP errors have much lower probability to appear compared to bad FCS, because the PLCP transmission is usually more robust and shorter than the rest of the frame.

### B. Monitoring Receiver Errors

Most commercial WiFi cards track the occurrence of different *receiver events*, such as the start of a synchronization trial, the detection of wrong PLCP, the end of a frame transmission, etc., by means of specific counters implemented in internal registers. As a reference WiFi receiver, we considered a WiFi card (namely, Broadcom bcm4318) for which the card internal

| | WiFi ch11 | | WiFi ch10 | | WiFi ch8 | | ZigBee HighPW | | ZigBee LowPW | | LTE-U | | Microwave | | **Model** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | Ev./s | (%) | Ev./s | (%) | Ev./s | (%) | Ev./s | (%) | Ev./s | (%) | Ev./s | (%) | Ev./s | (%) | (%) |
| Bad PLCP | 6.5 | (0.5) | 455.8 | (54.8) | 1694.2 | (75.7) | 266.9 | (69.1) | 984.4 | (72.9) | 372.2 | (80.0) | 116.1 | (73.6) | (75.0) |
| Good PLCP | 1110.0 | (99.4) | 375.8 | (45.2) | 542.9 | (24.3) | 119.6 | (30.9) | 366.0 | (27.1) | 121.9 | (20.0) | 41.7 | (26.4) | (25.0) |
| Invalid MAC Header | 4.0 | (0.4) | 286.8 | (76.3) | 359.1 | (66.1) | 84.9 | (71.0) | 243.1 | (66.4) | 91.9 | (75.1) | 31.27 | (74.9) | (75.0) |
| Good FCS | 1067.1 | (96.1) | 0 | (0.0) | 0 | (0.0) | 0 | (0.0) | 0.0 | (0.0) | 0.0 | (0.0) | 0.0 | (0.0) | (0.0) |
| Bad FCS | 9.0 | (0.8) | 368.3 | (98.0) | 285.8 | (52.6) | 69.3 | (58.2) | 147.6 | (40.3) | 55.7 | (45.7) | 23.1 | (55.4) | (56.9) |
| Too Short | 0.1 | (0.0) | 0 | (0.0) | 1.7 | (0.3) | 0.6 | (0.5) | 0.2 | (0.0) | 0.4 | (0.0) | 0 | (0.0) | (0.4) |
| Too Long | 0.2 | (0.0) | 0.3 | (0.1) | 251.8 | (46.4) | 49.4 | (41.3) | 218.3 | (59.5) | 66.6 | (54.3) | 18.5 | (44.6) | (42.7) |

TABLE I

EVENTS CAUSED ON WIFI CHANNEL 11 BY WIFI ON INTERFERING CHANNELS OR DURING ZIGBEE, LTE OR MICROWAVE INTERFERENCE (AND NO WIFI TRANSMISSION).

| Receiver Event | Description |
|---|---|
| Too Long | Frame longer than 2346 bytes |
| Too Short | Frame shorter than 16 bytes |
| Invalid MAC Header | Protocol Version is not 0 |
| Bad FCS | Checksum Failure on frame payload |
| Bad PLCP | Parity Check Failure on PLCP Header |
| Good PLCP | PLCP headers and Parity Check OK |
| Good FCS and RA match | Correct FCS matching the Receiver Address |
| Good FCS and not RA match | Correct FCS not matching the Receiver Address |

TABLE II

RECEIVER EVENTS REPORTED BY BCM4318 CARDS.

registers are documented and an interface for reading the register values is available [23]. Table II summarizes the receiver events tracked by this card, from which it is possible to derive the *receiver errors* discussed in the previous section. For producing a temporal trace of the receiver events, storing the ordered sequence of event type and occurrence time, we implemented a monitoring process devised to sample at regular intervals the receiver registers. Indeed, the event occurrence cannot be detected by the card host as an interrupt signal, but needs to be indirectly identified by comparing the state of the receiver registers in consecutive sampling times.

We set a sampling interval equal to $250\mu s$ as a trade-off between detection delay and tracking complexity, while avoiding the overloading of the card to host interface. Because of the periodic sampling, multiple receiver events can occur in the same monitoring interval. Event samples are represented by a vector of eight components, whose value represents the counter of each different event type. We also sampled another card register, called busy time register, which does not track the occurrence of receiver events but rather the cumulative time during which the receiver remains active. The differences among consecutive values of the busy time register can be mapped into a logical idle/busy state of the channel as observed by the receiver.

Figure 3 shows the operation of our monitoring process: a real trace of receiver errors is mapped into a time series of event vectors, in which we can easily recognize consecutive *error bursts* due to the same interfering transmission. Error bursts can be originated for many different reasons: for example, a checksum failure can follow the detection of a good PLCP, or multiple (failed or not) synchronization trials are performed after a bad PLCP event. The total number of receiver events in a burst depends on the duration of the interfering transmission and on the receiver implementation, i.e. on the reset time required by the demodulator for performing consecutive synchronization trials. Each burst can be delimited by observing the time interval elapsed from the previous and next events, and/or by considering the channel transitions from idle to busy and from busy to idle as delimitation times.

### C. Experimental Results

In order to experimentally validate our findings on the error occurrence in presence of interfering signals, we run some experiments in our lab at the University of Palermo in different hours of the day (i.e. under uncontrollable interference from other WiFi networks), by placing a monitoring WiFi card set on channel 11 in the same room with heterogeneous interfering sources. Four different interfering sources have been considered: a ZigBee transmitter, a LTE transmitter, a WiFi transmitter and a microwave oven. All transmitting nodes have been configured for working on different interfering and non-interfering channels, while their reciprocal distance has been set to a few meters.

Two types of ZigBee nodes where used in our testbed. Commercial Zolertia Z1 motes, based on Texas Instruments CC2420 transceiver, and two custom-made nodes based on Microchip MRF24J40 transceiver. Both transceivers are 802.15.4 compatible and, in the experiments, they both generated the same sequence of errors. For ease of presentation, the results shown in the paper are based on the MRF24J40 transceiver only. The ZigBee frames are transmitted at 250kbps with a length of 127 bytes. WiFi transmitter has been implemented by using the same Broadcom card used by the WiFi monitoring node, with a frame length of 1500 bytes transmitted at 24 or 36 Mbps. The LTE-U transmitter, instead, was implemented on a SDR platform based on USRP B-210 and the srsLTE framework [24]. We considered a downlink interfering stream with 5 MHz of bandwidth and 300 sub-carriers, centered on channel 11. Following the standard, the whole frame allocation time is $10ms$ composed of 10 sub-frames. The frame structure has been organized introducing silent intervals and a fixed sub-frame pattern, with mask [1,1,1,1,1,1,0,0,0,0] where 1 indicate transmission allowed and 0 transmission denied.

*1) Statistical Analysis:* We run different experiments by activating a single interference source in each experiment: a WiFi interfering link at channel 11, 10 or 8; a ZigBee interfering link with different transmission powers (0 dBm and -23 dBm); an LTE interfering link with a transmission power of 15 dBm; a Microwave oven. In case of WiFi link on channel 11, all the frames are detected with good PLCP and almost all the frames have also a correct checksum. When the link is moved on the adjacent channel 10, the monitoring station

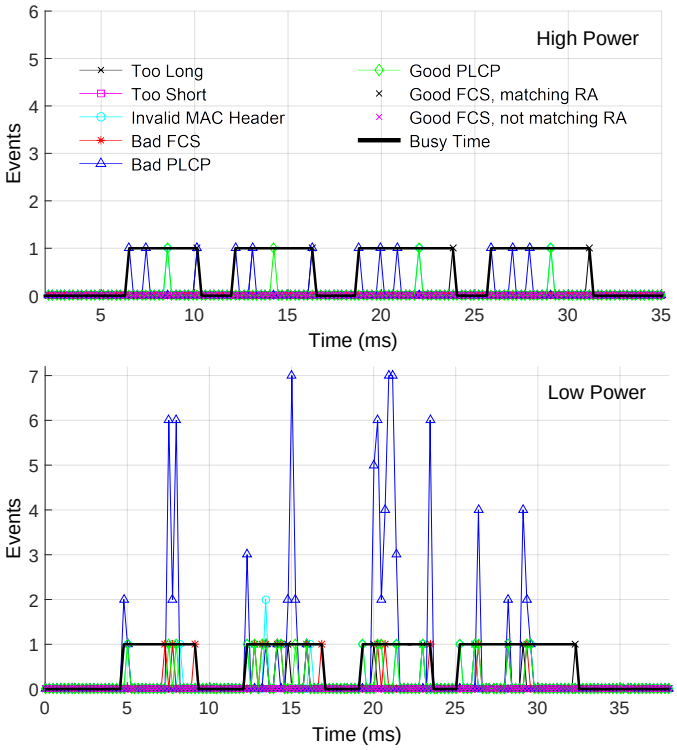



Fig. 4. Bursts of receiver events corresponding to the reception of ZigBee frames at high and low power.

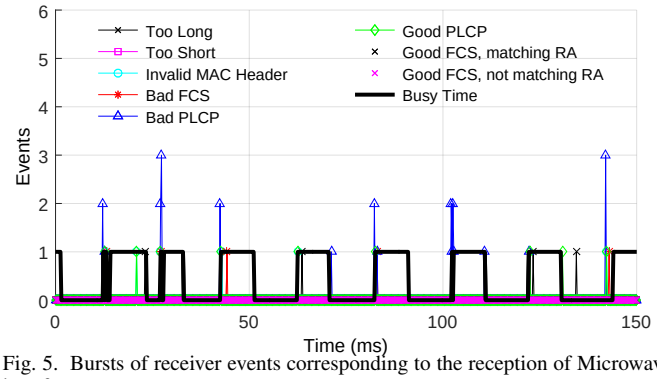



Fig. 5. Bursts of receiver events corresponding to the reception of Microwave interference.

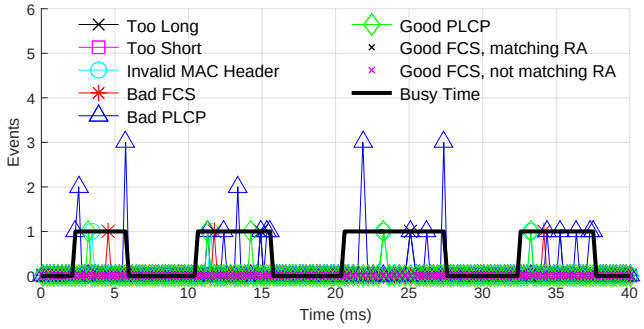



Fig. 6. Bursts of receiver events corresponding to the reception of LTE-U interference.

is able to correctly synchronize about one half of the frames (50% of the PLCP headers pass the parity check and have good rate values) which deterministically result in a failed FCS. Moving the link to channel 8, that is 15 MHz apart from the monitoring channel, significantly increases the detection of bad PLCP errors which reach over 1700 errors/s. This is due to the fact that when the receiver is not able to correctly synchronize the frame preamble, consecutive trials can be performed during the reception of the same frame and an higher number of error events can be generated for the same frame. Now, the error rates follow the statistics of non-WiFi modulated signals and *Too Long* errors appear.

Similar statistics are observed for ZigBee, LTE and Microwave interference. For the ZigBee case, the transmission power has an evident effect on the behavior of the WiFi receiver. The total number of events triggered by the receiver (the sum of Bad and Good PLCP events) increases from about 400 events/s to 1350 events/s with the same total number of interfering transmissions. This requires to further investigate on the receiver behavior, by observing the temporal trace of receiver events.

*2) Temporal Analysis:* In order to identify the bursts of errors generated by the same interfering transmission, we performed a temporal analysis of the error traces collected by our monitoring process. Figure 4 shows two exemplary temporal traces of receiver events in both the cases of high power and low power ZigBee transmissions with maximum payload size. When the interfering signal is high, the receiver employed in the Broadcom card is reset every $1ms$ for retrying to synchronize a preamble. At each reset, a good or bad PCLP event occurs with probability 1/4 and 3/4. This implies that during the reception of the ZigBee frame and corresponding acknowledgment (if any), the receiver generates a burst of events whose duration is about $4ms$ (for unacknowledged frames) or $4.5ms$ (for acknowledged frames). For example, in the top of figure 4, it is possible to easily recognize four consecutive ZigBee frames, with errors spaced about $1ms$ from each other. In case of low power transmissions (bottom of figure 4), the demodulator reset is no more regular and more receiver events are generated during each frame transmission, as already evident from the observation of the total number of receiver events in table I. The figure also shows the busy time intervals measured by the internal registers of the WiFi receiver. In principle, the transitions from idle to busy and from busy to idle should allow to easily identify each ZigBee interfering transmission. However, since the card implements both the actual and virtual carrier sense mechanism, in case of good PLCP events with valid headers, the card will assume that the channel will be busy for a time interval corresponding to: i) a frame length uniformly extracted in the range 14-4096 bytes, and ii) a transmission rate selected with equal probability (namely, 1/8) among the available ones. Specifically, the virtual duration is computed as the number of bytes indicated in the LENGTH field divided by the rate indicated in the RATE field. This explains why, when a good PLCP is raised during the reception of a ZigBee frame, the actual busy time (i.e. the maximum between the frame duration and the virtual busy time generated by the random bits) can exceed $4.5ms$.

Figure 5 shows a temporal trace of receiver events in case of interference due to a Microwave oven. The oven switches

periodically on and off as most Microwave ovens. During the radiation intervals, the WiFi monitoring node senses the channel as busy, as evident from the alternating busy and idle intervals plotted in the figure (whose length is 10 $ms$). Event sequences are pretty different from the ones observed in case of ZigBee transmissions: synchronization trials are performed only at the beginning and at the end of the radiation interval (rather than being continuously repeated). This can be due to the power-on and power-down ramp of the Microwave, being the demodulator unable to work when the radiation power is stable.

Finally, figure 6 shows the receiver events in presence of LTE transmissions. Under this interference source, the WiFi receiver behavior resembles the case of high-power ZigBee interference with the granularity of consecutive synchronization trials equal to regular intervals of 1 $ms$. However, occasionally, some events are closer to each other. We also observed, the occurrence of the first synchronization trial is not always synchronized with the the activation of the channel busy register: for example, in the figure at time 20 $ms$ the busy channel state switches to 1, while the first event vector with non-null components (namely, three Bad PLCP events) are revealed after 2 $ms$.

## V. Interference Detection

The experimental results presented in the previous section show that, although all non-WiFi interfering signals generate errors with similar statistics, their temporal analysis can be exploited for discriminating among different interfering sources. From the qualitative description of figures 4, 5 and 6 it clearly emerges that several features can be exploited for such a discrimination, such as:

1) *the number of simultaneous events* read by the monitoring process in the same sampling interval, which depends on the interfering power, with an higher number of synchronization trials performed in case of low power signals;
2) *the length of the error burst*, delimited by means of the correlation between the error vectors and the channel busy register, which depends on the transmission time of the interfering source;
3) the specific *sequence of error vectors*, which is affected by the variability of the interfering power during the same transmission (as in the case of Microwave ovens and LTE frames) and exhibits completely different occurrence probabilities in case of WiFi modulated signals;
4) *the time interval between consecutive error bursts* due to interfering transmissions, which depends on the typical activation timings of the interference source.

We therefore propose to classify different interference sources by analyzing the sequence of error vectors sampled at regular time intervals by the reference WiFi receiver. To this purpose, we modeled the receiver behavior under each interference source with a hidden Markov chain, whose state-dependent outputs are given by the sequence of *observable* error vectors. Indeed, the receiver behavior in consecutive sampling intervals exhibits some memory effects due to the power ramp of the interfering source at the beginning and at the end of the interfering transmission, and to the occurrence of a random good preamble, that can be modeled in terms of receiver internal states. Interference detection is then performed by selecting the model for which the posterior probability to obtain a given error sequence is maximized.

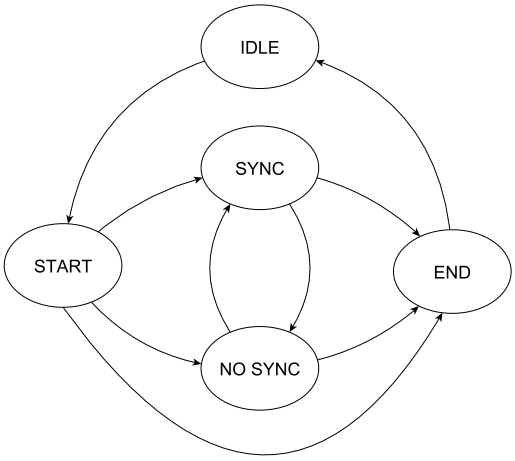



Fig. 7. Generalized state model of the receiver behavior: transition probabilities depend on the interference source.

### A. General receiver model

We propose to model the receiver behavior by means of a Hidden Markov Model (HMM), whose discrete evolution times correspond to the regular sampling intervals of our monitoring process. Although at a given time it is not possible to directly know which operations are performed by the receiver, such as a synchronization trial, the demodulation of a frame field, the gain adjustment, etc., the error vectors generated by our monitoring process can be considered as indirect observations of the receiver state. Being observations generated at discrete times, we assume that model evolutions are performed at the same time instants.

The adoption of a Markov chain is motivated by the need of modeling the memory effects described in the temporal analysis of the error vectors. Indeed, in case of high interfering power, due for example to a microwave oven, non-null error vectors are generated only at the beginning and at the end of the interfering signal. Moreover, some specific error events related to the non-valid frame formats (such as too long or too short frames) are triggered only after the detection of a valid preamble. Figure 7 shows our receiver model with five possible states: the IDLE state corresponds to the time during which the receiver is not active; the START and END states identifies the initial and final stage of the receiver activation; the SYNC state identifies the receiver operation after the synchronization of a valid preamble; the NO SYNC state characterizes the multiple synchronization trials performed when a valid preamble is not detected.

The probability of switching from one state to another depends on the errors detected by the receiver and on their typical timings. The probability to observe a given error

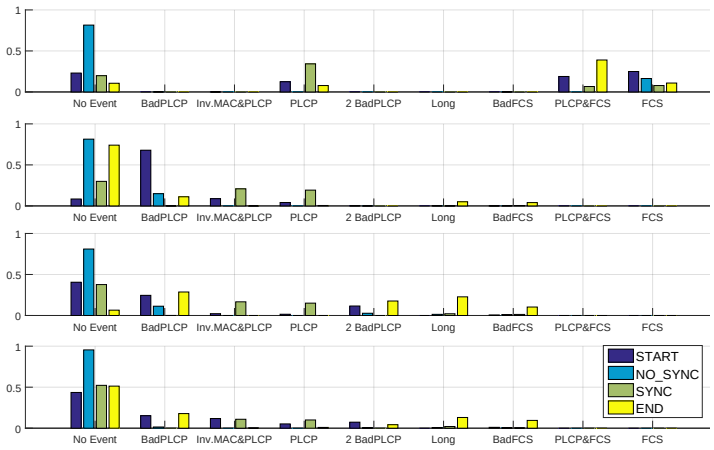



Fig. 8. Emission Probabilities of most significant observations for different experiments (from top to bottom: WiFi, ZigBee, LTE-U and Microwave).

vector, also called *emission probability*, mostly depends on the receiver internal state. The interfering source can affect the emission probability, because multiple synchronization trials in the same sampling interval can be performed in case of low power signals. Moreover, the statistics of interference inter-arrivals and durations are also incorporated into the receiver model, being these time intervals related to the typical timings of each technology. It follows that a different receiver model, specified in terms of transition and emission probabilities, can be used for characterizing the receiver behavior under specific interference conditions. Interfering signals which do not trigger the activation of the WiFi receiver are not detected by our scheme.

### *B. Model training under different interfering sources*

For describing the receiver model in presence of a given interference source, we need to specify the transition probability matrix governing the state evolution process, and emission probability matrix characterizing the probability to observe different error vectors from each state. To this purpose, we collected a trace of error vectors (i.e. observations) acquired in presence of a single interference source and tried to map the sequence of observations into a known state path. While the number of possible events summarized in table II is eight, the overall number of possible error vectors is higher because multiple events can be triggered during the sampling interval of the card registers. However, in most cases error vectors have a single non-null component and can be directly mapped into an event.

For deriving a known state path, we implemented the following approach. On the basis of the busy channel register, we organized the error vector trace into alternating idle and activity intervals of the receiver. During idle intervals, observations are given by the null vectors (i.e. no event is triggered) and the receiver remains in IDLE state. Conversely, activity intervals are generally characterized by a burst of non-null error vectors (although null vectors can appear within the burst). For example, in the top of figure 4 there are four activity intervals, with a last interval equal to the event sequence $\{$Bad PLCP, Bad PLCP, Bad PLCP, Good PLCP, Too Long$\}$. The state path corresponding to each activity interval can be easily derived by considering that the first and last observations are always performed from the START and END state, while all the others depend on the last preamble synchronization.

We collected three different event traces of $10s$ under WiFi traffic, ZigBee, LTE-U and Microwave interference. By using each trace and corresponding state path, we obtained the maximum likelihood estimates of the emission and transition probabilities from each state, devised to characterize the receiver behavior in presence of different signals. The derivation is based on the Baum-Welch algorithm. Figure 8 visualizes the emission probabilities of the most significant observations for different interference models. It is interesting to observe how the figure quantifies our previous qualitative considerations.

For the WiFi model, most observations result in a synchronized preamble followed by a correct checksum (that can be sampled into the same observation interval or into two consecutive observation intervals due to the short duration of WiFi frames). Packet duration is equal to about 350 $\mu s$, because we used frames with 1500 bytes transmitted at 36 Mbps. For the ZigBee model, bad preambles are generated very often: about 70% of error bursts start with such an event, while the other bad preambles are revealed during the intermediate model states. Checksum failures, too long frames or invalid MAC occur at the edge states or when the receiver is synchronized. For the Microwave oven, bad preambles are generated in the START and END states and the no event probability is higher than the previous ones (being the interference interval equal to 10 $ms$ and the demodulator active only during the power ramp). Finally, the LTE-U model falls somehow in between the ZigBee and the microwave model, with a slightly higher number of error events triggered.

Although the specific emission probabilities may depend on the receiver implementation, and in particular on the reaction times to synchronization errors and sensitivity to narrowband signals, the approach for training the hidden Markov chain is general and can be applied to different receiver implementations (provided that they can track the internal error events).

### *C. Classification schemes*

As a result of the training phase, we define four different HMM models characterizing the receiver behavior in presence of WiFi, ZigBee, Microwave and LTE-U interference. The number of hidden states in the general receiver model, as depicted in figure 7, is equal to 5. The number of possible error vectors is higher than the total number of possible events (which in our implementation is equal to 8), because multiple events can be triggered during the same sampling interval. However, being such a maximum number limited, the total number of possible configurations is limited too (in our experiments we found a maximum number of 40 different vectors). Let $n$ be the generic number of states and $m$ be the total number of error vectors with non-null occurrence probability. The receiver model in presence of the

$k$-th interference source is given by the transition probability matrix $P_k^{n\times n}$ and emission probability matrix $E_k^{n\times m}$ found by the training mechanism described in the previous section.

We propose two different approaches for classifying the interference sources acting on the target WiFi receiver. The first approach is based on the classification of the receiver behavior in a fixed time interval corresponding to $N$ samples of the error vectors. The classification interval is small enough (e.g. a few tens of $ms$) so that error bursts belonging to the same interval are likely generated by the same interference source. The second approach is based on the classification of the receiver behavior during a given error burst delimited by the channel busy register (i.e. a single frame or a single microwave radiation period). In this case, idle times between consecutive error bursts are not considered for the classification.

*Time-based classification.* For a given sequence of error vectors $\mathbf{e} = e_1, e_2, \cdots e_N$, our classification scheme works by selecting the interference model which maximizes the probability of obtaining the sequence $\mathbf{e}$, i.e. the interfering source is $k = argmax_k\ Pr\{\mathbf{e}|P_k, E_k\}$. Since the state path which generated the sequence is not known, the probability $Pr\{\mathbf{e}|P_k, E_k\}$ can be obtained by deriving the state probability at each sampling interval $i = 1, \cdots N$ of the sequence, and by weighting accordingly the emission probability of each observation $e_i$ from each state. A critical design parameter is choosing $N$: too short intervals could not include error bursts and inter-burst typical timings (on which classification is based), but as the observation interval increases, error bursts and idle times in the same sequence could be given by the overlapping of heterogeneous interference sources.

*Burst-based classification.* In this case, the sequence of error vectors $\mathbf{e}$ has a variable length $L$ and corresponds to a single error burst. The burst is delimited by using the busy channel register. Classification is still based on the selection of the interference model which maximizes the probability of obtaining the sequence $\mathbf{e}$, but the state path is partially known because it is delimited by the known START and END state, whose occurrence probability are equal to 1, respectively, at time 1 and time L of the burst. Moreover, transitions to IDLE, START and END states are not possible during the intermediate sampling intervals of the sequence, which corresponds to update the transition probabilities by considering these conditioning considerations. Since classification works on each error burst, consecutive bursts can be generated by heterogeneous sources and the scheme works as long as no collision occurs among interfering sources.

### D. Performance results

For assessing the performance of our classification schemes, we considered the case when a single interference source is active. Figures 9 and 10 visualize the classification results obtained by the time-based and burst-based classification in presence of ZigBee interference. The figure shows the logarithm of the occurrence probability of each sequence $\mathbf{e}$ computed according to the four interfering models: for the time-based approach the sequence lasts 50 $ms$ (i.e. 200 error vectors), while for the burst-based approach the average duration is 4.5 $ms$ (i.e. 18 error vectors). From the figure it is evident that the highest probability corresponds to the ZigBee interference source in almost all the cases. Moreover, in figure 10 the results provided by the WiFi model are very far from the other models, while in figure 9 they are not shown because outside the plotted range. Similar results were obtained also with the other interference sources.

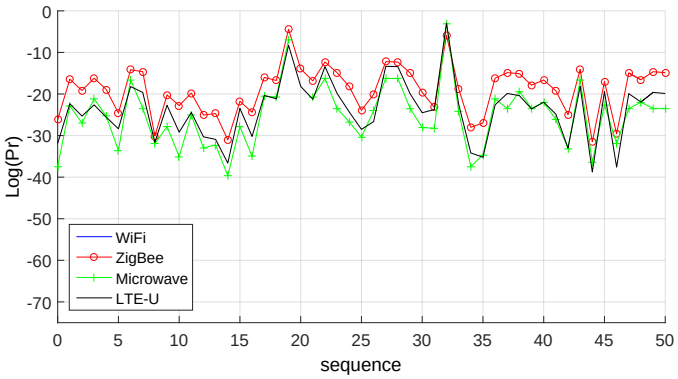



Fig. 9. Comparison between the time-based receiver models for a sequence of errors vectors due to ZigBee transmissions ($N = 200$).

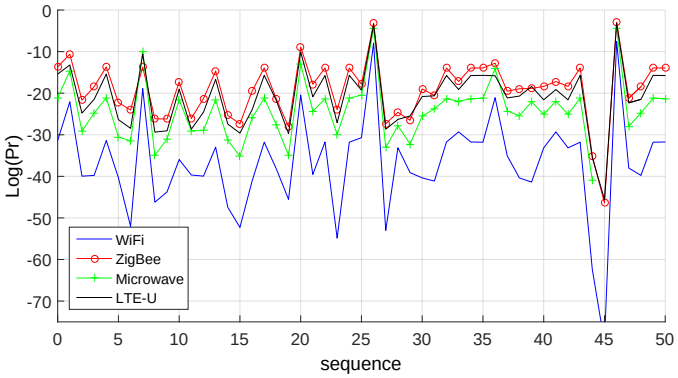



Fig. 10. Comparison between the burst-based receiver models for a sequence of error bursts due to ZigBee transmissions.

The classification accuracy can be defined as the ratio between the total number of correct decisions (in which the highest occurrence probability of a given sequence has been given by the correct model) and the total number of processed sequences. For the time-based approach, it is evident that the accuracy may depend on the length of the observation interval. Figure 11 shows the accuracy results obtained for different lengths of the observation interval (from 20 $ms$ to 200 $ms$) and for different interference sources. From the figure it is clear that in many cases the accuracy is above 95%, with the worst results given by the microwave oven, for which longer observation intervals can help. Table III shows the overall confusion matrix of our time-based classifier for $N = 200$ (i.e. 50 $ms$): indeed, microwave interference can be confused with LTE interference because of the periodic activity intervals.

Although the time-based classification works on longer sequences which include information on idle times between consecutive transmissions, the classification accuracy achieved by the burst-based scheme is comparable with the one obtained

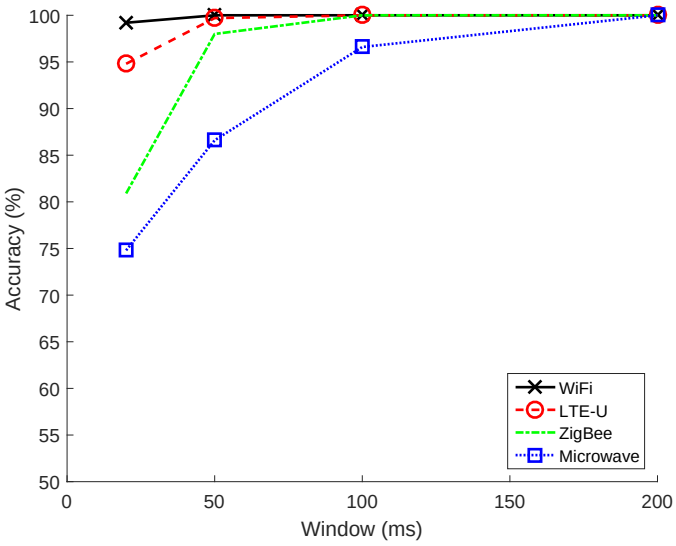



Fig. 11. Accuracy of the time-based receiver using different observation windows of 20, 50, 100 and 200 $ms$.

| | WiFi | ZigBee | Microwave | LTE-U |
|---|---|---|---|---|
| WiFi | **100.0** | 0.0 | 0.0 | 0.0 |
| ZigBee | 0.0 | **98.0** | 1.8 | 0.2 |
| Microwave | 0.0 | 1.7 | **86.6** | 11.7 |
| LTE-U | 0.0 | 0.0 | 0.3 | **99.7** |

TABLE III
CONFUSION MATRIX WITH TIME-BASED DECISIONS (N=200).

by the time-based approach. This consideration is quantified in table IV: the accuracy is on average close to 90% and never lower than 80%. It follows that, if the busy channel register is reliable for correctly identifying the bursts, this approach should be considered the most valuable approach for working in general scenarios where multiple interference sources are active. Classification of independent bursts (generates by different technologies) should work as in the case of single interference sources, apart from the case when the burst is generated by collisions between multiple interference sources. This type of combined interference, in principle, can be modeled for introducing more advanced interference detection schemes (able for example to recognize WiFi/ZigBee collisions). However, the identification of such events is of little interest and is out of the scope of this paper.

## VI. Conclusions and Future Work

In this work, we investigated on the possibility to detect ZigBee, LTE-U or microwave interference by using commodity WiFi cards. Differently from previous solutions, our approach is based on the analysis of the error signals generated by WiFi receivers when triggered by non-WiFi modulated signals. We prove that the statistics of these signals and the temporal sequences of the error events can be effectively correlated for detecting the presence of non-WiFi signals and identifying the interfering technology. In particular, the length of the error burst and the timing between consecutive bursts depend on the duration and access times typical of the interfering technology,

| | WiFi | ZigBee | Microwave | LTE-U |
|---|---|---|---|---|
| WiFi | **100.0** | 0.0 | 0.0 | 0.0 |
| ZigBee | 0.0 | **90.0** | 4.6 | 5.4 |
| Microwave | 0.2 | 1.7 | **89.6** | 8.5 |
| LTE-U | 0.0 | 4.9 | 13.0 | **82.2** |

TABLE IV
CONFUSION MATRIX WITH BURST-BASED DECISIONS.

while the number of error events generated in a sampling interval depends on the interfering power.

Starting from these observations, we propose to monitor the receiver events in consecutive sampling intervals for classifying the active interference sources. In particular, we defined a classifier based on a simple Hidden Markov Model, able to characterize the receiver behavior in presence of different interference sources. A methodology for training the model and segment the sequence of consecutive events in order to take run-time decisions has been designed and evaluated. Our experimental results show that the accuracy is on average over 90% even with the burst-based, per-packet analysis.

Although in this work we focused on interference detection from WiFi receivers, we expect that our methodology can be extended for working with other error types and receivers in ISM bands, such as commodity ZigBee receivers. We are also considering alternative approaches for classification, based for example on neural networks.

## References

[1] http://www.3gpp.org/ftp/Information/WORK_PLAN/Description_Releases/Rel-13_description_20150917.zip
[2] http://www.3gpp.org/ftp/Information/WORK_PLAN/Description_Releases/Rel-10_description_20140630.zip
[3] S. Rayanchu, A. Patro, and S. Banerjee. Airshark: detecting non-WiFi RF devices using commodity wifi hardware. In Proc. of IMC 2011.
[4] R. Chandra, R. Mahajan, V. Padmanabhan, and M. Zhang. Crawdad data set microsoft/osdi2006 (v. 2007-05-23), 2007.
[5] Yubing Jian, Chao-Fang Shih, Bhuvana Krishnaswamy, Raghupathy Sivakumar. Coexistence of Wi-Fi and LAA-LTE: Experimental Evaluation, Analysis and Insights. IEEE International Conference Communication Workshop (ICCW), 2015
[6] S. Pollin, I. Tan, B. Hodge, C. Chun, and A. Bahai. Harmful Coexistence Between 802.15.4 and 802.11: A Measurement-based Study. In Proc. of CrownCom, 2008.
[7] Y.S. Soo, S.P. Hong, H.K. Wook. Mutual interference analysis of IEEE 802.15.4 and IEEE 802.11b. In Comp. and Telecomm. Netw., 2007.
[8] R. Gummadi, D. Wetherall, B. Greenstein, S. Seshan. Understanding and Mitigating the Impact of RF Interference on 802.11 Networks. In Proc. of ACM SIGCOMM '07, Pages 385-396.
[9] X. Zhang, K. G. Shin. Enabling Coexistence of Heterogeneous Wireless Systems: Case for ZigBee and WiFi. In Proc. of ACM MobiHoc '11.
[10] C.-J. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis. Surviving Wi-Fi Interference in Low Power ZigBee Networks. In Proc. of SenSys 10, pages 309-322, 2010.
[11] A. M. Cavalcante et al., Performance Evaluation of LTE and Wi-Fi Coexistence in Unlicensed Bands, 2013 IEEE 77th Vehicular Technology Conference (VTC Spring), Dresden, 2013, pp. 1-6.
[12] E. Almeida et al., "Enabling LTE/WiFi coexistence by LTE blank subframe allocation," 2013 IEEE International Conference on Communications (ICC), Budapest, 2013, pp. 5083-5088.
[13] S. Yun and L. Qiu. Supporting WiFi and LTE co-existence. IEEE Conference on Computer Communications (INFOCOM), Kowloon, 2015.
[14] R. Gummadi, H. Balakrishnan, and S. Seshan. Metronome: Coordinating Spectrum Sharing in Heterogeneous Wireless Networks. 1st Int. Workshop on Communication Systems and Networks (COMSNETS), 2009.

[15] S. Gollakota, F. Adib, D. Katabi, and S. Seshan. Clearing the RF smog: making 802.11n robust to cross-technology interference. In Proc. of ACM SIGCOMM 11, pages 170-181, 2011
[16] J. Huang; G. Xing; G. Zhou; R. Zhou. Beyond Co-existence: Exploiting WiFi White Space for ZigBee Performance Assurance. ICNP, 2010.
[17] S. Sagari, S. Baysting, D. Saha, I. Seskar, W. Trappe and D. Raychaudhuri, "Coordinated dynamic spectrum management of LTE-U and Wi-Fi networks," 2015 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), Stockholm, 2015.
[18] Q. Chen, G. Yu and Z. Ding, "Optimizing Unlicensed Spectrum Sharing for LTE-U and WiFi Network Coexistence," in IEEE Journal on Selected Areas in Communications, vol. 34, no. 10, pp. 2562-2574, Oct. 2016.
[19] K. Lakshminarayanan, S. Sapra, S. Seshan, and P. Steenkiste. RFDump: An Architecture for Monitoring the Wireless Ether. In Procs. of CoNEXT 09, Dec. 2009.
[20] O. Zakaria. Blind signal detection and identification over the 2.4 GHz ISM band for cognitive radio. In MS Thesis USF09
[21] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty. Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey. Comput. Netw., 2006.
[22] D. Croce, D. Garlisi, F. Giuliano and I. Tinnirello, Learning from Errors: Detecting ZigBee Interference in WiFi networks, in Proc. 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET) 2014.
[23] http://bcm-v4.sipsolutions.net/802.11/Registers/
[24] https://github.com/srsLTE/srsLTE