CRABSS: CalRAdio-Based advanced Spectrum Scanner for cognitive networks

Riccardo Manfrin[*§], Luca Boscato[*], Andrea Zanella[*†§] and Michele Zorzi[*†§]
[*]Department of Information Engineering – University of Padova, Italy
[†]Consorzio Ferrara Ricerche (CFR) - Ferrara, Italy
[§]Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT) - Italy
E-mail: {rmanfrin, lboscato, zanella, zorzi}@dei.unipd.it

ABSTRACT
The first step required by the process of cognition is the intelligent observation of the environment that surrounds the actors of such a process. We present the CalRAdio-Based advanced Spectrum Scanner (CRABSS), an open platform developed to monitor the ISM 2.4-2.499 GHz band and reveal opportunities for a better utilization of the available spectrum resources. CRABSS is built through a modular approach by integrating the development platform CalRadio 1 with the Unified Link Layer API (ULLA) framework. This solution provides sensing capabilities while preserving the 802.11b standard compatibility on the CalRadio 1 platform. Moreover, it takes advantage of the ULLA framework to export spectrum occupancy information to prospective cognitive radio manager engines, through a standardized set of sensing APIs.

Categories and Subject Descriptors
K.6.1 [**Management of computing and information systems**]: Systems development; D.2.8 [**Software Engineering**]: Modules and interfaces

General Terms
Algorithms, Management, Measurement, Performance, Design, Experimentation

Keywords
CalRadio 1, MAC, 802.11b, ULLA, Spectum, scanner, channel, occupancy, energy, interference, detection, cognitive, allocation, Clear Channel Assessment, CCA, 802.11, 802.15, Zigbee, Bluetooth, ISM, cooperation, interference

1. INTRODUCTION
Economic and practical motivations are making wireless communication a winning technology in present and future network deployments. Depending on the specifications of the considered network scenarios, service types and size of the deployed network, different technologies (802.11a/b/g/n, 802.16/e, 802.15.4, Bluetooth, Hiperlan/2, etc.) can be exploited in order to best satisfy service requirements. Many of the aforementioned technologies share the same spectrum bands, hence potentially conflicting with each other. This phenomenon is a threat to wireless communications, especially in densely populated areas, where the ISM (Industrial, Scientific, Medical) bands are typically overcrowded by many private and public services. Only recently has the research community been considering the possibility of turning the issue of coexistence of heterogeneous access technologies competing within the same frequency bands into an advantage, by exploiting the principles of cooperation [1].

Yet, this challenge imposes clever management and optimization of the available spectrum resources, in order to allow their efficient use and the coexistence of multiple services within the same frequencies. In particular the first and foremost step to be able to coordinate heterogeneous technologies coexisting within the same frequency bands is to identify them. Most currently available commercial technologies support only basic interference detection and avoidance techniques, whereas active cooperation for the identification of and the coexistence with other radio systems is not available. In order to determine the level of utilization of different parts of the radio spectrum and to infer which technologies are active in the wireless neighborhood of a device, we developed CRABSS, i.e., a CalRAdio-Based advanced Spectrum Scanner platform, which integrates the power of the CalRadio 1 Software Defined Radio (SDR) [2] with the flexibility and modularity offered by the Unified Link Layer API [3] project.

CalRadio 1 is a development platform supporting an IEEE 802.11b RF transceiver and an open and completely reprogrammable MAC firmware. Taking advantage of the flexibility of the platform, we designed and developed a suitable MAC protocol that realizes the advanced spectrum scanning functionalities of CRABSS. More specifically, CRABSS provides two different operational modes, namely *horizontal* and *vertical*.

In the horizontal mode, CRABSS operates as a legacy IEEE 802.11b station, thus enabling the exchange of data with other stations, while collecting link-related performance measurements. With respect to most commercial cards, CRABSS offers a richer set of metrics that, besides the usual link-layer counters as the number of transmitted packets and ACK received, also includes system-related indices, such as number of active stations, channel utilization per station, station activity rate, and so on. Some of these metrics can also be obtained by using commercial WiFi cards in promiscuous mode that, however, does not allow for active data exchange. Conversely, CRABSS is capable of collecting these measure while maintaining its standard operational capabilities.

In the vertical mode, the software of the CalRadio 1 device is reconfigured in such a way that processing capabilities of the board are addressed to perform only Clear Channel Assessment (CCA), i.e., to reveal the presence of energy above threshold on a certain portion of the radio spectrum. With respect to other spectrum scanners, CRABSS offers much larger flexibility in setting the sniffing pattern, which can range from an extremely quick (tens of milliseconds) sweeping of the whole 2.4 GHz ISM band, spending just few milliseconds in each sub-band, to the persistent scanning of a single channel, or a group of channels. In short, CRABSS supports basically any frequency-scanning pattern, making it possible to specify the sequence of RF channels to be visited and the dwell time in each of them, within the limits of the CalRadio 1 hardware architecture.

The great flexibility of CRABSS can facilitate the identification of link or network critical situations, while offering a more complete picture of the current spectrum utilization that can be used to correctly drive cognitive optimization procedures. As an example, CRABSS can operate in the horizontal mode to exchange data with a legacy access point. During periods of inactivity, CRABSS can switch to the vertical mode to perform a quick scanning of the whole spectrum, thus maintaining an updated view of the current occupancy level of the different channels. If link measurements collected in the horizontal mode reveal that the link performance is going to degrade below a critical threshold, then CRABSS can make use of the information collected during its idle periods to switch to a better channel, thus saving time and limiting the performance loss.

The integration of CRABSS with the ULLA framework further enhances the potential of the solution by offering a standardized and rather intuitive interface to access and control CRABSS functionalities. In this way, CRABSS can be easily integrated in different cognitive architectures as either an advanced IEEE 802.11b station or a spectrum scanning device. The information collected by CRABSS and exported through ULLA can be stored in a database for different purposes, such as tracking the utilization level of the ISM spectrum over time, inferring the different technologies that are active in the frequency band, and so on.

In the remainder of this paper, we briefly describe the basic ingredients of CRABSS, namely CalRadio 1 and the ULLA framework. Then, we describe the CRABSS architecture, specifying some details of the supported functionalities, present some preliminary experimental results and discuss how CRABSS may be used to help alleviate the current problem of the overcrowded ISM bands. We conclude the paper with some final remarks.

2. CalRadio 1 ARCHITECTURE

CalRadio 1 is an open 802.11b-compatible development platform, designed and developed at the University of California, San Diego, with the aim of providing the research community with an open and fully reprogrammable board for experimental testing. The main purpose of this device is to study the 802.11 [4] MAC protocol to understand its critical points and possible enhancements.

In this section, we briefly recollect the most important features of the platform, referring to [5] for a more detailed analysis of CalRadio 1 functionalities and performance.

CalRadio 1 consists of four main components, namely an ARM (ARMv7TDMI) processor, a DSP (TI 5471 Digital Signal Processor) [6], a Baseband processor (Intersil HFA3863) [7] and an RF transceiver (MAX28281) [8]. With respect to the ISO/OSI reference protocol stack, the ARM processor supports the "higher layer" functionalities, from the network layer up, the DSP carries out the Data Link Layer (DLL) functionalities, including Medium Access Control (MAC), whereas the physical layer (PHY) functionalities are implemented by the Baseband processor and the RF transceiver, as schematically represented in Fig. 1.

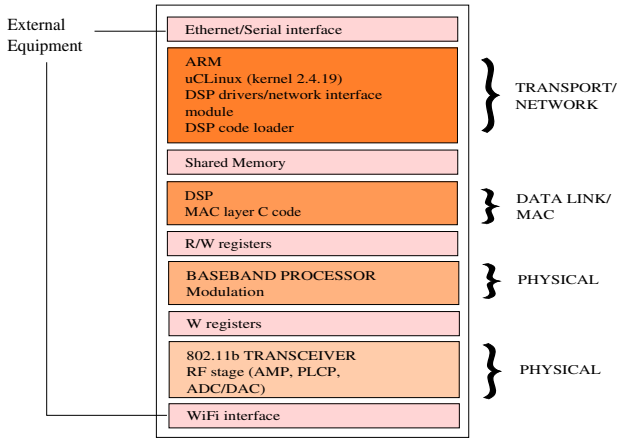

Figure 1: Network layers mapping into CalRadio 1 hardware architecture

The feature that distinguishes CalRadio 1 from common commercial WiFi cards is the ability to access and reprogram the DLL and MAC protocols, which are run on the DSP. Moreover, the DSP controls the Baseband processor and the RF transceiver setup, thus making it possible to directly play with most of the PHY settings also at runtime.

The core of the CalRadio 1 platform is a 100MHz clocked, programmable DSP. The DSP code can be written in standard ANSI-C language and compiled on an external PC. The generated binary code can be transferred to the ARM via standard File Transfer Protocol and then loaded on the DSP through a loader kernel module.

In Fig. 2 we sketch the general operational flow performed by the CalRadio 1 firmware [5].

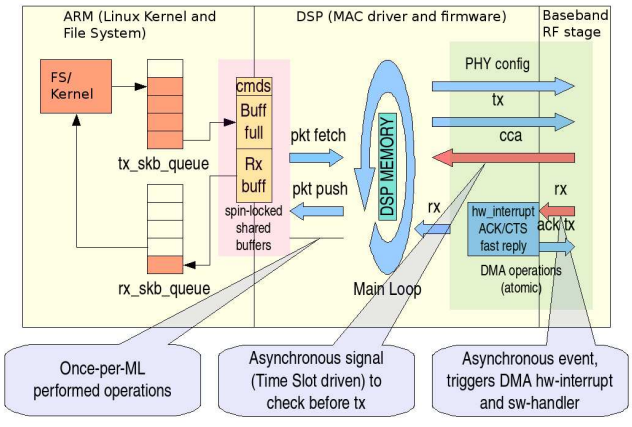

Figure 2: CalRadio 1 DSP code functional description

While cycling in the main loop (ML), the ARM processor delivers transmission packets to the DSP by copying them onto the shared

buffer. This segment of memory physically resides on the DSP. Once per ML the DSP checks for packets to transmit. If a new packet is found in the shared memory, the DSP transfers it to the Baseband processor for transmission. Packet reception at the physical layer is triggered by a correlation peak in the incoming signal that is generated by the Physical Layer Convergence Procedure (PLCP) synchronization sequence. The sensitivity of the reception circuit at the RF stage can be set using a dedicated register. This allows the reception procedure to be more or less selective.

3. ULLA FRAMEWORK

ULLA is an open framework developed within the European project "Generic Open Link-Layer API for Unified Media Access" (GOLLUM) [3]. The major focus of GOLLUM was to remedy the situation where a separate programming interface exists for almost every wireless technology. The ULLA framework solution proposed in GOLLUM implements an operating system-independent link-layer API to support heterogeneous systems, by unifying the various methods for accessing different wired and especially wireless links.

ULLA enables better portability of applications between devices using different communication interfaces. ULLA implements three distinct methods to access the resources of a network device. First of all, a query mechanism can be used for single request-response transactions between the ULLA framework and the targeted network device. In addition, ULLA supports methods to set up asynchronous notifications that can be triggered periodically or when an event verifies a predefined set of conditions. Finally, a command API allows direct access to the network device resources. All these APIs can be accessed by means of Universal Query Language (UQL) statements to express either requests, commands or conditions to be evaluated in order to perform a certain action (e.g., trigger a notification update).

As sketched in Fig. 3, the ULLA architecture has three major components, namely

- the ULLA framework core, which implements all the logic required to interface the upper network layers with the PHY and MAC layers of the physical devices;
- the Link Layer Adapter (LLA), which is responsible for translating UQL statements into device specific commands and device messages into ULLA compatible information;
- the Link Users (LU), namely any entity, layer or single application registering and accessing the ULLA core to gather PHY and MAC information from the ULLA registered network devices.

Each device that wants to interact with the ULLA core needs first to register with it and notify ULLA about its own capabilities. In this way, ULLA shall prevent Link Users from asking a device for information or commands that are not supported.

The ULLA Core optionally implements database information storage for statistical purposes, hence allowing a Cognitive Resource Manager (CRM) to perform not only short but also long term optimizations based on the data stored in the ULLA database.

Tab. 1 shows the classes of object abstractions defined by ULLA to

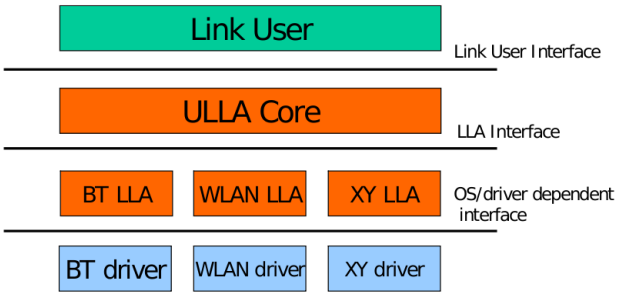

Figure 3: ULLA Architecture

Class/Object	Description
LinkProvider	Abstraction of the radio device attributes and operations. Radio devices are represented as a set of radio links that are supported by the available Radio Access Technologies. The scanAvailableLinks operation is used to populate the available links within the corresponding device profile.
Link	Represents a layer 2 link between the local radio device and another.
Channel	Represents a radio resource, which is used by one or more links.

Table 1: ULLA abstract object classes

operate commands and retrieve information. Each of these abstract objects can be accessed by any Link User.

4. CRABSS ARCHITECTURAL DESCRIPTION

The solution proposed in this paper is based on the integration of the CalRadio 1 platform with the ULLA framework. The first step to realize CRABSS was to enhance the CalRadio 1 MAC protocol with sensing capabilities while ensuring a minimal impact on the CalRadio 1 device efficiency, studied in [5]. To ease the setting of CRABSS parameters and the visualization of the collected measurements, we also developed a Link User application with a Graphical User Interface (GUI) that translates user's settings into ULLA commands and shows ULLA information in graphical form.

4.1 Providing CalRadio 1 with 802.11 MAC protocol and spectrum sensing

The CalRadio 1 MAC layer software purchased along with the board consists of a basic set of functionalities that support transmission and reception procedures. We designed and implemented the 802.11b MAC layer to be standard compliant and tested the communication of the developed DSP code with commercial boards (i.e., Atheros and Intel WiFi chipsets). With reference to Fig. 2, any time a packet is transferred to the DSP by the ARM, a new backoff procedure is started by the DSP. During this time, the Clear Channel Assessment (CCA) binary indicator is periodically checked to determine whether the current 20 μs time slot is busy or not. The backoff countdown process is frozen during busy periods and resumed after the channel has remained idle for a sufficient amount of time, as dictated by the IEEE 802.11 specifications.

While operating according to the 802.11 MAC protocol, the software also maintains a list of counters that reflect the current state of the link and of the network. These procedures realize what we call the *horizontal scanning mode*. Upon request, this operational mode can be suspended to activate the *vertical scanning mode*. In this case, all the functionalities of the 802.11 protocol are temporarily

disabled, and the platform starts collecting Clear Channel Assessment (CCA) samples from the PHY layer according to the selected frequency scanning pattern.

To support ULLA, we introduced a new set of ioctl primitives that map ULLA queries into device specific requests. We exploited the same shared memory used to transfer MAC Packets Data Units (MPDU) from the ARM processor to the DSP, to forward ULLA queries and commands. Inside the DSP, each request is queued and executed when the MAC goes into the idle state (no packets need to be transmitted or received). The number of subsequent requests that can be sent to CalRadio 1 is limited by the fact that, once the ARM has written a new query on the shared memory, it cannot write any other request until the DSP has finished fetching the previous one. In case multiple requests are pushed onto the CalRadio 1 driver, a mutex condition implemented on the shared buffer skips those that cannot be transferred to the DSP, to avoid deadlocks that could freeze and possibly crash the kernel.

4.2 CalRadio 1 PHY exported parameters

The Physical Layer parameters exported by CalRadio 1 are the Average Channel Busy Time and the Energy Burstiness detected on a channel. The former provides the amount of interference that is present on a WiFi channel (20 MHz band), in terms of the number of CCA readings that indicate the medium as busy divided by the total number of measured samples. The board provides access to the Energy Detection threshold used by the integrator circuit to generate the CCA binary signal.

The other indicator reports the average value and standard deviation for the power burstiness detected on a channel. This metric quantifies the average length of a communication or, considering the reciprocal value, the number of detected communications within a certain time frame. From such indicator it is possible to estimate the probability of successfully accessing the media for transmission. This access probability tends to decrease with the number of network entities competing for transmission and can be used to estimate the maximum transmission rate we can achieve. By considering such transmission rate and the collision probability we can determine the maximum goodput a link can sustain on a certain channel.

It is important to consider that, while these indicators are based on the energy detected on a channel, hence providing information on all the technologies competing in the channel band, the way this information is obtained is technology dependent. Specifically, CalRadio 1 is based on an 802.11b transceiver, whose RF reception filter has a bandwith of 20 MHz, as shown in Fig. 4.

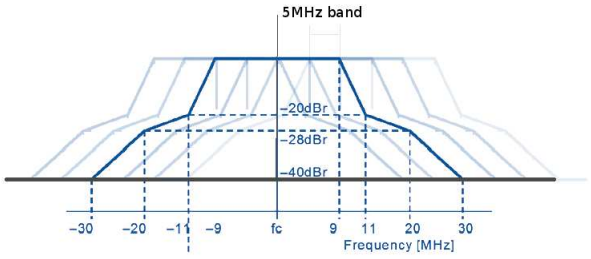

Figure 4: WiFi frequency masks showing channel overlapping

The standard specifies a frequency offset of 5 MHz for each WiFi channel in the $2.4 \div 2.499$ GHz ISM band. The RF transceiver actually permits frequency shifts of 1 MHz (hence potentially violating the standard). Having a frequency resolution of 20 MHz is, on the one hand, disadvantageous as we cannot tell which portions of these 20 MHz are actually being used for communication. On the other hand, this makes it possible to sweep the available frequencies in a shorter amount of time, hence having faster data refresh rates. This is important, for instance, in order to exploit the idle periods that may occur during standard 802.11 functioning to temporarily switch to vertical operation mode and collect useful information from the PHY layer. As reported in [5], the time for a frequency shift of 5 MHz is estimated as 7 μs. The minimum amount of time for CalRadio 1 to gather a reasonable amount of samples for a given channel and to complete the data transfer to the Link Layer Adapter has been estimated to be 10 ms, hence the time required for frequency shifting operations is negligible. To complete a whole scanning procedure of the 2.4GHz ISM band without frequency overlapping in the observation windows we need about 50 ms.

Besides being technology dependent, the acquired information is also hardware dependent: each piece of hardware has its own limitations and advantages; as an example the CalRadio 1 DSP provides access to 72K words of memory for program code and data. This means that no space is left on the DSP for data gathering and storage. For this reason we cannot export the whole trace of the detected energy over time, but can only provide the average value and standard deviation of such variable.

With respect to our implementation of the DSP firmware and to the hardware limitations of CalRadio 1, Tab. 2 summarizes the main features of CRABSS.

CCA reading period of 0.2 μs.
2^{16} CCA collected samples before reporting the total number of busy samples to the ULLA core.
CCA mean and standard deviation values recorded and exported to the ULLA core.
Reception filter bandwidth of 20MHz.
Carrier frequency freely shifted by 1 MHz within the $2.4-2.499$ GHz ISM band.
5 MHz carrier frequency shift performed in less than 7 μs.
Minimum dwell time of 10 ms for each channel.

Table 2: CalRadio 1 PHY sensing features of the CRABSS architecture

It is important to note that the proposed solution encompasses data gathering on multiple selectable frequencies for those parameters such as Average Channel Busy Time and Energy Burstiness that are helpful in order to switch to a new channel if the link conditions degrade below the requirements.

4.3 CalRadio 1 MAC exported parameters

While the physical measurements can be collected on any 20 MHz wide channel within the 2.4 GHz ISM band, the 802.11 MAC layer statistics are available in horizontal mode only and necessarily refer to standard 802.11 channels. MAC layer statistics are used to monitor link quality. When the current link conditions drop below the requirements, we can use the physical layer indicators to select a different channel.

In Tab. 3 we report some of the MAC indicators exported by CalRadio 1. Note that the indicators are collected for *any* link detected by

INDICATOR	DESCRIPTION	VALUE
#PoA	Number of advertised network 802.11 APs (through beaconing)	[0:inf]
#STA	Number of sensed 802.11 Stations	[0:inf]
PER_LINK_EX CHANGED_BYTES	Number of tx/rx bytes for each [STA,AP] couple	[0:inf] [Bytes]
PER_LINK_DATA _VS_ACK	Ratio between the number of sensed packets and the number of corresponding Acks	[1:inf] [#MPDU/#ACK]
PER_LINK_PHY _RATE	Rate adopted by each [STA, AP] couple	[1, 2, 5.5, 11, 6, 9, 12, 18, 24, 36, 48, 54] [Mbps]

Table 3: CalRadio 1 MAC layer exported statistics

CalRadio 1, even though the device is not directly involved in the communication. Furthermore, counters are updated even for packets with erroneous payload field, provided that PLCP and MAC headers are correct.

5. RESULTS

In this section we report and discuss some preliminary results obtained using CRABSS, with the aim of illustrating the potentialities and limits of the proposed solution.

5.1 Channel sensing and technology inference

One of the possibilities enabled by CRABSS is to infer which wireless technologies are active in a certain region of the 2.4 GHz ISM spectrum from the spectrogram provided by CRABSS vertical scanning mode.

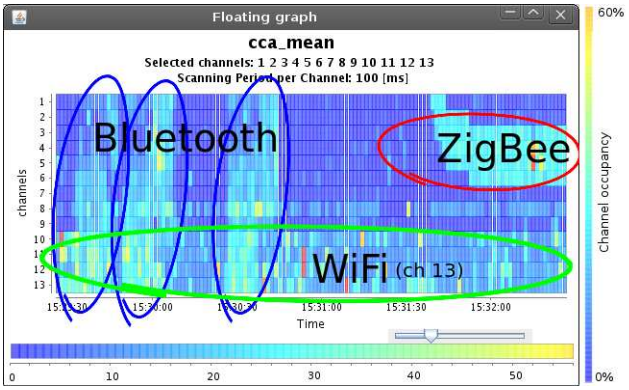

Figure 5: CRABSS Demonstration GUI snapshot

A snapshot of the spectrogram panel shown by the GUI Link User application is reported in Fig. 5, where the major commercial technologies in use in the 2.4 GHz ISM band are emphasized. The graph shows the energy detected on each of the 13 available 802.11 channels over time. Darker color pixels show a time/frequency slot where no interference was detected (a free channel). Lighter color pixels denote an interfered time/frequency slot. In the picture it is possible to observe the presence of an 802.11 network on channel 13 (lower part of the graph), a traffic burst generated by Tmote-Sky [9] Zigbee nodes in the rightmost part of the panel and the frequency hopping interference detected due to Bluetooth traffic for three distinct file transfers.

The data shown in Fig. 5 was obtained by setting a scanning period of 100 ms for each channel. For each period, the average value of the CCA busy readings is returned. If we increase the time resolution of the spectrogram, it is possible to isolate two distinct bursts for each Bluetooth transfer. The first is generated by the paging operations (handshake and synchronization between master and slave nodes), while the latter shows the data transfer itself.

Technology identification algorithms can be used to analyze the interference pattern, signals bandwidth and frequency hopping behavior over time to detect the presence of different frequency overlapping technologies. Hence, a cognitive engine can coordinate nodes and optimize the available resources with respect to the ongoing traffic flows.

We observe that the interference is measured in terms of fraction of CCA samples that report the channel as busy, though we cannot actually measure the interference power level. However, the CalRadio 1 platform makes it possible to set the power threshold used by the CCA circuitry to mark the channel as either idle or busy. In this way, we can adjust the sensitivity of CRABSS to our liking, for instance to reveal only major interference sources.

CRABSS data can be collected either on a very short time scale or for a longer term. While mid term optimization can be performed by knowing the state of the available channels on a short time scale, prediction and coordination become possible on a statistical basis when longer time range datasets are collected and analyzed.

As shown by the graph in Fig. 5, common technologies operating in the 2.4 GHz ISM band can be easily recognized by human inspection. However, the automatic inference of such technologies through pattern matching techniques is more problematic, in particular in case of co-presence of multiple technologies, which may shade each other's spectrum footprint. Another note regards the relationship between the distance of the transmitters and the bandwidth reported by CRABSS. When a node is transmitting nearby, the CalRadio 1 device may detect a wider band interference. As the distance from CalRadio 1 increases, we observe a narrower interfered band as CalRadio 1 reception filter roll-off regions do not capture enough power to indicate the channel as busy any more.

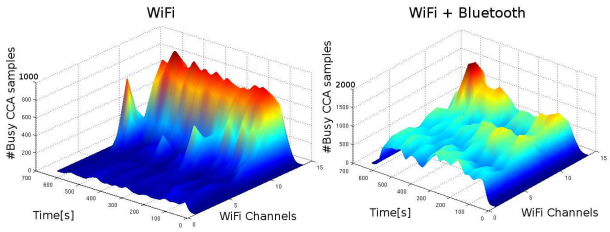

Figure 6: Bluetooth and WiFi overlapping transmissions

Fig. 6 provides a comparison between a 3-dimensional trace of a WiFi beaconing activity on channel 13 (2.472 GHz) and the trace of the interference introduced by a Bluetooth file transfer overlapping with the WiFi signal. The presence of the plateau in the right-hand side graph confirms that CRABSS is actually capable of revealing Bluetooth signals despite the rather fast frequency hopping pattern of this technology. Furthermore, the WiFi beaconing activity can still be detected regardless of the masking effect of the Bluetooth trace and the rather low traffic generated by beacons. Hence, CRABSS preserves the frequency fingerprint of the technologies that are active in the monitored area, potentially enabling their recognition through the analysis of the interference spectro-

gram. This information may enhance the context awareness of a cognitive engine, thus enabling better management of the transmission resources. The technology inference process may be automated by using different classification techniques, such as for example Discriminant Functions, Probabilistic Discriminative Models and so on [10]. The investigation of these techniques is currently in progress.

5.2 Channel switching criteria

In order for channel switching to be advantageous, the performance loss caused by the experienced interference must be larger than the coordination cost for switching. Below we provide a rule-of-thumb criterion that can be used to assess the effectiveness of channel switching. We assume that the link is saturated. Let τ_{obs} be the observation time required to detect an interfered channel and τ_{neg} the time for the two nodes to negotiate a new channel. We consider a worst case scenario in which no useful data can be transferred during the time τ_{sw} required to switch between channels and to re-establish the link between the two nodes. If we denote with R the saturation rate, R_i the interfered rate and T_i the duration of the interference, the switching turns out to be effective when the following relation is satisfied:

$$R \cdot (T_i - \tau_{obs} - \tau_{neg} - \tau_{sw}) > R_i \cdot (T_i - \tau_{obs}) \,. \qquad (1)$$

By inverting (1) we can specify the minimum T_i for which it is convenient to have coordination and switch channel as

$$T_{i_{\min}} = \tau_{obs} + (\tau_{sw} + \tau_{neg})(1 - R_i/R)^{-1} \,. \qquad (2)$$

We observe that the negotiation and switching times become more and more significant as the interfered rate R_i approximates R. Conversely, if the channel is heavily interfered, then channel switching shall be triggered as soon as the duration of the interference burst is expected to be longer than the negotiation plus switching time.

5.3 A case study

As an experiment we consider the transmission with physical rate $R_{PHY} = 250$ Kbit/s in 802.15.4 technology, between two Tmote-Sky wireless sensors operating in the 2.4 GHz ISM band. For this purpose we estimated the time for a frequency shifting operation to be $\tau_{sw} = 1.5$ ms (for adjacent channels).

The experiment was conducted with the following setup:

- carrier sensing mechanism activated
- no acknowledgement/retransmission mechanism
- no interference

In this scenario the two nodes could transmit bursts of 10000 packets of 20 Bytes each with an effective rate $R_{TX} = 20$ Kbit/s and a transmission efficiency

$$\eta = \frac{R_{TX}}{R_{PHY}} = 8\,\%. \qquad (3)$$

We repeated the experiment with two links transmitting with the same setup and measured the new (interfered) rate $R_i = 18.6$ Kbit/s on both links. τ_{obs} can be defined as the time required by CalRadio 1 to sweep the whole 2.4 GHz ISM band, which was estimated to be 50 ms. By applying these results in (2) we obtain

$$T_{i_{\min}} = 71.42\,\text{ms} + 14.28 \cdot \tau_{neg}. \qquad (4)$$

Assuming a three-way negotiation procedures, which requires the exchange of three control packets of 20 bytes each, the negotiation time can be estimated to be $\tau_{neg} = 20 \times 3/R_{TX} = 24$ ms. In these conditions, channel switching makes sense when the interference persists on the channel for a time longer than approximately half a second.

6. CONCLUSIONS

In this work we presented CRABSS, a new framework for interference detection and technology inference. The solution empowers 802.11 MAC protocol with sensing capabilities for multi-technology interference detection and avoidance, while preserving the 802.11 standard functionalities. The framework takes advantage of a modular cross-layer approach to transparently export MAC and PHY statistics to upper layers' prospective users (e.g., a Cognitive Resource Manager engine). The solution was tested by generating interference with the most common commercial technologies in the 2.4 GHz ISM band and proved to be effective in detecting and possibly identifying them. Current and future developments shall focus on creating an automated self-training technology inference algorithm based on energy pattern recognition and machine learning techniques.

Acknowledgment

This work was partially supported by the European Commission through the ARAGORN project (FP7 ICT-216856) and by the US Army Research Office through Grant No. W911NF-09-1-0456.

7. REFERENCES

[1] L. Badia, M. Levorato, F. Librino, M. Zorzi, "Cooperation techniques for wireless systems from a networking perspective," *IEEE Wireless Communications Magazine*, Apr. 2010.

[2] A. Jow, C. Schurgers, and D. Palmer, "CalRadio: A Portable, Flexible 802.11 Wireless Research Platform," in *International Conference On Mobile Systems, Applications And Services, ACM, New York*, 2007.

[3] RWTH Aachen University, "Generic Open Link-Layer API for Unified Media Access," 2006. [Online]. Available: http://ulla.sourceforge.net/

[4] *ANSI/IEEE Std 802.11, 1999 Edition*, IEEE Std., Jun. 2003.

[5] R. Manfrin, A. Zanella, and M. Zorzi, "Functional and Performance Analysis of CalRadio 1 Platform," in *Eighth IEEE International Symposium on Network Computing and Applications, NCA 2009*, Jul. 2009.

[6] Texas Instruments, *TMS320VC5471 Fixed-Point Digital Signal Processor Data Manual*, Dec. 2002.

[7] Interil, *HFA3863 Direct Sequence Spread Spectrum Baseband Processor*, Apr. 2000.

[8] Maxim, *MAX2820/2821 2.4GHz 802.11b Zero-IF Transceivers*, Nov. 2003.

[9] Crossbow, *TelosB Tmote-sky datasheet*, 2006. [Online]. Available: http://www.sentilla.com/pdf/eol/tmote-sky-datasheet.pdf

[10] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, 1st ed. Springer, 2006.