Coexistence Aware Clear Channel Assessment

From theory to practice on an FPGA SDR platform

Peter De Valck, Lieven Tytgat[1], Ingrid Moerman[1], Piet De Meester[1]

All authors are with:
Ghent University – IBCN – IBBT, Gaston Crommenlaan 8, 9050 Ghent, Belgium
Contact email: peter.devalck@intec.ugent.be
{firstname.lastname}@intec.ugent.be1

Abstract. Wireless sensor networks are used by an ever growing number of applications which have ever increasing Quality of Service requirements. The available unlicensed industrial scientific and medical bands – where wireless sensor networks typically operate – are crowded with a number of technologies interfering with each other. Delivering a sufficiently high QoS within these frequency bands is therefore becoming more and more difficult. A theoretic concept named Coexistence Aware Clear Channel Assessment (CACCA) promises more reliable QoS when different technologies utilize the same. Within this paper we propose two methods to perform CACCA and create an SDR prototype to show that CACCA can achieve a high packet error rate reduction in an IEEE 802.15.4 network when it coexists with IEEE 802.11.

Keywords: Coexistence, Sensor Network, Wireless, Interference Avoidance, IEEE 802.11, IEEE 802.15.4, CCA, SDR, WARP, FPGA

1 Introduction

The exponential growth in wireless devices during the last decade has presented a new problem for wireless sensor networks. The increased number of wireless technologies as well as the higher requirements for wireless communication put a high strain on the limited unlicensed spectrum available to these devices. Being resource and energy limited, wireless sensor nodes often get the short end of the stick when confronted with other technologies, resulting in severely degraded communication capabilities as shown in several publications [1 - 4].

Specifications in most communication standards ensure that users of the same technology are capable of coexisting in the same frequency band. Between different technologies however, this coexistence is often limited or nonexistent. Different technologies might still be needed in identical environments to support diverse needs of different applications. Eg. A wireless sensor network using the IEEE 802.15.4 technology might be co-located with IEEE 802.11 stations. The first one is capable of supporting very long term battery powered operation, while the latter is capable of

delivering higher bandwidth connectivity, be it at higher energy cost. Taking a look at the 2.4 GHz band and the interaction between some common digital wireless technologies we see that some standards include support for coexistence (like Adaptive Frequency Hopping for IEEE 802.15.1 – 2005 [5]), while no such provisions exist for the IEEE 802.11 [6] or 802.15.4 [7] standards. Indeed, several studies [1 - 4] have shown that severe throughput degradation can be observed when IEEE 802.11 and IEEE 802.15.4 devices interfere.

In [2] we introduce the concept of Coexistence Aware Clear Channel Assessment (CACCA), which is capable of reducing this interference, fostering cross-technology coexistence. CACCA can be applied on a single wireless technology or on multiple interfering technologies. In the case of IEEE 802.11 and IEEE 802.15.4 devices, the paper concludes that the highest reduction in packet error rate (PER) can be achieved by deploying the CACCA mechanism on IEEE 802.11 devices. Therefore this paper will focus on the implementation of CACCA on an IEEE 802.11 device and the effects on IEEE 802.15.4 communication.

While the concept of CACCA is fully explained in the paper, several factors need to be considered before an actual implementation can be achieved. In this paper we implement CACCA as an extension to existing IEEE 802.11 systems while remaining standards compliant. We used the WARP [8] software defined radio (SDR) as our implementation platform. Keeping in mind the standard compliance and the relative simplicity of the proposed extensions it should be relatively easy to port this effort to existing or future IEEE 802.11 devices. In section II the requirements of this backwards compatible CACCA are considered and two possible implementations are proposed.

In section III a simulation of these solutions is discussed, while the actual implementation is handled in section IV. We experimentally analyze and verify both implementations in section V. Section VI mentions future research and implementation possibilities, ending with a conclusion in part VII. In the scope of this paper, we will refer to the IEEE 802.15.4 standard as ZigBee and to the IEEE 802.11g standard as Wi-Fi.

2 Requirements and solutions

2.1 Coexistence Aware Clear Channel Assessment

In [2] the concept of Coexistence Aware Clear Channel Assessment is introduced as a means to improve coexistence between technologies. It comes down to complementing the existing CCA mechanisms of a technology with additional CCA modules capable of detecting other technologies.

In figure 1.A traditional sensing CCA operation is visualized: before transmitting, the radio will stay in receive mode for a short while and try to determine whether another user is using the channel. Depending on the used protocol, transmissions can be postponed (time based interference avoidance, figure 1.B) or moved to another channel (frequency based avoidance, figure 1.C). To sense the occupancy of the channel, both ZigBee and Wi-Fi support two CCA mechanisms, energy based CCA and preamble detection. For the first method the channel energy is compared to a predetermined threshold while the second method detects the presence of a technology specific sequence.

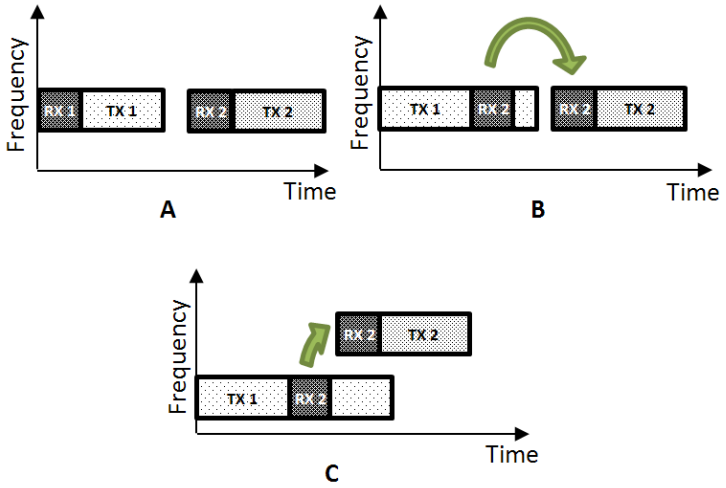

Fig. 1. Basic CCA operating principle (A) with time based avoidance (B) or frequency based avoidance (C)

Without any modification these standard CCA methods will not allow ZigBee and Wi-Fi devices to coexist. Due to the technology specific nature of preamble detection this technique is inherently incapable of detecting other technologies. The simplicity of the energy based CCA means that it could possibly detect other technologies, but in the case of Wi-Fi and ZigBee the different bandwidths and transmission levels mean that ZigBee will be overly sensitive to Wi-Fi, while Wi-Fi will be less sensitive to ZigBee. Theoretical analysis and experiments [1] confirm that both technologies indeed suffer severe throughput degradation when interfering – up to 80% depending on the exact configuration – so this energy based CCA is clearly not sufficient. Therefore a new technique is required to improve coexistence.

CACCA proposes extending a standard CCA with additional methods that are capable of detecting other technologies. This allows the sender to avoid colliding with packets from other technologies as shown in figure 2. Focusing on a ZigBee - Wi-Fi scenario, [2] concludes that the biggest throughput gains can be achieved by adding a

ZigBee CACCA to existing Wi-Fi devices. Additional gains can be achieved by modifying the ZigBee CCA but this was left as a future improvement. This paper will mainly focus on adding a ZigBee friendly CACCA to a Wi-Fi device.

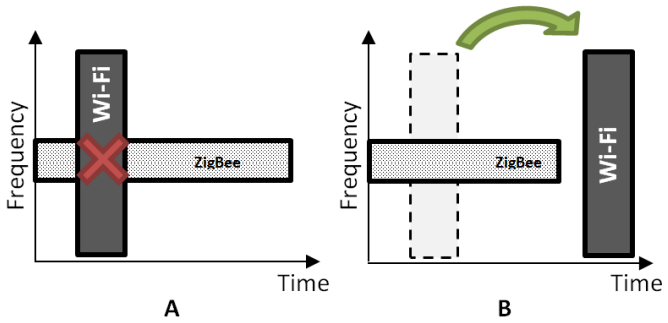

Fig. 2. Interaction between Wi-Fi and ZigBee without (A) and with (B) CACCA.

As mentioned in the introduction, additional design decisions had to be taken to get from the conceptual CACCA presented in [2] to a working implementation. In figure 3 a modified Wi-Fi system is outlined: in a traditional Wi-Fi setup the received signal is passed through an aliasing filter and digitized, after which CCA is performed. Thanks to the high bandwidth of a Wi-Fi receiver, the same digital signal can be used for the detection of ZigBee signals. As each Wi-Fi channel covers multiple ZigBee channels, CCA must be performed on all contained ZigBee channels. To achieve this, the high bandwidth Wi-Fi channel is mixed down to several low bandwidth ZigBee channels and CCA is performed on each channel. While not strictly necessary, we chose to implement this new CCA in a backwards compatible way, complying with the limits imposed on the standard Wi-Fi CCA.

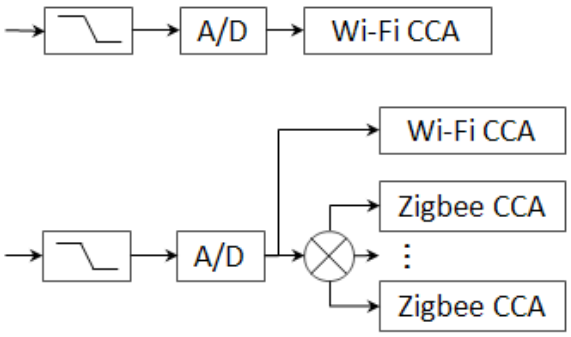

Fig. 3. Standard Wi-Fi CCA (top) and the extended CACCA version (bottom).

2.2 Constraints

Two types of constraints should be considered when a Wi-Fi station is to detect a ZigBee transmission while still conforming to the standard, namely *timing* and detection *sensitivity*.

The Extended Rate PHY (ERP) Wi-Fi standard defines the slot time to be 9 µs [6], consisting of 4 µs of actual channel sensing (CCA Time) and 5 µs for RX – TX turnaround (RxTx_TurnaroundTime). While not strictly necessary, to obtain the highest throughput and limit the additional energy requirements of the CACCA, the implementation should be able to determine the channel state reliably within these 4 µs. Hence not only the CCA itself, but also the signal processing should be finished within the CCA timeframe.

The main purpose of CACCA is avoiding collisions between packets, independent of the technologies used by these packets. Higher level solutions exist to reduce collisions (eg. RTS/CTS, scheduling … [9]) but the cross technology aspect of this problem excludes these solutions. They would require multi technology radios, completely defeating the relative simplicity of CACCA. The CCA sensitivity has as primary target the minimization of the hidden terminal problem with higher sensitivity leading to fewer collisions. CACCA should therefore be capable of detecting signals down to the lowest possible level.

The *timing* and *sensitivity* requirements are in direct conflict with each other as increasing the sensing time improves sensitivity [10]. However, the 4µs limit is a hard limit imposed by the standard and therefore we will strive to achieve the needed sensitivity within the standard defined times.

2.3 Detection methods

The CACCA requires reliable detection of channel state within the available 4µs CCA time. Multiple methods to perform this detection exist.

Energy detection is achieved by averaging the energy within the channel, without filtering the incoming signal. The channel state is determined by comparing the measured energy level with a predetermined threshold as outlined in figure 4. An energy detector can detect a wide variety of signals with a minimal computational overhead due to the fact that no a priori knowledge about the signal is required. However, the detection sensitivity for a specific technology can be improved through the addition of filtering.

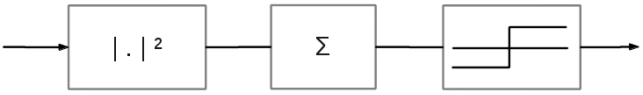

Fig. 4. Simple energy detection architecture

Filtering the received signal with a matched filter before performing the detection (figure 5) will increase the sensitivity [10]. Given complete a priori knowledge of the modulation scheme, this filter corresponds to the receive filter used for demodulation and will provide the best possible detection rate. The modulation scheme used by ZigBee in the 2.4 GHz band is O-QPSK and as such this filter is easily calculated. Not only will the additional filtering increase the processing time, it will also reduce the sensitivity to other signals. While the former reduces the effective sampling time from 4 µs to 3 µs (using a 1 µs filter, figure 6), the latter poses no additional problem for this application.

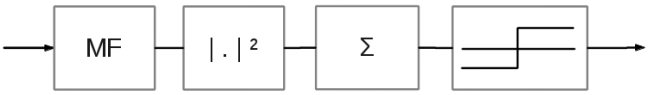

Fig. 5. Simple matched filtering architecture

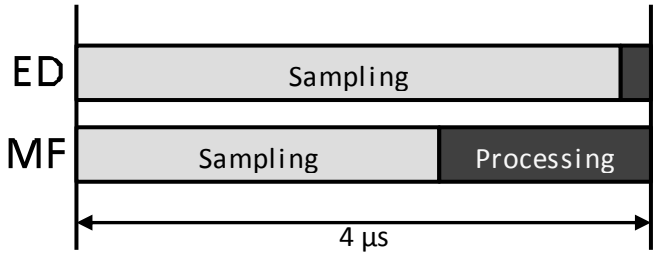

Fig. 6. Sampling and processing periods for energy detection and matched filtering.

3 Simulation

The detection methods were simulated in Matlab to verify their performance before they were implemented in hardware. As shown in figure 7, the simulation is split into three parts: signal generation, propagation and the detection algorithms.

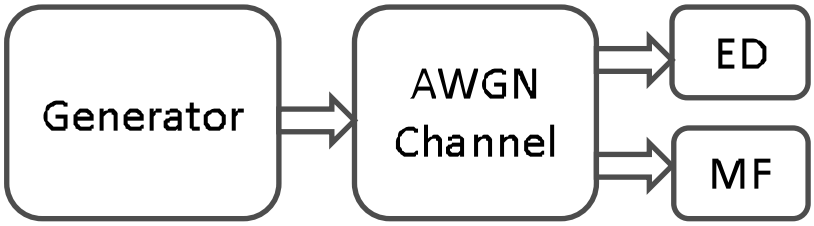

Fig. 7. Simulation overview

The signal generator generates a sample stream containing a ZigBee DSSS (Direct-Sequence Spread Spectrum) signal when the transmitter is active. This stream is passed through an AWGN (Additive White Gaussian Noise) channel model that adds white noise to obtain a predefined SNR (Signal-to-Noise Ratio). Finally, the detection algorithm tries to estimate the original state of the channel state from this noisy signal.

The implementation of the energy detector is fairly straightforward, but the matched filter detector requires the design of an additional filter. This filter is based on the transmit filter described in the IEEE 802.15.4 standard, section 6.5 [7]:

$$p(t) = \begin{cases} \sin\left(\pi \frac{t}{2T_c}\right), 0 \le t \le 2T_c \\ 0, otherwise \end{cases}, \text{with } T_c = \frac{1}{chip\ rate} = \frac{1}{2\ Mchips/s} = 0.5\ \mu s$$

To obtain the corresponding receive filter, the time reversed complex conjugate of this filter is taken, resulting in the filter shown in figure 8.

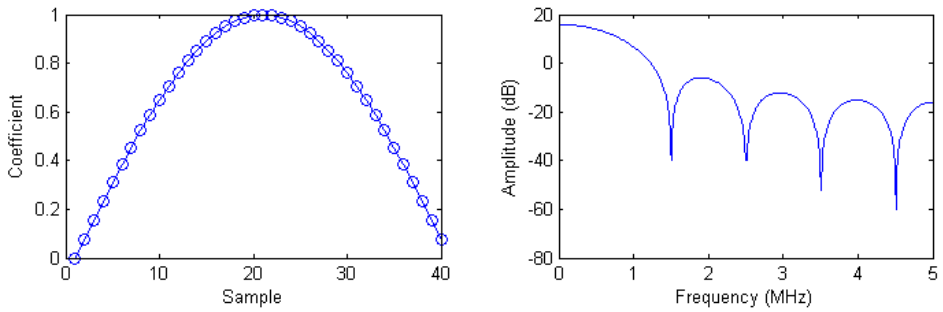

Fig. 8. The matched filter and its frequency response

Detection sensitivity is measured by varying the channel SNR and comparing the output of the detector with the actual channel state. Random 4 µs intervals were sampled by the detector and used to determine channel state.

The detection mechanisms should detect the presence of a ZigBee signal as reliably as possible, albeit keeping the false positive sufficiently low. Without this limit the Wi-Fi transmitter would always sense the channel as busy even though it is free. Therefore we chose the threshold for detection to keep the false positive rate (channel estimated as busy when it is free) below 5%.

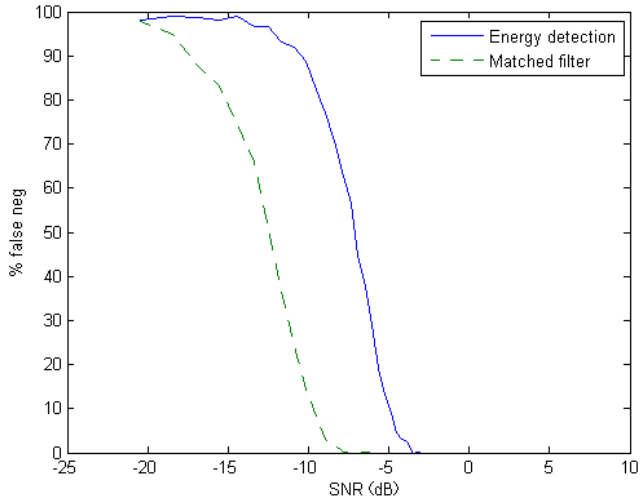

Fig. 9. Simulation results: false negative rate for MF (- -) and ED (—)

The results presented in figure 9 show that both methods are capable of detecting a ZigBee signal successfully. When keeping the false negative rate below 10%, matched filtering is reliable for SNR down to -10 dB, while energy detection is capable of detecting signals with an SNR of -5 dB. Overall, matched filtering provides a gain of approximately 5dB over energy detection, although both methods are suitable for detection.

To verify the correctness of the Matlab implementation, the detection algorithms were also applied to raw data captured from a ZigBee transmission. The processing was done off-line, but we were able to verify that the implemented algorithms are indeed capable of detecting the presence of real signals.

4 Implementation

The very strict timing requirements (section 2.2) mandate a hardware platform that allows both low-level and low-latency access to the RF signal. Several research platforms allow low-level access to the RF signal, but the low-latency requirement rules out host-based platforms like the USRP [11] or SORA [12]. Embedded solutions like the WARP [8] or the Sundance MIMO series [13] do not suffer this drawback and are therefore more suited for this task.

For our implementation the WARP (figure 10) was chosen. This device combines a powerful Virtex-4 FPGA with an RF frontend supporting bandwidths up to 40 MHz in the 2.4 and 5 GHz bands. The reconfigurable logic of the FPGA allows for high speed, low-latency processing of RF signals, while the processor embedded in the FPGA can handle sequential control tasks. Thanks to the tight integration between the processing and control systems, any communication overhead is kept to a minimum and most timing constraints can be met.

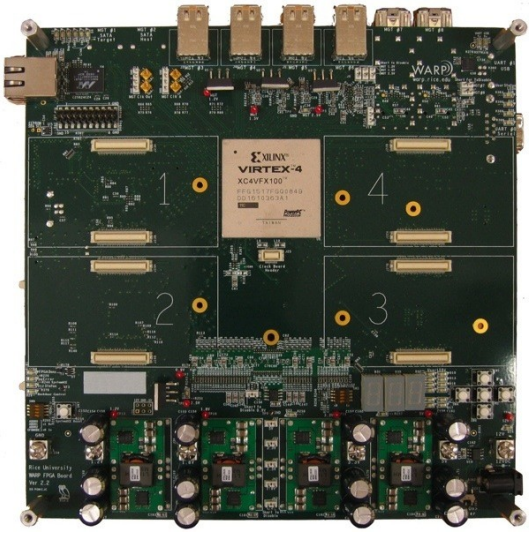

Fig. 10. The Wireless open-Access Research Platform SDR with Virtex-4 FPGA

We originally planned on modifying the CCA of an existing 802.11g implementation to perform CACCA. However, an adequate Wi-Fi MAC for the WARP is lacking so the effort was focused on developing independent detection cores. While a Wi-Fi-like PHY layer is available – the OFDM reference design – extensive modifications and a new MAC layer would be needed. While this would be an interesting topic in itself, we limited the implementation efforts to the CACCA detection cores. Compared to a complete wireless stack, the complexity of these detection cores will be very limited.

The energy detection and matched filtering algorithms were implemented in the reconfigurable logic of the FPGA as separate cores using the System Generator provided by Xilinx and Simulink. A simple transmit core was also designed to replay prerecorded sample streams.

The final system consists of these cores along with several standard cores from Xilinx and the WARP project. This hardware is connected by the Processor Local Bus (PLB) and a Local Link (LL) connection and is controlled by the software running on the PowerPCs embedded in the FPGA to provide the required functionality:

- Communication with a host PC to control experiments is handled by the serial and Ethernet cores. To support high throughputs the Ethernet core is also connected to the DDR2 memory installed on the WARP.
- RF control is handled by the aforementioned detection and transmission cores. Communication with the actual RF frontend is provided by the WARP radio core that presents an abstracted interface to the other RF cores.
- Additional direct feedback is supplied by several IO cores driving external LEDs.

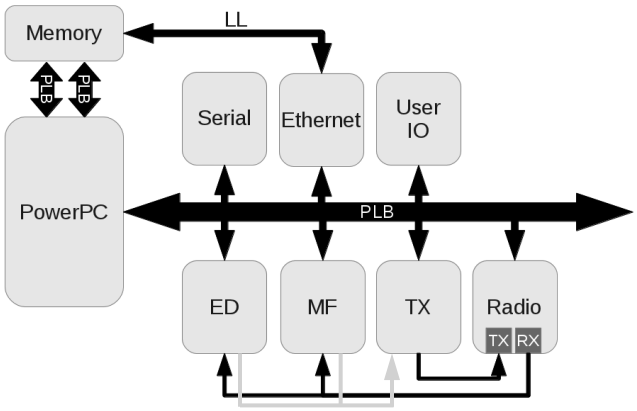

Fig. 11. Architectural overview of the system implemented on the WARP hardware

The complete system (figure 11) is capable of both detecting ZigBee signals using energy detection and matched filtering and interfering with these signals by transmitting pre-recorded samples. While it is capable to function as a stand-alone device, the software running on the PowerPC allows us to reconfigure the system at runtime and display measurement results when performing experiments using an external PC.

5 Experimental analysis

5.1 Setup

The IBCN research group has access to a controlled RF environment consisting of four shielded enclosures and a network of variable attenuators. Devices to be tested are placed inside these enclosures to minimize external interference and the variable attenuators control the coupling between the enclosures. This setup allows complete control of the RF environment and makes repeatable RF experiments possible. A WARP device was placed in one box and connected to 3 ZigBee nodes in the other boxes through the attenuator network (figure 12). Thanks to the flexible attenuation network this setup can be used to simulate most scenarios involving these devices without moving any hardware.

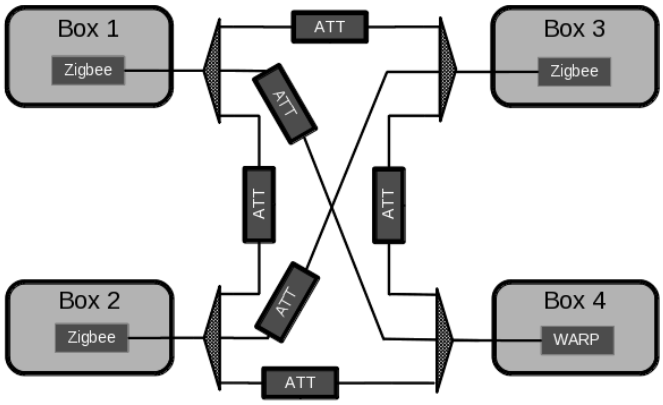

Fig. 12. The wireless test setup at the IBCN research group

5.2 Experiments

Experiments were performed to determine the detection rate and measure the influence of an interferer using CACCA on a standard link.

Sensitivity was measured by varying the attenuation between a transmitting ZigBee node and the WARP. Along with the RF signals, an additional IO signal was routed between the two nodes to indicate the radio state. This line was controlled by the transmitting node and provided the WARP with the state of the radio on the transmitting node. To reduce the influence of timing differences between the radio chip and the microcontroller on the transmitting node, results in a 20 µs interval around a transition on the IO line were discarded. The threshold was again chosen so the false positive rate was below 5%. The configuration of this experiment can be seen in figure 13.

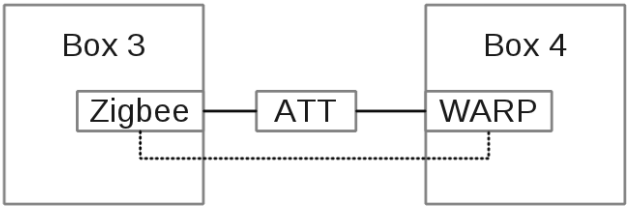

Fig. 13. Sensitivity experiment setup

The effect of CACCA on the goodput of a ZigBee link was measured in a second experiment. Two ZigBee nodes and a WARP were configured in a triangle setup as seen in figure 14: the WARP and the transmitting node were connected directly to the receiving node while a variable attenuator controlled the strength of the signal from the transmitter to the WARP. Interference is generated by the WARP performing CACCA and transmitting short prerecorded Wi-Fi fragments.

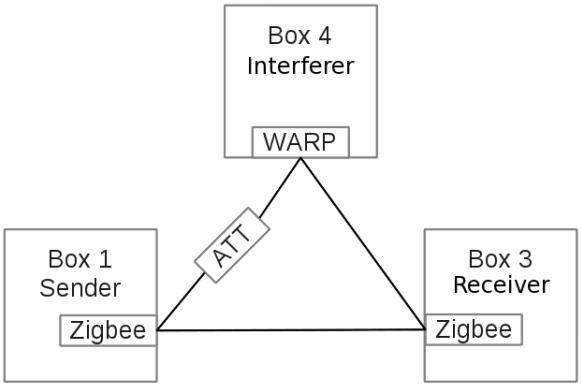

Fig. 14. Goodput experiment setup

5.3 Results

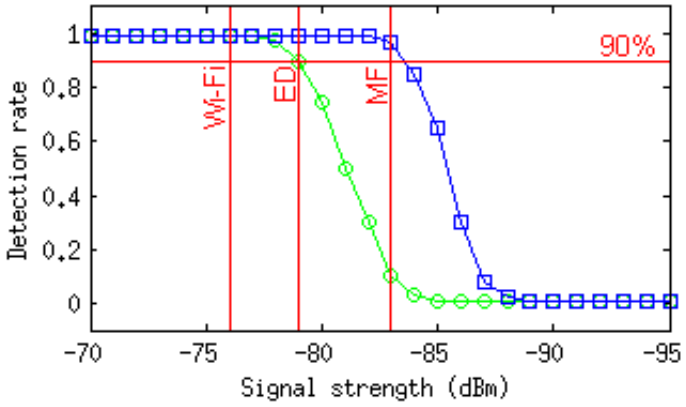

Fig. 15. Detection rate of a busy channel for energy detection (○) and matched filtering (□)

In the first experiment the detection rate of the two CACCA methods was measured by comparing the channel state estimations with the actual channel state. Postulating a 90% reliability, energy detection is capable of reaching this goal for signals down to -79 dBm, while matched filtering can handle signals down to -83 dBm as shown in figure 15. While the differences between both methods are not as big as expected from the simulations, there is still an improvement of about 4 dB.

Comparing these results to the thresholds specified in the Wi-Fi standard we see that both methods perform significantly better. According to the standard the Wi-Fi CCA requires 90% reliable detection of a signal at -76 dBm. For energy detection the improvement is limited to 3 dB, while matched filtering gives a more significant improvement of 7 dB.

In the ZigBee standard the CCA threshold is specified to be -85 dBm for a sampling time of 128 µs. While reaching this target is not necessary for our case (this implementation is targeted at Wi-Fi systems), comparing the results shows that energy detection is not capable of reaching this threshold in a 4 µs sampling window but matched filtering only falls short by 2 dB.

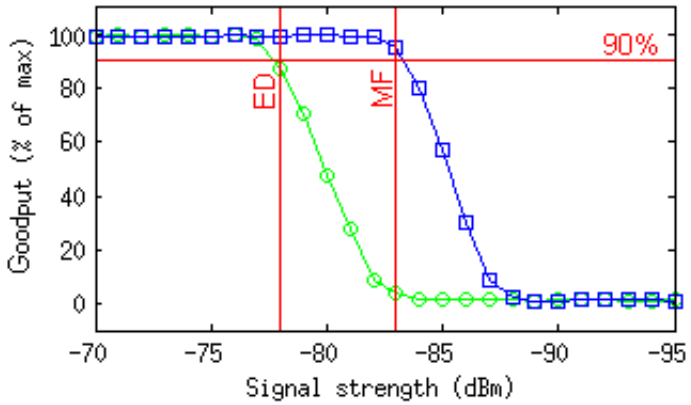

Fig. 16. Goodput for energy detection (○) and matched filtering (□)

In the second experiment the goodput between two standard ZigBee devices is measured while interference is generated by transmitting prerecorded Wi-Fi signals and using different CACCA methods. The results of this experiment are shown in figure 16. When using no CCA, the ZigBee traffic is completely drowned out by the Wi-Fi signal. The goodput is improved by using both CACCA methods: packet error rates < 10% can be achieved using the energy detection method when the signal strength at the interfering Wi-Fi station is above -78 dBm or above -86 dBm when using matched filtering.

6 Future work

Considering the work presented in this paper, two main areas of improvement can be identified:

The first is *extending* the CACCA to other protocols: the current implementation is focused on the interaction between IEEE 802.15.4 and IEEE 802.11g, but given the plethora of wireless standards and devices, the addition of other technologies like IEEE 802.15.1 will improve coexistence even more.

Secondly, *integration* of the detection cores: due to the low complexity of the detection methods compared to a complete wireless system, it should be possible to include the CACCA in a complete wireless stack. This would allow for additional testing and could possibly lead to an implementation in commodity hardware, as the current hardware is fairly specialized and expensive.

7 Conclusion

Wireless Sensor Networks are increasingly co-located with Wi-Fi, resulting in a decrease of its performance due to an increase in packet loss. Coexistence Aware CCA promises to reduce this packet loss significantly by extending existing CCA methods with methods capable of detecting other technologies. Within this paper we proposed a possible implementation method, implemented a prototype of a Wi-Fi side ZigBee CACCA and experimentally verified its operation.

The main goal of the implementation presented in this paper was remaining backwards compatible with the Wi-Fi standard and keeping the implementation as simple as possible. Therefore we chose the timeframe after which CACCA needs to deliver its channel assessment to be 4 μs, as specified in the Wi-Fi standard. Within this timeframe two possible CCA methods are viable, namely energy detection CCA and matched filtering CCA. With a Matlab simulation we concluded that the ED based approach can reliably detect ZigBee with an SNR down to -5 dB, while MF improves this down to -10 dB.

Both approaches were implemented on the WARP SDR platform, using a combination of software running on the embedded processor and dedicated hardware cores to meet the timing requirements. The original goal of modifying a complete Wi-Fi implementation was abandoned due to the lack of an existing Wi-Fi MAC implementation. Instead Wi-Fi interference was simulated by replaying prerecorded Wi-Fi samples. While this is no optimal solution, it allowed us to perform relevant experiments. The final system is capable of performing ZigBee CACCA and interfering with ZigBee traffic in a Wi-Fi like way.

In a first experiment we verified that the sensitivity of both CACCA methods is indeed better than the default sensitivity specified in the Wi-Fi standard. Compared to the -76 dBm standard, both energy detection (-79 dBm) and matched filtering (-84 dBm) offer significant improvements. A second experiment showed that ZigBee is still able to deliver the maximum throughput when its signal is received stronger than -84 dBm at the Wi-Fi sender. Concluding we can say that this proof of concept clearly shows the benefits of CACCA for technologies operating in the crowded ISM band.

As a final note we would like to address the business opportunities of CACCA. At first sight its business case might seem unclear, for we propose an enhancement to the popular Wi-Fi standard to obtain performance gains for ZigBee devices. However, in [14] we show that due to the relatively low implementation complexity and the more consolidated platform – where one chipset is used in a wide range of devices (ex. PC, laptop, table and phone) – viable business opportunities are feasible.

References

1. S. Pollin, I. Tan, B. Hodge, C. Chun, A. Bahai, “Harmful coexistence between 802.15.4 and 802.11: a measurement-based study”. *Proceedings of 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, pp. 1-6, 2008
2. Tytgat, L. and Yaron, O. and Pollin, S. and Moerman, I. and Demeester, P. “Avoiding collisions between IEEE 802.11 and IEEE 802.15. 4 through coexistence aware clear channel assessment”. *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, 2012
3. G. Thonet, P. Allard-Jacquin, P. Colle, “ZigBee – WiFi Coexistence White paper and Test Report,” [online], Available: http://www.zigbee.org
4. Wei Yuan, Xiangyu Wang, Linnartz, J.-P.M.G. , "A Coexistence Model of IEEE 802.15.4 and IEEE 802.11b/g," *14th IEEE Symposium on Communications and Vehicular Technology in the Benelux*, pp.1-5, Nov. 2007
5. IEEE Std. 802.15.1 - 2005, IEEE Standard for Information Technology - Telecommunications and Information exchange between systems - Local and metropolitan area networks - Specific requirements – Part 15.1: Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPANs)
6. IEEE Std. 802.11 - 2007, IEEE Standard for Information Technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications
7. IEEE Std. 802.15.4 - 2006, IEEE Standard for Information Technology - Telecommunications and Information exchange between systems - Local and metropolitan area networks - Specific requirements – Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)
8. “Rice University WARP Project,” [online], Available: http://warp.rice.edu
9. J.H. Schiller, “Medium Access Control” in *Mobile communications,* 2nd ed., Addison Wesley, 2003
10. Tandra, R. and Sahai, A. “SNR walls for signal detection”. *IEEE Journal of Selected Topics in Signal Processing,* vol. 2, pp. 4-17, 2008.
11. “Universal Software Radio Peripheral,” [online], Available: http://www.ettus.com
12. “Sora Project,” [online], Available: https://research.microsoft.com/en-us/projects/sora
13. “MIMO 4x4,” [online], Available: http://www.rapiddevsys.biz/product_info.php?products_id=72
14. L. Tytgat, M. Barrie, V. Gonçalves, O. Y. Yaron, I. Moerman, P. Demeester, S. Pollin, P. Ballon, S. Delaere: “Techno-economical Viability of Cognitive Solutions for a Factory Scenario”, published in *2011 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, Aachen, Germany, 3-6 may 2001, pp.182-192