RESEARCH   **Open Access**

Avoiding collisions between IEEE 802.11 and IEEE 802.15.4 through coexistence aware clear channel assessment

Lieven Tytgat[1*], Opher Yaron[1], Sofie Pollin[2], Ingrid Moerman[1] and Piet Demeester[1]

Abstract

More and more devices are becoming wirelessly connected. Many of these devices are operating in crowded unlicensed bands, where different wireless technologies compete for the same spectrum. A typical example is the unlicensed ISM band at 2.4 GHz, which is used by IEEE 802.11bgn, IEEE 802.15.4, and IEEE 802.15.1, among others. Each of these technologies implements appropriate Media Access Control (MAC) mechanisms to avoid packet collisions and optimize Quality of Service. Although different technologies use similar MAC mechanisms, they are not always compatible. For example, all CSMA/CA-based technologies use Clear Channel Assessment (CCA) to detect when the channel is free; however, in each case it is specifically designed to improve detection reliability of the specific technology. Unfortunately, this approach decreases the detection probability of other technologies, increasing the amount of cross-technology collisions. In this article, we introduce the concept of coexistence aware CCA (CACCA), which enables a node operating in one technology to backoff for other coexisting technologies as well. As a proof of concept, we analyze the Packet Error Rate(PER) incurred by an IEEE 802.15.4 network in the presence of IEEE 802.11bg interference, and assess the PER reduction that is achieved by using CACCA.

Keywords: IEEE 802.15.4, IEEE 802.11, coexistence, clear channel assessment (CCA), coexistence aware clear channel assessment

1. Introduction

Wi-Fi has since long been the major wireless technology connecting PCs with each other. Lately, we observe an evolution from powerful wireless devices to lightweight-embedded devices, while at the same time their density is increasing. The number of such wireless devices is expected to become an order of magnitude bigger than the current number of PCs, as can be seen in Figure 1. In addition, new types of application areas introduce new wireless communications solutions, which employ a variety of wireless technologies.

The problem when using different wireless technologies in the same frequency band is that most of them are not designed to be compatible with each other. Even if different technologies use a similar Medium Access Control (MAC) protocol, they might still impede each other.

Within this article we study the collisions between two heterogeneous CSMA/CA-based MAC technologies. As a proof of concept, we analyze the collisions between IEEE 802.11bg and IEEE 802.15.4. Throughout the article, we refer to IEEE 802.11bg with the term Wi-Fi, and to IEEE 802.15.4 with the term Zigbee. Note that IEEE 802.15.4 only defines the physical (PHY) layer and MAC layer, in contrast to Zigbee that also specifies higher layers of communication above IEEE 802.15.4. However, for the sake of simplicity we use the terms IEEE 802.15.4 and Zigbee to denote the same thing.

The co-existence behavior of Wi-Fi and Zigbee has been studied extensively. The physical layer effects of Wi-Fi and Zigbee coexistence are already considered in the IEEE 802.15.4 standard [1]. Zhen et al. [2] study the cross-technology detection probability of Clear Channel Assessment (CCA) between Zigbee and Wi-Fi. They conclude that Zigbee is oversensitive to Wi-Fi, while Wi-Fi is insensitive

* Correspondence: lieven.tytgat@intec.ugent.be
[1]Department of Information Technology - IBCN, Ghent University - IBBT, Gaston Crommenlaan 8 (Bus 201), 9050 Gent, Belgium
Full list of author information is available at the end of the article

Figure 1 Projected number of devices. *Source*: Morgan Stanley [15].
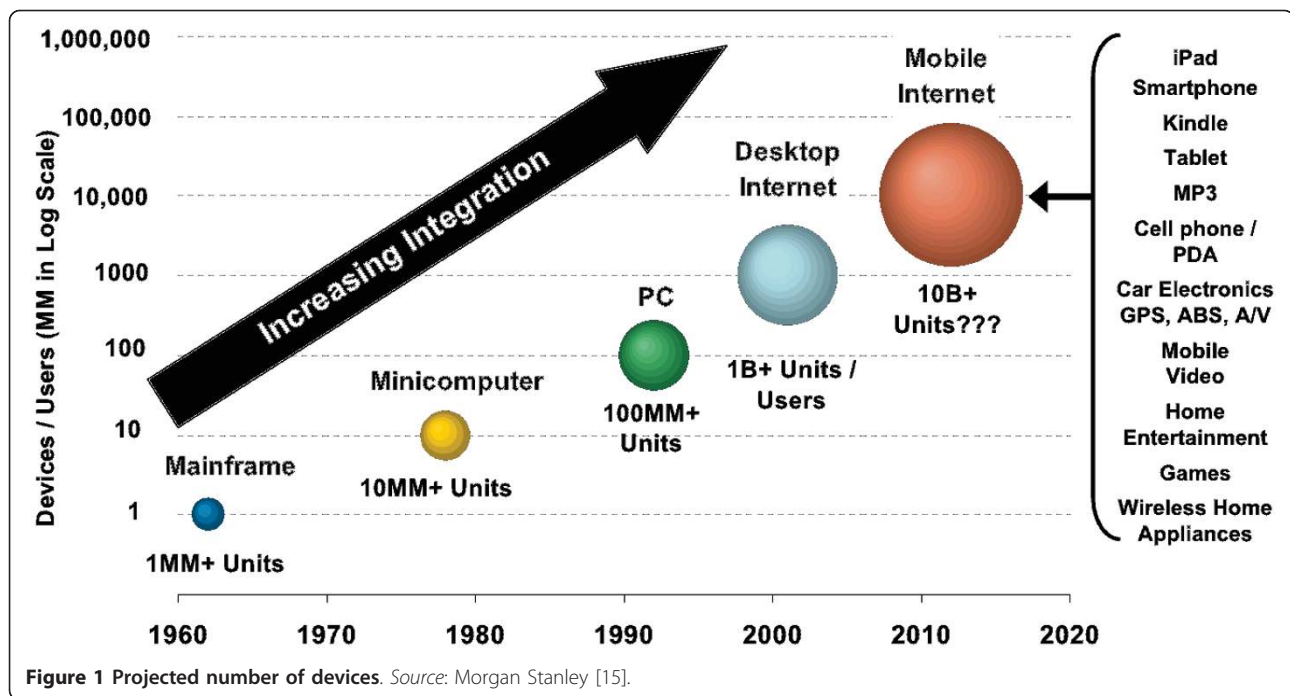
to Zigbee beyond a Heterogeneous Exclusive CCA Range, which they calculate to be 25m with the free space path-loss model. Yuan et al. [3] study the co-existence behavior of Zigbee and saturated Wi-Fi. They conclude through a model and simulation that 5.75% of the Zigbee throughput remains under the assumption that Wi-Fi and Zigbee CCA can avoid all cross-technology collisions. They also conclude through simulation that no throughput remains in case Wi-Fi does not detect Zigbee. Pollin et al. [4] measure the coexistence impact of Zigbee and Wi-Fi. They conclude that standard Wi-Fi devices do not backoff for Zigbee traffic, even in very close proximity. They also show that the CCA mechanism of Zigbee can reduce collisions with Wi-Fi, but it is too slow to avoid all Wi-Fi traffic. Thonet et al. [5] measure up to 85% Zigbee packet loss due to 802.11b traffic. Consequently, we conclude that Zigbee might incur severe packet loss when it coexists with Wi-Fi. However, no model predicting the performance degradation has been proposed. Out of [1], it is possible to determine the Packet Error Rate (PER) depending on the signal-to-interference ratio (SIR) and the size of the collision window, given there is a collision. However, the amount of collisions is dependant on the channel access mechanism of both Wi-Fi and Zigbee. Hence, a detailed model for cross-technology collisions that considers realistic Wi-Fi and Zigbee channel access mechanisms is a key open issue. In [6,7], we propose such a model and focus on exploring the economic value of introducing sensing engines in one specific business scenario. In this article, we focus on a thorough theoretical study of this

model, and verify it against real-life measurements in a testbed environment.

The remainder of the article is organized as follows. In Section 2, we analyze the CCA-based medium access in Wi-Fi and Zigbee. In Section 3, we derive the Zigbee PER model under Wi-Fi interference, look at the sensitivities it has and verify it through measurements. Out of this model, the major mechanism leading to the high Zigbee PER is identified. In Section 4, we analyze the different Coexistence Aware CCA (CACCA) implementation alternatives, and the implications of using a spectrum sensing engine as a CACCA agent. Section 5 gives an overview of potential topics for further research, while Section 6 concludes this article.

2. CCA operating principle

The operating principle of a CCA-based MAC consists of three steps, as depicted in Figure 2. Prior to any transmission, the radio remains in receive-mode for a time window of length T_{CCA}, during which it measures the average received power. If it is above a certain threshold, the radio assumes the channel is busy, and backs-off. Otherwise, the radio switches to transmit mode–which takes T_{Rx2Tx}–and starts to transmit the packet.

Both Wi-Fi and Zigbee use CCA, however their operating parameters such as Bandwidth (BW), Power Spectral Density (PSD), and timing (duration of the CCA window, packet time, etc.) differ, as can be seen in Table 1.

Figure 2 CCA-based packet transmission.
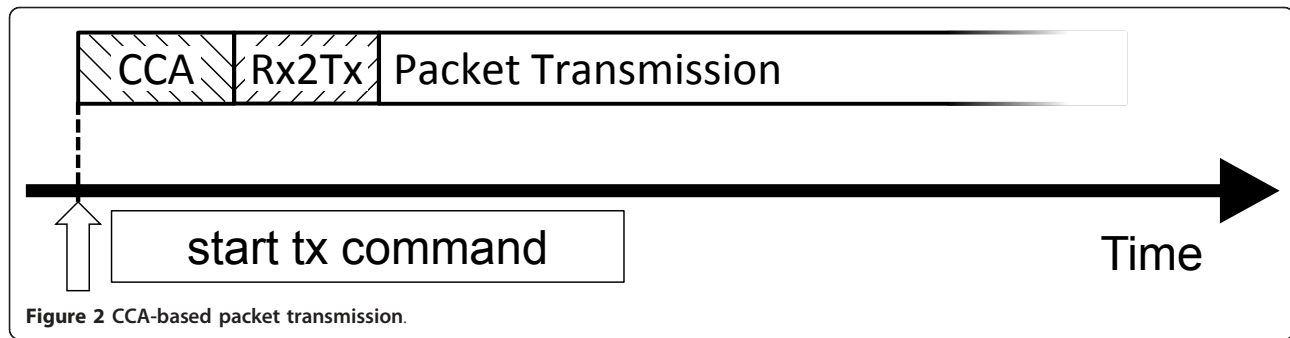
The difference in bandwidth and power–of which a spectral diagram is given in Figure 3–results in a difference in detection sensitivity. With a bandwidth of 22 MHz, Wi-Fi CCA captures the full power of both Wi-Fi and Zigbee transmissions. Zigbee transmits at 0 dBm, which is 20 dB lower than the Wi-Fi transmission, resulting in a 20-dB lower sensitivity to Zigbee than to Wi-Fi. On the other hand, with a bandwidth of only 2 MHz, Zigbee CCA captures the full power of other Zigbee transmissions in the same channel, but only 2/22th–or -10.4 dB–of the Wi-Fi transmit power, resulting in a 9.6-dB higher sensitivity to Wi-Fi than to Zigbee. These simple calculations support the observations of [2] that we mentioned earlier.

Both Wi-Fi and Zigbee allow preamble detection instead of energy detection as CCA. Preamble detection can improve sensitivity, but prevents cross-technology detection due to the differences in preambles between technologies. In Zigbee, this is usually disabled as the sensing time defined by the standard is sufficiently long to allow adequate sensing sensitivity. However, Wi-Fi enables this by default in order to reach the maximum sensing sensitivity within the short Wi-Fi CCA timeframe. We can therefore assume that standard Wi-Fi does not backoff at all for Zigbee traffic.

3. Zigbee PER under Wi-Fi interference
3.1. Analytical PER model
In the following, we assume that every collision between a WiFi packet and a Zigbee packet results in the Zigbee packet being lost. Although this is undoubtedly an

Table 1 Wi-Fi and Zigbee parameters [1,16]

	Zigbee	Wi-Fi
BW (MHz)	2	22
Tx power (dBm)	0	20
PSD (dBm/MHz)	-3	6.6
T_{CCA} (µs)	128	< 4
T_{Rx2Tx} (µs)	192	< 5
Min. packet time (µs)	320	28
Max. packet time (µs)	4256	12416

oversimplification, it allows us to clearly show the plausible PER reduction through the usage of CACCA.

We focus on the packet loss in the Zigbee network under Wi-Fi interference. For the sake of convenience, we sometimes use the Packet Success Rate (PSR), which is defined by

$$PSR: = 1 - PER \tag{1}$$

We identify three sources for Zigbee Packet Errors. First, there is packet loss due to the received Zigbee signal being too low compared to the radio noise ($PER_{Z,SNR}$). Second, Zigbee packets can get lost because of collisions with other Zigbee packets ($PER_{Z,Z}$). Finally, Zigbee packet loss can occur because of collisions between Zigbee and Wi-Fi packets ($PER_{Z,W}$). These independent events are not mutually exclusive, hence the total PER_Z is smaller than or equal to their sum. Moreover, $PER_{Z,W}$ is only one of the sources of PER_Z and thus also smaller than or equal to PER_Z. For the total Zigbee PER (PER_Z), we have

$$PER_{Z,W} + PER_Z + PER_{Z,W} + PER_{Z,Z} + PER_{Z,SNR} \tag{2}$$

$PER_{Z,SNR}$ has been studied extensively, as described for example in [1]. In addition, we will not discuss $PER_{Z,Z}$ in detail in this article. Nevertheless, under normal operating conditions–which means low load in the sensor network and sufficient link budget–$PER_{Z,Z} \approx 0$ and $PER_{Z,SNR} \approx 0$. Consequently,

$$PER_Z \approx PER_{Z,W} \tag{3}$$

In Figure 4, we illustrate the possible interactions between Wi-Fi and Zigbee broadcast traffic. Remember that Wi-Fi CCA does not detect Zigbee transmissions; therefore, the CCA and Rx2Tx windows of Wi-Fi are not visualized in Figure 4. In case 1, a Wi-Fi transmission starts and finishes without interaction with Zigbee, and thus no collision occurs. In the second case, a Wi-Fi packet starts close before the Zigbee device starts its CCA. Hence, the Zigbee CCA window will completely be overlapped by the Wi-Fi transmission and Zigbee will sense the channel as busy. In case 3, the Wi-Fi packet starts earlier than a certain percentage–β–of the CCA
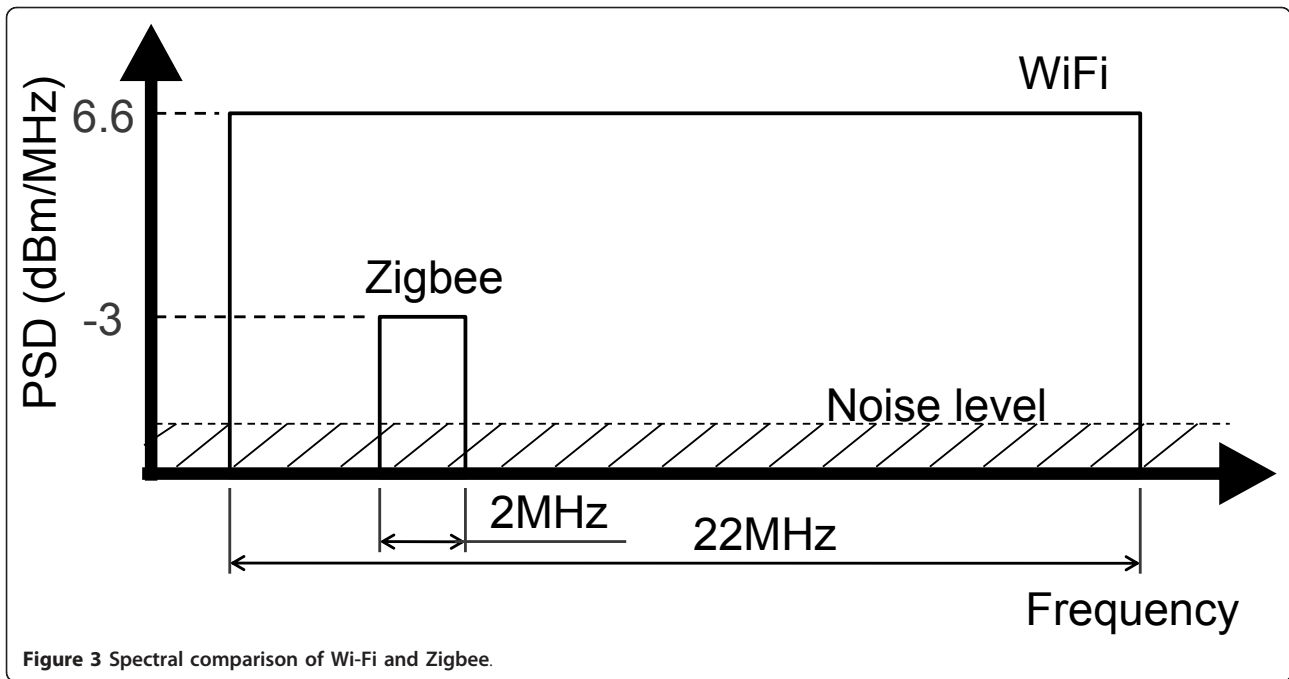
Figure 3 Spectral comparison of Wi-Fi and Zigbee.

window. β is defined as the percentage a transmitted Wi-Fi packet needs to cover the Zigbee CCA window in order for the Zigbee device to assess the channel as busy. Therefore, the Zigbee device will backoff, avoiding a collision. In case 4, the Wi-Fi packet starts beyond the β boundary, resulting in Zigbee assessing the channel as free. Initially, we assume that all Wi-Fi packets are longer than $\beta \times T_{Z,CCA} + T_{Z,Rx2Tx}$; therefore, the Wi-Fi packet

will have some overlap with the actual Zigbee packet, which will result in a collision. Later on we will also examine our model with shorter Wi-Fi packets. Finally, in the fifth case, the Wi-Fi packet starts during the Zigbee packet, resulting in a collision. We conclude that a collision happens whenever a Wi-Fi transmission starts during the $\beta \times T_{Z,CCA} + T_{Z,Rx2Tx} + T_Z$ timeframe of a Zigbee transmission. We further assume that all collisions
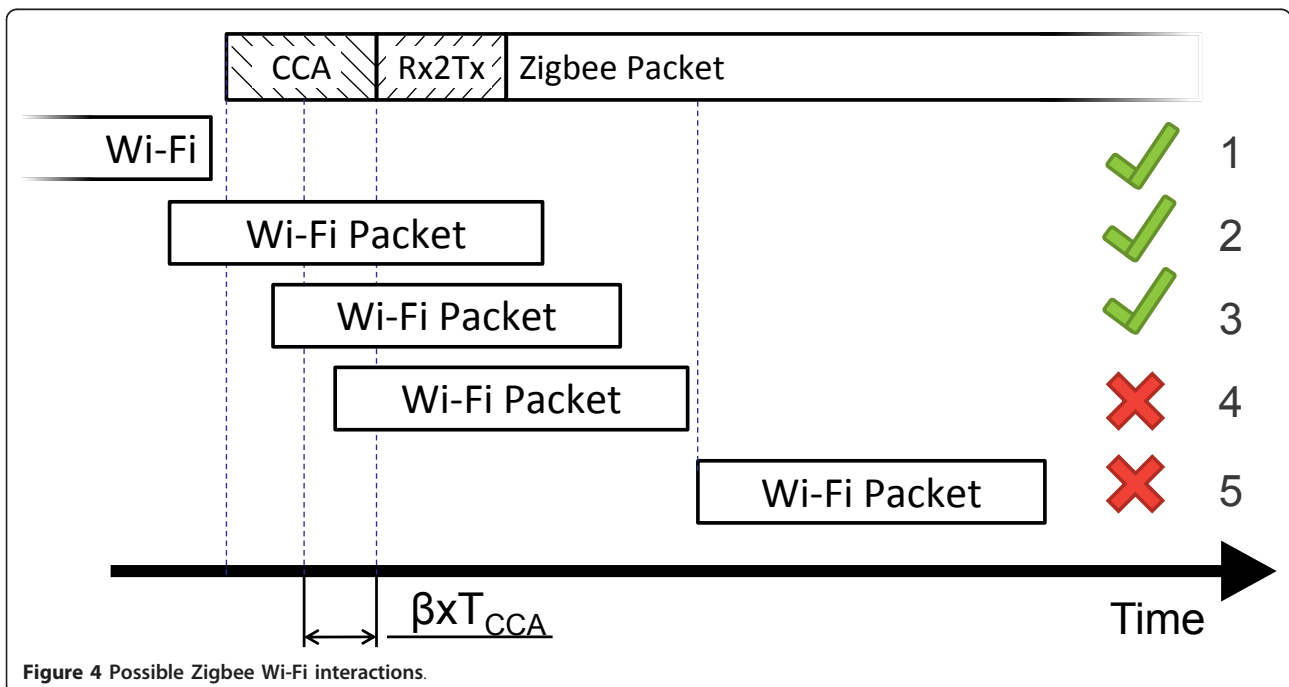

Figure 4 Possible Zigbee Wi-Fi interactions.

result in packet losses, and therefore the probability of not losing a packet, $\text{PSR}_{Z,W}$, equals to the probability of not having a Wi-Fi transmission starting during this time frame, which can be written as

$$\text{PSR}_{Z,W} = \Pr\left(t_{\overline{W}} > \beta \times T_{Z,\text{CCA}} + T_{Z,\text{Rx2Tx}} + T_Z\right) \quad (4)$$

with

$t_{\overline{W}}$ is a random variable that represents the time until the current Inter Packet Delay (IPD) of Wi-Fi terminates and a new Wi-Fi packet starts, T_Z is the average Zigbee packet length, $T_{Z,\text{CCA}}$ is the Zigbee CCA time, $T_{Z,\text{Rx2Tx}}$, is the Zigbee Rx to Tx turnaround time.

Since Wi-Fi CCA does not detect Zigbee transmissions, the instants of time at which Wi-Fi transmissions start are independent of the Zigbee transmissions. We assume that the distribution of Wi-Fi IPD can be approximated by the exponential distribution, with average $T_{\overline{W}}$. Note that it is typically assumed that the Wi-Fi IPD has a self-similar distribution (i.e., traffic bursts). However, traffic bursts can be divided into periods of intense traffic, and periods of less intense traffic. Within each period, we assume the distribution of IPD can reasonably be approximated by the exponential distribution, respectively, with a high and a low rate. This assumption allows to determine the $\text{PER}_{Z,W}$ during intense traffic as well as during low traffic periods, which is the major intent of this study.

Under these assumptions we can write

$$\text{PSR}_{Z,W} \approx e^{-\dfrac{\beta \times T_{Z,\text{CCA}} + T_{Z,\text{Rx2Tx}} + T_Z}{T_{\overline{W}}}} \quad (5)$$

Note that $T_{\overline{W}}$, T_Z, and β are variables, while $T_{Z,\text{CCA}}$ and $T_{Z,\text{Rx2Tx}}$ are constants that are defined by the Zigbee standard [1] (see Table 1).

In the remainder of this article, we use typical default values for the various parameters, as specified in Table 2 unless explicitly noted otherwise. In addition, we use a default value of $\beta = 1$. In Table 2 the MAC frame size–as well as the derived MAC load–consists of the MAC header + payload. We continue to use this MAC load throughout this article. The packet durations are derived according to [1] for Zigbee and [8] for Wi-Fi without ACKs or RTS/CTS.

Equation (5) does not depend explicitly on the average Wi-Fi packet duration T_W. However, $T_{\overline{W}}$ can be written as

$$T_{\overline{W}} = \frac{1}{R} - T_W \quad (6)$$

where R is the average Wi-Fi packet rate (packets/s), T_W is the average Wi-Fi packet duration (s).

Table 2 Parameters used

	Zigbee	Wi-Fi
MAC frame size (bytes)	127, <u>100</u>, 50, 5	1278
Data rate	250 kbps	1 Mbps, 11 Mbps, <u>54 Mbps</u>
Packet-rate (packets/s)	25	10
MAC load (Kbps)	20	102.2
Packet duration (µs)	127 b: 4256	1 Mbps: 10416
	<u>100 b: 3392</u>	11 Mbps: 1121
	50 b: 1792	<u>54 Mbps: 212</u>
	5 b: 352	

Default values are underlined.

Equation (6) shows that the influence of T_W on $T_{\overline{W}}$ remains relatively low as long as $1/R$ remains large compared to T_W. The duration of the default Wi-Fi packet at 1 Mbps is 10.4 ms, so in order to limit the deviation in $T_{\overline{W}}$ to 10%, the packet rate should remain below 10 packets/s (=102.2 kbps). We can therefore expect that below this throughput the different Wi-Fi data rates will result in almost identical $\text{PER}_{Z,W}$. We will therefore use the 100 kbps point (the highlighted vertical line in Figure 5) as a first comparison point throughout this article. Furthermore, we assume that a Zigbee network can cope with up to 10% packet loss. Hence, we use the Wi-Fi load resulting in 10% $\text{PER}_{Z,W}$ (the highlighted horizontal line in Figure 5) as a second comparison point throughout this article. Figure 5 plots $\text{PER}_{Z,W}$ as a function of the Wi-Fi load for a Zigbee frame size of 100 bytes. We calculate a Zigbee $\text{PER}_{Z,W}$ of 3.74% at the 100 kbps point. The load at this point for 54 Mbps Wi-Fi data rate equals 279 kbps.

3.2. Sensitivity analysis

The total Zigbee packet duration T_Z can vary between 320 and 4256 µs. Figure 6 shows the difference in $\text{PER}_{Z,W}$ for 54 Mbps Wi-Fi. There is a factor 8 difference in $\text{PER}_{Z,W}$ between the largest and smallest Zigbee packets.

β depends on the CCA threshold and the received signal energy. Determining the exact value of β is out of scope for this article. However, we explore the sensitivity of $\text{PER}_{Z,W}$ to the value of β. Equation (7) shows the $\text{PSR}_{Z,W}$ in the case $\beta = 0$ while Figure 7 compares the case of $\beta = 1$ with that of $\beta = 0$. There is a reduction of $\text{PER}_{Z,W}$ with a factor 1.23 at the 100 kbps point, and the 10% $\text{PER}_{Z,W}$ point shifts from 279 to 324 kbps.

$$\text{PSR}_{Z,W,\beta=0} \approx e^{-\dfrac{T_{Z,\text{Rx2Tx}} + T_Z}{T_{\overline{W}}}} \quad (7)$$

Recall that during the analysis we assumed the Wi-Fi packets to be longer than $T_{Z,\text{Rx2Tx}} + \beta \times T_{Z,\text{CCA}}$ (:= T_{Z0}). However, Wi-Fi can transmit smaller packets. A collision will then only occur if the actual Wi-Fi packet
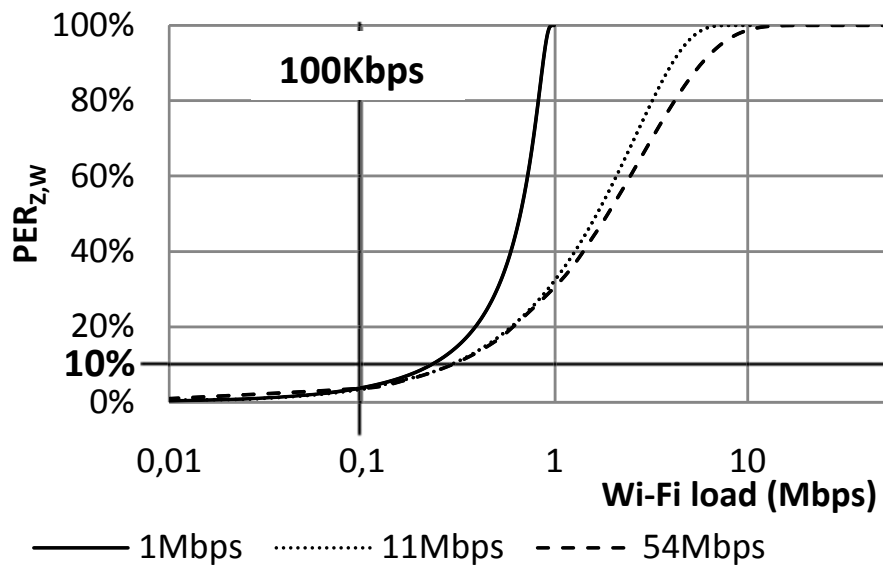
Figure 5 PER$_{Z,W}$ as a function of the Wi-Fi load for different Wi-Fi physical data rates.

transmission starts less than the duration of the Wi-Fi packet before the actual Zigbee packet starts. This change has the effect of replacing the term $T_{Z,Rx2Tx} + \beta \times T_{Z,CCA}$ in (5) with the actual duration of the Wi-Fi packet T_W:

$$\text{PSR}_{Z,W,T_W < T_{Z0}} \approx e^{\frac{T_Z + T_W}{T_W}} \tag{8}$$

The largest deviation to the base model is caused with the smallest Wi-Fi packets possible (28 μs). This possibility is also visualized in Figure 7. There is a factor 1.8 difference for 100 kbps Wi-Fi, and the 10% PER$_{Z,W}$ point shifts from 279 to 486 kbps.

3.3. Experimental model verification

We now turn to validate our model in practice. The experiments are conducted in the Wireless lab of the IBBT iLab.t technology center [9]. iLab.t has an RF-shielded environment of 4 shielded Qosmotec boxes, in which Zigbee and Wi-Fi devices are connected by coax cables. It can achieve full mesh connectivity between all four boxes through the use of a PC controlled attenuator. Hence, no external interference is received, and the attenuation of each link can be set. Using this setup allows for real devices to communicate in a controlled environment. For our experiments, we use three nodes with attenuation between them set as shown in Figure 8. The Wi-Fi transmitter broadcasts at 18 dBm, the Zigbee
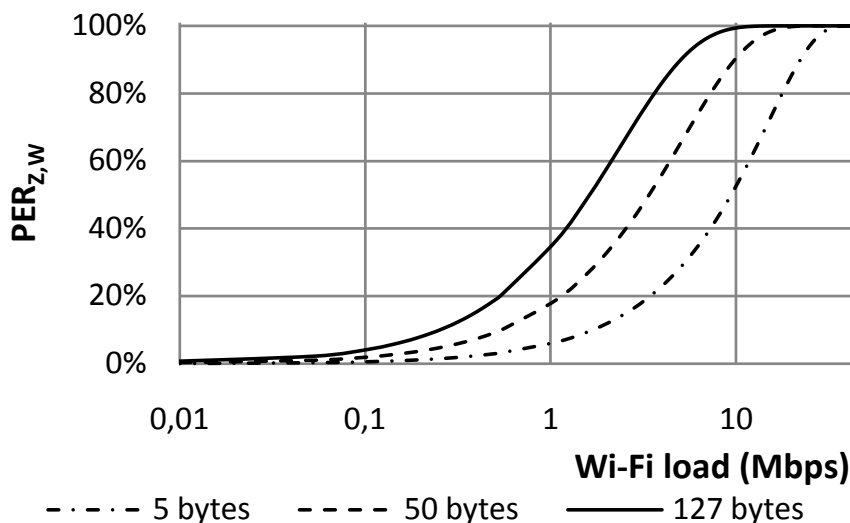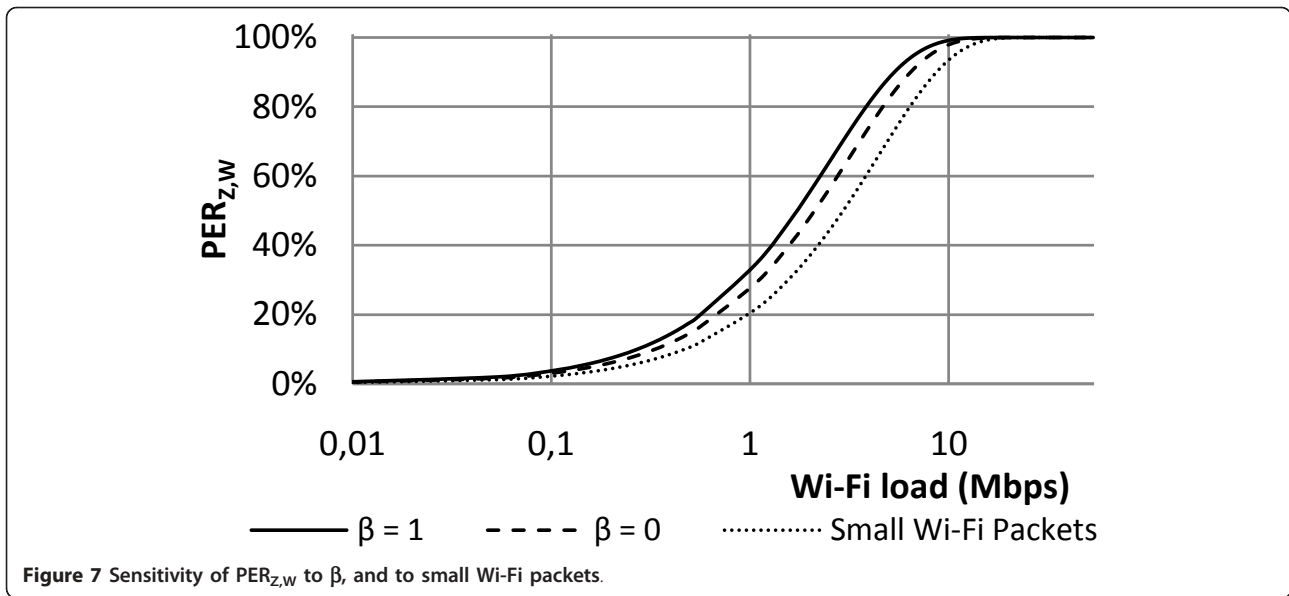

Figure 6 Sensitivity of PER$_{Z,W}$ to Zigbee packet size.

Figure 7 Sensitivity of PER$_{Z,W}$ to β, and to small Wi-Fi packets.

transmitter broadcasts at 0 dBm, and PER$_Z$ is measured at the Zigbee receiver. These settings result in SNR at the Zigbee receiver of about 25 dB, and SIR of -22.4 dB. Hence, all Zigbee packets which do not collide with Wi-Fi are received correctly, and all packets that collide with Wi-Fi are lost.

All experiments are run with 100 bytes Zigbee packets and 1278 bytes Wi-Fi packets sent at bitrates of 1, 11, and 54 Mbps. All packets are transmitted with a fixed IPD.

Figure 9 displays a comparison between the measurements and the model for 54 Mbps Wi-Fi. The PER$_Z \approx$ PER$_{Z,W}$ measurements are within a margin of 13% from the model for loads lower than 1 Mbps. A maximum deviation of 34% is measured at 2 Mbps application load (a packet-rate of 200 packets/s), which is the point where $T_{\overline{W}}$ becomes smaller than T_Z and obviously no Zigbee packet can be sent.
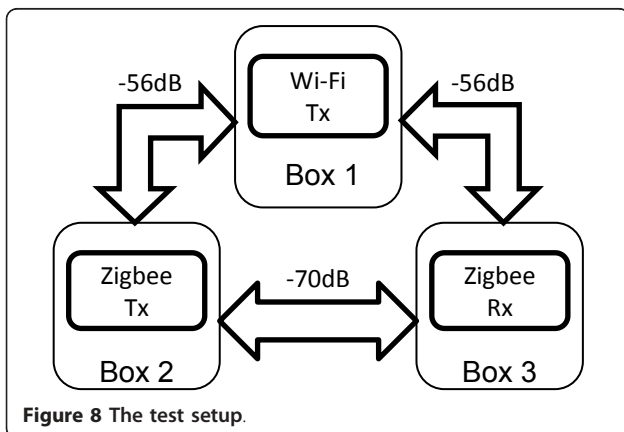
The PER$_{Z,W}$ for 11 Mbps Wi-Fi is depicted in Figure 10. The measurement outcome is similar to the 54 Mbps case as the Wi-Fi packet durations at 54 Mbps (=212 μs) and 11 Mbps (=1121 μs) are both relatively small compared to $1/R$ (=5000 μs). The maximum deviation to the model is also situated around 2 Mbps.

The comparison for 1 Mbps Wi-Fi is displayed in Figure 11. It shows that below the 0.1 Mbps load point the error remains below 13%. Beyond this point (0.1-0.4 Mbps), the model and the measured PER$_{Z,W}$ diverge. This is because the model assumes the Zigbee and Wi-Fi packet transmissions to be independent. However, due to Zigbee transmissions backing-off on the relatively long Wi-Fi packets (10 ms), they tend to start their transmissions close after a Wi-Fi transmission. This results in fewer collisions than expected because in the experiments the IPD for Wi-Fi is constant. Beyond the 0.4 Mbps point, $T_{\overline{W}}$ approaches T_Z, resulting in a fast increase in PER. Above 0.7 Mbps, $T_{\overline{W}}$ is smaller than T_Z, resulting in 100% packet loss.

As mentioned earlier, these tests are conducted with constant IPD for both Wi-Fi and Zigbee, and still the PER measurements are rather close to our calculations. The error remains below 25% in the region where the Zigbee network stays operational (PER$_{Z,W}$ < 10%). This indicates that the sensitivity of our model to the probability distribution of $T_{\overline{W}}$ and $T_{\overline{Z}}$ is rather low.

4. Deployment of sensing engine-based CACCA
4.1. Sensing engine characteristics
A spectral sensing engine is a fast and accurate device that measures spectral power density across a wide bandwidth (e.g., 174 MHz-6 GHz [10]). The internals of

Figure 8 The test setup.

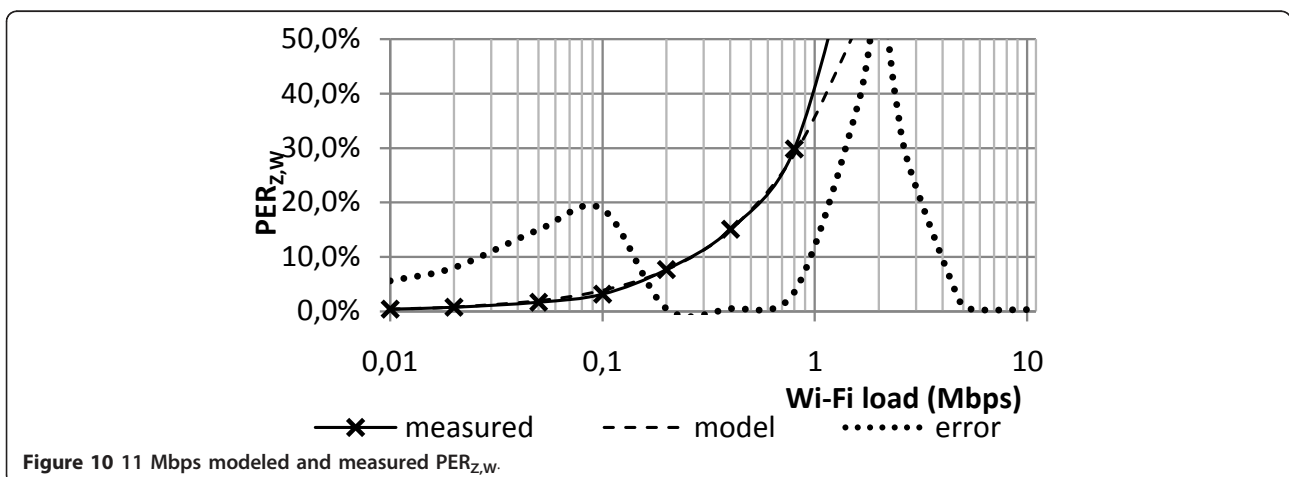**Figure 9** 54 Mbps modeled and measured $PER_{Z,W}$.
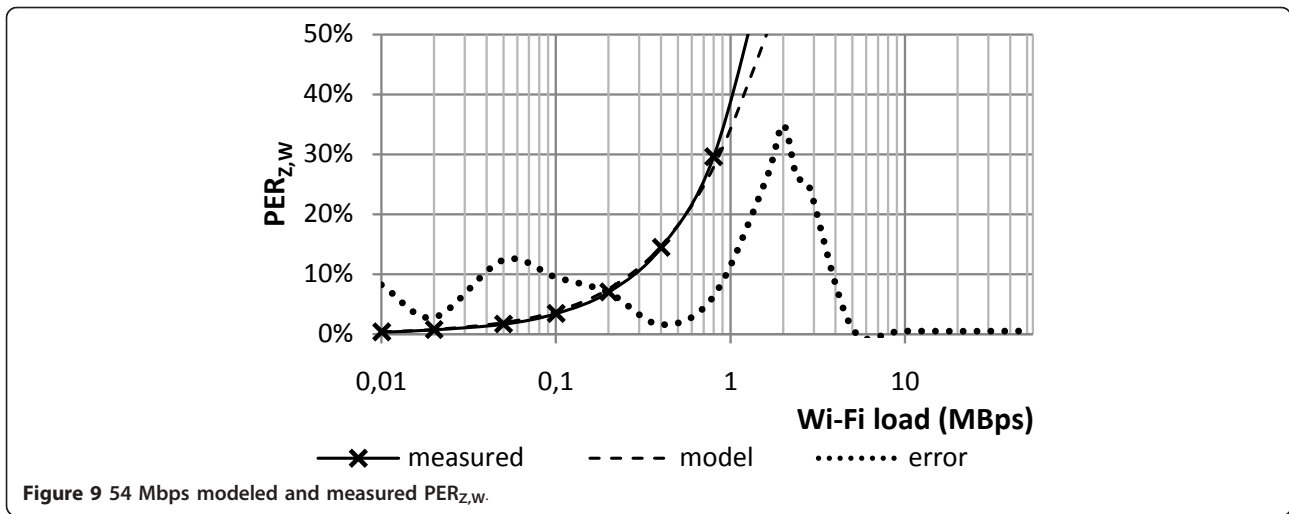
a sensing engine are detailed in [10,11]. A sensing engine is commonly used in space and frequency-based interference avoidance within a vertical spectrum sharing context [11-13]. However, in this article, we use a sensing engine as a CCA agent, which is a time domain function. Hence, in this study, we use a sensing engine to analyze a limited bandwidth within a very short timeframe. It is focused on achieving the highest detection reliability within a very small timeframe and therefore we assume that it can detect Zigbee reliably within the Wi-Fi CCA time. Accordingly, we assume that when deploying a sensing engine the Zigbee CCA time–$T_{ZS,CCA}$–can be reduced to 4 μs, which equals the Wi-Fi CCA time, and is 32 times faster than standard Zigbee. An overview of the resulting timings is given in Table 3.

The power consumption of a sensing engine detecting Wi-Fi is presented in [14], and equals 110 mW for the analog part, and 4 mW for the digital part to detect Wi-Fi, totaling to 114 mW. The sensing engine needs to be switched on during the 9 μs long CCA + Rx2Tx window,

resulting in a total energy consumption of 9 μs * 114 mW = 1.03 μJ. The minimal power consumption of a current CC2520 Zigbee Radio in transmit mode equals 45 mW, and the smallest Zigbee packet lasts 320 μs, resulting in a total minimal transmit energy of 12.8 μJ. Hence, the total impact on the power consumption of the sensing engine equals at most 8% per transmitted packet.

When deployed in a Zigbee device, the sensing engine creates a parallel receive chain to that of the Zigbee device, as depicted in Figure 12. Therefore, it can continue sensing the channel–and thus cancel the pending transmission–while the Zigbee device is switching towards transmit mode. Consequently, $T_{ZS,Rx2Tx}$ could in theory become negligible. To be realistic, we assume $T_{ZS,Rx2Tx}$ can be as short as $T_{W,Rx2Tx}$. The influence of implementing a sensing engine on Zigbee devices is visualized in Figure 13.

When deployed in a Wi-Fi device there is no need for a separate receive chain, as common Wi-Fi devices have
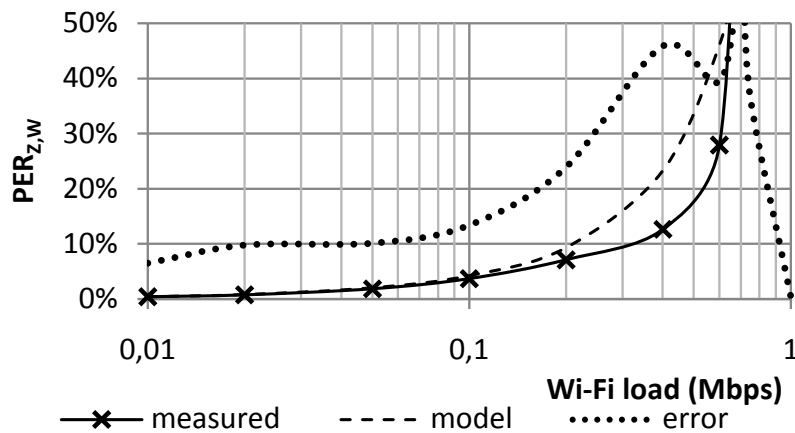

Figure 10 11 Mbps modeled and measured $PER_{Z,W}$.

Figure 11 1 Mbps modeled and measured PER$_{Z,W}$.

the necessary bandwidth and sensitivity. A Zigbee packet is detected by the sensing engine within a time-frame of 4 μs. The standard Wi-Fi CCA time is 4 μs, hence we assume that the implementation of a sensing engine in Wi-Fi devices will not change $T_{W,CCA}$ and $T_{W,Rx2Tx}$.

Only the digital part of a sensing engine will contribute to the energy consumption in a WiFi device. This 4 mW is only consumed during an 8-μs long timeframe, totaling to 32 nJ per transmission. An 18-dBm Wi-Fi transmission consumes at least 63 mW, using a 100% efficient radio. The shortest packet lasts 24.5 μs [8], resulting in an energy consumption of 1.5 μJ. The sensing engine energy consumption will thus contribute to at most 2% of the energy consumption per packet transmitted at 18 dBm.

4.2. Case 1: Zigbee side CACCA

A standard Zigbee device can detect Wi-Fi transmissions; therefore, the only effect of introducing sensing engines to Zigbee devices is that the CCA time $T_{Z,CCA}$ and the Rx2Tx transition time $T_{Z,Rx2Tx}$ are reduced to $T_{ZS,CCA}$ and $T_{ZS,Rx2Tx}$–the resulting PSR$_{ZS,W}$ is shown in (9) and PER$_{ZS,W}$ is depicted in Figure 14.

$$\text{PSR}_{ZS,W} \approx e^{-\frac{\beta \times T_{ZS,CCA} + T_{ZS,Rx2Tx} + T_Z}{T_{\overline{W}}}} \qquad (9)$$

PER$_{ZS,W}$ calculated at the 100 kbps point equals 1.05%. In other words, the inclusion of the sensing

Table 3 Regular CCA versus sensing engine based CACCA timings

	Zigbee	Wi-Fi
T_{CCA} (μs)	128	4
T_{Rx2Tx} (μs)	192	5
$T_{S,CCA}$ (μs)	4	4
$T_{S,Rx2Tx}$ (μs)	5	5

engine results in a PER$_{Z,W}$ drop of 24%. The 10% PER$_{Z,W}$ point shifts from 279 to 324 kbps.

Figure 15 depicts PER$_{ZS,W}$ for different Zigbee packet sizes. Comparing these results with the no sensing engine results of Figure 6 reveals the very modest difference. It is only for very small packets that a significant difference becomes noticeable. In this case, PER$_{Z,W}$ at 100 kbps Wi-Fi reduces with a factor 1.9 while the 10% PER$_{Z,W}$ point shifts from 279 to 580 kbps.

4.3. Case 2: Wi-Fi side CACCA

In Section 3, we came to the conclusion that one of the major reasons for packet loss is the inability of Wi-Fi to detect Zigbee packets. Adding a sensing engine to the Wi-Fi devices will solve this problem. Figure 16 illustrates the possible collision scenarios between standard Zigbee and a sensing engine-enabled Wi-Fi.

Figure 16 shows two different collision possibilities. The first (case 3 in the diagram) occurs when a Wi-Fi
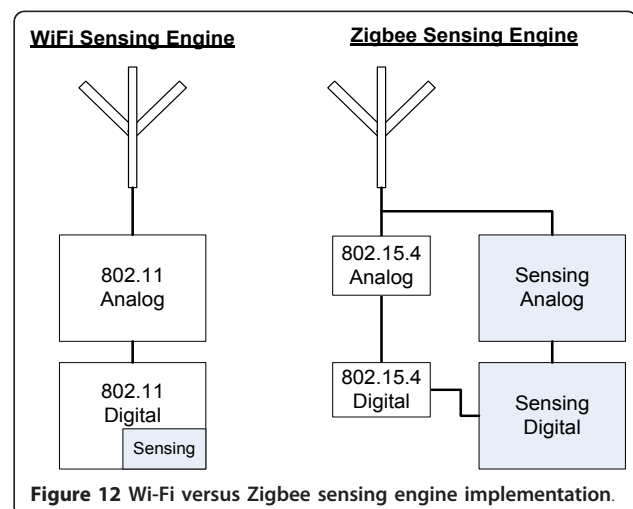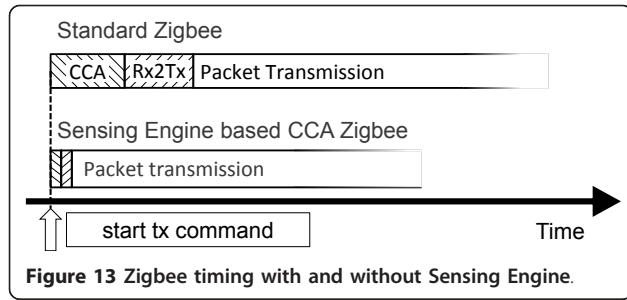

Figure 12 Wi-Fi versus Zigbee sensing engine implementation.

Figure 13 Zigbee timing with and without Sensing Engine.

transmission starts within the $\beta_1 \times T_{Z,CCA} + T_{Z,Rx2Tx}$ timeframe, in which Zigbee does not detect the Wi-Fi transmission. The second (case 4) is the reverse scenario where Wi-Fi does not detect the Zigbee transmission. Combining these two mutually exclusive events results in Equation (10), in which $T_{WS,CCA}$ and $T_{WS,Rx2Tx}$ equal, respectively, the Wi-Fi side sensing engine CCA and Rx2Tx times.

$$
\begin{aligned}
PER_{Z,W} = &\Pr(t_{\bar{W}} < \beta_1 \times T_{Z,CCA} + T_{Z,Rx2Tx}) \\
&+ \Pr(t_{\bar{Z}} < \beta_2 \times T_{WS,CCA} + T_{WS,Rx2Tx})
\end{aligned}
\tag{10}
$$

Approximating $t_{\overline{W}}$ and $t_{\overline{Z}}$ as exponentially distributed random variables results in

$$
PER_{Z,WS} \approx \left(1 - e^{-\frac{\beta_1 \times T_{Z,CCA} + T_{Z,Rx2Tx}}{T_{\overline{W}}}} \right) + \left(1 - e^{-\frac{\beta_2 \times T_{WS,CCA} + T_{WS,Rx2Tx}}{T_{\overline{Z}}}} \right)
\tag{11}
$$

Filling in the default values from Table 2 and assuming $\beta_1 = \beta_2 = 1$ gives

$$
PER_{Z,WS} \approx \left(1 - e^{-\frac{320 \times 10^{-6}}{T_{\overline{W}}}} \right) + \left(1 - e^{-\frac{9 \times 10^{-6}}{T_{\overline{Z}}}} \right)
\tag{12}
$$

In order to analyze the dependence on the Wi-Fi load, we assume $T_{\overline{Z}}$ sufficiently large, resulting in a negligible impact of the second part of (12). The first part of $PER_{Z,WS}$ is presented in Figure 17. We assess a $PER_{Z,WS}$ of 0.35% at the 100 kbps point, which is a reduction of 75% compared with $PER_{Z,W}$. The 10% $PER_{Z,W}$ point shifts from 279 to 3130 kbps.

Analyzing the dependence of $PER_{Z,WS}$ on the Zigbee load–the second part of formula (12)–can be achieved assuming $T_{\overline{W}}$ is sufficiently large. Figures 18 and 19 show the second part of $PER_{Z,WS}$ as a function of the Zigbee load. $PER_{Z,WS}$ stays below 1% as long as the Zigbee load remains below 200 kbps. The maximum $PER_{Z,WS}$ remains below 2.5% under all circumstances.

4.4. Case 3: Wi-Fi and Zigbee CACCA

In typical operating conditions, the Zigbee load is low, thus most of the contribution to $PER_{Z,W}$ comes from the first part of (12). This part highly depends on the Zigbee CCA + Rx2Tx time; therefore, it makes sense to also examine the effect of implementing the sensing engine on both Zigbee and Wi-Fi.

The model is identical in form with the model of case 2. The difference is found in $T_{Z,CCA}$ and $T_{Z,Rx2Tx}$ which are reduced to $T_{ZS,CCA}$ and $T_{ZS,Rx2Tx}$, respectively. Equation (13) shows the model incorporating Zigbee and Wi-Fi sensing.
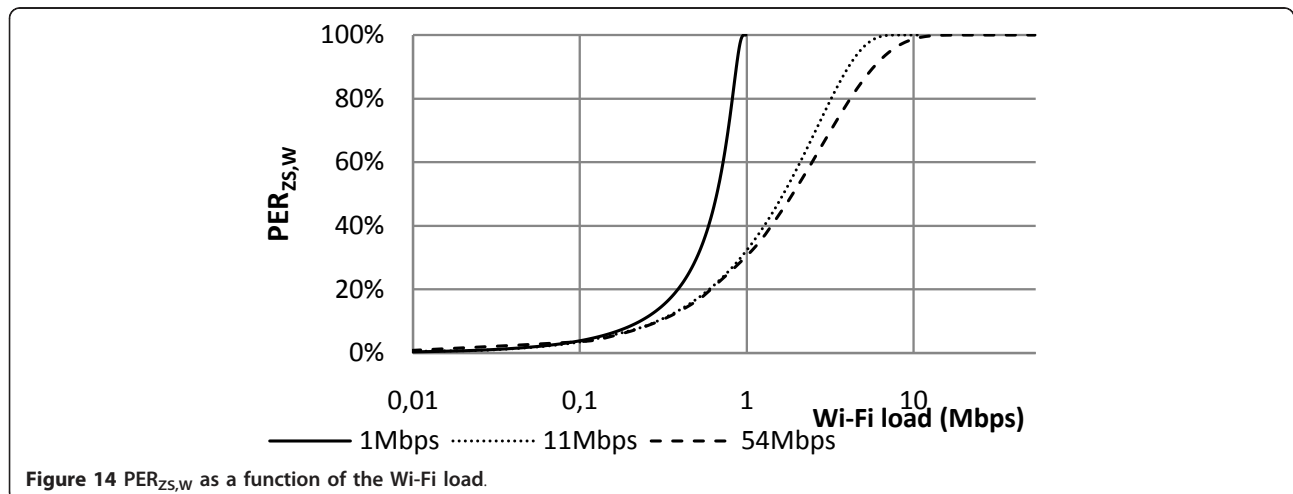

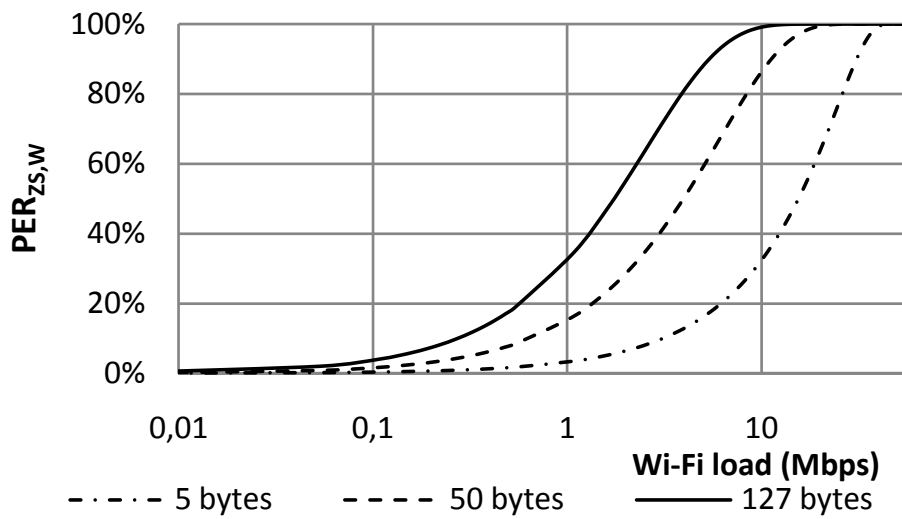**Figure 14 PER$_{ZS,W}$ as a function of the Wi-Fi load**.

Figure 15 Sensitivity of PER$_{ZS,w}$ to Zigbee packet size.

$$PER_{ZS,WS} \approx \left(1 - e^{-\frac{\beta_1 \times T_{ZS,CCA} + T_{ZS,Rx2Tx}}{T_{\overline{W}}}}\right) + \left(1 - e^{-\frac{\beta_2 \times T_{WS,CCA} + T_{WS,Rx2Tx}}{T_{\overline{Z}}}}\right) \quad (13)$$

Filling in the values gives us

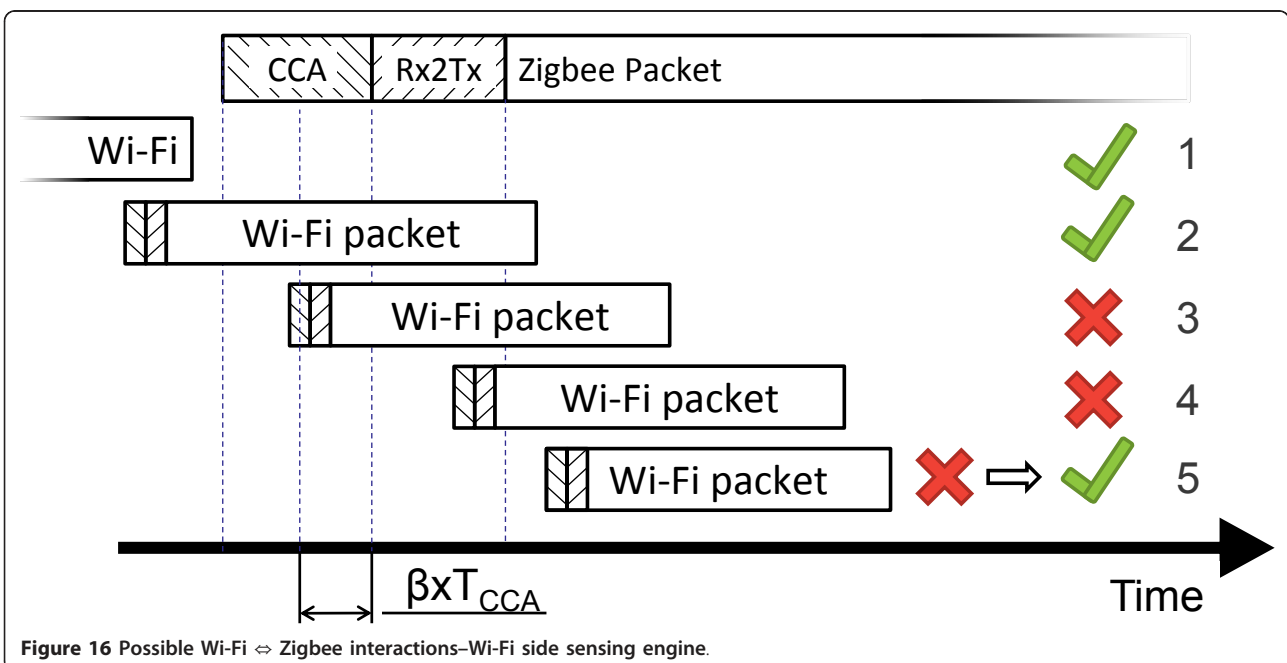$$PER_{Z,W} \approx \left(1 - e^{-\frac{9 \times 10^{-6}}{T_{\overline{W}}}}\right) + \left(1 - e^{-\frac{9 \times 10^{-6}}{T_{\overline{Z}}}}\right) \quad (14)$$

Figure 16 Possible Wi-Fi ⇔ Zigbee interactions–Wi-Fi side sensing engine.
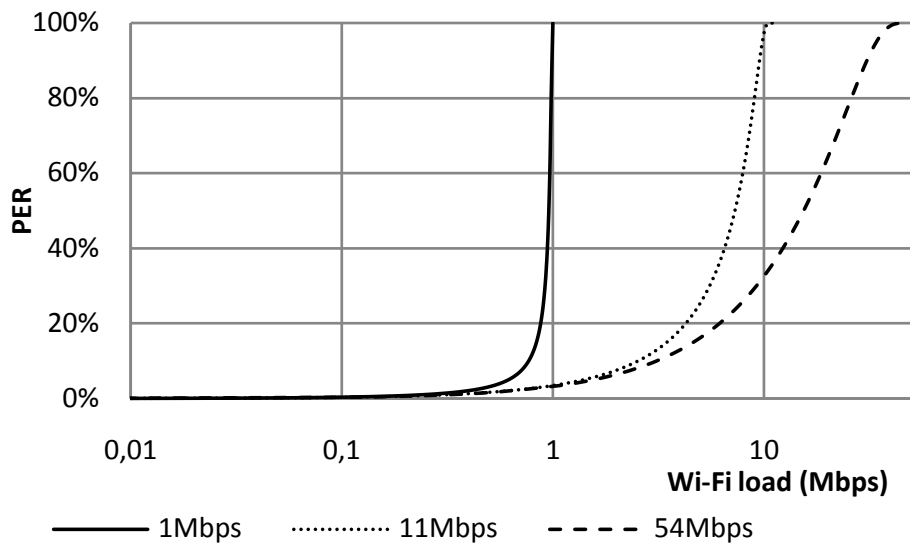
Figure 17 PER$_{Z,WS}$ as a function of the Wi-Fi load.

Again, we look at the two parts of the formula separately. The probability of Wi-Fi starting its transmission during the $T_{ZS,CCA} + T_{ZS,Rx2Tx}$ window is significantly lower compared to case 2, as this window now only lasts for 9 μs instead of 320 μs. The 100 kbps point has a calculated PER$_{ZS,WS}$ of 0.01%. In comparison with COTS hardware, this creates a drop in PER$_{Z,W}$ of 99.6%. The 10% PER$_{Z,W}$ point shifts from 279 kbps to 37 Mbps.

The dependence of PER$_{ZS,WS}$ on $T_{\overline{Z}}$ (second part of the formula) is identical to case 2.

4.5. Case comparisons

Case 1 handles the usage of the sensing engine on the Zigbee nodes. We conclude that PER$_{Z,W}$ is highly

dependent on T_Z and $T_{\overline{W}}$. The analysis shows reduction of 8-48% in PER$_{Z,W}$ (at 100 kbps Wi-Fi load), depending on the size of the Zigbee packets. The Wi-Fi load which leads to 10% Zigbee packet loss equals 324 kbps (for default size Zigbee packets of 100 bytes).

Case 2 handles the inclusion of the sensing engine in the Wi-Fi devices. The model shows that the dependence on $T_{\overline{W}}$ is reduced, while the dependence on the Zigbee packet size is almost completely removed. This case reduces PER$_{Z,W}$ at 100 kbps Wi-Fi load by 75% while the Wi-Fi load which leads to 10% Zigbee packet loss becomes 3130 kbps.

Case 3 considers the implementation of the sensing engine on both Zigbee and Wi-Fi nodes. This case has the lowest dependence on $T_{\overline{W}}$. It reduces PER$_{Z,W}$ at
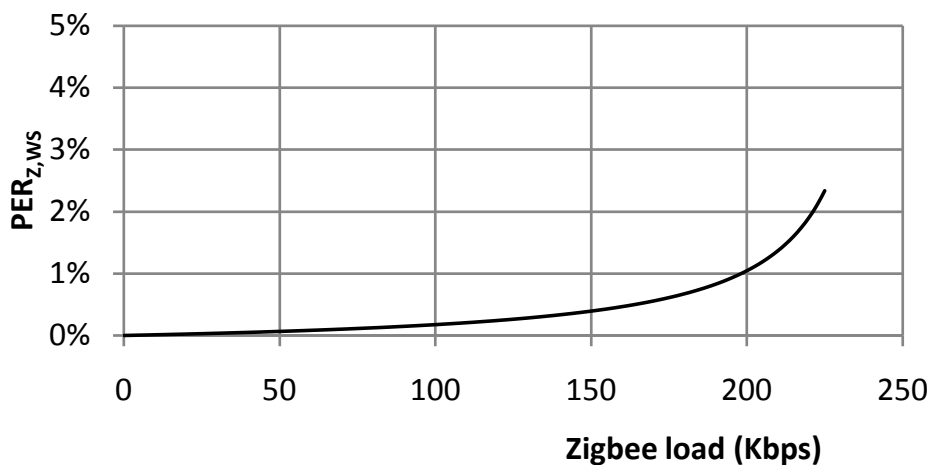

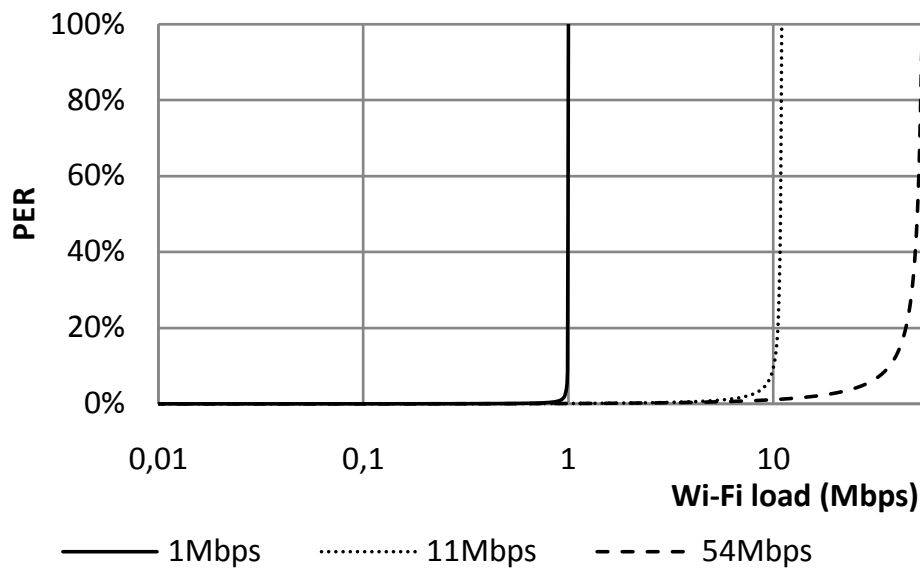**Figure 18 PER$_{Z,WS}$ as a function of the Zigbee load**.

Figure 19 PER$_{ZS,WS}$ as a function of the Wi-Fi load.

100 kbps Wi-Fi load by 99.6%, and achieves a Wi-Fi load resulting in a 10% Zigbee packet loss of 37 Mbps.

Figure 20 compares all the cases while Figure 21 zooms in on the–horizontal–10% Zigbee PER line, and the–vertical–100 kbps Wi-Fi load line. In addition, Table 4 summarizes all the cases and their dependencies on packet lengths (T_Z and T_W) and IPDs ($T_{\overline{Z}}$ and $T_{\overline{W}}$).

5. Future study
We instantiated the CACCA analysis within a Zigbee ⇔ Wi-Fi context. However, similar analysis can be done in other combinations of technologies, as well as identical technologies that operate in partially overlapping bands (e.g., IEEE 802.11bgn @ 2.4 GHz).

Another aspect we did not consider is the impact the sensing engine has on the Wi-Fi side. It does not only reduce PER$_{W,Z}$–which is a positive effect–but it also reduces the throughput of Wi-Fi–which is a negative effect. As such this remains an open issue.

This article only considers Wi-Fi broadcast traffic, without acknowledges or request to send/clear to send. An elaboration on their impact remains for future study.

A final direction for future study is to study the combination of the time domain collision avoidance, together
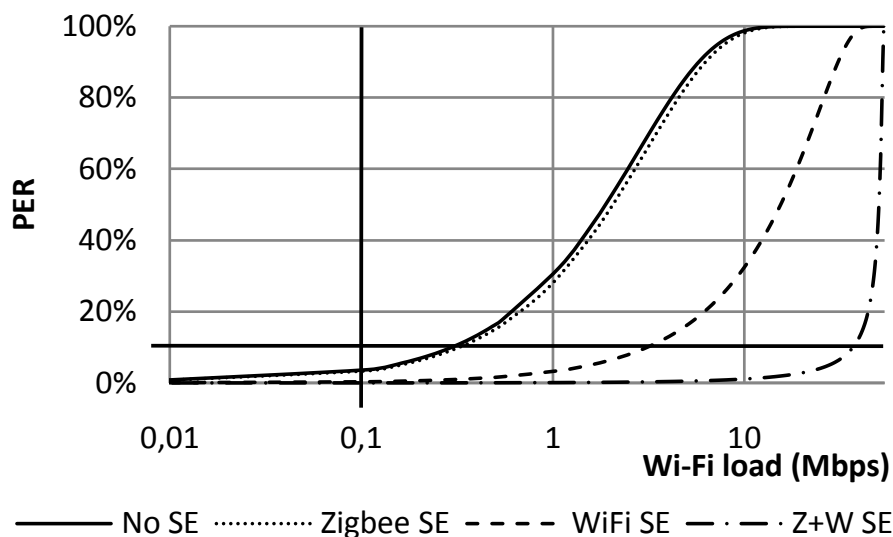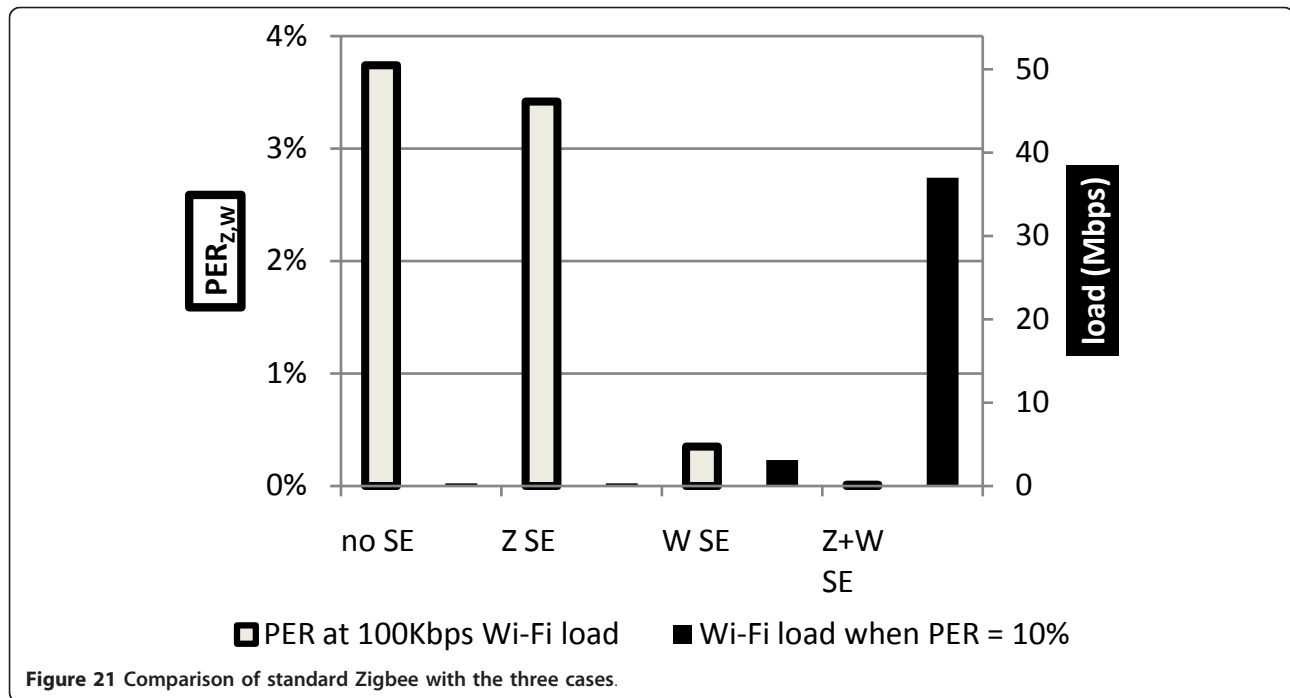

Figure 20 Comparison of PER$_{Z,w}$, PER$_{ZS,w}$, PER$_{Z,ws}$ and PER$_{ZS,ws}$.

Figure 21 Comparison of standard Zigbee with the three cases.

with frequency and/or space domain collision avoidance. This will exploit the possible benefits of a spectrum sensing engine to its fullest.

6. Conclusion

As more and more wireless technologies emerge, more of these technologies have to coexist with one another. One of the major open Wi-Fi ⇔ Zigbee coexistence issues is a model for cross-technology packet collisions. We propose a new analytical model for Zigbee packet loss due to collisions with Wi-Fi packets, analyze it theoretically and validate it experimentally. Out of this model, we conclude that the major cause of Zigbee packet loss is the inability of Wi-Fi to detect Zigbee transmissions.

In order to solve this problem, we propose the CACCA concept. CACCA enables Wi-Fi to detect Zigbee, and can be implemented through a sensing engine. There are three different deployment alternatives, namely, only Zigbee side deployment, only Wi-Fi side deployment, and Zigbee as well as Wi-Fi deployment. Deploying CACCA only on

Zigbee results in 24% packet loss reduction, deploying it on Wi-Fi results in 75% packet loss reduction while deploying it on both sides reduces Zigbee packet loss by 99.6%. The maximum allowable Wi-Fi load in order to have less than 10% Zigbee packet loss rises from 279 kbps in the regular CCA case to 324 kbps in the Zigbee only deployment alternative, 3.1 Mbps in the Wi-Fi only deployment alternative and 37 Mbps when deploying it on both sides. The added energy consumption of a sensing engine-based CACCA deployment equals to less than 8% per packet transmitted on the Zigbee side, and less than 2% on the Wi-Fi side.

We can conclude that the deployment of CACCA achieves substantial reduction of the Zigbee incurred packet loss, without needing any additional information exchange (and the incurred overhead), nor having a severe impact on the energy consumption. It can inherently cope with dynamic environments, and is backwards compatible with the IEEE 802.15.4 and IEEE 802.11 standards. Consequently, implementing CACCA increases the reliability of

Table 4 Comparison of regular CCA with the three CACCA deployment alternatives

	Regular CCA	Zigbee CACCA	Wi-Fi CACCA	Zigbee + Wi-Fi CACCA
$PER_{Z,W}$ @ 100 kbps (%)	3.74	3.42	0.35	0.01
Wi-Fi load @ 10% $PER_{Z,W}$ (Kbps)	297	324	3130	37000
$PER_{Z,W}$ dependence on				
T_Z	High	High	Low	Low
$T_{\overline{Z}}$	None	None	Low	Low
T_W	None	None	Low	Low
$T_{\overline{W}}$	High	High	Med.	Low

Zigbee while coexisting with Wi-Fi to an unprecedented level, without losing backwards compatibility with existing technologies.

As a final remark, we believe that while in the short-term CACCAmight be seen as a quick-fix for IEEE 802.11bgn ⇔ IEEE 802.15.4 coexistence, it can easily be extended to allow coexistence beyond current state-of-the-art technologies.

Author details
[1]Department of Information Technology - IBCN, Ghent University - IBBT, Gaston Crommenlaan 8 (Bus 201), 9050 Gent, Belgium [2]IMEC, Kapeldreef 75, 3001 Leuven, Belgium

Competing interests
The authors declare that they have no competing interests.

References
1. 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) (2006)
2. B Zhen, H-B Li, S Hara, R Kohno, Clear channel assessment in *integrated medical environments in EURASIP J. Wirel. Commun. Netw. vol. 2008*. (2008) doi:10.1155/2008/821756
3. W Yuan, X Wang, A coexistence model of IEEE 802.15.4 and IEEE 802.11b/g, in *Proceedings of 14th IEEE Symposium on Communications and Vehicular Technology in the Benelux* pp. 1–5 (2007). doi:10.1109/SCVT.2007.4436237
4. S Pollin, I Tan, B Hodge, C Chun, A Bahai, Harmful coexistence between 802.15.4 and 802.11: a measurement-based study, in *Proceedings of 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)* 1–6 (2008). doi:10.1109/CROWNCOM.2008.4562460
5. G Thonet, P Allard-Jacquin, P Colle, Zigbee-Wi-Fi coexistence white paper and test report. http://www.aduratech.com/pdf/ZigBee-WiFi-Coexistence.pdf. Accessed 15 Feb. 2012
6. L Tytgat, M Barrie, V Gonçalves, O Yaron, I Moerman, P Demeester, S Pollin, P Ballon, S Delaere, Techno-economical viability of cognitive solutions for a factory scenario, in *Proceedings of 2011 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, (Aachen, Germany, 2011), pp. 254–264
7. M Barrie, L Tytgat, V Gonçalves, O Yaron, I Moerman, P Demeester, S Pollin, P Ballon, S Delaere, Techno-economic evaluation of cognitive radio in a factory scenario, in *NETWORKING 2011 WORKSHOPS, from Lecture Notes in Computer Science 2011, issue Lecture Notes in Computer Science 2011*, vol. 6827, ed. by V Casares-Giner et al (Springer, 2011), pp. 52–61. n. 2011
8. Y Xiao, J Rosdahl, Throughput and delay limits of IEEE 802.11. IEEE Commun Lett. **6**(8), 355–357 (2002)
9. IBBT, iLab.t, technology centre http://www.ibbt.be/en/develop-test/ilab-t, Accessed 15 Feb. 2012
10. IMEC vzw, IMEC sensing engine development, http://www.imec.be/ScientificReport/SR2008/HTML/1225000.html, Accessed 15 Feb. 2012
11. AW Min, K Kim, K Shin, Robust cooperative sensing via state estimation in cognitive radio networks, in *2011 IEEE international symposium on Dynamic Spectrum Access Networks (DySPAN)*, (Aachen, Germany, 2011), pp. 185–196
12. R Balamuthi, H Joshi, C Nguyen., et al, A TV white space spectrum sensing prototype, in *2011 IEEE international symposium on Dynamic Spectrum Access Networks (DySPAN)*, (Aachen, Germany, 2011), pp. 297–307
13. P Van Wesemael, S Pollin, E Lopez, A Dejonghe, Performance evaluation of sensing solutions for LTE and DVB-T, in *2011 IEEE international symposium on Dynamic Spectrum Access Networks (DySPAN)*, (Aachen, Germany, 2011), pp. 531–537
14. S Pollin, L Hollevoet, F Naessens, P Van Wesemael, A Dejonghe, L Van der Perre, Versatile sensing for mobile devices: cost, performance and hardware prototypes, in *Proceedings of the 3rd ACM workshop on Cognitive radio networks (CoRoNet'11) 2011*, (ACM, New York, 2011), pp. 19–24. doi:10.1145/2030678.2030684
15. M Meekers, S Devitt, L Wu, Morgan stanley internet trends 04/12/2010 (Morgan Stanley Research 2010). http://www.morganstanley.com/institutional/techresearch/pdfs/Internet_Trends_041210.pdf. Accessed 15 Feb. 2012
16. 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