Minimising the effect of WiFi interference in 802.15.4 wireless sensor networks

Răzvan Musăloiu-E. * and Andreas Terzis

Department of Computer Science,
Johns Hopkins University, Baltimore, Maryland, USA
E-mail: razvanm@cs.jhu.edu
E-mail: terzis@cs.jhu.edu
*Corresponding author

Abstract: Interference from colocated networks operating over the same frequency range, becomes an increasingly severe problem as the number of networks overlapping geographically increases. Our experiments show that such interference is indeed a major problem, causing up to 58% packet loss to a multihop 802.15.4 sensor network competing for radio spectrum with a WiFi network. We present interference estimators that can be efficiently implemented on resource constrained motes using off-the-shelf radios and outline distributed algorithms that use these estimators to dynamically switch frequencies as interference is detected. Lastly, we evaluate the proposed algorithms in the context of a real-life application that downloads large amounts of data over multihop network paths. Our results show that the proposed approach successfully detects interference from competing WiFi channels and selects non-overlapping 802.15.4 channels. As a result, the proposed solution reduces end-to-end loss rate from 22%–58% to $< 1\%$.

Keywords: Wireless Sensor Networks; WSNs; radio interference; 802.15.4; WiFi.

Reference to this paper should be made as follows: Musăloiu-E., R. and Terzis, A. (2008) 'Minimising the effect of WiFi interference in 802.15.4 wireless sensor networks', *Int. J. Sensor Networks*, Vol. 3, No. 1, pp.43–54.

Biographical notes: Răzvan Musăloiu-E. received a BS and an MS in Computer Science from the University 'Politehnica' of Bucharest in 2003 and 2004, respectively. Currently, he is pursuing his PhD in Computer Science at Johns Hopkins University and he is a Member of the Hopkins InterNetworking Research (HiNRG) Group.

Andreas Terzis is an Assistant Professor in the Department of Computer Science at Johns Hopkins University. He joined the faculty in January 2003. Before joining JHU, he received his PhD in Computer Science from UCLA in 2000. He heads the Hopkins InterNetworking Research (HiNRG) Group and his main research interest is in the area of Wireless Sensor Networks. Terzis is a recipient of the NSF's prestigious CAREER award for his work in software tools for designing and deploying sensor networks.

1 Introduction

The vagaries of the wireless medium affect all Wireless Sensor Networks (WSNs). These vagaries cause intermittent network connectivity, packet loss and ultimately result in lower network throughput and increased energy expenditures. We classify the root causes behind these pathologies in two broad categories: *static* and *dynamic*. Static causes include factors such as site morphology, relative mote locations, as well as transceiver characteristics (e.g. transmission power and frequency, modulation scheme, etc.). The negative impact of these factors can be ameliorated, at least in planned deployments, through *site survey and planning tools* (Burns et al., 2006). These tools interject relay points and gateways in the original network topology to maximise its quality and consequently minimise packet loss and power consumption. There are however dynamic factors affecting network quality that cannot be accounted for at network design time. One of the primary ones is interference from geographically overlapping networks that use the same frequency range. As Zhou et al. (2006) have also argued, such interference will pose a growing problem as the number of WSNs deployed in overlapping geographical areas increases.

Our proposal is motivated by WSN applications that reliably extract large quantities of sensor data over multihop wireless paths. Examples of such applications include environmental monitoring (Musăloiu-E. et al., 2006), structural monitoring (Kim and Cullet, 2003; Xu et al., 2004) and condition-based maintenance (Adler et al., 2005). These applications are most sensitive to losses caused by interference since lost data must be retransmitted at considerable energy cost. Our measurements indicate that WSNs that use 802.15.4 radios experience packet loss between 3% and 58% when they compete with colocated WiFi networks, depending on the sending rate of the competing WiFi flow and the length of the WSN routing path. We show that interference from overlapping networks changes the Received Signal Strength Indicator (RSSI) which

can be used to build efficient interference estimators. Using a distributed coordination algorithm, WSN nodes participating in a multihop transfer decide which radio channel is the most 'quiet' and they switch to it prior to transmitting any sensor data.

This paper makes three contributions:

1 assess the impact of interference between WiFi networks and 802.15.4 networks used for large data downloads
2 propose mechanisms to dynamically detect and minimise the negative impact of interferences in sensor networks
3 evaluate the performance of these mechanisms through implementation and experimentation.

The rest of this paper is structured as follows: We begin by presenting related work in Section 2. In Section 3 we assess the impact of interference from colocated WiFi networks on the performance of 802.15.4 WSNs through measurements we conducted on outdoor as well as indoor deployments. We present a list of mechanisms to predict interference and evaluate their effectiveness as well as the complexity of their implementation in Section 4. In Section 5 we define the architecture of the heterogeneous sensor networks we consider and present algorithms to dynamically share the radio spectrum among wireless devices belonging to networks that reside in the same geographical area. Section 6 is devoted in the evaluation of the proposed mechanisms and we close in Section 7 with some concluding remarks.

2 Related work

The problem of interference and sharing of limited radio spectrum has been extensively studied in the wireless networking literature as well as the wireless communication literature in general. The traditional solution to this problem has been to license frequency bands to primary network users who are the only ones allowed to transmit in that frequency. This approach has been used in AM/FM radio, over-the-air TV broadcasts and even in cellular communications in which frequency bands are auctioned to wireless telephone carriers. While this approach removes the problem of interference, it results in low utilisation when the primary owner does not use the allocated spectrum frequently.

This disadvantage of static frequency allocations has led to the use of *shared* or *unlicensed* frequency bands that can be used by multiple networks at the same time. The 2.4 GHz band is a prime example of this paradigm, used by 802.11, 802.15.1 (Bluetooth) and 802.15.4 (Zigbee) data networks and even cordless telephones. The dominant technology used to reduce interference among multiple networks operating in the same frequency range employs a technique generally known as *Spread Spectrum*.

802.11 networks use the Direct Sequence Spread Spectrum (DSSS) mechanism in which the original bit stream is expanded into a larger sequence of chips according to a pseudo-random pattern and is subsequently spread out over a larger frequency range. The received signal is perceived as noise by all receivers other than the one which shares the same pseudo-random code with the transmitter. On the other hand, Bluetooth uses Frequency Hopping Spread Spectrum (FHSS), in which the sender rapidly switches among different frequencies while transmitting data, while the intended receiver switches the frequencies of its radio in concert.

Golmie et al. (2001) used analysis and PHY level simulations to show that WiFi can generate up to 15% packet loss to a colocated Bluetooth network. In our work, we show that the higher transmission power used in 802.11 networks can inundate Zigbee radios used in WSNs even though DSSS is used. Furthermore, while Bluetooth radios always change frequencies very rapidly to avoid random noise sources, in our approach motes change frequency only after detecting interference from an external network and continue using this frequency over extended periods of times (i.e. seconds) while a data transfer is active.

Multiple recent proposals have investigated the problem of channel allocation in the context of large scale 802.11 networks with multiple Access Points (APs). Halldorsson et al. (2004) studied WiFi channel assignments using the maximum graph colouring problem to identify Nash equilibria and provided a bound on the price of anarchy of these equilibria. Felegyhazi and Hubaux (2006) studied a similar setting and proposed algorithms for adjusting the transmission power used by the network's APs to minimise interference. Mishra et al. (2006) proposed a client-based algorithm for channel management in 802.11-based networks that leads to more efficient usage of the wireless spectrum. Demirhan et al. (2005) proposed a distributed solution for detecting interference from other WiFi networks and selecting frequencies in the context of WiFi mesh networks. Our work differs from these previous proposals in multiple ways. Firstly, while those proposals assume that all network nodes use the same transmission technology, our solution is able to detect interference *across networks* even when the receiver cannot decode the radio signal. Furthermore, while the goal of the previous work has been to maximise network utilisation and allocate spectrum resources fairly among users of the same network, our goal is to reduce interference among users of different networks.

In the context of sensor networks, an early study from Crossbow reported packet loss on Zigbee networks up to 15% caused by interference from an adjacent WiFi network (Crossbow Technology Inc., 2004). These results are lower than the 58% packet loss we observed in our tests. The underlying reason is that our tests were performed when sources in the WiFi network transmit at maximum speed and therefore represent the worst case scenario. On the other hand, both experiments show that interference from WiFi networks can create considerable packet loss to WSNs that use 802.15.4 transceivers. The Time Synchronised Mesh Protocol (Dust Networks, Inc., 2006) uses frequency hopping to limit the interference from competing RF sources. Since the frequency hop pattern is a pseudo-random sequence of all available channels a competing sender that is constantly sending at a particular frequency will still generate packet loss. Given that a WiFi channel overlaps with four 802.15.4 channels (cf Section 3.1), a WiFi source that sends packets constantly would still inflict packets loss equal to $4/16 = 25\%$ to a 802.15.4 WSN using TSMP. On the other

hand, our approach dynamically detects 'noisy' channels and completely avoids them, reducing the packet loss induced by RF sources such as one described above to zero. While other authors have speculated that RSSI could be used to detect and avoid interference (e.g. Woodings and Gerrior, 2006), our work provides quantitative results about the effect of interference from WiFi networks to Zigbee networks and evaluates the benefit from algorithms that detect interference and suggest alternate channels.

Perhaps closest to this work are recent spectrum sensing proposals presented in the context of Dynamic Spectrum Access (Challapali et al., 2005; Ganesan and Li, 2005; Ghasemi and Sousa, 2005). However, those proposals require specialised hardware (i.e. software radios) while our approach employs off-the-shelf, commercially available radios. Moreover, spectrum sensing techniques are primarily concerned about deploying multiple networks in a way that does not cause interference to the primary owner of the spectrum (e.g. a TV station). On the other hand, if we consider 802.11 as the primary spectrum owner, it is unlikely that low-power WSN radios will interfere with 802.11 APs. Instead, we address the problem of reducing losses in the WSN caused by the WiFi network.

Lastly, Srinivasan and Levis (2006) recently demonstrated that under certain conditions, there is strong correlation between RSSI measurements taken by a Zigbee receiver and the packet reception rate experienced by the same mote. The authors investigate RSSI during Zigbee transmissions and high values indicate good reception rate. On the other hand, we measure RSSI during periods of Zigbee silence and elevated RSSI values indicate potential WiFi activity. Whether the claims of the above paper still hold in the WiFi conditions we studied is beyond the scope of this paper.

3 Measurements

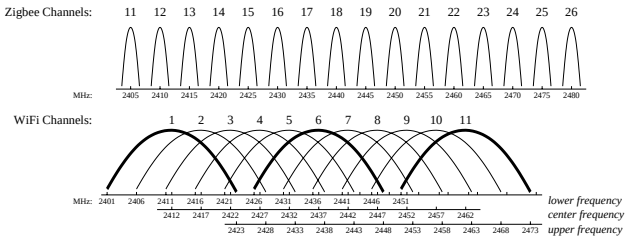
We begin our investigation by measuring the severity of the interference problem when sensor networks are deployed in environments also covered by other wireless networks. Specifically, we measure interference between 802.15.4 and WiFi networks as well as interference between two overlapping 802.15.4 networks.[1]

3.1 Background

Figure 1 depicts the frequencies that the 16 802.15.4 channels occupy as well as the corresponding WiFi channel frequencies. Each 802.15.4 channel is 3 MHz wide, centred around the frequency indicated in the figure. On the other hand, each WiFi channel is 22 MHz wide and in most cases it overlaps with 4 802.15.4 channels as well as with other 4–8 WiFi channels. The three non-overlapping WiFi channels (1, 6 and 11) are shown with bold lines in Figure 1. Zigbee channels 25 and 26 are special cases since they do not overlap with any WiFi channel. One could argue that the interference problem would be solved simply by using channels 25 and 26. Unfortunately, WiFi networks in Asia and Europe occupy two more channels on the higher end of the frequency band, overlapping with Zigbee channels 25 and 26.

The fact that both Zigbee and WiFi employ DSSS techniques to reduce crosstalk that could lead to the impression that inference is not an issue. This however is not true, since WiFi transmission power can be up to 100 mW (Cisco Systems Inc., 2006), 100 times higher than the maximum allowed 802.15.4 transmission power (Texas Instruments, 2006). Therefore, WiFi transmitters can create noise levels at an 802.15.4 receiver that overwhelm the interference resistance capabilities of DSSS. Furthermore, a WiFi channel completely covers an overlapping Zigbee channel so spreading the signal over the whole Zigbee channel does not avoid interference from the signal transmitted by the collocated WiFi radio.

Figure 1 Frequency ranges used by all the Zigbee and WiFi channels



3.2 Measurement methodology

The measurement methodology we designed explores the impact of different parameters on the level of interference. Since packet loss is the consequence of interference visible to applications, we adopt it as our evaluation metric. At the same time, we measure the increase in Received Signal Strength (RSSI) resulting from interference from competing networks. The reason for this will become apparent in Section 4, in which we discuss mechanisms to detect interference from colocated networks.

All our measurements use two networks: a *primary* network, which is always a Zigbee network using Tmote Sky motes (Polastre et al., 2005) and a *competing* network that is within reception range of the primary network. Each network consists of two nodes communicating with each other; in the case of WiFi, one of the two network nodes is an 802.11 AP. We use single hop networks to directly measure packet loss caused solely by interference rather than artifacts caused by network protocols associated with multihop network paths (i.e. routing and MAC protocols).

We measure the lossrate on the primary network using a stream of 2000 packets sent at a rate of 20 packets/sec. We also monitor the RSSI using a dedicated mote connected to a PC, that tunes its radio to each Zigbee channel and takes 100 RSSI samples at a rate of 20 samples/sec. We selected the 20 Hz rate to avoid overrunning the serial communication between the sensor node and the PC that processes the received RSSI samples. At the same time, one hundred RSSI samples are enough to collect meaningful statistics about the level of interference in an RF channel. A smaller sample size could be skewed by random interference (e.g. microwave ovens) that does not generate considerable packet loss to the primary network.

We modify this underlying measurement setup along the following axes:

Environment: we conducted measurements in two environments: inside an office building at an academic

institution and outdoors. We selected these two settings because they represent two environments in which sensor networks are commonly deployed.

Technology: we measured interference from a competing WiFi network as well as from another 802.15.4 network. The rationale was to evaluate not only the impact of transmission technology but also the relative transmission power on the packet loss measured in the primary network.

Frequency: from Figure 1, we expect that transmissions occurring in different WiFi channels will cause varying levels of interference on the primary 802.15.4 channel. At the same time, we expect that transmissions occurring at different 802.15.4 channels will not likely interfere given the band-gaps between consecutive channels. We tested the validity of these expectations by adjusting the channel of the primary network while keeping the competing channel constant.[2]

Transmission power: as mentioned above, the maximum transmission power of WiFi networks is 100 times larger than that of 802.15.4 networks and is thus likely to inundate the primary receiver. We adjusted the transmission power of the competing transmitter to evaluate at which point the primary receiver's radio can reliably decode the packet even in the presence of noise.

Transmission rate: finally, we adjusted the sending rate of the competing network flow. Intuitively, the higher the competing rate, the larger the probability that the level of noise in the primary receiver will be high, thus causing more packet losses. In all cases the competing sender uses maximum MTU packets and transmits at the maximum link rate (e.g. 54 Mbps for 802.11g).

3.3 Indoors measurements

Figure 2 depicts the office building used for the indoor measurements. The building is covered by an enterprise WiFi network whose APs are shown in the figure as dark circles. These APs operate on WiFi channels 3, 8 and 11. In addition to this network, at least two more WiFi networks (one of them using channel 4) used in other research projects have APs deployed over the same area. The lighter colour circles also shown in the figure are Tmote Connect mote gateways (Moteiv Corporation, 2006). Each gateway connects two Tmote Sky motes to the campus Ethernet. We use the two motes connected to gateway 117 as our primary network while the competing network comprised a laptop with an 802.11g card connected to the AP closest to gateway 117.

In out tests we did not attempt to change the WiFi network that already exists in the building in any way. As a result, there could be interfering traffic to/from or more of the WiFi APs that is not under our control. We tried to minimise such external traffic by conducting our experiments during periods of low network utilisation. Moreover, for experiments that measured the effect of WiFi interference on the primary network, we generated large amounts of WiFi traffic such that external traffic did not play a significant factor on the experiment's outcome.

3.3.1 Competing WiFi network

To test the interference from WiFi traffic we performed a series of UDP transfers on the WiFi network, using different sending rates. We used WiFi channel 8 (2436–2458 MHz) and varied the Zigbee channel used by the primary network from 10 to 26 (i.e. used all the channels).

The boxplots in Figure 3 illustrate the impact of the WiFi traffic on the following RSSI metrics: average, Inter-Quartile Range (IQR) and outlying RSSI values. The average in

Figure 2 Sensor Network deployed inside an office building. Mote gateway 117 is the first gateway from the right

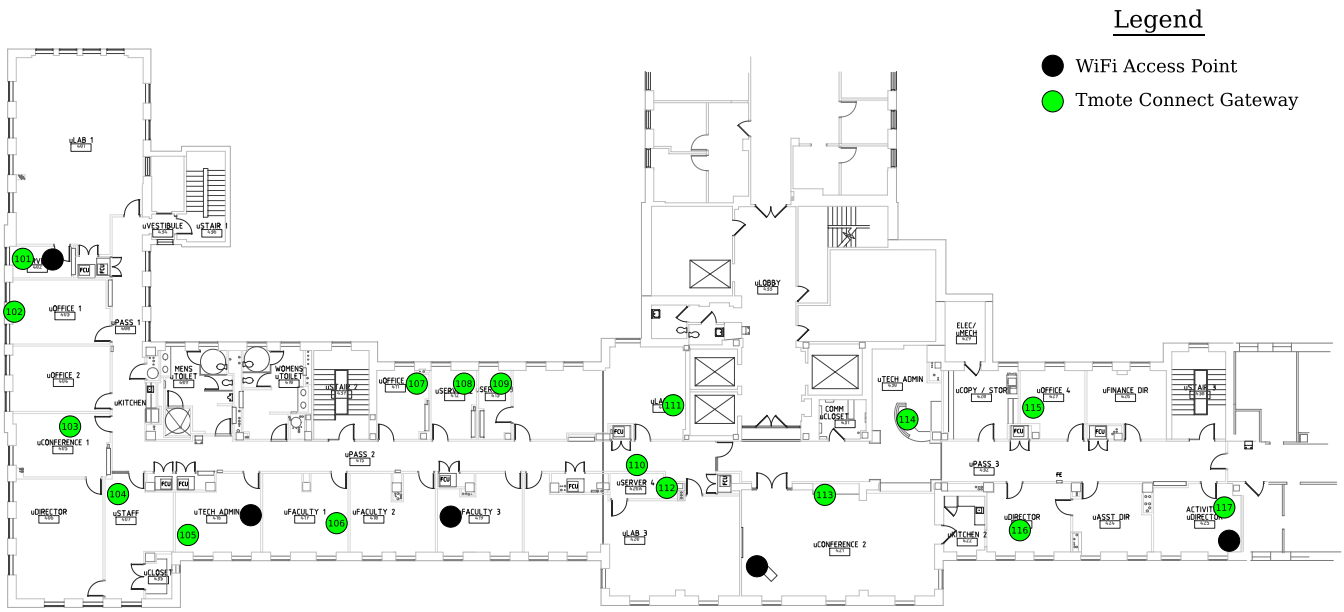
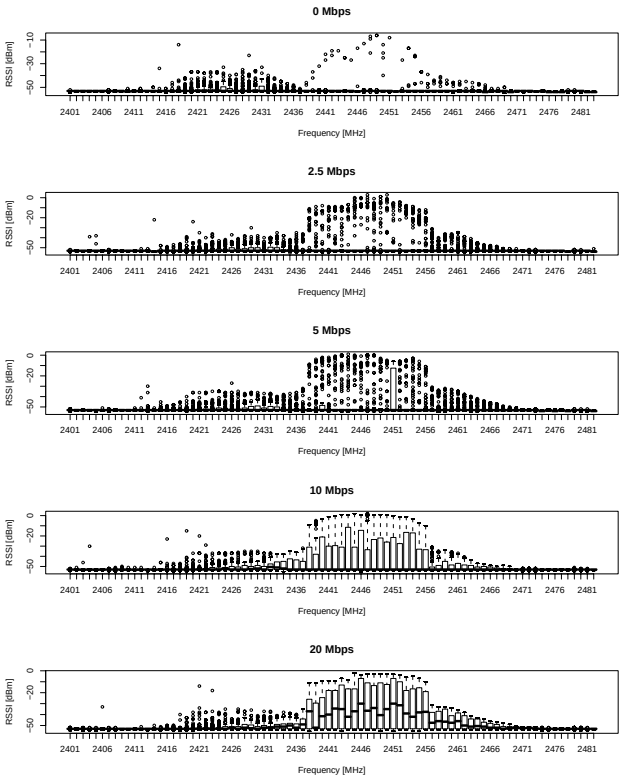


Figure 3 is depicted by a horizontal line inside the IQR, mostly visible in the graphs corresponding to the higher data rates. The IQR, which is the range covered by values lying between the first (25%) and third (75%) quartiles, is depicted by a vertical box for each frequency measured. Finally, we define outliers as measurements that are more than $1.5 \times$ IQR away from the closest quartile and depict them using small circles in Figure 3. The y-axis corresponds to the `RSSI_VAL` raw dBm readings provided by the CC2420 radio[3]. The top graph in Figure 3 corresponds to the measured RSSI when we do not initiate any transfers in the competing network. The remaining graphs correspond to cases in which the competing source actively injects traffic at different transmission rates.

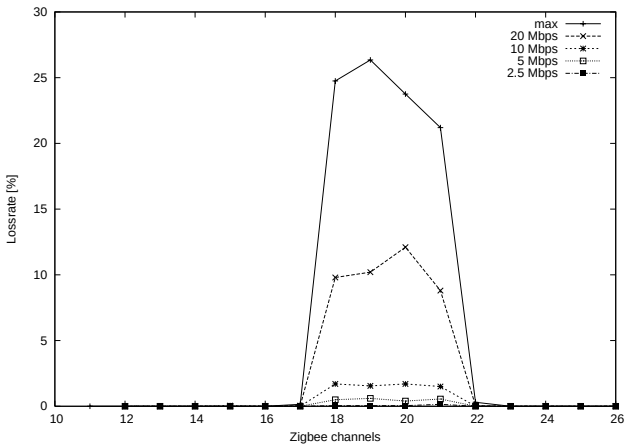
Figure 3 Measured RSSI on the primary 802.15.4 network as a function of transmissions with varying sending rate on the competing WiFi network. Note that no transmissions occur in the primary network



Firstly observation from these graphs is that two frequency ranges exhibit higher RSSI values. The rightmost range coincides with the frequency range of the 8th WiFi channel (used by the APs in the building's WiFi network). The other corresponds to WiFi channel 4, used by the APs of the research project mentioned in Section 3.3. It is also evident that while maximum recorded RSSI values increase across all data rates, the 10 Mbps transfer causes a significant change in IQR, while only the 20 Mbps competing transfer increases the average RSSI value significantly.

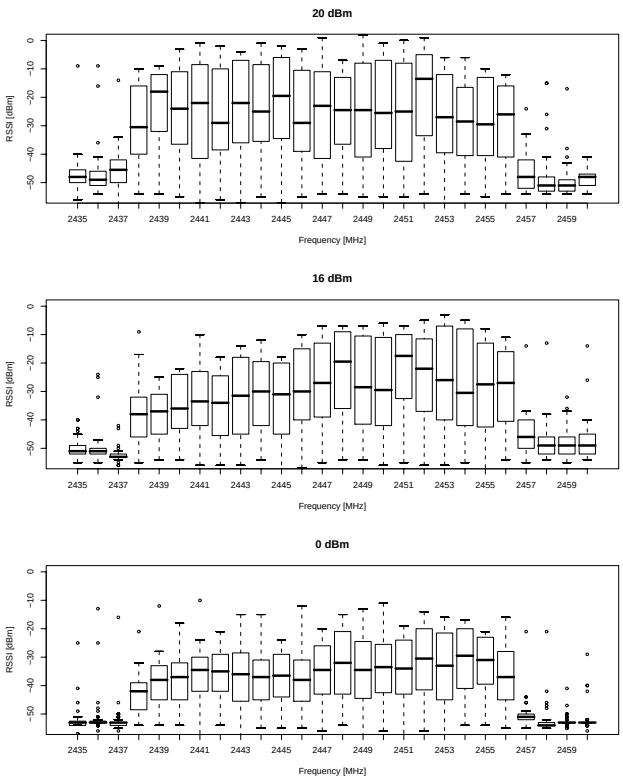
We investigated the impact of the WiFi interference measured in the previous experiment to the Zigbee channel by initiating periodic transmissions on the primary network. The loss rate test illustrated in Figure 4, reveals that only six Zigbee channels were affected, two of them very minimally (channels 17 and 22) while the center ones (18–21) show losses slightly higher than 20% in the worst case. Secondly result suggested by Figures 3 and 4 is that even though average RSSI values might not increase considerably, such as in the case of competing transmissions at 5 Mbps, the associated loss rate is not trivial ($\sim$ 5%). We will return to this point in Section 4, in which we discuss potential interference detection mechanisms.

Figure 4 Loss rate on the primary 802.15.4 channel as a function of transmission rate on the competing WiFi channel



Finally, Figure 5 reports the effect of WiFi transmission power on measured RSSI when the competing source transmits at full speed while we adjust the transmission power. In this test we focus only on the frequencies surrounding WiFi channel 8, while we adjust the transmission

Figure 5 Recorded RSSI on the primary 802.15.4 channel as a function of transmission power on the competing WiFi channel

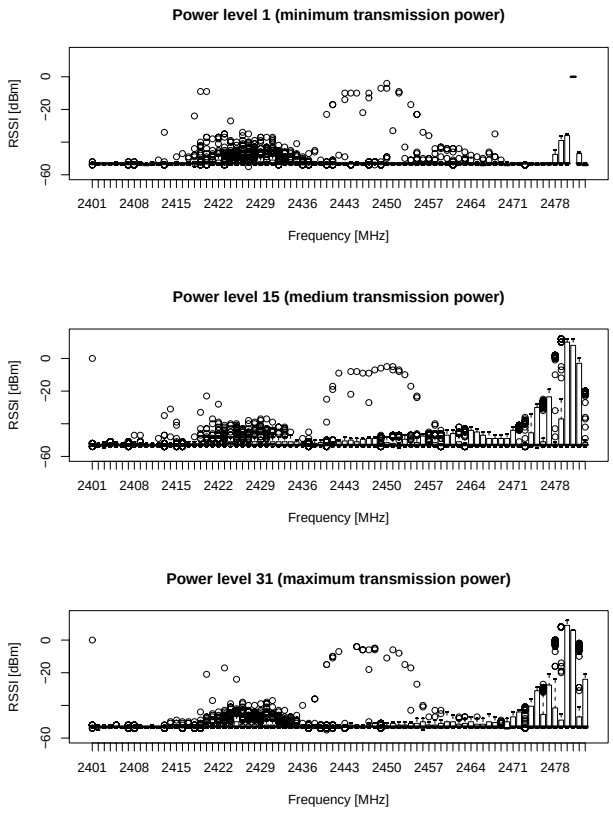


power of a laptop performing a bulk data transfer through the AP from 1 mW (0 dBm) up to 100 mW (20 dBm) (we did not adjust the transmission power of the AP). There is a almost linear relationship between increase in transmitted power and increase in received power recorded as higher RSSI values and therefore increased loss rates in the primary network.

3.3.2 Competing Zigbee network

In the second set of indoor experiments, we replace the competing WiFi network with a second pair of Tmote Sky motes operating on Zigbee channel 25 transmitting packets as fast as possible. We measured RSSI values for three different power levels in the competing network (minimum, intermediate and maximum). As Figure 6 illustrates, the competing network does not significantly raise the measured RSSI on the primary channel across all power settings.

Figure 6 RSSI values measured in the primary network, for different transmission power levels at the competing Zigbee network. The occasional high RSSI readings at frequencies between 2436 and 2457 MHz correspond to WiFi transmissions over channel generated by traffic not under our control



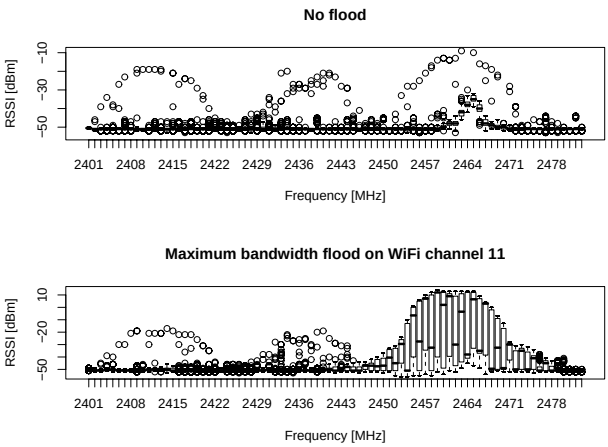
While we do not include the loss rate results for the sake of brevity, we note that they showed 1.5% packet loss in the primary network in the worst case. Given the low packet loss rate between two competing Zigbee networks, we focus on the more important problem of interference between WiFi and Zigbee networks for the rest of this paper.

3.4 Outdoors measurements

For the outdoors scenario we limit our measurements to RSSI measurements of the primary 802.15.4 network. The measurements were conducted at the site of a WSN deployed for environmental monitoring (we present details regarding this deployment in Section 5.1). What is important in this context is that the site also contains WiFi APs using channels 1, 6 and 11. The AP on channel 11 was used for the actual competing data transfer because it had the strongest signal.

Figure 7 presents the results of RSSI sampling in two cases: when we did not initiate a competing network flow and when we sent traffic at maximum speed using a laptop with a WiFi card. In both cases we can clearly see high RSSI values in frequencies related to WiFi channels 1, 6 and 11. Furthermore, the competing WiFi transfer we initiated significantly raised the RSSI in the Zigbee channels overlapping with WiFi channel 11. These results are encouraging in that they show high similarity between the two scenarios and therefore we expect to be able to apply the same approach to both indoor and outdoor environments.

Figure 7 Measured RSSI on the primary outdoors 802.15.4 network as a function of transmissions on the competing WiFi network



4 Design of interference estimators

The measurements presented in the previous section point to considerable correlation between high RSSI values and increased packet loss due to interference from competing networks. This correlation motivates the development of *interference estimators* that use RSSI measurements to predict interference and subsequently choose the optimal channel for future data transfers.

At a high level, the estimator applies an aggregation function to the collected RSSI measurements for each channel and ranks all channels by increasing level of interference. The channel with the lowest rank is then likely to have the lowest level of interference and should therefore be used for subsequent transmissions. At the same time, the ideal aggregation function should be easy to implement in resource constrained sensor nodes and should require as few measurements as possible. Lengthy measurements are unattractive because they are costly in terms of power (the

radio must stay on to take RSSI samples), require precious storage and postpone the actual download process.

Next, we present three candidate aggregation functions and evaluate their performance:

Cardinality: this function counts the number of unique RSSI values in the collected measurements. Channels with high cardinality get a high score (i.e. are classified as likely to be lossy). The intuition behind this choice is that channels with no interference are stable and thus will have a small number of RSSI values, while RSSI measurements from noisy channels will cover a larger range of values resulting in high cardinality scores. This argument is supported by the results presented in Figure 3, in which Zigbee channels far from the interfering WiFi channels 4 and 8 have consistently small range of RSSI readings.

Max and/or Mean RSSI value: channels with high levels of interference will record high maximum RSSI values as well as high average RSSI readings. An additional benefit of these estimators is that they have very low memory requirements since they can be updated each time a new sample arrives.

Threshold RSSI: this last function counts the number of RSSI measurements above a preconfigured threshold and assigns the highest score to the channel with the most measurements above that threshold. This threshold value represents the 'noise floor' and can be determined in advance by measuring the RSSI of an idle channel. The supporting argument for this metric comes again from Figure 3, in which channels with no competing traffic do not experience large RSSI values. We found experimentally that an appropriate threshold value for `RSSI_VAL` is -45 dBm. The chosen threshold value is related to the minimum receiver sensitivity. The value is published by Chipcon (Texas Instruments, 2006) and is -90 dBm which, after adjusting for `RSSI_OFFSET` offset, corresponds to exactly to a `RSSI_VAL` –45 dBm. Srinivasan and Levis (2006) have also identified experimentally that a similar value (-87 dBm) results in very low packet loss.

To evaluate the effectiveness of these heuristics we initiated a sequence of UDP floods on WiFi channel 8, while collecting RSSI samples on Zigbee channels 19 and 26 at a rate of 20 samples/sec. Each of the UDP floods lasted 10 sec and was separated by 20 sec of idle time from the next flood. Furthermore, packets from each flood were sent at different rates (54 Mbps, 20 Mbps, 10 Mbps, 5 Mbps, 2.5 Mbps and 1 Mbps) to measure the effect of packet inter-arrival times on the RSSI samples.

Figure 8 plots the raw RSSI time series, while Figure 9 presents the outcomes for each of the interference estimators, when run on these time series using a window of the latest 100 samples. We selected this sample size because it is large enough to collect meaningful statistics about the RF channel (i.e. it is not skewed by random noise spikes). As expected, Zigbee channel 19 shows considerable variation in recorded RSSI as a result of transfers in the competing WiFi channel, while Zigbee channel 26 is mostly quiet. From this figure we conclude that all heuristics other than maximum RSSI, can differentiate between periods of interference due to competing WiFi data transfers and silence in both Zigbee channels.

Figure 8 Sequence of raw RSSI measurements for Zigbee channels 19 and 26 when competing WiFi channel 8 is flooded by six consecutive UDP transfers, each with a different sending rate. Starting at $t = 20$ sec, each UDP transfer lasts 10 sec and is separated by 20 sec from the following transfer. UDP packets were sent at rates of 54, 20, 10, 5, 2.5 and 1 Mbps, respectively. The spikes during the idle periods are almost periodic and most likely correspond to WiFi AP beacons

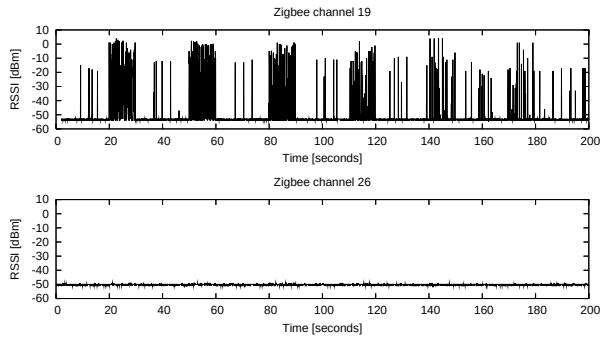
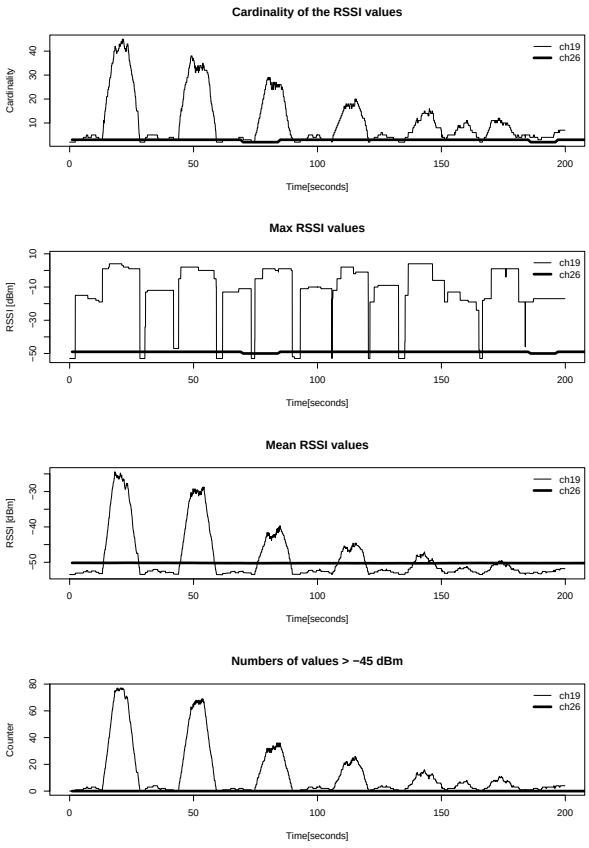


Figure 9 Results of the various interference estimator heuristics for the raw RSSI measurements presented in Figure 8. Notice that the value of the maximum RSSI estimator remains high even when the channel is quiescent



Next, we investigate whether the proposed functions can detect the Zigbee channels that overlap with the active WiFi AP(s), even if we do not initiate any data floods. To do so, we sequentially sampled the RSSI on all the Zigbee channels and applied each of the estimation heuristics on each of them. We expect to detect higher interference on the Zigbee channels that overlap with WiFi channels 3, 4, 8 and 11 given the environment in which the measurements were collected (cf. Section 3.3.) The results of this experiment are presented in Figure 10 and they show clearly that the Mean RSSI fails to distinguished between used and used channels (the range of values for Zigbee channels 18–26 is almost flat). The remaining two heuristics, cardinality and threshold, both detect WiFi activity (cf. Figure 1). It is however evident that the threshold estimator is more appealing due to the higher sensitivity it shows in the frequency range of WiFi channels 3 and 4.

Figure 10 Heuristics comparison for a sequential RSSI sampling of all the Zigbee channels. 2000 samples were collected from each channel. The sampling rate is 20 samples/sec

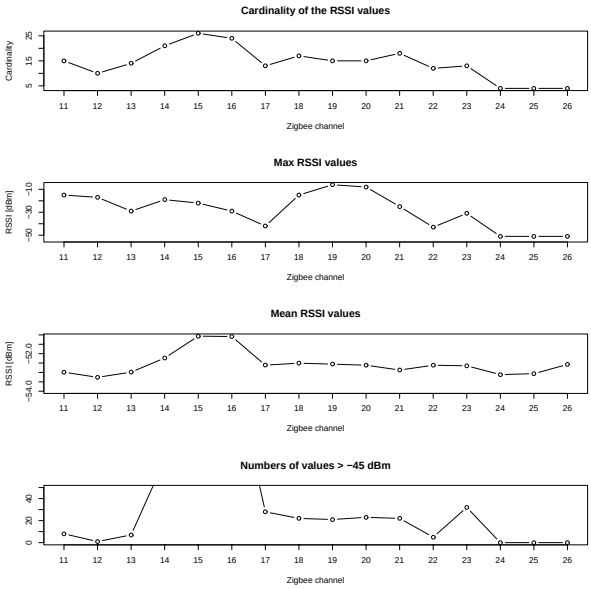


Figure 11 represents the performance of the threshold estimator when run on all the 16 Zigbee channels while WiFi transfers occur on channel 8 as described above. It is evident from this figure that the estimator is able to clearly detect interference in Zigbee channels 18–21 that directly overlap with the competing WiFi channel. Adjoining channels (16, 17, 22, 23) below and above the frequencies covered by the WiFi channel receive the next higher interference scores, while the outermost 6 channels (11–13, 24–26) have almost zero RSSI values above the threshold.

Finally, we perform a simple test to verify that the threshold estimator works well outdoors. We performed a single download at the maximum speed allowed by the network and the predictor output for Zigbee channels 18 and 22 is depicted in Figure 12. The predictor correctly differentiates between Zigbee channel 18 which is far from WiFi channel 11 and Zigbee channel 22 which is completely covered by the competing WiFi channel used in our transfer.

5 Mechanism design

So far, we have demonstrated that interference from competing networks can severely impact WSNs and presented efficient and effective estimators for detecting such interference. Next, we outline how interference detection and avoidance can be incorporated to WSN protocols.

5.1 Sample application

While the proposed solution applies to any 802.15.4 WSN that shares physical space with WiFi networks, we ground our design through an environmental monitoring application that has been deployed for over a year (Musăloiu-E. et al., 2006). The network is deployed in an urban forest next to a university campus and is within transmission range of the campus-wide WiFi network.

The purpose of our WSN is *soil monitoring*, in which motes periodically collect soil measurements including soil temperature and soil humidity, as well as ambient temperature and light. The key difference between this application and previous environmental monitoring networks (e.g. Mainwaring et al., 2002; Tolle et al., 2005) is that *all* collected measurements are reliably retrieved over multihop wireless paths at the network's base station using a Automatic Repeat Request (ARQ) transport protocol. This stringent reliability requirement is dictated by the domain scientists in charge of this project and the collaborative nature of the monitored site. Other applications that share the same requirements are structural monitoring (Xu et al., 2004), as well as sensor networks used in condition-based equipment maintenance (Adler et al., 2005).

5.2 Multihop data transfer

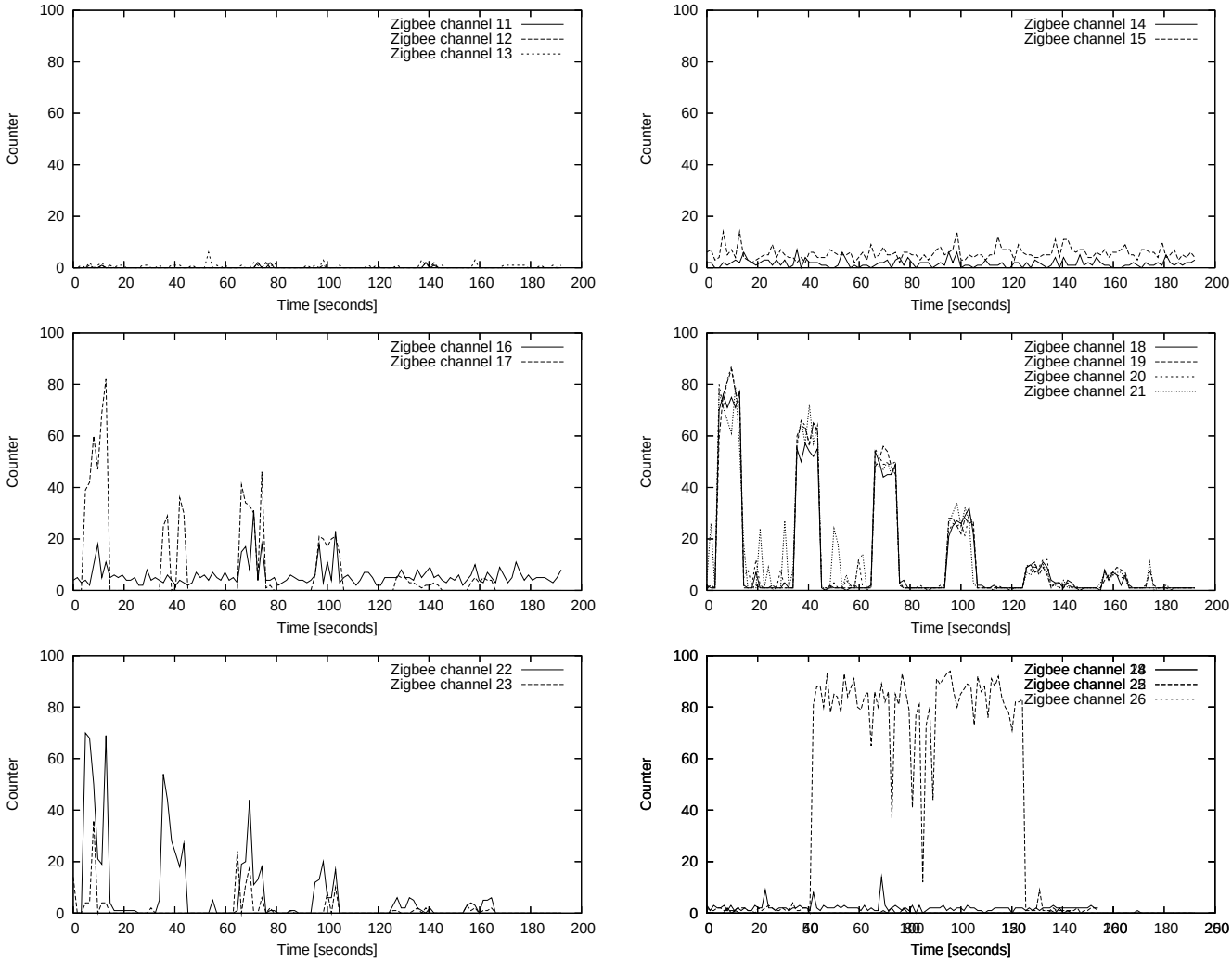
As we have already mentioned, collected measurements are extracted from the network over multihop paths comprising sensing nodes, one or more data relays and the base station connected to a PC through a serial connection.

The sensing node has two main states: sleeping, in which its radio is turned off and active, in which it sends periodic status messages describing its current state (i.e. number of samples collected in local flash and battery voltage). While in active state, the sensing node accepts download commands requesting for a specific region of its flash, transmitted at a certain rate. When the sensing node receives such a command, it pauses sending periodic reports and transmits all the requested data at the specified pace.

Data relay nodes act as simple packet switches forwarding packets towards the base station. We configure a Virtual Circuit (VC) for each sensing node describing the sequence of nodes that packets from that source should follow. Sources then add a VC identifier, called Data Link Connection Identifier (DLCI), to the packets they originate. When a data relay receives a packet, it looks up the incoming DLCI in its routing table and decides the next hop as well as the next DLCI that should be used.

To avoid collisions between copies of the same packet transmitted by different nodes on the path towards the gateway, the sensing node must inject packets at specific

Figure 11 Detailed plots of the performance of the threshold predictor across different channels



intervals. Figure 13 presents the timeline of such a data download transaction over a three hop network with two routers R_1 and R_2 separating the source from the base station. The process starts with sensing node S reading the next payload from flash, an action that takes T_r seconds. The source then transmits the packet over the radio channel. Given the short distance of all network links, propagation delay is negligible and therefore after T_{tx} seconds the packet is fully delivered to R_1.[4] After a pause of T_p seconds, R_1 forwards the received packet to R_2, however given the broadcast nature of the wireless interface, S also receives a copy of the original packet. R_2 in turn forwards the packet to the base station. The base station finally copies the packet to a PC connected to the internet through a serial connection, finishing the packet transfer after T_s seconds.

As Figure 13 suggests, in order to avoid any collisions at R_1 with packets retransmitted from R_2, the source should wait for:

$$D = 2T_{tx} + 2T_p \quad (1)$$

seconds before it reads the next packet from its local flash. While Figure 13 depicts T_{tx}, T_r and T_s as activities with fixed duration, in reality each of them completes after a probabilistic amount of time considering the inherent uncertainty of TinyOS split-phase operations (Hill et al., 2000). The high variation of T_{tx} is due to the underlying MAC protocol used in TinyOS. We therefore need to estimate bounds on these parameters to determine the correct value of D from Equation (1).

Figure 12 Outdoor performance of the threshold predictor

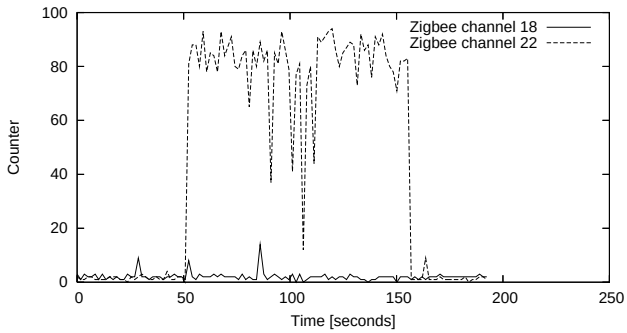


Table 1 presents the results of our measurements on a Tmote Sky mote for the operations mentioned above, using a payload of 22 bytes, which is big enough to hold one set of measurements taken by a sensing node. Given the large deviation in transmission time we selected an upper

bound for T_{tx} equal to the mean value plus two time the standard deviation (10 ms) while we set T_p to 2 ms, for a total $D = 24$ ms. We empirically validated that these values did not cause any packet losses due to collisions and we therefore use them in the remaining experiments.

Figure 13 Multihop data download transaction. Two timelines are shown for each node. The lower one indicates when packet are transmitted while the upper one indicates when packets are received by that node

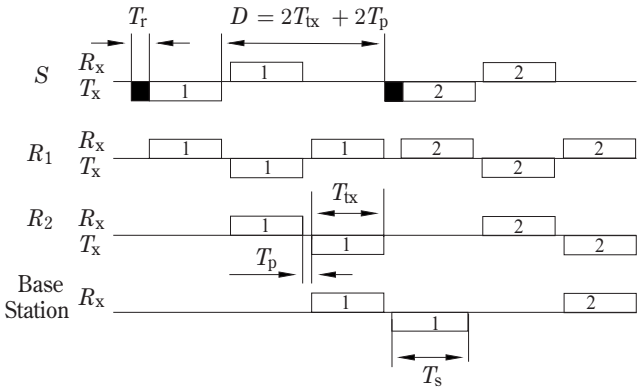


Table 1 Timings for Tmote Sky

Operation		*Mean [ms]*	*SD [ms]*	*Max [ms]*
Flash reading	T_r	1.250	0.008	1.437
Radio sending	T_{tx}	6.073	1.937	28.375
Serial sending	T_s	3.149	0.026	3.343

5.3 Interference detection and avoidance

The goal of the interference detection and avoidance mechanism is to select the radio channel that is least likely to have interference from colocated networks.

At a high level, the proposed algorithm executes before each download operation and consists of two phases: during the first phase, each of the nodes on the multihop path between the sensing node and the gateway independently senses the RF spectrum to select the least noisy radio channel. In the second phase the nodes collaborate to agree on the common channel that is least congested across the whole path. Once this distributed voting phase terminates, all nodes switch to the agreed upon channel and the actual data transfer occurs. The selected channel will be used throughout the entire download operation.

The interference detection component uses the threshold estimator presented in Section 4. We implemented this estimator as a TinyOS component that takes as input the RSSI sampling frequency and the number of samples for each Zigbee channel and responds with an array of 16 counters, each of which represents the number of RSSI samples above the threshold for the corresponding Zigbee channel. Because we are not sending the RSSI measurements over the serial port we increased the sampling rate to 1 kHz. The number of collected samples was the same as in the previous experiments (i.e. 100 samples per channel). This higher frequency allows us to complete a sweep of all 16 Zigbee channels in 1.6 sec which represents a small price compared to the the length of a typical data download which lasts several tens of seconds.

The proposed algorithm involves the following steps:

1 The base-station initiates the process by sending a request to the sensing node to perform RSSI sampling. These requests are routed using the same VC-routing mechanism described in Section .5.2.

2 All relay nodes leading to the destination, also start taking RSSI samples as soon as they forward the request towards the sensing node. When the RSSI sampling process terminates, each nodes sends its results to its parent on the routing path. Given the sequence of the sampling processes at each relay node, the upstream router will most likely have finished its sampling when the results flow upstream towards the base-station.

3 After the base-station collects the reports from all the nodes, it sums all the values for each channel and selects the channel with the lowest number of RSSI samples above the configured threshold, randomly breaking ties when necessary.

4 The base-station subsequently sends a change-of-channel request to the sensing node. When the sensing node receives this request, it switches to the requested channel and immediately starts sending status messages over the new channel. The base-station sends similar requests to each of the relay nodes starting from the destination towards the base-station. We use a decreasing TTL approach to ensure messages reach the intended relay without explicit knowledge of the IDs of the path relays. Finally, the base station switches its own radio to the new channel. Once the base-station starts receiving status updates from the sensing node, it knows that all nodes on the path have successfully switched channels.

Given the polling nature of the routing scheme described in Section 5.1, in which only a single network path is active at any given time, the algorithm presented above is optimised to coordinate channel selections of only the sensor nodes involved in the current download cycle. The same core idea however can be used in a variety of settings. For example, one could integrate it with TSMP (Dust Networks, Inc., 2006) in which case the hopping sequence would not include channels with high levels of interference. A similar approach is taken in the 1.2 Bluetooth standard using Adaptive Frequency Hopping (AFH) (Bluetooth Special Interest Group (SIG), 2003).

6 Evaluation

We evaluate the effectiveness of the proposed interference detection and avoidance mechanism through a series of tests. All tests were performed in the indoor testbed for two reasons: we did not want to disrupt data collection in our live soil monitoring network and it was considerably simpler to experiment and collect data in our internal testbed. Since the interference results presented in Section 3.4 show that interference in outdoor settings is not much lower than that

Table 2 Loss rate comparison using streaming

No.	*Zigbee channel 19*			*Best channel*		
of hops	*Min*	*Mean*	*Max*	*Min*	*Mean*	*Max*
1	0.226	0.226	0.226	0	0.0002	0.001
2	0.424	0.531	0.649	0.0007	0.0094	0.014
3	0.415	0.582	0.668	0.0005	0.0088	0.019

of indoor deployments, we believe the following results will carry over to open-field deployments.

In each of the experiments, we transfer 64,000 bytes of payload over the primary network path while at the same time we perform a UDP flood transfer over WiFi channel 8 at maximum speed. We use two different scenarios: in the first case the primary download occurs over Zigbee channel 19, which overlaps with WiFi channel 8. This scenario is designed to showcase the worst case loss rate due to cross-network interference. The primary channel in the second case is selected by the detection and avoidance algorithm before the actual data download occurs. We repeat both scenarios for network paths with one, two and three hops (i.e. one to two data relays) to evaluate the negative effect of link interference on end-to-end loss rate and the ability of our algorithm to coordinate channel selection at multiple network nodes.

Table 2 presents the minimum, average and maximum loss rates calculated after running each of the six combinations for five times. We note that the tabulated loss rates correspond to the percentage of packets lost during the initial 'bulk phase' of the download cycle, before the gateway has a chance to request for retransmissions following the ARQ protocol mentioned in Section 5.1. As expected, data transfers over channel 19 sustained heavy losses with their rate increasing as the length of the routing path increases. On the other hand, the selection algorithm was successful in detecting interference on channel 19 and avoiding it. In all experiments, the algorithm chose channels 11–13 and 24–26, with 13 being the most popular choice. As a result, the recorded loss rate was negligible -0.09% for three hops compared to 58.2% when no interference avoidance is used.

The interference avoidance algorithm incurs some overhead since it requires every node to keep its radio on while it samples the RF spectrum. However, this process completes only after 1.6 sec (assuming 100 samples/channel and sampling rate of 1 KHz). This is an overhead of at most 3.8% considering that the 'bulk' phase of the download for the fastest scenario (1 hop) takes around 42 sec. On the other hand, since almost all packets are successfully delivered during this phase, far fewer retransmissions will be necessary and therefore the radios will be turned off much faster thus saving energy.

Assuming that losses are independent, then, in the case of a three-hop network, each packet needs to be transmitted ~ 2.38 times in total (from the geometric distribution, the total number of transmissions is calculated as $1/1-p$, where $p = 0.58$ is the path loss probability) resulting in $\sim$138% increase in power expenditures. This rudimentary calculation highlights the significant benefits that interference avoidance can provide during long data transfers.

7 Summary

In this paper we identify the problem of interference caused by WiFi networks deployed in the same general geographical area with WSNs that use 802.15.4 radios. Given the big discrepancies between maximum transmission powers across the two radio technologies, interference from competing WiFi networks can cause high data loss rates in low-power, battery-operated WSNs. We verify this expectation through measurements collected from WSNs using 802.15.4 radios, deployed within an office building as well as in the field and find that interference from WiFi networks can cause up to 58% of packet loss in a WSN using multihop routing.

We propose interference estimation mechanisms that use RSSI samples to detect 802.15.4 channels overlapping with WiFi channels used by nearby 802.11 networks. These estimators are sufficiently lightweight that can be implemented in current generation motes and we show they can effectively detect noisy channels as well as the ideal channels from an interference perspective. These estimators are incorporated in algorithms which coordinate channel selection across multiple WSN nodes on the routing path between a data source and a gateway that wants to reliably extract data from that source. Our results show that the algorithms are extremely effective in detecting and avoiding interference, reducing loss rate from 58% to less than 1%.

References

Adler, R., Buonadonna, P., Chhabra, J., Flanigan, M., Krishnamurthy, L., Kushalnagar, N., Nachman, L. and Yarvis, M. (2005) 'Design and deployment of industrial sensor networks: experiences from north Sea and a semiconductor plant', *Proceedings of ACM Sensys 2005*, November.

Bluetooth Special Interest Group (SIG) (2003) 'Bluetooth SIG adopts version 1.2 of wireles technology specification', Available at: http://bluetooth.com/Bluetooth/Press/SIG/Bluetooth_SIG_Adopts_ Version_1%2_of_Wireless_Technology_Specification.htm, November.

Burns, R., Terzis, A. and Franklin, M. (2006) 'Software tools for sensor-based science', *Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets 2006)*, February.

Crossbow Technology Inc. (2004) 'Avoiding RF interference between WiFi and Zigbee', Available at: http://www.xbow.com/products/Product_pdf_files/Wireless_pdf/ZigBeeandWi%FiInterference.pdf, December.

Challapali, K., Shankar, S. and Cordeiro, C. (2005) 'Spectrum agile radios: utilization and sensing architectures', *Proceedings of DySPAN 2005*.

Cisco Systems Inc. (2006) 'Cisco Aironet 1240AG Series 802.11A/B/G access point', Available at: http://www.cisco.com/application/pdf/en/us/guest/products/ps6521/c1650/%cdccont_0900aecd8031c844.pdf.

Demirhan, M., Hazra, M., Yarvis, M. and Kushalnagar, N. (2005) 'Self configuring transmission channel for wireless mesh networks', *Proceedings of ACM SIGCOMM Asia Workshop 2005*, April.

Dust Networks, Inc. (2006) 'Time synchronized mesh protocol', Available at: http://www.dustnetworks.com/docs/TSMP_Whitepaper.pdf.

Felegyhazi, M. and Hubaux, J-P. (2005) 'Wireless operators in a shared spectrum', *Proceedings of INFOCOM*, April.

Ganesan, G. and Li, Y.G. (2005) 'Cooperative spectrum sensing in cognitive radio networks', *Proceedings of DySPAN 2005*.

Ghasemi, A. and Sousa, E.S. (2005) 'Collaborative spectrum sensing for opportunistic access in fading environments', *Proceedings of DySPAN 2005*.

Golmie, N., Van Dyck, R.E. and Soltanian, A. (2001) 'Interference of bluetooth and IEEE 802.11: simulation modeling and performance evaluation', *Proceedings of the ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, July.

Halldorsson, M.M. Halpern, J.Y. Li (Erran), L. and Mironki, V.S. (2004) 'On spectrum sharing games', *Proceedings of PODC 2004*, July.

Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D. and Pister, K. (2000) 'System architecture directions for network sensors', *Proceedings of ASPLOS 2000*, November.

Kim, S. and Culler, D. (2003) 'Structural health monitoring of the golden gate bridge', Available at: https://www.eecs.berkeley.edu/binetude/ggb/.

Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D. and Anderson, J. (2002) 'Wireless sensor networks for habitat monitoring', *Proceedings of 2002 ACM International Workshop on Wireless Sensor Networks and Applications*, September.

Mishra, A., Brik, V., Banerjee, S., Srinivasan, A. and Arbaugh, W. (2006) 'A client-driven approach for channel management in wireless LANs', *Proceedings of INFOCOM*, April.

Moteiv Corporation (2006) 'Tmote Connect: wireless gateway appliance software', Available at: http://www.moteiv.com/products/docs/tmote-connect-datasheet.pdf, February.

Musăloiu-E., R., Terzis, A., Szlavecz, K., Szalay, A., Cogan, J. and Gray, J. (2006) 'Life under your feet: a wireless soil ecology sensor network', *Proceedings of the Third Workshop on Embedded Networked Sensors (Em- Nets 2006)*, May.

Polastre, J., Szewczyk, R. and Culler, D. (2005) 'Telos: enabling ultra-low power wireless research', *Proceedings of the Fourth International Conference on Information Processing in Sensor Networks: Special track on Platform Tools and Design Methods for Network Embedded Sensors (IPSN/SPOTS)*, April.

Srinivasan, K. and Levis, P. (2006) 'RSSI is under appreciated', *Proceedings of the 3rd Workshop on Embedded Networked Sensors (EmNets)*, May.

Texas Instruments (2006) '2.4 GHz IEEE 802.15.4/ZigBee-ready RF Transceiver', Available at: http://www.chipcon.com/files/CC2420_Data_Sheet_1_3.pdf.

Tolle, G., Polastre, J., Szewczyk, R., Turner, N., Tu, K. Buonadonna, P., Burgess, S., Gay, D., Hong, W., Dawson, T. and Culler, D. (2005) 'A macroscope in the redwoods', *Proceedings of the Third ACM Conference on Embedded Networked Sensor Systems (SenSys)*, November.

Woodings, R.W. and Gerrior, M. (2006) 'Avoiding interference in the 2.4-GHz ISM band', Available at: http://www.wirelessnetdesignline.com/showArticle.jhtml?articleID=604012%06, July.

Xu, N., Rangwala, S., Chintalapudi, K.K., Ganesan, D., Broad, A., Govindan, R. and Estrin, D. (2004) 'A wireless sensor network for structural monitoring', *Proceedings of SenSys 2004*, November.

Zhou, G., Stankovic, J. and Son, S. (2006) 'Crowded spectrum in wireless sensor networks', *Proceedings of the 3rd Workshop on Embedded Networked Sensors (Em- Nets)*, May.

Notes

[1]For the rest of this paper we use the terms Zigbee and 802.15.4 interchangeably with the understanding that we refer to the PHY-level protocol of the Zigbee protocol stack.

[2]We followed this approach since it was cumbersome to change the channels used by the WiFi APs.

[3]The CC2420 radio specification suggests that the received power P can be calculated as P = `RSSI_VAL` + `RSSI_OFFSET` where `RSSI_OFFSET` is found empirically and should be approximately –45 dBm Texas Instruments, 2006. Since we did not directly measure P we chose to show the raw `RSSI_VAL` value instead of a potentially inaccurate estimate of P.

[4]In reality, R_2 could also receive the packet at the same time if it is within the transmission range of S. R_2 will however ignore the packet since it is not the packet's intended destination.