Minimizing 802.11 Interference on Zigbee Medical Sensors[*]

James Hou, Benjamin Chang, Dae-Ki Cho[†], Mario Gerla[†]
Department of Computer Science
University of California, Los Angeles
{jameshou,bychang}@ucla.edu,[†]{dkcho, gerla}@cs.ucla.edu

ABSTRACT

The rapidly growing market for wireless technologies (Body LAN, cellular and Wireless LAN) in medical environments has led to a critical need for effective cable replacement solutions. This will enable widespread use of wireless body sensors, utilizing both an effective transmission protocol as well as providing proper infrastructure support. One of the emerging solutions for the body network is the ZigBee technology; primarily because it utilizes small format, low-power, long battery life radios . It is generally used for applications that can tolerate a low transmission rate, but demand long battery life. An essential requirement of Body LANs for patient care is to guarantee reliable service. In this respect, ZigBee faces severe interference problems in the presence of various 802.11 networks, and its viability in the medical environment is greatly diminished. This interference is caused by the fact that ZigBee shares channel spectrum with the 802.11 protocols. In this paper, we first confirm the claims that ZigBee is vulnerable to interference from 802.11. Then, we propose a solution for minimizing interference from 802.11 in ZigBee medical sensors.

Keywords

ZigBee, 802.15.4, 802.11, interference

1. INTRODUCTION

As the medical industry continues to develop new devices to assist nurses and coaches to monitor the health of their patients, new technologies have become an essential stepping stone to providing the next level of care. One key innovation is the use of wireless technologies. By using wireless technologies, medical organizations might be able to leverage the use of additional sensors, which provide deeper insight into a patient's conditions [6]. Additionally, wireless sensors may allow for placements that might have otherwise been inconvenient, uncomfortable, or simply too complicated. These benefits improve the level of care but also introduce new potential issues. The potential exists that these devices will interfere with one another and cause disruptions in service and reliability. An "intelligent" method to deal with interference must be developed. This is especially important in an area where multiple protocols coexist.

In the realm of medical sensing, it is ideal to use a technology which can be battery powered, lightweight, and does not require frequent charging. For example, the most obvious application of wireless technology in a medical situation might be a wireless ECG [4]. The LifeSync wireless ECG system uses several Bluetooth sensors placed around the body. LifeSync argues that a wireless ECG would provide not only easier use of an ECG, but also would avoid potential bacteria that reside on traditional ECG wires. In order for a protocol to satisfy these medical applications, it must be low power, highly portable, affordable, and reliable. The two most obvious choices for these technologies would be ZigBee and Bluetooth. Bluetooth, however, is more expensive than ZigBee, requires more battery power, and is better suited to situations in which connections are persistent, rather than short quick bursts in spread out time intervals [7]. Therefore, Bluetooth does not suit these types of applications as well as the ZigBee protocol.

The ZigBee protocol shares radio spectrum with the 802.11 wireless networking protocol [5]. This means that in the presence of an 802.11 transmission, a ZigBee transmission could potentially be interfered with, or blocked altogether. In order for an issue to arise, it would be necessary for one of two conditions to happen. Firstly, it would be necessary for a high concentration of ZigBee and 802.11 devices to be within close proximity of each other, sending data at moderate rates. Another possibility would be a smaller number of devices, all transmitting at near-maximum rates of speed. Clearly, if a large number of devices were all transmitting at a high rate of speed, this could also cause a problem. This could be a significant problem in the medical setting, since there may be a large number of patients within close proximity of one another, especially if they are in transmission range of each other. The likelihood of this being a problem will be increased by the presence of 802.11 routers, which may serve actual medical purposes, or simply provide patients and visitors a connection to the internet. An example of what such a layout might look like is shown in Figure 1. Over time, the number of devices on these networks could rise quickly. While a hospital may start with the use of a wireless ECG device, they might later extend that to using sensors for situational awareness, or even tracking the whereabouts of a patient. The presence of 802.11 is particularly concerning, because it has a transmission power 30 times larger than ZigBee's, and an intensity 4 times larger

[*]This material is based upon work supported by the Microsoft Research and by the National Science Foundation under Grant No. 0832370. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

BodyNets '09 April 1-3, 2009 Los Angeles, CA, USA
Copyright 2008 ICST 978-963-9799-41-7 ...$5.00.

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029

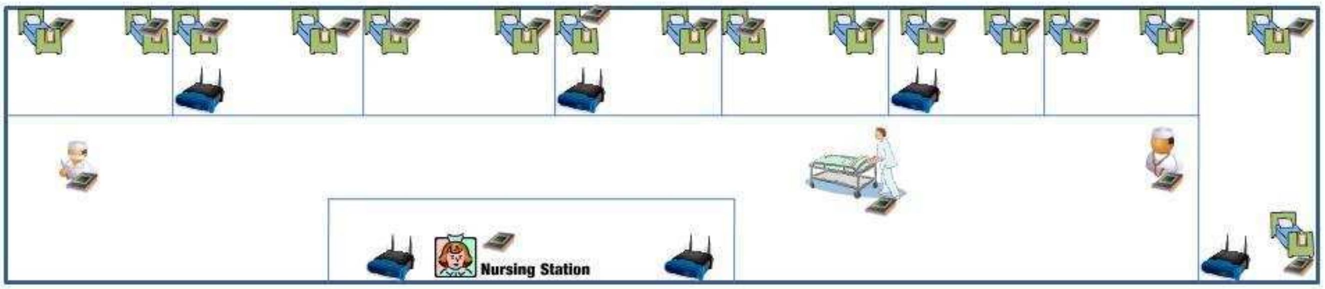

Figure 1: Sample layout of a hospital, showing sensors used for situational awareness, 802.11 networks, and medical devices. The number of potential participants is high.

[11].

In saying this, the potential for 802.11 to overpower a neighboring ZigBee transmission could be high. In addition, the CSMA/CA schemes implemented by 802.11 do not recognize the transmission efforts of ZigBee devices, meaning the ZigBee devices would be ignored if their transmission attempts were detected by the 802.11 devices.

ZigBee is a specification for a wireless standard based on the IEEE 802.15.4 standard [2]. Since it is based on the 802.15.4 standard, ZigBee is susceptible to many of the same problems as the 802.15.4 standard [12]. It has been claimed that ZigBee will face severe interference issues in the presence of numerous 802.11 networks, and that its viability in such an environment will be greatly diminished [5]. Those who allege that ZigBee will face interference issues state that ZigBee uses channels that overlap with 802.11, and will face at least interference, if not 100% packet loss from competing networks. However, the ZigBee Alliance has published a white paper refuting these claims and stating that while the channels do overlap, the nature of ZigBee's transmission protocols prevent 802.11 from interfering with ZigBee transmissions [3]. In theory, because ZigBee transmissions are short in nature, and infrequent, it is possible that 802.11 traffic may leave a large enough time interval open in its transmissions to allow for successful ZigBee transmissions. This paper seeks to establish that the claims that ZigBee is vulnerable to interference are accurate, and to propose several potential solutions to the interference problem ZigBee faces. The solutions presented will offer a few ways in which the interference problem can be dealt with, and show the results of the implemented solutions.

2. RELATED WORK

As stated, there has been previous work in trying to determine whether or not there is a genuine interference problem between ZigBee and 802.11. Figure 2 is a diagram showing where their communication channels overlap [5].

In figure 2, ZigBee is the protocol on top, and 802.11 on bottom. It can be plainly seen that these protocols overlap each other, and this can most definitely lead to interference. The Crossbow group evaluated ZigBee performance in an environment where the ZigBee and 802.11 channels overlapped. They were able to demonstrate that in this scenario, a detectable amount of packet loss was found. Their work showed that ZigBee might experience packet loss up to 5%. We would like to see if something more significant can be shown. In addition to the work done by Crossbow, an organization called the Z-Wave Alliance has also done prior research into this topic [3]. The summary of their findings was that in specific scenarios, they were able to completely prevent ZigBee from transmitting any data at all. They also found that if a ZigBee device was physically located on the same piece of hardware as an 802.11 device, the ZigBee device would never be able to transmit its message if the 802.11 device was transmitting. We were able to get around this by staggering the transmission times of both protocols so that they would not transmit simultaneously; this is explained in greater detail later on. Thirdly, Musaloui-E et al found that ZigBee and 802.11 experienced interference rates of up to 58% when baselining the potential interference faced by ZigBee in their work. There has been some controversy regarding whether or not interference is a legitimate interference problem between 802.11 and ZigBee [9]. The majority of papers which have documented the scenario have suggested that an interference problem exists, and our findings agree.

Additionally, Musaloui-E et al performed work in experimenting with the use of channel hopping to avoid interference. Their work was able to successfully reduce interference from 802.11 networks from as great as 58% to less than 1%. However, their approach requires that there are unoccupied wireless channels which can be utilized. For our work, in a hospital setting, it is quite possible that there may be a large number of networks and that all available channels could currently be occupied. As stated by Musaloui-E et al, there exist two channels which can be utilized by ZigBee, but not by 802.11. However, they also stated that these channels are potentially occupied by 802.11 in Asia, and therefore cannot be relied upon. Our method of resolving the interference issue is to directly reduce the traffic generated by the 802.11 devices. This approach makes more sense for a medical network setting; however, it also requires a more complicated hardware solution, and the availability of an intermediate hybrid device.

In Omaha, Midwest Surgical Hospital recently incorporated an 802.11 network into their infrastructure. They found that even using 802.11 devices with standard medical equipment was a difficult task. In some cases, very slight timing issues can cause significant problems. Incorporating a ZigBee network will add an additional layer of complexity, and interference between the different types of network will need to be resolved [10].

Finally, a technology called Wibree is currently being developed by Nokia [8]. This technology is a low power version of Bluetooth, which does not face the significant interference issues that ZigBee does. By adapting the Bluetooth standard to a lower power device

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029

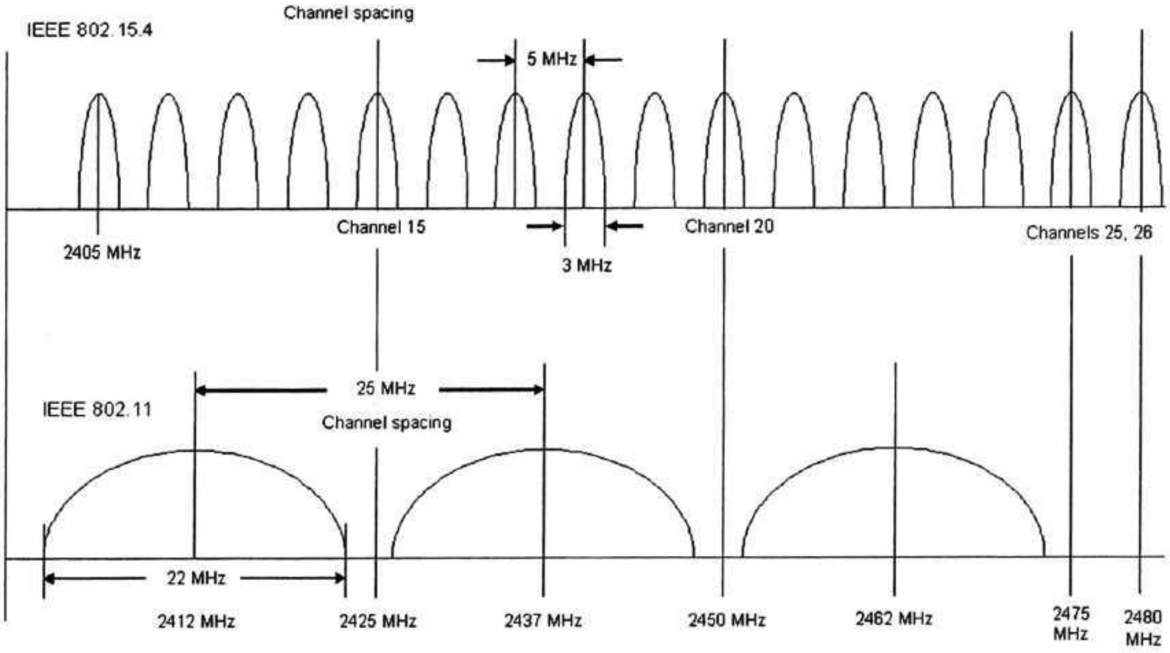

Figure 2: IEEE802.15.4 and IEEE802.11 Channels

designed to transmit small amount of data infrequently, the way ZigBee does, allows for Bluetooth to largely replace the need for a protocol like ZigBee. Another benefit to Wibree would be the built in frequency hopping methodology used by Bluetooth. This might be an excellent solution when it arrives, but is currently not ready for the market.

3. WIRELESS PROTOCOLS

ZigBee was developed and is supported by a group called the Zigbee Alliance [2]. This group is composed of notable members in industry, including Honeywell, Motorola, Philips, Siemens, Samsung, Mitsubishi, and Texas Instruments. The ZigBee standard is currently considered complete, and the last revision was released in 2006. Current applications for ZigBee include: heating controls, HVAC control, lighting control, automatic meter reading, demand response, environmental controls, home security, and medical sensing/monitoring. ZigBee features throughput of 250Kbps at 2.4GHz, with 16 channels available to it, and 40Kbps at 915MHz, with 10 channels available at that frequency. ZigBee is capable of transmitting data over distances of up to 100 meters. ZigBee sports a small footprint, requiring as little as 4 kilobytes of system resources, and up to 32 kilobytes. This compares to the 802.11 standard which requires over 1 megabyte. ZigBee is also designed to support a battery life of up to 1000 days based on its low-power design. ZigBee is also very scalable, supporting up to 64,000 nodes under a single coordinator. These coordinators may be linked together to create even larger networks. These factors make ZigBee an extremely attractive option for users who wish to create simple devices to sense or monitor conditions wirelessly.

The 802.11 standard was developed by the IEEE LAN/MAN standards committee to support wireless networks [1]. It is a family of sub-standards that is composed of several modulation techniques which use the same basic protocols. The primary sub-standards include a, b, g, and n. In fact, there are sub-standards that utilize every letter from a-z (excepting x), but these 4 are the most widely used. The intervening standards in between were often used to modify security protocol, international differences, or specific applications. For example, 802.11p was designated for the WAVE standard, which stands for Wireless Access for the Vehicular Environment. The a, b, g, and n revisions represented significant changes in technology and user functionality. 802.11a was released in October of 1999, operated at the 5GHz frequency range, and supported a data rate of 54 Mbps. It had a range of 35m, which was effectively shorter than other protocols using the 2.4GHz range. 802.11a utilized OFDM to obtain a higher overall throughput when compared to other standards of its time. 802.11b was released at the same time as 802.11a, and carried a maximum throughput of 11Mbps, with a range of 38m in the 2.4GHz spectrum. This standard was offered at a lower price than 802.11a, and became the defacto wireless standard of its time. Operating in the 2.4GHz range came with its own issues, since the protocol frequently experienced interference from objects which included Bluetooth devices, microwaves, and cordless telephones. It was later, in 2003, when 802.11g was released. 802.11g was backwards compatible with 802.11b. It operated in the same 2.4GHz spectrum, but used OFDM (like 802.11a) to increase its theoretical bandwidth peak to 54Mbps. Due to 802.11g being backwards compatible to 802.11b, an 802.11g network was susceptible to poor performance whenever an 802.11b device signed onto the network. It has been estimated that 802.11g networks encountered a performance reduction of 21% whenever an 802.11b device was a member of the network. Lastly, the 802.11n standard was developed, but not finalized. The 802.11n standard utilized MIMO (multiple in, multiple out) along with a slew of new features. 802.11n networks are capable of operating at both 5GHz and 2.4GHz ranges, and can reach speeds of up to 108Mbps. Range supported by this draft specification doubles that of previous wireless standards, at 70m. Each of these protocols uses the 2.4GHz ISM band. Since ZigBee operates in this band, we will only discuss the 2.4GHz band, and not the 5GHz

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029

band. Each band is divided into channels, 13 in most countries, or 14 in Japan. Each channel is spaced 5MHz apart, and each has a width of 22MHz. The channels which may be used once varied from country to country, however, all channels can be used in the vast majority of countries now. Due to signal attenuation, only every 4th or 5th channel may be used simultaneously. This may significantly reduce the number of channels which may be used in a network without experiencing interference.

In general, interference in wireless networks occurs when an object exists in the environment that is physically interfering with the signal, or when there is a device transmitting its own data which interferes with the signal. Environmental interference has no different effect upon ZigBee in the presence of an 802.11 network, though another device might. There have been many schemes devised to reduce the chance of device interference happening, ranging from direct-sequence spread spectrum (DSSS), time sharing, CSMA, and utilizing different channels. Despite these advances, interference is still an issue when two devices attempt to transmit data at the same time. This is especially true when these devices use different protocols. This happens because many devices share similar frequencies, such as the 2.4 GHz ISM band, and each standard proposes a different way to mitigate interference and collisions. ZigBee supports 16 channels; however, 15 of these channels overlap with channels used by 802.11. As stated above, in 802.11 networks, 802.11 devices may experience interference from one another if they operate within 4 channels of one another in close proximity. With ZigBee utilizing these same channels, it is highly likely that ZigBee devices will face interference from 802.11 devices. In theory, since ZigBee transmits data in short bursts, it could be possible for a small ZigBee network to subsist on this sole channel which is not used by 802.11 [12]. However, this is not to say that other types of networks might not also use this channel, such as Bluetooth. In addition, when a large ZigBee network is present, it may become necessary for ZigBee devices to use different channels. There also exists the possibility of a large number of small ZigBee networks tied together through their coordinators. Essentially, it is still important to determine whether or not ZigBee's interference model is strong enough to deal with any type of interference that may exist in its environment. However, because ZigBee uses such small time frames to transmit data, and it does not require a large window of time frequently, it can be reasoned that ZigBee will only have trouble when there is a heavy amount of traffic on a channel. As it currently stands, only 802.11 devices would sustain a data transmission with both the power and duration to realistically interfere with a ZigBee device for a prolonged period of time.

In order to determine the effect of an 802.11 network on a ZigBee network, it was necessary to setup an experiment which would test ZigBee's viability in the face of interference. For this experiment, an 802.11 ad-hoc network was setup between two laptops. These laptops transmitted data at rates between 300KBps to 1MBps, which represent light congestion to heavy congestion. Secondly, two ZigBee devices were setup and programmed to transmit at a rate of 500 Bps (a realistic real-world transmission rate). Both protocols were set to use the same channel so that collisions would be caused by simultaneous transmissions. However, in the real world, since 802.11 channels share some overlapping regions, it would be possible that even adjacent channels would generate some interference. However, regardless of whether the interference is caused by an adjacent channel or the current channel, the important factor is that interference is caused.

The first scenario that was tested was with two ZigBee devices transmitting at 500Bps with no 802.11 interference. In this scenario, no detectable interference or packet loss was encountered.

The next scenario would be to repeat the previous scenario, but with two 802.11 devices transmitting at the same time. This created the scenario in which 802.11 devices would theoretically interfere with ZigBee transmissions. When the 802.11 devices transmitted at 1MBps, the ZigBee devices experienced an 80% packet loss. However, when the transmission rate was reduced to 300KBps, the ZigBee devices experienced a 20% packet loss. This experiment showed that without a doubt, ZigBee was vulnerable to interference from 802.11, and that its collision avoidance schemes are not advanced enough to avoid packet loss in a highly congested environment.

4. PROPOSED SOLUTION

First, our work will show that the interference problem exists between 802.11 and ZigBee. We propose to do this by setting up an 802.11 network that will be transmitting data at a high rate of speed, so that it fully occupies the channel. Next, we will setup a pair of ZigBee transmitters to show that the ZigBee devices are interfered with indefinitely. Finally, we will attempt to resolve the interference problem.

The most difficult issue when trying to remedy the interference problem between ZigBee and 802.11 is due to the differences in their physical layers. ZigBee devices and 802.11 devices communicate with different modulation, slightly different frequencies, and different types of packets, with different headers and packet shapes. One standard cannot communicate with the other without significant modification to the underlying hardware. It is for this reason that ZigBee cannot issue a packet to the 802.11 devices indicating that it wishes to transmit data. We propose a solution which utilizes a hardware setup that includes both ZigBee and 802.11 transmitters. This will allow us to transmit both 802.11 and ZigBee messages. This hybrid device would be able to coordinate messages between 802.11 and ZigBee and act as a mediator between the protocols, thereby solving the more difficult aspect of the problem.

There are two primary ways for this hybrid device to intervene between 802.11 and ZigBee. The first method would be to transmit an 802.11 packet indicating that this packet would have an unusually long duration (perhaps 64ms or so), permitting ZigBee to transmit during this period in which other 802.11 devices will "sleep." The second method would be the use of RTS (Request to Send)/CTS (Clear to Send) messages to clear 802.11 traffic. This idea works on the theory that sending out a CTS message will block all 802.11 devices from transmitting for a specified period of time. The goal of both of these solutions is to temporarily block out 802.11 messages for a window of time large enough that ZigBee devices can successfully transmit their messages, thereby resolving the interference issue. If 802.11 traffic can be blocked for short periods of time which are long enough for ZigBee devices to transmit, then it will be possible to develop a market solution that will alleviate the contention issues between ZigBee and 802.11.

Our experiment will focus primarily around the second solution. The first solution mentioned would not be viable on most networks, requires an abuse of the 802.11 protocol, and is not guaranteed to work. In our work, we found the 802.11 devices completely ignored the rogue packets. The second solution will be able to prevent 802.11 devices from sending messages. It is important to

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029

note that typically, Windows devices themselves do not use the RTS/CTS scheme, but that they will still respect the rules set by this scheme. By developing a solution which will directly interfere with the 802.11 protocol's ability to transmit messages, this scheme is inherently unfair to 802.11. However, this unfairness should not be a significant issue, as ZigBee transmissions are typically very small, very short, and need only a small window of time to perform its task. It is also assumed that the nature of ZigBee devices is to transmit small amounts of data infrequently, rather than to transmit a large quantity of data over a short period of time. Lastly, it is assumed that ZigBee device messages may be life threatening, and that 802.11 traffic will not be. This means that this solution will most likely not be a significant detriment to the performance of 802.11. However, in order to guarantee that 802.11 is never completely shut down, it is possible to devise a solution which limits ZigBee's channel occupancy rate. This way, if the channel occupancy is ever too high, ZigBee can back off and allow 802.11 to transmit, so that the channel can be fairly shared between the two protocols.

5. IMPLEMENTATION

In order to implement our first proposed solution as described, it is necessary to be able to craft our own 802.11 frames and transmit these packets in a way that is ordinarily not allowed by the network stack. Most modern 802.11 network cards do not permit such promiscuous actions and only allow certain type of frames to be transmitted due to security issues. Crafting packets is considered a security threat and as such is not supported by most modern 802.11 network cards. As such, we had to use an older network card based on the Atheros chip which allowed such activity. On the software side, in order to support packet injection, we used the Lorcon framework to be able to craft our own packets and then inject these custom packets to the 802.11 network card to be transmitted. The second proposed solution would use the same basic setup, only it would not be necessary to craft invalid packets. Early on, we discovered that falsely declaring the size of a packet and hoping that 802.11 devices would remain silent for the duration of the supposed packet's length was an unviable solution. This was because the other 802.11 devices would simply detect that the channel was not in use, and these devices in turn would simply continue with their transmissions without interference. Upon experimenting with the CTS packets, we discovered that the behavior of 802.11 traffic with respect to CTS messages was exactly as was desired. It was originally thought that Windows devices might ignore the CTS messages; being able to effectively jam them makes this solution viable. However, one key limitation is that transmitting CTS packets is possible in Linux, but not in Windows, despite Windows devices respecting the rights of the CTS packets. In a hybrid solution, this should be a non-issue. Knowing that 802.11 devices would indeed back off after receiving these CTS packets, we then moved onto developing our proposed solutions. We then decided to develop two types of solutions, one of these would periodically jam 802.11, and see if ZigBee would be able to fit its messages into the empty time frames. The second solution would be to transmit a CTS message directly before a ZigBee message, and verify that the ZigBee message has a high delivery rate.

For the first proposed solution involving the periodic blocker, the periodic blocker would simply transmit these CTS packets periodically with a lockout of 32 ms. We decided to use 32 ms because it provides a window of time which is large enough for a ZigBee transmission to be sent, while minimizing the impact on 802.11. In addition, it was observed that CTS messages with exceptionally high delay values (i.e. 65536 ms) would simply be ignored or dropped by the target recipients. In addition, we sent these messages sporadically in the hopes of ZigBee devices being able to latch onto small periods of open windows during which they would be able to transmit packets interference free. This first implementation was created to ensure that blocking 802.11 packets could present some reasonable improvements to the transmission of ZigBee data. Our second solution was more controlled, and thus a much more in-depth solution had to be developed. We first created a ZigBee application that would transmit packets, which was then coupled with the 802.11 interference tool so that the ZigBee transmitter would then issue a CTS message followed by a ZigBee transmission. The CTS message would request a delay of 32 ms, as before, with the obvious advantage this time that the ZigBee message would be coordinated with its CTS interference message. This should guarantee a highly accurate packet delivery ratio.

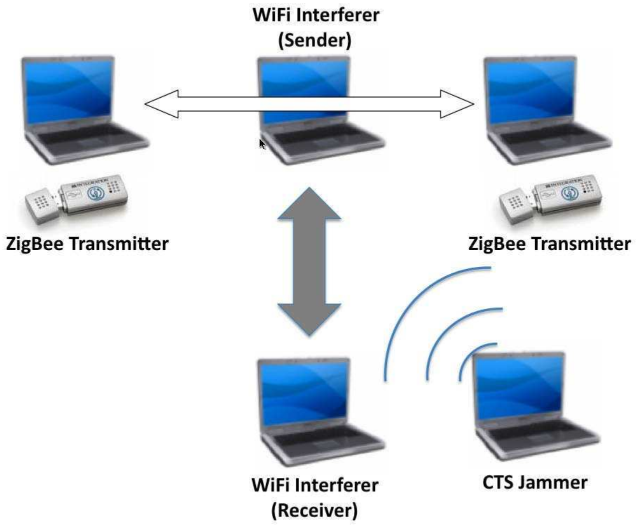

Figure 3: Experiment Setup

6. EXPERIMENTS

We conducted our experiments in a real world environment. It is more practical to observe real world behavior of wireless protocols rather than observing results obtained from simulation. Actual simulation also allowed us to observe behavior caused by the environment. We used a setup where the ZigBee devices were spaced 5 meters apart and the WiFi interferers were placed in between the ZigBee devices as shown in Figure 3. This experiment setup should create a situation in which the maximum amount of interference should occur between the ZigBee and 802.11 devices. This was done to demonstrate the maximum possible 802.11 interference.

Next, we measured the interference between the ZigBee devices in the presence of minimal to no 802.11 traffic. The barometer of interference that we used was the percentage of packets lost between the ZigBee devices. In the presence of minimal to no WiFi traffic, we observed 0% packet loss.

We then observed the interference between the ZigBee devices in the presence of heavy 802.11 traffic. To produce the 802.11 interference, we used two laptops with 802.11 network cards connected via an access point in an infrastructure-based WLAN. Our source 802.11 interferer would send 802.11 data packets to another 802.11

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029

device via a UDP connection. The source interferer would send 500 Byte data packets every 666 μs for a total of around 750000 Bytes/second, equal to 6 Mbps. With this setup, we observed a 56% packet loss. Furthermore, we repeated the same experiment, but in the presence of an ad-hoc network where the 802.11 interferers were connected to each other directly, without the use of a router. In this setup, we observed a 33% packet loss. Figure 4 shows a sample timeline of the difficulty that might be faced by a ZigBee device. Though experiment results were not as clear-cut, the point is illustrated accurately.

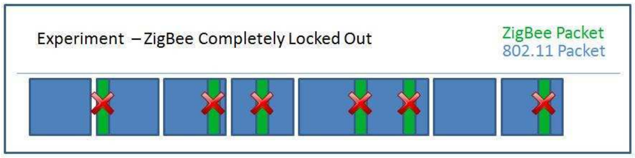

Figure 4: Timeline showing ZigBee devices unable to find an open time slot to utilize.

In theory, if an 802.11 device issued a nonstop stream of CTS messages, no other 802.11 device on that network would attempt to send any messages. This is shown in figure 5. By proving that 802.11 could be locked out with CTS messages, we found that we had the framework for building a practical interference device.

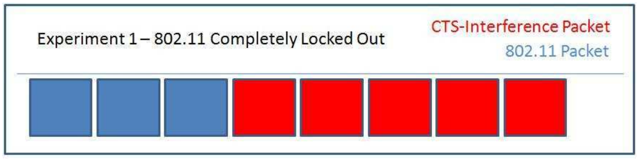

Figure 5: CTS interferer locking out 802.11 transmissions

Next, we needed to test our periodic blocking solution, which transmits a CTS packet randomly in a set time interval. To test our periodic blocking solution, we repeated the experiment in the presence of heavy 802.11 traffic, but with our periodic blocker injecting CTS packets at a set interval. With the periodic blocking solution, we observed only a 2.5% packet loss when the 802.11 interferers were connected via an access point and only an 18% packet loss when the 802.11 interferers were directly connected via an ad-hoc network.

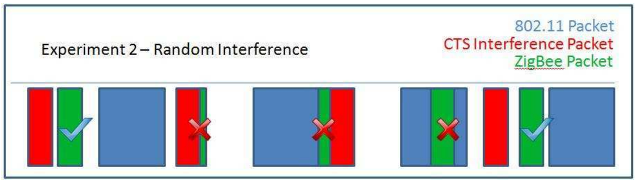

Figure 6: Periodic CTS messages can help or hurt the odds of a packet being transmitted successfully.

Unfortunately when we conducted additional experiments with our periodic blocking solution, we noticed that in the presence of medium 802.11 interference, the periodic blocking solution would actually increase the packet loss rate between the ZigBee devices. As shown in Figure 6, the randomly timed packets would collide with each other. Sometimes, the result was desirable. Other times, it might increase the interference faced. When the source 802.11 device sends around 2.56 Mbps of data to the destination device, we noticed only a 17% packet loss. However, when we activate our periodic blocking solution, we noticed the ZigBee packet loss rate increases from 17% to 35%. The results from this experiment can be seen in Table 1.

Table 1: Interference caused by Periodic Blocking

	WIFI	WIFI + Jammer	WIFI	WIFI + Jammer
Send Rate (packets/sec)	2000	2000	600	600
% of packets lost	87%	40%	17%	35%

Based upon these results, it became clear that a more refined solution would be necessary. We then devised a controlled blocking mechanism that would only send a CTS message just before the ZigBee devices sent out a transmission. This required direct modification of the ZigBee protocol instructing it to transmit a CTS packet prior to its own transmission. This change could be built into ZigBee driver software, running on the assumption that the ZigBee device in question will also have access to an 802.11 device. For actual implementation purposes, it would be possible to detect the presence of an 802.11 device prior to deciding whether or not to utilize the CTS blocking code. By submitting the CTS packets only before a ZigBee device was about to communicate, the controlled blocker would never accidentally interfere with outgoing ZigBee transmissions. With the controlled blocking solution, we were able to receive the overwhelming majority of these packets under heavy 802.11 interference from a infrastructure-based WLAN, showing less than 3% packet loss. In the presence of heavy traffic from an ad-hoc WLAN, we observed only a 14% packet loss. However, more importantly, with the controlled blocking solution we did not observe the same increase in packet loss in the presence of medium 802.11 traffic. Figure 8 displays the results from the final experiment using the controlled CTS blocking mechanism. The expected ordering of packet transmissions is shown in figure 7. This figure shows that we obtain the result we desire.

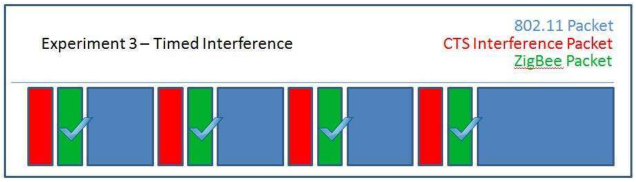

Figure 7: Intelligently timed CTS packets reduce the error rate significantly.

7. ANALYSIS

The first experiment was conducted to determine whether or not 802.11 traffic presented a genuine interference problem for ZigBee devices. The results obtained from that experiment showed that depending upon the network setup, ZigBee devices would potentially see between 33%-56% packet loss. This experiment showed that in the presence of two 802.11 devices, loss rates could easily reach 50%. In our experiments, due to some implementation issues, we were only able to achieve 6Mbps with our 802.11 devices. It is reasonable to conclude that 802.11 traffic can adversely affect the performance of ZigBee devices. We observed that with more 802.11 traffic, it became more and more difficult for ZigBee devices to correctly send a message across the network. With the assumption that our 802.11 devices might eventually be able to speed up to 56Mbps

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029

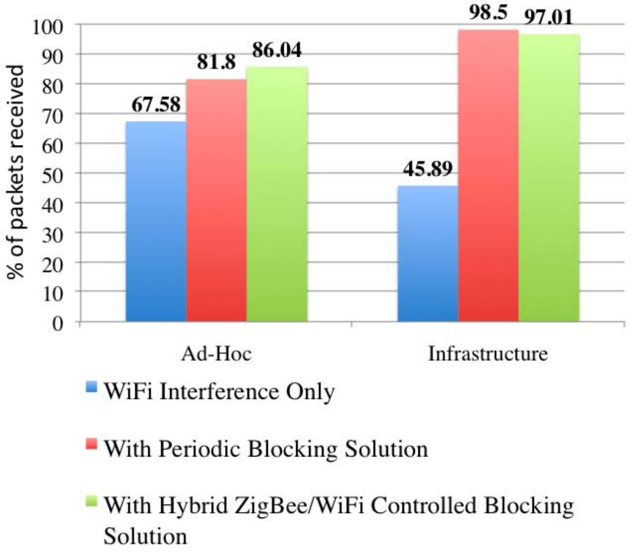

Figure 8: Experiment Results with Proposed Solutions

or higher, without a doubt ZigBee devices could be significantly hampered. It is currently assumed that in the presence of multiple networks and a larger number of devices, the interference problem might completely prevent ZigBee devices from sending any data.

The next experiment sought to show that it might be possible for a CTS inducing device to temporarily lock out 802.11 devices and allow ZigBee devices to transmit. What happened was that in the presence of heavy 802.11 traffic, the CTS interferer was able to temporarily lock out 802.11 transmissions for a short period of time. In doing so, by chance, some of the ZigBee transmissions were able to arrive at their destinations as intended. This of course, relied on the hope that the CTS interferer would block transmissions at precisely the right moments before a ZigBee device would attempt a transmission. The end result was that overall, the number of packets that were lost were reduced significantly. Experimental results showed that the percentage of packets lost when activating the CTS interferer at a high data rate went from 84% loss to a comparatively low 40%. While the interference rate was successfully halved, it is still questionable whether or not a successful transmission rate of 60% is acceptable. Additionally, it was also shown in the second experiment setup that lower levels of 802.11 traffic may cause the interferer to cause my interference than the 802.11 traffic itself. The reason for this is clear, the CTS interferer itself is sending an 802.11 packet, which might themselves interfere with ZigBee devices. If the 802.11 traffic is already sparsely distributed, and further transmissions by the CTS interferer are also evenly distributed, the overall result is higher channel use by 802.11 messages. This means that overall, more time slots are allocated to 802.11 transmissions, when they could've been allocated to a ZigBee transmission. What may likely be happening is that the 802.11 devices are not transmitting often enough to be interrupted by the CTS interferer, thus making the interferer not only ineffective, but a hinderance to ZigBee devices. Acknowledging this problem shows that solving the ZigBee interference problem cannot be done simply, or by brute force, and that the solution implemented for our third experiment was necessary.

Our third and final experiment was the implementation of a ZigBee device with an integrated CTS blocker. This implementation included a short CTS message just before every ZigBee transmission. In this case, regardless of the level of 802.11 traffic, interference was seen to drop significantly, as low as 3%. (in an infrastructure based 802.11 setup) With an ad-hoc network, interference rates were as low as 14%, in comparison to the baseline of 33%. This test case showed significant improvement, and would likely work as a viable solution to the interference problem encountered by ZigBee. This solution worked well because it induced 802.11 delays only when absolutely necessary, and timed each delay so that they came only before a ZigBee transmission. The result is that a solution to the problem encountered in the second experiment: regardless of the level of 802.11 interference, this ZigBee interference tool never increases the rate of packet loss experienced by ZigBee devices. However, this approach has a disadvantage, being that it requires an 802.11 transmitter on-board the ZigBee device. In many cases for medical applications, this can be prohibitively expensive, and negate many of the benefits of using ZigBee.

Thus, while both solutions reliably reduce the amount of interference caused by 802.11 devices, there are major complications in the applications of these two implementations. The advantage of using the periodic blocker is that it can be used in an external third party device. In theory, a third party device might be able to detect intervals which ZigBee devices might be using to transmit, in order to reduce the randomness of the intervals in which it distributes CTS messages. Additionally, a third party device could be low cost, effective, and may be able to support several ZigBee devices. Unfortunately, in some cases, the periodic blocker will currently add to the interference in ZigBee networks and inadvertently interfere with ZigBee transmissions causing additional packet loss. Furthermore, the periodic blocker is not friendly to the 802.11 protocol and will indiscriminately interfere with 802.11 transmissions. The controlled blocking solution however resolves the problem of interfering randomly with 802.11, as well as the issue with lowered transmission rates when 802.11 is not transmitting at high rates. Assuming that ZigBee devices will need to transmit infrequently and in short bursts, the 802.11 devices will not experience a significant amount of interference. ZigBee devices, however, will experience a significant increase in the reliability of their messages. Unfortunately, since this requires a hybrid ZigBee/802.11 device, the negation of ZigBee benefits might make this solution impractical.

8. FUTURE WORK

The experiments performed show that it is possible to successfully block 802.11 devices from interfering with ZigBee devices. However, placing an 802.11 transmission chip could significantly affect the size, performance, and cost of any ZigBee device, and would in fact negate the benefits which ZigBee provides. A possible workaround would be to implement a third party gateway, which itself will contain an 802.11 chip and ZigBee transmitter, but could serve multiple ZigBee devices. For example, this device might take registrations from ZigBee devices, and form a scheduled time interval for device transmissions, much like is done with Bluetooth. Another method might be to passively determine when a ZigBee device is attempting to send data, and determine its transmission intervals based on that passive data. The former idea would provide for a more precise device, which would in theory, provide 802.11 interference exactly when it is needed. However, it would also require additional overhead, since this device would also require tight coordination between the gateway and the ZigBee device. The latter idea is ad-hoc in nature, but depends upon the

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029

gateway's ability to detect the attempted ZigBee transmissions everytime they are desired. Further work is necessary to see if it is possible to create such a device that can support the tight coupling necessary to transmit ZigBee messages in the presence of 802.11 interference.

9. CONCLUSION

In conclusion, we were able to definitively show that it is possible for traffic from an 802.11 protocol to effectively prevent ZigBee from transmitting data. In a medical device, this can be especially important, since many medical applications can mean the difference between life and death. For example, if ZigBee was used on a remote ECG, to remove the necessity of wires hanging from the patient's body, this information would need to be transmitted to a receiver in real time. It would not be acceptable to wait for an 802.11 transmitter to finish its transmissions, so there is a need for immediate transmission. In addition to being able to confirm that 802.11 can interfere with ZigBee, we were able to demonstrate that the RTS/CTS scheme we devised was sufficient to block 802.11 traffic.

Finally, we combined the RTS/CTS jammer with an actual ZigBee transmitter, and showed that it was possible to couple their timing tightly enough so that 802.11 could be blocked in the exact space of time needed to send out a ZigBee transmission. The ZigBee device modification we made allowed our ZigBee device to successfully transmit data in the presence of significant 802.11 interference. In addition to being able to transmit our ZigBee data, we also preserved some sense of fairness, by not significantly affecting the throughput of the 802.11 devices. This method is only partially fair, since in this scheme, ZigBee devices are given transmission priority every time. The reason 802.11 traffic is not being significantly affected is because the size of ZigBee's transmissions are small, and the interruptions in 802.11 traffic are small, providing just as much time as ZigBee needs.

10. REFERENCES

[1] IEEE Std 802.11^{TM}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*, 2007.

[2] 802.15.4-2003, IEEE Standard for Information Technology-Part 15.4: Wireless Medium Access control (MAC) and Physical Layer (PHY) specifications for Low Rate Wireless Personal Area Networks (LR-WPANS). 2003.

[3] BGR. WLAN Interference Raises Doubts about ZigBee, IEEE 802.15.4 Products. *Zensys White Paper*, Mar. 2007.

[4] LifeSync Corporation. LifeSync Wireless ECG System with LeadWear Disposable Cable Replacement System Reduces Artifact and Increases ECG Alarm Accuracy, Oct. 2008.

[5] Avoiding RF Interference Between WiFi and Zigbee. `http://www.xbow.com`.

[6] P. Frehill, D. Chambers, and C. Rotariu. Using Zigbee to Integrate Medical Devices. In *IEEE Engineering in Medicine and Biology Society (EMBS)*, Convention Center, "Cite Internationale", Lyon, France, Aug. 2007.

[7] Bluetooth and ZigBee: compare and contrast. `http://www.techworld.com/mobility/features/index.cfm?featureid=1261`.

[8] EZURiO Ltd. An introduction to Wibree (WHP-050005-1V0). *An EZURiO white paper - explaining wireless*, 2006.

[9] Razvan Musaloiu-E. and Andreas Terzis. Minimising the effect of WiFi interference in 802.15.4 wireless sensor networks. *Int. J. Sen. Netw.*, 3(1):43–54, 2008.

[10] Midwest Surgical Gets Healthy Dose of Smart Wi-Fi to Improve Quality of Patient Treatment. `http://www.ruckuswireless.com/products/casestudies/midwest_surgical/`.

[11] ZigBee and the ISM-Coexistence Issue. `http://wireless.industrial-networking.com/articles/articledisplay.asp?id=765`.

[12] Gilles Thonet, Patrick Allard-Jacquin, and Pierre Colle. ZigBee - WiFi Coexistence. *Schneider Electric White Paper*, April. 2008.

Digital Object Identifier: 10.4108/ICST.BODYNETS2009.6029
http://dx.doi.org/10.4108/ICST.BODYNETS2009.6029