ENSC 427: Communication Networks

Simulation of the ZigBee PAN Protocol in OPNET as a Basis for the Comparison of Competing Sensor Network Technologies

Spring 2010

Final Project

Sonca Teng

http://www.sfu.ca/~rteng/ensc427

rteng@sfu.ca

Sonca Teng

July 14th 2010

Contents

Abstract

The three primary PAN (Personal Area Network) protocols on the market today are the IEEE 802.15.4 based ZigBee and MiWi protocols, and the ISO 18000-7 “Dash-7” protocol. We discuss the suitability of using ZigBee as the protocol of choice for use with low complexity home and commercial use sensor networks, in the context of comparison with the two other competing protocols.

This project is heavily based on modeling and simulation, and is carried out using OPNET 14.0, using the included ZigBee library model.

1. Project Introduction

This project focuses on examining PAN (Personal Area Network) protocols for their suitability for use with sensor networks, specifically for home and commercial use, with a particular focus on ZigBee. The primary motivation for this project is the Wirelessly Networked Faucet System project from ENSC 440 of Spring 2010. For that project, we used the MiWi P2P Protocol, which is based on the same IEEE 802.15.4 standard that ZigBee is based on.

The project will be split into three primary case studies.

First, we will examine the effect on end-to-end delay between one End Device and the Coordinator, incrementally adding more End Devices, and observing any notable behavior along the way.

Second, we will build upon the first case, by using an intermediary Router on a trajectory to examine whether or not there is a difference made by suddenly adding a flood of new devices onto the network. We will look for any end-to-end delay spikes and any lasting steady state effects.

These first two cases will allow us to make some conclusions about the MiWi P2P protocol as well, being that it is also based on the IEEE 802.15.4 standard, and that the ZigBee model in OPNET 14.0 turns out not to be able to support Beaconing, which makes the major difference between ZigBee and MiWi P2P protocols, when no routing is involved.

Third, we will examine a scenario where we incrementally add routers between a single End Device and a Coordinator, examining increases in end-to-end delay caused by extra hops.

2. Overview of PAN Technologies

The three primary protocols available on the market today are the IEEE 802.15.4 based ZigBee and MiWi protocols, and the ISO Dash7 standard. While it was found that we could not ultimately finish building a model of the MiWi protocol for use in OPNET 14.0, we have determined that modeling and simulating based on the ZigBee protocol would provide enough insight into the IEEE 802.15.4 standard in general for us to draw some pertinent conclusions, and that while we could not hope to model and simulate the Dash7 protocol while remaining within the scope of this course, we find later that we can draw some conclusions about the necessity of using Dash7 over an older 802.15.4 based protocol.

A brief comparison of Dash7, ZigBee, and MiWi is shown below.

	Underlying Standard	**Frequencies Used**	**Penetrates Water**	**Penetrates Concrete**	**Range**	**Average Power Draw**	**Average Latency**	**Multi-Hop Capabilities**	**Interference from 802.11n**	**Maximum Bitrate**
Dash7	ISO 18000-7	433 MHz	Yes	Yes	1 km	30-60 μW	2.5-5 s	Yes	No	200 Kbps

ZigBee	IEEE 802.15.4	2.4 GHz, 915 MHz, 868 MHz	No	No	30–500 m	125-400 µW	Varies from seconds to minutes	Yes	Yes	250 Kbps
MiWi P2P	IEEE 802.15.4	2.4 GHz	No	No	30-500 m	95-115 µW	Varies from seconds to minutes	No	Yes	250 Kbps

While Dash7 offers better penetration of water and concrete, and offers much better range due to use of the 433 MHz band which features a longer wavelength and non-interference with WiFi, it is a completely different protocol that is not nearly as well tested and proven as the 802.15.4 standard based protocols, and would require sourcing of more expensive, unproven RF transceivers. Note that it is capable of multiple hops via routing, but it is almost never employed in a home or small commercial setting because of the much longer range between nodes.[4]

ZigBee offers the advantage of having the greatest market penetration, and therefore most widely available stacks for use in a variety of systems, but in comparison to MiWi, is much bigger and more complex than may be necessary for a low-complexity home and commercial use sensor network. Note that it is capable of multiple hops via routing.[3]

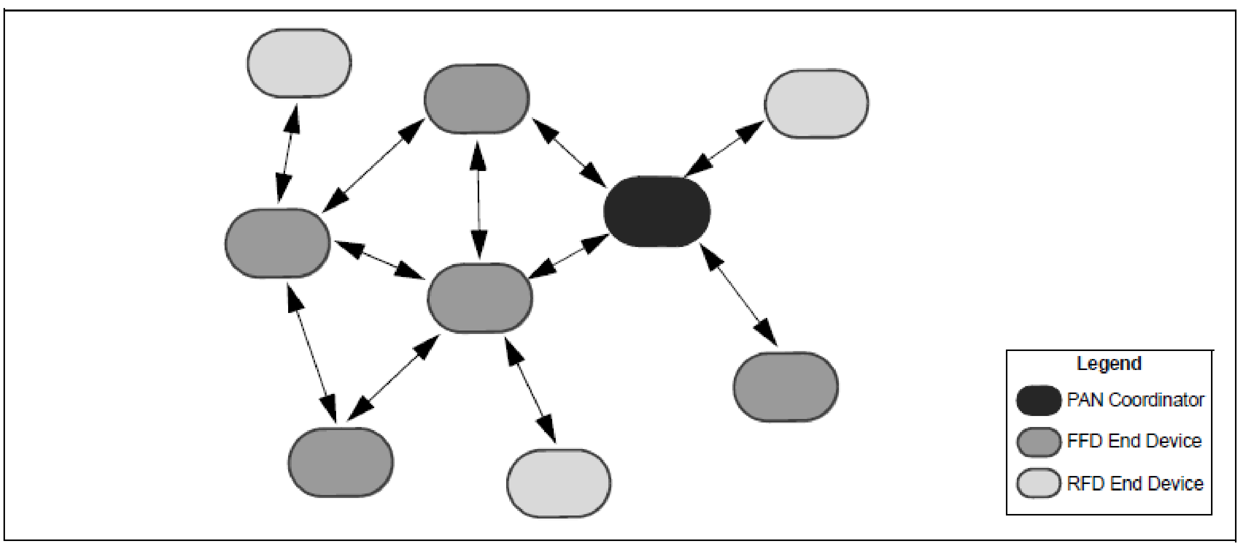

Figure 1 - Extent of Network Topology Capability for Dash7 and ZigBee

As mentioned above, MiWi offers the advantage of being much less complex and therefore offers a much smaller stack size, about 80% smaller. This allows much more room to be used in a microcontroller's ROM for the main program. Note that it is not capable of multiple hops via routing.[5]

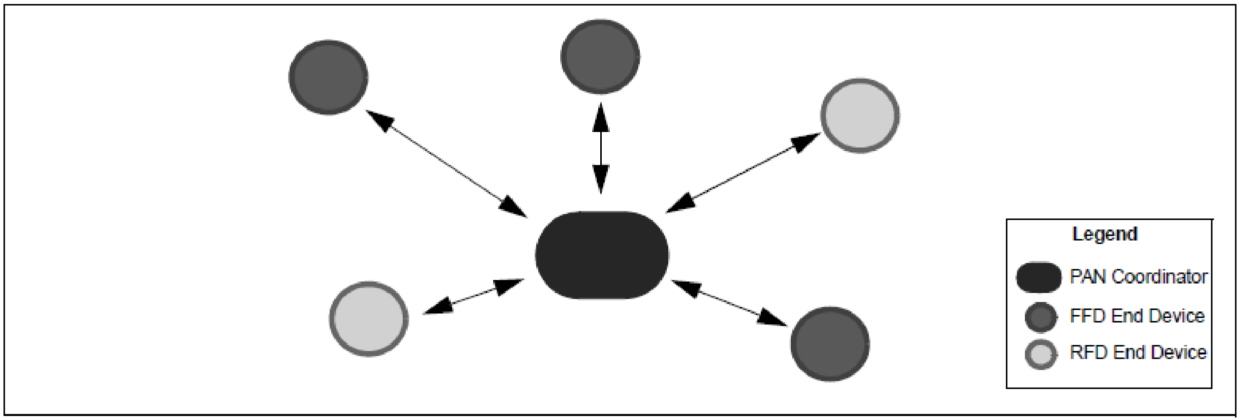

Figure 2 – Extent of MiWi P2P Network Topology Capability for MiWi P2P

In the project discussion to follow, we will determine network performance, stability and end-to-end delay under different circumstances, and finally whether the advantage of Dash7's being able to minimize number of hops (in most cases, will reduce to one single hop) is a significant concern for a small scale, low complexity home and commercial use sensor network.

3. Case Study Modeling and Simulations

This intent of this project is to compare and contrast the performance of the three dominant PAN technologies in different scenarios. However, as a result of the project being necessarily limited in scope, we employ solely ZigBee libraries provided in OPNET 14.0 as a basis for modeling and conducting simulations, and hold our discussion based on the results that follow from these simulations.

The secondary goal of this project is to evaluate and document the shortcomings of the incomplete OPNET 14.0 implementation of the ZigBee models for future students who may wish to conduct further research on ZigBee.

In the sections to follow, the three case studies conducted in this project, their evolving design, and their results are discussed in detail.

3.1 Increasing the Number of Devices in a PAN, and the Effect on Existing Devices

This first case focuses on examining the effect of an incrementally increasing number of ZigBee End Devices on an existing End Device in a PAN, on the sustainable data transfer rate, latency (end-to-end delay of packet transfer from End Device to Coordinator), and occurrences of dropped data due to contention between devices.

In the initial baseline scenario as shown below, the network model is comprised of an End Device (end_device0) and the Coordinator (coordinator0). The coordinator is sending data at a rate of 1300bps, with a constant arrival pattern.

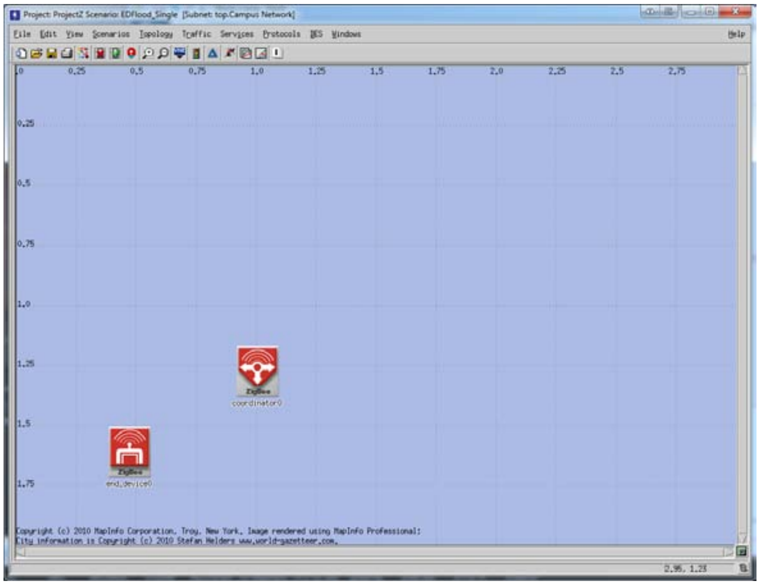

Figure 3 - Case 1 – Baseline Scenario

Next, devices were added to the PAN. They are added via an intermediary Router node (router0). Because the focus is on the end-to-end delay of end_device0, the original End Device on the PAN, we are not concerned with the fact that adding a hop adds end-to-end delay for the newly joined devices. Figures 2 through 9 as follow, show these new additions. They are respectively scenarios representing two added devices, seven, eight, nine, fourteen, fifteen, sixteen, and seventeen. The reason for the choice in number of added nodes will be clarified later.

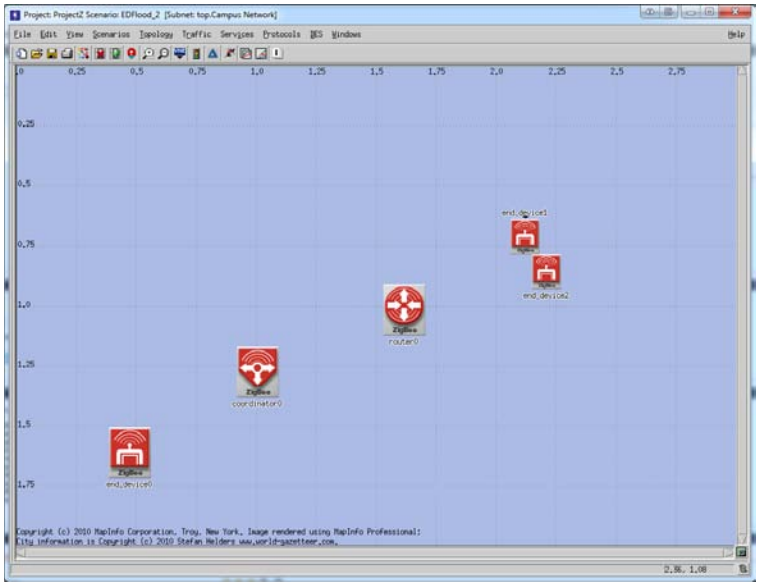

Figure 4 - Case 1 - Two Added End Devices

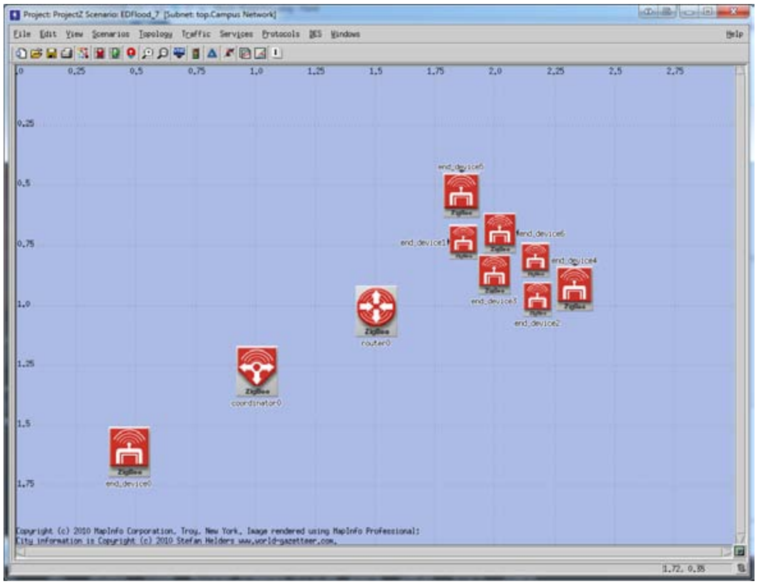

Figure 5 - Case 1 - Seven Added End Devices

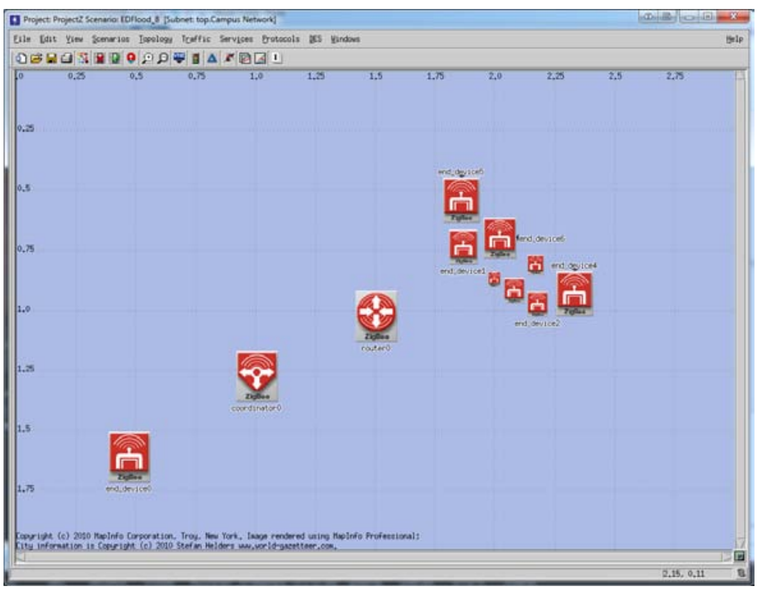

Figure 6 - Case 1 - Eight Added End Devices

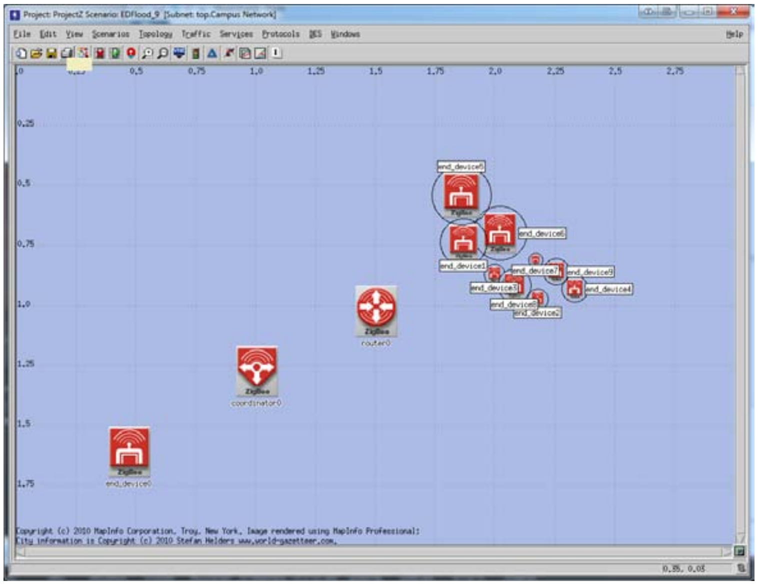

Figure 7 - Case 1 - Nine Added End Devices

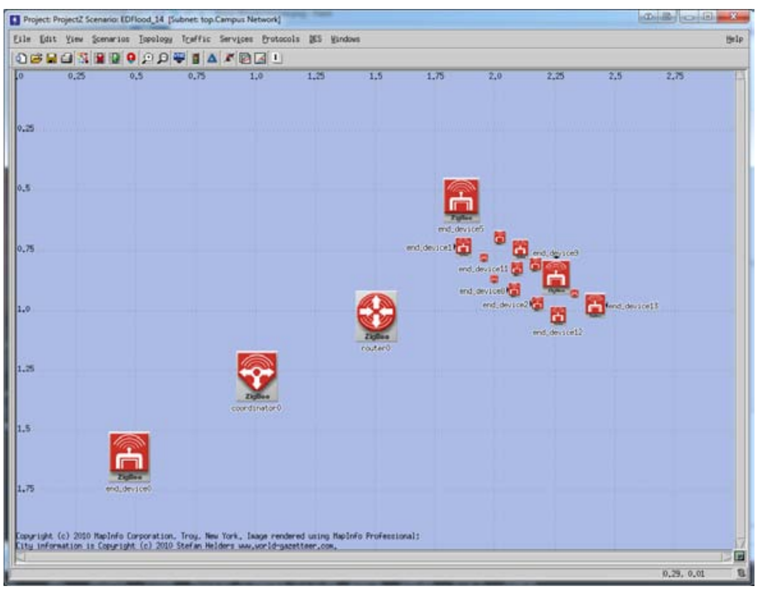

Figure 8 - Case 1 - Fourteen Added End Devices

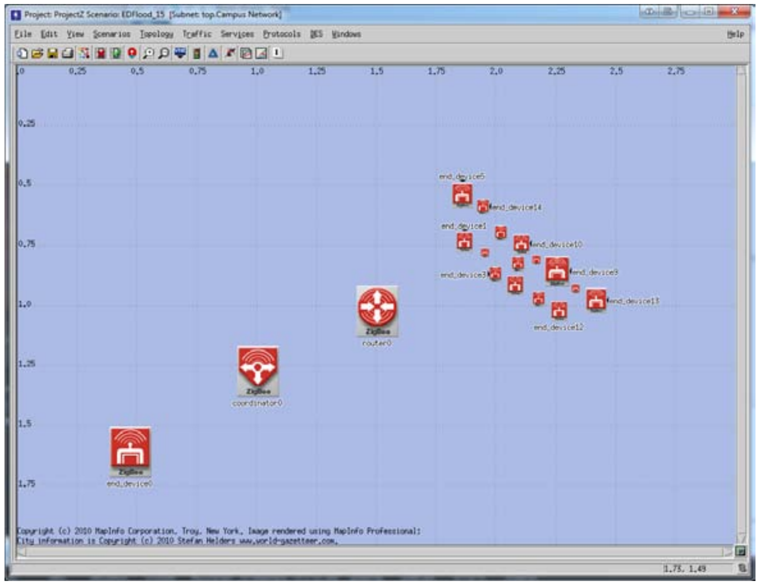

Figure 9 - Case 1 - Fifteen Added End Devices

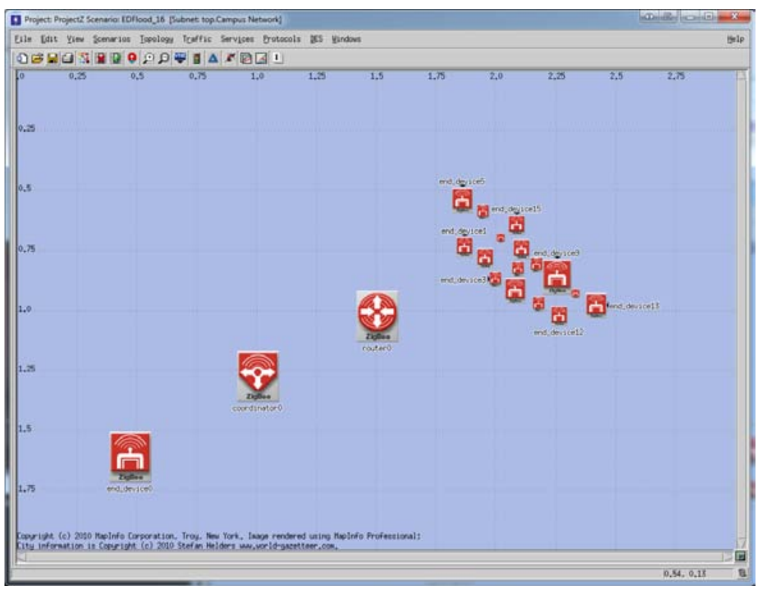

Figure 10 - Case 1 - Sixteen Added End Devices

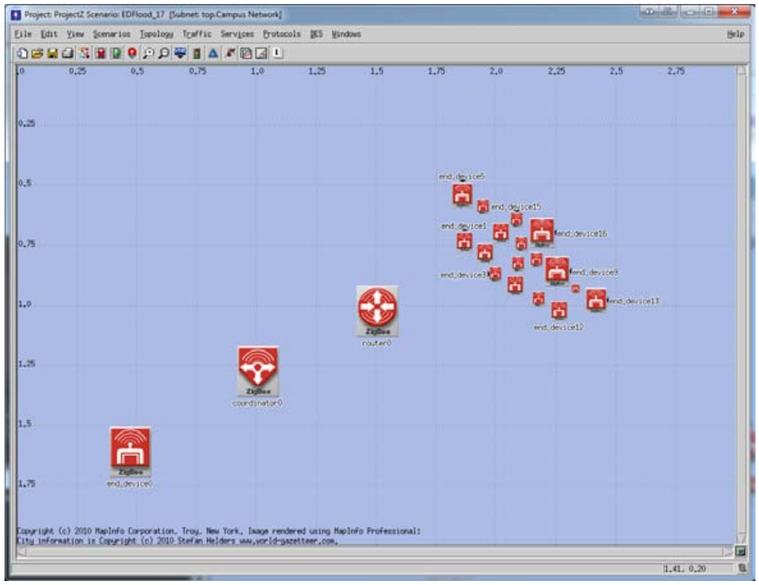

Figure 11 - Case 1 - Seventeen Added End Devices

These scenarios were then simulated. The first statistic to be gathered for this collection of scenarios was Received Data at the Coordinator (coordinator0) node. Figure 10 below shows this result.

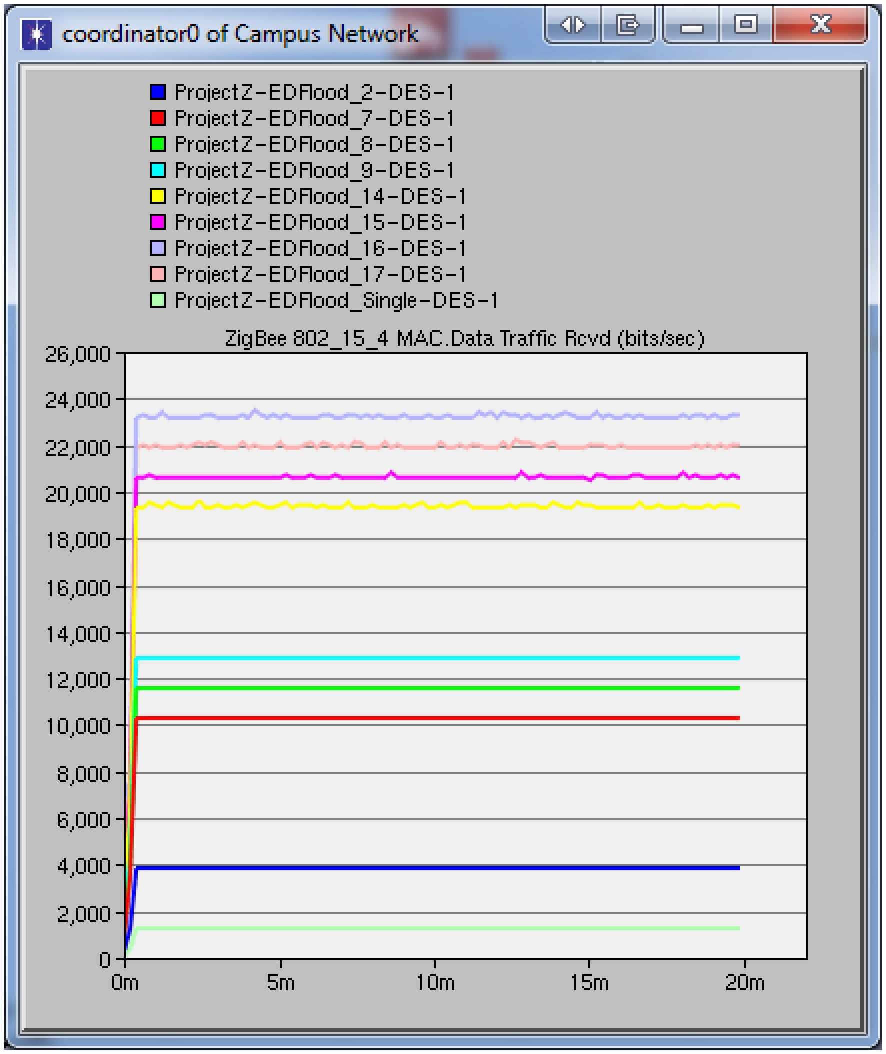

Figure 12 - Case 1 - Received Data at Coordinator

In Figure 10 above, the captions show the number of added End Devices to the PAN. For example, EDFlood_7 (red) indicates the scenario where seven End Devices are added to the PAN, for a total of eight including the original End Device positioned adjacent to the Coordinator.

The data sent by each End Device to the Coordinator, either directly or via the Router, was a constant stream of 1300 bits per packet, per second. In each progressing scenario, the amount of data received by the Coordinator equals exactly the number of nodes x 1300bps. This trend continues until EDFlood_17 (pink), where suddenly, the Data Traffic Received by the Coordinator is even less than in EDFlood_16 (light blue).

The reason for this is that there are only 16 available channels under 802.15.4 in the standard international 2.4GHz ISM (Industrial, Scientific, and Medical) band. Unfortunately, this is where a shortcoming of the ZigBee library that comes with OPNET 14.0 reveals itself: Beaconing Mode, and the Guaranteed Time Slots provided by it [2] is not available for us to experiment with, and to explore whether or not 16 really is the effective hard limit for number of direct children.

The number of added End Devices is specifically chosen to end with 15, 16, and 17 in order to close in on and confirm 16 as the number of simultaneously connected and communicating children.

Figure 11 below shows the raw End-to-End Delay of data packets being sent from the original end_device0 to coordinator0. Please see Figure 12 for the averaged results, a more understandable format.

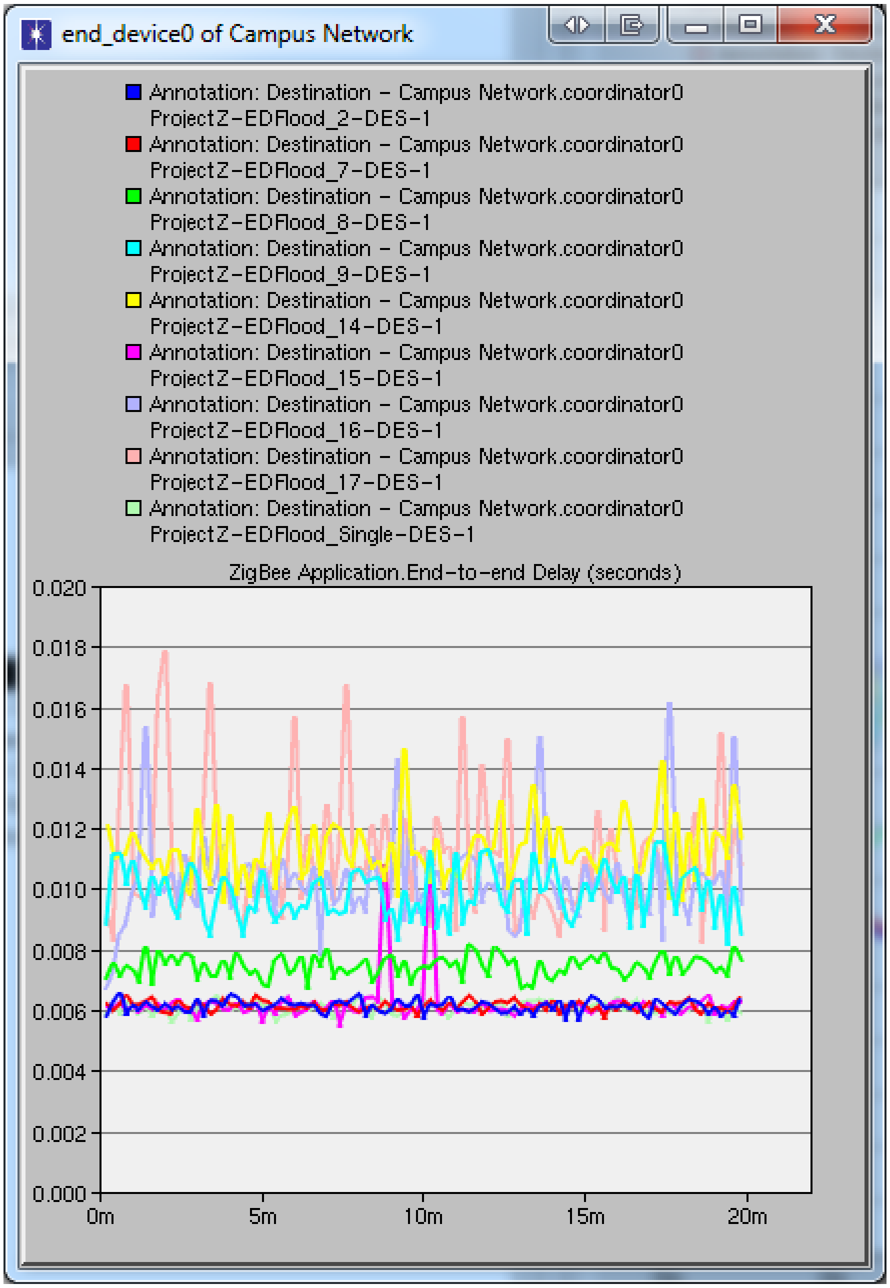

Figure 13 - Case 1 - End-to-End Delay

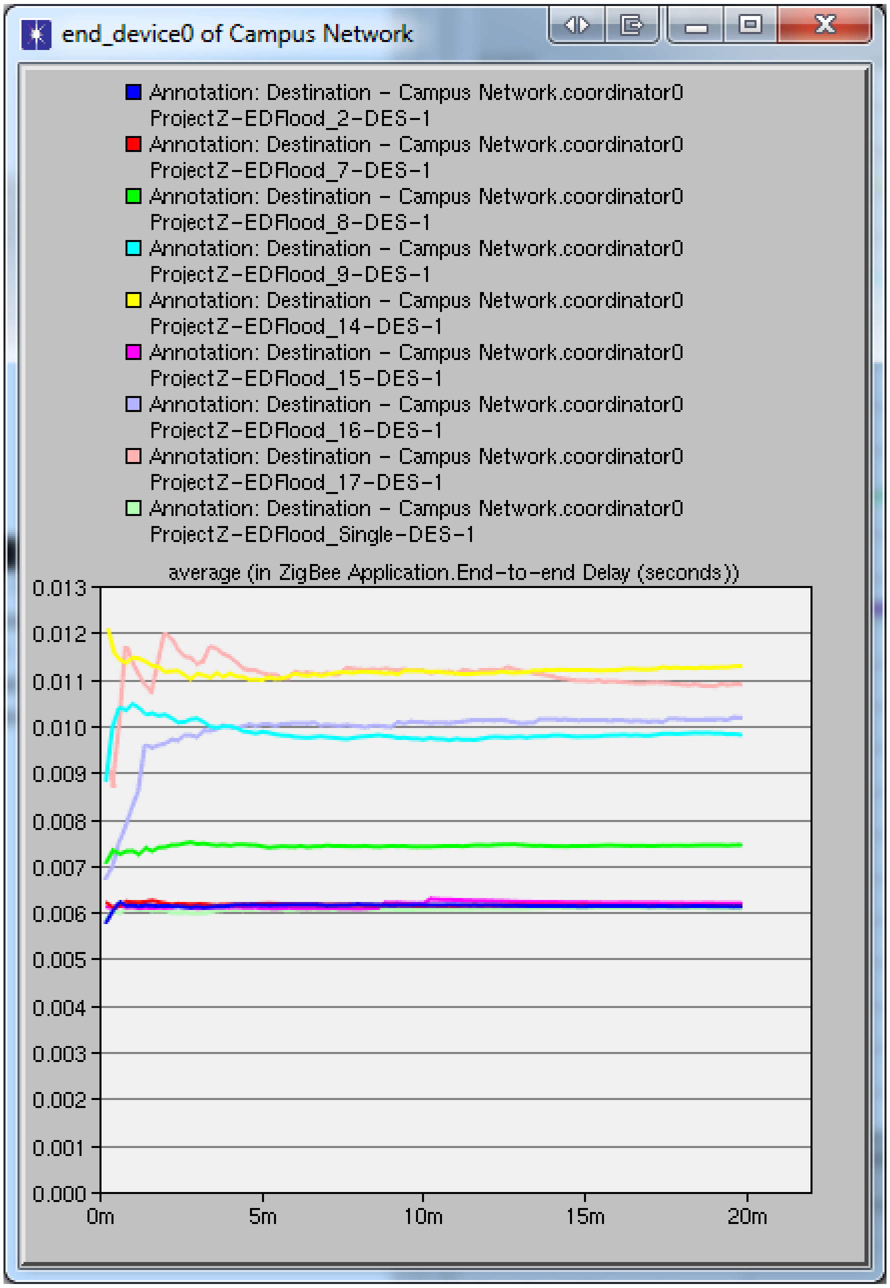

Figure 14 - Case 1 - Average End-to-End Delay

Referring to Figure 12 as precedes, we can see that the end-to-end delay between end_device0 and coordinator0 holds steady at 6ms for all scenarios up until EDFlood_8 (green), when there are eight added End Devices on the PAN. This new latency is seen to be 7.5ms. However, when we increment the number of added devices by just one more, to EDFlood_9 (teal), the end-to-end delay shoots up to a new high. The steady-state latency is 9.9ms, an astonishing increase.

The reason for this onset of latency was a mystery, and so an exploration of the issue was in order. Two new scenarios are now added for our exploration: we wish to find out if the sudden increase in end-to-end delay is caused by overwhelming the Coordinator with too much incoming bandwidth, or if it is perhaps due to channel contention. These two scenarios then, naturally, entail A)decreasing sent data traffic, and B)increasing number of available channels. In order to accomplish A, we simply halve our data packet size from 1300 bits per packet, to 650 bits per packet. In order to accomplish B, we open up available channels to include not just 2.4GHz (International), but 915MHz (USA/Australia-specific), and 868MHz (Europe-specific) as well.

Figures 13 and 14 below illustrate our new exploratory scenarios.

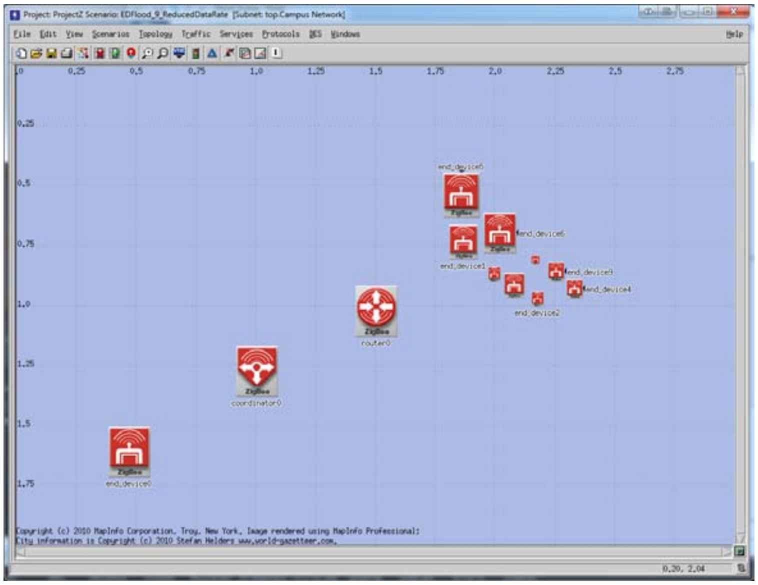

Figure 15 - Case 1 - 9 Added End Devices, Reduced Data Rate

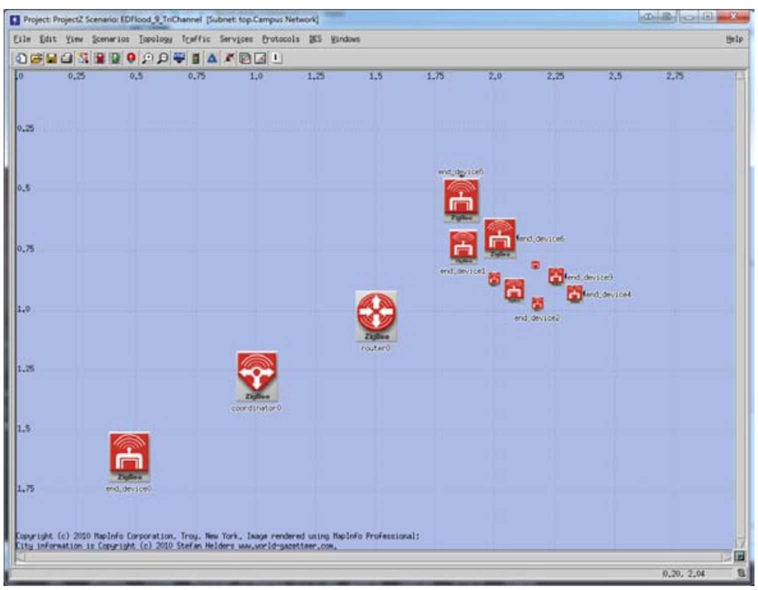

Figure 16 - Case 1 - 9 Added End Devices, Extra Bands

The simulation results follow below.

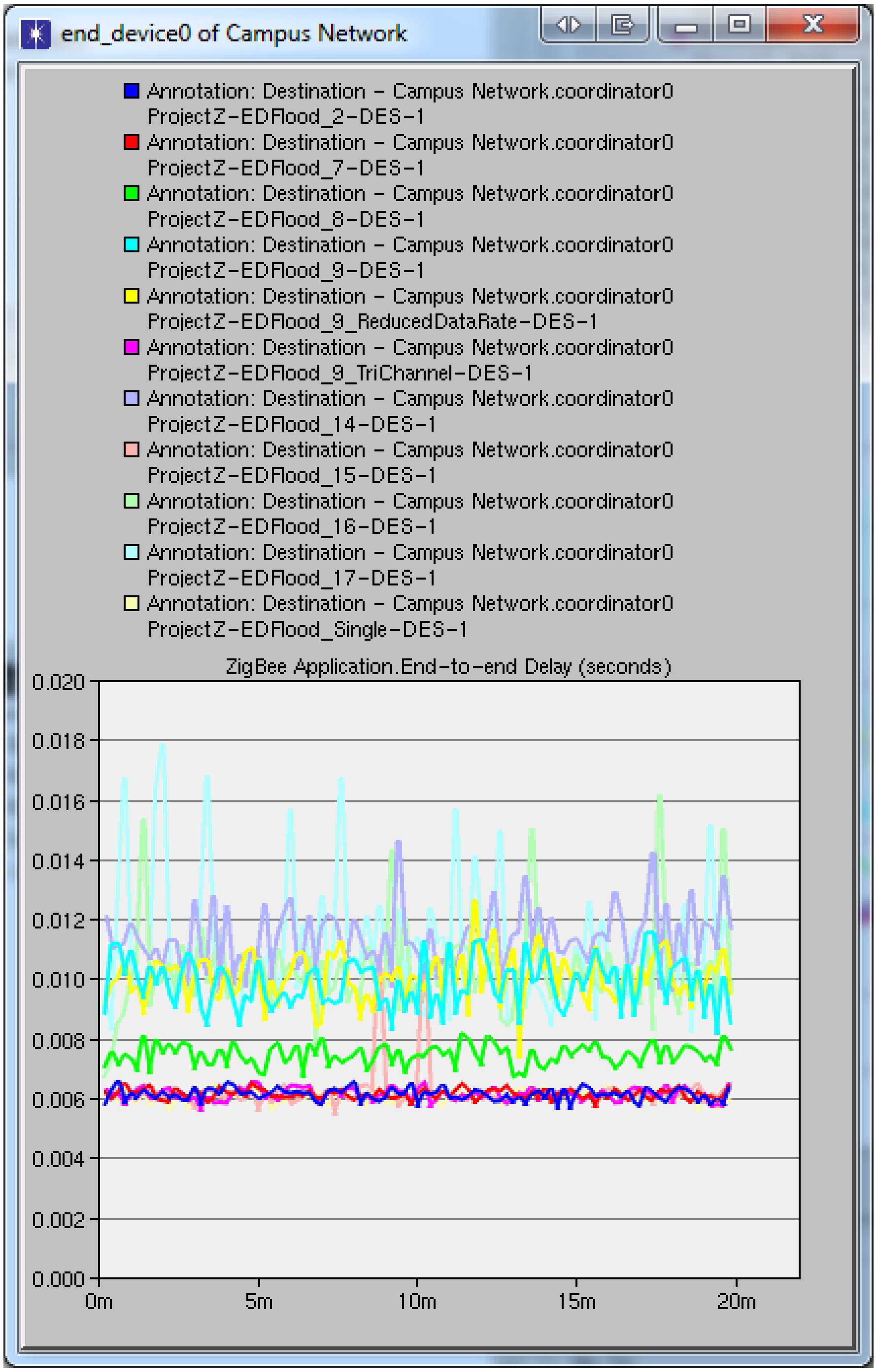

Figure 17 - Case 1 - New End-to-End Delay Results

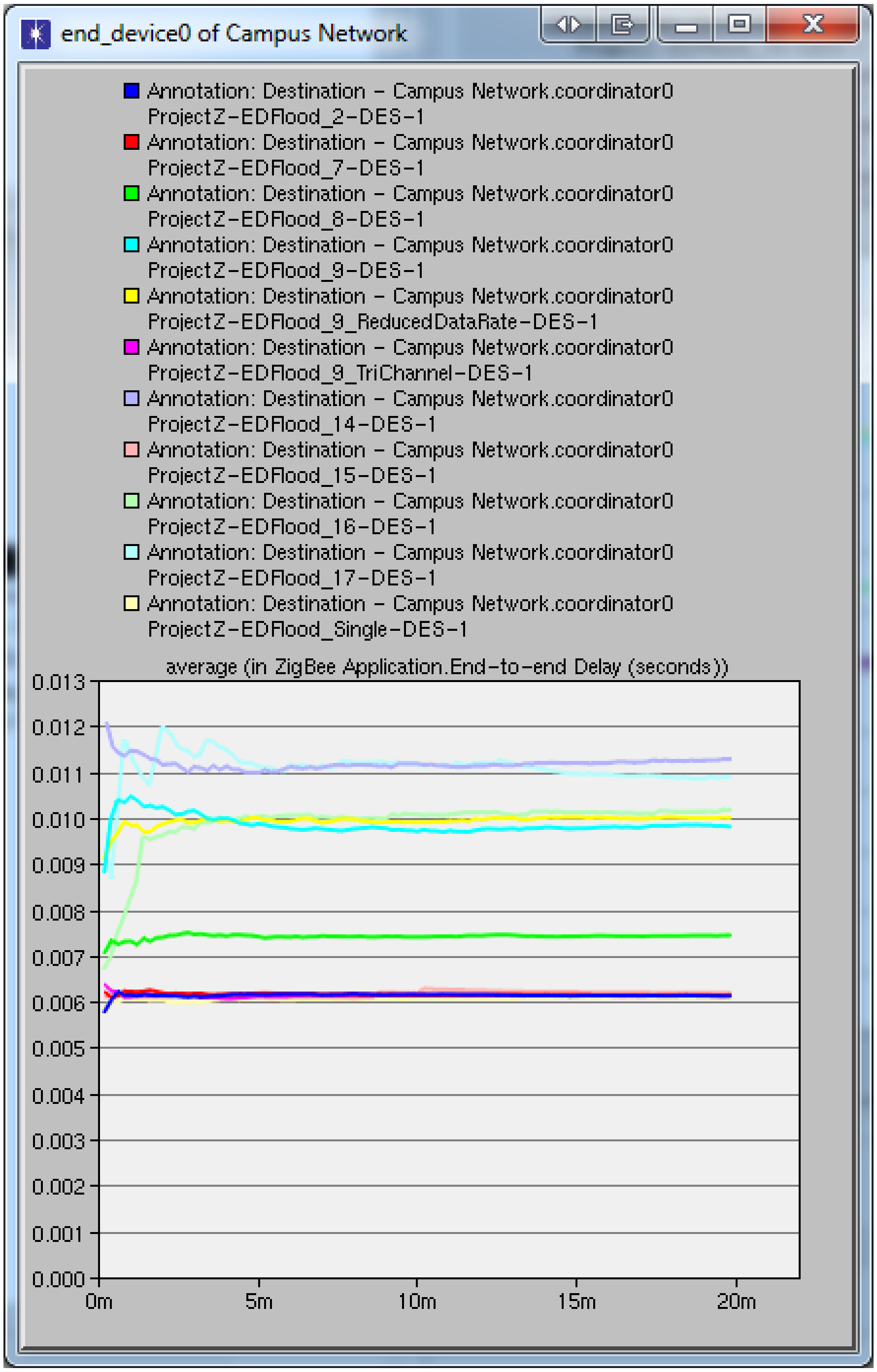

Figure 18 - Case 1 - New Average End-to-End Delay Results

From Figure 16, we can see that the average end-to-end delay for the A) scenario (yellow), where data rate was decreased from 1300bps to 650bps, has not changed. This was expected, as the specified maximum supported transfer rate is 250kbps [1], far exceeding that in any of our scenarios.

Next, we examine the average end-to-end delay for the B) scenario (purple), where two additional bands are introduced, adding 11 total additional channels. We can see here that the end-to-end delay is 6ms, effectively reducing delay right back to level seen before the eight device was added.

We are now convinced that the increase in end-to-end latency was introduced by an effect known as co-channel interference [2]. This is a very important observation, and is a prime example of how simulation of networks can catch issues that may not be obvious for some time after a commercial deployment.

This concludes Case 1, and we have confirmed that for 802.15.4-based protocols, 16 is the maximum number of channels available on the Worldwide 2.4 GHz band, and have discovered that 8 is the maximum number of usable channels without incurring co-channel interference, if we are restricted to the 16 channels available on the 2.4 GHz band.

3.2 Creating a Sudden Flood of New Devices to a PAN, and the Effect on Existing Devices

The second case revisits the first, this time adding to the scenario a sudden flood of new end devices. This is achieved by assigning the intermediary router a movement trajectory, such that only after a period of time is it both in range of the end devices and the coordinator. This allows us to study the effect of simultaneous joining of new devices on end-to-end delay of packets sent from an end device directly adjacent to the coordinator, to the coordinator. The significance of this case is in the evaluation of the ability of 802.15.4 based network technologies to cope with simultaneous joining of mass quantities of new end devices, a critical corner case in sensor network design.

The following figures show the implemented scenarios.

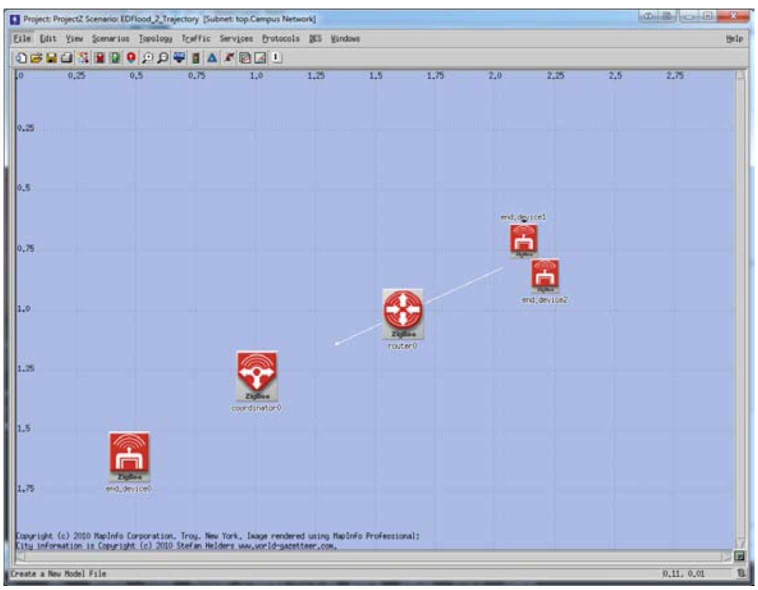

Figure 19 - Case 2 - Two Added End Devices, Router on Trajectory

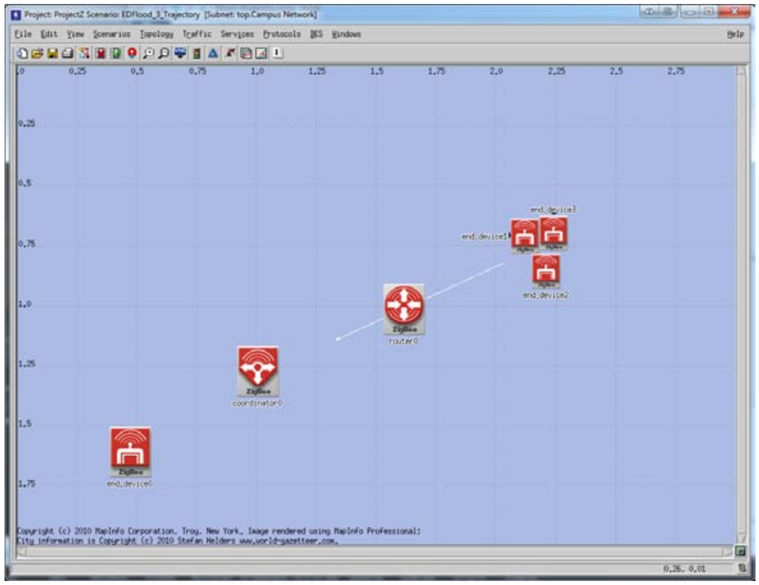

Figure 20 - Case 3 - Three Added End Devices, Router on Trajectory

Unfortunately, when we reached the above scenario where three devices are added, we made a discovery. The ZigBee library implemented in OPNET 14.0 does not support more than two devices per router. This will be discussed in more detail later.

For comparison purposes, we included a reference scenario in this case, borrowed from Case 1: this is the scenario of two End Devices being added, though the router does not carry a trajectory.

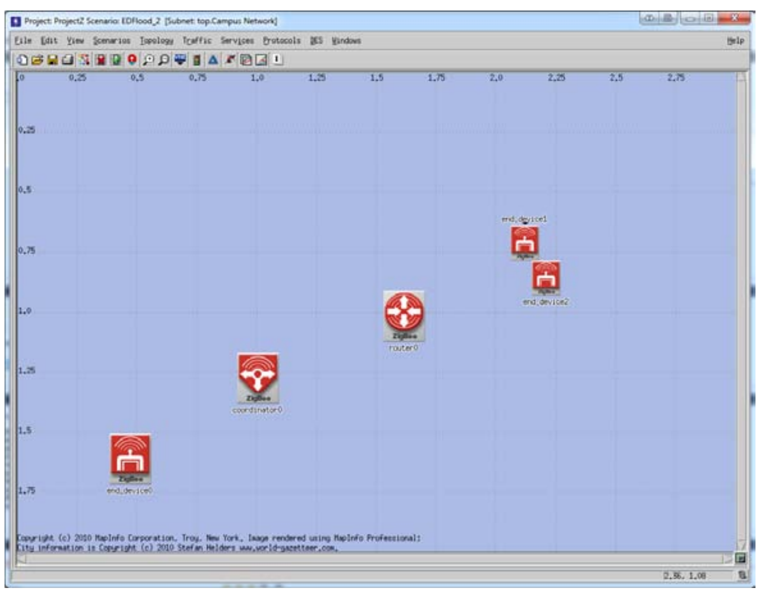

Figure 21 - Case 2 - Reference: Two End Devices Added, No Router Movement Trajectory

Simulations on received data rate at coordinator0, and the results are as follows:

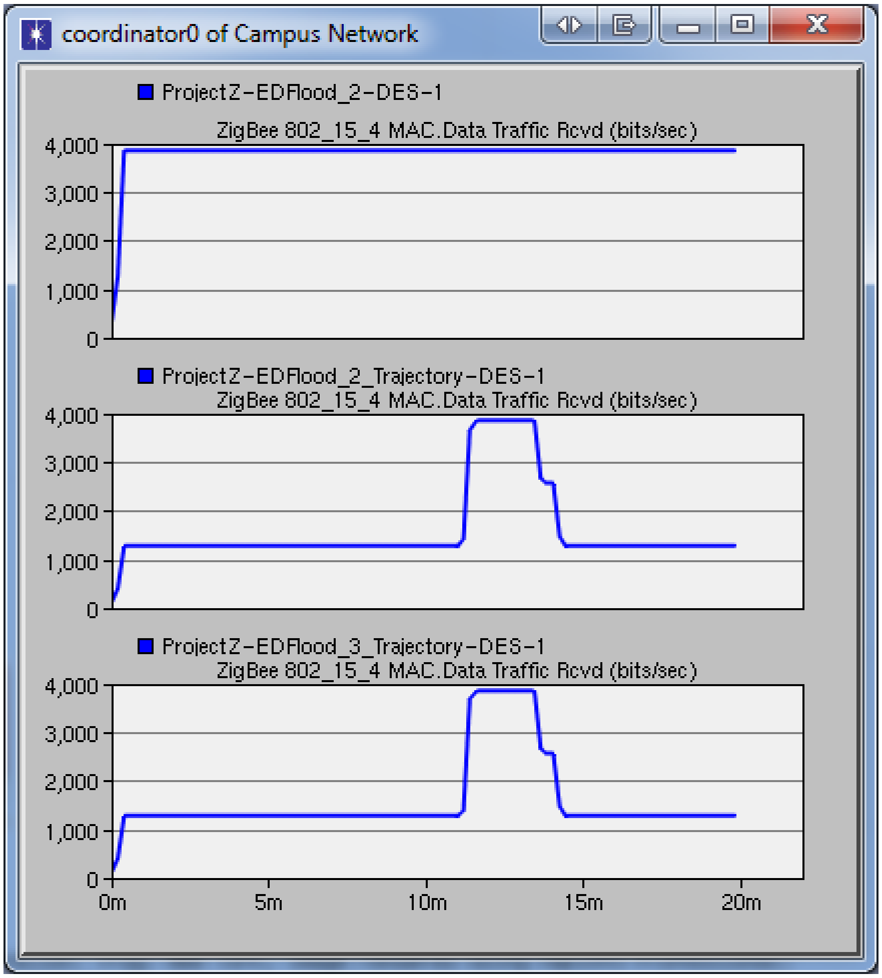

Figure 22 - Case 2 - Received Data at Coordinator0

From Figure 20, we can see that the received data rate at the coordinator for the scenario with no trajectory on the router – EDFlood_2 (top) shows the expected 3900 bps, from beginning to end. The scenario with the movement trajectory assigned to the router – EDFlood_2_Trajectory (middle), shows 1300 bps to begin – as expected, given that there is perpetually one End Device connected to the Coordinator which we are observing, then rapidly ramps to 3900 bps as the router enters within range of both the coordinator and the perimeter End Device nodes, and ramps back down to 1300 bps again after it falls out of range again.

The peculiarity is with the scenario where three End Devices are added to the network – EDFlood_3_Trajectory (bottom). We can see that the received data trace is exactly identical to that of EDFlood_2_Trajectory. Something is wrong here. In order to confirm our suspicions that one End Device is not sending packets all the way to the coordinator successfully at all, we examine the statistic Traffic Dropped en Route (packets) below.

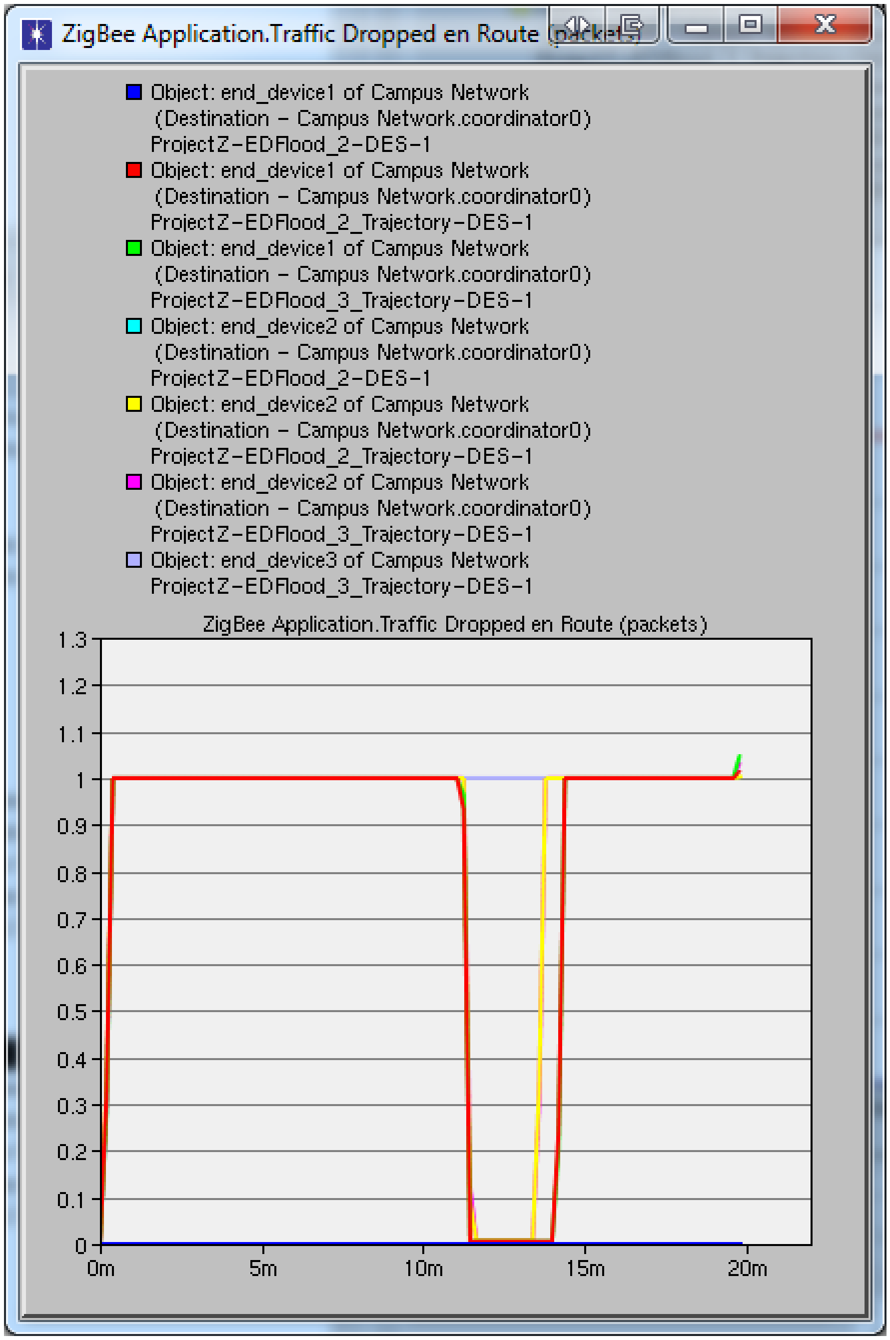

Figure 23 - Case 2 - Dropped Packets

This graph is very telling. As expected, scenario EDFlood_2 (blue, teal) with no trajectory shows no packets dropped from beginning to end. Scenario EDFlood_2_Trajectory (red, yellow) shows all packets being dropped right up until the router comes within range, then no packets dropped while it is in range, then again all packets dropped when the router moves back out of range. For scenario ED_Flood_3_Trajectory (green, purple, light blue), we see that while end_device1 and end_device2 follow exactly as with ED_Flood_2_Trajectory, the third End Device, end_device3 (light blue trace) shows all packets being dropped from beginning to end.

We have found then, that the third End Device never transmits to the coordinator. Further, we remind ourselves that in Case 1 we were able to connect up to 16 devices to one router with absolutely no issues, and certainly never coming across 100% of packets being dropped. We come to the conclusion that the ZigBee library for the 14.0 release of OPNET, does not allow for us to simulate more than two simultaneous connections to a router, IF the router is following a movement trajectory. If the router is NOT following a movement trajectory, the library imposes no such restrictions. This is definitely a major limitation. This should be made known to future users of OPNET 14.0 intending to carry out an evaluative project on ZigBee.

Finishing up with our analysis of Case 2, we examine end-to-end delay between end_device0 and the coordinator.

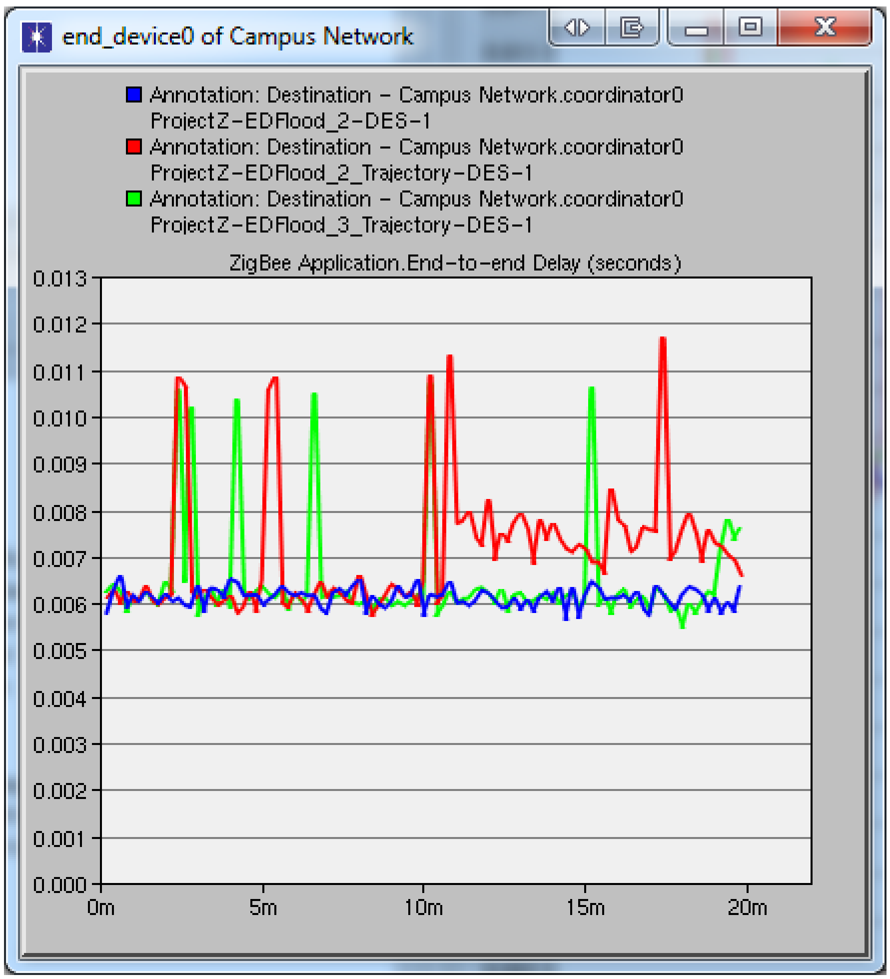

Figure 24 - Case 2 - End-to-End Delay Between end_device0 and coordinator0

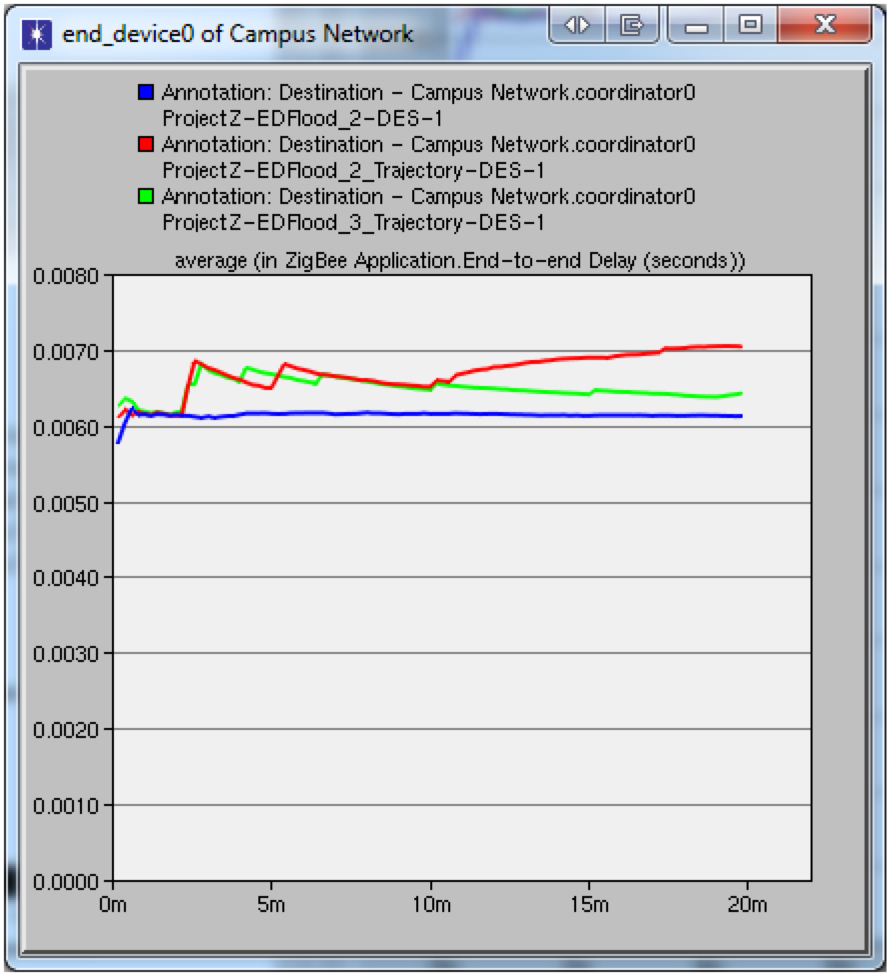

Figure 25 - Case 2 - End-to-End Delay Between end_device0 and coordinator0, Average

We immediately find another issue with the ZigBee library in OPNET 14.0. The spikes in end-to-end delay should not be occurring before 11m, when the router moves within range. The end-to-end delay spikes before 11m do not make any sense, because neither end_device0 nor coordinator0 has any knowledge of the Router node, nor the other End Device nodes until they come within range at 11m. There is the same issue with end-to-end delay spikes after 14m, because the router has moved well out of range by this point, and has therefore exited the PAN. Because of the severe interference from faulty simulation results, we cannot draw a conclusion regarding end-to-end delay between the original end_device0 and coordinator0 as a affected by suddenly flooded new end devices on the PAN. This is a severe limitation of the ZigBee library as of OPNET version 14.0.

3.3 Adding Hops to the Network Path, and the Effect on End-To-End Delay

The third scenario focuses on observing the effects of adding hops to a data path on the end-to-end delay of packet transfer from an End Device to the Coordinator. An advantage of the competing ISO Dash7 standard over the IEEE 802.15.4 standard is in its ability to eliminate most, if not all hops between devices in a sensor network. The results of this scenario allow us to evaluate the significance of this advantage in practice.

The following figures show our implemented scenarios for Case 3. Each scenario is comprised of one Coordinator node at the far left, an End Device node at the right right, and number of routers in between, ranging from none to four.

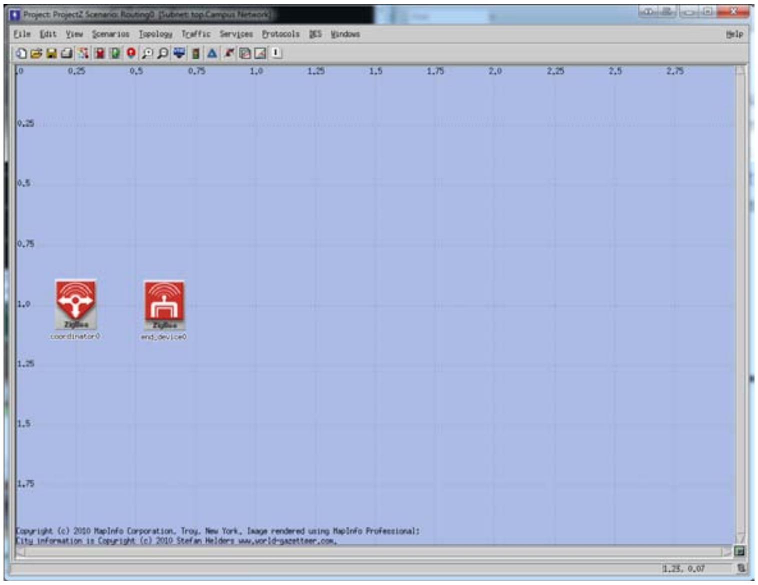

Figure 26 - Case 3 – One Hop; Direct Link

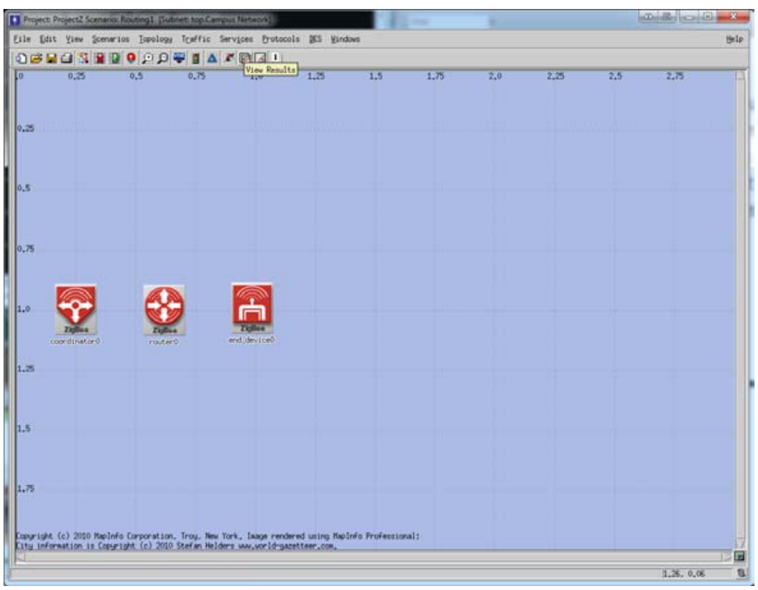

Figure 27 - Case 3 – Two Hops

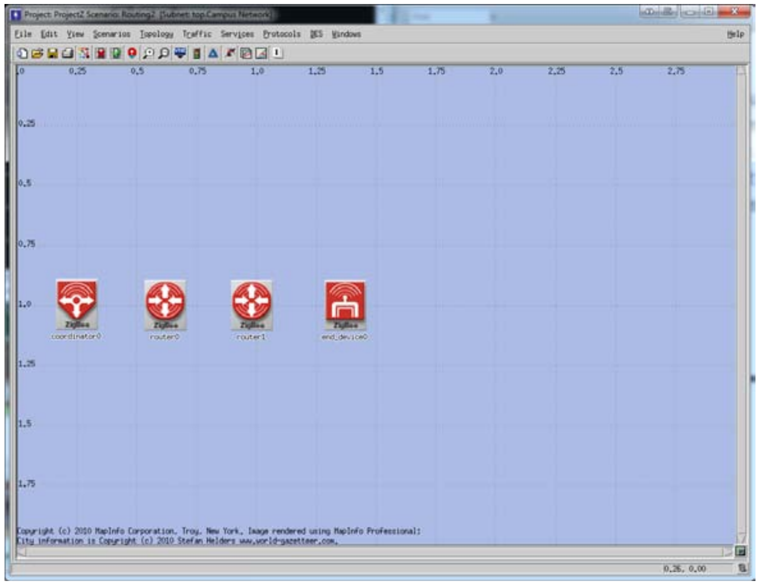

Figure 28 - Case 3 - Three Hops

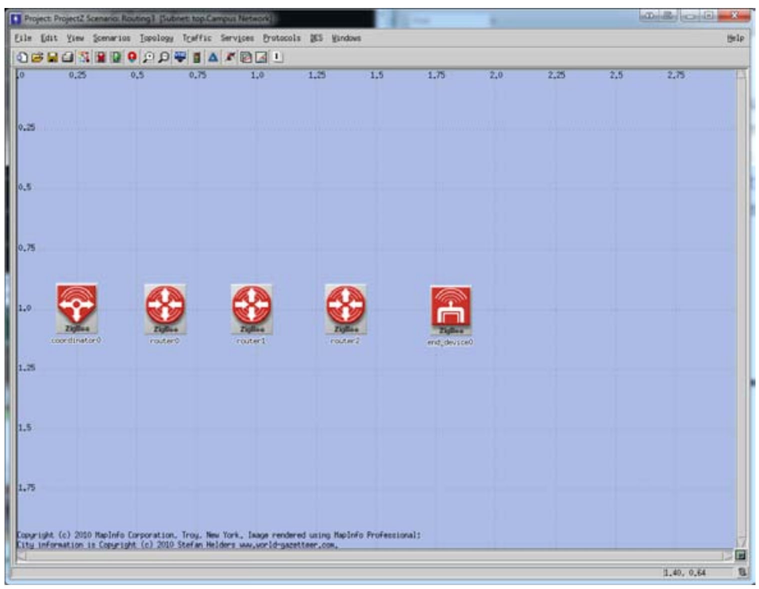

Figure 29 - Case 3 – Four Hops

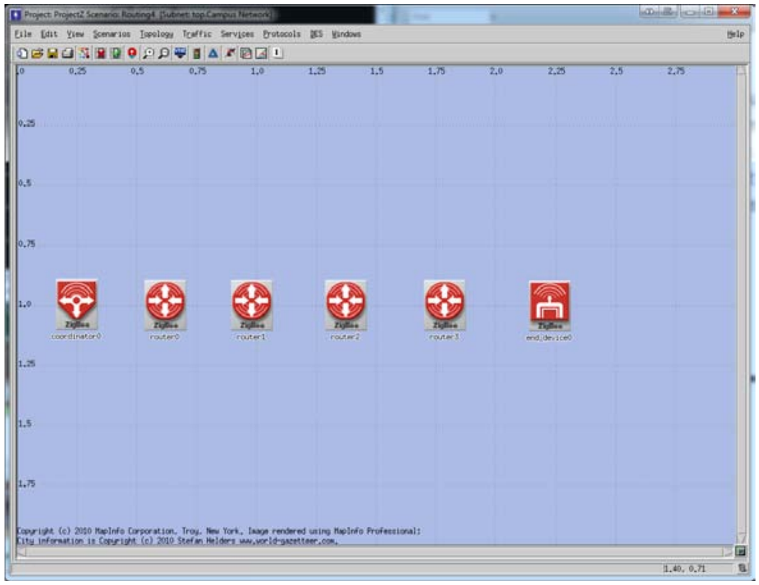

Figure 30 - Case 3 – Five Hops

We are interested in examining end-to-end delay between End Device and Coordinator nodes in Case 3, and so we run the simulation, and the results are as follows.

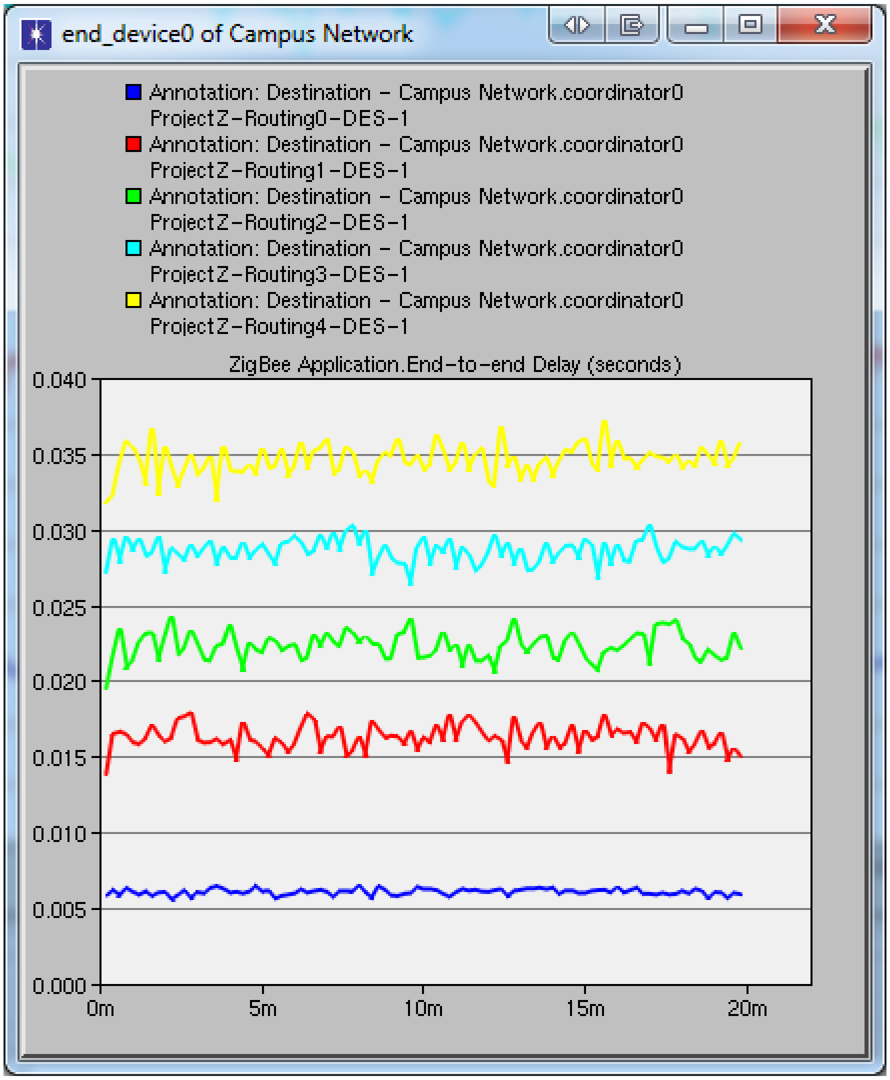

Figure 31 - Case 3 - Routing Delay Between end_device0 and coordinator0

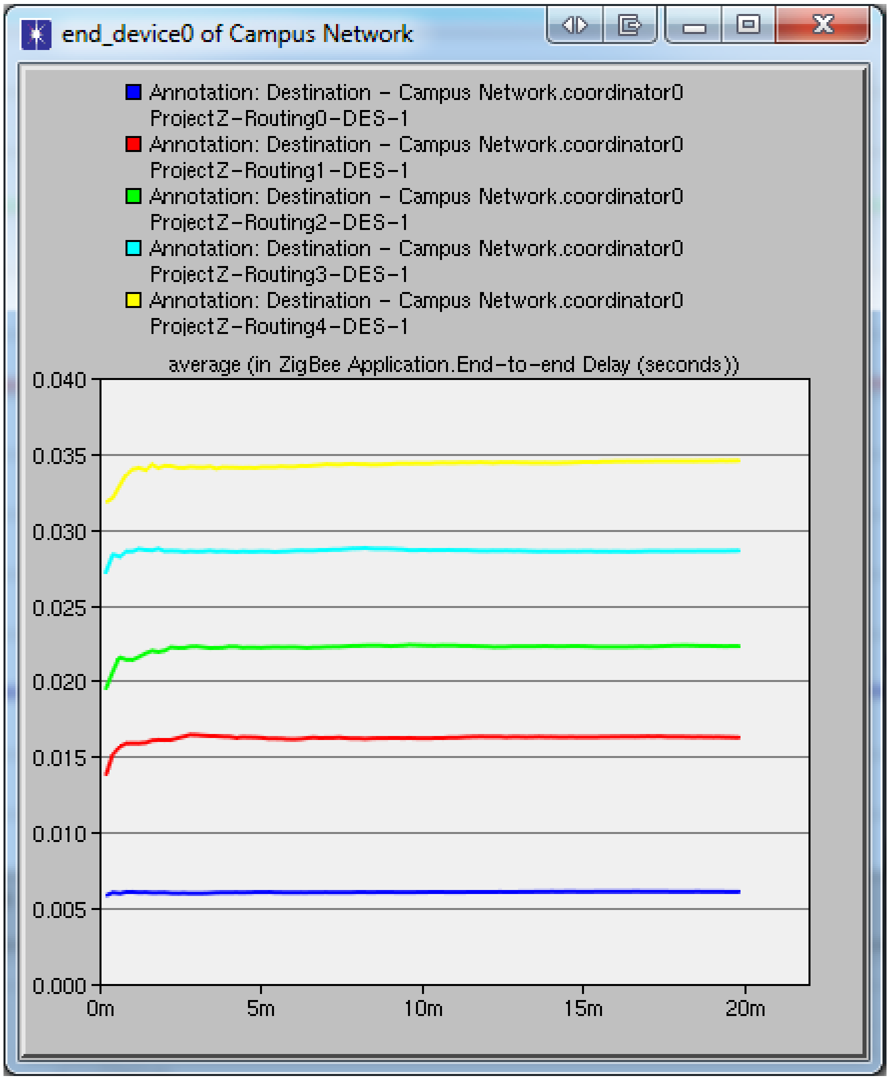

Figure 32 - Case 3 - Routing Delay Between end_device0 and coordinator0, Average

From Figure 30 above, we see that the base end-to-end delay with no intermediary routers is 6ms. The addition of one router in between shows an increase to 16ms. Two routers being added, forcing three hops, shows a delay of 22ms. Four hops shows 28ms. Five hops shows 34ms. The results are summarized below in Table 1.

Table 1 – End-to-End Delay Based on Number of Hops

Scenario	Number of Hops	Total End-to-End Delay
Routing0	1	6ms
Routing1	2	16ms
Routing2	3	22ms
Routing3	4	28ms
Routing4	5	34ms

It becomes clear that incrementally increasing number of hops introduces a linear increase of 6ms end-to-end delay. Between Routing0 and Routing1, we see that introducing a router introduces a full 10ms of end-to-end delay. This appears to be a fixed overhead delay that comes from the addition of routing functionality to a network.

Since this is a minimal increase in delay, and one does not often see more than five hops in a sensor network used in a non-industrial setting, we conclude that the major advantage of the ISO Dash7 standard of not needing any hops between nodes is a trivial one for non-industrial use.

4. Discussion and Conclusions

4.1 Case Study Conclusions, and Limitations in OPNET 14.0 ZigBee Implementation

Through the three case studies carried out for this project, several key discoveries were made.

For Case 1, we found that the expected end-to-end delay for a ZigBee PAN is 6ms, and as long as there is no co-channel interference (no more eight devices on the network), in which case the end-to-end delay increases to 7.5ms, and then to 9.9ms with nine added devices. Once 16 devices are reached, channel contention becomes a factor, and while end-to-end latency does not increase for other devices, the newly added device can almost never transmit. This is if Beaconing is disabled, which in turn disables Guaranteed Time Slots. This is a major limitation for the OPNET 14.0 ZigBee library model.

For Case 2, we found that there is a major limitation with the OPNET 14.0 ZigBee library in that it cannot handle more than two devices per router if the router is on a movement trajectory, and that there is an issue with latency simulation when the router is on a movement trajectory. This was discovered in a project by S. Leung, W. Gomez, and J. J. Kim in their Spring 2009 ENSC 427 Final Project [6], but they came to the false conclusion that it applied to all cases of a router with children. Their conclusion was that any router, with or without an assigned movement trajectory, could not communicate with more than two children. This was proved to be false in this project.

For Case 3, we found that the expected end-to-end delay for a ZigBee PAN is 6ms with no hops involved (as corroborated by Case 1 simulation results). Adding one hop increases delay by 10ms, and each concurrent added hop increases delay linearly by a further 6ms each. The introduction of a 10ms end-to-end delay by the first hop indicates that there there is a 4ms overhead delay involved in introducing routing to a network.

4.2 What Was Learned

The intent of this project was to evaluate ZigBee as a protocol to use with small-scale non-industrial sensor networks for home and commercial use. By designing the case studies, implementing network models, and running multiple simulations, a much deeper, intimate understanding of the ZigBee protocol itself was gained.

In addition, carrying out this project helped for gaining familiarity with OPNET, and network modeling and simulation software in general as a result. This will be invaluable for future pursuits, whether in a career in network modeling and simulation, or for verification of effectiveness device design for use in sensor networks, as was the case for this project.

The results of this project apply theoretically to the Networked Faucet System project for ENSC 440, completed successfully in Spring of 2010, and being continued into the future as a potential commercial product. Though we use MiWi P2P as the protocol for device networking, we gain valuable insight into the overall nuances of the parent 802.15.4 standard, such as issues of co-channel interference on the 2.4GHz band. Case 3 also allows us to understand that while the new ISO Dash7 standard shows much promise, it is not an immediately necessary upgrade, if at all, given that a major advantage is not significant for non-industrial use.

4.3 Other Comments

Included in the appendix is a note on how to find a MAC layer process model for MiWi P2P, and its accompanying code originally created for an earlier iteration of this project. Ultimately it could not be used, because the ZigBee Network Model source code is not available to the user, and it could not be referenced for help in designing the MiWi P2P Network model. Without a functioning MiWi P2P Network model, a network using the MiWi P2P protocol could not be simulated at all. This was finally determined to be an impassable barrier, and the project in its original form was unfortunately abandoned. The focus has now shifted to ZigBee, which has yielded many fruitful results that are applicable to the original inspiration for this project – the Networked Faucet System project of ENSC 440.

5. References

[1]I.S. Hammoodi, B.G. Stewart, A. Kocian, S.G. McMeekin, "A Comprehensive Performance Study of OPNET Modeler for ZigBee Wireless Sensor Networks Electrical and Computer Engineering", *Third International Conference on Next Generation Mobile Applications, Services and Technologies*, NGMAST '09, November 2009.

[2]A. Mishra, C. Na, D. Rosenburgh, "On Scheduling Guaranteed Time Slots for Time Sensitive Transactions in IEEE 802.15.4 Networks", *Military Communications Conference, 2007,* MILCOM '07, October 2007.

[3] D. Gascon, "802.15.4 vs Zigbee", [Online document], Nov. 2008, [cited June 12th, 2010], Available: http://www.sensor-networks.org/index.php?page=0823123150.

[4]Y. Yang, "Microchip MiWi™ P2P Wireless Protocol", *Microchip Application Notes 2008,* January 2008.

[5]J. P. Norair, "Introduction to DASH7 Technologies", *Dash7 Alliance Low Power RF Technical Overview,* March 2009.

[6]S. Leung, W. Gomez, J. J. Kim, "ZigBee Mesh Network Simulation Using OPNET and Study of Routing Selection", SFU ENSC 427 Project Spring '09, December 2009.

6. Appendix – Note on Models and Code Prepared but Not Used toward Final Project

The MiWi P2P MAC process model with its accompanying code was ultimately not used for this project, so it is sent to you in a separate attachment entitled **miwiwork.zip**. However, it was a large portion of my project work in Spring 2010, and would have been excellent for use in this project for direct comparison of the performance of the MiWi P2P protocol vs the ZigBee protocol.