Locating an object with Received Signal Strength Indication

Calder Lowenthal

Robotics Lab

Thomas Jefferson High School for Science and Technology

2018-2019

Abstract

This research was conducted to explore a means for locating objects in relation to other objects. The Received Signal Strength Indicator (RSSI) measures the strength of a signal. The objective of this research was to construct a formula that gave distance as a function of the RSSI value. After configuring the Digi XBee Pro S1s to act as local and remote radio devices, a range test was performed and RSSI values were recorded with corresponding distances between the two devices. The data confirmed that signal strength decreases as distance increases, and that signal strength is proportional to the reciprocal of the distance squared. An equation giving distance as a function of measured RSSI value was established and is somewhat accurate. The equation can be used as a somewhat reliable method for determining a range of distances between radio devices based on an RSSI reading. The next steps will be to attempt to configure XBees to work in a manner that allows multilateration, and therefore localization of a remote device.

Keywords: Received Signal Strength Indicator, XBee, multilateration

Introduction

Determining location is one of the oldest applications of science. The ancient sailors learned the sky and made maps and astrolabes to navigate their way about the globe. Until recently, most people used paper maps and asked other people for directions. More recently, humans have been looking at their cell phones, which tell them where they are. These mobile mapping services generally do not actually rely on GPS, but rather rely on cell networks and multilateration. We'll get back to this in a moment.

There are several ways to determine the location of an object of interest. Probably the simplest is simply having a visual. If you can see something, then you have more or less found it. For obvious reasons, this isn't the most useful method of finding things depending on the scale of your operation, as it is useful to be able to find objects that are not immediately visible, or in environments where vision is impaired by obstacles or for other reasons. Moving along then, there have been some illuminating studies into how animals such as rats produce mental maps of their surroundings with landmarks, and how robots can imitate this with fairly minimal location devices. Unfortunately, these methods tend to require significant scouting out of the area to produce the map before the robot really knows where it lies in that map. Another is to have a preexisting coordinate system, but once again, this requires knowing the area you are in prior to doing anything. If a system knows where it was and has kept track of its displacement, that is yet another means for determining location. Finally, an object can carry a GPS capable device and simply transmit its position.

The failures of many of these methods is that they either require preexisting knowledge of the area being moved through or a period during which your system looks about and gets its bearings. This is all less than ideal under certain circumstances, for instance, if you wanted to set up in an unfamiliar area and send out a robot and know where that robot was relative to you.

This is where Received Signal Strength Indication (RSSI) comes in. RSSI is a means for measuring the amount of power present with a radio signal. It provides a value for signal strength, as the name suggests, and is a way of figuring out how good a link is. As a general rule, if a transmitter is moved closer to a receiver, then the strength of the transmitted signal improves at that receiver. The inverse is also, true. The units for RSSI are dBm or decibel-milliwatts, which exist on a logarithmic scale. The magnitude of the RSSI value is inversely proportional to the square of the distance, in other words, the greater in magnitude the negative value returned is, the weaker the signal (Parameswaran, Husain, & Upadhyaya, 2009). For the sake of an example, this means that a -44 dBm signal is better than a -80 dBm signal (Digi, 2017). The relationship between the RSSI value and distance can be expressed as follows

$$RSSI \propto -log(1/distance^2)$$

This all matters because it means that with RSSI, you can get at least a crude idea of how far your receiver is from your transmitter. While it is nice to know about how good your signal is, having only that data, or even knowing how far you are from an object you wish to locate is not enough. It narrows it down to a certain range, but that range of possible locations is a circle, or really, a circle with another concentric circle to give an error range.

For the sake of an informal metaphor, it is like being told that you are about ten miles from someone's house, but you don't know which direction their house is in. Walking the circumference of this circle to find it is probably not an option, as walking 63 miles in a circle is an inefficient way of finding things. So how do you turn that into useful information? You add more constraints. Let us say that you now know that this house you are trying to find is ten miles from you and 12 miles from your car. You know that you are five miles from your car. Suddenly there are only two places that this house can be, and they both lie on a hyperbola. If you took TJ Math 4, you are probably groaning. Don't worry. What is nice about this all, is that by adding a third distance measurement, say the distance from your school to this person's house, you narrow down the area you are looking for the house into a fairly small area. You know where roughly on that hyperbola this house is, and you can find it. See Figure 1.

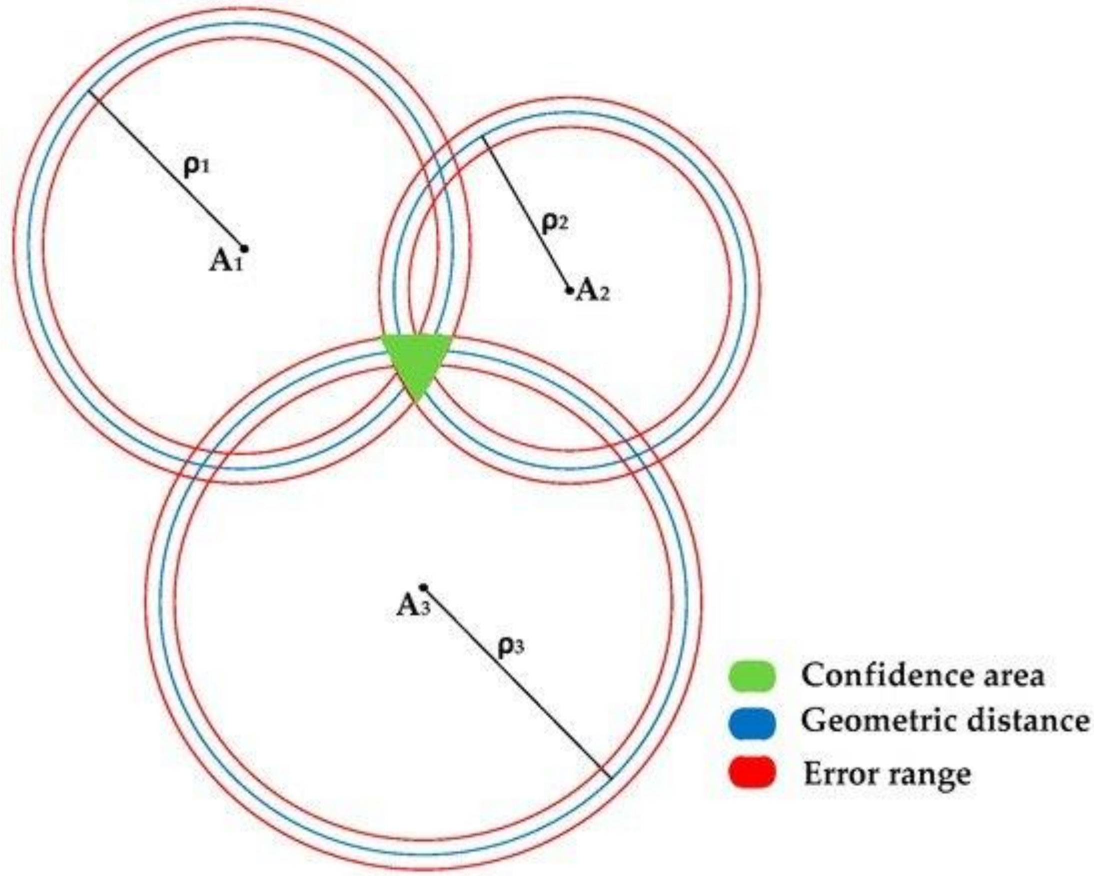


Figure 1: Visualization of multilateration with three distance measurements from three separate locations. Courtesy of Kapoor et al.

This concept is known as multilateration, and essentially, while adding more geometric distances will help narrow down the confidence area further, three is all that is necessary to come up with a fairly reasonable area. This is how a lot of cell phone maps work: they know where the cell towers are (those don't move), and they can figure out their location relative to those and place you on a local map pretty accurately. The abundance of cell towers in cities today only adds to the accuracy of such services (Bekcibasi & Tenruh, 2014).

There are, however, issues with using RSSI. A study by Parameswaran et al. found that RSSI was an unreliable means of localization at the extremes of a device's range given the sheer number of variables that signal strength is affected by. The study went on to suggest that subtle differences in distance may not be discernable, but it was possible to see larger changes, and multiple RSSI readings increase the reliability of localization methods that use RSSI (Parameswaran, Husain, & Upadhyaya, 2009). RSSI is also subject to interference from other signals in the same frequency bandwidth (VIAVI, 2016) as well as objects in the path of the receivers. Parameswaran, Husain, and Upadhyaya concluded that "it is extremely difficult to develop a graph between RSSI and distance which can be used as a standard reference for getting the distance value for a given RSSI value," (2009), which brings us rather nicely to the purpose of this project.

The goal of this research was to determine a function that could reasonably approximate distance as a function of RSSI value, and using that, accurately locate a receiver using three transmitters and the data gathered from RSSI tests.

Procedures

Because acquiring access to several cell towers is a bit beyond the convenient reach of a high school student, Digi XBee Pro S1s were used in their stead (Figure 2). They are relatively inexpensive, costing between 30 and 50 USD. They operate within the 2.4 GHz frequency band and can act as both receivers and transmitters. The XBees sat on top of Waveshare XBee USB Adapter boards (Figure 3), which enabled me to connect them to a computer with a USB to micro USB cord. These devices claim to have ranges of up to 300 feet indoors and up to a mile outdoors, and while I found that they do technically work at several hundred feet, significant packet loss tends to occur beyond distances of about 100 feet.

Figure 2: XBee Pro S1 on a Waveshare board

Figure 3: Waveshare board without XBee on top

I used XCTU to configure the XBees and perform the RSSI tests. I first configured both with the default settings to clear anything I did not want. Since the range test only requires two XBees, I only initially configured two. All the XBees were set with a serial interface baud rate of 9600 symbols per second. Data Bits are set to eight and both Parity and Flow Control are set to "None." Stop Bits is set to one. All this is shown in Figure 4.

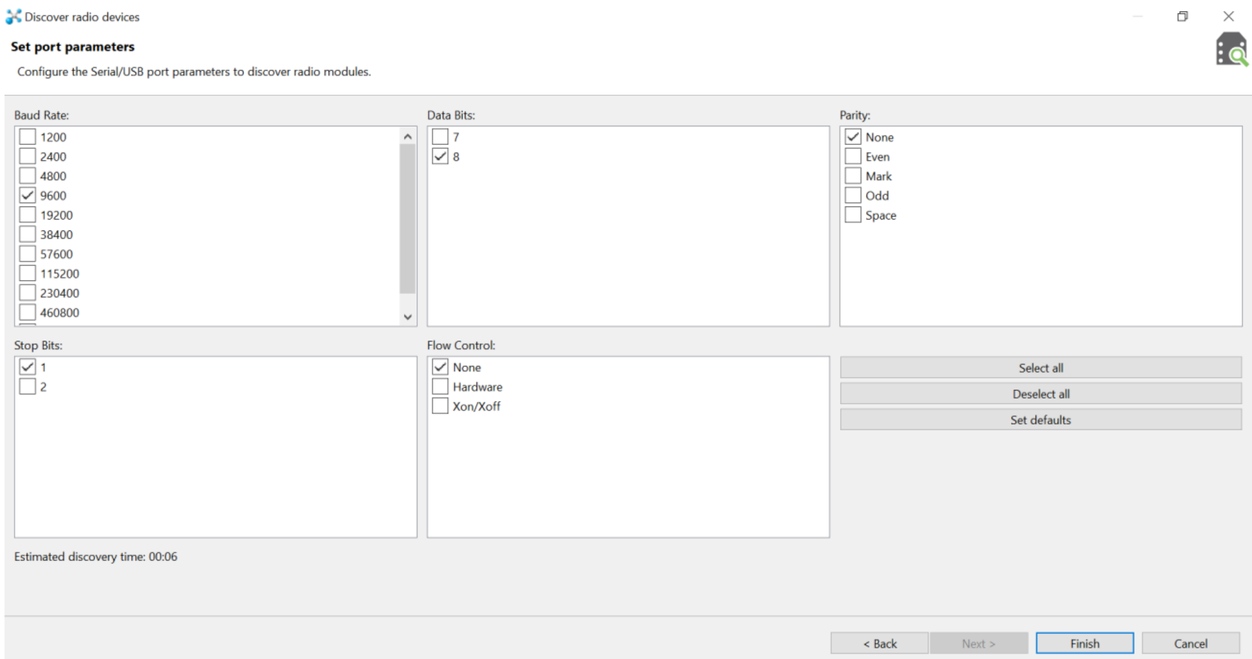


Figure 4: Port parameters

The XBees were configured according to instructions on the Digi website (2017). I set the channel (CH) for both XBees to “C,” and the ID for both to “D161.” The Destination Address High (DH) is set to “13A200,” which is simply the six characters in the MAC Mode (MAC) after the two first zeroes. Both XBees are set to be end devices in Coordinator Enable (CE). All of those settings have to be the same for both XBees. Each XBee has the Destination Address Low (DL) set to the Serial Number Low (SL) of the other XBee, or the last eight characters of the MAC of the other XBee. Most of the rest of the settings are defaults. This all is better detailed in Figure 5.

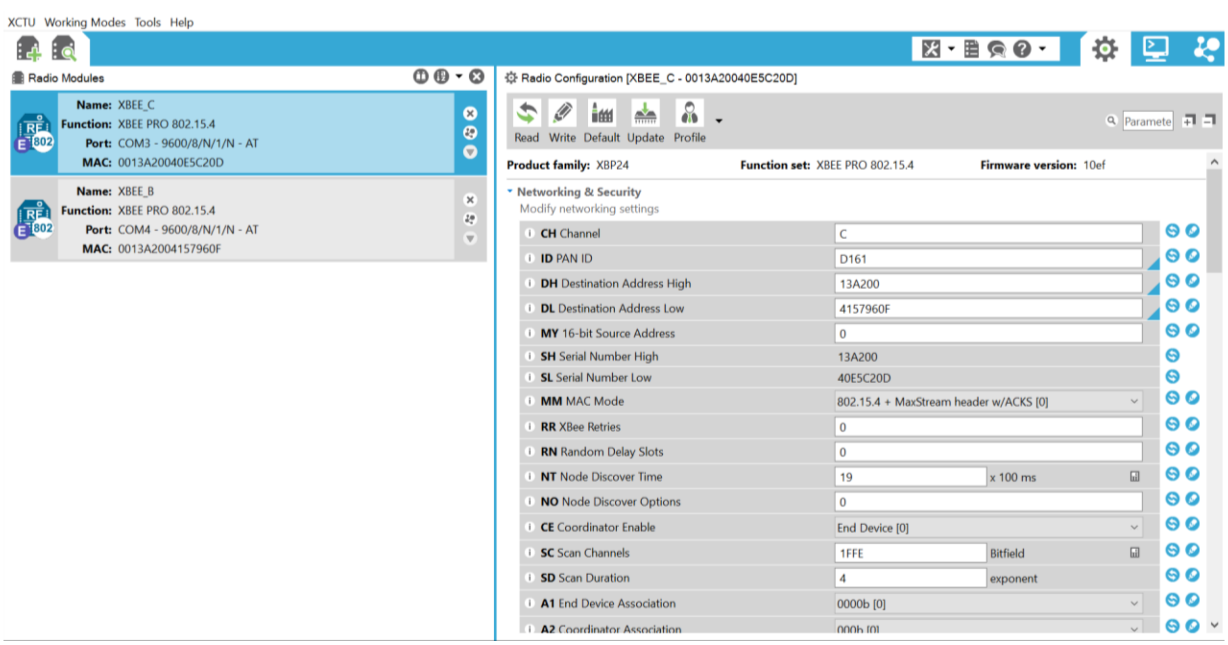


Figure 5: XCTU settings

The test is supposed to allow the local radio device (XBee that stays attached to computer) to be used in API mode, which allows the RSSI for the remote device to be viewed as well, but I could not get this to work and was forced to place both XBees in transparent mode (or AT) mode. This really just meant setting the Power Level (PL) to “Lowest” and setting API Enable (AP) to “API disabled.” See Figure 6.

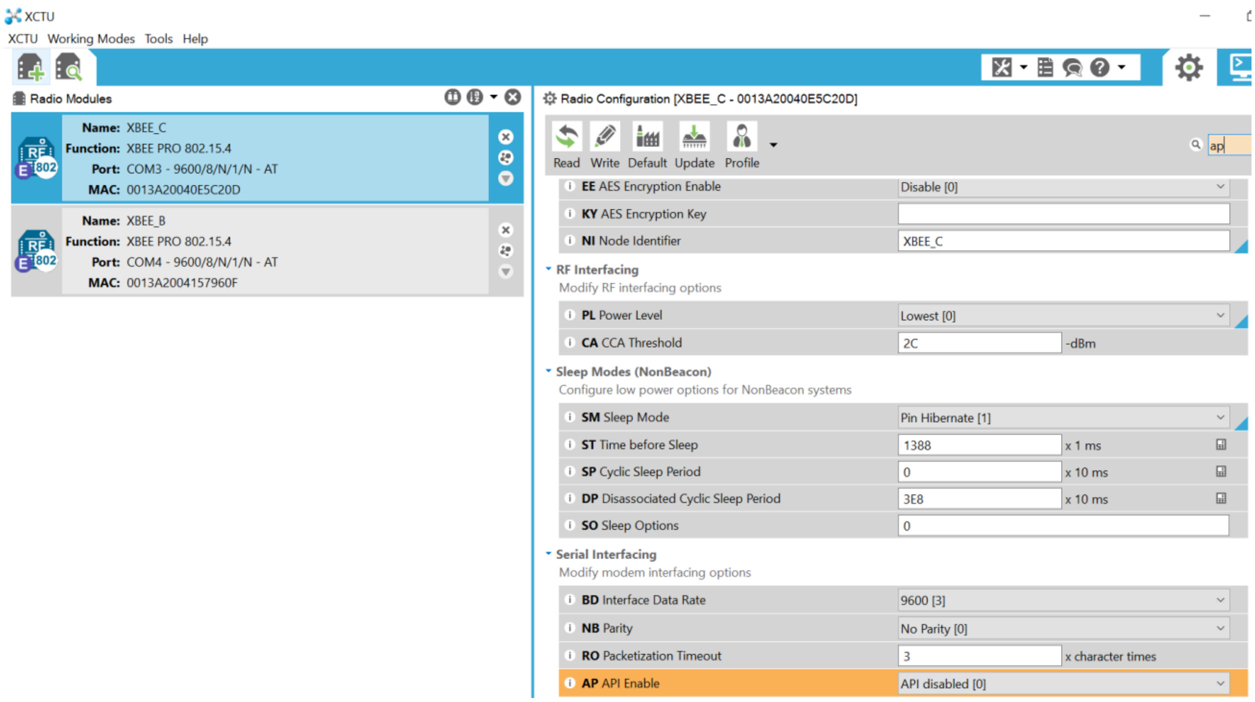


Figure 6: XCTU settings

Under “Radio Modules,” all the XBees attached to the computer are displayed. Choose one device to use as a local radio device and select the second or middle icon on the right to discover radio nodes in the same network. This will bring up any other XBees that are operating in the same channel. Choose one to be the remote device. It should now show up below your local radio device XBee, as shown in Figure 7.

If the remote device has the wrong function, attach the remote device directly to the computer and force write the settings again. This should change the function for the remote device. The function should be “XBEE PRO 802.15.4” and all the XBees should say this. The protocol for this function does not support Cluster range tests, so I used the other option: a loopback test.

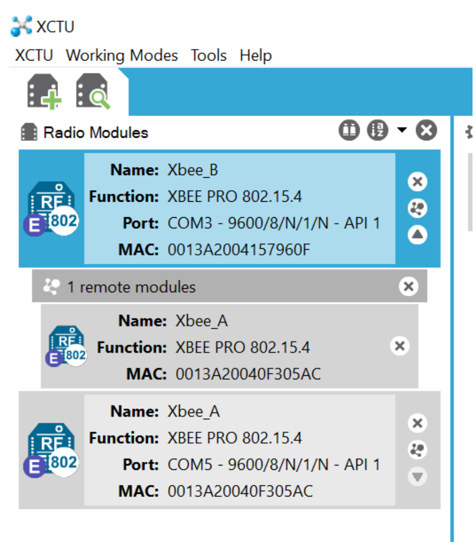


Figure 7: Remote Device attached

Since I used a loopback test, I needed a loopback jumper. This can be achieved by connecting the Rx and Tx pins on the Waveshare board for the remote device, and this should

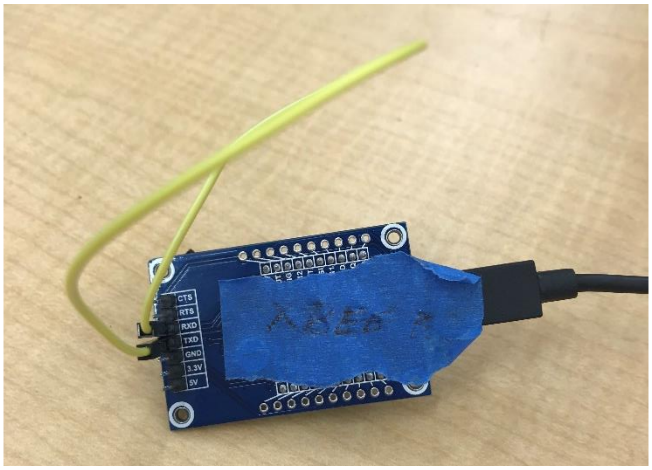


Figure 8: Waveshare and XBee with loopback jumper attached

only be done on the remote device. See Figure 8. This must be done after writing all settings to the board and XBee, as the loopback jumper stops the thing from sending signals through the COM ports and your computer likely won't even notice that the XBee is connected if you plug it in, much less actually write settings to it.

To perform a test, select "Range Test" from "Tools" in the XCTU configuration ribbon (Figure 9). Select the XBee connected to the computer as the local radio device, and then your remote device should pop up. Choose a loopback test, the only real difference is that a cluster test allows more flexibility, but the function does not support cluster tests. I set the Rx timeout and Tx interval to 500ms each for convenience. I also set it to loop infinitely as I found that collecting the data quickly was a challenge with the default 100 packets. See Figures 9 and 10.

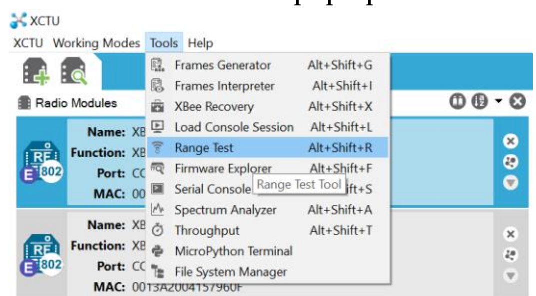


Figure 9: Range test

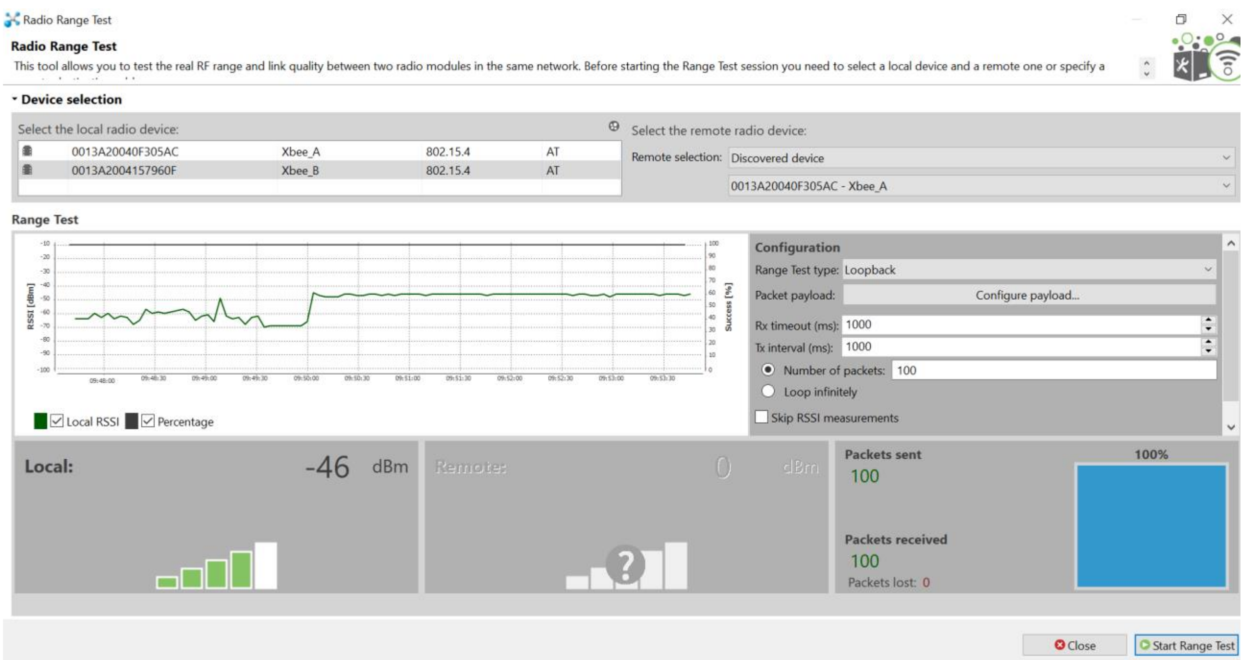


Figure 10: Range test. Note that in this particular image, the test is not set to loop infinitely and the Rx and Tx time intervals are at 1000 ms rather than 500 ms

Figure 11: Frisbees mark distances. Umbrella covers researcher and local radio device. Not pictured: remote device with battery pack, hapless intern

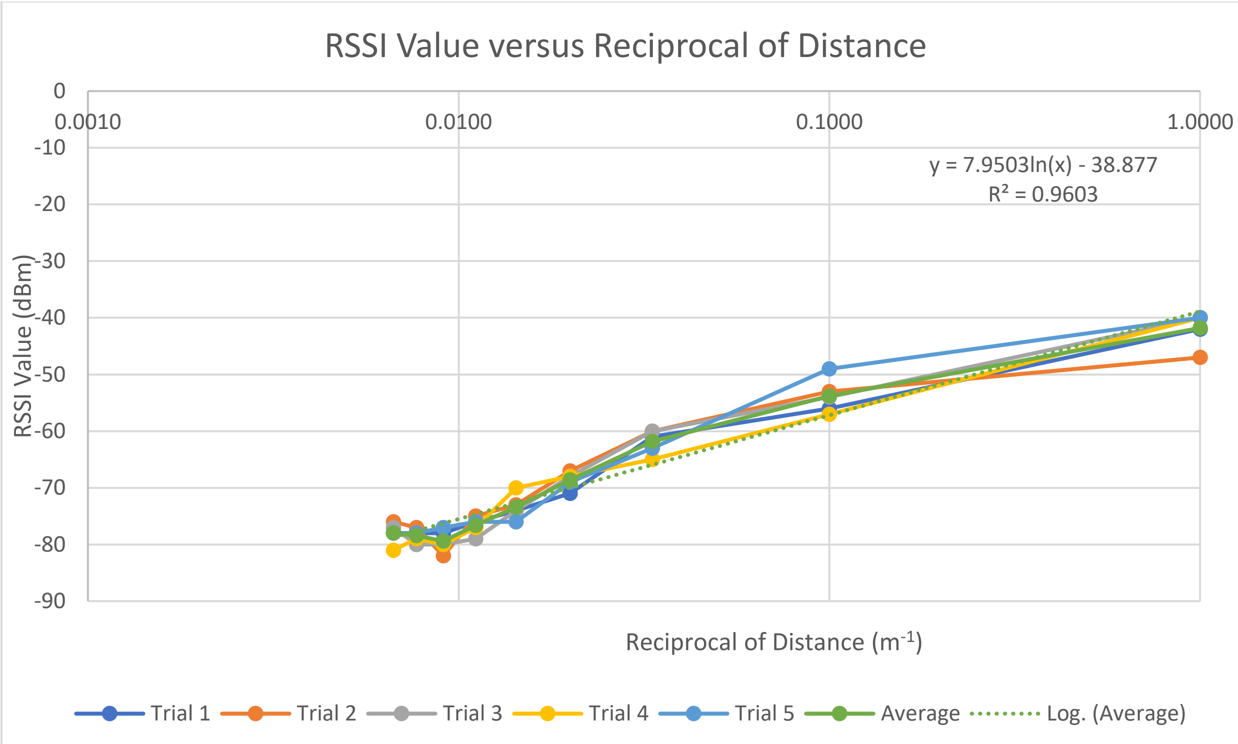
To take measurements, I set up an open field with a laptop and an XBee running a range test and marked out increments of distance from the XBee. I placed markers at 10 feet from the device, 30 ft, 50 ft, 70 ft, 90 ft, 110 ft, 130 ft, and 150 ft. I carried the remote device from a distance of 1 foot away from the local radio device to each marker and paused at each marker for several seconds to allow the RSSI readings to adjust. See Figure 11. I video-captured the range test on the laptop and recorded the movement of the remote device (which was attached to a portable battery pack via micro USB cable as a power source) using a phone camera. I synced the two videos up using the audio at the beginning of the clips and recorded the RSSI values for each distance. I repeated this process for a total of five trials.

Results and Discussion

As expected, there was a negative correlation between signal strength and distance between the XBees. The raw data is available in Table 1. RSSI values were relatively consistent across the trials. The ranges are a bit mercurial, but tend to decrease as the distance increases and the signal strength decreases. The standard deviation also decreases overall, although like the ranges, there was more variability at some distances than others. The overall trend makes sense given that it is a logarithmic scale. There is not a clear cause for the variability being greater at some distances than others. The reciprocal of the distance (1/Distance) column is included in Table 1 as it is the scale for the x-axis of Graph 1.

Table 1

		RSSI Value Measured (dBm)							
Distance	1/Distance	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Average	Range	StdDev
1	1.0000	-42	-47	-40	-40	-40	-41.8	7	3.0332
10	0.1000	-56	-53	-54	-57	-49	-53.8	8	3.1145
30	0.0333	-61	-60	-60	-65	-63	-61.8	5	2.1679
50	0.0200	-71	-67	-68	-68	-69	-68.6	4	1.5166
70	0.0143	-74	-73	-74	-70	-76	-73.4	6	2.1909
90	0.0111	-76	-75	-79	-77	-76	-76.6	4	1.5166
110	0.0091	-78	-82	-80	-80	-77	-79.4	5	1.9494
130	0.0077	-78	-77	-80	-79	-78	-78.4	3	1.1402
150	0.0067	-78	-76	-77	-81	-78	-78.0	5	1.8708



Graph 1

This graph plots the measured RSSI value as a function of the reciprocal of the distance at which the value was measured. The x-axis is a logarithmic scale, as is signal strength. The graph is roughly linear on this scale. It is worth noting that the readings from a distance of one foot away are not on the left, where traditional scales would put closer values, but rather on the right. As the distance increases, the data points move left and grow closer together.

The RSSI values decrease more markedly initially, then appear to plateau. In a few of the trials, the RSSI values measured at the farthest distance are actually higher than closer distances. I suspect that this is more due to experimental error than to any significant finding.

The R-squared value of 0.9603 is pretty close to 1 and indicates that the trendline is a good fit.

The reason the graph is set up this way is because signal strength is proportional to the reciprocal of distance squared, shown in the equation below.

$$S = \frac{c}{r^2}$$

If S is signal strength, r is the distance between the transmitter and receiver, and c is a constant, then armed with the knowledge that S is a logarithmic scale, we get:

$$\log S = \log \frac{c}{r^2}$$

The RSSI values are log S and and we can then multiply the right side by 2/2 to get us closer to something more useful on that end.

$$\log S = \frac{2}{2}\log\frac{c}{r^2} = 2\log\left(\frac{c}{r^2}\right)^{1/2} = 2\log\frac{\sqrt{c}}{r} = -2A\log\frac{1}{r} + \log\sqrt{c}$$

Where A is another constant. This is the same as the form of the trendline for the graph, with the only difference being that the trendline uses the natural logarithm rather than the more standard base ten. This is easily rectified by converting the natural logarithm to a logarithm in base ten.

$$\ln x = \frac{\log x}{\log e}$$

The trendline equation

y = 7.9503ln(x) - 38.877

then becomes

y = 18.3062log (x) – 38.877

In this equation, y is the RSSI value and x is the reciprocal of the distance. To find the distance as a function of the measured RSSI value, we solve for x, then take the reciprocal of that.

$$\frac{1}{distance} = x = 10^{\frac{(y+38.877)}{18.3062}}$$

$$\frac{1}{10^{\frac{(y+38.877)}{18.3062}}} = distance$$

Plugging in each of the average RSSI values gives us distances of 1.44 ft, 6.53 ft, 17.84 ft, 41.95 ft, 76.71 ft, 114.69 ft, 163.08 ft, 143.82 ft, 136.77 ft, which are 0.44 ft, 3.47 ft, 12.16 ft, 8.05 ft, 6.71 ft, 24.69 ft, 53.08 ft, 13.82 ft, and 13.23 ft off the actual distances respectively. That makes for an average of 15.07 feet off of the known distance, and if used in trilateration, gives a search area of approximately 450 square feet, which, while it seems like a large area, is desirable when faced with searching an area that could be 70,000 square feet or more. For the most part, those are not terrible estimates, with the exceptions being at 90 feet and 110 feet. Both of these distances were around where the XBees started to drop packets, so it is possible that these measurements were flawed.

It is also possible that the RSSI measurements at 150 feet are inaccurately high because the person holding the remote device was unable to crouch down as he did at every other distance due to the fact that there was a large puddle at that distance marker. The more elevated remote device may have had an easier time receiving and returning the signal, incorrectly making the RSSI values measured at that distance appear stronger than they should have been and thus throwing off the trend and therefore the distance predictions.

Another potential source of error was that there was some drizzling rain present throughout all the trials. It is unclear how much this may have affected the signal strength, but the rain was somewhat inconsistent and the humid atmosphere may have been a problem.

The human error in this study had two likely sources. The person carrying the remote device may have inadvertently body-blocked some of the signal, which would have affected the strength of the signal. The person recording the data, despite the synced videos, still had to determine which exact RSSI measurement to record, as the value tended to fluctuate a bit at each distance.

There were also some objects that were not directly in the way, such as basketball hoops and a soccer goal, as well as the umbrellas the researchers were carrying, that may have had some minor impacts on signal strength.

Finally, there may have been some interference because Digi XBee Pro S1s operate in the 2.4 GHz bandwidth, which is an extremely popular bandwidth to utilize. It is possible that other devices operating in the same bandwidth interfered with the range test.

In contrast with some other studies, I believe RSSI could be used in a rough estimate to determine distance and location under more ideal conditions. That said, signal strength is highly variable, and even with the fairly consistent conditions I operated under, there was a considerable amount of variability. Using more measurements from more XBees and more range tests would give more accurate data and having a better controlled environment would be useful were this experiment to be repeated. I would also encourage more experimentation with presets and configuration to try and find the ideal settings for something like this. It is also possible that other radio devices would yield more accurate readings. I believe that the next step is to figure out how to connect multiple radio devices in a manner that RSSI readings from multiple different locations can be read, so that the object can be tracked in two dimensional space and objects can be localized using RSSI multilateration. It may be possible to achieve this by writing a set of new settings to the remote device wirelessly and connecting it to a different local radio device to perform range tests from another location.

Works Cited

Digi International Inc. (2017). Signal Strength and the RSSI pin. Last updated November, 30, 2017. Retrieved from https://www.digi.com/resources/documentation/Digidocs/90001456-13/concepts/c_rssi_pin_and_signal_strength.htm

Digi International Inc. (2017). Signal Strength and the RSSI pin. Last updated November, 30, 2017. Retrieved from https://www.digi.com/resources/documentation/Digidocs/90001456-13/Default.htm#tasks/t_try_range_test.htm

Kapoor, R., Ramasamy S., Gardi, A., Bieber, C., Silverberg, L., & Sabatini, R. (2016). A Novel 3D Multilateration Sensor Using Distributed Ultrasonic Beacons for Indoor Navigation. *Sensors, 16(10), 1637. Retrieved from* https://doi.org/10.3390/s16101637

Parameswaran, A.T., Husain, M. I., Upadhyaya, S. (2009). Is RSSI a Reliable Parameter in Sensor Localization Algorithms – An Experimental Study. Department of Computer

Science and Engineering, State University of New York at Buffalo. Retrieved from https://cse.buffalo.edu/srds2009/F2DA/f2da09_RSSI_Parameswaran.pdf

Bekcibasi, U., Tenruh, M. (2014). Increasing RSSI Localization Accuracy with Distance Reference Anchor in Wireless Sensor Networks. *Acta Polytechnica Hungarica* 11(8), 103-120. Retrieved from https://uni-obuda.hu/journal/Bekcibasi_Tenruh_54.pdf

VIAVI. (2016). Solving a High RSSI Issue with InterferenceAdvisorTM. Retrieved from https://www.viavisolutions.com/en-us/literature/solving-high-rssi-issue-interferenceadvisor-case-studies-en.pdf