European Journal of Scientific Research
ISSN 1450-216X Vol.46 No.1 (2010), pp.048-061

http://www.eurojournals.com/ejsr.htm

Node Positioning in ZigBee Network Using Trilateration Method Based on the Received Signal Strength Indicator (RSSI)

R, Mardeni
Faculty of Engineering, Multimedia University 63100
Cyberjaya, Selangor, Malaysia
E-mail: mardeni.roslee@mmu.edu.my
Tel: +60383125481; Fax: +60383183029

Othman, Shaifull Nizam
Faculty of Engineering, Multimedia University 63100
Cyberjaya, Selangor, Malaysia
E-mail: snoengineer@gmail.com
Tel: +60193149136; Fax: +60326810354

Abstract

This paper investigates the possibility of implementing node positioning in the ZigBee wireless sensor network by using a readily available Received Signal Strength Indicator (RSSI) infrastructure provided by the physical layer of 802.15.4 networks. In this study the RSSI is converted to the distance providing the basis for using the trilateration methods for location estimation. The software written in C# is used to solve the trilateration problem and the final results of trilateration methods are mapped using Google maps. Providing node positioning capability to the ZigBee network offers an enormous benefit to the Wireless Sensor Networks applications, possibly extending the functionality of existing software solution to include node tracking and monitoring without an additional hardware investment.

Keywords: Positioning, ZigBee, Trilateration, Received Signal Strength Indicator (RSSI)

1. Introduction

Satellite-based Global Positioning System (GPS) has been used previously alongside the GIS applications to provide the spatial information to map real world location to the computer representation, thus making the planning and management of resources more informed. The GPS is a space-based global navigation satellite system that provides reliable location and time information in all weather conditions and at all times and anywhere on or near the Earth when there is an unobstructed line of sight to four or more GPS satellites. The positioning system that uses the GPS receiver adds cost to the solution infrastructure, increases power requirements to operate the on-board receiver chip and increases infrastructure maintenance points, making the deployment of such solution not commercially viable.

Wireless Sensor Networks, particularly the ZigBee devices offer a more viable positioning method that uses existing infrastructure without escalating the operational expenses. This paper explains the method used to do node positioning in the ZigBee network.

2. Problem Statement

The ZigBee node's physical hardware mainly comprises of low specification and low cost component to facilitate mass production, which makes it affordable to be deployed intensively in a monitoring zone. This has created a challenge in mapping the locations of sensor nodes as the hardware cannot provide precise timing in calculating time of flight of a packet, an important parameter in estimating distance between transmitting node and receiving node for time-based positioning methods such as Uplink-Time Difference of Arrival (U-TDOA) and Time of Arrival (TOA). The common use of cheap single Omni directional antenna in most of the ZigBee deployments also ruled out the possibility of using techniques that rely on packet Angle of Arrival (AoA) for estimating the location. This paper investigates positioning methodology that is based on received signal strength to estimates the distance to node and performing trilateration of those distances. This approach does not require any hardware modifications to the sensor node, providing a more viable positioning methodology in ZigBee networks.

3. Positioning In Wireless Sensor Networks

Wireless networks uses radio microwave to communicate with each other and particularly with the ZigBee network, it uses the 2.4 GHz radio frequency that can be measured for positioning purposes. The ZigBee network infrastructure provides radio signal properties as part of the Quality of Service such as the Time of Arrival (ToA), Time Difference of Arrival (TDoA), Angle of Arrival (AoA) and the Received Signal Strength Indicator (RSSI) [1].

i. Received Signal Strength Indicator (RSSI)

In an open environment such as a cattle breeder farm, with high probability of Line of Sight (LoS) and low multipath effect, it is possible to use the simple RSSI-based location algorithm if coarse accuracy is acceptable. The received signal strength can be measured for every packet received, the value indicating the signal strength is provided as part of the Link Quality Indicator (LQI) as a RSSI value and available at the PHY layer in IEEE 802.15.4 network. Using the RSSI value, a distance to node can be measured and trilateration calculation can be performed against other nodes with known positions.

ii. Uplink-Time Difference of Arrival (U-TDOA)

The signal time difference received by the device from cells antenna tower is being calculated in determining the positioning of the device. Using the time difference information gathered, the distance, on the other hand, from cells antenna tower or base station to the device could be easily estimated, thus leading to the coordinate positioning of the device.

iii. Time of Arrival (TOA)

It is similar to the U-TDOA; however, the only difference is that it uses the absolute time of arrival at a cells antenna tower or base station rather than the difference between two stations. As a result the distance can be calculated from the time of arrival as signals travel with a known velocity which in this case is the speed of light (300,000 kilometres per second). The different data of time arrival from two cells antenna tower or base station will formulate a position to two circles and the third cells antenna tower or base station is required to determine the precise device position.

iv. Angle of Arrival (AoA)

The AoA in ZigBee networks can be achieved by grouping together three or four nodes involved in a typical radio interferometric to form an antenna array, which acts as an anchor node. The bearing of the target node can then be estimated by computing the angle of hyperbola asymptote [2]. In this method, the AoA mechanism locates the node at the point where the lines along the angles from each cells antenna intersect.

4. Positioning Using Zigbee Infrastructure

The positioning of node using the RSSI value provided by LQI of the ZigBee node offers possibility of providing spatial data without any additional hardware requirements to the existing solution. The localisation process involves the use of trilateration calculation for intersection of three spheres of which the radius is obtained from the distance estimated from the RSSI value; to work this model requires that the transmitting node must be inside the intersection of three other receiver nodes of which the locations are known. One possibility to acquire a distance is measuring the received signal strength of the incoming radio signal. The idea behind RSS is that the configured transmission power at the transmitting device (P_{TX}) directly affects the receiving power at the receiving device (P_{RX}). According to Friis' free space transmission equation [3], the detected signal strength decreases quadratically with the distance to the sender.

$$P_{RX} = P_{TX} \, X \, G_{TX} \, X \, G_{RX} \left(\frac{\lambda}{4\pi d} \right)^2 \qquad (1)$$

Where;

P_{RX}	=	*Transmission power of sender*
P_{TX}	=	*Remaining power of wave at receiver*
G_{TX}	=	*Gain of transmitter*
G_{RX}	=	*Gain of receiver*
λ	=	*Wave length*
d	=	*Distance between sender and receiver*

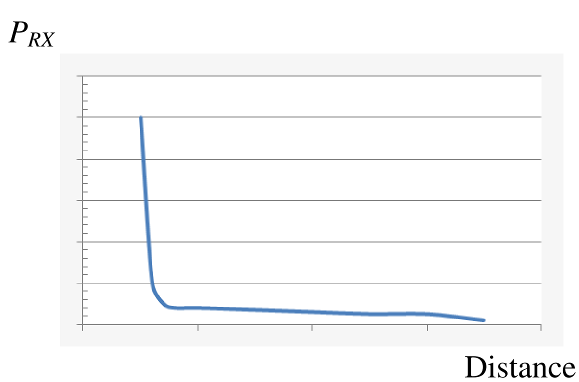
In embedded devices, the received signal strength is converted to a received signal strength indicator (RSSI) which is defined as ratio of the received power to the reference power P_{ref} [3]. Typically, the reference power represents an absolute value of P_{ref}=1mW.

The RSSI formula can be shown as in eqn. (2) as below;

$$RSSI = 10 \, X \log \left(\frac{P_{RX}}{P_{ref}} \right) \, dBm \qquad (2)$$

An increasing received power results a rising RSSI. Distance *(d)*, is indirect proportional to RSSI. In practical scenarios, the ideal distribution of P_{RX} *is* not applicable, because the propagation of the radio signal is interfered with a lot of influencing effects. The RSSI value is provided by the PHY layer of the ZigBee network.

Figure 1: Relationship between the Transmit power and the distance



There are few factors that degrade and impact the RSSI values in the wireless networks and ZigBee in particular:

- Reflections on metallic objects
- Superposition of electro-magnetic fields
- Diffraction at edges
- Refraction by media with different propagation velocity
- Polarisation of electro-magnetic fields
- Unadapted MAC protocols

The RSSI based distance estimation of the target node can then be used for positioning calculation using the trilateration or multilateration formula. Few multilateration methods has been proposed to solve the localisation problem in 3D space, such as Semidefinite Programming [4], MDS-MAP [5] for centralised algorithm approach and Diffusion based Multilateration [6] and Gradient based Multilateration [7].

4.1. The Mathematics of Trilateration

Trilateration is a method of determining the relative position of objects using the geometry of triangles in a similar fashion as triangulation. Unlike triangulation, which uses angle measurements (together with at least one known distance) to calculate the subject's location, trilateration uses the known locations of two or more reference points, and the measured distance between the subject and each reference point. To accurately and uniquely determine the relative location of a point on a 2D plane using trilateration alone, generally at least 3 reference points are needed (at least 4 points are needed in the 3D plane).

Consider the basic formula for a sphere as shown as in eqn. (3);

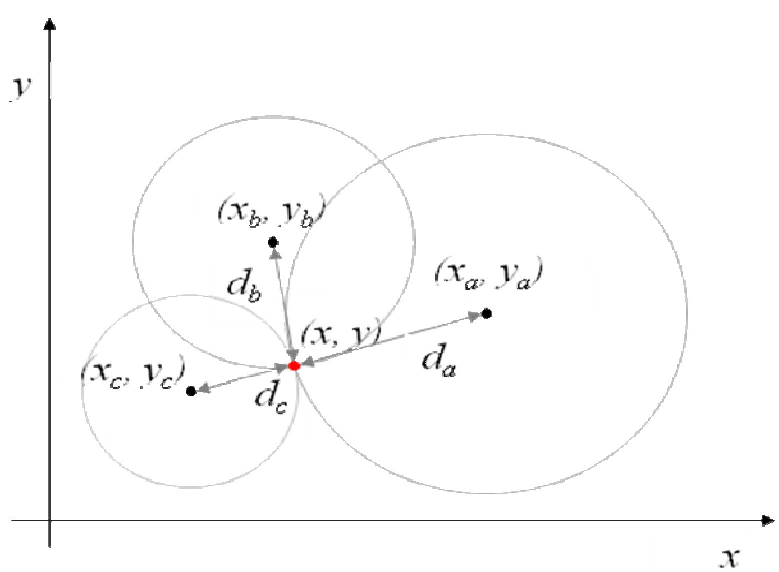
$$d^2 = x^2 + y^2 + z^2 \tag{3}$$

For a sphere centred at a point (x_a, y_a, z_a) the equation is simplified as shown as in eqn. (4);

$$d^2 = (x - x_a)^2 + (y - y_a)^2 + (z - z_a)^2 \tag{4}$$

Since we assume all the nodes spans out on the same plane, consider the three reference nodes *(a, b and c)* that has distance *(d_a, d_b, d_c)* to the target node as follows;

Figure 2: Intersection of 3 Spheres



The formula for the all spheres on one plane (circles) is as shown below eqn. (5), eqn. (6) and eqn. (7);

Sphere A; $d_a^2 = (x - x_a)^2 + (y - y_a)^2$ (5)

Sphere B; $d_b^2 = (x - x_b)^2 + (y - y_b)^2$ (6)

Sphere C; $d_c^2 = (x - x_c)^2 + (y - y_c)^2$ (7)

The eqn. (5), eqn. (6) and eqn. (7) are further expanded to become the following eqn. (8), eqn. (9) and eqn. (10);

$d_a^2 = x^2 - 2x.x_a + x_a^2 + y^2 - 2y.y_a + y_a^2$ (8)

$d_b^2 = x^2 - 2x.x_b + x_b^2 + y^2 - 2y.y_b + y_b^2$ (9)

$d_c^2 = x^2 - 2x.x_c + x_c^2 + y^2 - 2y.y_c + y_c^2$ (10)

The three eqn. (8), eqn. (9), and eqn. (10) are independent non-linear simultaneous equations which cannot be solved mathematically; however, using method proposed by Dixon [8] to obtain radical plane for sphere intersection, subtracts 2 spheres; eqn. (10) from eqn. (9) we get the following linear eqn. (11);

$d_b^2 - d_c^2 = 2x(x_c - x_b) + x_b^2 - x_c^2 + 2y(y_c + y_b) + y_b^2 - y_c^2$ (11)

And subtract eqn. (8) from eqn. (9), we get the following linear eqn. (12);

$d_b^2 - d_a^2 = 2x(x_a - x_b) + x_b^2 - x_a^2 + 2y(y_a + y_b) + y_b^2 - y_a^2$ (12)

Rearranging the eqn. (11), to produce a variable called V_a, as eqn. (13) as follows,

$$x(x_b - x_c) - y(y_b - y_c) = \frac{(x_c^2 - x_b^2) + (y_c^2 - y_b^2) + (d_b^2 - d_c^2)}{2} = V_a \quad (13)$$

Rearranging the eqn. (12), to produce a variable called V_b, as eqn. (14) as follows,

$$x(x_b - x_a) - y(y_b - y_a) = \frac{(x_a^2 - x_b^2) + (y_a^2 - y_b^2) + (d_b^2 - d_a^2)}{2} = V_b \quad (14)$$

Resolve the eqn. (13) and eqn. (14) to gain the intersection point *'x'* and 'y' of these two equations as the following eqn. (15) for 'y' value and eqn. (16) for 'x' value;

$$y = \frac{V_b(x_b - x_c) - V_a(x_b - x_a)}{(y_a - y_b)(x_b - x_c) - (y_c - y_b)(x_b - x_c)} \quad (15)$$

$$x = \frac{y(y_a - y_b) - V_b}{(x_b - x_c)} \quad (16)$$

If the equations do not produce real solutions; it means the two sphere does not have intersection point (possibly the spheres does not meet).

4.2. Node Positioning Using Trilateration from Known Fixed Node Location in Field Matrix

Trilateration (or multi lateration for that matter) in 2D surface requires intersection of at least three circles of which the coordinates of actual location is known. The real world object placement on the earth surface can be represented by few coordinates system, such as the (Latitude, Longitude pair), the Universal Transverse Mercator (UTM) and Universal Polar Stereographic (UPS) or even the crude Cartesian coordinate that place the centre of the earth as the point of origin. The latitude and longitude pair value is the most used coordinate system for civilian use, and quite simply becomes a standard for GPS and GIS mapping system.

Latitude (abbreviation: Lat., φ, or phi) is the angle from a point on the Earth's surface to the equatorial plane, measured from the center of the sphere. Lines joining points of the same latitude are called parallels, which trace concentric circles on the surface of the Earth, parallel to the equator. The North Pole is 90° N; the South Pole is 90° S. The 0° parallel of latitude is designated the equator, the fundamental plane of all geographic coordinate systems. The equator divides the globe into Northern and Southern Hemispheres [9]. Longitude (abbreviation: Long., λ, or lambda) is the angle east or west of a reference meridian between the two geographical poles to another meridian that passes through an arbitrary point. All meridians are halves of great circles, and are not parallel. They converge at the north and south poles [9].

A fixed node is the node in the ZigBee network of which their actual position in the earth surface is known. To achieve this, each of the fixed nodes will be placed on the field alongside with the mapping tool such as the portable GPS reader and the location profile is recorded and stored in the node location registry (database). The profiling process captures locations, descriptions, device functionality and among other things the device installation history. Another solution would be to provide the GPS module on the fixed node and read the value from the device real time; this allows dynamic positioning of the fixed node and allows position modification to be done without manual re calibrating. The automated profiling is an ideal solution for large breeder farms or farms that does not have permanent grazing fields.

The positioning of the Fixed Node is also crucial to enable maximum field coverage and ensure that the blind node transmission signal can be read by at least three fixed nodes. To ensure optimal read coverage, the fixed node will be placed on the field using the triangular matrix configuration as follows;

Where;
Maximum distance between nodes is 100 meters.
A = intersection point for node 1, 5 and 6
B = intersection point for node 1, 2 and 7
C = intersection point for node 1, 3 and 4

Figure 3: Node Detection Field

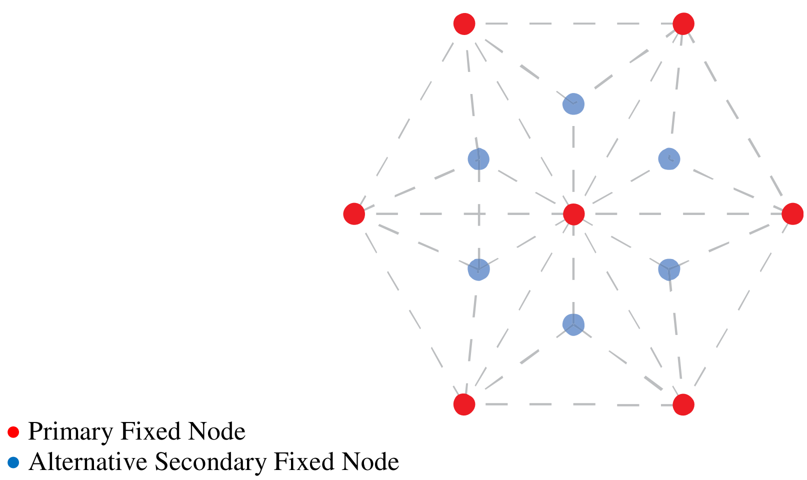
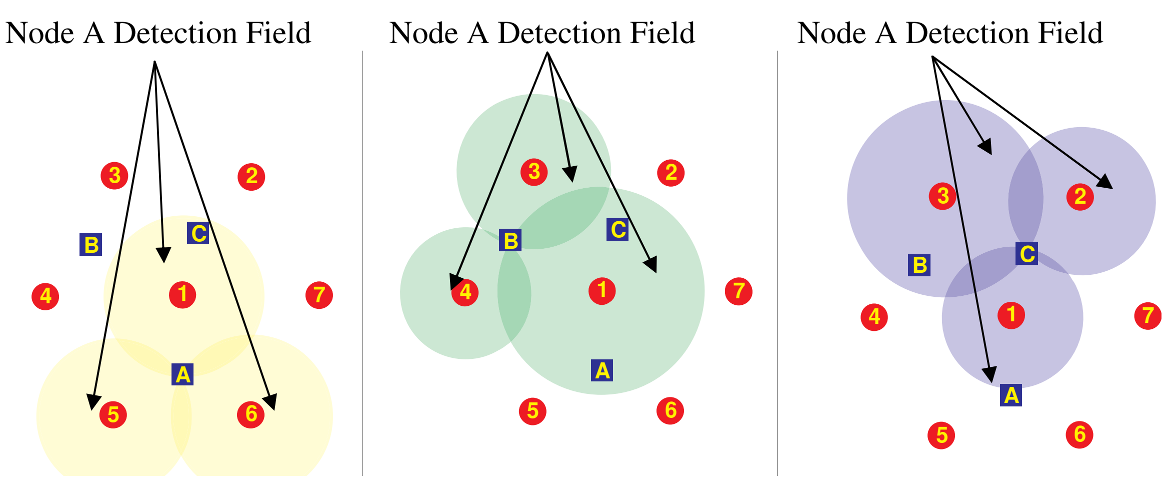


Figure 4: Node Detection Field



The matrix configuration field of the fixed nodes as in Figure 3 above will guarantee that any node that is inside the field will be detected by at least 3 fixed nodes. To improve the detection and location positioning, the alternative secondary Fixed Node can be installed as in Figure 4.

The fixed nodes that read the blind node will log the blind node device id (MAC address), RSSI, fixed node device id and the timestamp in the log file and stored in the gateway; it is updated every two seconds.

4.3. Fixed Nodes Profiles

In this project, the profiling is done manually by using a portable APOS Bluetooth GPS reader connected to a notebook running Google Earth [9] on Microsoft Windows Vista. The fixed nodes are installed in the topology illustrated in the Table 1 above and each of the node location is read and stored in the "CollectorProfile" table in the database. The collector location values are as follows;

Table 1: CollectorProfile" table storing the location of fixed node

ID	CollectorName	Location	TransmitPower	Latitude	Longitude
1	Collector1	SFKD	10	3.159167	101.557500
2	Collector2	SFKD	10	3.159961	101.558053
3	Collector3	SFKD	10	3.159997	101.557217
4	Collector4	SFKD	10	3.159214	101.556731
5	Collector5	SFKD	10	3.158542	101.557097
6	Collector6	SFKD	10	3.158425	101.557911
7	Collector7	SFKD	10	3.159403	101.558467

The "Latitude" and "Longitude" values are stored in the decimal degree format. The "TransmitPower" is the" internal settings of the node that determine the coverage radius and is not used in this demonstration. The node profile provides the fixed node positions that will be used when performing the trilateration.

4.4. Blind Node Monitors

The "BlindNodeMonitor" is tag monitor software running on the Gateway that records the tag and stores them in the log file. The following table lists the partial values extracted from the log file. However, d*ue to the nature of the log file, which has long records, the values are summarised here to save space.*

Table 2: "NodeLocationRegistry" Table

ID	CollectorID	TimeDetected	TagID	RSSI
1	Collector1	24/01/2010 13:26:19	Tag1	70
2	Collector1	24/01/2010 13:26:19	Tag3	50
3	Collector1	24/01/2010 13:26:19	Tag4	65
4	Collector1	24/01/2010 13:26:19	Tag6	65
5	Collector1	24/01/2010 13:26:19	Tag7	70
6	Collector2	24/01/2010 13:26:19	Tag7	50
7	Collector3	24/01/2010 13:26:19	Tag2	65
8	Collector5	24/01/2010 13:26:19	Tag3	70
9	Collector6	24/01/2010 13:26:19	Tag5	55
10	Collector6	24/01/2010 13:26:19	Tag6	80
11	Collector7	24/01/2010 13:26:19	Tag6	55
12	Collector7	24/01/2010 13:26:19	Tag6	55
13	Collector1	24/01/2010 13:26:20	Tag2	75
14	Collector1	24/01/2010 13:26:20	Tag4	65
15	Collector1	24/01/2010 13:26:20	Tag5	70
16	Collector2	24/01/2010 13:26:20	Tag1	65
17	Collector3	24/01/2010 13:26:20	Tag1	55

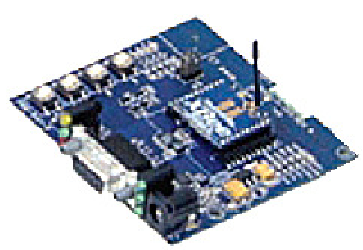
18	Collector4	24/01/2010 13:26:20	Tag2	50
19	Collector4	24/01/2010 13:26:20	Tag3	70
20	Collector5	24/01/2010 13:26:20	Tag4	55
21	Collector7	24/01/2010 13:26:20	Tag7	60
22	Collector1	24/01/2010 13:26:21	Tag1	70
23	Collector1	24/01/2010 13:26:21	Tag2	75
24	Collector1	24/01/2010 13:26:21	Tag3	50
25	Collector1	24/01/2010 13:26:21	Tag5	70
26	Collector1	24/01/2010 13:26:21	Tag6	65
27	Collector1	24/01/2010 13:26:21	Tag7	70
28	Collector2	24/01/2010 13:26:21	Tag1	65
29	Collector2	24/01/2010 13:26:21	Tag7	50
30	Collector3	24/01/2010 13:26:21	Tag1	55
31	Collector3	24/01/2010 13:26:21	Tag2	65
32	Collector4	24/01/2010 13:26:21	Tag2	50
33	Collector4	24/01/2010 13:26:21	Tag3	70
34	Collector5	24/01/2010 13:26:21	Tag3	70

Table 2 shows a snapshot of test data that was gathered by the “BlindNodeMonitor” application written in Python language running on the ZigBee gateway, the DiGi XConnectPort X8 (Figure 5).

In a large scale enterprise system, the node location registry can be use as an asset movement history and separate module can be written and installed to perform the analysis. This information is a crucial part of the supply chain system.

Figure 5: DIGI International XConnectPort X8 Gateway used as the Fixed Node

Figure 6: DIGI International MaxStream XBee Series 2 with the development board used as the Blind Node



4.5. RSSI to Distance Estimation

Received Signal Strength Indicator (RSSI) value is a useful data that is provided by the ZigBee physical layer but is not an accurate distance pointer. Due to factors discussed in the Positioning Using ZigBee Infrastructure section above, the RSSI can only provide estimation to the blind node distance from the detected fixed node. The use of distance estimation to provide localisation can generate results that are not as accurate as the results obtained from the GPS reader but can be viewed as an acceptable solution for livestock monitoring purposes, where exact meter to meter location is not of primary concern.

The following table is the field tested RSSI values achieved from the MaxStream XBee Series 2 blind node as shown in figure 6, communicating with DiGi XConnectPort X8 Fixed Node on January 24th 2010 with clear weather and the temperature is at 31°C. The polling exercise location is at the public park in Kota Damansara, Petaling Jaya, Selangor, Malaysia (03º 09' 33.000"N, 101º 33' 27.000"E).

Table 3: RSSI to Distance Table

RSSI (dBm)	Distance (M)	RSSI (dBm)	Distance (M)	RSSI (dBm)	Distance (M)
-40	2 to 10	-59	48	-78	86
-41	12	-60	50	-79	88
-42	14	-61	52	-80	90
-43	16	-62	54	-81	92
-44	18	-63	56	-82	94
-45	20	-64	58	-83	96
-46	22	-65	60	-84	98
-47	24	-66	62	-85	100
-48	26	-67	64		
-49	28	-68	66		
-50	30	-69	68		
-51	33	-70	70		
-52	34	-71	72		
-53	36	-72	74		
-54	38	-73	76		
-55	40	-74	78		
-56	42	-75	80		
-57	44	-76	82		
-58	46	-77	84		

Or the distance value can be obtained using the following linear equation as shown in eqn. (17);

$$\text{Distance in meters} = (((-\text{RSSI} - 40.0) + 1.0) \text{ X } 2.0) + 8.0 \tag{17}$$

The value illustrated in the Table 3will be used throughout this project for RSSI to Distance estimation value.

5. Simulation Result and Discussion

The following section performs localisation using the concept illustrated in the previous sections.

5.1. "TagPositioning" Software

The *"TagPositioning"* software is a C# application written specifically to solve the trilateration problem. The application is written using Microsoft Visual Studio 2010 and compiled on Microsoft Windows Server 2008. The actual structure and implementation of the software are beyond the scope of this journal.

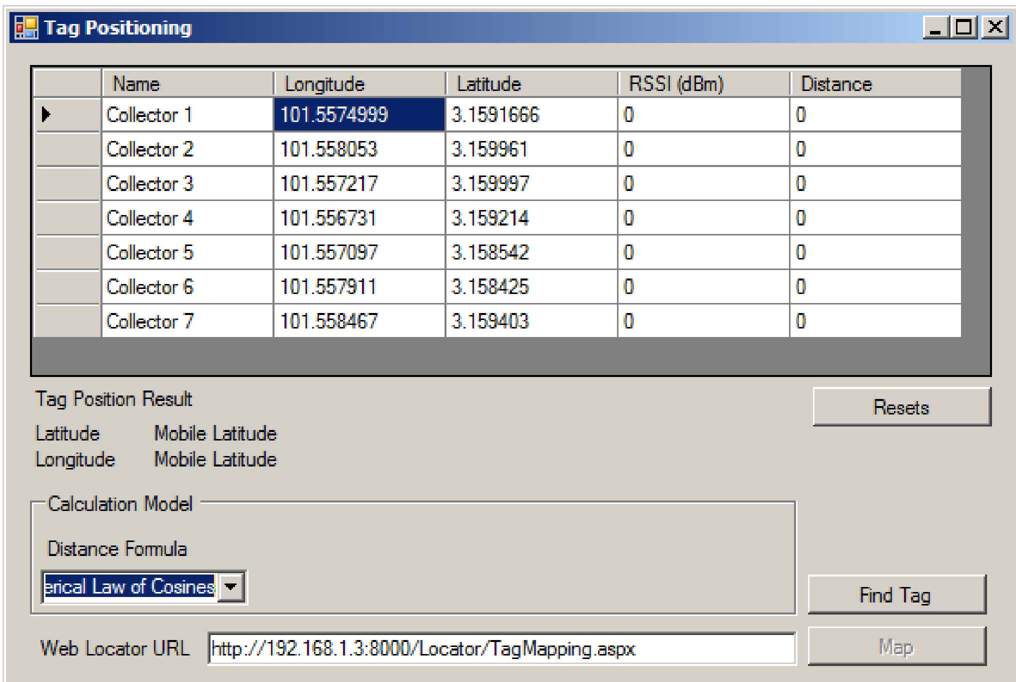
At the heart of the software is a *"TagPositioning"* class that provides the trilateration calculation, the C# class that implements the trilateration method is as follows:

The trilateration formula is implemented in the class as follows:

```
/// <summary>
  /// Intersections of 3 sphere using vector formula. This formula is discussed in details in the
  /// Journal and the thesis.
  ///
  /// </summary>
  /// <param name="point1">The point 1.</param>
  /// <param name="distance14">The radius of sphere 1</param>
  /// <param name="point2">The point 2.</param>
  /// <param name="distance24">The radius of sphere 2</param>
  /// <param name="point3">The point 3.</param>
  /// <param name="distance34">The radius of sphere 3</param>
  /// <returns></returns>

  private Position IntersectionGivenDistance(Position point1, double distance14, Position point2,
                                              double distance24, Position point3, double distance34)
  {
    // convert positions to the common base using km representation
    // when using the KM value, must set the latitude first, because
    // the longitude value depends on the latitude

    double x, y;

    // sets this point to be origins
    double y1 = 0;
    double x1 = 0;

    // because we are using the vector formula, we are calculating the intersection of 3 big circles
    // so convert everthing to a flat surface points, with the first point as the origin
    double y2 = point2.LatitudeInKM - point1.LatitudeInKM;
    double x2 = point2.LongitudeInKM - point1.LongitudeInKM;

    double y3 = point3.LatitudeInKM - point1.LatitudeInKM;
    double x3 = point3.LongitudeInKM - point1.LongitudeInKM;
    double Va, Vb;

    // Va = (x3²-x2²+y3²-y2²+r2²-r3²)/2
    Va = (Math.Pow(x3, 2) - Math.Pow(x2, 2) + Math.Pow(y3, 2) -
              Math.Pow(y2, 2) + Math.Pow(distance24, 2) - Math.Pow(distance34, 2)) / 2;




    // Vb = (x1²-x2²+y1²-y2²+r2²-r1²)/2
    Vb = (Math.Pow(x1, 2) - Math.Pow(x2, 2) + Math.Pow(y1, 2) -
                    Math.Pow(y2, 2) + Math.Pow(distance24, 2) - Math.Pow(distance14, 2)) / 2;

    y = ((Vb * (x2 - x3)) - (Va * (x2 - x1))) / (((y1 - y2) * (x2 - x3)) - ((y3 - y2) * (x2 - x3)));

    x = ((y * (y1 - y2)) - Vb) / (x2 - x1);
```

```
    Position point4 = new Position();

    // must always sets latitude first, this class will convert to degrees automatically
    point4.LatitudeInKM = y + point1.LatitudeInKM;
    point4.LongitudeInKM = x + point1.LongitudeInKM;

    return point4;
        }
```

Figure 7: “TagPositioning” Interface



5.2. Solving Localisation using “TagPositioning” Software

The software performs trilateration of blind node one at a time and maps the result using the Google Maps (http://maps.google.com)[10]. The “NodeLocationRegistry” table in Table 2 is used for the calculation in this section. The collector position (latitude and longitude) from the” CollectorProfile” has been entered in the “TagPositioning”.

To find the tag position, the software needs 3 data from unique collectors. The selection of data is based on the following criteria;

1. Select all the data for Tag1in current minute, if data is not enough (3 unique fixed node), expand the search to include the last 5 minutes, and keep expanding for every 5 minutes until the search returns at least 10 data, if data is not found. The trilateration cannot be performed on the selected cattle due to insufficient data available.
2. Sort the RSSI value in ascending order
3. Select the first 3 unique collectors that detect the Tag1.

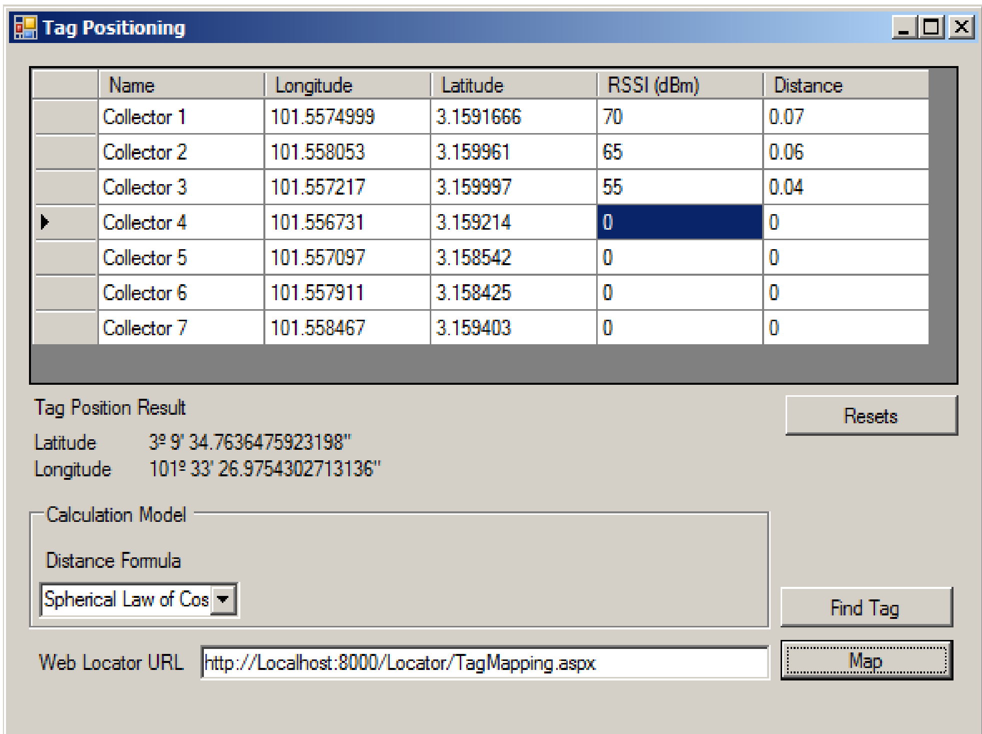
The values from Table 2 are sorted and the following 3 data has been selected as shown in Table 4 below;

Table 4: Result of 3 Data Collected

ID	CollectorID	TimeDetected	TagID	RSSI
1	Collector1	24/01/2010 13:26:19	Tag1	70
16	Collector2	24/01/2010 13:26:20	Tag1	65
17	Collector3	24/01/2010 13:26:20	Tag1	55

The value is entered in the "TagPositioning" software;

Figure 8: RSSI value gathered in "TagPositioning" Interface



And the results are collected;

The steps are repeated for all of the tags (blind nodes); the final results obtained are shown in the table 5 below;

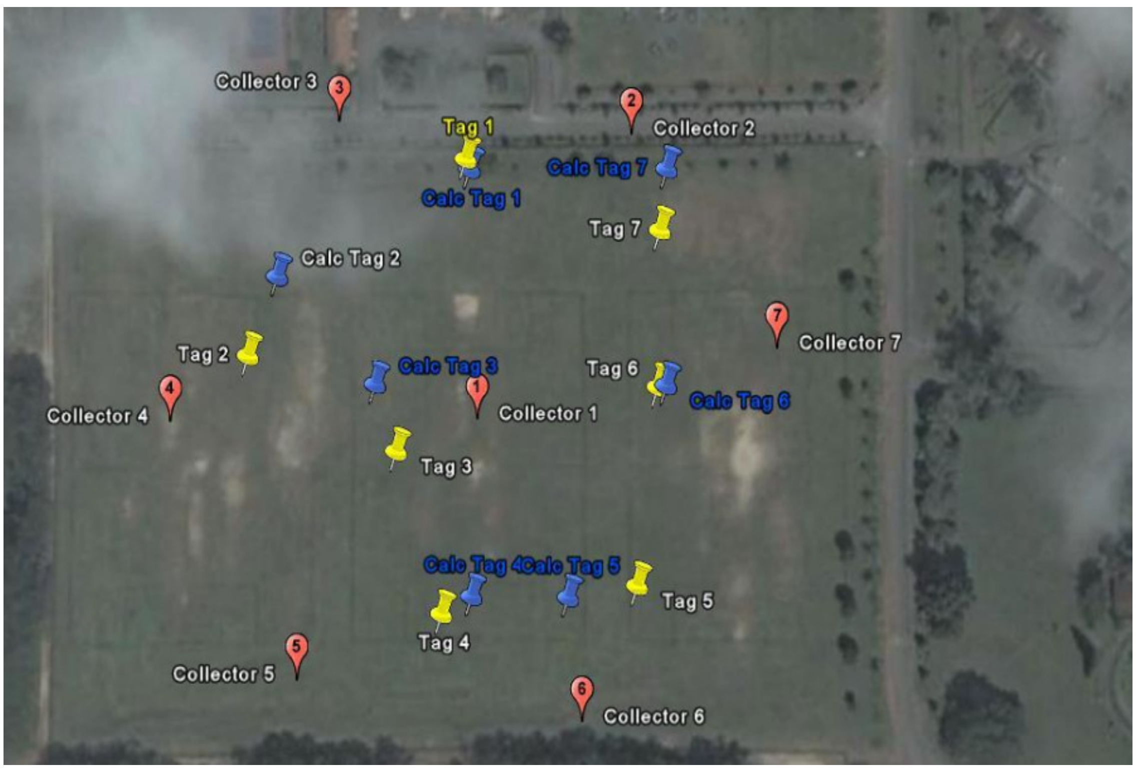
Table 5: Trilateration Results

Tag	Collector	RSSI	Tag Latitude	Tag Longitude
Tag1	Collector1	70	3.159657	101.557493
	Collector2	65		
	Collector3	55		
Tag2	Collector4	50	3.159559	101.556886
	Collector1	75		
	Collector3	75		
Tag3	Collector1	50	3.159078	101.557307
	Collector4	70		
	Collector5	70		
Tag4	Collector1	65	3.158564	101.557419
	Collector5	55		

	Collector6	60		
Tag5	Collector1	70	3.158722	101.557861
	Collector6 Collector7	55 80		
Tag6	Collector1 Collector6 Collector7	65 80 55	3.159142	101.558091
Tag7	Collector1 Collector2 Collector7	70 50 60	3.159667	101.558103

The final results are mapped on Google Earth [10] (as shown in Figure 9); the actual locations of the tags are also recorded during the test and the results are compared.

Figure 9: Google Mapping Result



Legend:

 — Actual Tag Positioning at the field

 — Calculated Tag Positioning at the field

 — Collector positioning at the field

6. Summary and Concluding Remarks

This paper provides an insight into the simplicity of implementing the node localisation in the ZigBee network using existing infrastructure and by using centralised algorithm as illustrated, the application can be built on top of the existing solution. This will provide an inexpensive positioning solution and add functionality to the Wireless Sensor network applications. Further implementation of the positioning application can be built using Service Oriented Architecture (SOA) based application model, which offers the flexibility of providing the location based service to an enterprise application, using the approach shown in this paper, an SOA based application can be built without tearing existing solution, providing an inexpensive location based functionality without having to rebuild existing solution.

Acknowledgment

The authors wish to express their sincere thanks to Kappa Solutions Sdn. Bhd. for providing the all the ZigBee devices and their technical information from Digi International to be used for this study.

References

[1] **Mannings, Robin**. *Ubiquitous positioning.* Norwood: Artech House, 2008.
[2] **Admundson, I., Sallai, J., Ketsoukos, X., & Ledeczi, A.** (2010). Radio Interferometric Angle of Arrival Estimation. *Wireless Sensor Networks: 7th European Conference* (p. 1). Portugal: Springer.
[3] **Rappaport, Theodore S.** *Wireless Communications: Principles and Practice.* New Jersey: Prentice-Hall Inc., 1996.
[4] **Doherty, L., Kristofer, P. S., & Laurent, E. G.** (2001). Convex Position Estimation in Wireless Sensor Network. Proceedings IEEE INFOCOM 2001: the conference on computer communications: proceedings: Twentieth annual joint conference of the IEEE Computer and Communications Societies (p. 1655). Alaska: IEEE Computer Society, IEEE Communications Society
[5] **Shang, Y., Ruml, W., Zhang, Y., & Fromherz, M.** (2003). Localization from Mere Connectivity. *The Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, (p. 210). Maryland.
[6] **Fitzpatrick, S and Meertens, L.** *Diffussion Based Localization.* s.l.: Private Communications, 2004.
[7] **Niculescu, D., & Nath, b.** (2003). Ad Hoc Positioning using AOA. *Proceedings of IEEE Infocom.* San Francisco: IEEE.
[8] **Dixon, John C**. Suspension Analysis and Computational Geometry: John Wiley & Sons Ltd, 2009
[9] **Wikipedia.** Geographic coordinate system. *Wikipedia, the free encyclopedia.* [Online] 2006. http://en.wikipedia.org/wiki/Geographic_coordinate_system.
[10] **Google Map Malaysia**, http://maps.google.com.my/