Implementation of Trilateration Based Localization Approach for Tree Monitoring in IoT

Naren Tada(✉), Tejas Patalia, and Shivam Trivedi

V.V.P. Engineering College, Rajkot, India
naren.tada@gmail.com, pataliatejas@rediffmail.com, shivtrivedi197@gmail.com

Abstract. The Internet of thing is widely used term as progression in innovation. It is a system of physical devices embedded with electronic motes, sensors, and software which empower these devices to exchange data through the internet. Moreover, to localize and monitor environmental conditions on the real time basis system may use Global Positioning System framework but it expends more energy as such system compromise resource constrain devices. In this research paper, we have concentrated to develop Global Positioning System independent localization algorithm for real-time tree monitoring. Our proposed algorithm comprises of two sub-algorithms which are Received Signal Strength Indicator and trilateration. Localization uses routing of data and it consumes more energy. Hence, we require energy efficient routing protocol. To fulfill the purpose we used (RPL) Routing Protocol for Low-Power and Lossy Networks routing protocol which is developed specially for low power and lossy networks. Simulation of the whole system is carried out In Contiki-OS with the help of built-in COOJA simulator.

Keywords: IoT · Environment monitoring · Localization · Contiki OS · RPL · COOJA · Wireless sensor network · Sensor network

1 Introduction

Trees are playing a crucial role in the ecosystem. Urban street trees has proven benefits such as providing oxygen, storing carbon dioxide, stabilizing soil, providing shelter and shades and food and raising property value etc. We think that urban landscape is a harsh place to thrive and survive. City foresters are trying hard to track and manage urban forests but almost 60% urban trees are grown on private property. This makes difficult to track down and manage each and every tree. Keeping in mind the importance of tree in life, Indian judiciary system have decided, once in five years, mandatory carrying out a tree census program which is also called as tree act 1975.

In April 2016, Pune Municipal Corporation started a high-tech tree census program to track the location, type and other details of every tree in the city area. In their high-tech project they are using GPS (Global Positioning System) & GIS (Geographic Information System) [1]. GPS is using 1.5 GHz high frequency; those radio waves could not penetrate through a dense medium. When we talk about the urban area, there are many high rise

N. Kumar and R. Venkatesha Prasad (Eds.): UBICNET 2019, LNICST 276, pp. 115–123, 2019.
https://doi.org/10.1007/978-3-030-20615-4_9

buildings so the sky becomes obscured and sensor barely sees the satellites, then it may be result in inaccurate reading. The goal of this paper is to localize blind nodes using low power routing. People has come up many routing proposal for low power and lossy network such as MEAL (Multi-hop Energy efficient leach) [2], LEACH (Low Energy Adaptive Clustering Hierarchy) etc. For the routing, we have used RPL protocol which was developed by IETF (Internet Engineering Tasks Force) for low power and lossy network [3]. With the best of our knowledge no one has come up with the solution for localizing tree with RPL routing protocol.

For localization three anchor nodes are used, anchor nodes are the nodes whose positions are known. They perform RSSI (received signal strength indicator) based trilateration to estimate the position of blind nodes in the deployed network [4]. In our work we have localize trees without using GPS and GIS system. We have modified RPL in such a manner so that we can calculate RSSI and pass this value through routing to the base station with the help of trilateration method.

Rest of the paper is organized is as follows, proposed work illustrate the overall idea of how system works, simulation of proposed work which illustrate the simulation platform and how proposed method is implemented in the given platform, the result and analysis we have presented result with some of the actual coordinates of estimated coordinates with some simulation error present in the estimated and finally conclude our work with the future scope for the improvements.

2 Proposed System

As trees are the static entity we need a system which gives us real-time data and precise location of the tree. After completion of first phase, sensors start monitoring tree. Sensor stores data and send it to the base station based on the information we get, we need to take action to regulated expected values (Figs. 2, 3 and 4).

A. Received Signal Strength Indicator

The distance estimation includes calculation of RSSI values from the anchor nodes and based on that value, the corresponding distance estimate is calculated by the blind node. The RSSI values are obtained by using CC2420 radio transceiver that is fitted in all the motes. The mathematical formulation is given as follows:

$$RSS = RSS_VAL + RSS_OFFSET[dBm] \tag{1}$$

Where RSS OFFSET is roughly −45[4] and RSS VAL is the power received by the C2420 radio transceiver. Utilizing RSSI value, the distance estimation is done as follows. Expecting that the transmission power Ptx, the path loss model, and the path loss coefficient are known, the receiver can use the received signal strength Prcvd to solve for the distance d in a path loss equation like;

$$d = \frac{(C * Ptx)}{Prcvd} * \alpha \tag{2}$$

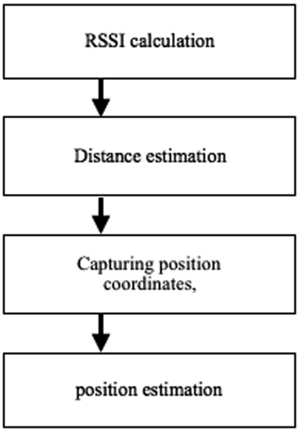


Fig. 1. Proposed system

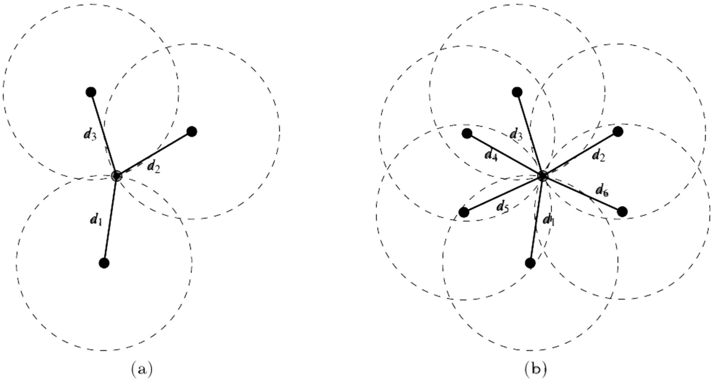


Fig. 2. Distance measurements and multilateration (Ranging circles). (a) Trilateration. (b) Multilateration.

Fig. 3. Small scale implementation of RPL in COOJA

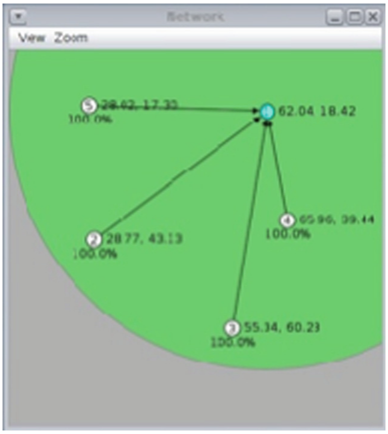


Fig. 4. Simulation of RSSI

B. Trilateration

In geometry, trilateration is the way toward deciding outright or relative areas focuses on estimation of separations, utilizing the geometry of circles, spheres or triangles. Trilateration involves measuring distance. Using anchor nodes, it locates blind nodes within their range. Multilateration [5] is used when the number of nodes are more than three. Below image illustrate the same.

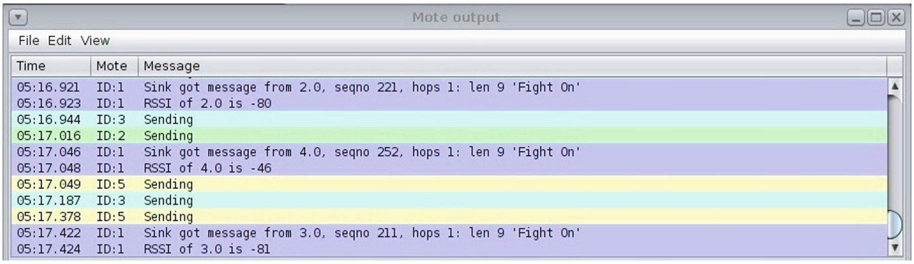
In the simulation node number 1 root node which is responsible for creating DODAG (Destination oriented directed acyclic graph) root. Other nodes are neighbor nodes. Node number 4 is directly in the transmission range of root node. Other nodes are not in direct range of root. Therefore, they need to make a connection with node number 4. Initially, the power consumption in RPL is very high due to all nodes are participating in the formation of stable DODAG. Once the DODAG is formed, there is no need to process data. Therefore, chronologically power consumption of entire topology will decrease. RPL topology is stable topology until any new node is added or any present node is removed from the topology.

3 Implementation and Simulation

All the simulations were performed using Contiki-OS v-2.7 and emulated sky motes [6]. The environment is set to unit disk graph medium (UDGM): distance loss for the deployment of sky motes in the network. Up to four nodes were deployed for the purpose of simulation that includes a blind node (also called as sink node) and three anchor nodes. The transmission range and interference range is set to 50 and 100 m respectively. The deployed anchor nodes should be within the transmission range of the blind node in order to send the packets.

A. RSSI Contiki Implementation

To estimate precise location using Trilateration method, we first calculate RSS value of each node. The distance estimation involves calculation of RSSI values from the anchor nodes and based on that value, the corresponding distance estimate is calculated by the blind node. The RSSI values are obtained by using CC2420 radio transceiver that is fitted in all the nodes (Fig 7).



Time	Mote	Message
05:16.921	ID:1	Sink got message from 2.0, seqno 221, hops 1: len 9 'Fight On'
05:16.923	ID:1	RSSI of 2.0 is -80
05:16.944	ID:3	Sending
05:17.016	ID:2	Sending
05:17.046	ID:1	Sink got message from 4.0, seqno 252, hops 1: len 9 'Fight On'
05:17.048	ID:1	RSSI of 4.0 is -46
05:17.049	ID:5	Sending
05:17.187	ID:3	Sending
05:17.378	ID:5	Sending
05:17.422	ID:1	Sink got message from 3.0, seqno 211, hops 1: len 9 'Fight On'
05:17.424	ID:1	RSSI of 3.0 is -81

Fig. 5. Mote output of RSSI

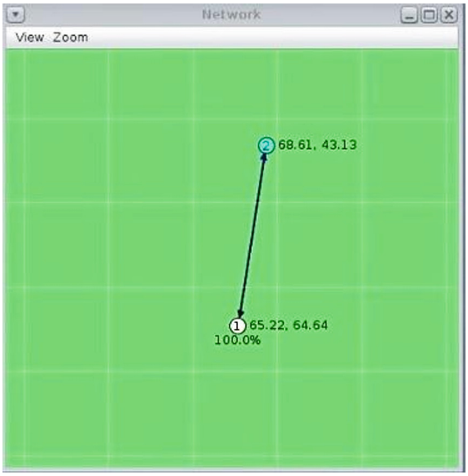
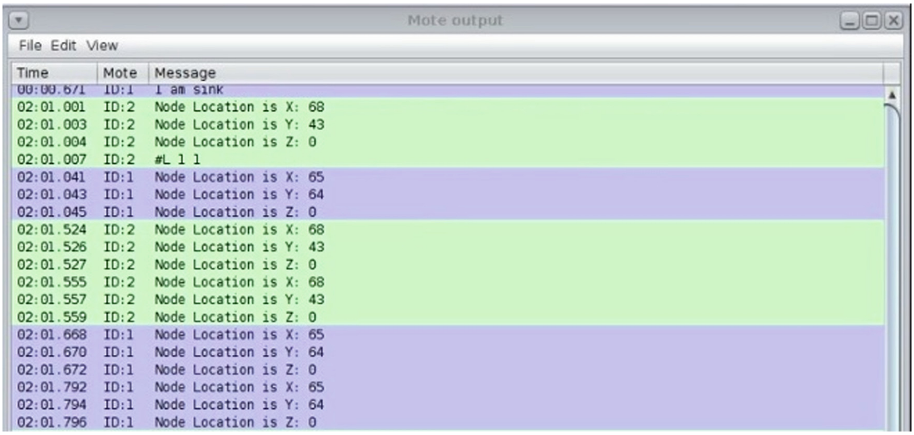


Fig. 6. Network topology of two nodes

Figure 5 shows the network of RSSI simulation. Node 1 is a root node and other nodes are neighbor nodes. RSSI is a line of sight communication; therefore, we cannot get the RSSI value of that particular node if any node is outside of the transmission range of root node. Here, in Fig. 5, all nodes are in the transmission range of root node so that we are able to get RSSI of each node.



Mote output

File Edit View

Time	Mote	Message
00:00.671	ID:1	I am sink
02:01.001	ID:2	Node Location is X: 68
02:01.003	ID:2	Node Location is Y: 43
02:01.004	ID:2	Node Location is Z: 0
02:01.007	ID:2	#L 1 1
02:01.041	ID:1	Node Location is X: 65
02:01.043	ID:1	Node Location is Y: 64
02:01.045	ID:1	Node Location is Z: 0
02:01.524	ID:2	Node Location is X: 68
02:01.526	ID:2	Node Location is Y: 43
02:01.527	ID:2	Node Location is Z: 0
02:01.555	ID:2	Node Location is X: 68
02:01.557	ID:2	Node Location is Y: 43
02:01.559	ID:2	Node Location is Z: 0
02:01.668	ID:1	Node Location is X: 65
02:01.670	ID:1	Node Location is Y: 64
02:01.672	ID:1	Node Location is Z: 0
02:01.792	ID:1	Node Location is X: 65
02:01.794	ID:1	Node Location is Y: 64
02:01.796	ID:1	Node Location is Z: 0

Fig. 7. Mote output for node position

RSSI refers to the transmitter power output as received by a reference antenna (receiver). Transmitter continuously sends packets to the receiver and receiver nodes continuously receive those packets. To calculate RSSI from those received packets we need proper function. In Contiki, We have packetbuf attr() which is used to create an outbound packet to store an inbound packet. When the driver reads out received frame, the RSSI observed during the reception will get copied to the packetbuf. Figure 5 shows the output of the RSSI value of the created network in Fig. 6. Root node which id node number 1 is getting output from the other nodes. Node number 2 and 3 are located far from the root node, therefore, we are getting high RSSI value for them which is −80 dbm and −81 dbm respectively. Node number 4 is located near to the root node; therefore, its RSSI value is high as compared to node number 2 and 3 which is −46 dbm

B. Getting Position of Nodes

We have created interfaces which can communicate between Contiki and COOJA. After adding mote interface selected C source file which calculates the position of nodes. Figure 6 shows the network topology of 2 nodes whose position is going to estimate. For the simplicity, we have chosen only 2 nodes. Node number 1 is a root node and node number 2 is a neighbor node. The simulation environment is 2D hence only X and Y coordinate will be computed. After hitting start in the simulator the script will run in the background and started getting the position of nodes. But as the compiler used in Contiki does not support floating point value we are unable to get exact floating value. The next thing is we are running a simulation in a 2D environment but the code is able to work in a 3D environment. That means the source code is designed to work in real time environment which captures all X, Y & Z coordinates. Due to simulation environment restrictions, we are only getting a value of X and Y.

C. Trilateration Algorithm

The term coordinate matrix stores the X & Y axis in an array, as we are using 2D environment for the simulation. Estimated distance matrix store the X & Y coordinates of a blind node [6].

The simulation was conducted at a specific 8 positions coordinates of the anchor nodes and resulting in actual and estimated coordinates of blind nodes. After estimating all position coordinates, the localization calculated as follows.

$$\text{Localization Error} = \sqrt{(X2 - X1)^2 + (Y2 - Y1)^2} \quad (3)$$

In the above equation (x1, y1) is actual position coordinates of a blind node and (x2, y2) is the estimated position coordinates of the blind node. Obtain values are finalize after the simulation was repeated for more than 50 times. The overall localization error was obtained by finding the average of all the obtained localization error during the simulation. For the simulation of trilateration algorithm, we have created a simulation using 8 sky motes in COOJA simulator. In Fig. 8, 8 sky motes [7] are shown using 10-m boundary grid. In the deployed network we require to define 3 anchor nodes to start the trilateration process, hence node 1(85.31, 41.94), node 6 (64.24, 37.49), and node 8 (89.54, 15.23) are anchor node and will try to estimate the position of remaining blind nodes.

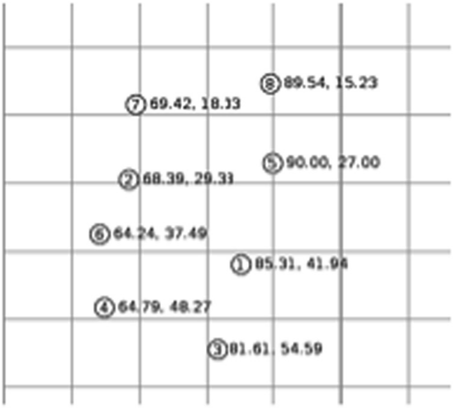


Fig. 8. Network topology for proposed system

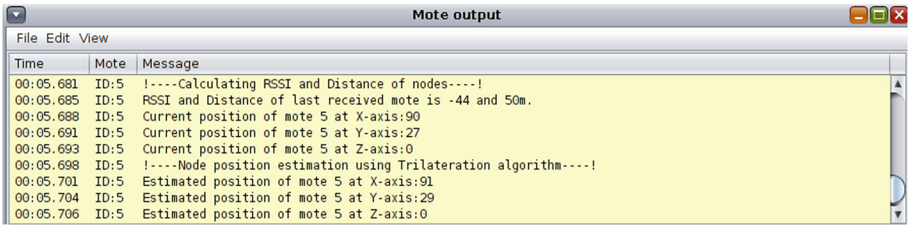


Fig. 9. Mote output of proposed algorithm

Table 1. Simulation Results and Localization Error

Mote No.	Actual Coordinates	Estimated Coordinates	Localization Error (m)
1	(85.31, 41.94)	(85.31, 41.94)	0.00
2	(68.39, 29.33)	(67.39, 28.33)	1.4142
3	(81.61, 54.59)	(80.61, 55.59)	1.4142
4	(64.79, 48.27)	(66.79, 46.27)	2.2828
5	(90.00, 27.00)	(91.00, 29.00)	2.236
6	(64.24,37.49)	(64.24,37.49)	0.00
7	(69.42, 18.33)	(70.42, 19.33)	3.6055
8	(89.54, 15.23)	(89.54, 15.23)	0.00

4 Result and Analysis

The steps involved in the proposed system are shown in Fig. 1. The very first step is to calculate the RSSI value of every node. After this, algorithm calculates the distance of the deployed nodes. For that 3 pre-defined anchor nodes send the packets to the blind nodes, and based on the received power RSSI value will be calculated. Based on the calculated RSSI value blind node estimates its distance from the anchor node. In Fig. 9, entire output is shown for the mote number 5. Energy efficiency consider as per the author has stated in [8].

The position coordinates of anchor nodes will be sent to the blind nodes. Utilizing the anchor nodes coordinates, blind nodes position estimation will be performing using trilateration algorithm. For the position estimation of blind nodes, we must require estimated distance and position coordinates of anchor nodes. Here, for the distance estimation, we are utilizing RSSI which are prone to error because of multipath propagation in the real environment. Actual position coordinates of the blind nodes and estimated position coordinates are shown in the table below with localization error. The simulation was repeated more than 10 times with fixed coordinates and the overall localization error was obtained by finding the average of all the obtained localization error during the simulation. The overall localization error obtained after performing all the simulation rounds is 1.37 m. Yet, the obtained localization error is acceptable in the real-time environment. Simulation of last round is shown in Table 1 for all the 8 deployed nodes.

5 Conclusion and Future Work

This proposed work is built for the purpose GPS independent localization scheme. By utilizing this scheme we can determine position coordinates of a blind node on the based on the distance estimation and position coordinates of the anchor nodes deployed in the network. For the simulation, we have utilized Contiki-OS with the help of COOJA simulator. After getting final simulation result the overall localization error obtained is about 1.37 m. In the localization field, this margin is not accurate enough.

Yet, it is acceptable in the environment in which this system will be going to use. This work can be additionally reached out later on by enhancing the positional accuracy of a blind node by considering distance estimations from more than 3 nodes; this is about utilizing multilateration strategy. The positional accuracy of a sensor node is relying upon the precision of the distance estimation based on the RSSI value.

References

1. P. M. Corporation, Tree census: Pune municipal corporation (2018). https://pmc.gov.in/en/tree-census-1. Accessed 15 Mar 2018
2. Patalia, T., Tada, N., Patel, C.: MEAL based routing in WSN. In: Satapathy, S.C., Das, S. (eds.) Proceedings of First International Conference on Information and Communication Technology for Intelligent Systems: Volume 2. SIST, vol. 51, pp. 109–114. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-30927-9_11
3. Winter, T., et al.: RPL: IPv6 routing protocol for low power and lossy networks. Roll Working Group (2011)
4. Chen, C.Z., Xia, F., Huang, T., et al.: A localization method for the Internet of Things. J. Supercomput. **63**, 657 (2013). https://doi.org/10.1007/s11227-011-0693-2
5. Asmaa, L., Hatim, K.A., Abdelaaziz, M.: Localization algorithms research in wireless sensor network based on multilateration and trilateration techniques. In: 2014 Third IEEE International Colloquium in Information Science and Technology (CIST), pp. 415–419. Tetouan (2014)
6. Contiki: The open source operating system for the internet of things (2018). http://www.contiki-os.org/. Accessed 2 Feb 2018
7. Chipcon products from texas instruments (2018). http://web.stanford.edu/class/cs244e/papers/cc2420.pdf. Accessed 10 Feb 2018
8. Naren, T., Dishita, B.: TVAKSHAS - an energy consumption and utilization measuring system for green computing environment. In: Kumar, N., Thakre, A. (eds.) UBICNET 2017. LNICST, vol. 218, pp. 37–45. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73423-1_4