A Hop-Count Based Positioning Algorithm for Wireless Ad-hoc Networks

Sarita Gurung [1] · **A. K. M. Mahtab Hossain** [*2] ·

Kanchana Kanchanasut[3]

Abstract We propose a range-free localization algorithm for a wireless ad-hoc network utilizing the hop-count metric's ability to indicate proximity to anchors (i.e., nodes with known positions). In traditional sense, hop-count generally means the number of intermediate routers a datagram has to go through between its source and the destination node. We analytically show that hop-count could be used to indicate proximity relative to an anchor node. Our proposed algorithm is computationally feasible for resource constrained wireless ad-hoc nodes, and gives reasonable accuracy. We perform both real experiments and simulations to evaluate the algorithm's performance. Experimental results show that our algorithm outperforms similar proximity based algorithms utilizing Received Signal Strength (RSS) and Expected Transmission Count (ETX). We also analyze the impact of various parameters like the number of anchor nodes, placements of anchor nodes and varying transmission powers of the nodes on the hop-count based localization algorithm's performance through simulation.

Keywords

Positioning System, Ad-hoc Networks, Localization Algorithm, Proximity Based Localization, Hop-count, Optimized Link State Routing (OLSR).

1 Introduction

A wireless ad-hoc network is a self-reliant collection of wireless devices (tablets, laptops, smart phones, sensors, etc.) that communicate with each other without wired connection, and collaborate in a distributed

---

[*]Corresponding Author – this work was done while he was working at intERLab, Asian Institute of Technology, Thailand.

[1]intERLab, Asian Institute of Technology, Thailand, E-mail: sarita@ait.ac.th

[2]Department of Computer Science, University College Cork, Ireland, E-mail: m.hossain@cs.ucc.ie

[3]intERLab, Asian Institute of Technology, Thailand, E-mail: kanchana@ait.ac.th

manner. They provide network functionalities without using any complex infrastructure. Due to the rapid proliferation of cheap, widely available and low power sensors, the ad-hoc networks are becoming more ubiquitous where the sensors are used in health-care, wildlife tracking, battle field surveillance, rescue operations in post disaster scenario and forecasting, etc [1]. Many of these applications require the location information [2].

The Global Positioning System (GPS) is one of the most popular measures of obtaining location information [3]. However, it is not a cost-effective approach to equip each ad-hoc node with GPS receivers [4–6]. Due to weak signal reception in indoor environments, GPS is known to have poor performance. On the contrary, radio positioning techniques support any wireless device, integrate well with existing data networks, and operate both outdoors and indoors. The ability to provide cost-effective solution for wireless ad-hoc networks is making this class of localization techniques popular. Traditional approaches utilize the Received Signal Strength (RSS) perceived by the transceiver to estimate the location of a wireless device equipped with radio technology. The RSS measurements between a node and the *anchors* (i.e., nodes with known locations) are generally converted into range (i.e., distance) estimates using RF propagation models [7], and then multi-lateration algorithms [8] are applied to calculate position of the node. This approach of localization is called range-based. On the other hand, range-free schemes do not require point-to-point distance or angle measurements, thereby incurring low overhead and minimum cost avoiding use of additional hardware. In this approach, the successful reception of radio messages from an anchor indicates whether the node is *connected* to that particular anchor or not (i.e., whether the node is close to the anchor in space). Subsequently, some connectivity-based algorithms like Centroid [4], APIT [9], etc. are applied for location estimation. Range-free schemes find more applicability in ad-hoc networks comprising of inexpensive nodes despite giving coarser accuracy. This is due to the fact that range-based schemes require complex hardware (e.g., directional antennae) in order to yield accurate result.

In this article, we aim for a GPS-less range-free positioning algorithm that requires little or no modification on a mobile ad-hoc network's (MANET's) protocol to be operational. We utilize the popular hop-count metric in order to locate a MANET node that is readily available from its routing table. The traditional RSS metric is generally not easily available from APIs of popular ad-hoc routing protocols such as DSR [10], DSDV [11], OLSR [12], AODV [13], etc. Thus, extracting RSS incurs overhead while developing localization solution for an ad-hoc node running these protocols. RSS measurements are also affected by interferences, multi-path, etc. which has impact on the localization algorithm's performance based on it.

In traditional sense, hop-count generally means the number of intermediate routers a datagram has to go through between its source and the destination node. The intermediate routers are chosen in such a way that the datagram reaches the destination by traveling through the minimum number of routers possible. Therefore, hop-count could be utilized to approximate the separation between two nodes as revealed in [14]. However, [14] requires modifications in the control messages exchanged in the Distance Vector (DV) routing protocol to approximate the separation between two nodes. We achieve the same goal without incurring any modifications to the existing routing protocols. Our algorithm just needs to access the routing table of a node which is readily available from the APIs of any routing daemon.

In this article, we propose a range-free localization algorithm for a wireless ad-hoc network utilizing the hop-count metric's ability to indicate proximity to anchors. First, we analyze and prove hop-count metric's proximity indication ability relative to an anchor node. Second, we develop a hop-count based localization algorithm, and compare its performance with other range-free algorithms targeted at wireless ad-hoc networks (e.g., ETX and RSS based algorithms). Third, we analyze the impact of various factors like the number of anchor nodes, varying transmission power of the nodes and placements of the anchor nodes on our localization algorithm's accuracy. We conduct both simulations and real experiments to analyze our localization algorithm's performance which shows favorable outcomes. This strengthens our claim that hop-count could be utilized as a location estimation parameter for a range-free positioning solution for wireless ad-hoc networks. We choose hop-count as a location estimation parameter for ad-hoc networks for the following reasons: i) it could be easily obtained from APIs of all the popular ad-hoc routing protocols, e.g., DSR, DSDV, AODV, OLSR, etc., ii) it could be used as an efficient proximity indication metric which is analyzed and proved in Section 3, and iii) it is a network layer metric so could be deployed in different machines/devices with more convenience compared to RSS [15].

The rest of the paper is organized as follows. We provide an overview of related works in Section 2. Section 3 introduces the hop-count metric, and analyzes how hop-count could be used as location estimation parameter. In Section 4, we explain our hop-count based localization algorithm. We present the experimental findings supporting our claims in Section 5. Finally, Section 6 depicts the conclusions drawn, and our future work.

2 Related Work

Range-based localization techniques rely on specialized hardware (e.g., ultrasound or infrared receivers, RF tags, etc.) for localization purpose [16–19]. They provide fine-grained location information, e.g., Active

BAT [17] system is shown to have 2 cm average accuracy. On the contrary, range-free schemes [4, 9, 20, 21] provide much coarser accuracy but require no infrastructure for positioning system. Thus, they are found to be applicable in localizing inexpensive nodes in wireless ad-hoc networks as discussed in the previous section. Generally, various link layer metrics, e.g., RSS [4, 22], ordered sequence of RSSs [20], Signal Strength Difference (SSD) [23], Signal-to-Noise Ratio (SNR), signal quality (e.g., Expected Transmission Count [15]), etc. are utilized as location estimation parameters in the various range-free algorithms.

DV-Hop [14], Amorphous [24] and Self-Configurable [25] localization algorithms are proposed mainly for ad-hoc networks that provide coarse-level accuracy. They use number of hops (a network layer metric) to reach a node as a measure of its distance from it. An algorithm based on hop-count quantization is proposed in [26] which transforms integer-valued hop-count into real number. The transformed hop-count is then used for localization purpose. In [27], the hop-count based weighted centroid localization algorithm is improved by adding the node degree on the paths to the referenced anchors into the weights. Merkel *et al.* uses the whole neighbor set information of a particular node to approximate its distance to each neighbor geometrically [6]. To calculate its distance to an anchor node, all distances along the shortest path between a node and the anchor node need to be aggregated.

The hop-count based approaches generally provide simplistic localization solutions; however, the accuracy is affected by node density. Thus, a multi-robot approach [28] utilizes mobility of robots for increasing the node density to improve the hop-count based localization's accuracy. The robots cooperate with sensor nodes for the most appropriate robot selection process to move to the place with low node density. The hop-count based localization schemes generally use connectivity information from anchor nodes. In [29], a new improvised technique is introduced which uses geometric constraints to narrow down a likely area where a sensor node is expected to lie. The hop-count based localization techniques are seen to perform quite well when nodes have high density and uniform distribution. Density-aware hop-count localization (DHL) [30] improves accuracy of location estimate when nodes are distributed non-uniformly. In general, the product of hop-count and transmission range gives the distance between reference node and any node , however, this is likely to overestimate the distance, in sparse networks. The concept of each hop traversed is equivalent to the maximum range distance is not strictly followed in sparse networks. Thus, the DHL uses range ratio i.e., ratio of expected hop-count and transmission ranges to adjust the accumulated hop-count. Additionally, the DHL classifies the node density into low, medium and high categories depending on number of neighbor nodes and the range ratio value differs with density.

In the following, we discuss two range-free algorithms that utilize RSS and Expected Transmission Count (ETX) as an indication of proximity to anchor nodes. Our hop-count based localization scheme is closely related to these works in the way it uses hop-count as proximity measure. Ecolocation [20] utilizes the signal strength measurements perceived by a node from two anchors, and picks the one with the stronger signal to be *closer* than the other. If $P(d_x)$ and $P(d_y)$ denote the RSSs from anchor$_x$ and anchor$_y$, which are at distances d_x and d_y from the target node, respectively, then a constraint of Ecolocation is defined as,

$$P(d_x) > P(d_y) \Rightarrow d_x < d_y. \tag{1}$$

ETX based positioning [15] follows similar approach and picks the anchor with lower ETX value to be closer than the other. ETX metric was first proposed in [31] to model the expected number of transmissions needed for sending a unicast packet over a link, including retransmissions. It is calculated by taking into account the successfully transmitted and received packets between a node and its neighbor within a certain time period. The ETX of a path is given by the sum of the ETX for each link in the path. If ETX$_x$ and ETX$_y$ denote the ETX values from anchor$_x$ and anchor$_y$, which are at distances d_x and d_y from the target node, respectively, then a constraint for ETX-based positioning is defined as,

$$\text{ETX}_x < \text{ETX}_y \Rightarrow d_x < d_y. \tag{2}$$

It is based upon the assumption that the link quality is inversely proportional to distance.

3 Hop-count metric: Overview and Analysis

We briefly discuss the hop-count metric in Section 3.1, and then show how it could be utilized as a proximity measure for localization techniques in Section 3.2.

3.1 Overview of hop-count metric

In traditional sense, hop-count generally means the number of intermediate routers a datagram has to go through between its source and the destination node. For example, if a source S sends a datagram to destination D, and the datagram travels through M routers before it reaches its destination, then the hop-count between S and D is M. The intermediate routers are chosen in such a way that the datagram reaches the destination by traveling through the minimum number of routers possible. For example, if there are two paths from S to D consisting of M and N hops, respectively where $M < N$, then the datagram would take the path with M routers. There are many routing protocols which help to determine this *efficient* path, e.g.,

RIP, OSPF [32], etc in traditional network (Internet) or DSR [10], AODV [13], OLSR [12], etc. in ad-hoc networks.

The routing protocols generally try to find the *smallest* hop-count path between the source and destination. However, there is a subtle difference between the definition of hop-count w.r.t. traditional networks like Internet and wireless ad-hoc networks. We know that in ad-hoc networks, all the nodes take on the responsibility of forwarding each other's datagrams. The hop-count metric of wireless ad-hoc network denotes the number of intermediate nodes between S and D including the source itself. So the hop-count of wireless ad-hoc networks is generally one higher than its traditional network counterpart for the same number of intermediate nodes between S and D. Therefore, the minimum hop-count between any two nodes in wireless ad-hoc networks is 1 when they are within communication range of each other. We adopt this definition of hop-count for our analysis since our localization technique is targeted at wireless ad-hoc networks.

3.2 Hop-count as a measure of proximity

In the field of wireless networking, the multi-hop radio network is often modeled with the help of unit disk graphs [33]. In this model, the nodes are assumed to be scattered over a Euclidean plane having identical (unit) communication range. The vertices of the graph represent the wireless nodes and an edge between the two vertices exists if the Euclidean distance between them is at most 1. In other words, the nodes can communicate only if they are within mutual communication range. Clearly, this behavior is neatly captured by the unit disk graph model, and it has become a standard when studying ad-hoc and sensor networks, especially for analysis purpose.

We also assume the unit disk graph model for the ad-hoc nodes of our analysis. First, let us consider a scenario where anchor node B is within the communication range of target node A. If HC_{AB} denotes the hop-count between nodes A and B, then from the definition of hop-count of the previous section, we obtain,

$$HC_{AB} = 1 \tag{3}$$

Now, consider another anchor node C which is inside the communication range of node B but outside the communication range of A. Since all the nodes' communication ranges are assumed to be similar and the anchors B and C are inside and outside the communication range of the node A, respectively, we have,

$$d_{AB} < d_{AC} \tag{4}$$

where, d_{AB} and d_{AC} are the distances of the node A from B and C, respectively. Since node C is inside the communication range of node B then,

$$HC_{\mathrm{BC}} = 1 \tag{5}$$

Using (3) and (5), the hop-count between nodes A and C can be denoted as,

$$HC_{\mathrm{AC}} = HC_{\mathrm{AB}} + HC_{\mathrm{BC}} = 1 + 1 = 2 \tag{6}$$

Combining (3) and (6), we obtain,

$$HC_{\mathrm{AB}} < HC_{\mathrm{AC}} \tag{7}$$

Both the constraints (4) and (7) are always satisfied for the scenario when nodes B and C are inside and outside of A's communication range, respectively. Hence, the hop-count could be used as a proximity measure for a multi-hop ad-hoc networks. In other words, just knowing the hop-count metric between nodes A & B, and nodes A & C, we could have inferred node A is actually closer to node B compared to C. We formally denote this information as the following constraint:

$$HC_{\mathrm{AB}} < HC_{\mathrm{AC}} \Rightarrow d_{\mathrm{AB}} < d_{\mathrm{AC}}. \tag{8}$$

In the above analysis, we considered a simplistic scenario when one anchor is within the communication range of the target node, while the other is not. In a practical context, both the anchors might be multiple hops away (*equal* or *unequal* number of hops) from the target node. These scenarios are considered next to inspect whether hop-count could still be considered as an indication of the proximity to the anchor nodes.

3.2.1 Anchors multiple hops away

We derive constraint (8) for the scenario when both anchors are multiple hops away from the target node. Let us suppose anchors B and C are m and n hops away from node A, respectively, where $m < n$. This implies, $HC_{\mathrm{AB}} = m < HC_{\mathrm{AC}} = n$.

We prove the constraint $HC_{\mathrm{AB}} < HC_{\mathrm{AC}} \Rightarrow d_{\mathrm{AB}} < d_{\mathrm{AC}}$ using induction.

If $m = 1$, and $n = 2$, the basis step of induction reflects the scenario mentioned in the previous section. Following similar analysis as before, we can see that $d_{\mathrm{AB}} < d_{\mathrm{AC}}$. $d_{\mathrm{AB}} \geq d_{\mathrm{AC}}$ would have resulted in $HC_{\mathrm{AC}} = n = 1$ based on our unit disk communication model assumption.

The hypothesis step yields $HC_{\mathrm{AB}}(m) < HC_{\mathrm{AC}}(n) \Rightarrow d_{\mathrm{AB}} < d_{\mathrm{AC}}$. Now, we have to prove $HC_{\mathrm{AB}}(m+1) < HC_{\mathrm{AC}}(n+1) \Rightarrow d_{\mathrm{AB}} < d_{\mathrm{AC}}$. The letters inside the parenthesis indicates the hop-count numbers between the nodes. Assuming the maximum possible separation between the nodes A and B, we have, $d_{\mathrm{AB}} = mr$

where r denotes the communication radius of each node. From the hypothesis, $d_{\mathrm{AC}} = mr + \delta > mr = d_{\mathrm{AB}}$ where $\delta > 0$. Subsequently, we obtain, $d_{\mathrm{AC}} = (m+1)r + \delta > (m+1)r = d_{\mathrm{AB}}$ which validates the inductive step. Therefore, the constraint (8) is satisfied for the scenario when the two anchors are *unequal* number of hops away from the target.

3.2.2 Anchors equal hops away

In this section, we inspect whether hop-count could provide proximity measure when both the anchors are equal hops away from the target. Let us assume nodes B and C are m and n hops away from node A, respectively where $m = n$. This implies, $HC_{AB} = HC_{AC} = m = n$.

Fig. 1: A simple topology with three nodes – all the nodes are within each other's communication range. The weights on the edges represent the ETX values of the links.

It can be easily seen that nodes equal hops away from the target does not conform to the condition that they are equal distances away, i.e.,

$$HC_{\mathrm{AB}} = HC_{\mathrm{AC}} \nRightarrow d_{\mathrm{AB}} = d_{\mathrm{AC}}.$$

In Fig. 1, all the nodes are assumed to be within each other's communication range, i.e., each node is 1-hop away from the other. However, their distance separations are not equal. Thus, hop-count is unable to provide proximity measure accurately when both the anchors are same number of hops away from the target. Since we aim at providing coarse accuracy, nodes equal hops away are assumed to be at equal distances in our localization algorithm. Consider the scenario in Fig. 2 where one anchor is almost coinciding with the target while the other is at the end of its communication radius. The larger the communication radius, the more inaccurate our assumption of equal distance will be.

Fig. 2: Communication radius's effect on proximity measure using hop-count when nodes are equal hops away.

This assumption of equality constraint only results in our algorithm's inability to utilize finer information to estimate location, i.e., our algorithm fails to differentiate that two anchor nodes equal hops away might be at different distances from the target node. We will explain how this phenomenon affects our localization algorithm's performance in Section 4.1 with an example scenario.

4 Hop-count based Localization Algorithm

Our hop-count based localization algorithm is modeled after Ecolocation [20] which utilizes the ordered sequence of RSS measurements to estimate the locations of inexpensive sensor nodes. The Ecolocation algorithm has been briefly described in Section 2.

We assume the existence of *anchor* nodes which acquire its position information with GPS or some other measures. The target node's location is estimated with the help of its relative distance information to these anchor nodes which is approximated via the hop-count metric. Section 3.2's analysis revealed that hop-count could be used as a relative proximity indication parameter between two anchor nodes (see Eq. (8)).

Suppose node A needs to be located which is at distances d_{AB} and d_{AC} from the anchor nodes B and C, respectively. Subsequently, the relative distance indication constraint for anchors B and C w.r.t. node A using hop-count metric could be represented with (8).

In our localization algorithm, first, calculation for the set of constraint for each grid point is done using the *right hand side* (RHS) of (8) for each pair of anchor nodes. Only the locations of anchor nodes are required in this phase – no hop-count metric collection surveys are necessary. For ease of implementation, let us represent the distance constraint of a particular grid (k, l) for anchors B and C by $M_{\mathrm{B} \times \mathrm{C}}^{k,l}$, where

$$M_{\mathrm{B} \times \mathrm{C}}^{k,l} = \begin{cases} 1 & \text{if } d_{AB} < d_{AC}, \\ -1 & \text{if } d_{AB} > d_{AC}, \\ 0 & \text{if } d_{AB} = d_{AC}. \end{cases} \tag{9}$$

Subsequently, considering the distance constraint for each pair of anchors we can come up with the matrix $M_{\beta \times \beta}^{k,l}$ that has β rows and columns where β is the number of anchor nodes. Each element of the matrix represents the grid's relative distance information between two anchor nodes, i.e., the RHS of constraint (8).

During location determination phase, the hop-count metric is being collected from the target node to each anchor. Let the hop-count constraint for anchors B and C representing the *left hand side* (LHS) of (8) is denoted by $HC_{\mathrm{B} \times \mathrm{C}}$, where

$$HC_{\mathrm{B} \times \mathrm{C}} = \begin{cases} 1 & \text{if } HC_{AB} < HC_{AC}, \\ -1 & \text{if } HC_{AB} > HC_{AC}, \\ 0 & \text{if } HC_{AB} = HC_{AC}. \end{cases} \tag{10}$$

As before, we can formulate the hop-count matrix $HC_{\beta \times \beta}$ considering the hop-count constraint for each pair of anchors. Note that, this matrix has similar properties as $M_{\beta \times \beta}^{k,l}$. Finally, the hop-count matrix, $HC_{\beta \times \beta}$, and the distance matrices $M_{\beta \times \beta}^{k,l}$ of all grid points (k, l) are matched against each other to find the location of the target node. More specifically, only the upper/lower triangular portion above/below the diagonal entries

of both need to be matched. The centroid of the grid points with the maximum number of constraints match

is returned as the location estimate.

Algorithm 1 Hop-count based localization algorithm

1: $maxConstraintMatch \leftarrow 0;$
2: $(x, y) \leftarrow (0, 0);$
3: $count \leftarrow 0;$
4: **for** each grid point (k, l) in the testbed **do**
5:     **for** each anchor node $i(\rightarrow 1......\beta)$ **do**
6:         $d_i^{k,l} \leftarrow ((P_{ix} - k)^2 + (P_{iy} - l)^2)^{1/2} \times \lambda;$
7:     **end for**
8: **end for**
9: generate distance constraint matrix $M_{\beta \times \beta}^{k,l}$ based on $d_i^{k,l}$
10: generate hop-count matrix $HC_{\beta \times \beta}$
    {Count number of constraint matched at each grid point and find out maximum number of constraint matched over all grid points.}
11: **for** each element (m, n) $(n > m)$ in $M_{\beta \times \beta}^{k,l}$ **do**
12:     **if** $M_{\beta \times \beta}^{k,l}(m, n) = HC_{\beta \times \beta}(m, n)$ **then**
13:         $matchCount^{k,l} \leftarrow matchCount^{k,l} + 1;$
14:     **end if**
15: **end for**
16: **if** $matchCount^{k,l} > maxConstraintMatch$ **then**
17:     $maxConstraintMatch \leftarrow matchCount^{k,l}$
18: **end if**
    {Look for grid points where the maximum number of constraints are matched and the final location estimate is centroid of those grid points.}
19: **for** each grid point (k, l) **do**
20:     **if** $matchCount^{k,l} = maxConstraintMatch$ **then**
21:         $(x, y) \leftarrow (x + i, y + j);$
22:         $count \leftarrow count + 1;$
23:     **end if**
24: **end for**
25: **return** $(\frac{x}{count}, \frac{y}{count})$ {Location Estimate}

We present the pseudo code of our hop-count based localization algorithm inside Algorithm 1. The inputs

to the algorithm are i) the number of anchor nodes (β), ii) the locations of anchors (P_{ix}, P_{iy}) where $i = 1 \ldots \beta$,

iii) the hop-count values from target node to each anchor HC_i where $i = 1 \ldots \beta$, iv) the localization area

dimension $(\gamma \times \gamma)$, and v) the distance between two consecutive grid points, i.e., scaling factor λ. The final

output of the algorithm is the location estimate of the target node.

If a wireless link is not available between any anchor and the target node then the number of anchor

nodes (β in Algorithm 1) does not include those anchors. In other words, only reachable anchors from the

target node (either directly or via multi-hop) are considered in the algorithm.

4.1 Illustrative Example of Our Localization Algorithm

In this section, we explain how the proposed algorithm works. We consider a 5×5 square grid testbed shown in Fig. 3 for illustration purpose. Three anchor nodes A, B and C marked with white circles are placed at locations (1,4), (4,4) and (4,0), respectively. The target node D marked with a black circle is located at (5,2). For simplicity, we assume the hop-count value between two consecutive horizontal and vertical grids is just 1-hop. Therefore, it can be seen that node D takes 6 hops to reach anchor A, i.e., $H_{\mathrm{DA}} = 6$ while it requires 3 hops to reach anchors B and C, i.e., $H_{\mathrm{DB}} = H_{\mathrm{DC}} = 3$. The estimated location of the target by our algorithm is shown as a star in Fig. 3.

Fig. 3: 5×5 square grid testbed – the three anchors A, B and C are marked as white circles while the target node D is marked as a black circle. The target's estimated location is shown as a star.

Utilizing (10) with the above hop-count values, the matrix $HC_{3\times3}$ can be denoted as,

$$HC_{3\times3} = \begin{array}{c} \\ A \\ B \\ C \end{array}\begin{array}{c} A \quad B \quad C \\ \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \end{array}$$

Before obtaining $HC_{3\times3}$, our algorithm would have already generated the distance constraint matrix $M_{3\times3}^{k,l}$ for all grid points (k,l) using (9). For example, the distance constraint matrix for grid points, $(3,2)$, $(4,2)$ or $(5,2)$ can be found as,

$$M_{3\times3}^{3,2} = M_{3\times3}^{4,2} = M_{3\times3}^{5,2} = \begin{array}{c} \\ A \\ B \\ C \end{array}\begin{array}{c} A \quad B \quad C \\ \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \end{array}$$

According to our algorithm (line 11–17), only the upper triangular portion above the diagonal entries of $HC_{\beta\times\beta}$, and $M_{\beta\times\beta}^{k,l}$ of each grid (k,l) are being matched. The number of matched constraints for each grid are then stored. Now, line 19–24 identifies the grids where the number of matched constraints are maximum, and ultimately returns the centroid of these grid locations at line 25. In our example, it is found that the maximum number of constraints (i.e., 3) are matched at locations (3,2), (4,2) and (5,2). So the final location estimate of the target node D is obtained as $(4,2)$. Note that, it is sufficient only to compare the upper or lower triangular portion above or below the diagonal entries respectively, since they reflect the same information. The diagonal entries will always be 0 for both matrices.

In the above example, the anchors (B and C) that are equal hops away from the target represent that they are equal distances away as well. Now, let us investigate the scenario when this constraint is not satisfied.

Suppose node B is moved a bit farther away from the target, i.e, above its previous location $(4, 4)$ which yields $d_{DB} > d_{DC}$. Consequently, the distance constraint matrix for grid points, $(3, 2)$, $(4, 2)$ or $(5, 2)$ changes to,

$$M_{3\times3}^{3,2} = M_{3\times3}^{4,2} = M_{3\times3}^{5,2} = \begin{array}{c} \\ A \\ B \\ C \end{array} \begin{array}{ccc} A & B & C \\ \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix} \end{array}.$$

Note that, now only 2 entries of the upper triangular portion above the diagonal entries could be matched with $H_{3\times3}$ unlike 3 entries as before. Additionally, searching all the grid points, another 8 grids could be found where the maximum number of constraints (i.e., 2 in this case) are matched as well. The final location estimate will be $(4.09, 1.36)$ which is the centroid of all 11 grid points. We notice that our algorithm's inability to differentiate that two anchor nodes equal hops away might be at different distances actually degrades the localization accuracy as revealed in Section 3.2.2.

4.2 Complexity Analysis

We argue that the algorithm is computationally feasible to be implemented inside the resource constrained MANET nodes. Let us analyze the upper bound on the computational costs for implementing this technique which is similar to Ecolocation [20]. Calculating the number of constraints matched at each grid point and identifying the maximum number of constraints matched over all grid points costs $O(\frac{\gamma^2\beta^2}{\lambda^2})$ time and $O(\frac{\gamma^2\beta^2}{\lambda^2} + \beta^2)$ space, respectively. Searching for the maximum number of constraint matches require $O(\frac{\gamma^2}{\lambda^2})$ time and $O(1)$ space. Therefore, the time and space complexities of the hop-count based algorithm could be denoted as, $O(\frac{\gamma^2\beta^2}{\lambda^2})$ and $O(\frac{\gamma^2\beta^2}{\lambda^2} + \beta^2)$, respectively.

Fig. 4: Our L-shaped experimental testbed inside our research lab – the six anchors are marked as stars, and all the data collection grid points are marked as crosses.

---

5 Experimental Study

We conduct both real experiments and simulation to evaluate the localization algorithm's performance. We resort to simulation for the test case scenarios where performing real experiments deemed quite extensive or almost impractical with our resources available. Simulations have been performed to examine the influence of certain parameters such as the number of anchor nodes, placements of the anchor nodes, transmission power, etc on the algorithm's performance. In Section 5.1, we first discuss our experiments conducted in real testbed, and then elaborate on the findings of the simulation in Section 5.2.

5.1 Experimental Results

First, we provide description of our experimental testbed setup and data collection procedure in Section 5.1.1. Then, we present our results and findings in Section 5.1.2.

5.1.1 Testbed setup and data collection

The experimental testbed is located inside the Internet Research and Education Laboratory (intERLab) of our university. Area of the testbed is approximately 478 m^2. The separation between two consecutive grids is 0.6m in both X and Y directions (see Fig. 4). Our indoor testbed consists of rooms separated by concrete walls and many small cubicles for the researchers and students. We conducted the experiments during office hours in order to reflect real settings which would invariably include other wireless devices carried by various personnel causing interference and noise. We have used two Asus Eee PC 901 and five DELL vostro 1310 laptops. All five DELL laptops and one Asus Eee PC serve as anchors, whereas other Asus Eee PC acts as mobile target node for our experiments. Ubuntu 10.04 LTS is installed on all the laptops. The locations of the six anchors are shown in Fig. 4, marked as stars. The crosses in the figure indicate the grid points where data are collected. We have attached seven Linksys WUSB54G ver. 4 USB WLAN adapters to the laptops where all of them create an ad-hoc network and run OLSR protocol [34].

The 266 grids of our testbed are marked with crosses in Fig. 4. ETX, hop-count metric and RSS measurements are collected in these grid points. For collecting RSS, we have used "tcpdump" [35] to extract the signal strength information at the target node for the grid points. We run tcpdump by enabling target node's NIC into *monitor mode*. It then retrieves the RSS information only from our anchors from the 802.11 packets. For ETX and hop-count, we consult the OLSR routing table at the target node periodically, and retrieve the information only for the anchors. All the nodes' (i.e., anchors and target) transmission powers have been set at 20 dBm.

5.1.2 Results and Findings

In this section, we compare the performance of our hop-count based localization algorithm with five other proximity based algorithms, namely the ETX, ETX induced hop-count, Ecolocation, Nearest Neighbour (NN) and Centroid. The RSS-based localization algorithm (i.e., Ecolocation [20]) utilizes the ordered sequence of RSS measurements in order to infer the target's relative proximity w.r.t. the anchors as explained in Section 2. In [15], ETX in place of RSS has been used for the same purpose. The ETX induced hop-count algorithm is actually another form of our hop-count based localization algorithm discussed in Section 4. The only

difference is that the weight of the link between two nodes is denoted by the ETX of the link rather than 1.0 for shortest path computation of a routing protocol like OLSR. In Fig. 1, the hop-count metric between nodes A and C of the pure hop-count and ETX induced hop-count can be found as 1 and 2, respectively. The NN algorithm returns the anchor's location from which the node receives strongest RSS as its own location estimate [36]. The Centroid method [4] defines a connectivity metric which indicates the closeness of a node to an anchor. During a certain time interval, all the anchors send a predefined number of beacons. The connectivity metric is defined as the number of beacons received by the node from a particular anchor to the number of beacons sent by it during a time interval. The final location estimate is the centroid of all the anchors for which, the connectivity metric is above a certain threshold. In our implementation, all the anchors are configured in similar way so that the number of beacons (RSS) sent by them are generally the same during a particular time interval. We just capture the received beacons at the target node, and compare with the threshold to decide whether it is connected to a particular anchor or not.

Fig. 5: Comparison of our hop-count based algorithm's performance with other proximity based localization techniques.

Fig. 5 depicts the cumulative probability graph of localization errors for all the six algorithms explained above. We calculate deviations (in meters) between actual and predicted locations by the algorithms for 183 locations among the 266 grid points where measurements are taken. Then we draw the graph of Fig. 5 where the y-axis of the graph indicates the percentage of locations predicted correctly within a specific error distance (x-axis). Some numerical values (e.g., percentiles and average) of this figure are listed in Table 1. The reasons for not using all the 266 measurements is that we did not receive enough samples at a few locations to be able to localize the target using Centroid scheme. The connectivity metrics for all the anchors were zero for those grid points. However, using 266 grid points also showed similar localization error trend as Fig. 5 for the other five algorithms excluding Centroid.

It can be seen from Fig. 5 and Table 1, that our hop-count based algorithm outperforms all the other five proximity based localization algorithms. Thereby, our experimental results verify our analysis in Section 3.2 where hop-count is shown to correlate with the distance. On the contrary, RSS is affected by reflection, diffraction and multi-path effects due to which it correlates poorly with distance. The performance of the hop-count based algorithm is improved significantly compared to RSS based algorithm as seen in Fig. 5. RSS measurements generally do not give accurate distance representation in the real world. Therefore, uncertainties could arise while using (1) as discussed in [20]. However, hop-count is more indicative of the

closeness between two nodes since it is generally the metric to be used for shortest path computation for a routing protocol. Evidently, hop-count seems to be better proximity indicator compared to RSS in our experiment.

Table 1: Percentile values and Average errors (in meters) for various proximity based localization algorithms.

Proximity Parameter	25th Percentile	Median	90th Percentile	Average
Hop-count	4.53	6.42	10.67	6.64
ETX	4.53	6.71	13.50	7.35
ETX based Hop-count	4.53	6.75	13.35	7.32
RSS	5.66	9.39	18.78	10.26
NN	4.37	8.65	22.49	10.94
Centroid	5.24	9.69	23.74	11.45

Fig. 5 also reveals that the accuracy of the hop-count based algorithm is better than both the ETX-based algorithms. ETX predicts the number of transmissions needed for sending a packet over a link including retransmissions. ETX for a link between two nearby nodes could also be high if the link quality is not good. In other words, ETX suffers from the same environmental factors (e.g., reflection, diffraction, interferences from other nodes etc.) similar to RSS discussed above. Therefore, it can be inferred that hop-count provides better proximity measure compared to ETX as well.

Furthermore, we notice that the performances of ETX based and ETX induced hop-count based localization algorithms are almost similar. This is due to the fact that this particular hop-count is actually induced from the ETX which was explained at the beginning of this section. Even though the two resultant graphs are almost similar but they are not exactly the same. The scenario in Fig. 1 could be used to explain the slight deviation between these two algorithms' performances. For example, ETX induced hop-count fails to satisfy the proximity constraint in Fig. 1 where $HC_{\mathrm{BA}} = HC_{\mathrm{BC}}$ but $d_{\mathrm{BA}} < d_{\mathrm{BC}}$. On the contrary, ETX based proximity constraint is satisfied, i.e., $\mathrm{ETX}_{\mathrm{BA}} < \mathrm{ETX}_{\mathrm{BC}} \Rightarrow d_{\mathrm{BA}} < d_{\mathrm{BC}}$. However, this does not necessarily guarantee that ETX-based algorithm always performs better than ETX induced hop-count algorithm. It performs well only when the reported link quality is inversely proportional to the distance [15] which may not always be the case because of various environmental factors discussed previously.

The separation between two testing locations of our experiments performed in Section 5 is 1.2m. The localization error of our algorithm concerning all the 266 grid points was found to be 6.58m. We have considered larger separations (e.g., 2.4m and 3.6m) as well, but the results remained almost similar. For grid resolution 2.4m, the average localization error was found to be 6.65m whereas for 3.6m, it was 6.94m. Note

that, the higher the grid separations, the less the number of locations that were considered in the localization algorithm (e.g., for 2.4m there were 148 testing locations whereas for 3.6m there were only 105).

5.2 Simulation Results

As explained previously, we perform simulations for analyzing some characteristics of our localization algorithm which are quite extensive or almost impractical to be carried out by real experiments with our resources available. We select ns-3 network simulator as our simulation platform. We consider a 40×40 square testbed spanning over an area of 1600 m^2. We create an ad-hoc network of the anchor nodes and a target node. For each experiment, the target node is placed at each of the grid positions, and the localization error is computed which is the deviation (in meters) of its estimated position from its actual position. OLSR has been chosen as the nodes' routing protocol, and 802.11b module of ns-3 has been selected as the underlying wireless technology. The HELLO and TC message interval are kept as 2 and 5 sec., respectively according to OLSR's standard setting. Simulation time for each grid point has been set to 100 seconds. Number of anchor nodes and transmission power vary according to particular experiment's needs. Simulation has been designed to analyze how localization algorithm's performance gets affected by certain parameters like the number of anchor nodes, placements of the anchor nodes, transmission power, etc.

5.2.1 Impact of the number of anchor nodes

For this experiment, the size of testbed (1600 m^2) and the transmission power (5 dBm) are kept constant whereas the number of anchor nodes are varied. The target node is stationed at each of the 1600 grids, and our localization algorithm is run using the hop-count metric from the anchor nodes obtained for each position. The experiment has been repeated for five, six, seven and eight anchors, and the resultant cumulative probability graph of localization errors is shown in Fig. 6. According to the graph, increase in localization accuracy is monotonic as the number of anchor node increases. This experiment has been performed with fewer anchors than 5 as well and the localization error was seen to increase monotonically with the decrement of anchors. This result is in-sync with the findings of other works in the localization literature which report similar phenomenon [28, 37, 38]. However, from the pattern recognition theory [39], it is well-known that the dimension of the feature vector (i.e., number of anchors in our case) cannot be arbitrarily increased to achieve better accuracy. The numerical results i.e., average localization errors and the standard deviations obtained from this experiment are listed in Table 2. As it can be seen that we achieved the best accuracy with eight anchors for our particular testbed.

Fig. 6: Localization accuracy improves with increasing number of anchor nodes.

5.2.2 Impact of the placements of anchor nodes

In this experiment, we analyze how the placements/distribution of anchor nodes inside the testbed affect localization performance. We consider three different types of distribution of anchors, namely the even, random and cluster (see Fig. 7). For even setting, the anchors are distributed evenly throughout the testbed. The random distribution indicates the anchor nodes are distributed randomly while in cluster setting, the anchors are situated very close to one another forming a cluster at some area of the testbed. For all types of distributions, we considered 8 anchors at 5 dBm transmission power.

Table 2: Average localization errors and standard deviations (in meters) of the hop-count based algorithm with different number of anchor nodes.

No. of Anchors	Average	Std. Deviation
5	8.25	5.84
6	6.87	2.75
7	6.16	3.14
8	5.63	2.76

We see from Fig. 8 that the accuracy of the localization algorithm not only depends on the number of anchors but also on their distribution inside the testbed. The even distribution is seen to perform the best while the cluster distribution's performance is the worst among the three which complies with the findings of other localization algorithms found in literature [38]. Table 3 lists the average localization errors and the standard deviations obtained for this experiment.

Fig. 7: Distribution of the 8 anchor nodes over our testbed.

We are in the view that the finding of this experiment could be particularly helpful for the scenarios where the anchor nodes are mobile (e.g., robotic motion planning [28]). Depending on the number of mobile anchor nodes, they could be instructed to move to certain areas of the testbed so that they are evenly distributed throughout the area in order to improve localization accuracy.

Fig. 8: The effects of distribution of anchor nodes on the localization accuracy.

5.2.3 Impact of varying the transmission power

The objective of this experiment is to inspect the effect of transmission power on localization accuracy. We perform this experiment with 2 dBm, 5 dBm and 10 dBm transmission powers for all the anchors and the target node. Eight anchors are distributed evenly for this experiment (Fig. 7).

Fig. 9: The effects of the nodes' transmission powers on the localization accuracy.

As can be seen from Fig. 9, the transmission powers of the nodes affect the localization performance as well. The numerical results of the experiment has been listed in Table 4. Setting the transmission power at 5 dBm provided the best localization accuracy for our particular testbed. However, for testbed with different sizes, we were required to vary the transmission power (i.e., setting at different values other than 5 dBm) to achieve optimum performance. Those results have been omitted for brevity. Therefore, a simple survey inside the deployment area might help setting the nodes' transmission powers appropriately before setting up the positioning system. Alternatively, the nodes may require the option to adjust the transmission power dynamically in order to obtain better localization accuracy.

Table 3: Average localization errors and standard deviations (in meters) of the hop-count based algorithm with different distribution of anchor nodes.

Distribution Type	Average	Std. Deviation
Even	5.63	2.76
Random	8.81	4.45
Cluster	12.62	4.82

For the experiment yielding Fig. 9, we observed that the target is reachable via 1-hop from each anchor (i.e., neighbor) irrespective of its position inside the testbed if we set the transmission powers of all the nodes at 10 dBm. Consequently, the constraint (8) failed to differentiate the relative distance separations between the target and each anchor utilizing the hop-count metric as discussed in Section 3.2.2. On the contrary, not all the anchors were reachable from the target inside the same testbed when we set the transmission powers of all the nodes at 2 dBm. Therefore, some anchors' relative distance information could not be approximated using (8) because of the unavailability of the hop-count metric towards them. This phenomenon also affected the localization accuracy that was seen to increase monotonically with the increment of anchors' hop-count information in Section 5.2.1. For 5 dBm transmission power setting inside the testbed, the anchors were observed to be reached via varying hops from the target which yielded better relative distance approximations using (8) that subsequently gave rise to finer accuracy as seen in Fig. 9.

We conclude that both anchors' density as seen in Section 5.2.1 and their transmission powers have effect on localization accuracy. Our algorithm is observed to perform well with increasing number of anchor nodes, and also by maintaining their transmission powers in such a way that all the anchors could be reached via varying hops from the target node in a particular setting.

Table 4: Average localization errors and standard deviations (in meters) of the hop-count based algorithm with varying transmission powers of the nodes.

TX Powers	Average	Std. Deviations
2 dBm	8.20	4.35
5 dBm	5.63	2.76
10 dBm	14.79	5.27

6 Conclusion and Future Works

In this article, we propose a hop-count based positioning algorithm for wireless ad-hoc networks. We analytically show that hop-count could be utilized as a proximity indication parameter and thereby, it can easily be adopted for the localization techniques designed for wireless ad-hoc networks comprising of inexpensive nodes. According to our experiments, the hop-count based algorithm outperforms a few other range-free algorithms, namely Nearest Neighbour, Centroid, Ecolocation and ETX based algorithms. Moreover, we analyze various aspects of our hop-count based localization algorithm through simulation. We observe that the localization algorithm's performance tends to improve with increasing number of anchors similar to the observation of other works in literature [28,37,38]. In addition, the results exhibit that the localization algorithm's accuracy depends on the anchor nodes' placements as well. We experimented with a few geometrical configurations, and achieved the best performance when the anchor nodes are evenly distributed over the area. We also observe the transmission power to affect localization accuracy, and for a particular density of anchor nodes, the testbed size generally dominates what value the transmission power should be set.

For this research work, we have only used an indoor testbed which spans over an uniform level i.e., without elevations. In future, we plan to perform experiments inside non-uniform settings. Additionally, we shall perform some outdoor experiments as well. Furthermore, we shall evaluate performance of the algorithm with different testbed settings, i.e, varying transmission power and number and placements of anchors, etc. We shall also investigate our hop-count based system's performance in an anchor-free scenario utilizing collaborative localization approach [40]. Additionally, the impact of mobility on the localization accuracy is certainly an interesting future work direction.
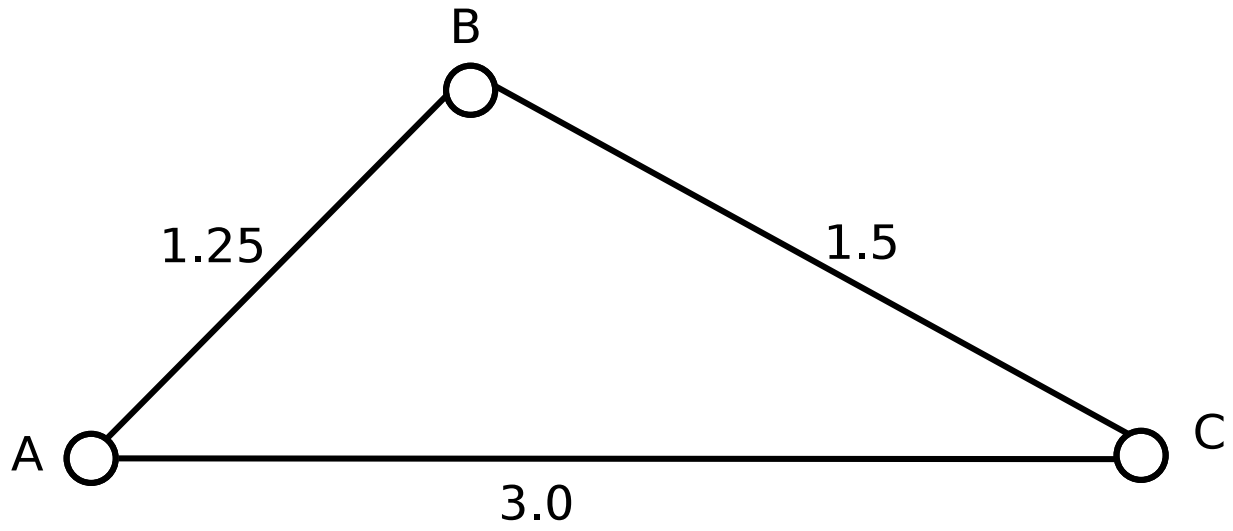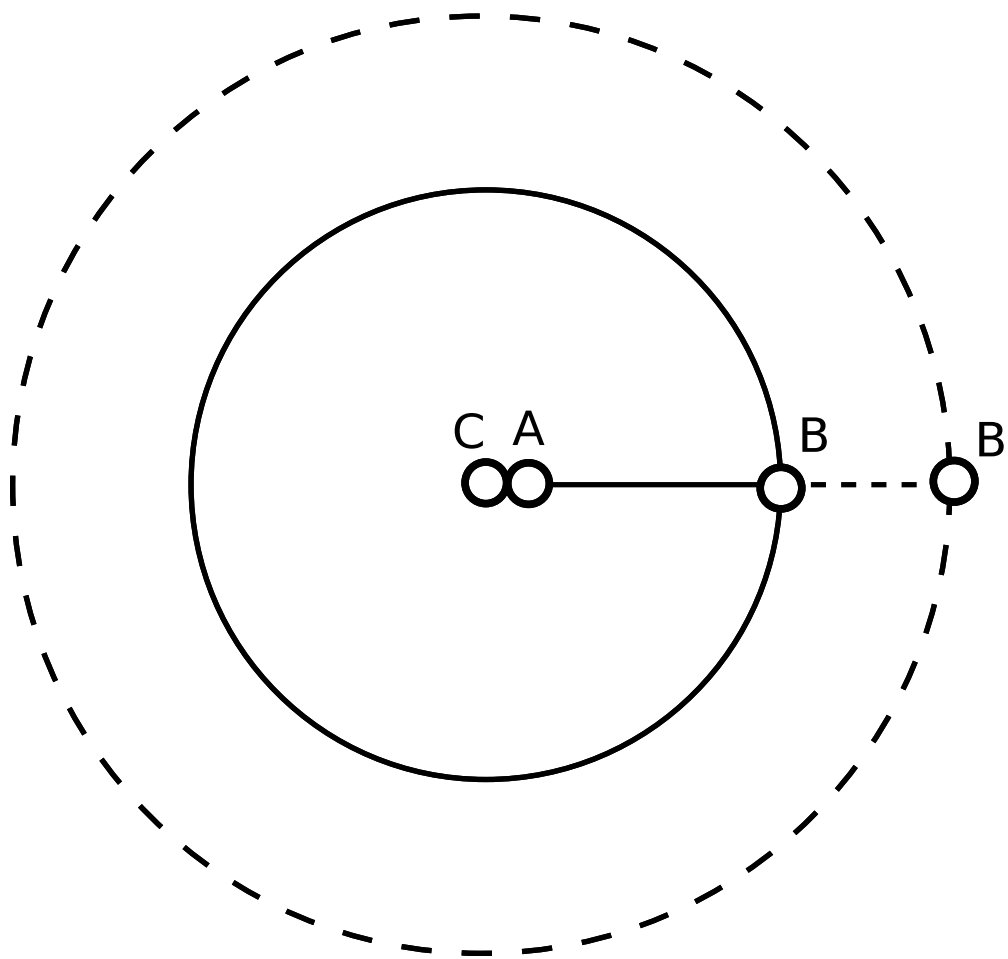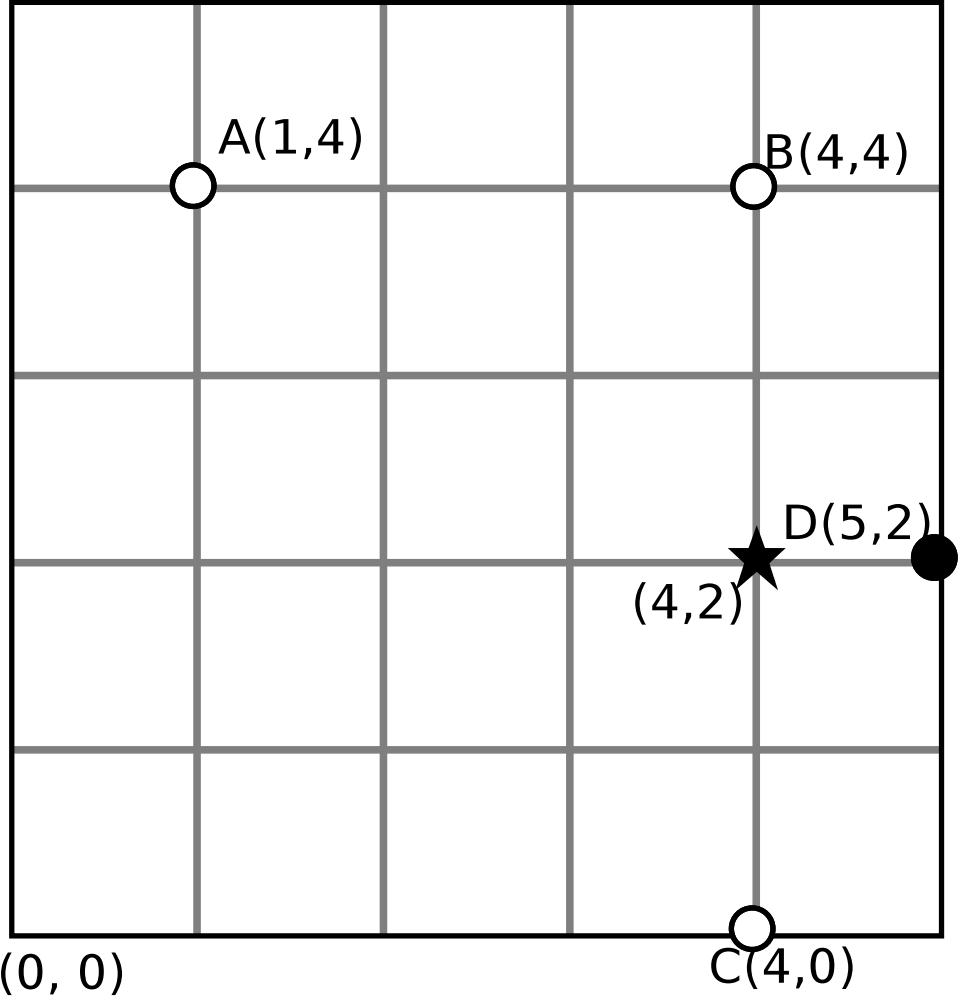
Acknowledgements

References

1. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E: **A Survey on Sensor Networks**. *IEEE Commun. Mag.* 2002, **40**(8):102–114.
2. Villafuerte FL, Schiller J, Tapia E, Ramírez M, Valdemar E: **Evaluating Parameters for Localization in Wireless Sensor Networks: A survey**. In *4th International Congress on Electronics and Biomedical Engineering, Computer Science and Informatics (CONCIBE 2008). Guadalajara, Mexico* 2008.
3. Enge P, Misra P: **Special Issue on GPS: The Global Positioning System**. *Proc. of the IEEE* 1999, **87**:3–172.
4. Bulusu N, Heidemann J, Estrin D: **GPS-less low-cost outdoor localization for very small devices**. *IEEE Personal Communications Magazine* 2000, **7**(5):28–34.
5. Wong M, Aksoy D: **QUAD: Quadrant-based relative location estimates for representative topologies in wireless sensor networks**. *Computer Networks* 2009, **53**(12):1967 – 1979.
6. Merkel S, Mostaghim S, Schmeck H: **Distributed Geometric Distance Estimation in Ad Hoc Networks**. In *Ad-hoc, Mobile, and Wireless Networks*, *Volume 7363 of* Lecture Notes in Computer Science, Springer Berlin Heidelberg 2012:28–41.
7. Rappaport TS: *Wireless Communications – Principles and Practice*. Prentice Hall 1996.
8. Savvides A, Han CC, Strivastava MB: **Dynamic fine-grained localization in Ad-Hoc networks of sensors**. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, New York, NY, USA 2001:166–179.
9. He T, Huang C, Blum BM, Stankovic JA, Abdelzaher T: **Range-free localization schemes in large scale sensor networks**. In *Proc. ACM/IEEE Mobicom'03* 2003:81–95.
10. Johnson DB: **Routing in ad hoc networks of mobile hosts**. In *Proc. of the IEEE Workshop on Mobile Computing Systems and Applications* 1994:158–163.
11. Perkins CE, Bhagwat P: **Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers**. *SIGCOMM Comput. Commun. Rev.* 1994, **24**(4):234–244.
12. Jacquet P, Mühlethaler P, Clausen T, Laouiti A, Qayyum A, Viennot L: **Optimized Link State Routing Protocol for Ad Hoc Networks**. In *Proc. of IEEE INMIC* 2001:62–68.
13. Perkins CE, Royer EM: **Ad-hoc On-Demand Distance Vector Routing**. In *Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications* 1999:90–100.
14. Nicolescu D, Nath B: **DV Based Positioning in Ad hoc Networks**. *Journal of Telecommunications Systems* 2003.
15. Hossain A, Mekbungwan P, Kanchanasut K: **An ETX Based Positioning System for Wireless Ad-Hoc Networks**. In *Journal of Telecommunications Systems*, *Volume 73 of* The Lecture Notes of ICST (LNICST) 2012:174–185.
16. Want R, Hopper A, Falcão V, Gibbons J: **The Active Badge Location System**. *ACM Trans. on Information Systems* 1992, **10**:91–102.
17. Ward A, Jones A, Hopper A: **A new location technique for the active office**. *IEEE Personal Communications* 1997, **4**(5):42–47.
18. Priyantha N, Chakraborty A, Balakrishnan H: **The Cricket Location-Support System**. In *Proc. ACM MobiCom'00*, Boston, MA 2000:32–43.
19. Boccadoro M, De Angelis G, Valigi P: **TDOA positioning in NLOS scenarios by particle filtering**. *Wireless Networks* 2012, **18**(5):579–589.
20. Yedavalli K, Krishnamachari B, Ravula S, Srinivasan B: **Ecolocation: a sequence based technique for RF localization in wireless sensor networks**. In *Proc. ISPN'05* 2005.
21. Gopakumar A, Jacob L: **Power-aware range-free wireless sensor network localization using neighbor distance distribution**. *Wireless Communications and Mobile Computing* 2011.
22. Lim H, Kung LC, Hou J, Luo H: **Zero-configuration indoor localization over IEEE 802.11 wireless infrastructure**. *Wireless Networks* 2010, **16**(2):405–420.
23. Mahtab Hossain A, Jin Y, Soh WS, Van HN: **SSD: A Robust RF Location Fingerprint Addressing Mobile Devices' Heterogeneity**. *Mobile Computing, IEEE Transactions on* 2013, **12**:65–77.
24. Nagpal R: **Organizing a global coordinate system from local information on an amorphous computer**. Tech. Rep. A.I. Memo 1666, MIT A.I. Laboratory 1999.
25. Wu H, Wang C, Tzeng NF: **Novel Self-Configurable Positioning Technique for Multi-hop Wireless Networks**. *IEEE/ACM Transactions on Networking* 2005, **13**(3):609–621.
26. Ma D, Er M, Wang B, Lim H: **Range-free wireless sensro networks localization based on hop-count quantization**. *Telecommunication Systems* 2011, :199–213.
27. Rico R, Stefan T: **Node Degree based Improved Hop Count Weighted Centroid Localization Algorithm**. In *17th GI/ITG Conference on Communication in Distributed Systems (KiVS 2011)*, *Volume 17 of* OpenAccess Series in Informatics (OASIcs), Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik 2011:194–199.
28. Sit TCH, Liu Z, Jr MHA, Seah WKG: **Multi-robot mobility enhanced hop-count based localization in ad hoc networks**. *Robotics and Autonomous Systems* 2007, **55**(3):244 – 252.
29. Lee S, Kim E, Kim C, Kim K: **Hop-count based localization using geometric constraints in wireless sensor networks**. In *Communications, 2008. APCC 2008. 14th Asia-Pacific Conference on* 2008:1 –5.
30. Wong SY, Lim JG, Rao S, Seah W: **Density-aware hop-count localization (DHL) in wireless sensor networks with variable density**. In *IEEE WCNC* 2005:1848 – 1853.
31. De Couto DSJ, Aguayo D, Bicket J, Morris R: **A high-throughput path metric for multi-hop wireless routing**. In *Proc of MobiCom'03*, New York, NY, USA 2003:134–146.
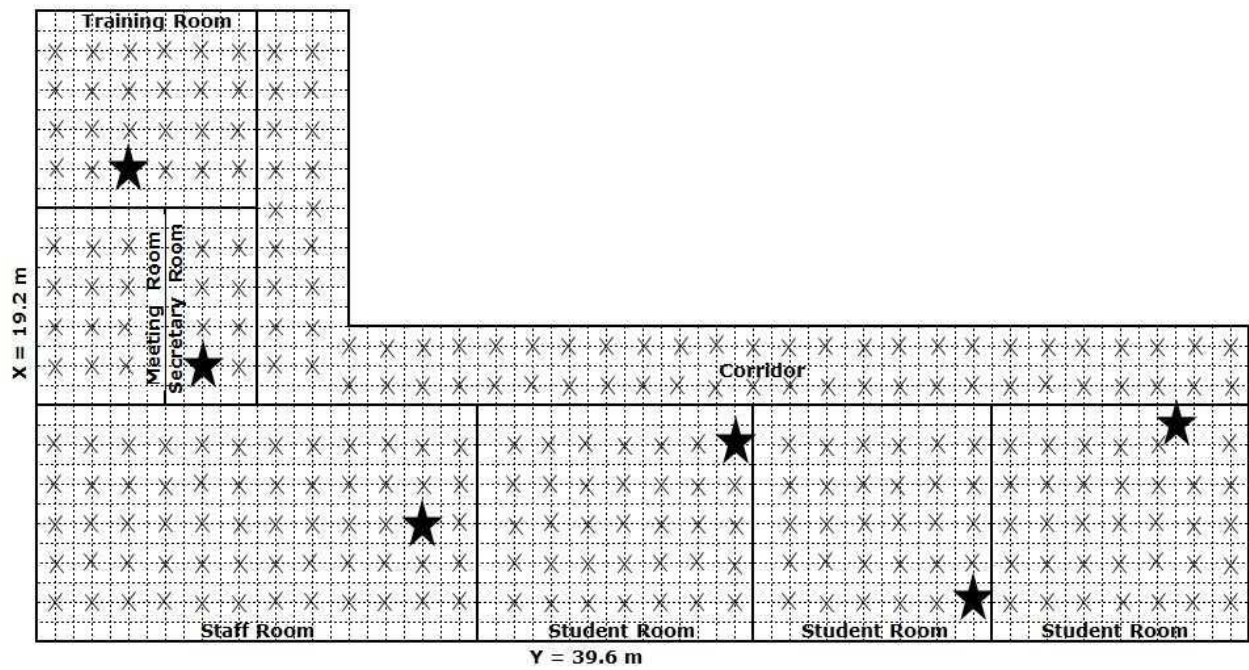
32. Kurose JF, Ross KW: *Computer Networking: A Top-Down Approach*. Addison-Wesley Publishing Company 2009.

33. Clark BN, Colbourn CJ, Johnson DS: **Unit disk graphs**. *Discrete Math.* 1991, **86**(1-3):165–177.

34. olsrd: **An adhoc wireless mesh routing daemon**[http://www.olsr.org].

35. Tcpdump: **TCPDUMP/LIBPCAP public repository**[http://www.tcpdump.org/].

36. Bahl P, Padmanabhan VN: **RADAR: An In-Building RF-Based User Location and Tracking System**. In *Proc. IEEE INFOCOM*, Tel Aviv, Israel 2000:775–784.

37. Kaemarungsi K, Krishnamurthy P: **Properties of indoor received signal strength for WLAN location fingerprinting**. In *Proc. MobiQuitous'04*, San Diego, CA 2004:14–23.

38. Hossain A, Soh WS: **Cramer-Rao Bound Analysis of Localization Using Signal Strength Difference as Location Fingerprint**. In *IEEE INFOCOM* 2010:1–9.

39. Jain AK, Duin R, Mao J: **Statistical Pattern Recognition: A Review**. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2000, **22**:4–37.

40. Priyantha NB, Balakrishnan H, Demaine E, Teller S: **Anchor-Free Distributed Localization in Sensor Networks**. In *Proc. of SenSys'03* 2003:340–341.
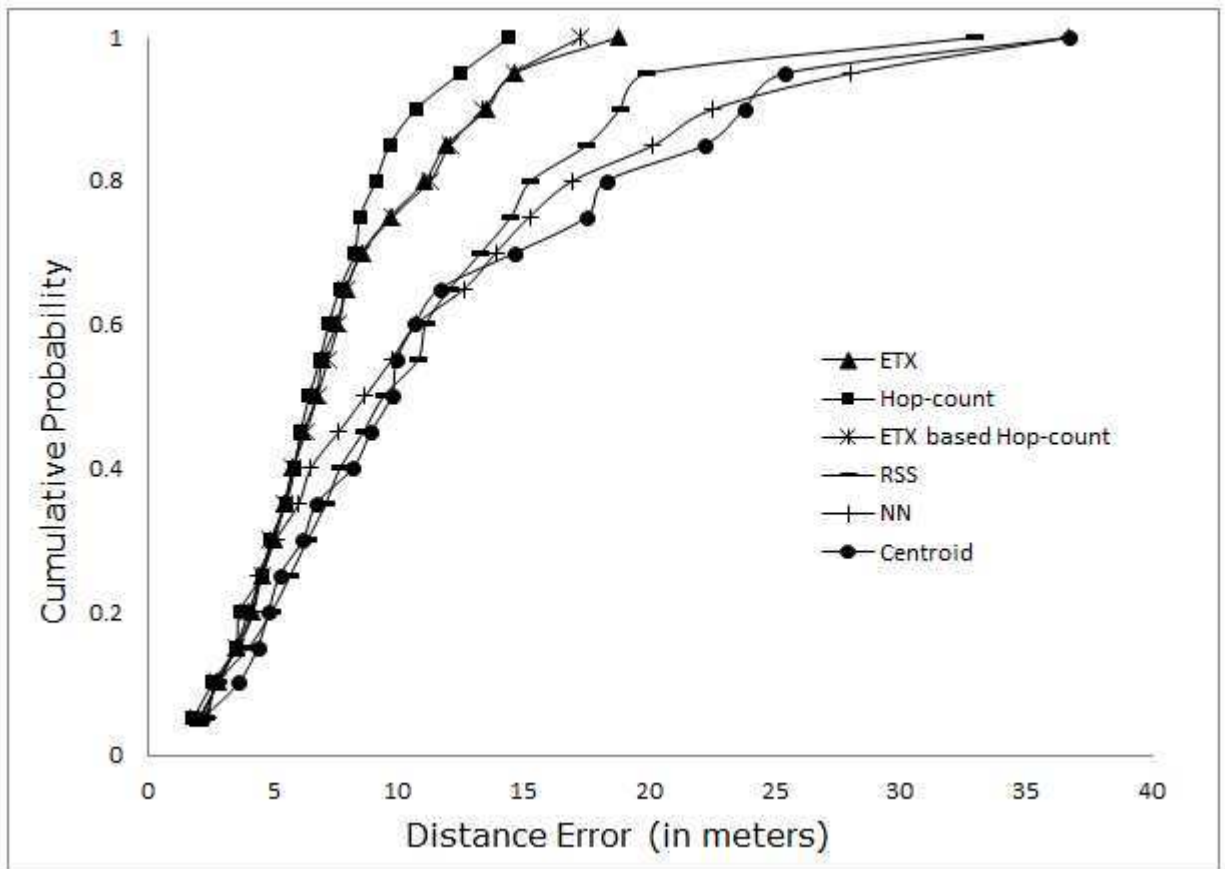
A(1,4) B(4,4) D(5,2) (4,2) (0, 0) C(4,0)

3

X = 19.2 m

Y = 39.6 m

Training Room

Meeting Room

Secretary Room
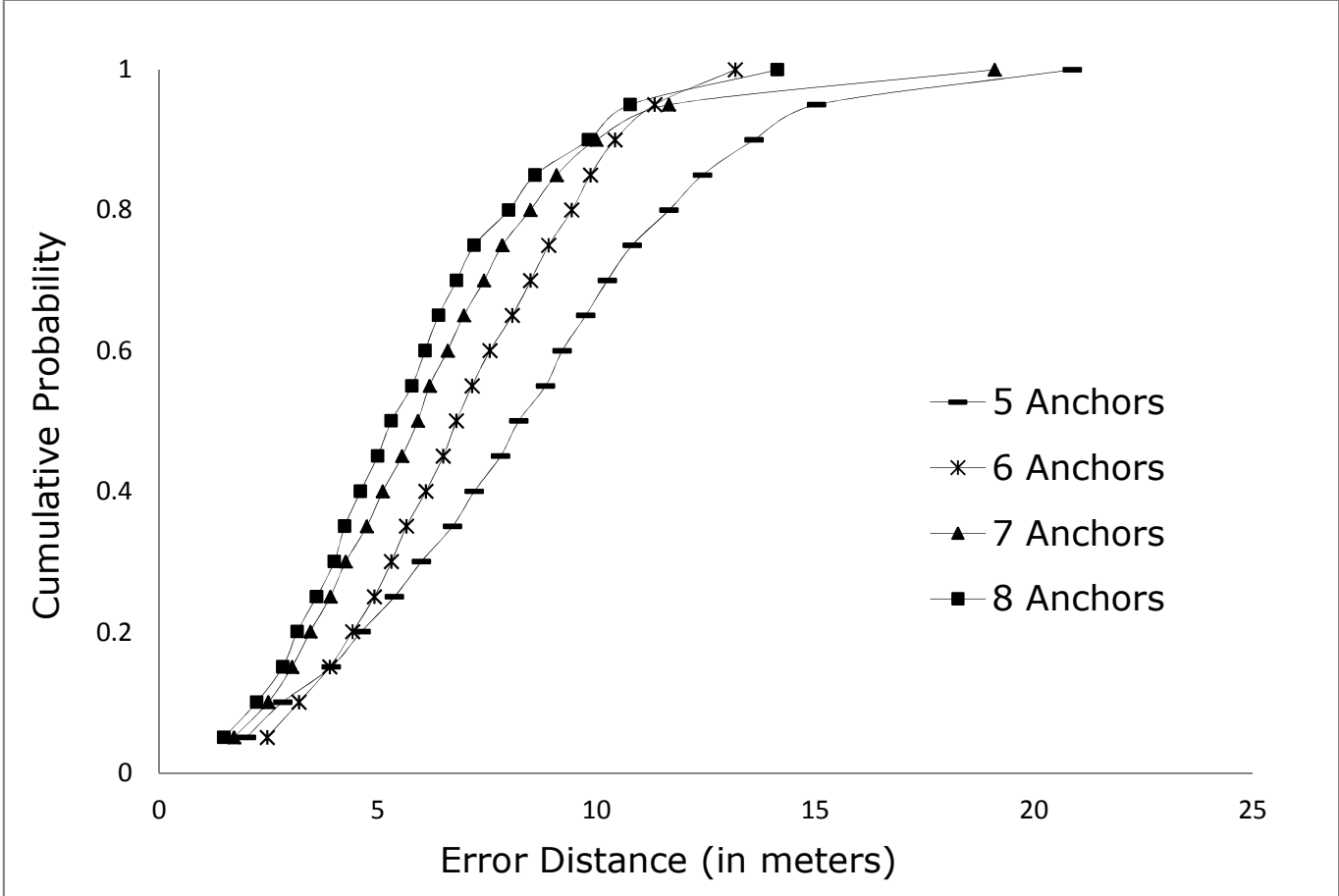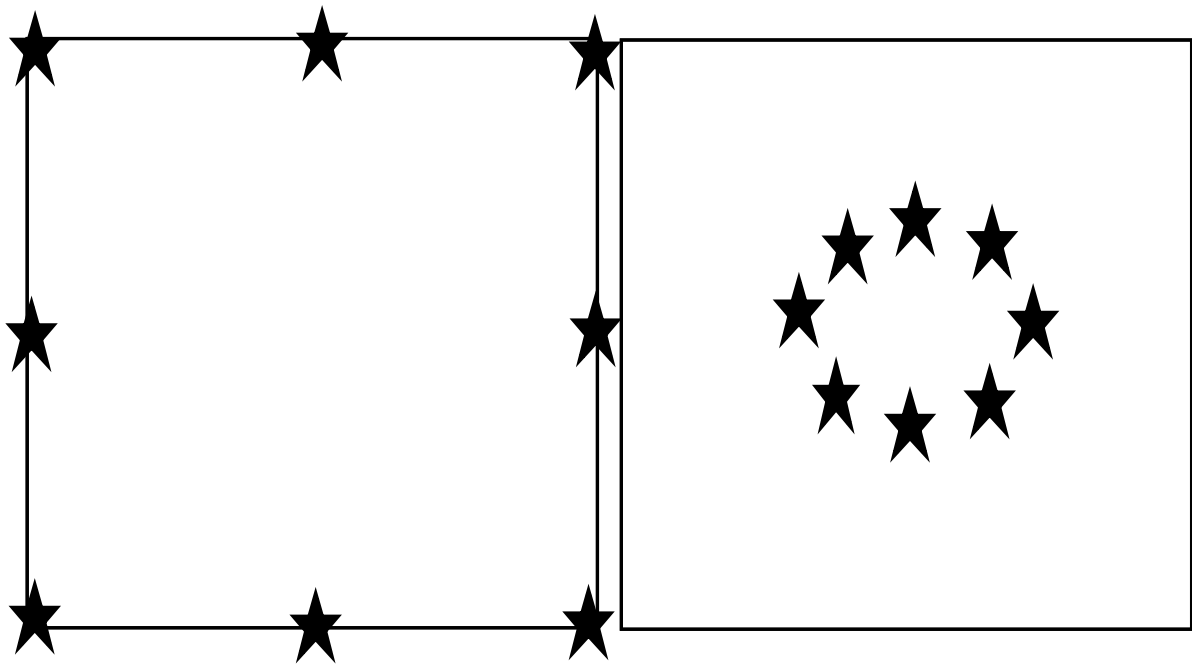
Corridor

Staff Room

Student Room

Student Room

Student Room

4

(a) Even Distribution   (b) Cluster Distribution