Wireless Pers Commun (2011) 59:375–396
DOI 10.1007/s11277-010-9923-9

Efficient Color-theory-based Dynamic Localization for Mobile Wireless Sensor Networks

Shen-Hai Shee · Tzu-Chien Chang · Kuochen Wang · Yi-Ling Hsieh

Published online: 6 February 2010

Abstract Location information is critical to mobile wireless sensor networks (WSN) applications. With the help of location information, for example, routing can be performed more efficiently. In this paper, we propose a novel localization approach, *Color-theory based Dynamic Localization* (CDL), which is based on *color theory* to exploit localization in mobile WSNs. CDL makes use of the broadcast information, such as locations and RGB values, from all anchors (a small portion of nodes with GPS receivers attached), to help the server to create a location database and assist each sensor node to compute its RGB value. Then, the RGB values of all sensor nodes are sent to the server for localization of the sensor nodes. A unique feature of our color-theory based mechanism is that it can use one color to represent the distances of a sensor node to all anchors. Since CDL is easy to implement and is a centralized approach, it is very suitable for applications that need a centralized server to collect user (sensor) data and monitor user activities, such as community health-care systems and hospital monitoring systems. Evaluation results have shown that for mobile WSNs, the location accuracy of CDL (E-CDL, an enhanced version of CDL) is 40–50% (75–80%) better than that of MCL (Hu, L., & Evans, D. (2004). Localization for mobile sensor networks. In *Proceedings of the 10th annual international conference on mobile computing and networking*, pp. 45–57). In addition, we have implemented and validated our E-CDL algorithm on the MICAz Mote Developer's Kit.

Keywords Color theory · Dynamic localization · Average hop distance · Mobile wireless sensor network

---

This work was supported by the National Science Council, under Grants NSC93-2213-E-009-124 and NSC94-2213-E-009-043.

S.-H. Shee · T.-C. Chang · K. Wang (✉) · Y.-L. Hsieh
Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan
e-mail: kwang@cs.nctu.edu.tw

1 Introduction

A wireless sensor network (WSN) consists of a collection of wireless sensor nodes operating in an ad hoc manner within a particular area. Because of the properties of low-power, small size and low-cost, WSNs can be applied to several areas, including battlefields, emergency systems, health-care systems, and smart home. For WSNs, localization is a critical issue. By exploiting location information, for instance, routing protocols can operate more efficiently. To obtain location information, a sensor node may be equipped with a GPS receiver, which is considered accurate and reliable; however, a GPS receiver is relatively expensive compared to a sensor node and is not feasible for the sensor node due to the power constraint.

Existing localization approaches can be divided into two categories roughly: *range-based* and *range-free*. In range-based approaches, certain ranging hardware is employed by sensor nodes to measure the distances between nodes. In range-free approaches, localization is performed without the need of ranging hardware; thus they are more cost-efficient and consume less energy. But as it should be, they are coarser in terms of location accuracy.

For the design of localization approaches for mobile MSNs, mobility presents a critical issue. Several localization approaches were presented [24,31,2]. In [24], four mobile beacons were used to form a square and nodes within their center of detection area calculate their corresponding locations as the centroid of the square. In [31], the issue of mobility prediction in WSNs was discussed. With the help of mobility prediction, the connectivity failure can be predicted and prevented. In [2], mobility was taken into account to assess its effect on localization. In this paper, we propose a range-free localization approach for mobile WSNs.

The paper is organized as follows. In Sect. 2, some related work of localization is reviewed. In Sect. 3, the design approach is presented. Simulation results of the proposed approach are evaluated in Sect. 4. In Sect. 5, implementation issues and experimental results are discussed. Finally, concluding remarks are given in Sect. 6.

2 Related Work

In this section, we review existing localization approaches and compare them qualitatively.

2.1 Categories of Existing Localization Approaches

Because of the diverse properties of environments and signals used, existing localization approaches are classified into *indoor* and *outdoor* localization approaches.

2.1.1 Indoor Localization Approaches

The design and deployment of indoor localization approaches are challenging tasks due to indoor characteristics. With the existence of metal and reflective materials that influence the propagation of signals and cause multipath effects and interference, the way of designing indoor localization approaches is completely different from that of outdoor localization approaches. Besides, the in-building localization method can exploit the spatial properties of the region to promote efficiency and accuracy. In [21], a predictive model for indoor tracking of mobile users was proposed. A building is segmented into several spatial components and the trajectory of the users is represented by an abstract spatial graph. In [17], mobile agents move in an indoor environment and their locations can be estimated even if the measured data from sensor nodes are not reliable. In addition to the conventional localization method that tracks users using signal strength and direction [25], the properties of the location fingerprint were employed to improve the design and deployment of the localization algorithms [1,13]. In RADAR [1], the signal data of various locations are collected and recorded as a function of the user location. The database stores the collected signal data, which is known as *location fingerprint database*, which is a critical step in the design of location algorithms. The properties of signals should be taken into consideration carefully [13]. An analytical model was proposed to analyze indoor positioning approaches [14]. However, the approaches that use signal strength for localization, such as RADAR, are single-hop-based, where anchors (or also called seeds) and sensor nodes are only one hop away. In contrast, our proposed approach is multi-hop-based which does not restrict nodes to be within the direct radio communication range to anchors. Besides, for signal strength-based approaches, a precise signal strength value is critical to the location accuracy and thus it requires additional hardware support. In contrast, our proposed approach has no such requirement.

2.1.2 Outdoor Localization Approaches

Outdoor localization approaches are featured in large-scale, hop-by-hop transmitting, and a small number of anchors, and can be classified according to the *mobility* of WSNs.

2.1.2.1 Static Localization Approaches

Ad hoc positioning system (APS) [23]: It uses one of three propagation methods to perform localization: *DV-hop*, *DV-distance* and *Euclidean propagation* [23]. Hop to hop propagation is used to obtain the distances between the nodes and landmarks (or called anchors). Once an arbitrary node has obtained distances to more than three landmarks, it is able to perform triangulation to estimate its location. The drawback of APS is that the topology of APS needs to be regular or symmetrical. Besides, the high density of landmarks is needed to guarantee certain location accuracy.

Multidimensional scaling (MDS) [27]: It determines the relative locations of nodes by using local distance information and MDS-MAP [28]. With very few seeds, the performance of MDS is still good even within anisotropic topologies and complex regions. The disadvantages of MDS are high computation complexity and poor scalability.

Localization of wireless sensor networks with a mobile beacon (LMB) [29]: This approach exploited a mobile beacon to perform localization. Each node receiving packets from a mobile beacon uses the information including the coordinate of the mobile beacon to estimate its location. Each node applies *Bayesian inference* to calculate its new position from its old estimated position. After receiving all of the packets from the mobile beacon, the final position estimate is computed. It is cost effective and simple because only a mobile beacon is needed. The drawback is that it lacks of robustness. In other words, the system would fail if the mobile beacon collapses. Moreover, it is necessary for the mobile beacon to have enough energy.

Location estimation system using wireless ad hoc networks (LES) [15]: The system uses an *accumulator host* to perform localization. Sensor nodes measure distances to their neighbors and elect a host among them as an accumulator host. The mission of the accumulator host is gathering estimated distances from other sensor nodes and estimates the locations of them. Because of the way of gathering information, it lacks of scalability. In addition, the overhead of the system is expected to soar substantially when the number of sensor nodes increases.

Robust region based localization for practical sensor networks (FGR) [38]: Sensor networks are affected by certain factors such as node density, imprecise seed position, and so on.

This system represents a sensor node position as a feasible geographic region (FGR) where the node must reside. The system first constructs a constraint set to bound sensor node positions with the consideration of above factors. And then, the FGR of a sensor node is obtained by projecting the feasible region of this constraint set to the geographic deployment plane, with a distributed and iterative implementation. A smaller FGR implies better location accuracy.

Others: In [24], it performs localization using more than one mobile beacon, which is different from LMB [29] that uses only one mobile beacon. In [24], four mobile beacons move towards the node to be localized and form a square. The location of the sensor node is the centroid of the four mobile beacons' locations. The comparison of localization algorithms for sensor networks with only one mobile beacon was presented in [32]. The mobile beacon algorithms are characterized by its simplicity of implementation and flexibility. However, the robustness is a critical problem with the concern of the failure of mobile beacons.

2.1.2.2 Mobile Localization Approaches

Mobile networks play an important role in the field of wireless networks. Most concerns on mobility were focused on the influence of mobility upon wireless sensor networks. [22] uses node position and speed to predict when partitioning will occur and which link is critical. [31] predicts the change of network topology by exploiting the regularity in mobility patterns and thus improves existing routing protocols.

GPS-less low-cost outdoor localization for very small devices (the *centroid* method) [4]: This method assumes that the ranges of coverage of reference points overlap with each other. With the assumption, each node uses the connectivity metric to get a subset of reference nodes and localizes it to the centroid of the selected reference nodes. While the method is simple and requires no coordination among sensor nodes, its applications are limited due to the assumption of overlapping coverage of the reference points. In addition, the WEP method [3] was proposed to enhance proximity based localization techniques, such as the *centroid* method [4]. In the centroid method, the position of a node is calculated as the mean value of the coordinates of the closest reference devices [3]. Although the principle of WEP came from the centroid method, each of the input parts (the coordinates of the closest reference points) is individually weighted. The main benefit of the WEP algorithm is the location accuracy gained by combining information contributed from multiple input parts [3].

Localization for mobile sensor networks (MCL) [11]: In this method, a sequential Monte Carlo method is used to estimate the posterior distribution of discrete time dynamic models. There are two stages involved, which are *prediction* and *filtering*. In the *prediction* step, a node computes its possible location by applying the mobility model to each sample. In the *filtering* step, a node filters possible locations based on new observations. The MCL needs high anchor (seed) density to improve location accuracy during the exploitation of seeds in the filtering phase. Recently, some MCL-based methods are proposed, such as MSL [26] and RMCL [8]. However, each method needs additional supports. For the MSL, not only seeds but all other nodes have to exchange their information for localization. For the RMCL, nodes need to equip distance measurement hardware.

Mobility-enhanced positioning in ad hoc networks (MAP) [19]: In MAP, mobility is concerned, and simulation results showed that location accuracy was improved with the aid of mobility. That is, mobile nodes can improve the location accuracy by bridging gaps within neighborhoods.

Table 1 Comparison of different localization approaches

Approach	Centralized or distributed	Mobility	Outdoor or indoor	Number of reference nodes (anchors)
RADAR [1]	Centralized	Low	Indoor	High
APS [23]	Distributed	Medium	Outdoor	Medium
Centroid method [4]	Distributed	No	Outdoor	High
HiRLoc [18]	Distributed	No	Outdoor	Low
MDS [27]	Centralized	No	Outdoor	Low
LMB [29]	Distributed	No	Outdoor	Low
LES [15]	Centralized	Low	Outdoor	Medium
FGR[38]	Distributed	No	Outdoor	Medium
MCL [11]	Distributed	Medium	Outdoor	High
CDL (proposed)	Centralized	Medium	Outdoor	Low

RSSI based localization algorithm using a mobile anchor node for wireless sensor networks [39]: In this approach, the mobile anchor node moves in a sensor area and broadcasts its current position periodically. A stationary sensor node computes its location using two positions of the mobile anchor node. After the first round localization is done, the un-located sensor nodes can compute their locations with the help of localized stationary sensor nodes. However, this approach need signal strength measurement capability.

2.2 Comparison of Different Localization Approaches

The above mentioned approaches are compared qualitatively as shown in Table 1, according to the following factors: centralized or distributed, mobility, outdoor or indoor, and number of reference nodes. The proposed *color-theory based dynamic localization (CDL)* is also included in the table, which will be described in Sect. 3. First, the metric of *centralized or distributed* indicates that whether there is a server in the network responsible for localization of all nodes or not. In the APS, Centroid method, HiRLoc [18], LMB and MCL approaches, nodes perform localization locally; the other approaches rely on a server to perform localization. Secondly, mobility indicates whether nodes in a network are fixed or mobile. MCL exploits mobility of nodes to improve location accuracy. Thirdly, the outdoor or indoor metric is related to the environment of networks. Only the RADAR is for indoors. Fourthly, the number of reference nodes is the number of anchors required in a network. HiRLoc [18] is a range-free scheme called high-resolution range-independent localization. In HiLoc, sensors determine their locations based on the intersection of the areas covered by beacons transmitted by multiple reference points [18]. MDS uses distance information among nodes and needs less reference nodes. LMB only involves one mobile beacon to deliver location information. Each node in CDL is able to use information of all reference nodes to do localization and thus it performs well with few reference nodes.

3 Design Approach

In this section, we propose a novel *color-theory-based dynamic localization* (CDL) algorithm. The main idea of CDL is first introduced and then the proposed CDL localization algorithm is described.

3.1 Main Idea of CDL

The motivation of CDL is to use a color to represent a position instead of using a coordinate to reduce computation and communication overhead. By using a color to represent a position, it can integrate two data: (1) each anchor's position and (2) the distance between the node and each anchor. This can reduce communication overhead. The transformation between RGB and HSV, which will be described later, involves only simple computation in comparison with that in the triangulation method [23]. Therefore, computation overhead can be reduced as well. The first data is broadcast by each anchor. To obtain the second data, there exists a classic approach, DV-hop [19], in which the distance between any two nodes is computed as average-hop-distance multiplied by hop-count. CDL is also DV-hop based, but it uses a simple way, using colors, to integrate the first data and the second data. Note that a color can be represented in two kinds of form: RGB (Red, Green, and Blue) or HSV (Hue, Saturation, and Value) [34], and one form can be easily transformed into the other form. In the RGB system, any color can be obtained by mixing together various amounts of three basic colors: red, green and blue colors [37], where each color can be expressed by assigning a value within some scale, for example, 0 to 1. In the HSV system, it uses these three values: Hue, Saturation, and Value to describe a color [7].

For the first data, each anchor generates a color (in RGB form) randomly and the RGB form is then transformed to the HSV form. For the second data, each anchor periodically broadcasts its color (in HSV) to all nodes, and the brightness, represented by V (Value), is degraded as the color propagates hop by hop. The more hops the color traverse the less brightness of the color remains. That is, the more hops a node is away from an anchor, the less brightness of the anchor's HSV color the node receives. And this HSV color is then transformed back to the RGB form. Finally, a node can derive its own color (in RGB form) by averaging the brightness-degraded colors received from each anchor (in RGB form).

We use an example WSN to illustrate the above process. Assume there are four anchors, A_1–A_4, and a target sensor node N, as shown in Fig. 1 [5]. The colors of anchors A_1–A_4 are RGB_1–RGB_4, respectively. RGB_1–RGB_4 are first transformed to the HSV form, HSV_1–HSV_4. After node N receives the four brightness-degraded colors (the V value was decreased in proportion to the hop count between node N and each anchor), HSV'_1–HSV'_4, it

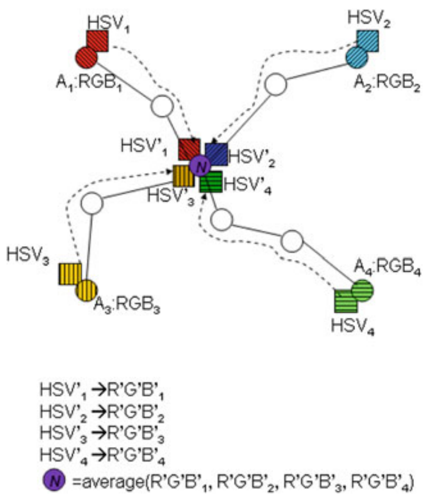


Fig. 1 The CDL color derivation for a sensor node

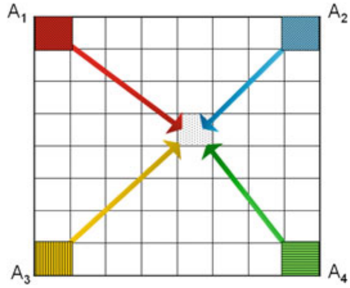


Fig. 2 CDL database establishment. The color of the dotted square can be derived by averaging the brightness-degraded colors in HSV form of the four anchors (A_1–A_4). The four colored arrows represent the brightness degraded as the colors propagate from the anchors to the dotted square

transforms HSV'_1–HSV'_4 back to $R'G'B'_1$–$R'G'B'_4$, and then node *N* can derive its own RGB color by averaging $R'G'B'_1$–$R'G'B'_4$. By looking up this RGB color in the location database, node *N*'s corresponding coordinate (location) can be obtained.

Next, we describe how to create a location database. As illustrated in Fig. 2, each anchor (A_1, A_2, A_3 or A_4) is represented by a colored square, which is maintained in the location database. Initially, only the colors of the anchors (in this example, four anchors in the corners) are known, and the colors of the other squares can be derived. The derivation of the colors in other squares is similar to that shown in Fig. 1. Note that the brightness in HSV form is degraded proportional to the Euclidian distance between each square and an anchor's square.

3.2 Operation of CDL

In CDL, an anchor is a sensor node aware of its location by equipping a GPS receiver, and each anchor generates its own color (in RGB form) randomly. The location database is established first, and then we can use the proposed CDL to localize mobile sensor nodes. Besides, since our approach involves both the RGB and HSV forms, conversion algorithms between HSV and RGB are needed. We adopted two algorithms described in [33] to transform colors between HSV and RGB.

There are four phases involved in the CDL. In the *first* phase, we obtain the average hop distance based on the DV-hop approach [19,23]. First, each anchor broadcasts a packet including its RGB value along with its location to all other anchors. The other anchors receiving the packet would record the number of hops to the anchor. After completing the broadcast, each anchor receives a set of hop count values and locations from all other anchors, and is able to calculate the distances to all other anchors. The total distances and total hop counts of each anchor to all other anchors are computed and then propagated through all nodes in the WSNs [19,23]. Upon obtaining the information from all anchors, each node computes the average hop distance [19,23].

In the *second* phase, once a node receives the RGB values and locations from the anchors, they convert the RGB values to the corresponding HSV values by an algorithm *RGBtoHSV* [33]. Based on Encyclopedia American Online (Grolier) [9], the lightness of color fades out with the increasing of propagating distances. Also, the *V* of the HSV value of an anchor is corresponding to the lightness [30]. Therefore, we decrease an anchor's *V* value in proportion to the distance from the anchor to the node, and we use this new HSV value as the anchor's color the node receives.

In the *third* phase, the sensor node derives its own color by mixing the anchors' colors it received. The anchors' brightness-degraded colors derived in the second phase (in HSV form) are transformed into the RGB form by using algorithm *HSVtoRGB* [33], and then we use the average value of these anchors' RGB colors as the sensor node's own color.

In the *fourth* phase, the RGB values of sensor nodes are transmitted to the server. The server can decide the current location of each sensor node according to its RGB value by looking up the location database. The details of the proposed CDL algorithm are described as follows, and the location database construction is described in Sect. 3.4.

3.3 The Information Delivery of Anchors

In this section, we first define some notations for CDL and then describe the distribution of the RGB values of anchors to all other sensor nodes in detail, which is the second phase of the proposed approach.

- (R_k, G_k, B_k) is the RGB value of anchor k.
- (R_{ik}, G_{ik}, B_{ik}) is the RGB value of anchor k received by node i.
- (H_{ik}, S_{ik}, V_{ik}) is the HSV value of anchor k received by node i.
- D_{avg} is the average hop distance.
- h_{ij} is the hop count between nodes i and j.
- D_{ik} represents the distance from anchor k to node i and $D_{ik} = D_{avg} \times h_{ik}$.
- *Range* represents the maximum distance that a color can propagate.

After a sensor node obtains each anchor's RGB value, the average hop distance and the hop counts to other anchors, the RGB value is first converted to a HSV value by Eq. 1.

$$(H_k, S_k, V_k) = \boldsymbol{RGBtoHSV}(R_k, G_k, B_k) \tag{1}$$

We update the HSV value of anchor k (decrease its brightness) received by node i by Eq. 2.

$$H_{ik} = H_k, \quad S_{ik} = S_k, V_{ik} = \left(1 - \frac{D_{ik}}{Range}\right) \times V_k \tag{2}$$

The corresponding RGB value of anchor k received by node i is then obtained:

$$(R_{ik}, G_{ik}, B_{ik}) = \boldsymbol{HSVtoRGB}(H_{ik}, S_{ik}, V_{ik}) \tag{3}$$

The RGB value of node i is the mean of the RGB values of all anchors:

$$(R_i, G_i, B_i) = \frac{1}{n} \times \sum_{k=1}^{n} (R_{ik}, G_{ik}, B_{ik}) \tag{4}$$

where n is the number of anchors that node i received.

3.4 The Establishment of Location Database

The establishment of a location database is performed after the server has obtained the RGB values from all anchors. The R, G, and B values of each anchor are assigned randomly from 0 to 1. And we used 10^{-2} order floating point precision for the values of R, G, and B. Let M be the number of all possible RGB values (apparently, $M = 10^6$) and A be the number of anchors. Thus, the correlation probability of the anchors' RGB values is

$$1 - \frac{M(M-1)(M-2)\cdots(M-(A-1))}{M^A}.$$

Note that there are $M(M-1)(M-2)\cdots(M-(A-1))$ possible RGB values for A anchors without correlation. As an example, for $A = 32$ (the same number of anchors used by our scheme and MCL [11]), the correlation probability of the anchors' RGB values is 0.000495882. Therefore, the correlation probability of the anchors' RGB values is very low.

The location database establishment mechanism is based on the theorem of the mixture of different colors. With the RGB values of all anchors, the RGB values of all locations can be computed by exploiting the ideas of color propagation and the mixture of these colors from the anchors. In the first place, the distance between each location i and anchor k is obtained:

$$d_{ik} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \tag{5}$$

where (x_i, y_i) is the coordinate of location i, and (x_k, y_k) is the location of anchor k. First, we obtain the HSV value of anchor k:

$$(H_k, S_k, V_k) = \boldsymbol{RGBtoHSV}(R_k, G_k, B_k), \quad k = 1, 2, \ldots, N \tag{6}$$

where N is the number of anchors.

Also, each anchor's V value is decreased in proportion to its distance to location i:

$$H_{ik} = H_k, \quad S_{ik} = S_k, V_{ik} = \left(1 - \frac{d_{ik}}{Range}\right) \times V_k \tag{7}$$

And we transform the above HSV value back to the RGB form:

$$(R_{ik}, G_{ik}, B_{ik}) = \boldsymbol{HSVtoRGB}(H_{ik}, S_{ik}, V_{ik}) \tag{8}$$

Then, we obtain the RGB value of location i:

$$(R_i, G_i, B_i) = \frac{1}{N} \times \sum_{k=1}^{N} (R_{ik}, G_{ik}, B_{ik}) \tag{9}$$

Finally, the RGB value, (R_i, G_i, B_i), along with coordinate (x_i, y_i) of each location i can be stored in the location database. After the location database is established, the location of a sensor node can be acquired by looking up the location database using the sensor node's RGB value. In summary, by our approach, a sensor node can send its RGB value to the server instead of multiple anchors data. Our signature-like matching is thus more efficient because it only needs to match one single value. That is, each sensor node integrates the colors from all anchors it receives into a single RGB value based on the color theory, and the localization system still has high matching quality for high location accuracy.

In addition, our centralized architecture is especially suitable for applications that need a centralized server to collect user (sensor) data and monitor user activities, such as community health-care systems and hospital monitoring systems. In such systems, each sensor node can send its RGB value to the server so as to obtain its position by looking up the location database. A unique feature of our color-theory based approach is that it can use one RGB value to represent the distances of a sensor node to all anchors. Each sensor node does not need to calculate its position by itself like the triangulation method. In other words, our centralized localization approach can relieve the localization computation workload of each sensor node and utilize more anchors data to increase the location accuracy, compared to distributed approaches. We will compare our proposed approach with the MCL which is a distributed approach in Sect. 4. Moreover, the CDL has also been successfully integrated to an energy-efficient routing algorithm for mobile wireless sensor networks [5].

In the following, we describe three enhancements of the CDL to further improve the location accuracy.

3.5 Three Enhancements of CDL (E-CDL)

Since the CDL is based on the DV-hop approach [19,23], the error propagation during the average hop distance derivation is possible, so a correct estimation of the average hop distance is very critical to the location accuracy. To reduce the effect of error propagation during the average hop distance derivation, we introduced two enhancements to CDL, CDL1 and CDL2, to decrease the effect of error propagation. Besides, by employing mobile anchors, we can decrease possible isolations of sensor nodes in the multihop environment [23], which is the third enhancement, CDL3. We will present these three enhancements as follows, and we call CDL with the three enhancements as E-CDL.

3.5.1 Average Hop Distance Estimation

We propose two simple and quick enhancements to improve the average hop distance estimate. When a sensor node intends to route data to some destination, it would select a nearby node that is close to the destination. According to this observation, we analyzed the behavior of sensors and discovered that the next hop located between $0.5r$ and r (r is the radio range) would be selected. We assume sensor nodes are uniformly distributed in the area. As shown in Fig. 3a, sensor node S should choose a neighbor node 1 which is located larger than half of the radio range, as the next hop. However, in Fig. 3b, if sensor node S chooses a neighbor node 1, which is located less than half of the radio range, as its next hop, the following hop (node 2) must be larger than half of the radio range; otherwise it will result in the situation of Fig. 3c, where both nodes 1 and 2 are within the radio range. To have a small hop count, the routing algorithm tends to choose a sensor node located between $0.5r$ and r as the next hop, so we assume that the candidates of the next hop are within the gray area in Fig. 3. We compute the expected value of the next hop distance as follows:

$$\mathrm{P}\{x = \text{points in gray area}\} = \frac{2\pi x}{\frac{3}{4}\pi r^2} = \frac{8x}{3r^2}, \quad 0.5r \leq x \leq r$$

$$\mathrm{E}[x] = \int_{0.5r}^{\mathrm{r}} x \cdot \mathrm{P}\{x = \text{points in gray area}\} \cdot dx = \frac{7}{9}r$$

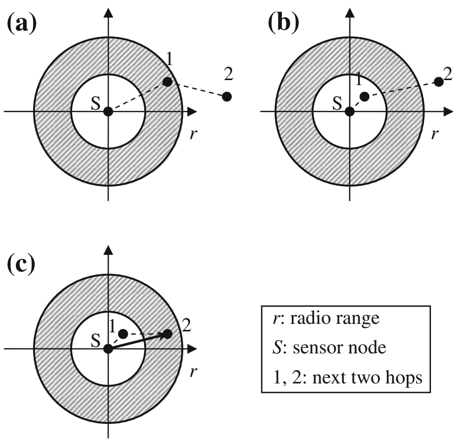
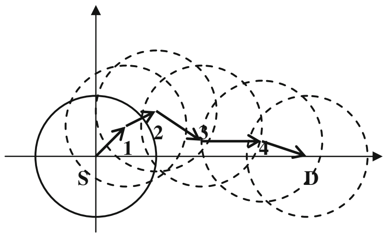


Fig. 3 (**a**) Sensor node S chooses node 1, which is located within the gray area, as the next hop; (**b**) Sensor node S chooses node 1, which is located less than half of the radio range, as the next hop; (**c**) Sensor node S chooses nodes 1 and 2 as its next two hops

Fig. 4 S → 1 → 2 → 3 → 4 → D is the shortest path from S to D. The straight line, $\overline{SD}$, is apparently shorter than the shortest path



The advantage of this first enhancement (CDL1) is that it is simple and quick for estimating the average hop distance. For the DV-Hop scheme [4], it must wait until the convergence of the topology to obtain a better average hop distance estimate. In addition, CDL1 can help reduce the location estimate error, which has been verified by simulation, as described in Sect. 4.

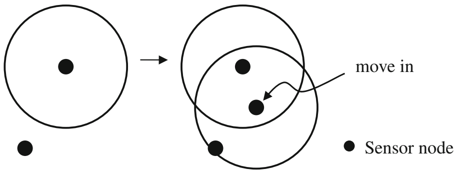
We now present our second enhancement (CDL2) to further improve the average hop distance estimate. In Fig. 4, it is obvious that the shortest path length (S → 1 → 2 → 3 → 4 → D) is larger than the Euclidean distance, $\overline{SD}$, especially when the node density is low [35]. Apparently if the node density is large enough, the shortest path length will be close to the Euclidean distance. We have verified this observation via simulation in Sect. 4. The average hop distance will be adjusted based on the ratio (R) of the Euclidean distance to the shortest path length. We can derive such a ratio via simulation offline. Note that the average hop distance can also be derived by a known formula [16] based on the node density. However, in our approach we allow the situation that the number of nodes is unknown initially. That is, the node density information is not required using our approach.

3.5.2 Mobility

In [23] it showed mobile nodes can bridge gaps between nodes where the accurate location information cannot be obtained. In addition, it also showed that mobility is helpful for the improvement of location accuracy. When a mobile node i arrives at a new location, it sends an anchor information request to its neighbors. If the neighbors have the anchors' RGB values, they transmit packets that include the RGB values of anchors and the hop counts from the anchors to the node. After receiving the packets from neighbors, mobile sensor node i compares and calculates D_{ik} to the kth anchor and chooses the smallest D_{ik}. With the RGB values and D_{ik} to all anchors, mobile sensor node i can calculate its RGB value using Eqs. 1, 2, 3 and 4. The RGB value is then transmitted to the server and the position of mobile sensor node i can be obtained by looking up the location database.

Therefore, to further improve the location accuracy, our third enhancement (CDL3) is to let not only regular sensor nodes but also anchors be mobile, in order to reduce possible isolations of sensor nodes in a multihop environment. As shown in Fig. 5, sensor node movements can help reduce possible isolations of sensor nodes in the multihop environment. We put anchors in the four corners of the square. And anchors will move a radio range (r) along the square in every time slot. By employing mobile anchors, we can reduce possible disconnections and the isolation problem of sensor nodes can be relieved. Thus, the location accuracy can be improved. Finally, we summarize the E-CDL algorithm in Fig. 6. Note that in CDL, the location database is established first and then can be used for localization; but in E-CDL, the anchors are mobile, so location database is periodically updated according to the localization interval.

Fig. 5 Sensor node movements can help reduce possible isolations of sensor nodes in a multihop environment



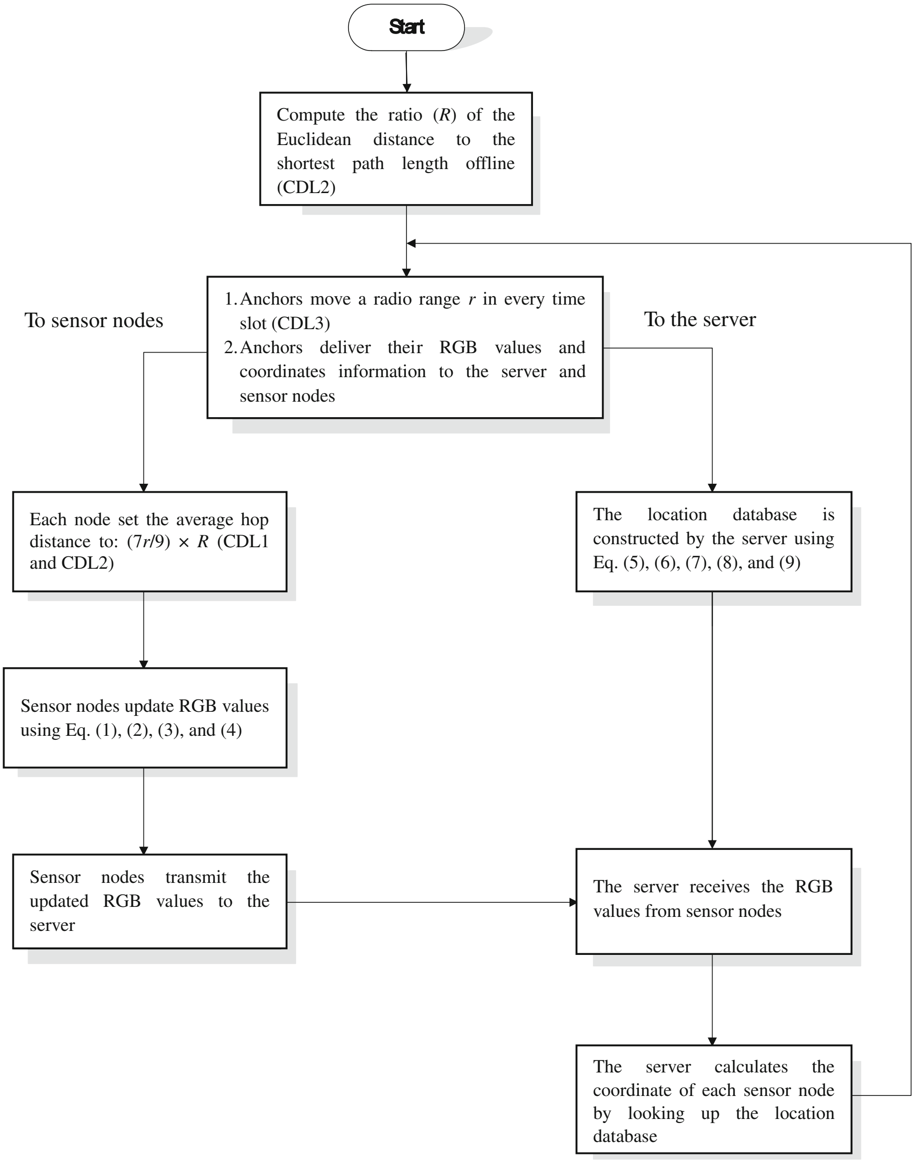


Fig. 6 The E-CDL

Table 2 Simulation parameters [11]

Parameter	Value
Area size	$500 \times 500\,\mathrm{m}^2$
Node speed	Randomly choose from $[V_{\min}, V_{\max}]$
Radio range (r)	50 m
Pause time	0
Number of samples maintained (for MCL)	50
Measurement period	$50t_u$
Time slot length (time unit)	t_u
Anchor speed (for E-CDL)	r/t_u

4 Simulation Results and Discussion

We have evaluated the proposed CDL and E-CDL algorithms by measuring their location estimate errors with various parameter settings and compared them with a classical method, MCL [11]. The simulation was conducted using *Matlab*, which is a tool for numerical computations and is powerful for its computation with matrices and vectors, and is easy for representing numerical data in graphs.

4.1 Simulation Model

Our simulation model is a mobile WSN where all sensor nodes are placed randomly in a $500\,\mathrm{m} \times 500\,\mathrm{m}$ area [11]. We first introduce the random waypoint model [12,20], which is a popular mobility model for mobile sensor networks. In this model, a node selects its destination and velocity randomly. After it reaches the destination, it pauses for a period of time (*pause time*) selected randomly. In [36], the random waypoint model was shown to fail to provide a steady state because of the decreasing of the average node speed over time. In [36], it provided a proof for the phenomenon mentioned above and came up with a methodology which guarantees the constancy of the average node speed distribution. For comparing with MCL [11], we adopted the modified random waypoint model from [36], in which nodes randomly choose their speed during each movement instead of selecting a certain speed for each destination. In addition, we assumed the radio range is a perfect circle [11]. The node speed is uniformly distributed within $[V_{\min}, V_{\max}]$. Table 2 lists simulation parameters and some terms are defined as follows [11]:

- *Seed (anchor) density* (s_d): the average number of anchors in one hop transmission range.
- t_u: time slot length between location announcements.
- *Estimate error*: expressed in terms of the node transmission range r.
- *Node density* (n_d): the average number of sensor nodes in one hop transmission range.

Figure 7 shows the location estimate error of CDL measured in terms of the node transmission range r, for different node density (n_d). The higher the node density is, the lower the location estimate error of sensor nodes is. The location estimate error can be archived below $0.3r$ when the node density is greater than 6. Figure 8 shows the location estimate error of CDL in terms of r with different anchor densities, with $V_{\max} = r$ in terms of meters per time slot for $n_d = 10$. Remind that an anchor is a sensor node that is aware of its location (with

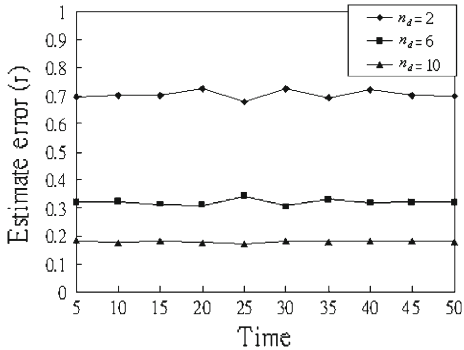


Fig. 7 Impact of node density on location estimate error of CDL

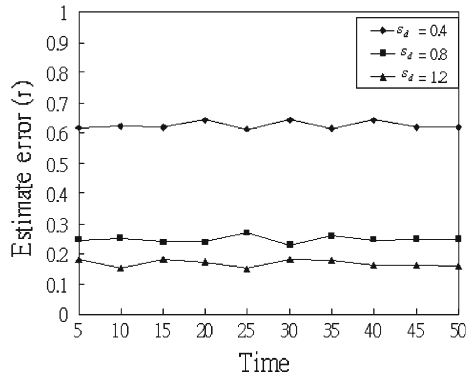


Fig. 8 Impact of anchor density on location estimate error of CDL

GPS attached). It can assist other sensor nodes to estimate their locations. That is, with the help of information given by anchors, sensor nodes can localize their positions. As expected, the higher the anchor density is, the lower the location estimate error is.

4.2 Comparison with MCL

We compare the CDL with MCL [11], which is also a range-free approach for mobile WSNs. Figure 9 compares the location estimate error between MCL and CDL over time with $s_d = 0.5$, $V_{\min} = r$ and $n_d = 10$. The MCL can exploit past information, so its accuracy improves over time. Although the CDL does not use the past information, it still performs better than MCL and is stable over time.

The node density plays an important role in localization. Since the location information is provided by neighbors, the higher the node density, the lower the location estimate error. Besides, the location estimate error caused by the uneven distribution of nodes can be reduced significantly by increasing the node density. Figure 10 shows the impact of node density on the location estimate error, with $V_{\max} = r$ and $s_d = 0.5$. The location accuracy of each scheme can be improved quickly by increasing the node density. However, in any case, CDL always performs better than MCL regardless of the node density. In the following, Fig. 11 is to show the location estimation error under different anchor densities. Figures 12, 13, 14 is to show the location estimation error improvement for each of the three enhancements against that of CDL, respectively. And Fig. 15 is to show the comparison among CDL, E-CDL (with three enhancements), and MCL by observing the location estimation error under a period of time.

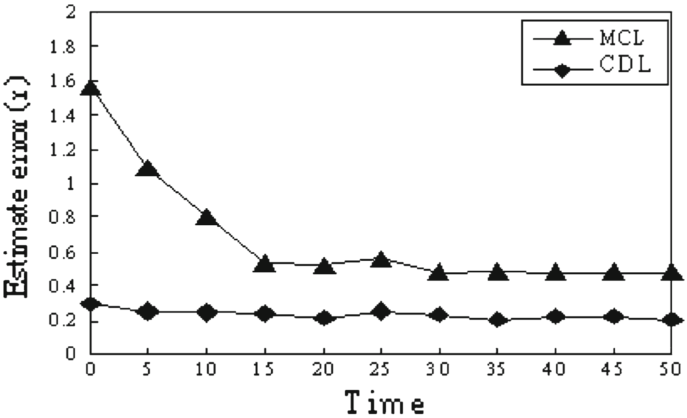


Fig. 9 Location estimate error comparison between CDL and MCL

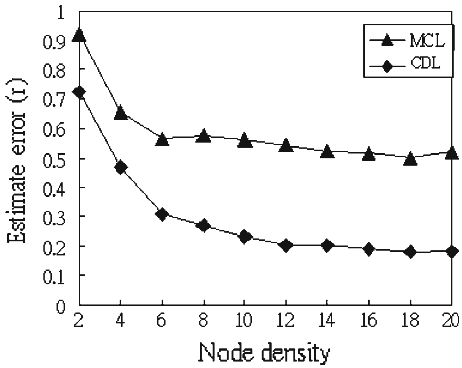


Fig. 10 Impact of node density on location estimate error between CDL and MCL

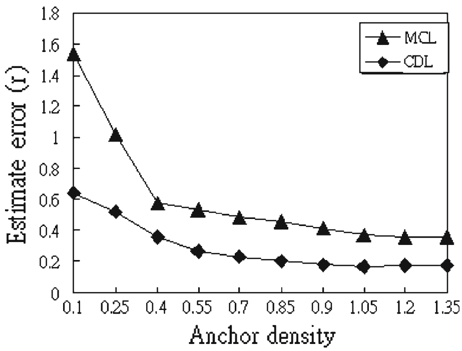


Fig. 11 Impact of anchor density on location estimate error between CDL and MCL

Figure 11 shows that the location accuracy of MCL and CDL improve with the increasing of anchor density, with $V_{max} = r$ and $n_d = 10$. MCL performs localization by collecting information from anchors or sensor nodes which are one hop away from anchors. Note that the anchor density is a critical factor for MCL. The location accuracy improves significantly as the anchor density soars. For CDL, its location accuracy does not rely on sufficient numbers of anchors, because CDL collects location information from all anchors. It can still perform well with fewer anchors, compared to MCL.

Fig. 12 Location estimate error comparison between CDL and CDL1 via simulation ($s_d = 10$, $V_{\max} = r$)

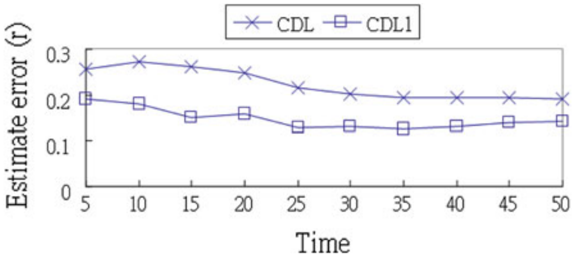


Fig. 13 Location estimate error comparison between CDL and CDL2 via simulations ($s_d = 10$, $V_{\max} = r$)

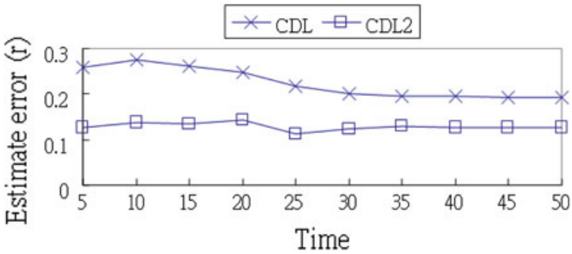
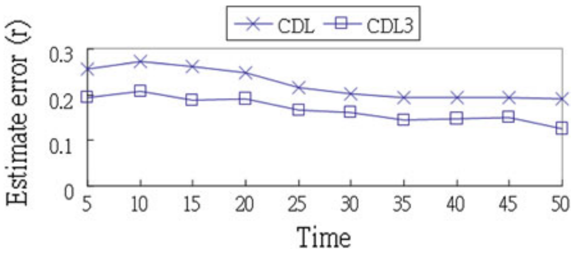


Fig. 14 Location estimate error comparison between CDL and CDL3 via simulations ($s_d = 10$, $V_{\max} = r$)



4.3 Evaluation of the Three Enhancements

By evaluating each enhancement separately, we can know how each individual enhancement contributes to the improvement of the location estimate error. Based on Fig. 3, our first enhancement (CDL1) is to set the average hop distance to $7r/9$. In Fig. 12, the location estimate error has been reduced from $0.22r$ (CDL) to $0.15r$ (CDL1) in average.

Our second enhancement (CDL2) is to compute the ratio of the Euclidean distance to the shortest path length (called *hop distance adjusting ratio*, R) offline for different source and destination pairs. Based on the anchor density ($s_d = 10$), we randomly generated different distributions of sensor nodes and compute R which is about 0.88. We used this ratio to adjust the average hop distance. In Fig. 13, the location estimate error has been reduced form $0.22r$ (CDL) to $0.13r$ (CDL2) in average.

The last enhancement (CDL3) is to employ mobile anchor nodes. By allowing four mobile anchor nodes to move around the corners, the location estimate error can be further reduced. This is demonstrated in Fig. 14, where the location estimate error has been reduced from $0.22r$ (CDL) to $0.17r$ (CDL3) in average.

Finally, we compare E-CDL, CDL, and MCL. These three range-free localization approaches were designed for mobile WSNs. Note that E-CDL is a CDL with the three enhancements. Figure 15 shows that location estimate error of E-CDL ($0.1r$) is better than

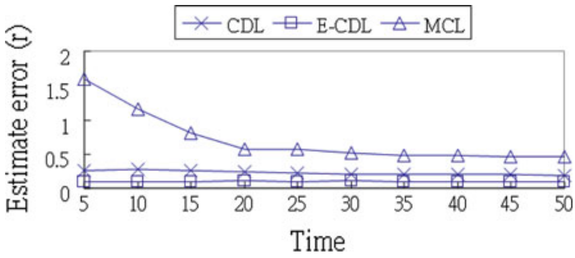


Fig. 15 Location estimate error comparison among CDL, E-CDL and MCL via simulation ($s_d = 10$, $V_{\max} = r$)

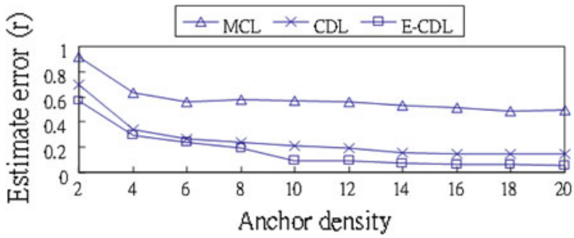


Fig. 16 Impact of anchor density on location estimate error via simulation ($V_{\max} = r$)

CDL ($0.22r$) and MCL ($0.44r$). Because MCL would exploit past information, its location estimate error improves over time [11], while E-CDL and CDL perform stable over time.

In Fig. 16, we can see that the anchor density is a significant parameter in localization, and the location accuracy increases as the anchor density increases. However, if the anchor density is low and sensor nodes are not uniformly distributed in the local area, the average hop distance derivation would be inaccurate. Hence, when the anchor density is below 8, the location estimate accuracy of E-CDL is close to CDL.

We also evaluated the impact of anchor density on the location estimate error of E-CDL. From simulation results, the location estimate error of E-CDL with four anchors in the corners is about $0.1r$. If we place more than four anchors, the location estimate error stays within $0.09r$. Therefore, we only placed four anchors in the corners for localization.

5 Implementation and Experimental Results

5.1 Experimental Setup

We have implemented and evaluated E-CDL on the MICAz Mote Development Kit [6]. The MICAz is a 2.4 GHz, Zigbee compliant Mote module used for enabling low-power, wireless sensor networks [6]. The MICAz uses the *nesC* language [10] on the TinyOS platform. We can easily construct a localization system with this kit. Due to the limitation of available sensor nodes (only 7 MICAz Motes available), we implemented the experiment with a smaller scale. The objective of the experiment is to validate the simulation results (in Sect. 4) of CDL and CDL2. The experimental results indeed support the simulation results. As to the aspect of significant results, we have evaluated our approach with classical localization approaches through simulation, as described in Sect. 4.

In our experiment, we conducted several experiments in a 20 m × 20 m area with seven motes. In addition, we assigned three motes as anchors and four as sensor nodes. Sensor nodes were randomly deployed in the area; however, there are two ways to deploy anchor

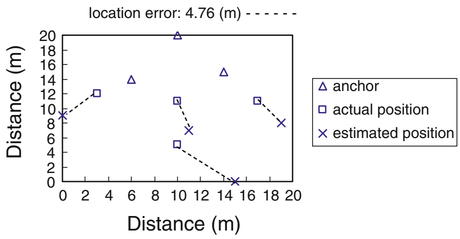


Fig. 17 Location estimate error of CDL via experiments with random anchors

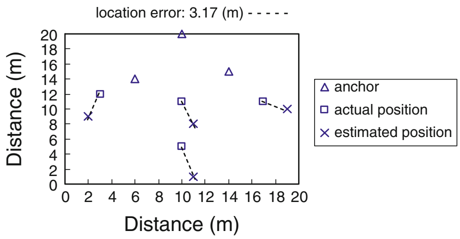


Fig. 18 Location estimate error of CDL2 via experiments with random anchors

nodes: (1) anchor nodes being deployed randomly and (2) anchor nodes being deployed in the corners. We will compare these two ways of deployment in Sect. 5.2.

5.2 Experimental Results

5.2.1 Anchor Nodes being Deployed Randomly

In this experiment, we first implemented the CDL algorithm with sensor and anchor nodes randomly deployed. In Fig. 17, the location estimate error of CDL is about 4.76 m. To enhance the location accuracy, we computed the hop distance adjusting ratio (R) offline for different distribution of source and destination pairs. The ratio (R) we used is 0.88. We used this ratio to adjust the average hop distance. In Fig. 18, the location estimate error of CDL2 is about 3.17 m which is better than that of CDL (4.76 m).

5.2.2 Anchor Nodes being Deployed in the Corners

In this experiment, we placed anchor nodes in the corners [35]. In Fig. 19, the location estimate error of CDL is about 2.38 m. Again, we used the ratio, $R = 0.88$, to adjust the average hop distance. In Fig. 20, the location estimate error of CDL2 is about 1.87 m. This demonstrated that the location estimate error of CDL has been reduced from 4.76 to 2.38 m and that of CDL2 has been reduced from 3.17 to 1.87 m by placing the anchor nodes in the corners. Therefore putting anchor nodes in the corners can improve the location accuracy.

5.3 Comparison of Simulation and Experimental Results

In Sect. 4, the location estimate error of E-CDL based on simulation results are about $0.1r$, which is apparently better than the location estimate error ($0.21r$) from experimental results. This is because the sample size for experiments was small and sensor nodes were stationary. In this situation, sensor nodes might not be uniformly distributed and might result in possible disconnections. In the future work, we will use a large sample size and mobile sensors to further validate the location accuracy of E-CDL.

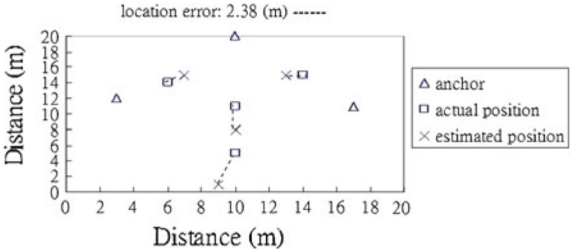


Fig. 19 Location estimate error of CDL via experiments with anchors in the corners

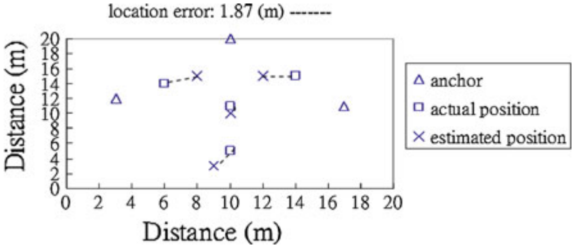


Fig. 20 Location estimate error of CDL2 via experiments with anchors in the corners

6 Conclusions

Localization is a critical issue in wireless sensor networks. With the aid of locations of sensor nodes, for example, the efficiency of routing can be improved significantly. We have proposed an efficient Color-theory based Dynamic Localization (CDL) for mobile wireless sensor networks. The basic idea of CDL is to exploit the changes of colors with distances to localize sensor nodes. The evaluation results reveal that the location accuracy of CDL is 40–50% better than that of MCL. In addition, we have incorporated three enhancements (E-CDL) to further improve the location accuracy. The basic ideas of the enhancements are more accurate estimate of the average hop distance and with the assistance of mobile anchors that are placed in the corners. The evaluation results have shown the location accuracy of E-CDL is 50–55% better than that of CDL, and 75–80% better than that of MCL. In summary, E-CDL is an efficient range-free and centralized localization scheme, and is therefore very suitable for health-care and hospital monitoring systems.

References

1. Bahl, P., & Padmanabhan, V. N. (2000). RADAR: An in-building RF-based user location and tracking system. In *Proceedings of the IEEE INFOCOM*, pp. 775–784.
2. Bergamo, P., & Mazzini, G. (2002). Localization in sensor networks with fading and mobility. In *Proceedings of the* 13th *IEEE international symposium on personal, indoor and mobile radio communications*, pp. 750–754.
3. Brída, P., Dúha, J., & Krasnovský, M. On the accuracy of weighted proximity based localization in wireless sensor networks. In B. Simak, R. Bestak, & E. Kozlowska (Eds.), IFIP international federation for information processing, Vol. 245, Personal wireless communications (pp. 423–432). Boston: Springer.
4. Bulusu, N., Heidemann, J., & Estrin, D. (2000). GPS-less low-cost outdoor localization for very small devices. In *IEEE personal communications*, Vol. 7, pp. 28–34.
5. Chang, T.-J., Wang, K., & Hsieh, Y.-L. (2008). A color-theory-based energy efficient routing algorithm for mobile wireless sensor networks. *Computer Networks*, pp. 531–541.
6. Crossbow Technology, Inc., "MICAz," [online] available: http://www.xbow.com/.

7. DevX.com, "Color Models," [online] available: http://www.devx.com/projectcool/Article/19954/0/page/7.
8. Dil, B., Dulman, S., & Havinga, P. (2006). Range-based localization in mobile sensor networks. In *Proceedings of the third European workshop on wireless sensor networks*, pp. 164–179.
9. Encyclopedia American online (grolier), "light," [online] available: http://go.grolier.com.
10. Gay, D., Levis, P., & Behren, R. (2003). The *nesC* language: A holistic approach to networked embedded systems. In *Proceedings of the programming language design and implementation*, pp. 1–11.
11. Hu, L., & Evans, D. (2004). Localization for mobile sensor networks. In *Proceedings of the* 10th *annual international conference on mobile computing and networking*, pp. 45–57.
12. Johnson, D. B., & Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. In *Mobile computing*. (pp. 153–181). Kluwer.
13. Kaemarungsi, K., & Krishnamurthy, P. (2004a). Properties of indoor received signal strength for WLAN location fingerprinting. in *Proceedings of the MOBIQUITOUS*, pp. 14–23.
14. Kaemarungsi, K., & Krishnamurthy, P. (2004b). Modeling of indoor positioning systems based on location fingerprinting. In *Proceedings of the IEEE INFOCOM*, Vol. 2, pp. 1012–1022.
15. Kitasuka, T., Nakanishi, T., & Fukuda, A. (2002). Location estimation system using wireless ad-hoc networks. In *Wireless personal multimedia communications*, Vol. 1, pp. 305–309.
16. Kleinrock, L., & Silvester, J. (1978). Optimum transmission radii for packet radio networks or why six is a magic number. In *Proceedings of the national telecommunication conference*, pp. 4.3.1–4.3.5.
17. Kwon, J.-W., Park, M.-S., & Chwa, D. (2009). Localization of the mobile agent using indirect Kalman filter in distributed sensor networks. In *Proceedings of the 3rd international conference on ubiquitous information management and communication*, pp. 574–579.
18. Lazos, L., & Poovendran, R. (2006). HiRLoc: high-resolution robust localization for wireless sensor networks. *IEEE Journal on Selected Areas in Communications, 24*(2), 233–246.
19. Lim, J. G., & Rao, S. V. (2003). Mobility-enhanced positioning in ad hoc networks. In *Proceedings of the IEEE international conference on wireless communications and networking*, pp. 1832–1837.
20. Lin, G., Noubir, G., & Rajaraman, R. (2004). Mobility models for ad hoc network simulation. In *Proceedings of the IEEE INFOCOM*, Vol. 2 pp. 1312–1321.
21. Mainetto, G. (2003). A predictive model for indoor tracking of mobile users. In *Proceedings of the* 14th *international workshop on database and expert systems applications*, pp. 921–925.
22. Milic, B., Milanovic, N., & Malek, M. (2005). Prediction of partitioning in location-aware mobile ad hoc networks. In *Proceedings of the* 38th *Hawaii international conference on system sciences*, pp. 306–306.
23. Niculescu, D., & Nath, B. (2001). Ad hoc positioning system (APS). In *IEEE GLOBECOM*, Vol. 5, pp. 2926–2931.
24. Patro, R. K. (2004). Localization in wireless sensor network with mobile beacons. In *Proceedings of the 23rd IEEE convention on electrical and electronics engineers*, pp. 22–24.
25. Priyantha, N. B., Chakraborty, A., Balakrishnan, H. (2000). The cricket location-support system. In *Proceedings of the MOBICOM 2000*, pp. 32–43.
26. Rudafshani, M., & Datta, S. (2007). Localization in wireless sensor networks. In *Proceedings of the 6th international conference on information processing in sensor networks*, pp. 51–60.
27. Shang, Y., Meng, J., & Shi, H. (2004). A new algorithm for relative localization in wireless sensor networks. In *Proceedings of the parallel and distributed processing symposium*, p. 24a.
28. Shang, Y., Ruml, W., Zhang, Y., & Fromherz, M. (2003). Localization from mere connectivity. In *Proceedings of the ACM MobiHoc*, pp. 201–212.
29. Sichitiu, M. L., & Ramadurai, V. (2004). Localization of wireless sensor networks with a mobile beacon. In *Proceedings of the IEEE international conference on mobile ad-hoc and sensor systems*, pp. 174–183.
30. Smith, A. R. (1978). Color gamut transform pairs. In *Proceedings of the* 5th *annual conference on computer graphics and interactive techniques, ACM*, pp. 12–19.
31. Su, W., Lee, S.J., & Gerla, M. (2000). Mobility prediction in wireless networks. In *Proceedings of the MILCOM 2000*, Vol. 1, pp. 491–495.
32. Suu, G. L., & Guo, W. (2004). Comparison of distributed localization algorithms for sensor network with a mobile beacon. In *Proceedings of the IEEE international conference on networking, sensing and control*, Vol. 1, pp. 536–540.
33. Vishnevsky, E. "RGB to HSV & HSV to RGB," [online] available: http://www.cs.rit.edu/~ncs/color/t_convert.html.
34. Wikipedia, "HSV color space," [online] available: http://www.answers.com/topic/hsv-color-space.
35. Wu, H., Wang, C., & Tzeng, N.-F. (2005). Novel self-configurable positioning technique for multihop wireless networks. *IEEE/ACM Transactions on Networking, 13*(3), 609–621.

36. Yoon, J., Liu, M., & Noble, B. (2003). Random waypoint considered harmful. In *Proceedings of the IEEE INFOCOM*, Vol. 2, pp. 1312–1321.
37. Wikipedia, "RGB color model," [online] available: http://en.wikipedia.org/wiki/RGB_color_model.
38. Zhen, F., Wei, L., Kanru, X., & Wenqing, C. (2007). Robust region based localization for practical sensor networks. In *Proceedings of the IEEE wireless communications and networking conference*, pp. 3670–3674.
39. Zhu, Y., Zhang, B., Yu, F., & Ning, S. (2009). A RSSI based localization algorithm using a mobile anchor node for wireless sensor networks. In *Proceedings of the 2009 international joint conference on computational sciences and optimization*, pp. 123–126.

Author Biographies

Shen-Hai Shee received the B.S. and M.S. degree in computer science from the National Chiao Tung University, Taiwan, in 2003, and 2005, respectively. He is currently an engineer in the Realtek Semiconductor Corp, which is a fabless IC design house, manufacturing communications network ICs and computer peripheral and multimedia ICs. His specialized areas include WiFi network driver development, TCP/IP protocols, SDIO/USB protocols, FPGA verification, and ad hoc network localization algorithms.

Tzu-Chien Chang received the B.S. degree in quantitative finance from the National Tsing Hua University, Taiwan, in 2004, and the M.S. degrees in computer science and engineering from the National Chiao Tung University, Taiwan, in 2006. He is currently an engineer in ASML which is the world's leading provider of lithography systems for the semiconductor industry. He is engaged in the computational lithography for integrated circuit lithography-driven design and manufacturing, including resolution enhancement technology (RET) design inspection, optical proximity correction (OPC), source mask optimization (SMO), double pattern technology (DPT), double dipole lithography (DDL), model base scattering bar, inverse lithography technology (ILT) and design for manufacturing (DFM).

Kuochen Wang received the B.S. degree in control engineering from the National Chiao Tung University, Taiwan, in 1978, and the M.S. and Ph.D. degrees in electrical engineering from the University of Arizona in 1986 and 1991, respectively. He is currently a Professor/Director in the Department of Computer Science, National Chiao Tung University. He was a visiting scholar at the Department of Electrical Engineering, University of Washington from July 2001 to February 2002. From 1980 to 1984, he was a Senior Engineer at the Directorate General of Telecommunications in Taiwan. He served in the army as a second lieutenant communication platoon leader from 1978 to 1980. His research interests include wireless (ad hoc/sensor/VANET) networks, mobile/cloud computing, and power management for portable wireless/multimedia devices.

Yi-Ling Hsieh received the B.S. and M.S. degrees in computer and information science from the National Chiao Tung University, Taiwan, in 2003 and 2005, respectively. She is currently a Ph.D. student in the Department of Computer Science, National Chiao Tung University. Her research interests include wireless ad hoc/sensor networks, mobile computing, and efficient energy management for portable wireless/multimedia devices.