An Improved DV-Hop Localization Algorithm with Reduced Node Location Error for Wireless Sensor Networks

Hongyang CHEN(a), Nonmember, Kaoru SEZAKI, Member, Ping DENG(b), Nonmember, and Hing Cheung SO(c), Member

SUMMARY In this paper, we propose a new localization algorithm and improve the DV-Hop algorithm by using a differential error correction scheme that is designed to reduce the location error accumulated over multiple hops. This scheme needs no additional hardware support and can be implemented in a distributed way. The proposed method can improve location accuracy without increasing communication traffic and computing complexity. Simulation results show the performance of the proposed algorithm is superior to that of the DV-Hop algorithm.

key words: DV-Hop, localization, wireless sensor networks

1. Introduction

Due to the recent advances on sensing techniques, the development of low-power electronics and radio devices, the miniaturization of hardware and its consequent cost reduction, wireless sensors have been put into application giving birth to the Wireless Sensor Networks (WSNs) research field. WSNs can be applied to many areas, such as military affairs, commerce, medical care, environmental monitoring, and have become a new research focus in computer and communication fields. In many applications, e.g. humidity and temperature monitoring, data collected by sensor nodes without their physical positions are invaluable. Because of the constraints in size, power, and cost of sensor nodes, the investigation of efficient localization algorithms which satisfy the basic accuracy requirement for WSNs meets new challenges [1].

Many localization algorithms for sensor networks have been proposed to provide per-node location information. Based on the type of knowledge used in localization, we divide these localization protocols into two categories: range-based and range-free. Range-based protocols use absolute point-to-point distance or angle information to calculate the location between neighboring sensor nodes. The second class of methods, range-free approach, finds the distances from the non-anchor nodes to the anchor nodes. Several ranging techniques are possible for range measurement, such as angle-of-arrival (AOA) [2], received signal strength indicator (RSSI) [3], time-of-arrival (TOA) [4] or time-difference-of-arrival (TDOA) [5]. Because of the hardware limitations of WSNs devices, solutions in range-free localization are being pursued as a cost-effective alternative to more expensive range-based approaches. In the range-free localization algorithms, Niculescu et al. [6] proposed the DV-Hop localization scheme, which is similar to the traditional routing schemes based on distance vector. He et al. proposed an approximate point-in-triangulation test (APIT) algorithm in [7]. The Amorphous algorithm [8] is similar to the DV-Hop, but it assumes knowledge about the network density in advance, and uses offline hop-distance estimations. The Centroid algorithm [9] is a simple range-free localization algorithm. The node receives signals of anchor nodes in its communication area and calculates its coordinates as the centroid of these anchor nodes. Additional hardware for localization is not required for this algorithm.

Because of the advantages on power and cost on sensor node, this paper focuses the investigation on the range-free algorithms for WSNs [10], [11].

In this paper, we present an improved DV-Hop algorithm. The proposed method can improve location accuracy without increasing the hardware cost of a sensor node. Simulation results show that the performance of this algorithm is superior to the conventional DV-Hop algorithm. Compared with DV-Hop, it is more available for WSNs.

This paper makes two major contributions to the localization problem in WSNs. First, we present a practical, fast and easy-to-use localization scheme with relatively high accuracy and low cost for WSNs. Second, the proposed algorithm improves the location accuracy of the DV-Hop algorithm.

The rest of this paper is organized as follows. Section 2 presents the derivation of the proposed improved DV-Hop algorithm. In Sect. 3, simulation results are shown and localization performances are discussed. Finally, we present our conclusions in Sect. 4.

2. Algorithm Development

Niculescu and Nath have proposed DV-Hop, which is a distributed, hop-by-hop positioning algorithm. The algorithm implementation is comprised of three steps. First, it employs a classical distance vector exchange so that all nodes in the network get distances, in hops, to the anchors. And then, it estimates an average size for one hop, which is deployed as...
a correction to the entire network. Finally, unknown nodes compute their location by trilateration [12]. The DV-Hop will only work for isotropic networks, that is, networks in which the connectivity properties are the same in all directions, so that the corrections that are deployed can be used to reasonably estimate the distances between hops. The DV-Hop might not work in anisotropic sensor networks or when a node's processing power is low, since it will be more difficult to obtain the shortest path to an anchor node using the distance vector procedure in these scenarios. Another shortcoming of DV-Hop is that it requires at least three anchor nodes so that trilateration can be performed.

2.1 DV-Hop Algorithm

In this section, we present the original DV-Hop algorithm. In the first step, each anchor node broadcasts a beacon throughout the network containing its location with a hop-count value initialized to one. Each receiving node maintains the minimum hop-count value per anchor of all beacons it receives. Beacons with higher hop-count values to a particular anchor are defined as stale information and will be ignored. The non-stale beacons are flooded outward with hop-count values incremented at every intermediate hop. Through this mechanism, all nodes in the network get the minimal hop-count to every anchor node.

In the second step, once an anchor node gets the hop-count value to other anchor nodes, it estimates an average size for one hop, which is then flooded to the entire network. After receiving hop-size, unknown nodes multiply the hop-size by the hop-count value to derive the physical distance to the anchor node. The average hop-size is estimated by anchor node $i$ using the following formula:

$$
HopSize_i = \frac{\sum_{j \neq i} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j \neq i} h_{ij}}
$$

(1)

where $(x_i, y_i)$, $(x_j, y_j)$ are the coordinates of anchor node $i$ and anchor node $j$, and is the number of hops between anchor node $i$ and anchor node $j$.

Each anchor node broadcasts its hop-size to the network using controlled flooding. Unknown nodes receive the hop-size information, and save the first one. At the same time, they forward the hop-size to their neighbor nodes. This scheme could assure that the most nodes receive the hop-size from the anchor node that has the least hops between them. In the end of this step, unknown nodes compute the distance to the anchor nodes based on the smallest received hop-size and the number of hops to the anchor nodes. Let $(x, y)$ be the unknown node D location and $(x_i, y_i)$ be the known location of the $i$-th anchor node. The distance $d_i$ between D and $i$ is given by the following formula:

\[
\begin{align*}
(x - x_1)^2 + (y - y_1)^2 &= d_1^2 \\
(x - x_2)^2 + (y - y_2)^2 &= d_2^2 \\
&\vdots \\
(x - x_i)^2 + (y - y_i)^2 &= d_i^2,
\end{align*}
\]

(2)

the coordinates of $D$ are computed by the following formula:

$$
P = (A^T A)^{-1} A^T B
$$

(3)

where

$$
A = 2 \times \begin{bmatrix}
x_1 - x_i & y_1 - y_i \\
x_2 - x_i & y_2 - y_i \\
\vdots & \vdots \\
 x_{i-1} - x_i & y_{i-1} - y_i
\end{bmatrix}
$$

$$
B = \begin{bmatrix}
d_1^2 - d_i^2 - x_1^2 + x_i^2 - y_1^2 + y_i^2 \\
d_2^2 - d_i^2 - x_2^2 + x_i^2 - y_2^2 + y_i^2 \\
\vdots \\
 d_{i-1}^2 - d_i^2 - x_{i-1}^2 + x_i^2 - y_{i-1}^2 + y_i^2
\end{bmatrix}
$$

$$
P = \begin{bmatrix}
x \\
y
\end{bmatrix}
$$

(4)

2.2 Improved DV-Hop Algorithm

In this paper, we have improved DV-Hop algorithm by focusing on its step 2 and step 3.

Step 2: After obtaining the hop-size in the first step, it is straightforward that one can also estimate the distance between two anchor nodes $i$ and $j \neq i$, denoted as $d_{est}^{ij}$, as follows:

$$
d_{est}^{ij} = HopSize_i \times h_{ij}.
$$

(5)

On the other hand, the actual distance, $d_{true}^{ij}$, between anchor node $i$ and anchor node $j$ can be determined as

$$
d_{true}^{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}
$$

(6)

where $(x_i, y_i)$ and $(x_j, y_j)$ are the coordinates of anchor nodes $i$ and $j$, respectively, and $h_{ij}$ is the number of hops between them.

Following (5) and (6), the difference between the estimated and actual distances, denoted by $e^{ij}$ [13], is expressed as:

$$
e^{ij} = d_{est}^{ij} - d_{true}^{ij}
$$

(7)

which corresponds to the estimation error.

Our first proposal is to use the above differential error $e^{ij}$ in (7) as the correction factor of the original hop-size estimation represented by $HopSize_i$ in (1). The effective average hop-size, $HopSize_{eff}^{ij}$, between anchor node $i$ and $j$ is defined as:

$$
HopSize_{eff}^{ij} = HopSize_i - \frac{e^{ij} + e^{im}}{h_{ij} + h_{jm}}
$$

(8)
where \( h_{i,j} \) and \( h_{i,m} \) are the number of hops in-between, and \( m \) is the closest anchor node to anchor node \( i \).

From the correction defined in (8), each anchor node obtains and broadcasts its effective average \( hop-size \). On the other hand, an unknown node \( k \) can compute its distance \( d_{k,y}^{eff} \) to an anchor node \( y \) based on the effective average \( hop-size \) \( \text{HopSize}_{eff}^z \), which is obtained based on its nearest anchor node \( z \) and anchor node \( y \), through the formula:

\[
d_{k,y}^{eff} = \text{HopSize}_{eff}^z \times h_{k,y}
\]

where \( h_{k,y} \) is the number of hops between unknown node \( k \) and anchor node \( y \). If anchor node \( z \) is the closest anchor node to the unknown node \( k \), it is more accurate to estimate the distance between \( k \) and \( y \) by using the \( \text{HopSize}_{eff}^z \). From this principle, a generalization to any random unknown node is possible. That is, a random unknown node can use the effective average \( hop-size \) obtained by its closest anchor node to calculate the distances to at least three anchor nodes.

Step 3: A general model for two-dimensional (2-D) position estimation of a node using \( M \) anchor nodes is described below. Let \((x, y)\) be the coordinates of the node location to be determined while \((X_i, Y_i)\) be the coordinates of the known location of the \( i \)-th anchor node, where \( i = 1, 2, 3, \ldots, M \). We denote the distance between the unknown node and anchor node \( i \) as \( d_i \). It is clear that

\[
d_i = \sqrt{(X_i - x)^2 + (Y_i - y)^2}.
\]

The range difference between anchor node \( i \) with respect to the first anchor node, \( d_{i,1} \), is simply given by

\[
d_{i,1} = d_i - d_1 = \sqrt{(X_i - x)^2 + (Y_i - y)^2} - \sqrt{(X_1 - x)^2 + (Y_1 - y)^2}.
\]

Note that a node can estimate its distance to an anchor node by the effective average \( hop-size \) and the number of hops from the anchor node. In our proposed algorithm, we follow the result as given by (8) and (9).

In DV-Hop algorithm, the estimated physical distances are used together with the anchor positions to perform a triangulation in order to get the final location coordinates. In our improved DV-Hop localization system, we do not adopt the triangulation algorithm but use the 2-D Hyperbolic location algorithm [14].

From the definitions of (10) and (11), we have the following expression:

\[
d_{i,1}^2 + 2d_{i,1}d_1 = -2X_{i,1}x - 2Y_{i,1}y + E_i - E_1
\]

where \( E_i = X_i^2 + Y_i^2, X_{i,1} = X_i - X_1, \)

\( Y_{i,1} = Y_i - Y_1. \) Then

\[
h_e = G_e Z_e
\]

where \( Z_e = [x, y]^T \) and

\[
h_e \frac{1}{2} \begin{bmatrix}
d_1^2 - d_i^2 - (E_2 - E_1) \\
d_1^2 - d_i^2 - (E_3 - E_1) \\
\vdots \\
d_1^2 - d_i^2 - (E_M - E_1)
\end{bmatrix}
\]

\[
G_e = \begin{bmatrix}
X_{2,1} & Y_{2,1} \\
X_{3,1} & Y_{3,1} \\
\vdots & \vdots \\
X_{M,1} & Y_{M,1}
\end{bmatrix}
\]

Using the Weighted Least Square (WLS) algorithm, we have

\[
Z_e = (G_e^T Q^{-1} G_e)^{-1} G_e^T Q^{-1} h_e
\]

where \( Q \) is the covariance matrix for range estimate errors. It can be obtained from the estimated errors of the ranges computed by the proposed algorithm using the effective average \( hop-size \). Then the unknown node location \((x, y)\) is expressed as \( x = Z_e(1) \) and \( y = Z_e(2) \).

3. Simulation Results and Performance Analysis

In this section, simulation results are presented and analyzed. In this paper, our performance evaluation focuses on the location accuracy of the localization algorithm. In order to do this, we simulated DV-Hop and our proposed algorithm.

We consider an experiment region of square area of \( 50 \text{ m} \times 50 \text{ m} \). Similar to [6], [9], we assume the sensor nodes have the same maximum radio range \( R \), which is used for normalization only. We deploy 300 sensor nodes randomly on the two-dimensional plane and \( R \) is set to 10 meters. The number of sensor nodes and the radio range of sensor nodes will be varied then. For easy comparison and analysis at different simulation scenarios, location errors are normalized to the radio range (i.e., 0.5 location error means a distance of half the range of the radio between the real and estimated positions). We have implemented a number of experiments to cover a wide range of algorithm configurations including varying the ratio of anchor nodes, the number of unknown nodes, and the radio range. As can be seen from Figs. 1–4, the location accuracy of DV-Hop and our proposed algorithm are affected with the variety of the ratio of anchor nodes, the number of unknown nodes and the radio range. We also observe the cumulative distribution functions of estimation error to evaluate the performance of the localization algorithms studied. The simulation results are presented as flow figures.
0.62\(R\), whereas the DV-Hop has an average error of about 0.74\(R\).

The number of unknown nodes also affects the DV-Hop algorithm. In this experiment, the number of anchor nodes is fixed to 20. We can see from Fig. 2 that the location error of these two algorithms is decreased with increasing the number of unknown nodes. This is because with the increase of the unknown nodes, the node density (average number of nodes per node radio area) in networks is increased, consequently the average number of neighbors is also increased. Thus, the network will be well connected and has a higher connectivity. This increases the probability that there exist unknown nodes located on the line between anchor node \(i\) and \(j\) in each broadcast of hop count. Then the average hop-size estimated by any pair of anchor nodes will be more accurate and thus the estimated distance between the unknown node and the anchor node using average hop-size will be closer to the true distance between the unknown node and the anchor node. So the location error of the algorithm is decreased with increasing the number of unknown nodes. Our improved DV-Hop algorithm also achieves better performance than the DV-Hop algorithm in the scenario.

Figure 3 shows the cumulative density function of location error. Herein, the number of unknown nodes is fixed to 400 and the number of anchor nodes is fixed to 15. Over 99% of the nodes have less than a 6-meter error (i.e., 60% of the radio range) in our proposed algorithm, compared to 82% of the nodes for the conventional DV-Hop algorithm.

Figure 4 shows the location error of each approach for different radio ranges. Herein, the number of unknown nodes is fixed to 300 and the number of anchor nodes is fixed to 20. The location accuracy increases as the radio range increases. This increase in location accuracy is due to the fact that the estimated error between estimated distance using average hop-size and true distance between the unknown node and the anchor node is decreased. Considering that the connectivity of sensor nodes can be controlled by specifying its radio range, an increase to the radio range leads to an increase of the network connectivity. Consequently, the number of neighboring anchor nodes per unknown node will also increase.

4. Conclusion

We have proposed a novel localization method that improves the basic DV-Hop algorithm significantly in this paper. The
location error of our proposed algorithm is decreased by using a differential error correction scheme that is designed to reduce the location error accumulated over the multiple hops. It is shown in the simulation results that the proposed algorithm can improve the location accuracy of the original DV-Hop algorithm.

References