Reducing Localization Errors in Sensor Ad Hoc Networks

Dmitri Perkins and Ramesh Tumati
The Center for Advanced Computer Studies
University of Louisiana at Lafayette
Lafayette, LA 70504-4330
{perkins,rxt2909}@cacs.louisiana.edu

Abstract
In this paper, we present a localization system comprising two simple and efficient localization algorithms for sensor networks. We refer to the proposed localization system as the Differential Ad-Hoc Positioning System (DAPS). The underlying idea on which the proposed localization system is based was motivated by Differential GPS. Specifically, the algorithms use a differential error correction scheme that is designed to reduce the cumulative distance and positioning error accumulated over the multiple hops. Using simulation, we investigate DAPS and compare the proposed algorithms with non-differential based schemes. The key contribution of this work is the illustration of how differential error corrections can be calculated and incorporated into the localization process to effectively reduce the range measurement errors, and as such, significantly improve positioning accuracy.

1. Introduction
A sensor network is typically a large ad hoc network consisting of densely distributed light, small, cheap sensor nodes which are equipped with low power transceivers and have limited networking and computational capabilities. These capabilities are combined with its physical sensing capabilities to support a wide range of applications for monitoring and controlling the physical world. Many applications of sensor networks require that sensor nodes be aware of their absolute or relative (with respect to other nodes) location. This location information can be used to accomplish both application specific tasks and networking functions efficiently. For example, a sensor node operating in a monitoring or tracking system is typically required to report that an event of interest has occurred but is also responsible for reporting the location of the event. As such, the node must be capable of automatically estimating its current position. The process in which a node estimates its position in some spatial coordinate system is referred to as localization [1]. Localization in sensor networks is required to support location aware applications, object tracking, location based routing [2], [3], coverage management [4] and collaborative signal processing. Location estimates are also used in ad hoc routing to conserve energy by load balancing and to control network utilization.

Due to their characteristics, automatic node localization in ad hoc sensor networks is a tremendously challenging problem. These characteristics include (1) ad hoc deployment, (2) low bandwidth high error rate wireless communication, and (3) energy constrained nodes. The GPS [5], [6] navigational system, which uses a constellation of 24 satellites along with ground stations to provide position estimates to GPS receivers, can be used world wide and is now a viable straightforward solution to the position estimation problem for many types of devices. However, as discussed in [1], [7], [8], GPS is not a practical solution for node localization in an ad hoc sensor network for several reasons, including line-of-sight requirements between the satellite and the node wishing to determine its position, cost, high energy consumption, and form factor.

The critical challenge in designing a localization scheme for ad hoc networks is to design a node localization scheme that is accurate, robust, scalable, and energy efficient. In this paper, we present two simple and efficient localization algorithms for ad hoc sensor networks. The key contributions of this work include (1) the design and evaluation of a differential localization system for ad hoc networks that can be used to effectively reduce the ranging errors; and (2) the design of two simple, robust, and efficient algorithms based on the differential error correction scheme.

The remainder of this paper is organized as follows. In the next Section, we present each of the proposed algorithms. The simulation model and performance results are presented in Section 4. Finally, we close with conclusions in Section 5.

2. Proposed Algorithms
In this section, we propose two simple and efficient localization algorithms for ad hoc sensor networks: (1) the HopCount Method and (2) Hop Distance Method. The
algorithms are based on the same basic idea (which will be discussed in the next section), and are designed for ad hoc sensor networks with at least four beacon nodes. Beacon nodes are sensor nodes that know their location in some specified coordinate system. Nodes that are unaware of their positions are called unknown nodes. Figure 1 shows the basic network model for which the proposed algorithms are designed to operate. The algorithms differ in how distances are estimated and how error correction information is used to reduce localization errors. Although the proposed algorithm can operate in two or three-dimensional space, we will, for simplicity and pedagogical purposes, assume a two-dimensional coordinate system in this paper. In the next section, we discuss the fundamental idea which motivated our design.

2.1 The Underlying Idea

The fundamental idea of our proposed algorithms was motivated by the Differential Global Positioning System (DGPS) [9]. Figure 2 illustrates the operation of DGPS and shows how the system effectively reduces the position errors for receivers. DGPS works by placing a high-performance GPS receiver (called the reference station) at a known location on the Earth’s surface. Since the receiver knows its exact location, it can determine the errors in the satellite signals. It does this by measuring the ranges to each satellite using the signals received and comparing these measured ranges to the actual ranges calculated from its known position. The difference between the measured and calculated range is the total error. The error data for each tracked satellite is formatted into a correction message and transmitted to GPS users. These differential corrections are then applied to the GPS calculations, thus removing most of the satellite signal error and improving positioning accuracy.

A similar approach is used in our proposed algorithms where an unknown node uses error correction factors obtained from the nearest beacon node to estimate the effective distance to at least three beacon nodes and subsequently, to reduce the errors in the position estimate.

2.2 Overview of Algorithms

Each of the proposed algorithms operate in three phases: (1) effective ranging and error calculations (2) distance estimation and (3) position estimation.

- Phase 1: Phase one consists of two steps. First, the estimated distance between each beacon node pair must be established. Each beacon node broadcasts a HELLO message (see Figure 3(a)) to the other three beacon nodes announcing its position \((x, y)\)-coordinates in the sensor ad hoc network. As the hello message is forwarded over the network, a cumulative distance or ranging metric (e.g., hop count, and signal propagation time) is maintained which will be used to determine the estimated distance \(d_{est}^{i,j}\) between the beacon nodes. An unknown node, upon receiving a HELLO message, will create a table entry storing the beacon node ID, its coordinates, and the current value of the distance metric. The proposed algorithms differ in the distance metric used and in how the estimated distance between two nodes is determined.

After receiving a hello message from beacon node \(j\), the Euclidean distance \(d_{a}^{i,j}\) between beacon nodes \(i\) and \(j\) can be determined by

\[
d_{a}^{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}; \quad i \neq j
\]  

The second step in phase 1 is to determine the error in \(d_{est}^{i,j}\), which has accumulated over each intermediate hop. We estimate this error as simply the difference between the estimated and actual (Euclidian) distance, given by

\[
e^{i,j} = d_{est}^{i,j} - d_{a}^{i,j}
\]  

In this paper, \(e^{i,j}\), called the error correction factor, is used to reduce the error in the estimated distance between an unknown node and a beacon node. The proposed schemes differ in how \(e^{i,j}\) is used.
2.3 Hop Count Method

The distance between nodes is measured in hops. The question that we address here is how to estimate the effective hop length along a path between node pairs. Initially, we assume that the hop length is equal to the node transmission range, $T_x$. However, the accuracy of the estimate is refined using an error correction factor, resulting in the effective hop length $h_{eff}$. The effective hop length will be used by unknown nodes to estimate their positions. Each beacon node $k$ is responsible for calculating the effective hop length $h_{eff}$ along the path to each beacon node $j$. The beacon nodes must then forward this information to the unknown nodes. We now present the three phases of the hop-count method.

1) **Effective Hop Size Estimation:** In the Hop Count Method, each node is assumed to have the same effective transmission range $T_x$ and hop count is used as the distance metric field shown in Figure 3(a). For the hop count algorithm, we will refer to the distance metric field as the hop-count field. The hop count field is initially set to 1 by originating beacon node before broadcasting it. Each node (both unknown and beacon nodes) in the network maintains a table, which contains an entry for each beacon node. Each entry in the table contains the position of a beacon node and the distance (in hops) to the beacon.

Each beacon node $i$ will use the number of hops $n_{i,j}$ to beacon node $j$ to estimate the distance using

$$d_{est}^{i,j} = T_x \times n_{i,j}$$  \hspace{1cm} (3)

While unknown nodes have an accurate value for the number of hops, they must wait for Phase 2 to obtain the effective length of each hop instead of the erroneous $T_x$. Using equation 2, each beacon node calculates the error correction factor $e^{i,j}$ for each beacon node pair (i,j). Each beacon node $i$ can now compute the average effective hop length of each path to all other beacon nodes in the network using

$$h_{eff}^{i,j} = T_x - \frac{e^{i,j}}{n_{i,j}}$$  \hspace{1cm} (4)

Figure 4(a) illustrates how beacon node 5 uses the information obtained from other beacon nodes to estimate the hop sizes to be used by the nearby unknown nodes.

2) **Effective Distance Estimation:** During phase 2 of the Hop Count Method, each beacon broadcasts a DISTANCE-INFO message (see Figure 3(b)). An unknown node will use the hop count field shown in Figure 3(a) for the Hop Count method or error correction factors (for the Hop Distance method) along a path to the beacon node and the distance (in hops) to the beacon. For example, in Figure 1 unknown node $u_{k1}$ will only use the error correction factors received from beacon node $B_1$ to estimate the distance to other beacon nodes. The idea here is that the network perspective of a local beacon node is more reliable than that of a distance beacon node. In the case where an unknown node is equi-distance from two beacon nodes, the error correction factors between node pairs are averaged.

3) **Position Estimation:** In phase 3, an unknown node now has the location (obtained in Phase 1) of at least three beacon nodes and the effective distance (obtained in Phase 2) to the beacon nodes. As such, each unknown can now use trilateration to calculate its current position.

As stated above, proposed algorithms differ in how the effective distance $d_{eff}^{i,k}$ between a beacon node $i$ and unknown $k$ is determined. In the remainder of this section, we present a detailed description of each algorithm, including possible advantages and disadvantages of each approach.

### 2.3.1 Hop Count Method

In the Hop Count Method, the distance between nodes is measured in hops. The question that we address here is how to estimate the effective hop length along a path between node pairs. Initially, we assume that the hop length is equal to the node transmission range, $T_x$. However, the accuracy of the estimate is refined using an error correction factor, resulting in the effective hop length $h_{eff}$. The effective hop length will be used by unknown nodes to estimate their positions. Each beacon node $i$ is responsible for calculating the effective hop length $h_{eff}^{i,j}$ along the path to each beacon node $j$. The beacon nodes must then forward this information to the unknown nodes. We now present the three phases of the hop-count method.

#### Phase 2: Phase two begins with each beacon node $i$ broadcasting a DISTANCE-INFO message (see Figure 3(b)), which contains its node id and effective distance metrics (for the Hop Count method) or error correction factors (for the Hop Distance method) that will be used by nearby unknown nodes to estimate their positions. When an unknown node $k$ receives a DISTANCE-INFO message from a beacon node, it uses this information to estimate its effective distance $d_{eff}^{k,i}$ to each beacon node $i$. An unknown node will only use DISTANCE-INFO messages received from its nearest beacon. For example, in Figure 1 unknown node $u_{k1}$ will only use the error correction factors received from beacon node $B_1$ to estimate the distance to other beacon nodes. The idea here is that the network perspective of a local beacon node is more reliable than that of a distance beacon node. In the case where an unknown node is equi-distance from two beacon nodes, the error correction factors between node pairs are averaged.

#### Phase 3: In phase three, an unknown node now has the location (obtained in Phase 1) of at least three beacon nodes and the effective distance (obtained in Phase 2) to the beacon nodes. As such, each unknown can now use trilateration to calculate its current position.

As stated above, proposed algorithms differ in how the effective distance $d_{eff}^{i,k}$ between a beacon node $i$ and unknown $k$ is determined. In the remainder of this section, we present a detailed description of each algorithm, including possible advantages and disadvantages of each approach.
4) Advantages and Disadvantage: In the proposed scheme, an unknown node is not required to be within range of three beacon nodes and, as such, the transmission range of the beacon nodes need not cover the entire network. The proposed algorithms take advantage of the multi-hop routing and forward capabilities which typically existing in ad hoc networks [10], [11]. This is an important advantage with regards to efficient spectrum and bandwidth utilization and power consumption. The proposed localization algorithm is simple, robust, and scalable. Each node is responsible for finding its own position, computation is minimal and distributed over all nodes in the network, resulting in increased energy efficiency due to load balancing.

The hop-by-hop technique can be a disadvantage due to the accumulation of errors. In our approach, these errors are greatly reduced by the use of the differential error correction scheme. The assumption that all nodes have the same transmission range may also be a limiting factor in some network scenarios. However, it is also likely that in many network scenarios nodes will have the same or similar transmission ranges. For example, it is typically the case that an organization will make a decision concerning which wireless technology and data rates (e.g., 802.11a, 802.11b, etc.) are most appropriate for its needs and all devices will support the same technology.

2.4 Hop Distance Method

1) Estimating Error Corrections: In Phase 1, the Hop Distance Method also requires the beacon nodes to broadcast their position information. Each node on obtaining the position information from a beacon node creates an entry in its table for that beacon node which contains the beacon position and the cumulative hop distance to that beacon node. The distance between the neighboring nodes is estimated using time of flight of the signal between the nodes and the propagation speed of the radio signal. In this approach beacon nodes are responsible for computing the corrections (or errors) in distances to the remaining beacon nodes. Using the cumulative hop distances and the actual distances to the remaining beacon nodes, the error correction factor between beacon i and j is computed as follows:

$$e_{i,j} = d_{est}^{i,j} - d_{eff}^{i,j}$$  \(6\)

Figure 5(a) shows how the beacon node 5 computes the corrections for the nearby nodes. In a similar manner all the beacon nodes in the network compute the corrections (or errors) in distances to the remaining beacon nodes. Beacon nodes periodically compute corrections and broadcast the errors to the nearby nodes.

2) Estimating Distance: In phase 2, the beacon nodes broadcasts these error correction factors to the nearby unknown nodes, so that they can remove these errors from the corresponding estimated distances (obtained in Phase 1) to beacon nodes. An unknown node k, with nearby beacon node i estimates the effective distances $d_{eff}^{k,j}$ to another beacon j using the estimated distances $d_{est}^{k,j}$ and the error corrections $e_{i,j}$ (from beacon node i) as follows:

$$d_{eff}^{k,j} = d_{est}^{k,j} - e_{i,j}$$  \(7\)

Figure 5(b) shows how an unknown node k estimates its distances to the beacon nodes using correction factors from nearby beacon node 5.

3) Position Estimation: During phase 3, the unknown node estimates its position using beacon node coordinates (obtained in Phase 1) and the effective distances (obtained in Phase 2) using trilateration.

4) Advantages and Disadvantages: This algorithm is simple, distributed and scalable. Results show that the algorithm is more robust than the previous method. This algorithm has less response time and enables the nodes to estimate their positions with minimum computational overhead. A potentially strong disadvantage of this approach is that accurate timing/synchronization mechanisms may be required to use time-of-flight techniques.
3. Simulation Models, Assumptions, and Performance Metric

In this section, we provide an overview of the network model and simulation environment used in the evaluation of the proposed localization schemes. Our initial simulation results consider a grid topology. Figure 3 provides an example of a grid network comprising beacon nodes (the encircled nodes) and unknown nodes. The grid topology has a network area of $(1500 \times 1500)$ and a grid unit of 200 meters is considered. While several beacon node placements have been considered, the results presented in this work are based on topology where three beacons are placed at the corners of the network and one is placed at the center of the network. The beacon nodes are responsible for computing the error corrections (for the Hop Distance method) or effective hop sizes (for the Hop Count method) which will be broadcast to nearby unknown nodes. The network model used in this study is based on several assumptions described below:

- The network must contain at least four beacons for positioning of nodes in 2-dimensional space or at least five beacons for positioning of nodes in 3-dimensional space.
- The wireless links between the nodes are symmetrical.
- All nodes have identical transmission range.

We have implemented the proposed algorithms using the GloMoSim simulation library [12] for mobile ad hoc networks and the PARSEC simulation language [13]. The free space propagation model is used to determine if a node is reachable. The free space model predicts received signal strength when the transmitter and receiver have a clear, unobstructed line-of-sight path between them. Received power decays as a function of the transmitter-receiver separation distance. All nodes in the network use the IEEE 802.11 medium access control protocol for channel access. The optional RTS/CTS handshake is enabled and is used for all packet sizes. All nodes (i.e., reference and unknown nodes) have the same transmission range with a radius $R$ meters. We use Ad Hoc On-demand Distance Vector (AODV) [11] to do multi-hop routing and communication.

In this paper, our performance evaluation focuses on the position estimation accuracy of the proposed schemes. The position error is computed using

$$
\text{error} = \sqrt{(x_{est} - x_a)^2 + (y_{est} - y_a)^2}
$$

The results presented in the Section 4 show the positioning error relative to a node’s transmission radius, $R$ meters. Understanding the positioning accuracy with respect to a node’s transmission radius is important in many scenarios. For example, in a wireless broadcast environment, a location or direction-based routing protocol may only require that the estimated position lie within the target node’s transmission range. That is, an exact position estimate is not required, but instead specific broadcast or transmission area.

4. Simulation Results

In this section, we present some initial simulation results highlighting the potential performance gains of our proposed differential-based localization schemes over a non-differential-based scheme. We also investigate the impact of the transmission range on the accuracy of
4.1 Differential vs. Non-Differential Algorithms

We first investigate whether differential localization results in significantly better accuracy than non-differential-based techniques. The non-differential based techniques are implemented using the hop-count and hop-distance approach but do not include the differential components of the algorithms.

Figures 7 and 8 are histograms showing the percentage of nodes in different positioning error ranges. The positioning error is given as a percentage of the transmission radius, $R$. For example, a node with a 100m transmission radius and 10% error implies that the node estimated position lies within 10 meters of the actual position. Alternately, a node with greater than 100% error implies that the estimated position of node lies more than $R$ meters from the actual node location. We see from Figures 7 and 8 that both differential-based methods result in approximately 10% more estimates in the 0 - 10% error range. Careful examination of Figure 7 shows that the hop-count method results in approximately 70% of the position estimates being within $\sqrt{2}R$ meters of the actual node position. In contrast, the non-differential-based hop-count method results in 50% of the position estimates being greater $R$ meters (i.e., outside the node transmission radius) from of the actual node position.

4.2 The Impact of Node Transmission Radius

We now focus on evaluating the impact the node transmission range has on the performance accuracy of the proposed schemes. Many simulations were performed using the grid topology and different values of $R$. Here we show a set of representative results for both the hop-count and hop-distance schemes. Figures 9 and 10 are histograms showing the accuracy (as a percentage of the radio range) of each approach for three different values of $R$. The graphs clearly indicate that the radio-range impacts the accuracy of the proposed schemes. For hop-count method as shown in Figure 9, the accuracy of the position estimation decreases with increase in transmission range of the node. Specifically, for the hop-count method using a transmission range 200 meters, approximately 56% of position estimates lie within .2$R$ meters of the actual node location and approximately 70% of the position estimates lie within .3$R$ of the actual node location. However, as the node transmission range is increased to 300m and 400m, the percentage of nodes with position errors within .2$R$ decreased to 37% and 29% respectively. The percentage of nodes with position errors less than .3$R$ also decreased to 61% and 31% as the transmission range is increased to 300m and 400m respectively. The decrease in position accuracy of the hop count method is due to the over estimation of the hop length. In contrast to the Hop-count method, the position estimation accuracy of the hop-distance method increases as the transmission range is increased which can be observed from Figure 10. This increase in accuracy is due to the fact that an increase in transmission range results in a range estimate closer to the euclidean distance between two nodes.

5. Conclusions

In this paper we have proposed two simple and efficient localization algorithms for sensor networks: (1) hop count method and (2) hop distance method. These algorithms are based on the idea used in differential GPS to reduce the errors in the distance estimates. In our algorithms, beacon nodes function in a similar way as a reference station in DGPS and are responsible for computing corrections for the nearby nodes to improve the accuracy of the position estimates. The proposed algorithms are distributed, have low computation overhead and low response time. As the nodes compute their positions locally, our algorithms are
expected to be highly scalable. The simulation results also showed that 75% of the nodes are having their position errors less than 50% of their radio range. A key contribution of this work is the illustration of how error corrections from the beacon nodes can be used to reduce the range measurement errors, significantly improving position estimates. Our future work includes investigating the impact of beacon node placement and node density on the accuracy of the algorithms.

References