Abstract — Sensor localization has become an essential requirement for realistic applications over wireless sensor networks. The stringent constraint on hardware cost, however, makes localization in wireless sensor networks very challenging. In a hostile environment such as battlefield or forest fire monitoring system, sensors are short-lived since they may be destroyed. Therefore, to lower system cost, sensors within the hostile environment must be extremely simple and cheap. Also, it is financially undesirable to use powerful anchor nodes within the hostile area for sensor localization. We present a very low cost range-free sensor localization scheme that does not require any powerful anchor nodes within the deployment area. Our algorithm is based on random deployment which is the cheapest method for deploying a large number of sensors. Analysis and simulation evaluation are provided.

I. INTRODUCTION

Extremely small sensors have been used broadly for various applications such as habitant monitoring, battlefield surveillance, and forest fire monitoring. Such tiny sensors rely on short-range radio communication to form Wireless Sensor Networks (WSNs) [1]. The major motivation to use tiny sensors is to save energy consumption and reduce system cost, as recent tiny sensors can work without recharging for several months. To obtain long life, such tiny sensors are designed with very limited radio transmission range and computing capability, which in turn pose great challenges for some essential service supports such as message transmission, sensor synchronization, and sensor localization.

Generally speaking, measurement data must include temporal and spatial values. For instance, in building monitoring systems, a temperature reading must be associated with time (when this temperature is measured) and location (where it comes from), since otherwise the measurement data will be meaningless. As such, location information plays an essential role in understanding the application context within a WSN.

Many localization algorithms for WSNs have been proposed to provide per-node location information [2], [5]-[12]. They can be divided into two categories: range-based methods and range-free methods [5]. The former ones depend on range information (absolute point-to-point distance information estimated by measuring received signal strength) to obtain nodes’ locations, while the latter ones do not require any range information at all.

Range-based localization depends on the assumption that the absolute distance between a sender and a receiver can be estimated by received signal strength or by the time-of-flight of communication signal from the sender to the receiver. The accuracy of such estimation, however, is subject to the transmission medium and surrounding environment and usually relies on complex hardware [4], [11]. As the miniaturization of sensor nodes has become an inevitable trend, expensive hardware cost on supporting range-based localization may finally render this method obsolete for wireless sensor networks. In contrast, range-free localization never tries to estimate the absolute point-to-point distance based on received signal strength. As such, the design of hardware can be greatly simplified, making this method very appealing for WSNs. In both methods, anchors—nodes that know their own locations—are utilized. Anchor nodes are usually installed with GPS (Global Positioning System) and powerful radio transceiver and thus are much more expensive compared to normal sensor nodes.

In this paper, we consider a very difficult localization task under several constraints. First, in some hostile environments, such as battlefield and fire monitoring system, sensors are short-lived since they may be destroyed. To save system cost, it is not permitted to use expensive anchors within the deployment area. Instead, anchors must be installed only at the edges or corners of the deployment area, which are assumed safe. Second, the number of anchors should be minimal. Third, the localization algorithm must be range-free, meaning that it is impossible to estimate the absolute point-to-point distance based on received signal strength. Finally, the locations of sensors within the deployment area are unknown, and each sensor has limited radio range only.

Under the above constraints, most existing localization schemes [2], [5]-[12] cannot achieve good location estimation without relying on a large number of anchors. This paper is targeted at solving this problem and has made the following contributions.

First, based on random sensor deployment (the cheapest and the most convenient deployment method [13] for large scale sensor networks), we propose a Very Low Cost Localization (VeLoC) scheme that only requires four powerful anchors at each corner of a rectangular deployment area and achieves...
desirable localization accuracy. Second, we analyze the relationship between the estimation accuracy and the density of sensors, illustrating that the proposed scheme requires high density of sensors if each sensor communicates with its one-hop neighbor only. Third, based on our analysis, we propose an enhanced method that collects multi-hop neighborhood information to improve estimation accuracy. Finally, simulation evaluation is provided to demonstrate the effectiveness of the VeLCoL scheme.

II. VERY LOW COST LOCALIZATION (VeLCoL) FOR HOSTILE ENVIRONMENTS

A. Illustration of VeLCoL

We use a rectangular deployment area as an example to illustrate the operation of VeLCoL. The design principle of VeLCoL is applicable to other shapes of deployment area, if the shape and the size of the area are known in advance or can be estimated with some methods [3].

As shown in Fig. 1, we assume that all sensors are deployed randomly and independently within a rectangular hostile area. To minimize the system cost, we place four powerful anchor nodes, each at a corner of the rectangle. The radio range for each anchor node should be larger than the diagonal length of the rectangle. Such requirement is to guarantee that all sensors within the deployment area can receive messages from anchors. This requirement is not expensive since the number of anchors is small (four) and anchors are located at safe positions. Note that we assume the edges and the corners of the deployment area are safe.

Since the anchors are in corners, when we draw circles centered at anchors, any two circles with centers at the same edge, if intersecting, can only have one intersection point within the rectangular area. Therefore, if we can estimate the distances between a sensor and two anchors at the same edge, the location of the sensor can be decided. For example, in Fig. 1, if sensor $S$ can estimate its distance to anchors $A_1$ and $A_2$ that are at the same edge, its location can be determined as the intersection of two circles— one is centered at $A_1$ with radius equal to the distance between $A_1$ and $S$, and the other is centered at $A_2$ with radius equal to the distance between $A_2$ and $S$.

Notably, although in principle two anchors are enough for the above method if the size and the boundary of the area are known in advance, it should be much easier in practice to use four anchors at each corner of the area, because four anchors clearly indicate the boundary of the deployment area. Moreover, four anchors can improve overall estimation accuracy by decreasing the distance from sensor nodes to reference anchors.

As stated before, it is very inaccurate to estimate the absolute distance between a sender and a receiver by measuring received signal strength, since such estimation relies on radio condition and transmission environment. Nevertheless, experiments demonstrate that in a roughly same direction, a receiver far away from the sender will receive weaker signal strength than a receiver near the sender [5]. VeLCoL utilizes this fact to estimate the distance between a sensor and an anchor. Specifically, VeLCoL is based on two assumptions. First, the individual sensors have a fixed radius of broadcast whose maximum length can be accurately estimated. This assumption is trivial since the maximum radio transmission distance can be obtained by simply checking sensors’ hardware specification. Second, the maximum radio transmission distance of an anchor is larger than the diagonal length of the rectangle, and the closer a receiver is to an anchor, the stronger the signal received by the receiver from the anchor. However, neither the correlation between the distance from the anchor to the receiver nor the correlation between the radius and the signal strength of the anchor is known in advance.

Initially, all anchors broadcast their locations. Each sensor records the anchors’ locations and the signal strength received from the anchors. After that, it locally broadcasts a RASS (Received Anchor Signal Strength) message, indicating the received signal strength from different anchors. By collecting this information from neighboring sensors and comparing this information to its own received signal strength, a sensor can know the number of neighboring sensors (denoted as $n_1$), which have received weaker signal strength from a specific anchor, as well as the number of neighboring sensors (denoted as $n_2$), which have received stronger signal strength from that anchor. Note that neighboring sensors having the same received signal strength from that anchor will not be taken into consideration. As shown in Fig. 1, if sensor $S$ knows the above values regarding anchor $A_1$, it can estimate the size of the intersection area of two circles: one circle is centered at anchor $A_1$ with the radius equal to the distance between $A_1$ and $S$, and the other circle is centered at $S$ with radius equal to $S$’s maximum radio transmission distance. Divided by $S$’s maximum radio coverage, the size of the intersection area can be approximated as $\frac{n_2}{n_1+n_2}$ if all sensors are randomly and independently deployed. We call this value as the size of the normalized intersection area and denote it as $I$.

Knowing its maximum radio range and the value of $I$, sensor $S$ can then estimate its distance to anchor $A_1$. To save computing cost, each sensor can build a table, indicating the one-to-one relationship between $I$ and the distance. Knowing the distance to two anchors at the same edge, a sensor can estimate its location as shown in Fig. 1. The details of building such a table will be introduced in the following section.

Fig. 1. An Example of VeLCoL.
B. Mapping between $I$ and Distance

As shown in Fig. 2, assume that the radio range of sensor $S$ is 1 and the distance between sensor $S$ and anchor $A$ is $d(>1)$. The size of the shadowed area, denoted as $\Delta$, is:

\[
\Delta = \int_{0}^{1} \sqrt{d^2 - (x - d)^2} \, dx + \int_{d}^{1} \sqrt{1 - x^2} \, dx
\]

\[= \int_{-d}^{\frac{1}{2}d} \sqrt{d^2 - x^2} \, dx + \int_{\frac{1}{2}d}^{1} \sqrt{1 - x^2} \, dx \]

\[= d^2 \int_{-1}^{\frac{1}{2}d} \sqrt{1 - x^2} \, dx + \int_{\frac{1}{2}d}^{1} \sqrt{1 - x^2} \, dx \]

\[= d^2 \left( f(\frac{1}{2}d^2) - 1 - f(-1) \right) + \left( f(1) - f(\frac{1}{2}d) \right) \]  

(1)

where $f(x) = \frac{x}{2} \sqrt{1 - x^2} + \frac{1}{2} \arcsin x$. Therefore, the size of the normalized intersection area is

\[I = \frac{2 \times \Delta}{\pi}. \]  

(2)

Based on Equation (1) and Equation (2), if $I$ is obtained by calculating $\frac{n_1}{n_1 + n_2}$ as in the previous section, the distance $d$ can be calculated from the inverse function of $\Delta$. However, such calculation will be very complicated and time consuming. To avoid this, we can simply build a table including different $d$ values and the corresponding $I$ values. Note that $d$ could be float values depending on required accuracy. This table can be stored initially in each sensor. Once a sensor has calculated the size of the normalized intersection area, it simply finds the $d$ whose corresponding $I$ value in the table is the closest to $\frac{n_1}{n_1 + n_2}$.

Fig. 3 shows the numerical results of $I$ with changing $d$. From the figure, when $d$ value is large, a small change of $I$ will result in a large difference in $d$. Therefore, the estimation of $I$ might be very inaccurate when $d$ is large. We will analyze the problem that how many sensors are required to get a good estimation of $I$ in the next section.

C. Analysis on Estimation Accuracy

Intuitively, the accuracy of VeLCoL depends on the estimation of the normalized intersection area $I$. Since VeLCoL uses the ratio of number of sensors to estimate the ratio of the size of areas, the accuracy of such approximation is acceptable only when we collect enough RSSA information from a large amount of nearby sensors. We have two ways to achieve this purpose: we can make the sensor network very dense, or we can let each sensor broadcast the RASS message multiple hops so that each sensor can collect the RASS information from other sensors in a larger range. Obviously, multihop broadcast of RASS is more practical, since increasing network density will inevitably increases total system cost. We now analyze the relationship between the number of neighboring (or neighboring in multiple hops) sensors and the estimation accuracy, that is, how many sensors are required to estimate the value of $I$ with high confidence.

The problem can be modeled as follows. Throw $n$ points randomly and independently within a circle. For a specific area within the circle, each point falls within this specific area with the probability of $p(=I)$. What is the probability that the number of points within the specific area over $n$ is larger than $(1+\delta)p$ or smaller than $(1-\delta)p$ where $\delta$ is an arbitrarily small value? If this probability is very small, we can claim that the estimation of $I$ is accurate.

Let $X_1, \ldots, X_n$ be a sequence of independent Poisson trials. For $i = 1, \ldots, n$, let $X_i = 1$ if the $i^{th}$ point falls within the intersection area and 0 otherwise, that is, $Pr(X_i = 1) = p = I$ and $Pr(X_i = 0) = 1-p$. Let $X = \sum_{i=1}^{n} X_i$, and $\mu = E[X] = np$. Using Chernoff bound, we can get

\[Pr(|X - \mu| \geq \delta \mu) \leq 2e^{-\frac{\delta^2 \mu}{3}}. \]  

(3)

Note that the left hand of (3) is equivalent to $Pr(\frac{X}{n} \geq (1+\delta)p) + Pr(\frac{X}{n} \leq (1-\delta)p)$. The above inequality indicates that the larger the number of sensors, the smaller the value of the right hand of the above inequality, and thus the more accurate the estimation of $I$.

Based on the above analysis, it will be impractical to get reasonably good location estimation if a sensor only knows the RASS information of its direct neighbors. According to the above inequality, if we assume that $p = 0.4$, $\delta = 0.2$, the $n$ value should be larger than 432 to make the right hand of the inequality smaller than 20%. It will be unrealistic for a sensor having over 400 direct neighbors. In the next section,
we propose a method to solve this problem.

D. Enhancement of VeLCoL

Fortunately, the above problem can be effectively solved by making sensors broadcast RASS messages multiple hops. Broadcasting RASS messages for multiple hops has the following advantages. First, each sensor will be able to collect more RASS messages from its nearby sensors. This is equivalent to increasing the value of $n$ in Inequality (3), and thus the estimation accuracy can be improved. Second, the real meaning of $d$ in Equation (1) is actually the distance between the sensor and the anchor normalized by the distance within which the sensor can collect RASS information. Multihop broadcast lets each sensor collect RASS information from a large range, implying that the $d$ value is decreased. As shown in Fig. 3, when $d$ is small, it is not required to have a very accurate estimation on $I$, because small change in $I$ will not cause big difference in $d$ in this case.

Such multihop broadcast of RASS messages obviously consumes more energy. Nevertheless, the cost is acceptable considering the fact that localization is a one-time task and in the worst case the total number of RASS messages is $n^2$ even if each RASS message is flooded globally, where $n$ is the total number of sensors in the network.

Note that the memory cost is negligible. Each sensor only needs to maintain two counters for each anchor: one for recording the number of sensors that have stronger received signal strength from the anchor, and the other for recording the number of sensors that have weaker received signal strength from the anchor. In order to avoid infinite loop of message broadcast, each sensor needs to remember the IDs of sensors whose RASS message has been received and broadcast. The maximum memory cost for such purpose is $n$ where $n$ is the total number of sensors in the whole network. If RASS messages are not broadcast globally, the actual memory cost in each sensor will be much less than $n$. Furthermore, localization is usually done immediately after sensor deployment, and all memory can be released for other tasks after localization.

As demonstrated via following simulation, the above enhancement, although seeming trivial, is very effective in improving estimation accuracy.

III. SIMULATION EVALUATION

A. Simulation Model

We simulated VeLCoL in a 500 meter x 500 meter square area with four anchors deployed at each corner. The radio range for all sensors was set to 50 meters. After a sensor estimates its distance to different anchors, it selects two nearest anchors at the same edge to decide its location. We also studied the performance of VeLCoL with and without enhancement. In the enhanced VeLCoL scheme, we assumed that RASS messages were flooded globally.

For comparison purpose, we simulated the DV-Hop [9] localization algorithm, which uses similar message flooding mechanism but estimates distance based on hop count between two nodes. With this method, the anchors flood their location information to all sensor nodes in the networks. Each sensor node maintains the minimum hop count value to each anchor. As such, all nodes, including sensors and anchors, can obtain the smallest hop count to every anchor. Each anchor then calculates the average length of each hop, called $\text{hopSize}$, as the ratio of the sum of the absolute distance to other anchors over the sum of the minimum hop count to other anchors. After the calculation, it broadcasts the $\text{hopSize}$ information to the whole network. A sensor node will take $\text{hopCount} \times \text{hopSize}$ as its estimated distance to a specific anchor, where $\text{hopCount}$ is the minimum hop count to the anchor and the $\text{hopSize}$ is the $\text{hopSize}$ information broadcast from that anchor. Once a sensor gets its distance to three or more anchors, it uses multilateration to calculate its location.

The location estimation error is defined as the Euclidian distance between the real location of a node and its estimated location. For each simulation scenario, thirty runs with different random seeds were executed and the results were averaged. The normalized average location error is used as the metric to evaluate the accuracy of location estimation. It is defined as the mean of location estimation errors collected over all sensor nodes in thirty runs normalized by sensors’ radio range.

B. Simulation Results

Fig. 4 shows the results of normalized average location error with DV-Hop, VeLCoL without enhancement, and VeLCoL with enhancement. From the results, we can observe that estimation accuracy for both VeLCoL and DV-Hop is improved with the increase of number of sensors. This is because both methods rely on the density of networks to obtain good estimation. According to Inequality (3), a high density means a large $n$ value and thus a small estimation error for VeLCoL. Similarly, a high density implies accurate estimation on $\text{hopSize}$ for DV-Hop. Note that although the distance from a sensor to reference anchors can impact estimation accuracy according to the discussion in Section II-B, the impact is slight for the enhanced VeLCoL scheme since RASS messages are flooded globally in the simulation. Furthermore, the strategy that a sensor only selects two nearest anchors to decide its location greatly reduces the location errors.

We can also observe that VeLCoL depending only on one-hop neighbor information has very bad estimation performance. With enhancement, the estimation accuracy of VeLCoL can be improved greatly. This phenomenon has been explained in our previous analysis (refer to Sections II-C and II-D).

From Fig. 4, it is clear that for the same network density, the enhanced VeLCoL has smaller estimation errors than DV-Hop, with number of sensors ranging from as small as 400 to as large as 1400. We carefully traced the simulation and found that the main factors causing the estimation error of DV-Hop are the inaccuracy on $\text{hopSize}$ and the inaccuracy on $\text{hopCount}$. Since the $\text{hopSize}$ value transmitted from an anchor is calculated as the ratio of the sum of its absolute distance to other anchors over the sum of its minimum hop count to other anchors, this average value may be too large or too small for some specific sensors, resulting in inaccurate
distance estimation for these sensors. This is the inherent problem of DV-Hop and is hard to solve. We have tried different sensor radio ranges to test DV-Hop, but unfortunately, the observed performance remains roughly the same or worse.

IV. CONCLUSION

This paper adds a new idea into the pool of sensor localization algorithms. The presented VeLCoL algorithm is aimed at supporting sensor localization for hostile environments where sensors may be destroyed finally. To save system cost, such application scenarios require extremely low cost per sensor, and it is undesirable to deploy expensive anchor nodes within the hostile field. Moreover, the absolute point-to-point distance estimation should not depend purely on received signal strength, since such estimation is subject to transmission media and surrounding environment and requires expensive hardware support. Under these constraints, most existing localization algorithms cannot work well without relying on a large number of anchor nodes [2], [5]-[12].

Our proposed algorithm does not use received signal strength directly to estimate absolute distance. Instead, the distance is estimated based on the comparison of relative signal strength among sensors. VeLCoL also utilizes the fact that for random sensor deployment, the number of sensors within an area is proportional to the size of that area.

Our analysis illustrates that a sensor requires RSSA information from a large amount of multihop neighbors in order to achieve acceptable estimation accuracy. This fact implies that VeLCoL is effective for large scale, dense networks with up to thousands of sensors. For networks with very small number of sensors, we should not expect VeLCoL to work well. Nevertheless, when the number of sensors is very small, the total system cost is not a big concern in any rate, since localization can be easily solved by installing GPS in each sensor.

As our final remarks, we want to mention some drawbacks of using VeLCoL. First, the irregularity of radio pattern will cause inaccurate location estimation. For a very irregular radio model that greatly changes the circular shape of radio propagation, VeLCoL is not effective. Second, sensors near the boundaries of the deployment area will have a less accurate location estimation. We stress that the above problems also exist in most existing range-free methods [7] and deserve further investigation.

ACKNOWLEDGMENT

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and Canada Foundation for Innovation (CFI).

REFERENCES