

Distributed Localization Scheme for Mobile Sensor Networks

Jang-Ping Sheu, *Fellow, IEEE*, Wei-Kai Hu, *Student Member, IEEE*, and Jen-Chiao Lin

Abstract—Localization is an essential and important research issue in wireless sensor networks (WSNs). Most localization schemes focus on static sensor networks. However, mobile sensors are required in some applications such that the sensed area can be enlarged. As such, a localization scheme designed for mobile sensor networks is necessary. In this paper, we propose a localization scheme to improve the localization accuracy of previous work. In this proposed scheme, the normal nodes without location information can estimate their own locations by gathering the positions of location-aware nodes (anchor nodes) and the one-hop normal nodes whose locations are estimated from the anchor nodes. In addition, we propose a scheme that predicts the moving direction of sensor nodes to increase localization accuracy. Simulation results show that the localization error in our proposed scheme is lower than the previous schemes in various mobility models and moving speeds.

Index Terms—Localization, mobile sensors, wireless sensor networks.

1 INTRODUCTION

IN recent years, wireless sensor networks (WSNs) [1], [5] have been widely used in a wide range of applications such as military operations, medical treatments, and the monitoring of animal activity and the environment in the forest. The basic assumption in many applications is that sensor nodes have to know their positions. For example, the sensed data must combine with location information, for a server instantly to know where an event has happened. In order to get sensors' positions, one simple and precise solution is that each sensor node must carry Global Positioning System (GPS) equipment. Unfortunately, it is too expensive to realize and is useless indoors. Moreover, most applications require coarse localization accuracy. As such, the reasonable solution is that some nodes of sensor network should be equipped with a GPS device, while the others get their positions automatically by a localization scheme. In general, the location-aware nodes are called *anchor nodes*, and the remaining nodes are called *normal nodes*.

Many localization schemes have been proposed in the past few years. Most of them are designed for static sensor networks [11], [13], [14], [20], [26]. However, some applications assume that sensors are mobile and location-aware. For example, in target tracking, the sensor nodes know their areas by tracking locations of moving objects.

In addition, sensor nodes are mobile for enlarging the sensing region. Thus, a designed localization scheme for mobile sensor networks is necessary. A Monte Carlo Localization (MCL) scheme specifically designed for a mobile sensor network is proposed in [12]. In MCL, all sensor nodes are mobile. Each normal node collects the locations of its one-hop and two-hop anchor nodes via message exchange, and constructs a new possible location set in each time slot. The possible location set consists of various coordinates where the normal node may locate. The possible locations are also constrained by the communication range of anchor nodes and the moving region of location set in the previous time slot. However, the localization error with low anchor density in MCL does not work well. The Mobile and Static sensor network Localization (MSL*) [19] is one another range-free algorithm that uses the Monte Carlo method. MSL* improves localization accuracy by using the location estimation of all neighbors (not just anchor nodes). The above methods are time-consuming because they need to keep sampling and filtering until enough samples are obtained to construct a new possible location set in each time slot. A bounding-box (BB) method used to reduce the scope of searching the candidate samples is proposed in [22].

In this paper, we propose a distributed localization approach based on the Monte Carlo method to improve the localization error of previous work. The possible locations of a normal node are not only constrained from anchor nodes but also constrained from its one-hop normal nodes whose locations are estimated from the anchor nodes. Furthermore, each normal node predicts its moving direction to filter some impossible positions from the possible location set. Simulation results show that the performance of our proposed scheme is better than those of previous schemes.

The rest of this paper is organized as follows: In Section 2, we review the related work of localization schemes. In Section 3, we describe our proposed scheme. Simulation results are shown in Section 4. Section 5 concludes this paper.

- J.-P. Sheu is with the Department of Computer Science, National Tsing Hua University, 101, Section 2, Kuang-Fu Road, Hsinchu 30013, Taiwan, ROC. E-mail: sheujp@cs.nthu.edu.tw.
- W.-K. Hu is with the Department of Computer Science and Information Engineering, National Central University, No. 197-14, Jianxia, Toufen Township, Miaoli County 351, Taiwan, ROC. E-mail: cathie@axp1.csie.ncu.edu.tw.
- J.-C. Lin is with the Mobile Devices Department, Chi-Mei Communication Systems, Inc., 6F., No. 1-8, Nanya E. Rd., Banqiao City, Taipei County 220, Taiwan, ROC. E-mail: jenchiao@axp1.csie.ncu.edu.tw.

Manuscript received 19 Nov. 2008; revised 29 Apr. 2009; accepted 4 Aug. 2009; published online 12 Aug. 2009.

For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-2008-11-0466. Digital Object Identifier no. 10.1109/TMC.2009.149.

2 RELATED WORK

The localization schemes can be divided into two categories: range-based and range-free [25]. The range-based localization schemes depend on calculating the absolute distance or angle between two nodes. Each node can estimate distance by Time of Arrival (TOA), Time Difference of Arrival (TDOA), or Received Signal Strength Indicator (RSSI). Under the TOA and TDOA approaches, we can calculate the distance between any two sensor nodes according to the signal propagation time between them and the velocity of signal. With Angle of Arrival (AOA), the sensor node usually equips directional antenna and estimates the relative angle with neighbors. In [15], sensor nodes use the RSSI value to infer the physical distance. After getting the distance information, the node estimates its position by triangulation or rigidity [8], [16].

On the contrary, the sensor node under range-free schemes estimates its position by network connectivity instead of absolute distance. Each sensor node confirms the connectivity with neighbors by transmitting a packet and estimates its location by gathering information from neighboring nodes. It only achieves a low level of accuracy in most situations, but it is easy to implement for WSNs. Many range-free localization schemes for static sensor networks have been proposed recently. In most localization schemes, each normal node estimates its position by collecting the locations of anchor nodes. The anchor nodes are divided into *near anchors* and *farther anchors*. The normal nodes can receive messages directly from near anchors. Conversely, the farther anchors are two-hop away from the normal nodes.

The authors in [4] proposed a Centroid algorithm. Each sensor node collects the positions of near anchor nodes and averages these positions as the estimated location. The transmission range of a sensor node is also a useful term for localization scheme [6]. In the Convex Position Estimation (CPE) algorithm [7] and Distributed Location Estimating (DLE) algorithm [21], the transmission region of the near anchor is represented by a square, and the normal node is located in the overlapping region of multiple squares that is called the Estimated Rectangle (ER). The estimated location with CPE and DLE is the central point of the area of ER.

Except for the messages from anchor nodes, the information received from location-unaware normal nodes is also useful in localization scheme. In Sextant [9], the authors proposed the relative constraints from the neighboring normal nodes. Some localization schemes proposed in [23], [24] increase the accuracy by using the moving anchor nodes. The anchor node moves around the distributed field and periodically broadcasts a beacon packet with its current position. Each normal node can collect more information from mobile anchors, and then, the localization error decreases.

The MCL scheme for mobile sensor networks is proposed in [12]. There are two basic assumptions in MCL to make the localization problem for mobile sensors simple. First, the time is divided into several time slots. Second, the maximum moving distance of every sensor node in each time slot does not exceed V_{max} . In MCL, each anchor node periodically forwards its actual location to

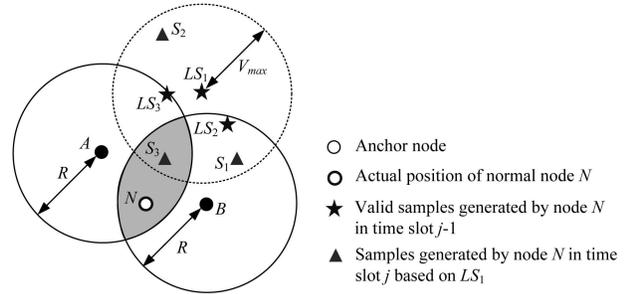


Fig. 1. An example for choosing valid samples.

two-hop neighbors. If a normal node receives messages from anchors, its location is restricted by *anchor constraints* that determine where the normal node is possibly located inside the region. The anchor constraints can be sorted into *near anchor constraints* and *farther anchor constraints*. A near anchor constraint is the communication region with radius R centered on the location of a near anchor. A farther anchor constraint is the region within $(R, 2R)$ centered on the location of a farther anchor. After gathering the locations of anchors, the normal node starts to estimate its location. The process of estimating the location of a normal node can be separated into three phases: initial phase, prediction phase, and filtering phase.

During the initial phase, each normal node constructs a possible location set $L_0 = \{l_0^0, l_0^1, l_0^2, \dots, l_0^{49}\}$ that consists of 50 coordinate points, called *samples*, used to represent its possible located positions. Each normal node in each time slot t then establishes a new possible location set L_t . In the prediction phase, each normal node selects new samples from the circle with radius V_{max} centered on each sample of L_{t-1} . Since these selected samples are the possible located positions of the normal node, they must locate within all near anchor constraints and farther anchor constraints. In the filtering phase, the impossible samples located outside the anchor constraints will be removed from the new location set L_t . For getting enough samples, the prediction and filtering phases will be repeated until the sample set L_t is fulfilled by 50 samples. The estimated location of the normal node is determined by the average of the locations of 50 samples.

For example, in Fig. 1, assume that a normal node N in time slot j can receive the location messages transmitted from anchor nodes A and B . Then, normal node N needs to generate new samples to represent its possible locations in current time slot j . The new samples are randomly selected from the circle with radius V_{max} centered on each sample of the last time slot, where V_{max} represents the maximum moving distance of a normal node during a time slot. However, the newly selected samples must be located in the communication range of anchor nodes A and B , as shown in the shaded region of Fig. 1. For example, LS_1 is a valid sample selected in time slot $j-1$. Normal node N will randomly generate a new sample from the circle with radius V_{max} centered on LS_1 . The new generated sample will be verified whether it is located in the communication range of anchor nodes A and B or not. If the new sample cannot satisfy the anchor constraint, we will randomly generate another new sample based on LS_1 until a sample

can satisfy the constraint. In Fig. 1, S_3 is a valid new sample for LS_1 but S_1 and S_2 are not valid samples, which are not located in the communication range of anchor nodes A and B . Normal node N will repeat the above procedures to replace all the samples generated in time slot $j - 1$. It may cost several times of sampling to find a valid sample. After finding the enough new samples, normal node N can estimate its current location as the average locations of these valid samples.

The Monte-Carlo-Localization-Boxed (MCB) scheme based on MCL is proposed in [2]. The anchor constraints are bounded by squares, which are called anchor boxes. Thus, the new samples in time slot t are randomly chosen from the overlapping rectangle of anchor boxes and the extended region of previous sample region in time slot $t - 1$. The anchor information is not only used in the filtering phase, but also in the prediction phase. MCB successfully decreases the probability of choosing wrong samples and increases the efficiency on sample selection, but it cannot reduce the localization error if MCB and MCL have the same number of valid samples.

The sampling approach in MCL successfully draws the possible locations of normal nodes. However, MCL and MCB adopt the static number of samples and filter the impossible samples by using the information from anchors only. There are two drawbacks in the above methods. First, in low anchor density, each normal node gets less anchor constraints, and the estimated location error becomes large. Second, in high anchor density, each normal node gets more location constraints from anchors, and the possible located region of the normal node becomes small. This will increase the possibility for the selected samples to get close to each other. If the overlapping region of anchor constraints is fairly small, a few samples may be enough to represent the possible located positions of a normal node. Moreover, it is difficult to find many valid samples in a small region. Therefore, too many close samples in a small region cannot increase the localization accuracy, but instead cause unnecessary waste of memory and computation overhead.

The MSL^* scheme [19] is another range-free algorithm that uses Monte Carlo method. In MSL^* , the authors improve on MCL by using the information from all one-hop and two-hop neighbors of normal nodes and anchor nodes. Each normal node adapts information from only those neighbors that have better location estimated than it. This modification results in faster convergence of the localization error and better estimation of the locations. However, there exist two disadvantages for the MSL^* scheme. First, MSL^* has lower location accuracy in higher mobility environment. Second, MSL^* spends a lot of communication cost in forwarding location information. In [22], the authors proposed a localization algorithm (BB) that can reduce the computation cost of obtaining the samples. The BB scheme can achieve higher location accuracy under higher density of common neighbor nodes.

3 OUR PROPOSED SCHEME

Since the location of a normal node is estimated from the average locations of valid samples, the location estimated by the normal node will be close to its actual position if the valid

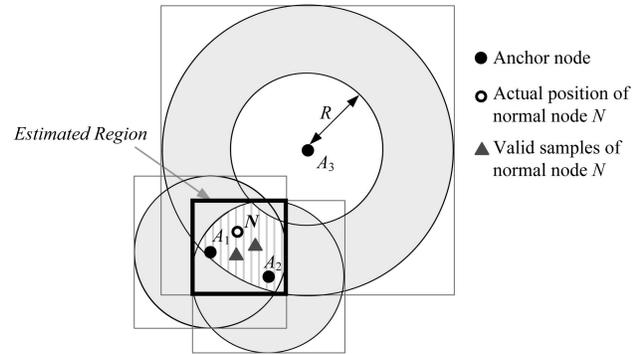


Fig. 2. Example of ER.

samples are near to the actual position of the normal node. In our Improved MCL (IMCL) localization scheme, we propose three constraints including anchor constraint, neighbor constraint, and moving direction constraint to confine the region of the valid samples near the actual position of the normal nodes. The assumptions of our scheme are the same as MCL: time is discrete. A few sensors, called as anchor nodes, already know their locations. In each time slot, anchor nodes broadcast their physical locations and the remaining nodes estimate their locations after gathering location information from the neighboring nodes. The maximum moving distance of all nodes during one time slot will not exceed V_{max} , and the communication range of all sensors is fixed by R .

Our proposed IMCL consists of three phases: sample selection phase, neighbor constraint exchange phase, and refinement phase. In each time slot, each normal node executes these three phases once and gets its estimated location. In the first phase, each normal node determines the number of samples by the location information received from anchors, and then, chooses samples in the possible location set. In the second phase, each normal node broadcasts its possible located region to neighbors. In the last phase, the samples will be refined by neighbor constraint and moving direction constraint to improve the localization error. Finally, each normal node can estimate its location as the average of the locations of samples in the location set. The detail of each phase is described as follows:

3.1 Sample Selection Phase

In this phase, each normal node gathers the locations of neighboring anchor nodes and selects samples to represent its possible located positions. The samples are selected from the circle with radius V_{max} centered on each sample in the last time slot. The selected samples must be placed in the *sampling region* whose sampling points satisfy the near anchor and farther anchor constraints as shown in the region with oblique lines of Fig. 2. Note that a sampling point satisfying *near anchor constraint* means that the point is located within the intersection region of near anchors. And a sampling point satisfying *farther anchor constraint* means that the point is within $2R$ communication range but is not in R communication range of the *farther anchor*. For example, the small triangles in Fig. 2 are valid samples which satisfy the *near anchor constraint* and *farther anchor constraint*. The valid samples are filtered out and become

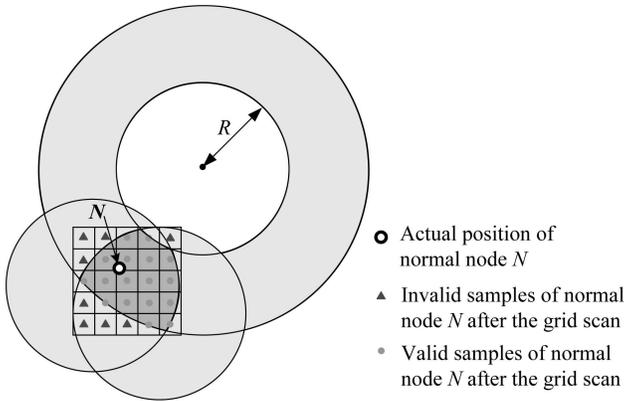


Fig. 3. The sample filtering procedure in the first time slot.

the actual samples to represent the possible positions of a normal node in this phase. In fact, the sampling region becomes smaller and the selected samples will be close to each other if there are more anchor constraints. A lot of samples located in a small region cannot increase the localization accuracy, but they will waste memory space and computation cost. Thus, in IMCL, each normal node will dynamically adjust the number of the samples, which is proportional to the area of sampling region.

At the beginning of IMCL, each anchor node broadcasts its physical location to its one-hop neighbors. This information packet will be forwarded to two-hop neighbors of the anchor nodes. After collecting packets from near anchors and farther anchors, each normal node will decide the number of samples based on the size of sampling region. However, the area of the sampling region is irregular and it is difficult to calculate for the resource-limited sensors. Thus, a rectangle surrounding the sampling region called ER is used to replace the exact sampling region on deciding the number of samples.

Each normal node utilizes squares with edge length $2R$ and $4R$ to surrounding near anchor constraint and farther anchor constraint, respectively. The overlapping region of these squares is defined as an ER. An example with two near anchors and one farther anchor is shown in Fig. 2. Assume that sensor N can receive packets from near anchors A_1 and A_2 , and farther anchor A_3 . Thus, sensor N will locate in the ER region, which is the intersection area of the three squares with centered at A_1 , A_2 , and A_3 , respectively. The boundaries of ER can be easily calculated by comparing the borders of all squares.

Although the ER does not show the actual shape of the possible sampling region, it can show the area variations of the possible sampling region. Thus, each normal node will evaluate the number of samples k according to the area of ER. Let $k = \lceil Max_Num \times (ER_{Area} / ER_{Threshold}) \rceil$, where ER_{Area} , $ER_{Threshold}$, and Max_Num denote the area of ER, a threshold value, and the maximum number of samples, respectively. In order to avoid the excess number of samples, k is set to Max_Num if $ER_{Area} \geq ER_{Threshold}$. In our simulations, each normal node will receive at least one near anchor constraint in most situations. Therefore, $ER_{Threshold}$ is set to $4R^2$, which is equal to the area of ER surrounded with one near anchor constraint. Once the

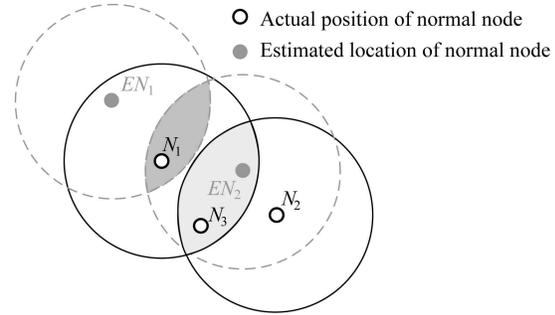


Fig. 4. The overlapping region centered on N_1 and N_2 versus the overlapping region centered on EN_1 and EN_2 .

number of samples is determined, each normal node picks k valid samples that are randomly generated from the region with radius V_{max} centered on samples in the last time slot and located in the sampling region.

Note that in the first time slot, a normal node has no sample in the last time slot, and one way to select the first set of samples is that randomly choose k samples from ER. Here, we use a more effective method. Each normal node divides the area of ER into grids. The default length of a grid side is set to $0.25R$ (R is the radius of communication range). The center points of grids become the initial samples, as shown in Fig. 3. Then, each grid is scanned to check if its center point can satisfy the *near anchor constraint* and *farther anchor constraint*.

3.2 Neighbor Constraint Exchange Phase

In the MCL scheme, each normal node only uses the constraints arising from anchor nodes, and it does not work well in low anchor density. If a normal node does not receive any anchor's information, it will estimate its position by utilizing the samples selected in the last time slot, and the localization error will become large until the new location information from the anchors is received. In order to improve localization accuracy, each normal node in IMCL can rely on the constraints arising from the anchor nodes and neighboring normal nodes. An additional constraint is that each normal node must locate in the communication range of its neighboring normal nodes. Note that the location of a normal node is estimated from its neighbor locations and there exists error between the estimated and actual positions. If we directly use the estimated locations to be the positioning constraints of normal nodes, it may increase the localization error of the normal nodes.

An example is illustrated in Fig. 4. Assume that the normal node N_3 has two neighboring normal nodes N_1 and N_2 whose estimated locations are EN_1 and EN_2 , respectively. The possible located region of N_3 will be located in the overlapping region of the two circles centered on N_1 and N_2 , as shown in Fig. 4. If we use EN_1 and EN_2 as the actual positions of N_1 and N_2 , N_3 will be considered in the overlapping region of two circles centered on EN_1 and EN_2 . However, the overlapping region centered on N_1 and N_2 is different with the overlapping region centered on EN_1 and EN_2 . In order to reduce the localization error accumulated from the neighboring normal nodes, each normal node will broadcast its *possible located region* instead of its estimated

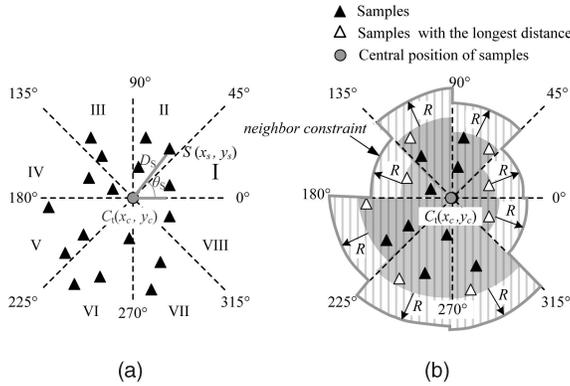


Fig. 5. (a) Samples are divided into eight groups. (b) The possible located region with neighbor constraint.

position to neighbors. In this paper, the possible located region of a normal node is enclosed by the distribution region of samples selected in the *sample selection phase* of current time slot. Since the distribution region of samples is irregular, each normal node will estimate its possible located region as follows:

First of all, each normal node calculates its central point $C_t(x_c, y_c)$ of samples selected in the sample selection phase. The variables x_c and y_c are calculated by averaging the x -coordinate and y -coordinate of samples, respectively. Then, each normal node constructs a two-dimensional coordinates and uses (x_c, y_c) as the origin. The coordinates are partitioned into eight directions and each direction differs by 45 degree. The selected samples can be divided into eight groups according to their direction angles beginning with positive x -axis. The reason of partition the selected samples into eight groups will be explained in later. An example is shown in Fig. 5a. The solid circle represents the central point $C_t(x_c, y_c)$ of samples, and samples are denoted as triangles. Assume that a sample s is placed in (x_s, y_s) . The direction angle of sample s can be calculated by (1):

$$\theta_s = \tan^{-1}\left(\frac{y_s - y_c}{x_s - x_c}\right). \quad (1)$$

The distance between s and central point $C_t(x_s, y_s)$ is

$$D_s = \sqrt{(x_s - x_c)^2 + (y_s - y_c)^2}. \quad (2)$$

In order to estimate the possible located region of each normal node, we will pick up the sample s with the longest D_s in each group. The longest D_s of each group is used as the sector radius of the corresponding sector, and each normal node can form its possible located region by consisting of eight sectors with the origin (x_c, y_c) , as shown in the shaded region of Fig. 5b. Then, each normal node broadcasts a possible located region packet with the origin (x_c, y_c) and eight sectors' radii to its neighboring nodes. When a normal node A receives a location packet sent from its neighboring normal node B , the estimated location of node A will be constrained in the region which extends R from node B 's possible located region called *neighbor constraint*. An example of the neighbor constraint is shown in Fig. 5b in which the radius of each sector is increased by R from the possible located region.

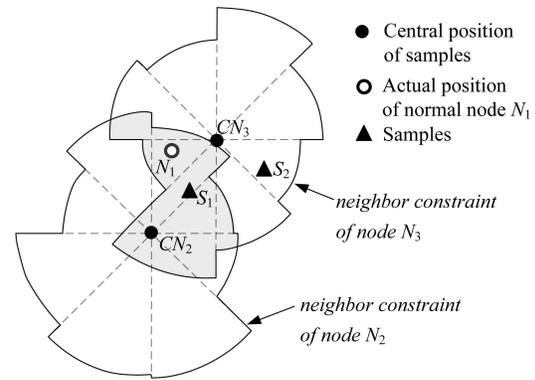


Fig. 6. Sample refinement with neighbor constraint.

Here, we explain why the samples are partitioned into eight sectors to present the possible located region of a normal node. It is obvious that partitioning the samples of a normal node into more sectors is more close to the shape of the possible located region of the normal node. According to our simulations, the localization error decreases as the number of sectors increases. However, the improvement of localization error is stable when the number of sectors is larger than eight. This is because that the difference of the possible located region of a normal node estimated by using eight sectors and more than eight sectors is very small. Therefore, we adopt eight sectors to estimate the possible located region of a normal node.

3.3 Refinement Phase

In this phase, each normal node refines samples selected in the sample selection phase. All impossible samples are filtered by constraints, including the neighbor constraint received from the neighboring normal nodes and moving direction constraint achieved by predicting the moving direction of normal nodes. In order to keep the number of valid samples, if one sample does not satisfy the new constraints, the normal node generates a new valid one to replace it. After receiving neighbor constraint, each normal node checks if each sample satisfies the neighbor constraint. An example is shown in Fig. 6. Assume that node N_1 receives two neighbor constraints from normal nodes N_2 and N_3 and their central positions are C_{N_2} and C_{N_3} , respectively. Assume that S_1 and S_2 are samples of node N_1 selected in the sample selection phase. S_1 is a valid sample that satisfies the neighbor constraints of normal nodes N_2 and N_3 . On the contrary, sample S_2 is an invalid sample because it cannot satisfy the neighbor constraint of node N_2 . Therefore, we will select a new sample to replace S_2 .

An additional constraint called *moving direction constraint* refers to the prediction of the moving direction of normal nodes. Since a mobile sensor node may not change its moving direction during a short time in real applications, we can utilize the estimated locations in the previous two time slots to predict the possible moving direction of a normal node in current time slot. Let E_{t-1} and E_{t-2} be the estimated locations of a normal node N in time slots $t-1$ and $t-2$, respectively. In time slot t , the probability of node N moving along the direction of $\overline{E_{t-1}E_{t-2}}$ is larger

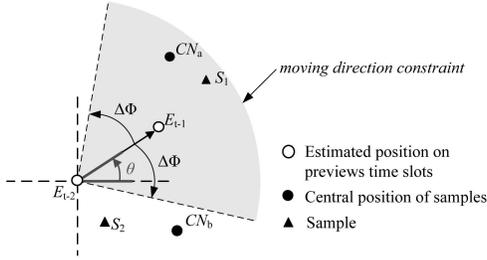


Fig. 7. The moving constraint formed with the estimated locations in previous time slots $t-1$ and $t-2$.

than in other directions. Since the locations E_{t-1} and E_{t-2} are not exact locations of node N , the moving direction of node N will relax with an angle $\Delta\Phi$. That is, if the angle of moving direction $\overline{E_{t-1}E_{t-2}}$ is θ corresponding to x -coordinate, the possible range of moving direction is relaxed to $[\theta \pm \Delta\Phi]$, as shown in Fig. 7. Therefore, the *moving constraint* is the sector centered on E_{t-2} with angle $[\theta \pm \Delta\Phi]$.

Since the mobile node may change its moving direction in each time slot, the moving constraint is used only if the following condition holds. Let C_t denote the central point location that is calculated in neighbor constraint exchange phase in time slot t . If the central point C_t is located in the moving constraint range, it means that the moving direction of normal node in the current time slot is highly related to the previous two time slots. Thus, adopting the moving constraint in the refinement phase can help decrease the localization error. On the other hand, we do not adopt the moving constraint. For example, in Fig. 7, if the central point C_t of a normal node is located in CN_a , the moving constraint will be adopted. If it is located in CN_b , we will not take the moving constraint into consideration. If we adopt the moving constraint, all samples of the normal node will be verified by the moving constraint. For example, if we use the moving constraint in Fig. 7, sample S_1 is a valid sample but S_2 is an invalid one.

In order to confirm if all samples in L_t satisfy all constraints and keep enough samples, each normal node will generate new valid samples to replace the invalid ones. Finally, the estimated location in current time slot t can be calculated by averaging the x -axis and y -axis values of all valid samples. The summary of IMCL is shown below.

Improved MCL Localization Scheme Algorithm

For each anchor node in each time slot t : Anchor node broadcasts a packet with its position. This packet will be forwarded to two-hop neighbors.

For each normal node in each time slot t :

Phase 1: Sample Selection Phase

1. Determines the number of samples k according to the area of ER.
2. Selects k valid samples, which satisfy the anchor constraints, from the regions with radius V_{max} centered on each of k samples in time slot $t-1$.

Phase 2: Neighbor Constraint Exchange Phase

1. Each normal node estimates its possible located region from the k samples selected in Phase 1 and broadcasts the possible located region to neighboring nodes.

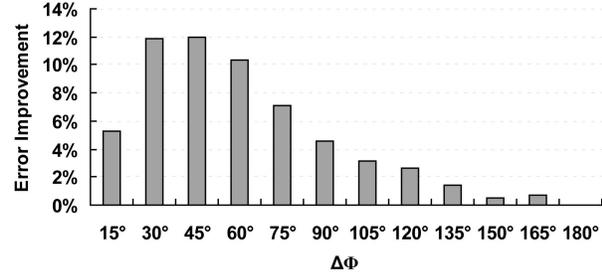


Fig. 8. The improved ratio of localization error with different $\Delta\Phi$.

Phase 3: Refinement Phase

1. Filter impossible samples by neighbor constraint and moving direction constraint.
2. Generate new valid samples to replace the invalid ones.
3. Average the x -axis and y -axis of all valid samples to get the estimated location in the current slot.

4 SIMULATION RESULTS

In order to verify the efficiency of our proposed IMCL scheme, we compare the performance of our protocol IMCL with the MCL [12], MSL* [19], and BB [22] schemes through ns-2 simulator. In our simulations, time is divided into slots and each time slot is equal to 100 seconds. The communication range (R) of all sensor nodes, including anchor nodes and normal nodes, is set by 50 m. All sensor nodes are randomly distributed in a $10R \times 10R$ sensing field. We adopt the modified random-way point mobility model [3], [12] on normal nodes and anchor nodes and each sensor randomly chooses its destination, and then, moves toward it. The moving distance of a sensor in each time slot is randomly selected from $[0, V_{max}]$. The unit of measure of V_{max} is expressed by R . For example, if the V_{max} is 20 m, we denote $V_{max} = 0.4R$. Furthermore, the anchor node density (Ad) is defined as $m/(n+m)$, where n and m are the number of normal nodes and number of anchor nodes, respectively. The estimated error of each normal node is the distance between the estimated position and actual physical location. We evaluate the localization schemes by the localization error, which is the average value of estimated error of all normal nodes.

In IMCL, there are three important variables: the range of angle to predict the moving direction ($\Delta\Phi$), the maximum, and minimum number of samples. In order to understand how these variables affect the localization error of IMCL, we analyze the performance of IMCL under different simulation parameters. The most suitable parameters will be picked and used in simulations of the following sections. First, we will determine the variable $\Delta\Phi$ in the following simulations. Here, the total number of sensors is 350, $V_{max} = R$, and $Ad = 8\%$. As shown in Fig. 8, $\Delta\Phi$ is varied from 15 to 180 degrees. The percentage of improved localization error of normal nodes with different $\Delta\Phi$ is defined as $(Error_{180^\circ} - Error_{\Delta\Phi})/Error_{180^\circ}$, where $Error_{\Delta\Phi}$ is the localization error with the predicted moving angle $\Delta\Phi$. When $\Delta\Phi = 180^\circ$, the normal nodes filter their samples without using the moving constraint. In our simulation,

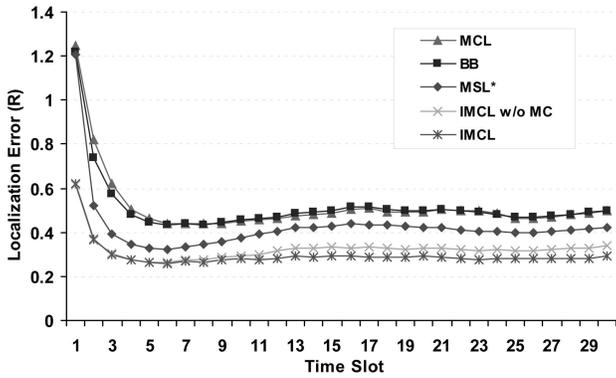


Fig. 9. Comparison of localization error for $V_{max} = R$.

when $\Delta\Phi$ is 45 degree the percentage of improved localization error achieves 12 percent. As the angle increases, the percentage goes down. This is because the tolerable angle error of predicting the moving direction is too large to filter the impossible samples, which are located on incorrect moving direction. However, when the angle is less than 30 degree, each normal node may filter too many right samples, and the localization error goes higher. With these results, we set $\Delta\Phi = 45^\circ$ in all following simulations.

Another important variable is the number of samples. In order to study the effect of the number of samples, the number is varied from 10 to 100. When the number of samples is greater than or equal to 50, the localization error is constant at $0.28R$ in our simulations. In order to reduce the memory space and computation cost, Max_Num is set at 50 in all the following simulations. In our proposed IMCL, sample number k is dynamically adjusted with the area of ER except in the first time slot. In order to observe the effect of the adaptive number of samples for localization error, we fix Max_Num to 50, and let Min_Num be the minimum number of samples. Then, k is equal to Min_Num if $k < Min_Num$ according to the formula $k = \lceil Max_Num \times (ER_{Area}/ER_{Threshold}) \rceil$ which is proposed in Section 3.1. In our simulations, when Min_Num is varied from 50 to 10, the maximum variation of localization error is only $0.01R$. Thus, the localization error is not affected by reducing the number of samples.

4.1 Localization Error

In the following, we demonstrate the localization error of MCL, MSL*, BB, and IMCL. The total number of sensors is 350, $V_{max} = R$, and $Ad = 8\%$. The simulation time is set at 30 time units, and the results are shown in Fig. 9. In the beginning, each normal node does not collect enough precise samples, and the localization error is large. When simulation time goes by, the possibility that the normal node would select the accurate sample is increased. As simulation time passes 10 time units, the localization error becomes stable and fluctuates lightly about a constant value. Accordingly, the localization error in each simulation is the average of the last 20 time slots. To reduce the simulation error, each simulation result is obtained from the average of 20 simulation runs. As shown in Fig. 9, the localization errors of MCL ($0.482R$), BB ($0.487R$), and MSL* ($0.412R$) are larger than the localization error of IMCL ($0.286R$). The localization error of IMCL without

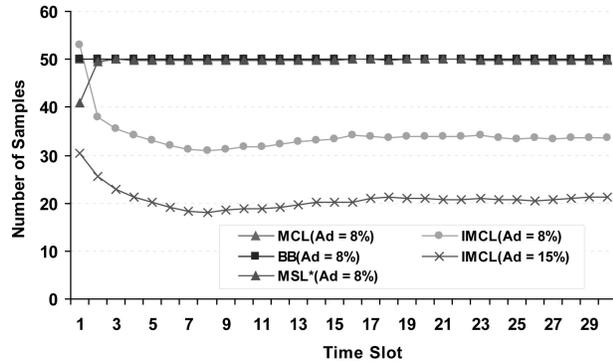


Fig. 10. Number of samples.

moving constraint (IMCL w/o MC) ($0.324R$) is a little higher than IMCL with moving constraint.

Here, we study the average number of samples used in the four schemes. The total number of sensor nodes is fixed at 350 and V_{max} is set by R . In Fig. 10, as the anchor node density is 8 percent, each normal node in MCL, MSL*, and BB keeps 50 samples during each time slot. However, the sample number of IMCL is varied between 31 and 38 except in the first time slot and the average number of samples is about 34. For IMCL, when the anchor node density is increased to 15 percent, the average number of samples reduces to 21. Simulation results show that IMCL utilizes fewer samples than other schemes, and thus, saves the memory usage. Note that it may cost several times of sampling to find a valid sample for all schemes. In our experiments, each node generates 451 (376) samples in average to get 34 (21) valid samples in IMCL when the anchor node density is 8 percent (15 percent). On the other hand, the average times of a normal node to find 50 valid samples for MCL, BB, and MSL* are 1,279, 167, and 868, respectively, when the anchor node density is 8 percent. When the anchor node density is increased to 15 percent, the average times of a normal node to find 50 valid samples for MCL, BB, and MSL* arise to 2,536, 213, and 1,728, respectively. It is more time-consuming for the MCL, BB, and MSL* schemes to find 50 valid samples in a smaller sampling region.

4.2 Impact of the Anchor Node Density

In order to analyze the effect of the anchor node density on localization error, we keep the total number of sensors to 350, $V_{max} = R$, and vary the anchor node density from 1 to 15 percent. The simulation results are shown in Fig. 11. As increasing the anchor node density, each normal node gets more anchor constraints to reduce the possible located region, and thus, the accuracy of estimated location arises. The localization error of each scheme also gets close as the number of anchor nodes increases. The localization error in IMCL is always lower than other three schemes. In low-density state of anchor nodes ($\leq 5\%$), the improved accuracy of our scheme is much better than MCL, MSL*, and BB schemes. Therefore, our protocol can utilize fewer anchor nodes to reach the same performance as MCL, MSL*, and BB. MSL* has the worst performance as the anchor node density is lower than 9 percent. As the anchor node density goes higher than 9 percent, MSL* has lower localization error than MCL.

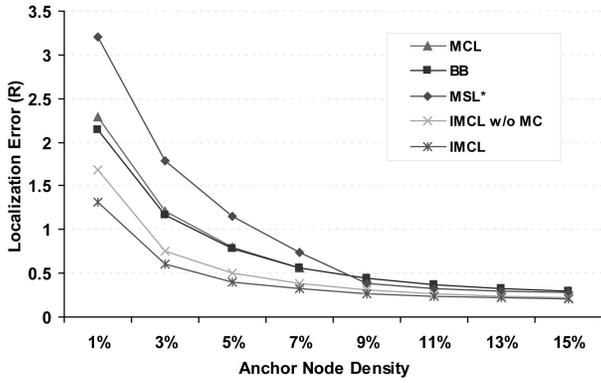


Fig. 11. Anchor node density versus localization error.

4.3 Impact of the Number of Normal Nodes

In this experiment, we fix the number of anchor nodes = 28, $V_{max} = R$, and vary the number of normal nodes to see the impact on localization error. The simulation results are shown in Fig. 12. The localization error of each scheme goes down as the number of normal nodes is increased. Although MCL only uses the location information of one-hop and two-hop anchor neighbors, increasing the number of normal nodes provides a higher probability for a normal node to have two-hop anchor neighbors, and thus, the localization accuracy improves. In BB, MSL*, and IMCL, as the number of the normal nodes goes up, each node can get more locations of one-hop normal nodes and two-hop anchor nodes and have lower localization error. However, the localization error in IMCL will keep stable when the number of normal nodes reaches 480.

4.4 Impact of the Moving Speed

Here, we show the effect of the moving speed to localization error. The total number of sensor nodes is 350 and the anchor node density is fixed by 8 percent. We vary V_{max} from $0.2R$ to $2R$. Fig. 13 shows that the localization error of IMCL is lower than those of MCL and BB with various moving speeds. The localization error of our scheme is also better than MSL* as V_{max} is equal or larger than $0.5R$. Therefore, our protocol is more suitable than MSL* if the network topology is changed quickly.

4.5 Impact of the Irregular Communication Range

There are many physical and environmental factors that affect the wireless signal transmission range. In order to

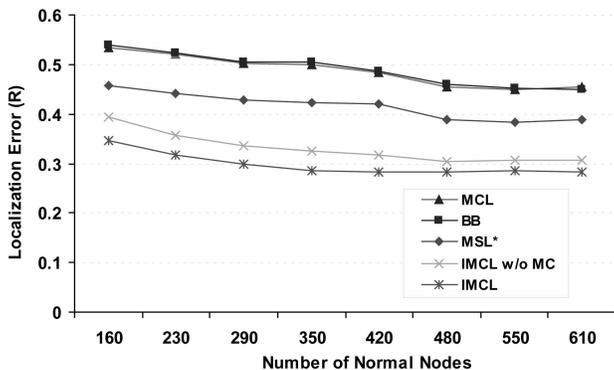


Fig. 12. Number of normal nodes versus localization error.

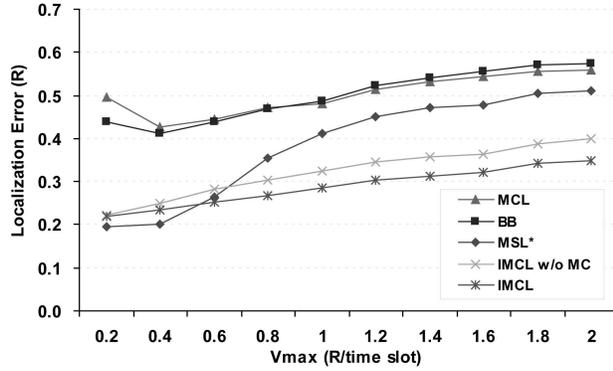


Fig. 13. Maximum moving distance (V_{max}) versus localization error.

simulate real environment, we assume that the communication region of sensors is irregular by adopting an irregular radio model. The degree of irregularity is denoted by DOI. Assume that the communication range under the different direction is randomly chosen from $[(1-DOI) \times R, R]$. For example, if $DOI = 0.5$, the communication range is randomly selected from $[0.5R, R]$. Thus, as DOI goes up, the variance of the maximum transmission range under different direction is increased.

In the DOI model, the farther anchor's position of a normal node may locate within the communication range R of the normal node. Therefore, a conflict between the near anchor constraint and farther anchor constraint may happen in the sample selection phase. For example, in Fig. 14, assume that the anchor nodes A_1, A_2 , and A_3 are within the communication range of normal node N_1 , but the packet broadcasted by the anchor node A_3 cannot be received by node N_1 due to the impact of DOI. If node N_1 receives the packet of node A_3 through near anchor A_2 , the node A_3 will become the farther anchor of node N_1 . Since the distance from any point in the overlapping area of near anchors A_1 and A_2 to the farther anchor A_3 is smaller than R , we cannot find any sample that can satisfy the near anchor and farther anchor constraints for node N_1 . Therefore, when a normal node cannot find any valid sample within a number of trials, we will ignore the farther anchor constraint.

In the simulations of various DOI, the total number of sensors is 350, $V_{max} = R$, and $Ad = 8\%$. In Fig. 15, the localization error arises for all schemes as the degree of irregularity increases. However, the localization error of

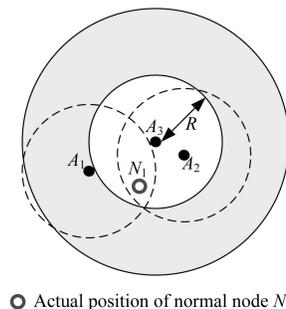


Fig. 14. The farther anchor A_3 is within the communication range R of N_1 .

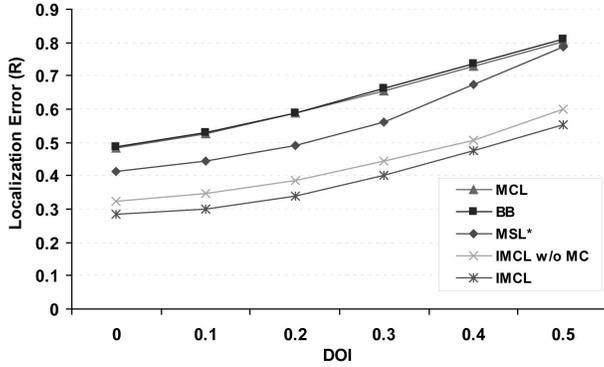


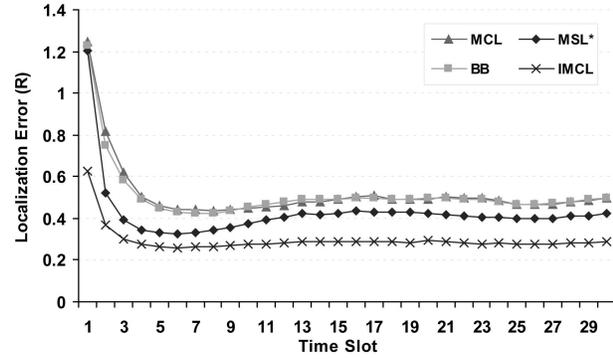
Fig. 15. Degree of irregularity (DOI) versus localization error.

IMCL is lower than those of MCL, MSL*, and BB. The improved accuracy is at least 30, 28, and 31 percent with regard to MCL, MSL*, and BB, respectively. The localization error of MSL* is more sensitive than other schemes as DOI is increasing. This is because that MSL* uses both the locations of two-hop normal nodes and anchor nodes for localization.

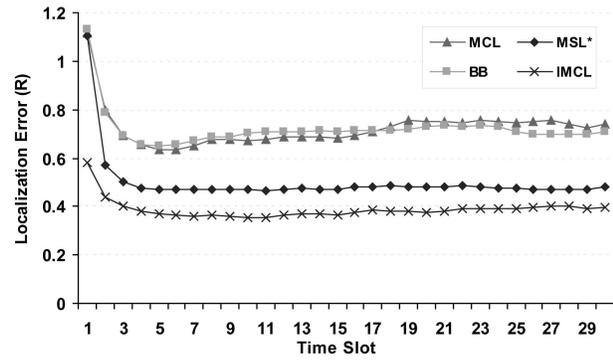
4.6 Impact of the Mobility Model

In many protocols, the mobility model plays an important role in performance results. We show the effect of different mobility models on localization accuracy. Here, we adopt three mobility models. The first one is the modified random waypoint mobility model used in previous simulations. Another one is the random direction mobility model proposed in [18]. A sensor node under this mobility model randomly selects one direction between 0 and 359 degrees. In each time slot, each sensor randomly generates a velocity and moves along the assigned direction with the given velocity. Upon reaching the boundary of the sensing field, the sensor node chooses another direction and velocity. The last model is modified from boundless simulation area mobility model [10]. The generation of velocity and moving direction in this model is the same with the random waypoint model, but the variable quantity of velocity and direction with two continued time slots will not exceed the limit of acceleration ($\Delta\alpha$) and the limit of angle variance ($\Delta\omega$). In our simulations, we set $\Delta\alpha = 0.6R$ per time slot and $\Delta\omega = 10^\circ$.

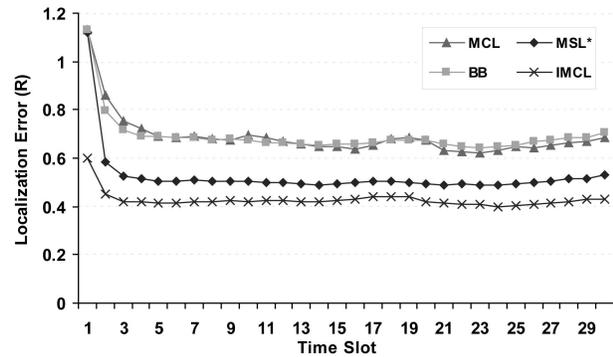
Here, the total number of sensor nodes is 350, the anchor node density is fixed by 8 percent, and the V_{max} of all nodes is set to R . In Fig. 16, the simulation results show that the mobility models will affect the localization error of sensor nodes. In the random waypoint model, most sensor nodes will aggregate in the middle area of the sensing field [17]. In the other two mobility models, all sensor nodes will uniformly distributed in the sensing field. Thus, most sensors in the random waypoint model gather more anchors' information than in the other two models, and then, the localization error in the random waypoint model is lower than in the other two. In Fig. 16, we can see that the localization error of IMCL is lower than that of MCL, MSL*, and BB for different mobility models. The variation of localization error in our IMCL scheme is also smaller than other schemes. Thus, the IMCL is robust with various mobility models.



(a)



(b)



(c)

Fig. 16. Various mobility models versus localization error. (a) The modified random waypoint mobility model. (b) The random direction mobility model. (c) The modified boundless mobility model.

4.7 Communication Cost

The number of packets sent by nodes per time slot for the four schemes is shown in Table 1. In all schemes, the anchor nodes need to broadcast their locations to one-hop and two-hop neighbors in each time slot. In addition, since the normal nodes and anchor nodes in MSL* need to flood their packets two-hop away, MSL* has the largest communication overhead among all schemes. The MCL scheme has the least communication overhead since each anchor node only forwards its actual location to two-hop neighbors. In our scheme and BB, the anchor nodes forward their packets to two-hop neighbors but the normal nodes forward their packets to one-hop neighbors only.

Furthermore, in Table 1, we also show the number of bytes transmitted in each time slot for all schemes. Each anchor packet includes IP header, sender's ID, anchor's ID,

TABLE 1
The Total Communication Cost with 350 Nodes
(322 Normal Nodes and 28 Anchor Nodes)

Scheme	MCL	MSL*	BB	IMCL
Number of packets (per time slot)	447	5579	769	769
Number of bytes (per time slot)	15192	3268568	30004	36444

anchor coordinate, and hop count. The size of an anchor packet is 34 bytes. The normal nodes of the four schemes except MCL need to forward their packets to one-hop neighbors or two-hop neighbors too. In MSL*, the packets sent by a normal node include IP header, sender's ID, estimated location, hop count, and the coordinates with weights of 50 valid samples in the last time slot. The packet size of normal nodes in MSL* is 634 bytes. In BB, the packet sent by a normal node includes IP header, sender's ID, estimated location, hop count, etc. The packet size of normal nodes in BB is 46 bytes. In IMCL, the packets sent by a normal node includes IP header, sender's ID, estimated location, hop count, eight sectors' lengths, etc. The packet size of normal nodes in IMCL is 66 bytes. Since MCL only has anchor packets, it has the minimum number of transmission bytes. Therefore, the communication overhead of our scheme is comparable to BB but much better than MSL*.

5 CONCLUSIONS

Many applications in WSNs must combine with locations of sensor nodes. In order to get location information, many localization schemes are proposed to automatically estimate sensors' positions. In the mobile sensor networks, the localization scheme becomes difficult to implement because of node mobility. Thus, developing a simple localization scheme with low estimated error is a big challenge for mobile sensor networks. In this paper, we proposed a distributed localization scheme called IMCL to improve the localization accuracy of the previous schemes. We add two more sampling constraints, the neighbor constraint and moving direction constraint, to improve the localization error of the previous work. The normal nodes need to exchange their possible located regions with each other for the neighbor constraint. To reduce the communication cost, we use a simple sectoring scheme to represent the possible located region of each normal node. To reduce the computation cost and memory occupation, the number of samples is adaptive to the estimated sampling region. Thus, the proposed scheme is suitable to be implemented on the resource-limited sensor nodes. With the simulation results, our scheme has lower localization error than the previous work in most scenarios.

ACKNOWLEDGMENTS

This work was supported by the National Science Council of Taiwan, ROC, under Grants NSC 97-2218-E-007-013 and NSC 97-2221-E-007-039-MY3.

REFERENCES

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *IEEE Comm. Magazine*, vol. 40, no. 8, pp. 102-114, Aug. 2002.
- [2] A. Baggio and K. Langendoen, "Monte-Carlo Localization for Mobile Wireless Sensor Networks," *Proc. Second Int'l Conf. Mobile Ad-Hoc and Sensor Networks*, pp. 317-328, Dec. 2006.
- [3] J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," *Proc. ACM MobiCom*, pp. 85-97, Oct. 1998.
- [4] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-Less Low Cost Outdoor Localization for Very Small Devices," *IEEE Personal Comm. Magazine*, vol. 7, no. 5, pp. 28-34, Oct. 2000.
- [5] A. Caruso, S. Chessa, S. De, and A. Urpi, "GPS Free Coordinate Assignment and Routing in Wireless Networks," *Proc. IEEE INFOCOM*, vol. 1, pp. 150-160, Mar. 2005.
- [6] S. Datta, C. Kinowski, and M.S. Khaleque, "Distributed Localization in Static and Mobile Sensor Networks," *Proc. Second Int'l Conf. Wireless and Mobile Computing, Networking and Comm.*, pp. 69-76, June 2006.
- [7] L. Doherty, K.S.J. Pister, and L. El Ghaoui, "Convex Position Estimation in Wireless Sensor Networks," *Proc. IEEE INFOCOM*, vol. 3, pp. 1655-1663, Apr. 2001.
- [8] D.K. Goldenberg, P. Bihler, M. Cao, and J. Fang, "Localization in Sparse Networks Using Sweeps," *Proc. ACM MobiCom*, pp. 110-121, Sept. 2006.
- [9] S. Guha, R. Murty, and E.G. Sirer, "Sextant: A Unified Node and Event Localization Framework Using Non-Convex Constraints," *Proc. ACM MobiHoc*, pp. 205-216, May 2005.
- [10] Z.J. Haas, "A New Routing Protocol for Reconfigurable Wireless Networks," *Proc. IEEE Int'l Conf. Universal Personal Comm.*, pp. 562-566, Oct. 1997.
- [11] T. He, C. Huang, B.M. Blum, J.A. Stankovic, and T.F. Abdelzaher, "Range-Free Localization Schemes in Large Scale Sensor Networks," *Proc. ACM MobiCom*, pp. 81-95, Sept. 2003.
- [12] L. Hu and D. Evans, "Localization for Mobile Sensor Networks," *Proc. ACM MobiCom*, pp. 45-47, Sept. 2004.
- [13] C. Liu and K. Wu, "Performance Evaluation of Range-Free Localization Methods for Wireless Sensor Networks," *Proc. IEEE Int'l Performance, Computing, and Comm. Conf. (IPCCC)*, pp. 59-66, Apr. 2005.
- [14] C. Liu, K. Wu, and T. He, "Sensor Localization with Ring Overlapping Based on Comparison of Received Signal Strength Indicator," *Proc. IEEE Int'l Conf. Mobile Ad-Hoc and Sensor Systems (MASS)*, pp. 516-518, Oct. 2004.
- [15] R. Peng and M.L. Sichitiu, "Robust, Probabilistic, Constraint-Based Localization for Wireless Sensor Networks," *Proc. Ann. IEEE Conf. Sensor and Ad Hoc Comm. and Networks (SECON)*, pp. 541-550, Sept. 2005.
- [16] N.B. Priyantha, H. Balakrishnan, E.D. Demaine, and S. Teller, "Mobile-Assisted Localization in Wireless Sensor Networks," *Proc. IEEE INFOCOM*, vol. 1, pp. 172-183, Mar. 2005.
- [17] G. Resta and P. Santi, "An Analysis of the Nodal Spatial Distributed of the Random Waypoint Mobility Model for Ad Hoc Networks," *Proc. Second ACM Int'l Workshop Principles of Mobile Computing*, pp. 44-50, Oct. 2002.
- [18] E.M. Royer, P.M. Melliar-Smith, and L.E. Moser, "An Analysis of the Optimum Node Density for Ad Hoc Mobile Networks," *Proc. IEEE Int'l Conf. Comm. (ICC)*, vol. 3, pp. 857-861, June 2001.
- [19] M. Rudafshani and S. Datta, "Localization in Wireless Sensor Networks," *Proc. Int'l Conf. Information Processing in Sensor Networks (IPSN)*, pp. 51-60, Apr. 2007.
- [20] J.-P. Sheu, P.-C. Chen, and C.-S. Hsu, "A Distributed Localization Scheme for Wireless Sensor Networks with Improved Grid-Scan and Vector-Based Refinement," *IEEE Trans. Mobile Computing*, vol. 7, no. 9, pp. 1110-1123, Sept. 2008.
- [21] J.-P. Sheu, J.-M. Li, and C.-S. Hsu, "A Distributed Location Estimating Algorithm for Wireless Sensor Networks," *Proc. IEEE Int'l Conf. Sensor Networks, Ubiquitous, and Trustworthy Computing*, vol. 1, pp. 218-225, June 2006.
- [22] S. Zhang, J. Cao, L. Chen, and D. Chen, "Locating Nodes in Mobile Sensor Networks More Accurately and Faster," *Proc. Ann. IEEE Conf. Sensor and Ad Hoc Comm. and Networks (SECON)*, pp. 37-45, June 2008.

- [23] M.L. Sichitiu and V. Ramadurai, "Localization of Wireless Sensor Networks with a Mobile Beacon," *Proc. IEEE Int'l Conf. Mobile Ad-Hoc and Sensor Systems (MASS)*, pp. 174-183, Oct. 2004.
- [24] K.-F. Ssu, C.-H. Ou, and H.C. Jiau, "Localization with Mobile Anchor Points in Wireless Sensor Networks," *IEEE Trans. Vehicular Technology*, vol. 54, no. 3, pp. 1187-1197, May 2005.
- [25] R. Stoleru, T. He, and A. Stankovic, "Range-Free Localization," *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks*, vol. 30, Springer, 2007.
- [26] V. Vivekanandan and V.W.S. Wong, "Concentric Anchor-Beacons (CAB) Localization for Wireless Sensor Networks," *Proc. IEEE Int'l Conf. Comm. (ICC)*, vol. 9, pp. 3972-3977, June 2006.



Jang-Ping Sheu received the BS degree in computer science from Tamkang University, Taiwan, ROC, in 1981, and the MS and PhD degrees in computer science from the National Tsing Hua University, Taiwan, ROC, in 1983 and 1987, respectively. He is currently a chair professor in the Department of Computer Science, National Tsing Hua University. He was the chair in the Department of Computer Science and Information Engineering, National

Central University, from 1997 to 1999. He was the director of Computer Center, National Central University, from 2003 to 2006. His current research interests include wireless communications and mobile computing. He was an associate editor of the *IEEE Transactions on Parallel and Distributed Systems*. He is an associate editor of the *International Journal of Ad Hoc and Ubiquitous Computing*, and the *International Journal of Sensor Networks*. He received the Distinguished Research Awards of the National Science Council of the Republic of China in 1993-1994, 1995-1996, and 1997-1998. He received the Distinguished Engineering Professor Award of the Chinese Institute of Engineers in 2003. He received the K.T. Li Research Breakthrough Award of the Institute of Information and Computing Machinery in 2007. He received the Y.Z. Hsu Scientific Chair Professor Award in 2009. Dr. Sheu is a fellow of the IEEE and a member of the ACM and Phi Tau Phi Society.



Wei-Kai Hu received the BS degree from the Department of Computer Science and Information Engineering, Tunghai University, Taiwan, in 2002. Since September 2003, he has been working toward the PhD degree at the Department of Computer Science and Information Engineering, National Central University, Taiwan. His current research interests include mobile computing, ad hoc networks, and wireless sensor networks.



Jen-Chiao Lin received the BS degree from the Department of Computer Science and Engineering, Yuan Ze University, Taiwan, Republic of China, in 2005, and the MS degree from the Department of Computer Science and Information Engineering, National Central University, Taiwan, Republic of China, in 2007. Currently, she is a software engineer in the Mobile Devices Department of Chi-Mei Communication Systems, Inc.

► **For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.**