A Localization Technique for Mobile Sensor Networks using Archived Anchor Information

Jiyoung Yi, Jahyoung Koo, Hojung Cha
Department of Computer Science
Yonsei University
Seoul, Korea
{jyyi, koojh, hjcha}@cs.yonsei.ac.kr

Abstract—**One interesting issue in recent sensor network research is the application of sensor nodes to mobile objects, such as people or animals, to gather non-stationary information. Apparently, locating mobile nodes is essential in many of the mobile sensor applications. Although several localization techniques have been proposed to locate mobile nodes, the mechanisms are usually based on many impractical assumptions, and mobility pattern of the mobile object is seldom considered. In this paper, we propose a practical and yet simple mobile node localization system using nearby anchor information which contains absolute time and positions. A history of anchor information is used to characterize the mobility of mobile devices. The unknown node then calculates its position with the archived anchor using a regression model. The simulation results show that the proposed algorithm outperforms previous methods under realistic conditions. We have also implemented the algorithm on real hardware, and diverse experiments were conducted to validate the feasibility of the proposed approach.**

Keywords-— location estimation; mobility; wireless sensor network

I. INTRODUCTION

Localization has been widely studied as a fundamental technique in wireless sensor networks (WSN). Some research efforts concentrate on developing highly accurate localization techniques with precise measuring methods[1][2]. Other approaches propose various techniques which approximate the position of unknown nodes using only information exchanges between the nodes and anchors[3][4]. Although many localization schemes have been developed, they are rarely applicable in real applications due to impractical assumptions. Lack of consideration of the specific localization environments is another problem in WSN. Many of the localization techniques try to solve a general problem, and fail to focus on target-specific applications. It is, therefore, crucial to understand the specific requirements of target applications when developing any realistic localization technique.

Early studies in WSN typically considered densely-deployed sensors nodes that are applied to many applications such as environment monitoring[5], object tracking[6], and so on. Although some applications are implemented on real hardware, most of them are not used in real life since the development did not seriously consider the practical requirements of underlying applications.

Recently, sensor network applications using mobile sensor devices carried by people or wild animals have increasingly drawn attention in the research community. One interesting application is a mobile heath monitoring system for patients. A patient may carry one or several devices to monitor his/her health and send the result to hospitals. A localization technique is essential in this case to track a person in an emergency. Object tracking to observe children's movement and human navigation systems are also good examples of mobile localization. In these examples, achieving a reasonable degree of localization accuracy is good enough for the applications, and yet the localization process should be as efficient as possible.

In this paper, we are mainly interested in the effective determination of the location of mobile nodes which are carried by moving objects. The mobile sensor networks consist of position-aware anchor nodes and position unknown general nodes. We focus on finding the mobility pattern of mobile nodes from the history of anchor information. The unknown nodes receive beacons from nearby anchors, find a mobility pattern, and then estimate their location in real time. Our algorithm can easily be implemented on off-the-shelf devices. The system is free from additional hardware as well as awareness of the maximum transmission range. Several technical assumptions are required for the system, but the assumptions are reasonable for practical use. Based on both simulation and real implementation, we compared the proposed system with previous techniques and validated our approach.

The rest of this paper is organized as follows. In Section Ⅱ, we describe the motivations for our research. The proposed localization technique is explained in Section Ⅲ. In Section Ⅳ, we analyze the performance of our system through both experiments and simulations. Section Ⅴ describes some of the previous localization schemes. We conclude the paper in Section Ⅵ.

978-1-4244-1777-3/08/$25.00 © IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE SECON 2008 proceedings.

II. MOTIVATION

A. *Mobility of mobile sensor devices*

Our primary focus is on localizing mobile sensor devices that are carried by mobile objects moving according to their own will. In this situation, the moving patterns of mobile nodes are not perfectly random[7]. The carriers may change their moving direction or speed, but their movements still exhibit a pattern during a certain period of time. As the period is shorter, the trace of mobility is closer to a straight line with uniform velocity. If a system is implemented in an urban environment, the moving patterns would be restricted by road layouts. Few of the previous localization methods have tried to use this mobility pattern, although an efficient localization algorithm can be developed by using the predictable pattern of mobility.

B. *GPS in real environment*

GPS[8] is commonly considered to be a good solution for outdoor localization. However, the GPS system is still costly and hence inefficient to be used for large number of devices in WSN. Most localization techniques in WSN were designed to use a small number of anchors that are equipped with GPS. Meanwhile, a great number of GPS receivers are already widely used in people's daily lives. Vehicle navigation systems using GPS are a good example. Many recent cell-phones have also been sold with a built-in GPS module. Although the GPS receiver does not have communication ability on its own, we expect that the system will soon include a low power transceiver such as the IEEE 802.15 or Bluetooth. Localization techniques should, therefore, take advantage of these widely-deployed GPS modules embedded in diverse devices.

Even if GPS receivers become cheaper and are used in every node, the nodes cannot actively use the GPS in mobile sensor networks. A GPS module typically consumes more energy than sensors and low-power transceivers. For statically deployed nodes, their positions are obtained by using the GPS once, but periodic use of GPS is inevitable for mobile sensor networks to track the mobility of the object. The Princeton Zebranet project[9] is an example of using GPS in a mobile network. Every device for Zebranet contains a GPS receiver with a relatively sufficient power supplier. However, the Zebranet tracking methods are designed to balance the energy usage of the GPS with the accuracy of the positions because the GPS unit consumes a great deal of energy. As a result, the GPS units in Zebranet are powered on for tens of seconds over several minutes. If an application requires continuous location information, the localization algorithm should deal with infrequent GPS samples or anchor beaconing.

The problem of using GPS in a real environment also exists in GPS itself. GPS in normal outdoor environments typically shows 10~20m of error unless it uses a costly mechanism such as Differential GPS[8]. If a GPS unit is adversely affected by line of sight problems or lack of energy, the estimation error would increase significantly.

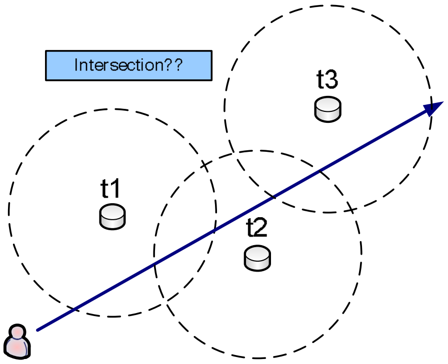


Figure 1. Problem of discrete time assumption

Deploying a large number of GPS in mobile sensor networks has both limits and possibilities. Our work targets a practical scenario where a sufficient number of anchors exists in the field, yet their accuracy is not optimal. In the worst case of anchors being deployed in one geometric location, possibly due to the unbalanced mobility of nodes, the localization technique needs to include comprehensive countermeasures.

C. *Problems of using previous mechanisms*

Localization for mobile networks is one of the key interests in recent WSN research, and many algorithms for mobile localization have been proposed in the literature. Some of the previous work, such as MCL[10] and MCB[11], could possibly be used in a real environment. From a simulation setup of a 10R1x10R WSN field, with 10 unknown sensor nodes and an anchor per transmission range, the algorithms show localization errors of approximately 0.5R, when both sensor nodes and anchors move at a speed of R meters/unit time. Unfortunately, these algorithms are based on inappropriate hypotheses. For example, both MCL and MCB require knowledge of the radio range as well as the maximum speed of nodes. In a real environment, however, the radio transmission ranges are always changing due to various factors such as the height of the node deployment, the energy residual of nodes, and deployed environments. Furthermore, the maximum velocity of each node cannot be defined in the real world since every node may have different speed limits.

One important supposition of the existing solutions is that they assume time is divided into discrete time units. Figure 1 shows the problem of discrete time unit assumption in the real world. The only way to implement the concept is to regard a time period as a certain point of time. Suppose a node moved following the arrow of Figure 1 and received three beaconing messages from anchors during a period. Since the existing solutions assume the node received three messages

---

[1] R = one hop radio transmission range

simultaneously, they would choose a position inside the intersection of three radio transmission ranges from three anchors. The assumption causes a significant difference between the estimated location and the actual location. In the worst case, as described in Figure 1, the radio ranges do not form an intersection. Any practical localization scheme should consider a continuous time system appropriately.

III. A MOBILE LOCALIZATION TECHNIQUE WITH ARCHIVED ANCHOR INFORMATION

This section details the concept of a localization technique using archived beacon messages from time-synchronized GPS-enabled anchors. In addition, the required assumptions and the design principles are discussed.

A. Design principles and assumptions

We draw on several design principles for a localization system that would be suitable for a people-centric mobile sensor network in a real environment. The following factors are considered in designing a new scheme.

- Simple and approximate solution

Due to the restricted resource of sensor nodes, the localization overhead in a sensor network should be minimized. The localization algorithms should, therefore, be simple from a computation and communication point of view. We designed an algorithm to relieve the transmission overheads of general nodes. Our estimation algorithm also has low computational overheads to approximate the positions of nodes.

- Considering continuous time

Most of the previous localization algorithms for mobile sensor networks assume that time is divided into discrete time units. The assumption makes it possible for the algorithms to receive many beacon messages simultaneously in a certain moment. In real environments, however, time passes continuously. Sensor nodes also keep moving rather than jumping to other point. If a node received three different beacon messages, the node would have received them at different positions as illustrated in Figure 1. Even though a node receives beacons at the same time, there would be a time difference between the received time and estimating time. Our localization algorithm estimates current position using the records of past anchor information and works at any time when nodes want to estimate their position.

- Free from density, radio transmission range and velocity of nodes.

To use the localization system in a real environment, impractical assumptions should be removed. Research on traditional wireless sensor network is based on the concept of densely deployed inexpensive sensor nodes in a large network. However, the node density changes unevenly for mobile situations. An ideal solution for a mobile sensor network should consider the mobility and the uneven density of nodes.

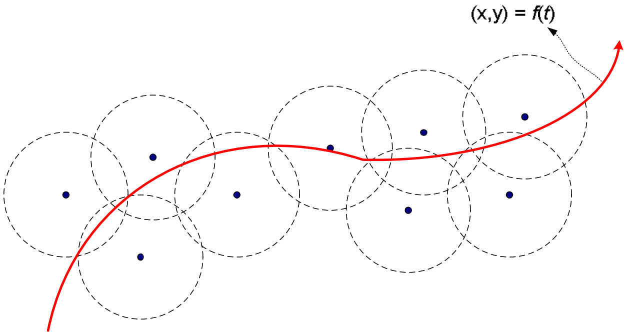


Figure 2. System overview

A priori knowledge of parameters, such as radio transmission range or node velocity, is indeed an impractical assumption. The appropriate radio range and node velocity will improve accuracy of localization methods; however, these factors change constantly in a real environment. Localization for a mobile sensor network should avoid impractical hypotheses and be based on realistic assumptions.

Our work is based on several assumptions that take account of real environments. First, an anchor should know not only its position but also the global time of the moment when the anchor obtains its location. Time synchronization is generally not an easy task to achieve in a wireless sensor network, but in our case it is obtained by using GPS for anchor nodes since GPS provides both location and global time. Only anchors need to be synchronized; hence, no extra communications are required for time synchronization.

Second, our algorithm requires a sufficient number of anchors. The algorithm is based on nearby anchor information. A large number of anchor nodes improves the accuracy of position estimates. This assumption is also true for other mobile localization algorithms, and is not inappropriate, as described in Section Ⅱ. In previous methods, if an unknown node did not receive any anchors in a certain period of time, the node failed to estimate its position or randomly chose a new position based on the previous location. In our work, as long as the history of anchors exists, the system approximates the current position. We presume that our assumptions are applicable for real environment deployment.

B. Localization Scheme

an The system consists of two components: GPS-enabled anchors and position-unaware mobile nodes. Both anchors and unknown nodes can move within the network. Anchors broadcast beaconing messages while unknown nodes receive the message to find their current position. The objective of the algorithm is to find a new position at a current time with the logged data of beacon messages. Figure 2 shows the basic idea

```
Event GPS.receive  {
    T1 := CURRENT_LOCALTIME();
    Update beacon message [Latit, Longit, time];
    b_update := TRUE;
}

Event beaconingTimer.fired  {
    if (b_update = TRUE) {
        T2 := CURRENT_LOCALTIME();
        time_diff := T2 - T1;
        Update beacon message [Latit, Longit, time, time_diff];
        Send_Beacon();
        b_update := FALSE;
    }
}
```

Figure 3. Pseudo-code of anchor node with GPS module

of the algorithm. Suppose a node went through a region and received several beaconing messages from anchors sequentially. If the node knows the sequence of the anchor's information, it can figure out an approximated trace of its user's movement. Since we assume that anchors are globally time-synchronized, unknown nodes are aware of the generated sequence of the beacon messages.

1) *Anchor node*

Anchor's duty is to inform its position to nearby unknown nodes. We assume the anchors' clocks are synchronized. The synchronized clock is not essential for our mechanism, but it removes a possible source of errors. We use GPS equipped nodes as anchors; hence, we can assume that time-synchronization among the anchors has already been done.

Figure 3 shows the pseudo-code of anchor node behavior. Anchors have two jobs to do. First, anchors need to localize themselves and synchronize their clocks together. These are done immediately by GPS. When an anchor node obtains GPS information from an attached GPS module, the node stores its local time as T1, together with location-time data [Latitude, Longitude, time]. Second, the anchor node broadcasts the GPS information. Since receiving the data from the GPS module and broadcasting it do not occur simultaneously, the time difference between receiving and broadcasting GPS data should be calculated. We store another local time, T2, and calculate the time difference between T1 and T2, right before broadcasting beacon message. As the time difference is attached to a beacon message, the anchor broadcasts the message to the network.

2) *Unknown node*

Unknown nodes perform complex operations. Once a node receives a beaconing message from an anchor, it stores the local time as *time_recv*. After storing the received local time, the node constructs or updates the anchor history table. We denote the anchor history table as AHT. AHT contains all the data from beacon messages and their *time_recv* value. When the node writes a new beacon message to AHT, insertion sorting is done for the beacon message by its global time. If the AHT is full of old information, the node discards the oldest data and stores the new message.

After updating AHT, the data in AHT is normalized. Since the location-time information from GPS contains a large amount of data, the normalization process helps to reduce the complexity of ensuing operations. Both Latitude and Longitude have degrees, minutes and seconds. To compare the data with a unified denomination, the data are expressed as seconds. If Latitude and Longitude are converted to seconds, the values are unnecessarily large. Hence, in our implementation, we subtract the smallest Latitude and Longitude from each entity in AHT. Global time is also normalized in the same manner. This step is not essential for position estimation, but prevents the complex calculation of huge numbers.

Nodes try to find the relation between location and time at regression phase. We assume that a user's movement forms a specific pattern such as a straight line or complicated curve. The unknown mobile nodes are programmed to use a specific regression model for archived anchor information. The node finds the coefficients of the regression model that fits with the history of anchor positions. The node obtains two functions that represent latitude and longitude with global time. Many different regression models can be applied, but we used a simple linear regression model, Y = a + bX, on implementation to simplify the calculation. The coefficients of simple regression models are easily arranged with the least square method[12]. The linear regression model and its coefficients are represented as follows:

$$L = a + bT \tag{1}$$

$$b = \frac{n\sum T_i L_i - (\sum T_i)(\sum L_i)}{n\sum T_i^2 - (\sum T_i)^2} \tag{2}$$

$$a = (\sum L_i - b\sum T_i)/n \tag{3}$$

(where

Li – Latitude or Longitude in AHT

Ti – Global time of each location entity in AHT

n – number of entities used for calculation)

Unknown nodes periodically compute their positions at current time. Since the nodes have functions of latitude and longitude related to global time, the current position is calculated with current global time. Current global time is obtained with the global time, *time_diff*, *time_recv* of the latest logged data, and current local time. Equation (4) shows the calculation of the normalized value of current global time. CGT and CLT represent current global time and current local time, respectively. Regardless of whether the node receives new GPS information, a new position is calculated periodically depending on current global time.

$$CGT_{norm} = T_{n\_norm} + time\_diff_n + (CLT - time\_recv_n) \quad (4)$$

Suppose a situation occurs in which an unknown node has moved far from the anchors and is no longer receiving anchors. If the node continues to move with the same direction and speed as before, the estimation result will follow the actual trajectory of the node. However, if the node changes its moving direction or speed, a location error would grow until the node received new anchors. To prevent this situation, unknown nodes have a threshold and check the time difference between the most recent beacon and current time. If the duration of the node's beaconless condition is longer than this threshold, the node transmits a message to local neighbors to share information. The local neighbors return their current anchor information, and the node regards the center of the returned information as its new position. In this case, the node removes old anchor archives that are not acceptable for characterizing mobility. When the node receives a new beacon message, it operates the regression phase using the new anchor information with recent archives.

C. *Source of Errors*

There are possible sources of errors in the proposed localization algorithm.

- *Distance between anchors and nodes* – The location difference between anchors and a node is the fundamental source of error. Conceptually, unknown nodes regard neighbor anchors' location as their location at certain moments. The error could grow up to 1R in the worst case.
- *Error in anchor's information* – GPS error also causes localization error. A regression phase with archived data from several different anchors may lessen these errors, since the errors and location differences of different anchors cancel out the others.
- *Incorrect prediction* – This error arises when the movement of an unknown node is not fit to a fixed regression model. Even though the restricted movement of nodes is assumed, the nodes can change moving direction and speeds any time. When a node changes its velocity and does not receive new anchors, the estimation error becomes serious.

IV. EVALUATION

We evaluated our approach using both simulation and real implementation. The simulation results show the performance characteristics of our method in mobile sensor networks. The experiments based on actual system implementation validate the feasibility of the proposed system.

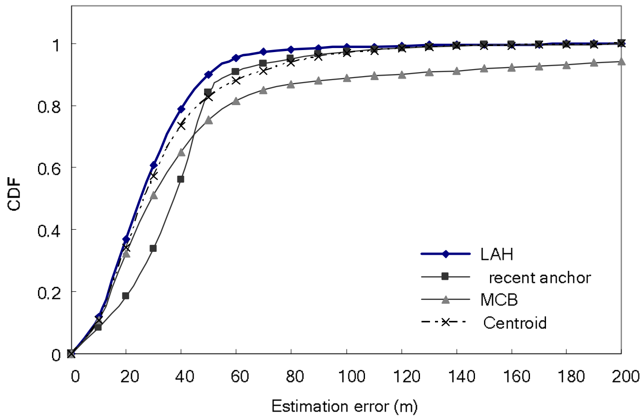


Figure 4. Cumulative distribution function of different localization methods,

A. *Simulation*

1) *Simulation parameters*

The performance characteristics of our work were analyzed through simulation. We observed the characteristics of the proposed localization algorithm in various conditions. In our simulation, we varied the parameters of the sensor networks and the localization algorithm. For all experiments, 400 sensor nodes were randomly distributed in a 500m x 500m region. We assumed 50m of fixed transmission range for all nodes. All localization schemes evaluated in the simulation estimated the nodes' position within at every 5 seconds. We varied the parameters, such as the maximum speed of nodes v_{max}, the number of anchors *na*, the beaconing period *bt*, error of anchor e_a, and the size of the AHT N_h in our algorithm.

We adopted the random waypoint model [13] for mobility of both unknown nodes and anchors. A node randomly chooses its destination, speed of movement, and pause time after arriving at the destination. In our simulation, the nodes randomly chose their speed, which was less than v_{max}. Pause time was also chosen randomly from 0 to 100 seconds.

Our localization scheme was compared with three other methods: MCB, Centroid and *recent anchor*. *Recent anchor* is a simple way to localize a node. Nodes regard the most recent GPS information as their location. MCB is an extended version of MCL, which is a representative localization scheme for mobile sensor networks. Centroid takes averages of anchors' information as a node's location. Experiment results are obtained with an average of 30 executions with different random seeds.

2) *Overall Accuracy*

Figure 4 shows the cumulative distribution function of the estimation error of different algorithms. 40 anchors over 400 unknown nodes, 10 seconds of beaconing period, 1.5m/s of speed, and 10 anchor data were used for the simulation. We chose the number of anchors and speed of nodes by

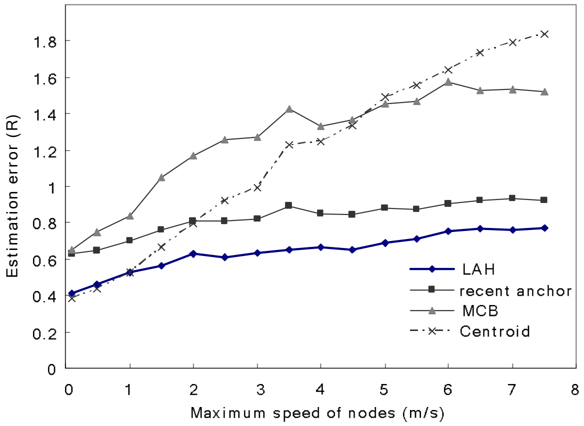


Figure 5. Impact of nodes speed, $na = 40, bt = 10sec, N_h=10, e_a=0$

considering a realistic environment. In this context, 1.5m/s of speed as human mobility and 40 nodes capable of beaconing are a reasonable set up. Our algorithm is represented as LAH (Localization with Anchor History). In our algorithm, approximately 90% of the nodes have less then 50m of estimation error with the parameters. Even if LAH outperforms other methods in this experiment, the difference seems small. The specific characteristics and advantages of LAH will be presented with other simulation results in diverse network conditions.

3) *Speed of node*

A localization technique for mobile networks should work with diverse node speeds. Figure 5 illustrates the average estimation error of each localization method when the speed of nodes changes. The accuracy of both LAH and *recent anchor* are relatively stable. On the other hand, the accuracy of Centroid linearly decreases as the speed of node increases. Even if Centroid shows the best performance when the nodes are not moving, the experimental result shows that the algorithm does not fit with mobile networks.

MCB (or MCL) is shown as stable for the change of node speed. However, the result draws a different result. The disparity comes from the difference in simulation parameters. Compared to the previous research, our experiment uses less number of anchors and longer beaconing period. The effect of anchor density is described in next.

4) *Anchor density*

The accuracy of localization is directly related to the number of anchors. Although our algorithm needs sufficient number of anchors, the algorithm should work reliably with adverse deployment conditions. Figure 6 shows the influence of the number of anchors on localization error. In the case of MCB, estimation errors include un-localized nodes that chose random positions as their location. When unknown nodes barely met anchors, MCB failed to localize the nodes unless it increased the estimation period. Hence, MCB has the largest error among the localization techniques with a small number of anchors. The other mechanisms show similar performance, but LAH has the least on average. Even if the node does not receive any beacon messages, LAH predicts its current location based on old anchor information. An incorrect prediction may have occurred during the simulation, but the result shows that the average location error is reasonable.

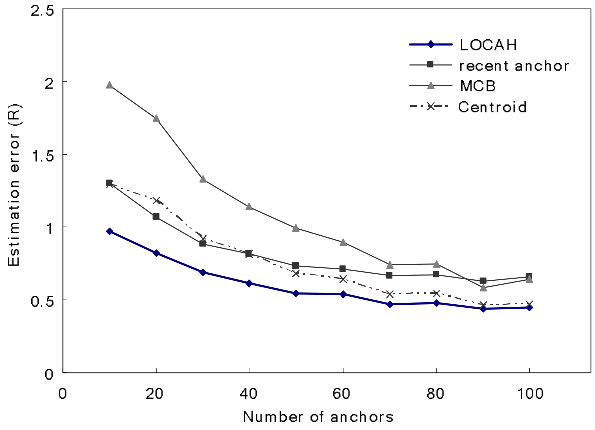


Figure 6. Impact of number of anchors, $v_{max} = 2m/s, bt = 10sec, N_h=10, e_a=0$

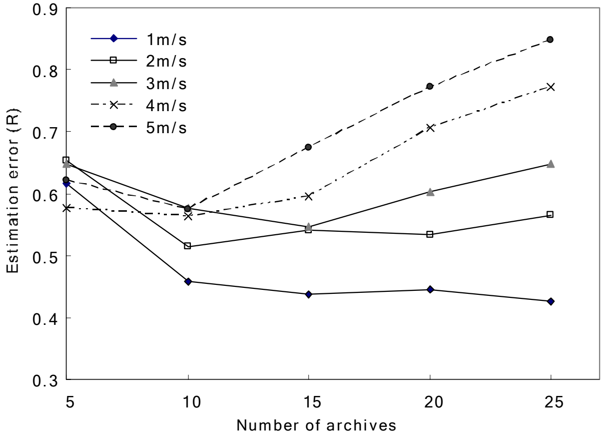


Figure 7. Impact of number of archives, $bt = 10sec, N_h=10, e_a=0$

5) *Size of anchor history table*

The parameter that affects accuracy in our method is the size of the anchor archives used for the regression algorithm. We varied the size of AHT with various speeds. A small number of archives adjusts the estimation quickly when nodes change mobility. A large number of archives refines the information from many different anchors. As illustrated in Figure 7, using 10 archives shows the best result in every case on average. In actual implementation, we used a maximum of 10 anchor archives for the regression algorithm.

B. *Implementation*

1) *Experiment Configuration*

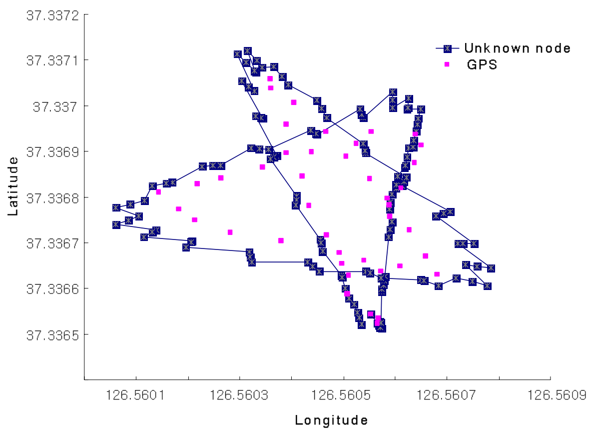


Figure 8 Localization result of star-shaped movement

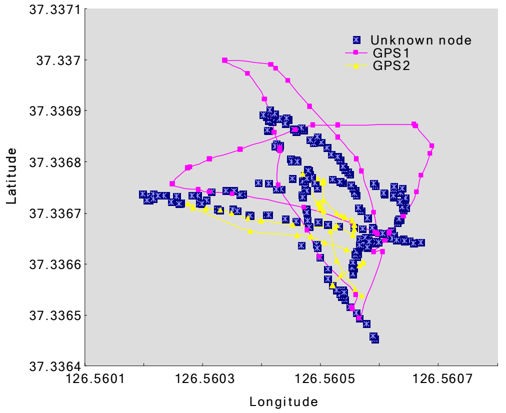
Further evaluation was conducted by implementing the proposed system on real hardware. In order to validate our algorithm, we used several sensor nodes, which were Crossbow's MicaZ[14] and Moteiv's Tmote Sky[15]. Each anchor node was implemented on MicaZ connected to a GPS module. General mobile nodes were implemented on Tmote Sky, which was connected to a host PC. The PC was used to store estimation results and receive beacon information. Every calculation required for our algorithm was operated in individual motes.

For the experiments, anchors broadcast beacon messages with a period of approximately eleven seconds. Mobile nodes store ten anchor archives for location estimation. At most, seven anchors were deployed in the field. Encountering more anchors improves accuracy but localization succeeds only with occasional beacon messages.

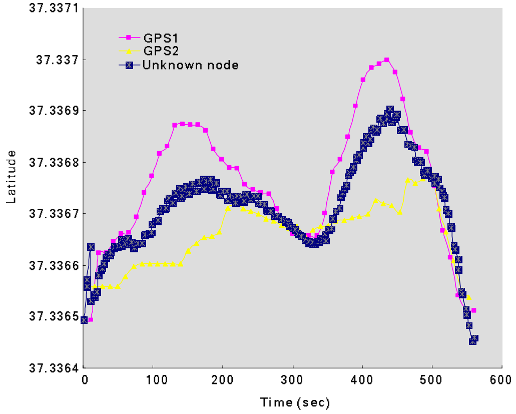
We have evaluated our system at the athletic field and the campus road of Yonsei University. Location estimation was operated every five seconds and also moments when the nodes received beacons. The calculated locations are expressed with latitude and longitude in degrees and decimal minutes. We tested various shapes of trajectory, spanning the whole area. One or two anchors as well as one general node were moved together.

2) *Variety of Speed and Direction*

The major cause of estimation error in our algorithm is inaccurate prediction, which occurs through the sudden change of moving speed and direction. We tested the feasibility of our algorithm by observing errors with varying speed and direction. Figure 8 shows the GPS information from an anchor and the estimation result of an unknown node. During the experiment, an anchor is followed by the unknown node. The nodes moved at walking speed and moved in a star-shaped pattern. Estimation is accurate when the node moves straight, but error occurs when the node changes direction. The actual distance of maximum estimation error was approximately 25m. In this case, the unknown node received beacon messages once in 10 seconds. If the node received more information from many anchors, the error would decrease. The error would grow if the node moved faster; however, the estimation would be corrected as soon as it received a new beacon message.



(a) star-shaped movement



(b) Time vs Latitude

Figure 9 Localization result with poor GPS

3) *Effect of anchor's error*

In a practical environment, a perfect anchor does not exist unless the stationary anchors are purposely well deployed. Since GPS is regarded as a good component to act as anchors, we conducted the experiments on a cloudy day when GPS shows poor results. Two anchors equipped with GPS and one unknown node moved in a star-shape. Figure 9 shows the experimental results with poor anchors. As illustrated in Figure 9(a), two GPS receivers produced different results even though they moved together. In particular, one of the GPS generated at most 50m of error. The estimation result cannot be free from the error of anchors. However, our localization algorithm finds the best trajectory using the results of two GPS receivers.

4) *Overall accuracy in an open field*

We evaluated our algorithm in a more general environment. The deployment area was approximately 100 × 100 m. Three anchors, or GPS receivers, were statically deployed, and three mobile anchors and an unknown node moved around the field freely. We divided the result into four regions, as shown in Figure 10. At region 1, the unknown node made a left turn. Even though the node received fluctuated beacon messages

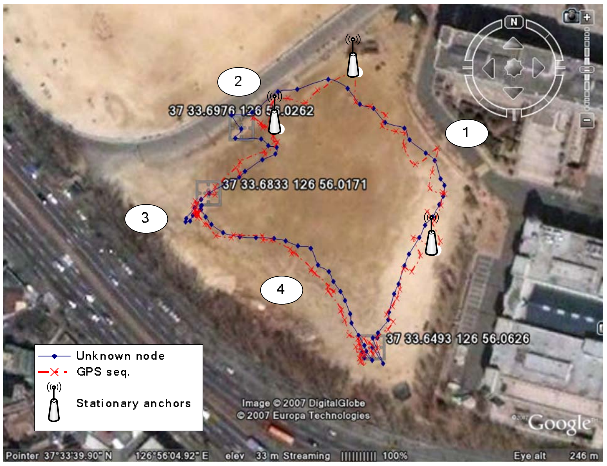


Figure 10 Experiment on open field

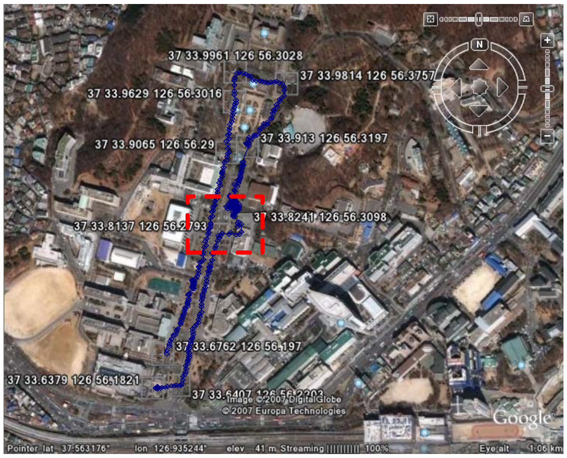
from different anchors, the estimation result formed a smooth movement. The unknown node changed direction roughly in region 2 and temporarily ceased to move at region 3. The sudden change of movement generated some degree of localization error, but as soon as the node received a new beacon message, the estimation reflected the movement changes. The estimation result followed the gentle curve in region 4 well.

5) *Overall accuracy in an urban road*

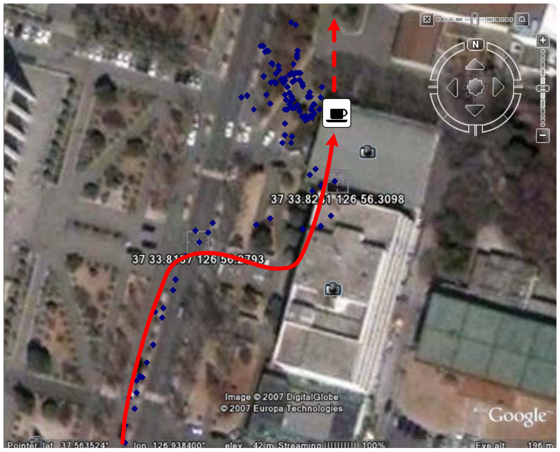
Since our localization algorithm is especially designed for a moving person, we examined our algorithm with the usual moving path of a person in real life. The system was tested on a campus road. Although we assumed that a sufficient number of anchors is deployed in a large field, we cannot construct the same environment with several anchors. Instead, we used six anchors that moved dynamically around the object. The unknown node received an average of 5 beacon messages in a minute. However, the node did not receive any beacon messages in some sections. The experiment took 30 minutes. The total length of the unknown node's trajectory was approximately 1.5km. To imitate the normal activity of a person, the experimenter moved into a building lobby and took a break for several minutes.

Figure 11 (a) shows the total view of the experiment result. The actual trajectory was very close to the estimated locations. Figure 11 (b) illustrates a specific section of the experiment. As the trajectory of the node shows, the experimenter walked across a street, moved into a building and took a rest in a cafeteria. Some anchors, which entered the building as general node, transmitted beacon messages that included a non-trivial amount of error. After a while, some of the anchors failed to find their positions. Useful beacons came from anchors passing by the outside of the building. Hence, the estimated positions during the break stayed on the road. The estimation error was at most 30m.

The experimental results show that our algorithm has tens of meters of error in realistic environments. As Figure 11 (a) shows, the errors are not prominent in road level localization.



(a) Entire view of experiment on campus roads



(b) A part of result on campus road (taking break)

Figure 11. Experiments on Urban Road

The primary goal of localization for mobile objects in an outdoor environment is to answer the question 'Where is he/she?'. We believe our system provides adequate answers for the question.

V. RELATED WORK

Many localization techniques for wireless sensor nodes are found in the literature. The well-known range-based methods typically use received radio signal strength [16], time difference of arrival of different signals [1] or angle of arrival [17] as a ranging measure. On the other hand, various localization schemes have been designed to localize sensor nodes without ranging for outdoor environments. Many such techniques assume that few nodes are aware of their global positions, and information from these nodes is used to estimate the unknown node's location. Centroid [3] by Nirupama et al. is a simple localization scheme without ranging. Each node in the system calculates the center of the locations of all anchor nodes that the unknown node hears. Niculescu et al. [4] proposed ad-hoc positioning systems (APS). APS includes three different propagation methods: DV-hop, DV-distance, and Euclidean. These methods obtain relatively accurate results in multi-hop networks. Among the methods, DV-hop converts hop count to the distance between an unknown node and an anchor node. Lim et al. [18] proposed the Proximity Distance Map (PDM), which calculates the particular transformation matrix to convert hop count to distance. The matrix characterizes anisotropic network topologies and proximities to the anchor nodes in all directions. Shang et al. [19], [20] proposed the multi-dimensional-scaling-based localization methods MDS-map and MDS-map(P). MDS-map, the first model of their scheme, gathers the connectivity information of the sensor nodes and builds a relative map of the sensor field by using multi-dimensional scaling. MDS-map(P) supports the distributed-fashion of MDS-map.

Due to the recent interest in mobile sensor networks, the localization issue is further in the center of research. A commonly known localization algorithm for mobile sensor networks is MCL (Monte Carlo Localization)[10]. MCL periodically updates its *samples*, which are a node's probabilistic distribution of locations, by repeating the *prediction and filtering* process. When the nodes obtain a sufficient number of position *samples*, the locations are estimated by calculating the center of the sample positions. A range-based version of MCL has also been proposed [21]. This version provides a sampling and filtering method based on range measurements, and weighs each valid sample to obtain accurate estimation results. Baggio et al. [11] proposed an enhanced version of MCL. By reducing the sampling area, MCB draws good samples, and thereby the computation overhead is reduced; however, the algorithm still depends on specific parameters such as a fixed radio transmission range. Evidently, their inappropriate assumptions make these methods unsuitable for many scenarios in real environments.

VI. CONCLUSION

We proposed a system for mobile node localization suitable for real environments when both anchors and unknown nodes are moving. We focused on using the past information of anchors to figure out the current position of nodes. The archived anchor information models the user's movement and the movement models are used for discovering new positions. The system does not require any infrastructure or impractical assumptions. Our algorithm is simple enough to run on a machine with low computational ability and limited resources. Simulations have been done in diverse network conditions, and the result shows that our approach outperforms other methods. To validate the feasibility of our approach, a real system was implemented and experiments were executed on a large field. Our system was tested in various situations and the experimental results show reasonable accuracy which is adaptable for a general environment with appropriate settings. In our further efforts, we will develop a localization device which can check the trajectory on geographic map software in real-time.

ACKNOWLEDGMENT

This research was supported by the National Research Laboratory (NRL) program of the Korean Science and Engineering Foundation (No.M10500000059- 6J0000- 05910) and the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2007-C1090-0701-0015).

REFERENCES

[1] N. Pryiantha, A. Chakaborty, and H. Balakrishnan, "The Cricket location-support system," in *Proc. MobiCom*, Boston, 2000, pp. 32–43.

[2] A. Harter, A. Hopper, P. Steggles, A. Ward and P. Webster "The anatomy of a Context-Aware Application", In Wireless Networks, Vol. 8, pp. 187-197, February 2002

[3] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," IEEE Personal Communications Magazine, Vol. 7, No. 5, pp. 28-34. October, 2000.

[4] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," in *Proc. Globecom*, San Antonio, 2001, pp. 2926–2931.

[5] G. Werner-Allen, K. Lorincz, J. Johnson, J. Lee and M. Welsh, "Fidelity and Yield in a Volcano Monitoring Sensor Network," in *Proc. OSDI*, Seattle, 2006.

[6] J. Aslam, Z. Butler, F. Constantin, V. Crespi, G. Cybenko, and D. Rus,"Tracking a Moving Object with a Binary Sensor Network," in *Proc. SenSys*, Los Angeles, 2003.

[7] M. Kim, D. Kotz and S. Kim, "Extracting a mobility model from user traces," in *Proc. INFOCOM*, Barcelona, 2006.

[8] B. Hoffman-Wellenhof, H. Lichteneeger, and J. Collins, Global Positioning System: Theory and Practice (4th ed.). New York: Springer-Verlag, 1997.

[9] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet," in *Proc. ASPLOS-X*, San Jose, 2002, pp. 96-107.

[10] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proc. Mobicom*, Philadelphia, 2004, pp. 45–57.

[11] A. Baggio and K. Langendoen, "Monte-Carlo Localization for mobile wireless sensor networks," in *Proc. MSN*, Hong Kong, 2006.

[12] J. Lawson and J. Erjavec, Modern Statistics for Engineering and Quality Improvement (1st ed.). Duxbury Press, 2000

[13] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research,". In Wireless Communications & Mobile Computing. WCMC, Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, 2002.

[14] MicaZ. Available : http://www.xbow.com

[15] Tmote Sky. Available: http://www.moteiv.com.

[16] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. INFOCOM*, Tel Aviv, Israel, 2000, pp. 775–784.

[17] D. Niculescu and B. Nath, "Ad-hoc positioning system (APS) using AOA," in *Proc. INFOCOM*, San Francisco, 2003, pp. 1734-1743.

[18] H. Lim and J. C. Hou, "Localization for anisotropic sensor networks," in *Proc. INFOCOM*, Miami, 2005, pp. 138–149.

[19] Y. Shang, W. Ruml, and Y. Zhang, "Localization from mere connectivity," in *Proc. MobiHoc*, Annapolis, 2003, pp. 201–212.

[20] Y. Shang and W. Ruml, "Improved MDS-based localization," in *Proc. INFOCOM*, Hong Kong, 2004, pp. 2640–2651.

[21] B. Dil, S. Dulman, and P. Havinga, "Range-based localization in mobile sensor networks," in Proc. *EWSN*, Zurich, 2006, Volume 3868