This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2009 proceedings.

Centralized Mobile SensorGroup for Exploring Border of Target Area

Kuen-Liang Sue and Jing-Wei Lin
Department of Information Management, National Central University
Jhongli City, Taoyuan, Taiwan
Email:klsue@mgt.ncu.edu.tw

***Abstract*—A low-cost mobile sensor network structure with a sensing mechanism is presented to explore the border of unknown target areas effectively. At first, mobile sensors with GPS receivers are organized into groups called SensorGroup (SG) which works well for the detecting jobs. After that, an advanced structure called Centralized SensorGroup (CSG) is proposed to further reduce the device cost since only the center node of the group is equipped with a GPS receiver. To detect the target area, an S-shaped sensing mechanism is also presented to cooperate with SG and CSG. The operation processes and detection phases are provided and verified. The detection performance is evaluated by detection completeness and accuracy in different scenarios in a square area of 100m*100m sensing environment. The simulation results show that CSG has good detection accuracy and the same performance as that of SG.**

Keywords- mobile sensor network; GPS; localization

I. INTRODUCTION

Traditional sensor networks detect the environment by randomly deploying immobile sensors within the area. It is necessary to increase the number of immobile sensors for improving the accuracy of the detection. The mobility and feasibility of mobile sensors (MSs) makes it possible to detect the area even though the environment is changing dynamically. Mobile sensors formed through various wireless interfaces can move to collect information in the environment under their own control or under the control of the environment [1][2]. The location information of sensing data provides usages in various applications [3][4]. One of the applications is to detect the specific areas such as the oil spoiled area in the ocean.

In mobile sensor networks, each MS moves and operates according to its current location. So a wireless localization system is essential for the construction of mobile sensor network. A satellite localization system (GPS) is able to locate target objects by using the signal from satellites surrounding the earth [5]. GPS calculates the locations of objects by triangulation. Since GPS receivers are miniaturized nowadays, sensors can get their location information by equipping GPS receivers [6]. Because there can be large numbers of sensors deployed randomly, it is not cost-effective to equip GPS receivers on all sensors. Anchor nodes were added to reduce the cost and to support the localization. The locations of anchor nodes are known prior and can be obtained by equipping wireless receivers. After anchor nodes are set, these anchors can act just like the satellites in GPS, each sensor can calculates its location with triangulation by receiving signals from at least three anchors and by measuring the distance to each anchor through the distance measuring technologies such as the time of arrival (TOA) and the received signal strength indication (RSSI) [7]. The anchor-based localization system without the satellites can reduce the cost. However, if the size of sensor network is enlarged, more cost and complicated considerations are required to set up the anchor nodes. RSSI using mobile-anchor (RSSIUMA) localization system was proposed [8]. A mobile-anchor moves and broadcasts its location periodically. Each sensor measures the distance to the mobile-anchor through RSSI when it received signals from the mobile-anchor. Then a circle was calculated through the mobile-anchor coordinate and the distance measured by RSSI. Each sensor can get a number of circles after receiving signals continuously. Finally, its possible location can be found based on the intersection area of all circles.

Because the detection area changes dynamically, it is inappropriate to set fixed-anchor nodes in mobile sensor networks. Therefore, a novel network structure with a localization algorithm is constructed based on the mobile-anchor localization system. Also, a sensing mechanism helps to detect the target area. Typically, the cost of MSs is higher than that of immobile sensors due to their mobility characteristics. Hence, how to decrease the number of MSs deployed in mobile sensor networks becomes an important issue.

II. SENSOR GROUP NETWORK STRUCTURE

SensorGroup (SG) is defined as a small mobile sensor network constructed by a number of MS nodes grouped within a specific range. It is assume that all MSs in SG are equipped with devices which can get global coordinates such as GPS receivers. If one of MSs in SG detects the border of the target area while moving in a sensing environment, its current location is calculated and marked as a sensing record. The movement of MSs is controlled by the reference point group mobility (RPGM) [9]. As shown in Fig. 1, a center node (C node) is an MS allocated in the center of the sensor group. It is responsible for the controlling of all operations and the broadcasting of its location to other member nodes (M nodes) within its transmission range (TR) periodically. M nodes which move randomly within the group range (GR) are deployed around C node.

The research was supported by the National Science Council, Taiwan, under the contract NSC96-2416-H-008-021.

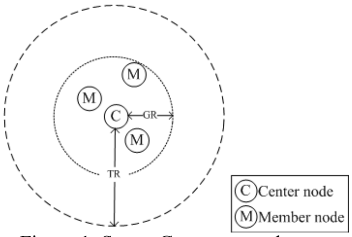


Figure 1. SensorGroup network structure

SensorGroup Operation Process
1. Set initial location
2. C moves with speed V_c
3. **Iterate** every T period
4. $L_c \leftarrow$ GPS location of C
5. C RPGMbroadcast(L_c)
6. **for** each M do
7. Receives L_c form C
8. $L_{m\_init} \leftarrow$ GPS location of M
9. $L_{m\_dest} \leftarrow$ Random destination within GR
10. M moves from L_{m_init} to L_{m_dest} with speed V_m
11. **end for**

Figure 2. Operation process of SG

The operation process of SG is shown in Fig. 2. At first, the initial location of SG is set (step 1). C node starts moving with the speed V_c and leads SG to detect the possible targets (step 2). The operation processes from step 3 to step 12 which called an "RPGM round" are iterated with a period of T seconds. While getting the GPS location L_c, C node broadcasts L_c with an *RPGMbroadcast* packet to all M nodes (steps 4-5). M nodes receive the *RPGMbroadcast* signal from C node to end the previous (n-1)th RPGM round and to start the current nth RPGM round (step 7). After receiving the packet, M node can get the current location as its initial location L_{m_init} (step 8). Each M node decides a random destination L_{m_dest} within GR around L_c (step 9). When L_{m_init} and L_{m_dest} are both decided, M node calculates a moving vector based on these two coordinates. Then M node moves along this vector with the speed V_m (step 10). During each RPGM round, the sensing records should be stored if the border is detected.

TR, GR, V_c, and V_m are set according to the node communication and the group movement. C node broadcasts its location to all M nodes during all RPGM rounds. The distance between C node and M node is calculated as (V_cT+GR).TR is smaller than TR_{max} which is the longest distance to receive broadcast for M nodes. To let M nodes keep moving around C node, V_c should be slower than V_m .To keep all M nodes moving within the range of SG, the distance that an M node can move in every RPGM round is V_mT. According to these settings, the faster SG can move, the smaller GR should be set.

III. PROPOSED CENTRALIZED SENSORGROUP NETWORK

In SG network structure, all MSs are assumed to be equipped with GPS devices which demand high construction fee. A novel Centralized SensorGroup (CSG) network structure is proposed to reduce the cost. The network structure and parameter settings of CSG are the same as those of SG. C node acts as a mobile anchor. M nodes are equipped only with a low-cost short-distance transceiver to measure its movement. The difference between SG and CSG is that M nodes can calculate the locations using a modified localization algorithm from [10].

A. Localization Algorithm

The communication between each node is shown in Fig. 3. The algorithm is separated into the "initial phase" and the "verification phase". In the initial phase, C node broadcasts its location with packets Loc1 and Loc2 at time t1 and t2, respectively. In the verification phase, each M node exchanges the candidate locations with each other by broadcasting a *VeriInfo* packet. When M node receives *VeriInfo*, it measures the distance to the packet sender and verifies the candidate locations. M node calculates a weight for the verified answer.

During the initial phase, C node adds its location (x_0, y_0) into Loc1 and broadcasts to other M nodes at t1 as shown in Fig. 4 (a). The distance d_1 between C node and M node at (x_2, y_2) is measured by TOA when M node receives Loc1. Then, C node and M node moves to (x_1, y_1) and (x_3, y_3), respectively. At t2, C node broadcast its location (x_1, y_1) with Loc2. M node measures d_2 when it receives Loc2. M node can decide two circles, C1 and C2, by using (x_0, y_0), (x_1, y_1), d_1, and d_2. C1 and C2 define the sets of possible locations of M node at t1. Because M node can calculates it displacement from t1 to t2, it can calculates a new circle C1' by displacing C1 with the vector ($v_2\cos\beta$, $v_2\sin\beta$). The center point of the new circle becomes ($x_0+v_2\cos\beta$, $y_0+v_2\sin\beta$) as shown in Fig. 4(b). M node can get its candidate locations at t2 by calculating the intersection of C1' and C2. The possible locations are depicted as (x_3', y_3') and (x_3'', y_3'') in Fig. 4(b).

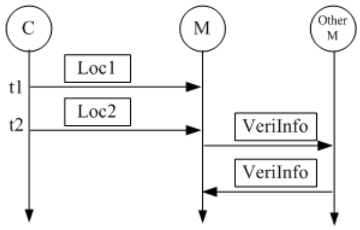
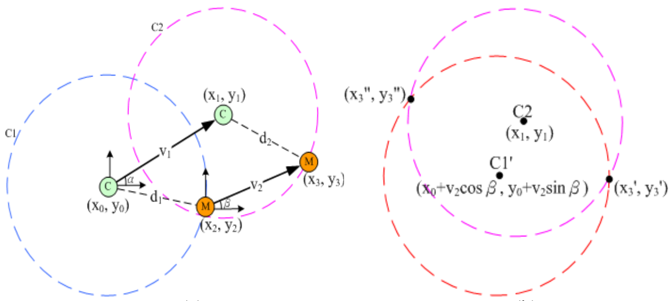
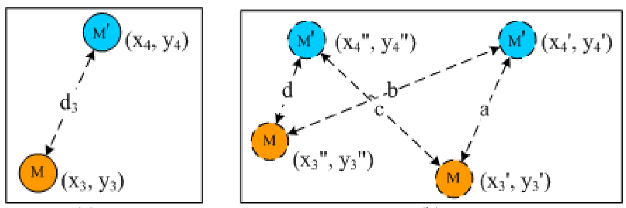


Figure 3. Node communication in the localization algorithm for CSG.



(a) (b)

Figure 4. Initial phase (a) C and M nodes, (b) Candidate locations of M



(a) (b)

Figure 5. Verification phase (a) Two M nodes, (b) Four candidate locations

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2009 proceedings.

M node measures d_3 when M node receives *VeriInfo* from another node M' as shown in Fig. 5 (a). M node can compose at most four sets of candidate locations using the locations of M and M' in Fig. 5 (b). The candidate locations (a, b, c, and d) are compared with d_3. “a” has the smallest distance with d_3. (x_3', y_3') is chosen as the verified answer of *VeriInfo*. (x_3', y_3') is given a weighted value by calculating $1/|d_3–a|$. A weighted value is given whenever M node receives *VeriInfo* from other M nodes. The weighted values for each candidate location are accumulated. At the end of an RPGM round, M node can get the final t2 localization result by calculating the weighted mean of its candidate locations. t1 localization result is given using its displacement vector from t1 to t2 in the initial phase.

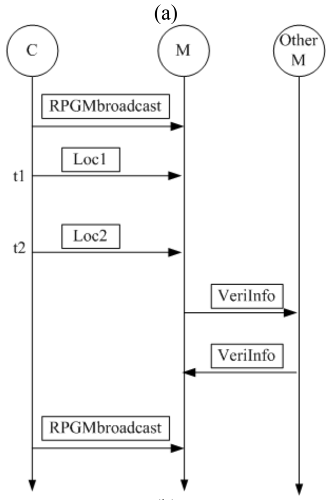
At the start of the next RPGM round, M node calculates the initial location of the next RPGM round by adding localization result at t2 with the displacement from t2 to the end of this RPGM round. If the distance and the direction are measured inaccurately, the initial phase calculation of the localization algorithm would fail. The intersection of C1' and C2 can not be calculated. To keep the localization algorithm working, when M node detects the localization failure, it sets the initial location of the current RPGM round as its t1 localization result and calculates localization result at t2 by adding the initial location in current round.

B. Operation Process

Centralized SensorGroup Operation Process

1. Set initial location of CSG
2. C moves with speed V_c
3. **iterate** every T period
4. $L_c \leftarrow$ GPS location of C
5. C RPGMbroadcast (L_c)
6. **for** each M do
7. Receive L_c form C
8. Location determination
9. Record conversion
10. $L_{m\_init} \leftarrow$ initial location
11. $L_{m\_dest} \leftarrow$ Random destination
12. M moves with speed V_m
13. **end for**
14. Localization process
15. **end iterate**

(a)



(b)

Figure 6. (a) Operation process and (b) Communication in CSG

The operation process and the communication of CSG are shown in Fig.6. The operation process of steps 1-5 are the same as those of SG. When M node receives *RPGMbroadcast*, it ends the (n-1)th RPGM round including the localization determination (step8) and the conversion of the temporary sensing records in (n-1)th RPGM round (step9). Because M nodes can not get their current locations instantly, all sensing record in (n-1)th RPGM round are temporarily stored as the displacement vectors from t1 to the time when they detect the border of the target. After the localization result at t1 is determined in step 8, M nodes convert the temporary sensing records in (n-1)th RPGM round to the actual sensing record by adding t1 localization result to all temporary records. M node then starts the process of nth RPGM round. M node calculates the initial location L_{m_init} in nth RPGM round by adding t2 localization result of (n-1)th RPGM round with the displacement form t2 to the time when it receives *RPGMbroadcast* of nth RPGM round (step 10). A random destination L_{m_dest} is then decided (step 11). M node moves with speed V_m (step 12). The localization process of nth RPGM round is executed (step 14). After T seconds, C node rebroadcasts *RPGMbroadcast* to start (n+1)th RPGM round.

C. Fall Out Problem

Due to the localization error, M node may move away from C node. M node will fall out of TR and depart from network eventually. M node detects the fallen out problem if it does not receive *RPGMbroadcast* within T seconds. M node will try moving back to the network. To provide M node sufficient information to execute the moving back, *RPGMbroadcast* should include the location of the moving destination of C node L_{c_next} in every RPGM round except L_c. L_{c_next} can be assigned by C node according to the sensing mechanism.

IV. S-SHAPED SENSING MECHANISM

An S-shaped sensing mechanism (SSM) is proposed for detecting the target areas by executing a repeated S-shaped movement. It is assumed that no obstacles are existed in the environment. In “complete detection phase”, the environment is split into number of regions and scanned sequentially. In the “complement detection phase”, the doubtful regions are found according to the sensing records collected in the previous phase and makes complement detection. The process of complete detection phase is shown in Fig. 7.

Complete detection phase

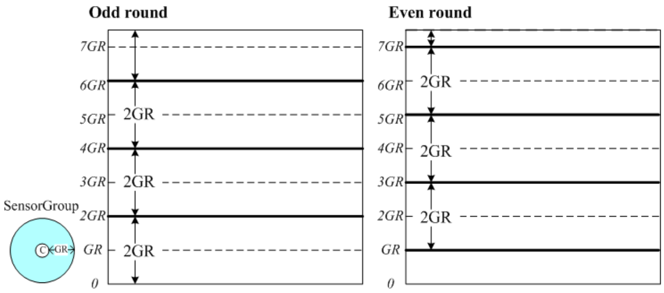
1. Create RL by splitting area into rectangle regions with GR*2
2. Initialize location of SG
3. **for** each region r in RL
4. C moves along r
5. **if** C or M senses border of target area then
6. Save location
7. **end if**
8. **if** C senses border of target area then
9. C changes SG into scan mode
10. **end if**
11. **if** C reaches destination of last r in RL then
12. Enter next phase
13. **end if**
14. **end for**

Figure 7. Operation process of complete detection phase

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2009 proceedings.

In the initial phase, the sensing environment is split into number of rectangle regions with the width of GR*2. The centerlines of the regions are the moving paths of C node (step 1). It is possible that the borders of each region can not be detected. Hence, C node splits the regions with an interlaced style in odd and even rounds of SSM as shown in Fig. 8. After all regions are constructed, C node numbers the moving path of each region and adds these paths into a *RegionList* (RL). The initial position is set on the left side of the first path in RL (step 2). C node starts the detection by scanning each path r in RL with an S-shaped movement (steps 3-14). MSs keep detecting the border of target (steps 5-7). An MS stores its location as a sensing record when it moves through the border. If C node detects the border, it changes from the normal mode to the scan mode and executes step 8-10. ID of recent region is stored into *RegionIndexRecordList* (RIRL). C node stops moving for s seconds and asks all M nodes to expand their moving range from GR to TR at the initial of next round. The reason for the expansion of range is that the border is assumed to be continuous. There should be the extension of border near the location of C node. The sensing records can be collected successfully if it expands the detection range around C node. M nodes must have enough time to move to any location in the expanded range. The stop time s should be set longer than $TR*2/V_m$. C node changes to the normal mode after s seconds and asks all M nodes to gather. C node waits w seconds for all M nodes to back to the range of GR after informing them. To insure that all M nodes have enough time to move back, w is set to be TR/V_m. After waiting for w seconds, C node continues the S-shaped movement. SG enters the second phase of SSM if it reaches the destination of the last r in RL (steps 11-13).

The process of complement detection phase is shown in Fig. 9. C node checks whether RIRL is null (step 1). If RIRL is null, C node can not detect target with the paths in the first phase. C node enters the next round and detect with interlaced paths (step 8). Otherwise, C node constructs *SecondRegionList* (SRL) based on the records in the first phase (step 2).

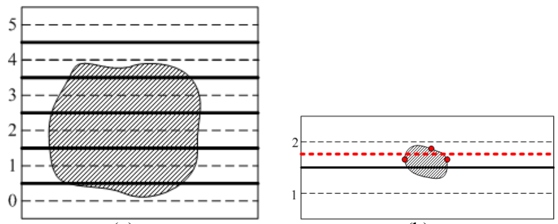


(a) (b)

Figure 8. Region splitting rule (a) Odd round (b) Even round

```
Complement detection phase
1.  if RIRL is not empty then
2.     Finds doubtful regions and creates SRL
3.     C moves along each region in SRL by executing step 3-14
4.     if C reaches the destination of last region in SRL then
5.        Enters next round
6.     end if
7.  else
8.     Enters next round
9.  end if
```

Figure 9. Process of complement detection phase



(a) (b)

Figure 10. SRL construction (a) large area (b) small area

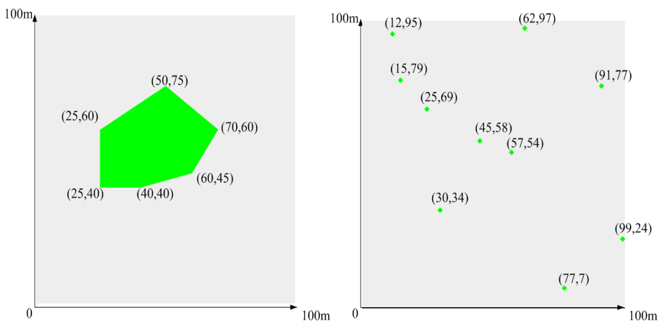
SRL is constructed with the set of paths based on the records of RIRL. C node groups all continuous ID records and sets the neighbor regions of each group as doubtful regions. The centerlines of these regions are added to SRL. As shown in Fig.10(a), the target area is deployed in regions 0, 1, 2, 3, and 4. The border does not touch the centerlines of regions 0 and 4. ID records of 0 and 4 would not be recorded in RIRL. Hence, C node sets the centerlines of regions 0 and 4 as the moving paths and stored into SRL. The other set of paths constructed by the sensing records from M nodes. C node finds the region whose ID is not recorded in RIRL but there exists sensing records of M node. Then C node sets such region as a doubtful region. A path in RIRL is set by calculating the mean of y coordinates of all sensing records. As shown as Fig. 10(b), the target area is deployed in regions 1 and 2. IDs of 1 and 2 can not be recorded in RIRL but M nodes detect the target in regions 1. C node can find the sensing records in region 1 which is a doubtful region. A path in SRL is set by calculating the mean of y coordinates of all records in region 1. After all paths are set, C node sorts the paths in SRL according to the current location. The detection is started by scanning each path in SRL with an S-shaped movement. If C node reaches the destination of the last path, the next round of SSM is started.

There are differences in the processes between SG and CSG. All M nodes in CSG should exchange information with each other to execute the localization algorithm. M nodes keep each other within TR. The expansion range of the scan mode of CSG using SSM is TR/2. The stopping time is TR/V_m. The waiting time is $(TR/ V_m)/2$ to gather all M nodes.

V. SIMULATIPN RESULTS

A. *Scenario and Parameter Settings*

The sensing environment in the simulation is a square area of 100m*100m. Fig. 11(a) shows an apexes polygon as a single scenario. A discrete scenario in Fig. 11(b) shows ten distributed target areas which are rhombuses with width and height of 2m.



(a) (b)

Figure 11. (a) Apexes polygon scenario (b) Discrete rhombus scenario

NS2 is used as the simulation platform. The simulation time is 6000 seconds. Both SG and CSG are composed of one C node and three M nodes. The transmission range is 15m. The period of an RPGM round is 1 second. V_c and V_m are 3m/s and 5m/s, respectively. GR is 2m. In the localization algorithm of CSG, the execution times of the initial phase and the verification phase in an RPGM round are 0.9 and 0.1 second, respectively. The expansion range of the scan mode of SG and CSG are 15m and 7.5m, respectively. In the localization algorithm of CSG, each MS measures the distances between them and their moving directions. The information may be influenced by the interferences in the environment which degrades the accuracy of localization. Random distance error and angle error are added to analyze the effect of interferences. The distance error is set from 0 to 10%. The angle error is set from 0 to 10°. The accuracy of GPS is assumed to be accurate.

The sensing execution rate is used to evaluate the operation of the proposed network. A correct sensing is defined as M node can execute all processes in an RPGM round completely. The sensing execution rate is calculated as

$$R_{ser} = \frac{N_{cs}}{N_{ics}} \quad , \tag{1}$$

where R_{ser} is the sensing execution rate, N_{cs} is the number of correct sensing, and N_{ics} is the ideal number of correct sensing done by all M nodes during the simulation time. The ideal number of the correct sensing can be calculated by

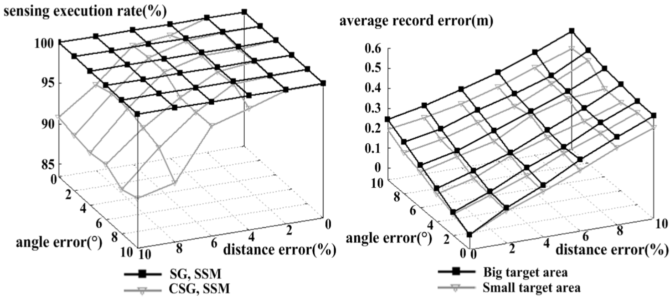
$$N_{ics} = \frac{t_s}{T} \times M \, , \tag{2}$$

where t_s is the simulation time, T is the iteration period, and M is the number of nodes. The higher sensing executing rate, the more stable the operation of CSG is. If the sensing executing rate is decreased, the operation of CSG becomes unstable. M nodes fall out of CSG frequently. Thus the sensing function of M nodes can not be operated correctly.

B. Simulation Results

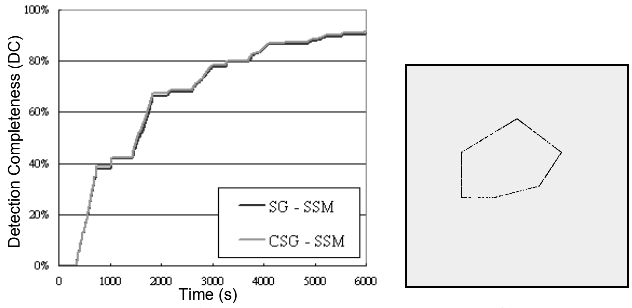
The detection accuracy and the detection completeness are used to analyze the detection performance. The detection accuracy is evaluated by calculating the average error of all sensing records. The error of a sensing record is defined as the shortest distance for the record to the border. The detection completeness is the number of anchor coordinates on the border every 0.2m. If one or more sensing records existed around an anchor within 0.2m, the status of the anchor is set to be "sampled". Hence, the detection completeness is evaluated by calculating the ratio of the number of sampled anchor and the total number of anchor after matching the anchor coordinates with all sensing records.

The average sensing execution rate of both SG and CSG using SSM are shown in Fig. 12(a). All M nodes in SG are equipped with GPS, the accurate locations are obtained. For M node does not fall out of SG, the sensing executing rate of SG is 100%. However, the localization error is increasing in CSG when the interference rising. The localization error increases the possibility of falling out and decreases the sensing execution rate of CSG. The sensing execution rate of CSG decreases to 90% when the distance error and the angle error are both 10%. The detection accuracy shown in Fig. 12(b) is



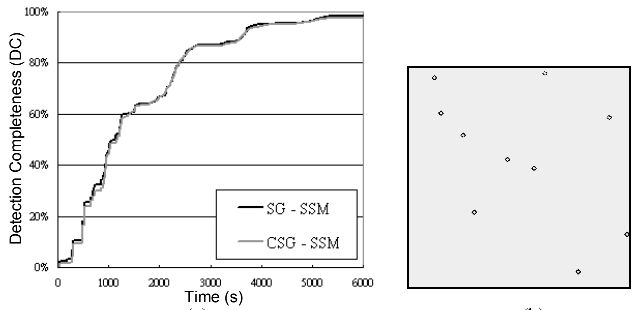
(a) (b)

Figure 12. (a)Sensing execution rate (b) Detection accuracy of CSG



(a) (b)

Figure 13. (a) DC (b) 80% DC of apexes polygon scenario

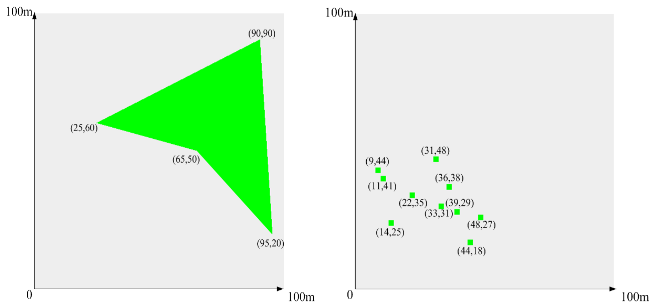


(a) (b)

Figure 14. (a) DC (b) 90% DC of discrete rhombus senerio

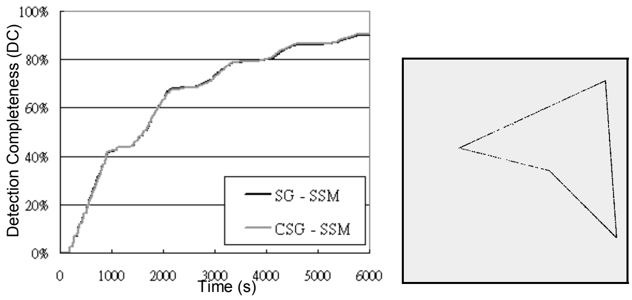
decided by the localization accuracy when sensing records are collected. Since all M nodes in SG are equipped with GPS devices, the error collected by SG is 0m. The increasing of the localization error of CSG results in the increasing of the record errors and the decreasing of the detection accuracy. While distance error and angle error are set to be 10% and 10°, the average record errors in apexes polygon and discrete rhombus scenarios are 0.53m and 0.44m, respectively.

Fig. 13(a) shows the detection completeness with zero angle error and zero distance error for the apexes polygon scenario. The average time per round is 1117 seconds. The average detection time required to achieve 80% detection completeness is 3325 seconds. CSG can acheive the same detection function as that of SG without equiping GPSs on all mobile sensors. Fig. 13(b) shows 80% detection completeness of the sensing records collected per simulation with detection time 3268 seconds. Fig. 14(a) depicts the detection completeness with zero angle error and zero distance error for discrete rhombus scenario. The average sensing time per round is 1387 seconds. The average detection time required to achieve 90% detection completeness is 3649 sconds. Fig.14 (b) reveals 90% detection completeness of the sensing records collected per simulation with detection time 3671 seconds.



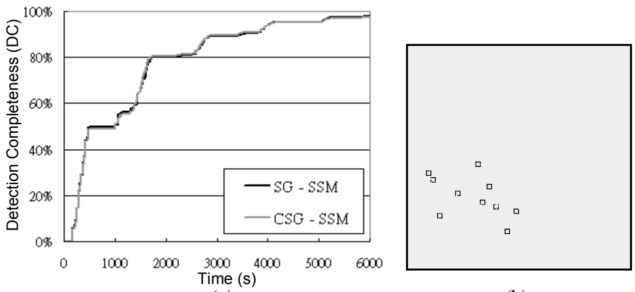
(a) (b)

Figure 15. (a) Concave polygon scenario (b) Discrete square scenario



(a) (b)

Figure 16. (a) DC (b) 80% DC of concave polygon scenario



(a) (b)

Figure 17. (a) DC (b) 90% DC of discrete square scenario

The performance of SG and CSG are the same. However, the total length of borders in the discrete rhombus scenario is shorter. The discrete rhombus scenario requires fewer records to achieve the same dectection completeness. Also the borders of discrete rhombus scenario are denser than those of the the apexes polygon scenario, M node in discrete rhombus scenario can detect the target in one movement with higher possibility. C node in discrete rhombus scenario changes easily into the scan mode so as to correct the sensing records more actively. Thus, higher detection completeness can be achieved in discrete rhombus scenario under the same level of interference.

C. Different Scenarios

Fig.15 (a) provides a concave polygon scenario. Fig.15 (b) shows a discrete square scenario which the targets are squares with width and height of 2 m. Fig.16 (a) depicts the detection completeness in the concave polygon scenario. The average time per round is 1198 seconds. The average detection time required to achieve 80% detection completeness is 3669 seconds. Fig.16 (b) is 80% detection completeness of the sensing records collected in one simulation. The detection time is 3530 seconds. Fig.17 (a) shows the detection completeness in a discrete square scenario. The average time per round is 1878 seconds. The average detection time to achieve 90% detection completeness is 3136 seconds. Fig. 17 (b) is 90% detection completeness of the sensing records collected in one simulation. The detection time is 2838 seconds. As seen from the above results, the detection performances of CG and CSG network structures with S-shaped sensing mechanism are not affected by different scenario settings.

VI. CONCLUSION

To identify target areas automatically is an important application of mobile sensor networks. The proposed network structures, SensorGroup (SG) and Centralized SensorGroup (CSG), can both achieve the area detection with an S-shaped sensing mechanism. In SG, all mobile sensors are equipped with GPS devices. CSG can reduce the device cost since only center node of the group needs a GPS. During the exploring process, GPS-free member nodes can localize themselves by computing information from center node and their historic data. The operation processes and detection phases for both SG and CSG are provided and verified.

The detection performance is evaluated by detection completeness and accuracy for different scenarios in a 100m*100m square environment. The simulation results show that CSG has the same performance as that of SG. The average detection times required to achieve 80% and 90% detection completeness are 3325 and 3649 seconds respectively for polygon scenarios. The average record errors which represent the detection accuracy in different scenarios vary between 0.44m and 0.53m. Both SG and CSG can achieve the exploring jobs in the same performance; even CSG is much more cost-effective than SG. The numerical results also demonstrate that CSG has quite good detection accuracy.

REFERENCES

[1] D. W. Casbeer, D. B. Kingston, and R. W. Beard, “Cooperative forest fire surveillance using a team of small unmanned air vehicles,” International J. of Systems Sciences, vol. 37, pp. 351-360, May 2006.

[2] N. Patwari et al., “Locating the nodes: cooperative localization in wireless sensor networks,” Signal Processing Magazine, pp.54-69, 2005.

[3] J. Clark and R. Fierro, “Cooperative hybrid control of robotic sensors for perimeter detection and tracking,” Proceedings of American Control Conference, vol. 5, pp. 3500–3505, Jun. 2005

[4] F. Franceschini, M. Galetto and D. Maisano, “A review of localization algorithms for distributed wireless sensor networks in manufacturing,” International J. of Computer Integrated Manufacturing, pp. 1-19, 2007.

[5] J. Ray, D. Crump, and M. Chin, “New global positioning system reference station in Brazil,” J. of GPS Solutions, vol. 11, pp. 1-10, 2007.

[6] J. G. McNeff, “The Global Positioning System,” IEEE Transactions on Microwave Theory and Techniques, vol.50, pp. 645-652, Mar. 2002.

[7] R. Reghelin and A. A. Frohlich, “A decentralized location system for sensor networks using cooperative calibration and heuristics,” Proc. of 9th ACM International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems, pp. 139-146, 2006.

[8] M. L. Sichitiu and V. Ramadurai, “Localization of wireless sensor networks with a mobile beacon,” Proc. of IEEE International Conf. on Mobile Ad-hoc and Sensor Systems, pp. 174-183, Oct. 2004.

[9] X. Hong, M. Gerla, and G. Pei, “A group mobility model for ad hoc wireless networks,” Proc. of 2nd International Workshop on Modeling, Analysis and Simulation of Wireless Mobile Systems, pp. 53–60, 1999.

[10] H. Akcan, V. Kriakov, and H. Brönnimann, “GPS-Free node localization in mobile wireless sensor networks,” Proc. of 5th International workshop on Data Engineering for Wireless and Mobile Access, pp. 35-42, 2006.