Mobile Sensor Node Deployment and Asynchronous Power Management for Wireless Sensor Networks

Ren C. Luo, *Fellow, IEEE*, and Ogst Chen

***Abstract*—Mobile sensor node deployment and power management are important issues in the wireless sensor network system. This study designs a mobile sensor node platform to achieve a highly accurate localization mechanism by using ultrasonic, dead reckoning, and radio frequency information which is processed through a particle filter algorithm. Mobile sensor node with accurate localization ability is of great interest to basic research works and applications, such as sensor deployment, coverage management, dynamic power management, etc. In this paper, we propose an efficient mobile sensor node deployment method, grid deployment, where the map is divided into multiple individual grids and the weight of each grid is determined by environmental factors such as predeployed nodes, boundaries, and obstacles. The grid with minimum values is the goal of the mobile node. We also design an asynchronous power management strategy in our sensor node to reduce power consumption of the sensor network. Several factors such as probability of event generation, battery status, coverage issues, and communication situations have also been taken into consideration. In network communication, we propose an asynchronous awakening scheme so that each node is free to switch on or off its components according to observed event statistics and make a tradeoff between communication and power consumption. The deepest sleep state period is determined by the residual power. By combining these methods, the power consumption of the sensor node can be reduced.**

***Index Terms*—Coverage, dynamic power management, mobile sensor node deployment, sensor network.**

I. Introduction

Wireless sensor network is an emerging technology of great interest to many academic units and research centers worldwide. A sensor network is composed of multiple sensor nodes that communicate with each other via a wireless network, and the data of each node are integrated. Motionless design is a common design because long-term motion will consume a lot of energy. From a different perspective, a mobile sensor node is useful to deploy sensor nodes based on the requirement, which makes the system more flexible. Recently, some studies have been devoted to the design of mobile nodes [1], [2] and tried to achieve the best usage of their mobility.

Assigning nodes' location is an important topic on sensor networks, and lots of researchers investigate different efficient deployment methods for various purposes. In terms of communication quality, Corke *et al.* [3] present a sensor network deployment method that determines gaps in connectivity of the deployed network and generates a plan for repair, which completes the connectivity. In the field of target tracking, Chakrabarty *et al.* [4] propose the first systematic theory that leads to novel sensor deployment strategies for effective surveillance and target location.

A distributed energy-efficient deployment algorithm is proposed by Heo and Varshney [5]. The goal is the formation of an energy-efficient node topology for a longer system lifetime. In order to achieve this goal, they employ a synergistic combination of cluster structuring and a peer-to-peer deployment scheme. Moreover, an energy-efficient deployment algorithm based on Voronoi diagrams is also proposed here. Rahimi *et al.* [6] show another deployment method that extends the lifetime of a wireless sensor network. In their system, some nodes are autonomously mobile, allowing them to move in search of energy, recharge, and deliver energy to static energy-depleted nodes.

Coverage is the most considered factor in nodes' distribution, and there are three main deployment methods. The first method discusses the variety of regular deployment topologies, including circular and star deployments, as well as triangular, square, and grid deployments, and analyzes each topology's performance [7], [8]. The second method utilizes the virtual forces to determine the placement of all sensor nodes [9]–[12]. The third protocol is based on the principle of moving sensors from a heavily deployed region to a dispersedly deployed area. For example, Wang *et al.* [13] use Voronoi diagrams to discover the coverage vacancies and design three movement-assisted sensor deployment protocols, including VEC (vector based), VOR (Voronoi based), and minimax.

In [14], the author proposes an event-based power management policy, where the node can update the probability of event generation. It is a useful method on a single node, but it has some shortcomings in a large sensor networking system.

In this study, we develop a mobile sensor node, propose a grid deployment method, and design asynchronous power

Manuscript received August 24, 2010; revised April 25, 2011; accepted July 18, 2011. Date of publication September 15, 2011; date of current version February 3, 2012.

R. C. Luo is with the Center for Intelligent Robotics and Automation Research, National Taiwan University, Taipei 10617, Taiwan (e-mail: renluo@ntu.edu.tw).

O. Chen is with the Department of Electrical Engineering, National Chung Cheng University, Minhsiung 62102, Taiwan, and also with the Center for Intelligent Robotics and Automation Research, National Taiwan University, Taipei 10617, Taiwan (e-mail: jchen@ira.ee.ntu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2011.2167889

0278-0046/$26.00 © 2011 IEEE

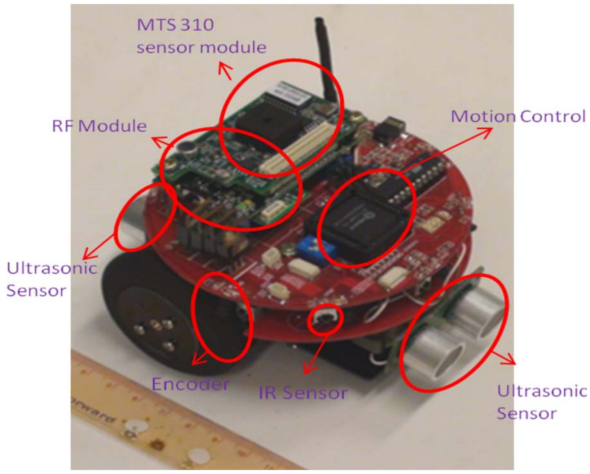

Fig. 1. "LuoMote" mobile sensor node.

management for a wireless sensor network system. The main contributions of this study are outlined as follows.

1) Develop a relatively accurate mobile sensor node localization approach to implement the algorithm of the mobile sensor network.
2) Speed up the calculation on fusing dead reckoning (DR) and ultrasonic (US) by the use of the received signal strength (RSS) of radio frequency (RF).
3) Develop an efficient mobile sensor node deployment method in the environment where predeployed boundary conditions, obstacle effects, and hot zone issues have been considered.
4) Extend the dynamic power management approach to sensor network and take into account the battery capacity and the coverage area during the state transition. Power conservation can be achieved based on the asynchronous power management method.

This paper is organized as follows. The hardware architecture of the mobile sensor node is described in Section II. Section III discusses the fusion-based localization system. The grid deployment is presented in Section IV. Section V describes the proposed asynchronous power management, which can improve the efficiency of the sensor network system. The simulation and experimental results are demonstrated in Section VI.

II. Hardware Architecture

The main design considerations for the mobile sensor node include compactness, effectiveness, and low cost. The design principle can be found in [15]. Under these considerations, the sensor node MTS310 with a wireless communication module MICAz manufactured by Texas Instruments connected to the mobile platform is developed. More details can be found in [16].

Fig. 1 shows a LuoMote mobile node measured 14 cm in height, 16 cm in width, and 14 cm in length. The main components include the MICAz and MTS310 sensor modules which are responsible for sensing data and communicating with other nodes. In addition, US and IR sensors are devoted to obstacle avoidance. Data obtained from these sensors are transmitted to 8051, and the motion path can be modified dynamically.

III. Localization System

The localization system of LuoMote uses US, DR, and RF information to locate the coordinates of the mobile sensor node. Through a particle filter mechanism, the location of the mobile sensor node can be found accurately and efficiently.

A. RSS

RSS is defined as the voltage measured from the RSS indicator circuit. RSS signals can be measured via a receiver without additional energy consumption while communication was established. RSS measurement is a relatively inexpensive method and easy to implement. It is very useful in localization research.

1) Statistical Model: The most widely used signal propagation model is based on a variety of empirical results [17]–[20] and analysis of evidences [21]

$$P_d = P_{d_0} - 10 n_p \log\left(\frac{d}{d_0}\right) + X_\sigma \tag{1}$$

where P_{d_0} is the RSS in decibels at reference distance, n_p is denoted as the path loss exponent typically between two and four [17], and X_σ is a Gaussian random variable with zero mean and σ^2 variance. The d and d_0 in (1) are defined as the estimated distance and reference distance.

2) CRB Formulation: The Cramér-Rao bound (CRB) calculates a lower bound on the covariance of any unbiased location estimator, which is useful to developers. The CRB can be used to test localization algorithms and the bound's functional dependence. A detailed derivation of the CRB is provided in [22]. The most notable advantage of the CRB is that the lower bound on the estimation variance can be calculated. All that is required is to calculate a CRB of the random measurements. The CRB can be expressed as

$$Cov(\hat{\theta}) \geq \left\{ E\left[-\frac{\partial^2}{\partial^2\theta} \ln f(Z|\theta) \right] \right\}^{-1} \tag{2}$$

where $Cov(\hat{\theta})$ is the covariance of the estimator, $E[\cdot]$ indicates the expected value, $f(Z|\theta)$ is the probability density function, Z is the random measurement, and θ are the parameters that are to be estimated from the measurements.

B. US Localization System

The sensing range of US sensors used in this study ranges from 2 to 300 cm. The prior map information is required for the US localization system. The occupancy grid map is used in this research as a map building technique.

C. DR

DR [23]–[25] allows a navigator to determine its present position by projecting its past orientation and speed over ground from a known past position. It can also determine its future position by projecting an ordered path and speed of advance from a known present position.

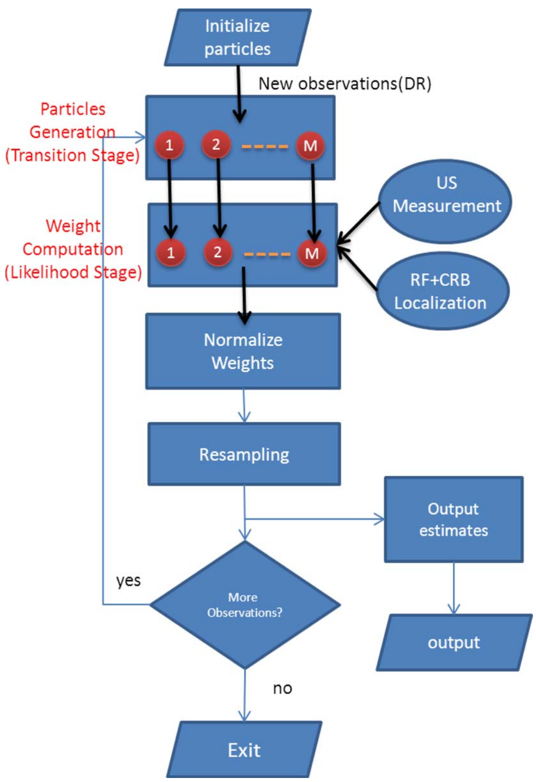

Fig. 2. Flowchart of the particle filter approach.

D. Particle Filter

Particle filter [24] is an alternative nonparametric implementation of the Bayes filter. The goal of particle filter is to estimate the sequence of the hidden parameter, based only on the observed sensor data.

Fig. 2 shows the flowchart of particle filter which is used in this research. The main steps in particle filter are initialization stage, transition stage, likelihood stage, resample stage, and output stage.

In the initialization stage, all particles are distributed in the same pose at the starting point. The summation of all weights is equal to one. Each particle has its own state which consists of its pose and important factor.

In the transition stage, each particle will use DR to update its pose by a control input. The control input is the differential velocities of two wheels V_l and V_r, which represent the velocities of the left and right wheels, respectively.

Moreover, all particles go through this operation in this system. The important factor or weight of each particle is updated while measuring data inputs in the likelihood stage.

In the likelihood stage, each particle will generate virtual sensor data. While measuring the (US and RF) data inputs, each important factor of a particle will be calculated according to the distance between measurement data and virtual sensor data.

In the resample stage, the weight in each particle is normalized and then resampled to obtain a uniform weight distribution. The summation of all weights is equal to one.

In the output stage, the center of gravity or average of weight from all particles is calculated.

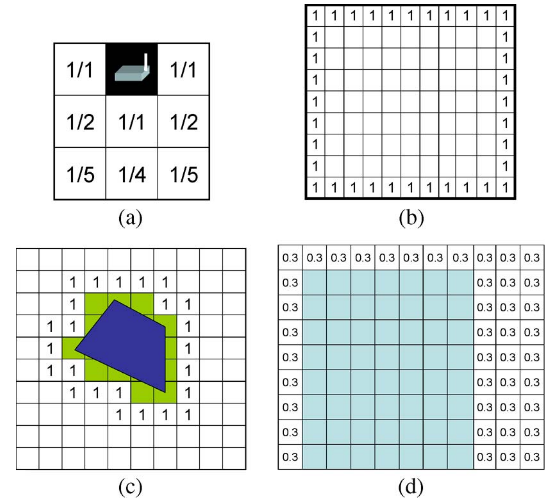

Fig. 3. (a) Weighting field built up by the top node. (b) Weighting field built up by the boundary. (c) Weighting field built up by the obstacle. (d) Weighting field built up by the "hot zone" effect.

IV. Automatic Grid Deployment Method

We assume that the monitored area is deployed with location-known wireless sensor nodes. The mobile nodes will move to the suitable position so that the coverage and uniformity of the system can be improved. Here, a grid method is proposed for finding the exact destination.

The map of the environment is divided into multiple grids and is known to the control center in our system. Each grid is a square with the same edge, and the grid size is adaptive. Smaller grids can achieve better resolution, while larger grids can reduce the complexity of computing. The grid deployment method will evaluate the weighting value of each grid and calculate the minimal region.

A. Predeployed Node Effect

A grid will be marked when a static node is placed within its range. Other grids calculate their weighting values and form the weighting field of this node at the same time. The length of the grid edge is defined as one unit, and we compute the difference of (x, y) and find the square between the marked grid and the evaluated one as defined in (3). The weighting value of the grid can be obtained by taking the inverse of the evaluated value. A typical scenario is described as follows and shown in Fig. 3(a). Assume that the map is divided into nine grids and a static node is placed at the top central corner. Then, all weighting values could be built up immediately. For example, the distance square between the centers of the bottom left grid and the marked grid is five, so the weighting value of this grid is one-fifth. Other values are also set up by the same way, and the weighting field of this static node is established by integrating all individual values

$$V_i(x_i, y_i) = \sum_i \frac{1}{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{3}$$

where $V_i(x_j, y_j)$ is the weighting value of grid (x_j, y_j) built by node i.

B. Boundary Effect

As shown in Fig. 3(b), the coverage of the sensor node next to the boundary will be composed, and the mobile node might hit against the wall due to the inaccuracy of mobility. To avoid these situations, this algorithm can be modified by defining the values of grids next to the boundary as one

$$V_{\mathrm{bou}}(x_j, y_j) = 1. \tag{4}$$

C. Obstacle Effect

The grids covered by obstacles also need to be considered as shown in Fig. 3(c). This can prevent the mobile node from being deployed inside the obstacle. For the same reason above, the values of grids next to the obstacle are set as one

$$V_{\mathrm{obs}}(x_j, y_j) = 1. \tag{5}$$

D. Hot Zone Effect

Uniform distribution is not always the best policy under every situation. For some specific purposes, one may hope that the sensor nodes aggregated in certain areas of interest, which are referred to as the "hot zone." In this condition, we can assign smaller values to "hot zone" and larger values to other regions. To avoid overimbalanced disposition, the weighting parameter caused by the hot zone effect should not exceed the range between 0 and 0.4 in general. For instance, the region in the light blue color is a hot zone in Fig. 3(d). This means that more nodes are required to be placed in this region, so that the values of grids with the light blue color are defined as 0 and others are defined as 0.3

$$V_{\mathrm{hot}}(x_j, y_j) = [0 : 0.4]. \tag{6}$$

After considering all the weighting fields caused by environmental factors such as node effects, boundary effects, obstacle effects, or hot zone effects, a final total weighting field can be obtained by summing them up. The grid with the minimal weighting value is the position, required to be deployed with the highest priority, which is exactly the goal of our mobile node

$$V_{\mathrm{tot}}(j) = \sum_i V_i(j) + V_{\mathrm{bou}}(j) + V_{\mathrm{obs}}(j) + V_{\mathrm{hot}}(j) \tag{7}$$

$$Goal = \min\{V_{\mathrm{tot}}(1), V_{\mathrm{tot}}(2), \ldots, V_{\mathrm{tot}}(n)\} \tag{8}$$

where $V_{\mathrm{tot}}(j)$ is the total weighting value of grid j built by all environmental factors. $V_i(j)$, $V_{\mathrm{bou}}(j)$, $V_{\mathrm{obs}}(j)$, and $V_{\mathrm{hot}}(j)$ denote the values caused by the boundary, obstacle, and hot zone effects incrementally, and n stands for the total number of grids.

V. Dynamic Power Management Strategy

Power management is the most critical issue in wireless sensor networks. Without careful management, the lifetime of the sensor network will be reduced.

TABLE I
Sleep States of the Sensor Node

	CPU	Memory	Sensor	Radio
State0	Active	Active	On	Tx, Rx
State1	Idle	Sleep	On	Rx
State2	Sleep	Sleep	On	Rx
State3	Sleep	Sleep	On	Off
State4	Sleep	Sleep	Off	Off

A. System Model

We modify the algorithm proposed in [26] by considering more factors such as battery status, coverage of the whole sensor system, and communication situation. All these factors determine the sleep state and the sleep period of a single node.

1) Sleep State Model: In our design, we define five sleep states S_k $(k = 0, 1, \ldots, 4)$, and the larger the n, the deeper the sleep state. S_0 represents an awaken state, and $S_1 - S_3$ represent shallower sleep states. S_4 is the deepest sleep state in which almost all components are turned off. The five states are listed in Table I.

Ideally, it is most efficient that the node enters the sleep mode immediately after all tasks are finished and wakes up until an event happens. However, it is not true in reality. Before entering sleep states, the node will save some computed data and system parameters into its register. Awakening a sensor node also needs lots of energy and extra time. All of these will cost energy overhead and time latency.

Therefore, if state transition occurs too often, a lot of energy will be wasted. We define P_{sk} as the power consumption in sleep state k and P_{s0} as the power consumption in active state. We also define the additional energy consumption due to state transition as W_{add} and the time delay caused by state transition as T_{add}. We can find out that only when

$$P_{sk} \times (T_{\mathrm{add}} + T) + W_{\mathrm{add}} \leq P_{s0} \times T. \tag{9}$$

In other words, only when the sleeping interval T satisfies

$$T \geq \left[\frac{P_{sk}}{P_{s0} - P_{sk}} \times T_{\mathrm{add}} + \frac{W_{\mathrm{add}}}{P_{s0} - P_{sk}}\right] \tag{10}$$

the transition will be worthwhile. The overhead and latency caused by state transition are different with k. The larger the k, the greater the latency and overhead will be. Here, we define $T_{\mathrm{th}}(k)$ as the least time that the node should stay in state k.

2) Event Generation Model: According to the event generation model presented in [17], we assume that the spatial event generation distribution of the whole monitored area is a Poisson process with an average event rate given by λ_{tot}. Each sensor has a uniform sensing area a; P_{xy} denotes the probability of event generation of a specific point (x, y), so the probability of event generation in the sensing area of node i can be modeled as

$$P_{ei} = \frac{\int_a P_{xy}(x, y)dxdy}{\int_A P_{xy}(x, y)dxdy}. \tag{11}$$

Here, $P_i(t, n)$ denotes the probability that n events happen in the sensing area of node i during the t interval. The probability that no event has happened in threshold interval $T_{\rm th}$ is

$$P_i(T_{\rm th}, 0) = \sum_{i=0}^{\infty} \frac{e^{-\lambda_{\rm tot} T_{\rm th}} (\lambda_{\rm tot} T_{\rm th})^i}{i!} (1 - P_{ei})^i = e^{-P_{ei} \lambda_{\rm tot} T_{\rm th}} \quad (12)$$

where $P_i(T_{\rm th}, 0)$ is an important parameter used to determine which sleep state the node will enter.

B. Asynchronous Power Management Algorithm

Dynamic power management achieves the objective of power saving by turning off individual components of the node. Here, we design an efficient power management algorithm for solving the problem of energy saving, coverage, and overdensity. In networking communication, an asynchronous paging scheme that each node is free to switch on or off its components based on observed event statistics is proposed. We also make a trade-off between communication and power consumption, which will influence the transmission rate slightly. Finally, we adjust the deepest sleeping interval by taking account of the battery status. By combing these methods, the power consumption of the sensor node can be reduced.

1) Sleeping Policy: The sensor node cannot detect the event or receive message from other nodes in the deepest sleep state. Therefore, how to determine the deepest sleeping probability and the sleep period becomes an important issue. We determine the deepest sleeping probability by an "n-duplicate-covered" method. An "n-duplicate-covered" node means that there are more than $(n + 1)$ nodes monitoring this area.

Therefore, the node only needs to bear $1/n + 1$ of the general workload. Thus, a sensor has $n/(n + 1)$ probability of entering the deepest sleep state (S_4) and $1/(n + 1)$ probability of entering the shallower sleep states (S_1, S_2, S_3) after it finishes all tasks. The node in the deepest sleep state is awakened only by the clock counter after a sleep period. We define the deepest sleep state period T as

$$T = \mu \times e^{\left(\frac{V_{\rm sta}}{V_{\rm pre}}\right)}. \quad (13)$$

Here, we can choose any μ as needed. $V_{\rm sta}$ denotes the standard working voltage of the battery, and $V_{\rm pre}$ represents the present voltage of the battery. For example, the standard working voltage of a Li battery is 3.6 V, but the actual working voltage is between 2.8 and 4.2 V. The node in the shallower sleep states will determine its sleep state ($k = 1$, 2, or 3) according to the event generation probability. Using the formula

$$P_i(T_{\rm th}, 0) = e^{-P_{ei} \lambda_{\rm tot} T_{\rm th}} = e^{-\lambda_i T_{\rm th}} \quad (14)$$

we can easily obtain the probability that no event happens during the $T_{\rm th}(k)$ interval. Here, the parameter λ_i indicates the mean rate of event generation (time elapsed divides by the total number of events registered by node i), and the value of λ_i may change with time. If $P_i(T_{\rm th}(k), 0)$ is greater than a fixed value P, the node will enter sleep state k.

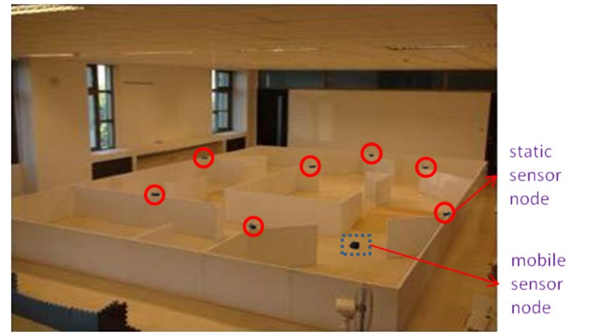

Fig. 4. Testing environment with installed static and mobile sensor node.

2) Awakening Policy: We propose event-, message-, and timer-driven methods to awaken the sensor node in our mechanism as follows.

Event driven. An event-driven policy works under a shallower sleep state. When an event such as an abrupt change in temperature or a signal generated by a moving object occurs, the sensor produces an interrupt and awakens the CPU. The CPU processes the signals with data fusion algorithms or transmits them to the other nodes, and then, it goes to sleep again.

Message driven. A message-driven policy works under $k = 1$ or 2 because the receiver is still on in these sleep states. When node i needs to transmit packets to node j in sleep state k, it will send a message first to awaken node j. After receiving the awakening message, node j will check if the sleeping time is longer than $T_{\rm th}(k)$. If it is true, the node wakes up instantly; otherwise, it will awaken the node until the sleep time t is more than $T_{\rm th}(k)$. Then, the active node j will send an acknowledgment to node i. If node i receives the acknowledgment in $T_{\rm th}(2)$, it will send the packets to node j; if not, it will send an awakening message to another node. Such method can avoid the huge energy consumption caused by failed packet transmission.

Timer driven. The third policy uses a counter interrupt, which is the only way to awaken the node in the deepest sleep state. According to the battery status and the μ parameter, we can obtain the deepest sleeping period.

By using our methods, we can easily implement dynamic power management that considers the probability of event generation, the packet transmission, the coverage issue, and the battery status.

VI. Simulation and Experimental Results

A. Mobile Node Localization Experimental Result

The testing environment for the mobile sensor node is constructed in our Intelligent Robotics and Automation Laboratory in National Taiwan University as shown in Fig. 4. The red circles in Fig. 4 represent static sensor nodes, and the blue dotted rectangle represents a mobile sensor node.

We test LuoMote several times in this environment to obtain the DR information. Fig. 5(a) shows the real trajectory of LuoMote moving several times on a rectangular path, and Fig. 5(b) shows the information measured from DR alone.

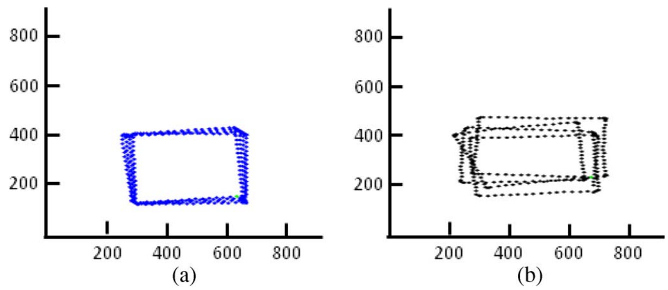

Fig. 5. (a) Real trajectory of the mobile node. (b) Measurement result from the DR system.

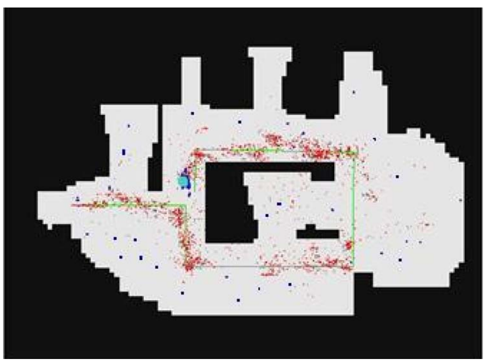

Fig. 6. US localization using the particle filter approach.

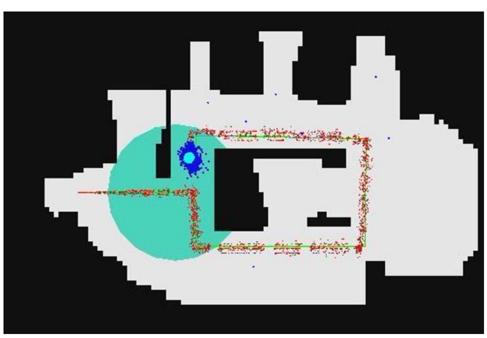

Fig. 7. US and RSS localization system using the particle filter approach.

In Figs. 6 and 7, the black regions represent possible obstacles on the map, the red dots represent the pose prediction of the mobile node, the green lines represent the desired trajectory of a mobile node, the blue dots represent the particles at the present time, the big light blue circle represents the CRB of the RSS, and the small light blue circle represents 5% weight of particles to reduce the influence of relatively irrelevant particles. Fig. 6 shows the estimated result using particle filter [24] without using the RSS localization mechanism, and the mean of particles drifts from the real trajectory. It can be found that some particles do not concentrate on the present position. Fig. 7 shows the fusion result of using the RSS localization mechanism which is useful to help particles converge quickly. Particles concentrate on the RSS covariance region which is generated from the CRB method. The mean of particles is more aligned with the real trajectory than the mean of particles without using the RSS localization mechanism. Fig. 8 shows the fused result of the DR system with US and RSS, and the estimated results are close to the real trajectory.

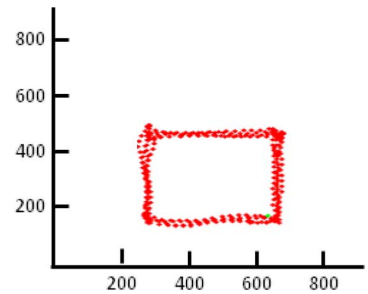

Fig. 8. Mobile node trajectory by the fusion of DR, US, and RSS.

B. Simulation Result of Grid Deployment Method

The 50-m-by-50-m square monitored area is divided into 2500 small uniform square grids. Each grid has the same length of 1 m, each node is equipped with identical sensor, and the sensing radius is equal to 5 m.

The weighting values of the grids within three units from the boundary are all set to one. If there are no predeployed static nodes, the configuration of 25 mobile nodes according to our method is shown in Fig. 9(a). Here, the red points indicate the mobile ones, and the number next to each point represents the order of deploying. For example, 1 and 2 represent the first and second mobile nodes being deployed. According to the simulation, it has an almost perfect distribution of 25 nodes in terms of coverage and uniformity.

Next, consider the situation that some static nodes are placed in advance. Fig. 9(b) shows the distribution of 20 static nodes randomly deployed in this region. By using the automatic grid deployment policy, the first deployed location can be obtained as shown in Fig. 9(c). After deploying 20 nodes incrementally, the configuration is shown in Fig. 9(d). Fig. 9(e) shows the distribution of 20 static nodes with three obstacles placed in the environment. Fig. 9(f) shows the final topology after adding ten mobile nodes. Note that we set the values of the grids within two units near the obstacles to be one. The blue points represent static nodes, and the green rectangles indicate obstacles.

The result shows good uniformity, particularly after distributing a certain number of nodes. More nodes make uniformity better according to our algorithm. It helps a lot to increase the system efficiency, improve energy conservation, and decrease the probability of missing an event.

The coverage of nodes is the most important factor to evaluate the deployment performance. It shows the extent that one can monitor in the environment. The better the coverage, the higher the probability that the events can be detected in this area. Covering more regions with less sensor nodes means that this policy is more powerful.

An overtight distribution will cause an "overlap" situation and make the system inefficient; looser deployment results in "coverage vacancy" (namely, the area that cannot be monitored by any node) and influences the quality of surveillance. Here, the coverage is defined as the ratio of areas located in some node's sensing region to the whole monitored environment by referring to [9]

$$Coverage = \frac{\bigcup_{i=1,\ldots,N} A_i}{A} \tag{15}$$

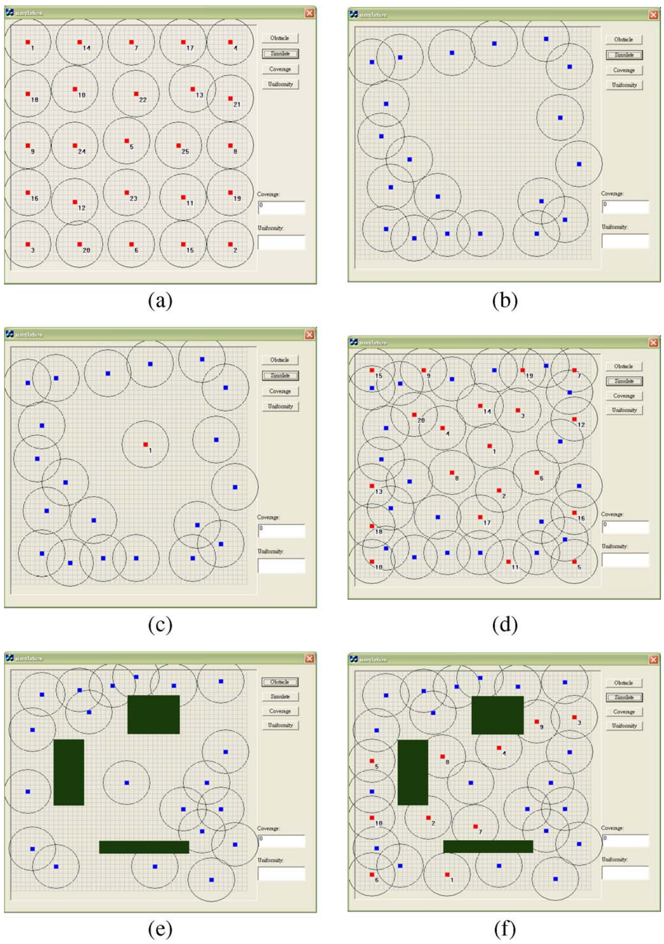

Fig. 9. Simulation results of different situations.

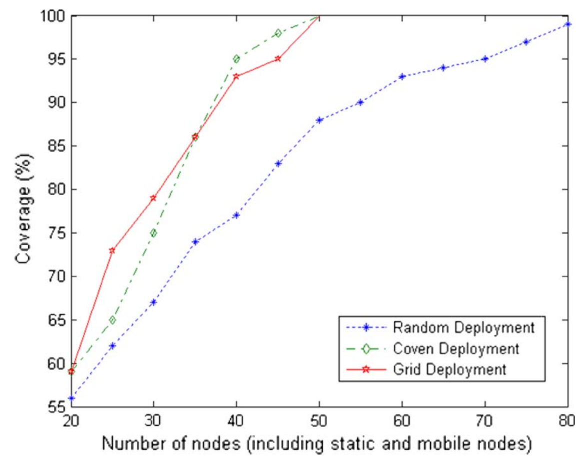

Fig. 10. Comparison between the COVEN, grid, and random deployment methods.

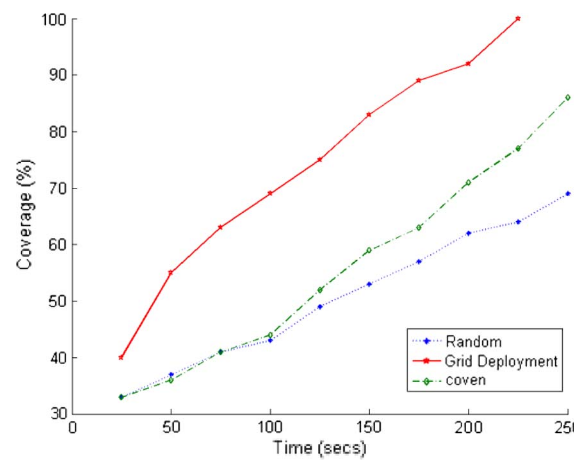

Fig. 11. Time versus coverage.

where A_i stands for the area covered by the ith node, N denotes the total number of nodes, and A is the area of the monitored environment.

Now, we place mobile nodes incrementally into the environment and calculate the individual coverage after deploying the nodes with our method. The result shows that the coverage rises rapidly with the increase of sensor nodes. Since the "coverage vacancy" is getting smaller with the increase of deployed nodes, the "overlap" situation will happen when the biggest hole is less than the sensing circle. Thus, the increasing trend of coverage slows down after depositing a certain number of nodes.

The scenario in [14] is most similar to our scheme. It is also suitable for mixed sensor networks, and many of its assumptions are the same as those of our scheme: The static ones should be predeployed in the monitored environment, and the mobile ones are navigated into the coverage vacancy. The algorithm is named the COVerage ENhancing (COVEN) algorithm, and it uses a "Voronoi polygon" to determine the placed positions and the number of estimated holes. Then, it will assign exactly this number of mobile nodes into the environment. Fig. 10 shows the coverage performance of different methods under the same situation; the coverage of the proposed method is superior to those of the random and COVEN deployment methods before 35 nodes. Both grid and COVEN achieve 100% coverage when it deploys 50 nodes. However, the computation time of COVEN is higher than that of the proposed method as shown in Fig. 11. Fig. 11 shows the performance of computation time which is simulated using 50 nodes. In comparison with the proposed grid deployment method, the COVEN mechanism is more complicated in geometric calculation and conflict solving.

C. Simulation of Asynchronous Power Management Result

Fig. 12 shows the power conservation comparison in the random and grid deployment methods. The on-duty nodes can achieve power saving by using the transition algorithm during shallower sleep states. Suppose that 900 on-duty nodes are distributed uniformly over a 30-m-by-30-m area; the configuration of nodes in shallower sleep states is shown in Fig. 13. The node energy consumption tracks the event frequency as shown in Fig. 14. In a scenario without shallower states, there is uniform energy consumption at about 56 mW in all nodes.

VII. Conclusion

In this paper, we have developed a mobile sensor node which uses US, DR, and RSS localization mechanism with particle filter fusion approach to obtain location information with relatively higher accuracy. With accurate location information,

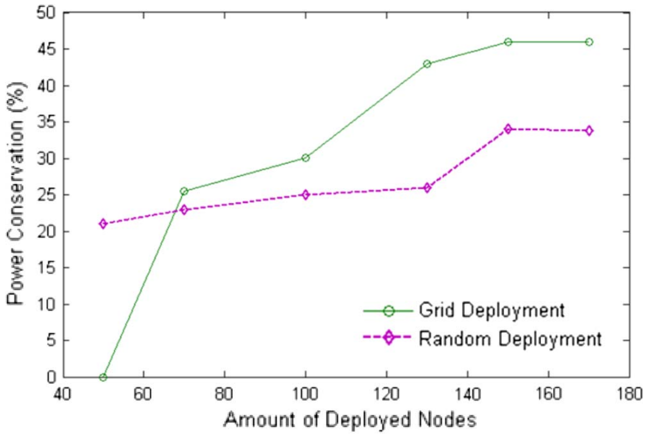

Fig. 12. Power conservation of different deployment methods.

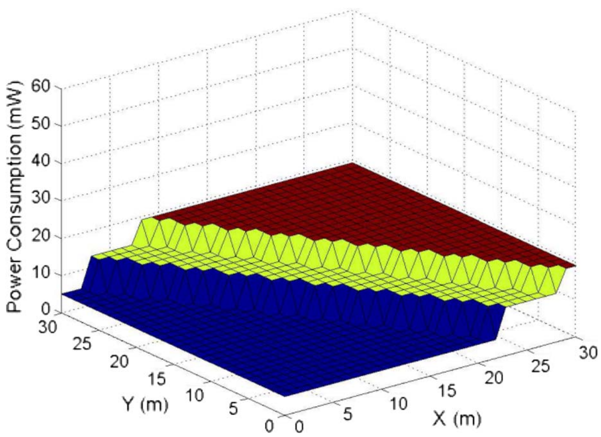

Fig. 13. Simulation result of shallower sleep states.

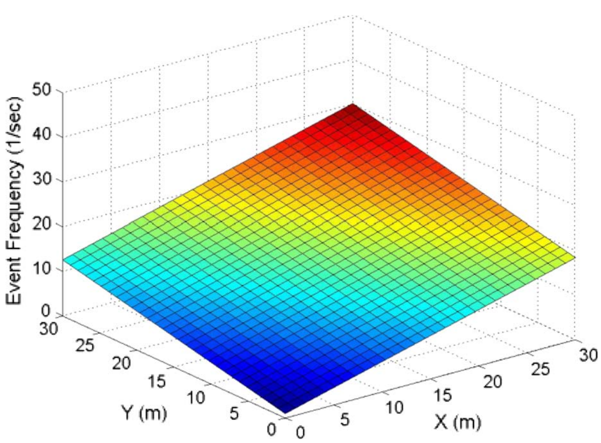

Fig. 14. Event generation frequency.

automatic deployment and high efficient dynamic power management can be achieved.

We have also proposed a grid deployment method which has relatively higher performance of coverage with easy implementation and low computational cost. Only environment information and initial conditions are required. All previously deployed nodes do not need to change their status, and we just assign a mobile sensor node to the destination. The results show that grid deployment can achieve good coverage and uniformity performances.

Asynchronous power management is efficient to reduce power depletion on a sensor network. We design an "n-duplicate-covered" algorithm to guarantee the maximum sensing region. It is similar to the coverage-based off-duty eligibility rule, but we can easily avoid the situation that there are more than two nodes making off-duty decisions simultaneously. Comparing with the "k-perimeter-covered" algorithm, we only need to know the relative positions of other nodes instead of every k value of all the nodes in this sensing area. It can reduce the number of querying and the energy consumption. According to our result, it can easily achieve energy saving in a dense sensor network.

References

[1] G. T. Sibley, M. H. Rahimi, and G. S. Sukhatme, "Robomote: A tiny mobile robot platform for large-scale ad-hoc sensor networks," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, vol. 2, pp. 1143–1148.

[2] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. S. Sukhatme, "Robomote: Enabling mobility in sensor networks," Center Robot. Embedded Syst., Viterbi School Eng., Univ. Southern California, Los Angeles, CA, Tech. Rep. CRES-04-006, 2004.

[3] P. Corke, S. Hrabar, R. Peterson, D. Rus, S. Saripalli, and G. Sukhatme, "Deployment and connectivity repair of a sensor net with a flying robot," in *Proc. ISER*, 2004, pp. 333–343.

[4] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, "Coding theory framework for target location in distributed sensor networks," in *Proc. Inf. Technol.: Coding Comput.*, 2001, pp. 130–134.

[5] N. Heo and P. K. Varshney, "Energy-efficient deployment of intelligent mobile sensor networks," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 1, pp. 78–92, Jan. 2005.

[6] M. Rahimi, H. Shah, G. Sukhatme, J. Heidemann, and D. Estrin, "Studying the feasibility of energy harvesting in a mobile sensor network," in *Proc. ICRA*, Sep. 2003, vol. 1, pp. 19–24.

[7] E. S. Biagioni and G. Sasaki, "Wireless sensor placement for reliable and efficient data collection," in *Proc. 36th Annu. Hawaii Int. Conf. Syst. Sci. (HISS)*, Jan. 2003, p. 127.2.

[8] X. Wu, "Coverage-driven energy-efficient deployment and self-organization in sensor networks," Thesis for the Degree of Doctor of Philosophy, Dept. Comput. Eng., Graduate School, Kyung Hee Univ., Seoul, Korea, Aug., 2008.

[9] N. Heo and P. K. Varshney, "A distributed self spreading algorithm for mobile wireless sensor networks," in *Proc. WCNC*, Mar. 2003, vol. 3, pp. 1597–1602.

[10] S. Poduri and G. S. Sukhatme, "Constrained coverage for mobile sensor networks," in *Proc. IEEE ICRA*, May 2004, vol. 1, pp. 165–171.

[11] D. Popa, H. Stephanou, C. Helm, and A. Sanderson, "Robotic deployment of sensor networks using potential fields," in *Proc. IEEE ICRA*, May 2004, vol. 1, pp. 642–647.

[12] A. Howard, M. J. Matari'c, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Proc. 6th Int. Symp. DARS*, Jun. 25–27, 2002, pp. 299–308.

[13] G. Wang, G. Cao, and T. F. L. Porta, "Movement-assisted sensor deployment," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, Mar. 2004, vol. 4, pp. 2469–2479.

[14] A. Ghosh, "Estimating coverage holes and enhancing coverage in mixed sensor networks," in *Proc. 29th Annu. IEEE Int. Conf. LCN*, Nov. 2004, pp. 68–76.

[15] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4258–4265, Oct. 2009.

[16] [Online]. Available: http://www.xbow.com

[17] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[18] G. Durgin, T. S. Rappaport, and H. Xu, "Measurements and models for radio path loss and penetration loss in and around homes and trees at 5.85 GHz," *IEEE J. Sel. Areas Commun.*, vol. 46, no. 11, pp. 1484–1496, Nov. 1998.

[19] N. Patwari, A. O. Hero, III, M. Perkins, N. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 51, no. 8, pp. 2137–2148, Aug. 2003.

[20] N. Patwari, R. J. O'Dea, and Y. Wang, "Relative location in wireless networks," in *Proc. IEEE VTC*, May 2001, vol. 2, pp. 1149–1153.
[21] A. J. Coulson, A. G. Williamson, and R. G. Vaughan, "A statistical basis for lognormal shadowing effects in multipath fading channels," *IEEE Trans. Commun.*, vol. 46, no. 4, pp. 494–502, Apr. 1998.
[22] H. L. V. Trees, *Detection, Estimation, and Modulation Theory, Part I*. Hoboken, NJ: Wiley, 1968, pt. I.
[23] E. T. Baumgartner and S. B. Skaar, "An autonomous vision-based mobile robot," *IEEE Trans. Autom. Control*, vol. 39, no. 3, pp. 493–502, Mar. 1994.
[24] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005.
[25] T.-H. S. Li, Y. C. Yeh, J. D. Wu, M. Y. Hsiao, and C. Y. Chen, "Multifunctional intelligent autonomous parking controllers for carlike mobile robots," *IEEE Trans. Ind. Electron.*, vol. 57, no. 5, pp. 1687–1700, May 2010.
[26] A. Sinha and A. Chandrakasan, "Dynamic power management in wireless sensor networks," *IEEE Des. Test Comput.*, vol. 18, no. 2, pp. 62–74, Mar./Apr. 2001.
[27] G. Anastasi, M. Conti, and M. Di Francesco, "Extending the lifetime of wireless sensor networks through adaptive sleep," *IEEE Trans. Ind. Informat.*, vol. 5, no. 3, pp. 351–365, Aug. 2009.
[28] L. LoBello and E. Toscano, "An adaptive approach to topology management in large and dense real-time wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 5, no. 3, pp. 314–324, Aug. 2009.
[29] C. F. Huang and Y. C. Tseng, "The coverage problem in a wireless sensor network," in *Proc. Int. Workshop WSNA*, Sep. 2003, pp. 115–121.
[30] R. C. Luo, L. C. Tu, and O. Chen, "An efficient dynamic power management policy on sensor network," in *Proc. 19th Int. Conf. AINA*, 2005, pp. 341–344.
[31] R. C. Luo, L. C. Tu, and O. Chen, "Auto-deployment of mobile nodes in wireless sensor networks using grid method," in *Proc. IEEE ICIT*, Hong Kong, 2005, pp. 359–364.

Ren C. Luo (M'83–SM'88–F'92) received the Ph.D. degree in electrical engineering from the Technische Universität Berlin, Berlin, Germany.

He served two terms as President of National Chung Cheng University, Minhsiung, Taiwan. He was a Full Professor with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, and a Toshiba Chair Professor with The University of Tokyo, Tokyo, Japan. He is currently an Irvin T. Ho Chair and a Distinguished Professor with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan. He has authored more than 400 papers on various topics, which have been published in refereed technical journals and conference proceedings. He is the holder of several patents.

Dr. Luo was the Editor-in-Chief of the IEEE/ASME TRANSACTIONS ON MECHATRONICS (2003–2007). He served as President of the IEEE Industrial Electronics Society (2000–2001). He also served as President of the Chinese Institute of Automation Engineers; Program Director of the Automation Technology Division, National Science Council; Adviser of the Ministry of Economic Affairs; and Technical Adviser of the Prime Minister's Office in Taiwan. He is currently the President of the Robotics Society of Taiwan.

Ogst Chen received the B.S. and M.S. degrees in electrical engineering from National Chung Cheng University, Minhsiung, Taiwan, in 1996 and 1998, respectively, where he is currently working toward the Ph.D. degree.

He is also a Research Assistant with the Center for Intelligent Robotics and Automation Research, National Taiwan University, Taipei, Taiwan. His research interests include multisensor fusion, sensor networks, and intelligent robotics.