

Gradient-Driven Target Acquisition in Mobile Wireless Sensor Networks

Qingquan Zhang¹, Gerald Sobelman¹, Tian He²

¹Department of Electrical and Computer Engineering, University of Minnesota, Twin cities, USA

²Department of Computer Science and Engineering, University of Minnesota, Twin cities, USA

{zhan0511, sobelman}@umn.edu
{tianhe}@cs.umn.edu

Abstract. Navigation of mobile wireless sensor networks and fast target acquisition without a map are two challenging problems in search and rescue applications. In this paper, we propose and evaluate a novel Gradient Driven method, called GraDrive. Our approach integrates per-node prediction with global collaborative prediction to estimate the position of a stationary target and to direct mobile nodes towards the target along the shortest path. We demonstrate that a high accuracy in localization can be achieved much faster than other random walk models without any assistance from stationary sensor networks. We evaluate our model through a light-intensity matching experiment in MicaZ motes, which indicates that our model works well in a wireless sensor network environment. Through simulation, we demonstrate almost a 40% reduction in the target acquisition time, compared to a random walk model, while obtaining less than 2 unit error in target position estimation.

Key words: Wireless Sensor Network, Navigation, Localization, Probabilistic Model, Rescue

1 Introduction

Wireless sensor networks have gained extensive attention in many applications such as tracking, differentiated surveillance, and environment monitoring [1-3]. Moreover, the hybrid systems of mobile objects (e.g. Robots) and sensor networks create new frontiers for civilian and military applications, such as search and rescue missions in which the background environments are inaccessible to humans. A heterogeneous searching team consisting of robots and a wireless sensor network has greater advantage, considering its distributed computation and navigation capability achieved through the cooperation of embedded wireless sensor networks.

Although the applications of mobile sensor networks keep diversifying, several underlying capabilities remain fundamental and critical. In this work, we

focus on the target acquisition – finding the locations of stationary targets using mobile sensor nodes. The challenging problem we address in this work is *to navigate a team of mobile sensor nodes toward the stationary targets fast and accurately while consuming the least amount of energy and other resources*. In this emerging research arena, most research groups employed static wireless sensor networks to navigate the mobile sensor nodes. Tan in [4] used distributed static sensor networks to collect the data and execute local calculations to generate a path for a mobile sensor network to move toward the goal. Although the in-network calculation implemented in that project was quite efficient in creating the shortest routing path, the additional requirement of a stationary distributed sensor network sets a barrier for rescue applications, because of the high cost to cover a large geographic area with a large number of sensors. Other research groups [5] proposed gradient methods in which the mobile wireless sensor nodes move toward the gradient direction assuming that targets carried the most intensive strength of interested signals. However, in all of their implementations, the assistance of a stationary wireless sensor network was assumed to be available in generating a local signal distribution map. A probabilistic navigation algorithm is presented in [6], where a discrete probability distribution of vertex is introduced to point to the moving direction. This algorithm computes the utilities for every state and then picks the actions that yield a path toward the goal with maximum expected utility. The shortcoming of this method is that it requires the arrival of a mobile sensor node to localize the target position and significant communication overhead is introduced by the iteration process.

2 Contribution

In this paper, we propose to compensate those deficiencies by incorporating a prediction model of real-time processes into a mobile sensor network sensing and navigation architecture. We are interested in the mutually beneficial collaboration of the algorithms described above but seek to reduce the costs and provide faster target localization. *The novelty of our approach is the seamless integration of a per-node prediction model with a global prediction model*. The per-node prediction model guarantees that a mobile node can acquire the position of a target alone, while the global prediction significantly reduces the navigation overhead and time, if collaboration among the nodes is available. Specifically, the main contributions of our prediction models are:

- Our model provides more meaningful description of individual sensor readings in term of accuracy and confidence.
- Our model works with a single mobile sensor node as well as a swarm of mobile sensor nodes. In the latter case, the sensor nodes have the ability to share local information in order to draw a global picture, which helps each sensor node to acquire the target along a significantly shorter path.
- The in-network prediction algorithm enables faster yet accurate target position acquisition: sensor nodes would be required to reach the target only

when the model prediction is not accurate enough to satisfy the requirement with an acceptable confidence. This allows a significant reduction in navigation energy.

The remainder of this paper is organized as follows. Section 3 defines the assumptions. Section 4 overviews the design. In Section 5, we present the in-network per-node prediction model. Section 6 describes target acquisition in the context of global prediction and the corresponding mobile sensor node navigation protocol. Section 7 presents empirical data obtained from the MICAZ platform as well as simulation results. Finally, in Section 8, we present our conclusions and future works.

3 ASSUMPTIONS

Our design is based on two assumptions: network connectivity and the self-localization of mobile wireless sensors.

- **Connectivity:** First, wireless sensor nodes in the network are assumed to be able to ensure connectivity. Individual mobile sensor nodes deployed in large area is likely to lose connection to a central base station, if the routing information is not updated. Therefore, it is desirable to maintain connections across a team of mobile sensor nodes while minimizing power consumption and allowing the sensor nodes to achieve their individual goals.
- **Node Self-Localization:** The second assumption hinges on the localization availability for a mobile wireless sensor network. If a mobile sensor node enters an unknown area, it must be able to specify its own location dynamically without a map. This location can be obtained either through GPS such as used in ZebraNet [7] and VigilNet [3]. It can also use a dynamic localization scheme [8] that adjusts the estimated location of a node periodically based on the recent observed motion.

4 OVERVIEW OF PREDICTION MODEL

The objective of our GraDrive target acquisition scheme is to predict the location of stationary targets within allowable uncertainty (or a confidence level) dictated by a rescue plan. To illustrate the design of GraDrive, we start our description with a rescue scenario shown in Fig. 1. Here we note that our method is independent of this rescue application and can be applied in other scenarios as well.

- **Objective:** The control center (base) disseminates a search objective to a mobile sensor network with two parameters, *error tolerances* and *confidence level* of the target, specifying the quality of target acquisition. For example, the objective would be locating a target within 2 meters with at least 95% confidence. The tolerance levels for each mobile sensor nodes can vary correspondingly in case different nodes are designed for different purposes.

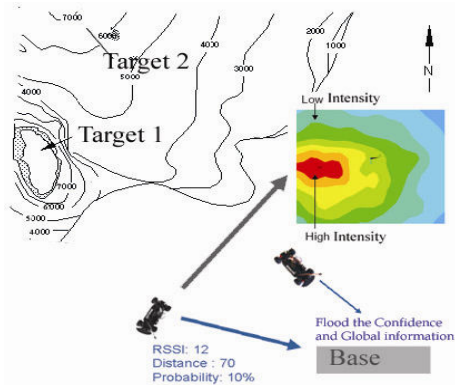


Fig. 1. The Architecture schematic of GraDrive

- **Individual Prediction Model:** Once the search objectives are received by the mobile nodes, individual node decides their most efficient way to locate the potential target with the requested confidence *individually*, using the per-node prediction model. It starts to move toward the direction in which it anticipates the fastest path to reach the confidence.
- **Collaborative Prediction Model:** In addition to its own plan and navigation, sensor nodes also report back to a base station, where all the individual nodes' readings and plans are collected and computed to create a global map and an uncertainty area. If computation results show probability increase by certain interval, e.g.5% to its previous state computation, the base station will disseminate the global prediction value over the network so that each sensor nodes in network can update their model. In other words, the prediction result based on collaborative information overrules the results from the individual prediction model.

As demonstrated in Fig. 2 from (A) to (D), the individual sensor node continuously predicts the target position with increasing probability and move toward the target, the uncertainty area where the target is located shrinks through collaboration among mobile sensor nodes. If collaborative probability calculated reaches the dictated objective, a success of rescue plan is achieved. The position it reports is the exact target position specified. Compared to other static sensor node navigation plans, the prediction results computed by our model still provide considerably more information than MobileRobot [6] and SafeRobot [9].

5 GRADRIVE MODEL DETAILS

In this section, we formally describe our per-node prediction model to estimate the position of a stationary target with certain confidence. This per-node prediction model forms the basis for global collaborative prediction described in Section 6. We note even though we consider an unknown area with multiple

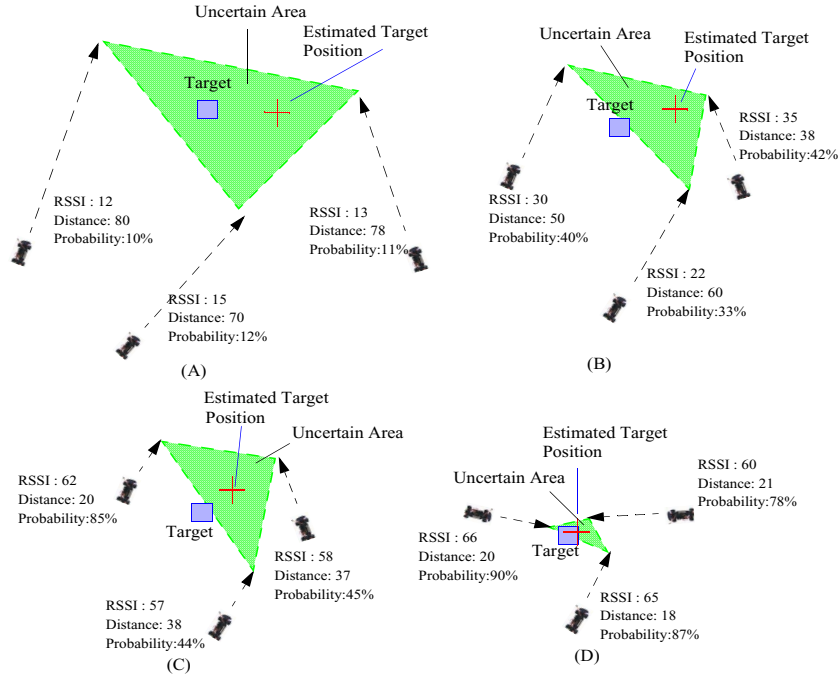


Fig. 2. Collaborative Prediction Scheme of GraDrive

targets, the searching for separate targets is independent to each other as long as the field (RSSI) generated by one target doesn't overwhelm that generated by others. Therefore in the remaining of paper, we focus on only single target acquisition problem.

5.1 Prediction Problem Formulation

Conventionally, we begin with a value-prediction problem, which creates a Received Signal Strength Indicator $F(\theta)$ over a parameter set θ . For example, if $\theta = (d, t, v)$, RSSI is related to d , the distance between a mobile sensor node and the target, t , the time of sampling, and v , the speed of mobile sensor nodes. This model can be established by getting consecutive sensing readings (system states) when a mobile sensor node moves. Typically, the number of parameters in θ is much less than the number of states collected and changing one parameter changes the estimated value of many states. To approximate our model appropriately, we seek to minimize the mean squared error over some distribution, P , of the inputs. There are generally far more states than components in θ . The flexibility of the function estimator is thus a scarce resource. Better approximation at some states can be gained generally only at the expense of worse approximation at other states.

5.2 Distance Prediction Model

In GraDrive, we extend the familiar one-dimensional normal probability density function known as Gaussian distribution to two variants multidimensional distribution. The predicted distance from sensor nodes' current position to predicted target position can be queried or estimated from the model. The multidimensional Gaussian distribution function over two attributes, trust interval and RSSI, can be expressed as a function of two parameters: a 2-tuple vector of means, μ , and a 2×2 matrix of covariances, Σ . Further, we assume the trust interval set by a rescue team is independent of the RSSI received, which means the trust interval of the predicted distance estimation T_i to the mean of historic results μ doesn't change dynamically along the searching process. The two dimensional distribution can be separated for description purposes. Without loss of generality, it is assumed that the predicted distance d is disproportional to RSSI, that can be expressed as $d = r_1/RSSI + r_2$, where r_1 and r_2 are two adaptive parameters that can be determined before the searching process. We note here other RSSI attenuation models can be used here as well without invalidating our approach. We then use historical data or experience data to construct the models, providing r_1 and r_2 at each RSSI value appropriately. Besides offering the predicted distance, a probability model associated the d is also constructed to provide confidence of the prediction, e.g. given a predicted distance of 2 feet, the confidence for this prediction is 95%. The models must be trained before it can be used, a general limitation for probabilistic model. The accuracy of the model, therefore, relies on the accuracy of data used to train it. Once the initial model is constructed, each sensor nodes can query the predicted distance map from saving model and come up with a confidence value. One distribution of the distance d against the confidence p over one RSSI is a Gaussian distribution. Suppose that rescue team have set a trust interval of T_i , given the distribution of distance over one RSSI, we can get the points d_i that satisfied that $P(d_i) - P(u) \leq T_i$. Here we emphases that if the trust interval is too small, the amount of data needed to train the model will increase exponentially.

5.3 Signal Strength Distribution Prediction Model

Besides obtaining the distance d information based on measured $RSSI$, we can further refine the RSSI distribution Model. This distribution model can then be used to navigate the mobile sensor network toward the target at a shortest path. The central element in our approach is to construct a prediction model that represents attributes as accurate as possible in a mobile sensor network. As we discuss above, if the predicted RSSI distribution function depends on parameters including distance d and confidence or probability p , the function can be expressed as $F(d, p)$ considering d and p 's distribution are independent. If we do the Tylor expansion on function F , a polynomial function of attributes d and p is achieved, shown as

$$F(d, p) = f(d_0, d_1, d_2 \dots) f(p) \quad (1)$$

where d_i is the function of distance variable d . To reduce the computation energy consumption, only second order polynomial is considered in our case, which offers a 3-tuple vector of $D = [d_0, d_1, d_2]$:

$$\begin{aligned} d_0 &= c_0 \\ d_1 &= 1/(d + c_1) \\ d_2 &= 1/(d^2 + c_2) \end{aligned} \quad (2)$$

where c_1, c_2, c_3 are constants used to avoid singularity when $d = 0$. Now we can define our gradient distribution function into a simple format as:

$$F = D \bullet A \bullet p \text{ where } A = [a_0, a_1, a_2] \quad (3)$$

Equation 3 is our probabilistic gradient distribution prediction function for attributes of d and p . suppose that each sensor nodes observe the value of attribute D_j to be d_j , we now input sensing reading into a vector of D_j . Thus the vector D is extended as a matrix.

If enough sensing samplings are provided, we can apply non-linear Least Square Fitting to estimate the parameters A . For nonlinear least squares fitting to our undetermined parameters, linear least squares fitting may be applied iteratively to a literalized form of the function until convergence is achieved. Since we can anticipate the power type of fit and have decided initial parameters chosen for our models, the nonlinear fitting has good convergence properties.

In general, the computation of the matrix does cost a large amount of the wireless nodes' energy. The solution in GraDrive is to simplify the prediction distribution function as above, given that prediction function computation can be distributed over the network with collaboration of its neighbors or the data to be delivered back to a base station where stronger computation ability and energy are normally not limitations. If this is the case, the base station creates a gradient distribution map globally using a weighted average method as a function of probability and predicted distribution. This kind of global information is sent back to each individual node involved in application.

6 TARGET LOCALIZATION USING THE COLLABORATIVE PREDICTION MODEL

Based on the per-node prediction model, the mobile sensor nodes can infer the position of target (x, y) and the associated confidence value p . This information is then used to perform global predictions. Specifically, we propose to use a probability-weighted average model for global collaborative prediction, due to its high efficiency and low cost characteristics. The simple rational behind our method is that the sensor nodes having a higher probability are much closer to the intended target.

Generally, if the predicted target location provided by sensor nodes n_1, n_2, \dots, n_k , are $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$ associated with probability value p_1, p_2, \dots, p_k . The estimated position of the target is given as:

$$X = \frac{\sum_{i=1}^k p_k x_k}{\sum_{i=1}^k p_k} \quad Y = \frac{\sum_{i=1}^k p_k y_k}{\sum_{i=1}^k p_k} \quad (4)$$

6.1 Collaborative Navigation and Prediction Protocol

With per-node and global prediction models established, we are now ready to describe how the sensor nodes navigate using these two models.

Initially, sensor nodes enter an intended region with certain moving speeds, moving directions and trust intervals. It should be noted that different mobile sensor nodes could have different moving speed or initial moving direction. After the entrance, the mobile sensor nodes continue to detect the RSSI in its sensing range. The detected RSSI readings are an important input for training the model it is assigned initially. Thus they use a default navigation plan, which is to keep moving forward unless they detect a smaller sensing reading. During the moving process, nodes themselves perform per-node prediction calculation to construct the local RSSI map as described in Section 5.3. Meanwhile, the sensor nodes estimate their distance to the target position according to the sensing RSSI, randomly pick one prediction within its trust interval. The predicted target location information is forwarded back to a base station. To prevent excessive energy consumption in communication, the frequency of updates can be specified in advance. As long as the global picture is not available, individual sensor nodes navigate according to the per-node prediction model. However, if the base station notifies the sensor nodes that it has constructed a global RSSI distribution with certain confidence, each sensor node will combine the information with its current model together and change its direction toward the gradient direction. This process will be repeated until the target position has been discovered locally or at the base station within acceptable confidence.

6.2 Default Navigation Plan when Global Prediction Unavailable

If initially there is no global picture constructed by the base station with acceptable confidence, or if there is only one separated node in the network for rescue plan, or if the network is partitioned or unable to deliver the data, the mobile sensor nodes fall back to the per-node prediction model. Given its current sensing reading, it compares with previous readings stored in memory at each motion step. After getting a smaller sensing reading, it rotates 90 degrees clockwise. The reason for that is that the target position is most likely located perpendicularly to its previous moving direction.

7 EXPERIMENTS AND SIMULATION

7.1 Model Matching Experiment

In order to verify the feasibility of the proposed prediction model and parameter-fitting algorithms, we have prototyped a light sensing system based on Berke-

ley MICAZ nodes. Even though it is stationary, the prediction model and parameter-fitting algorithms can still be verified at the base station site which can be transferred to individual sensor nodes and implemented. Light signal strength is used as an example of RSSI to feed the model. One laptop equipped with motherboard acts as the base station. A lamp works as a target and a series of sensor nodes are deployed as shown in Fig. 3. The sensor nodes detect the sensing reading and exchange the readings to their neighbors. The base station calculates the parameters for the sensor nodes by using the least square fitting method. Fig. 4 shows one set of data fitted by the prediction model. The distance between two adjacent sensor nodes is equal and unified for matching purpose. Since the received signal strength is not an accurate measurement, probability approximation model comes into play. From the matching results, it is shown that the least square method tries to reduce the deviation among the sensing data collected. Other sets of data can also be collected and used to train the model before it can be applied into the mobile sensor scenario.

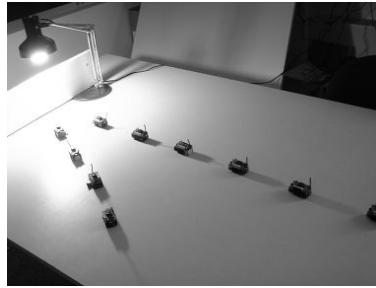


Fig. 3. Model fitting experiment with light as source of signal and using Micaz nodes in array to sense the signal strength

7.2 Simulation Setup

We have developed a program to verify the advantage of using our prediction model to locate the target in a faster approach. In our simulation, a 200×200 m² area is regarded as an unknown space with a target located at the center and a distribution along the diameter is defined. Essentially, it would be any random distribution that having a gradient toward the center. Each distance unit is represented as the smallest unit that the mobile sensor nodes can travel each time during simulation. The navigation algorithm is used to simulate the mobility of objects. Initially, the mobile nodes are located at the edges of the area. The initial direction is randomly picked by each mobile sensor node. If some sensor nodes move outside the simulation region, they bounce their moving direction back into simulation area. Under simulation, each mobile sensor node moves at a constant speed in integer multiples of 1m/s. After each time unit (1

second in our case), a node determines their next moving direction according to our algorithm.

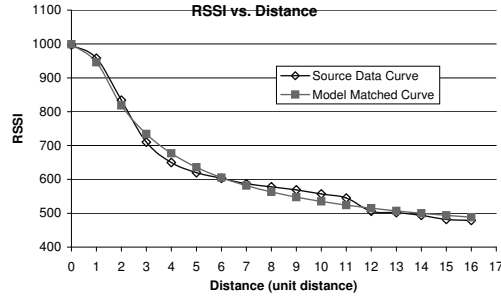


Fig. 4. The predicted model with real sensing data

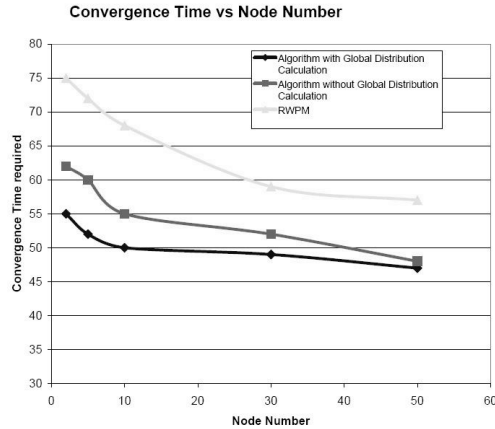


Fig. 5. Convergence time with node number for different models

7.3 Delay in Target Acquisition

We first experiment on comparing our algorithm (w/o global distribution calculation option) against Random Way Point Model. The simulation results (Fig. 5) suggest that even without a global distribution calculation mode turning on, our default algorithm (rotating 90 degree counterclockwise) still provides 30% faster estimation than the random way method. If global calculation mode is on, then initially the sensor nodes still use default plan, but if the global signal strength distribution is available, it moves faster than the default algorithm.

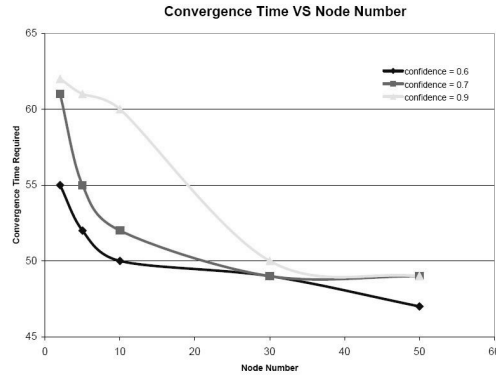


Fig. 6. Convergence time with Node number under different required confidence level

7.4 Impact of the Confidence p

We also compare the impact of different required confidence level on the convergence time as shown in Fig. 6. It is clear that if the required confidence level goes beyond 90%, it will take much longer to simulate simply because it requires at least 2 nodes to get very close to target position. It is reasonable to choose a relative high confidence level e.g. 80% in order to balance accuracy and time cost.

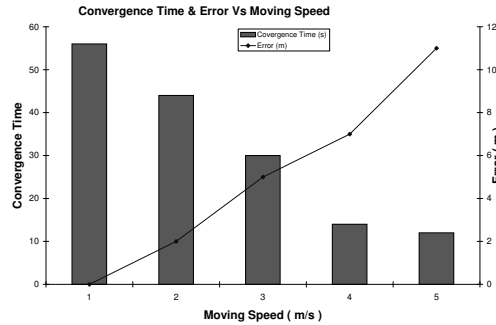


Fig. 7. Convergence time and Accuracy with different moving speeds of mobile sensor nodes

7.5 Impact of the Target Speed

In Fig. 7, we further investigate the relationship between the moving speed of sensor nodes and prediction accuracy of target location. The convergence time correlated directly with moving speed of each sensor node since the average time for sensor nodes to get closer to target is reduced. However, the accuracy of

prediction gets worse if the speed increases because the minimum deviation for the prediction is increased as well. Therefore the error continues to grow in the prediction as node moves faster from its original location. In the situation of high speed, accuracy error larger than 10 units is shown. To protect against inaccuracies in the prediction model of mobile sensor nodes, a user must set a limit for moving speed of sensor nodes.

8 CONCLUSION AND FUTURE WORK

In this paper we present a probabilistic prediction model for dynamic target localization and evaluation of the localization algorithm. Our model does not require any known map to determine the positions of potential targets. Also the proposed gradient driven algorithm leads to a 40% reduction in time compared to that of a random working model. The relationship between sensor density and convergence time can be used as a reference of consideration for doing planning of such a mobile sensor network. Even though the computation power could be large, the error of the predicted target position can reach to almost zero and in a short time (about only 47sec). As future work, we would like to implement our algorithm on off-the-shelf hardware platforms. We would also need to design a speed self-adjusting algorithm so that the sensor node has the ability to trade off performance and cost.

References

1. A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita, "A Wireless Sensor Network for Target Detection, Classification, and Tracking," *Computer Networks (Elsevier)*, 2004.
2. J. Liu, J. Reich, and F. Zhao, "Collaborative In-Network Processing for Target Tracking," *J. on Applied Signal Processing*, March 2003.
3. T. He, S. Krishnamurthy, J. A. Stankovic, and T. Abdelzaher, "An Energy-Efficient Surveillance System Using Wireless Sensor Networks," in *MobiSys'04*, June 2004.
4. J. Tan, A. Verma, and H. Sawant, "Selection and navigation of mobile sensor nodes using a sensor network," 2006.
5. I. Chatzigiannakis, S. Nikolettseas, and P. Spirakis, "Distributed communication algorithms for ad hoc mobile networks," *J. Parallel Distrib. Comput.*, vol. 63, no. 1, pp. 58–74, 2003.
6. M. Batalin, G. Sukhatme, and M. Hattig, "Mobile robot navigation using a sensor network," in *IEEE International Conference on Robotics and Automation*, 2004.
7. P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet," in *Proc. of ASPLOS-X*, October 2002.
8. S. Tilak and et al., "Dynamic localization control for mobile sensor networks," in *Conference Proceedings of the 2005 IEEE International Performance, Computing and Communications Conference*, 2005.
9. H. H. Gonzalez-Banos and J. C. Latombe, "Robot navigation for automatic model construction using safe regions," in *ISER '00: Experimental Robotics VII*. London, UK: Springer-Verlag, 2001, pp. 405–415.