Polygon-Based Tracking Framework in Surveillance Wireless Sensor Networks

Md. Zakirul Alam Bhuiyan[1], Guojun Wang[1,2,*], and Jie Wu[2]

[1]School of Information Science and Engineering
Central South University, Changsha 410083, China
[*]Corresponding author: csgjwang@mail.csu.edu.cn

[2]Department of Computer and Information Sciences
Temple University
Philadelphia, PA 19122, USA

Abstract

We propose a new tracking framework by organizing nodes into a polygonal spatial neighborhood in order to detect and track unauthorized traversals in surveillance wireless sensor networks. During a tracking, the neighborhood is further constructed ahead of the target traversal, and this features a timely forwarding with guaranteed delivery property. Instead of estimating future movement and position separately in a polygon, we find an original target tracking path in a graph, and create a brink on the graph called "critical region" by introducing a brink detection algorithm to know a target's route, and to also achieve reliable internode communications. In addition to the basic design, an optimal sensor selection algorithm was developed to select which sensors to query, and dynamically guide the target information to a sink. Simulation results validated that the proposed approach has better tracking accuracy, reduced localization error, and is robust to strong environment noise, while using a minimum number of sensors.

Index Terms - Brink construction, polygonal neighborhood, sensor selection, target localization, target tracking

1. Introduction

Wireless sensor networks (WSNs) have gained much attention in both public and research communities because they are expected to bring the interaction between humans, environments, and machines to a new paradigm. Despite being a fascinating topic, with a number of visions of a more intelligent world, there still exists a huge gap in the realizations of WSNs. The WSN was originally developed for military purposes in the battle field, however, the development of such networks has encouraged its use in healthcare, environmental industries, and monitoring, including event/target detection, localization, and tracking [10].

The objective of this paper is to design a target tracking application in WSNs. This application is able to detect and track a target, and report information about the target to the sink. Many protocols for different environments have been proposed in the literature for target tracking [21], [2], [18], [15], [20], [17].

Research about target tracking can be roughly divided into three categories: (1) tree-based scheme [20], (2) cluster-based scheme [17], and (3) prediction-based scheme [2], [18]. In addition to these three schemes, we are proposing a polygon-based target tracking framework in surveillance wireless sensor networks (PTT), which is to the best of our knowledge, the first of its kind. The concept of the scheme is inspired by geographic routing [8], [12], and face routing [11], in particular.

The idea of this tracking framework comes from a deliberate attempt to eliminate some of the drawbacks of existing tracking systems. In recent years, the use of computational and convex geometry for the application of network design has enormously increased. The concept of a planarized graph, such as the Voronoi diagram and the Delaunay triangulation, is mostly used in the network domain [13]. In case of the planarized graph, the partitioning of a plane with p points into polygonal regions such that each polygon contains some generated points, and all points inside a given polygon are connected. Those points are closer to each other, and every two points share a common edge.

The two points become *couple nodes*, from among all the points (neighbor nodes), through a rigorous selection process in the spatial neighborhood known as a *face*, in face routing [11]. The face could be of a different size and shape. For working with the shapes, it is simply called, *polygon*, in this paper. Although a node is enough to detect and generate target information in the polygonal region, we consider the couple nodes for reliability concern.

One of the main goals in this idea is to find edges in a polygon. When the target passes through an edge, the target is observed. For the sake of simplicity, we reconstruct the polygon conceptually, that is, when a target traverses the

polygon, it is momentarily considered as a "critical region". The edge of the polygon is called a *brink*. During the target movement, the brink between the couple nodes is a way of making a space around the target as it moves toward a given node. The space could be called a "follow spot," as it moves with a musician in a stage show. It is easy to think about the moving spotlight from the "space + time" point of view, where points of space, and instants of time, are distinct and absolutely present, but keep changing along the series of times from earlier times to later times. We assume the area of the follow spots is produced by the brinks. For the purpose of localizing the target in a timely fashion, we further consider two kinds of follow spots, namely, the square and rectangular. The square one can be produced at the time the target is first detected by the couple nodes. The rectangular one can be produced at the time the target is about to cross a brink, and the couple nodes relay a joint-message about the target to the next polygon. Thus, organizing the sensor nodes in the polygonal form is to prepare in advance, and track the target on time.

The major challenge in the system is to select appropriate sensor nodes as couple nodes closest to the target, which is due to numerous reasons: irregular signal patterns emitted from the target, environment noise, and irregularities, especially when a neighborhood size is too large or too small. By applying the optimal sensor selection algorithm obtained from the maximum information utility [4], which is important for a tracking system, we demonstrate that PTT has a lower detection error, and a better tracking accuracy.

In summary, the main contributions of this paper include the following: (1) we introduce a polygon sequence based target tracking method, inspired by the planarized algorithm, that does not rely on any global topology; (2) we develop a brink detection algorithm, which works well against the target missing, and allows the system to work timely. Besides, we tackle the localization error using a covariance bound; (3) In order to keep the number of working sensors to a minimum, we formulate an optimal sensor selection mechanism to choose the appropriate sensor near the target; (4) we evaluate the performance of the PTT, and compare the tracking accuracy with the existing literature.

The rest of the paper is structured as follows: In Section 2, we give our proposed system model. Section 3 illustrates the proposed polygon-based target tracking approach. Section 4 evaluates the design with extensive simulation. Section 5 concludes the paper.

2. System model

2.1. Network model

We consider a WSN composed of N nodes in a 2D square planar field, and sensor nodes are able to tune their range up to communication range r_c. Let $N(u) = \{v||uv| \leq r_c\}$ be the set of physical neighbors (within r_c) of node u, and *S* be a sink or actor in the network. Consequently, all $u \in V$, and $v \in V$ together define a unitdisk graph (UDG), which has an edge uv if and only if the Euclidean distance $||uv|| \leq 1$ (one unit). In order to track the target route, extracting planar graphs from graphs is needed to guarantee the information delivery before the target arrives at a region [5]. RNG and GG are examples of algorithms that create a planar graph [8], [13]. The main idea of both algorithms is that two nodes u and v from a planar graph, as shown in Figure 1, are within each other's communication range, if there is no other neighbor w called witness within their common area. We can obtain a connected planar subgraph $G' = (V, E')$ that maintains connectivity with fewer edges in both graphs. The planar subgraph is an closed *polygon*. Thus, a polygon in the plane, denoted by P_i, contains at least three nodes.

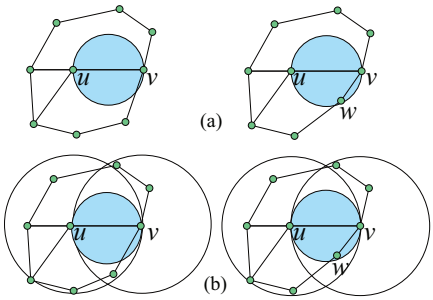

Figure 1. (a) GG and (b) RNG planar graph showing a witness *w*, which must not fall within the shaded circle

Note that the planarized algorithm assumes that all static wireless nodes have distinctive identities, and locally accessible information about their neighbors . By one-hop broadcasting, a node gathers the location information of all nodes within the transmission range, and its corresponding region in which the target is. Otherwise, it communicates with the other nodes through multihop wireless links by using intermediate nodes [16].

2.2. Distributed observation model

We model our sensor measurement problem using a standard estimation theory [9], [14]. In a certain polygon, success of detecting a target with the help of a sensor network, shows how efficiently the tracking task can be performed. There are many ways of target signal processing, for example, acoustic, seismic, and electromagnetic signals. In this tracking framework, we assume that all sensors are acoustic, measuring only the amplitude of the sound signal, such that the vector $\bar{x}$ is the unknown target position. The acoustic signal received at the i^{th} sensor, where $i = 1, 2, ...N$,

can be represented as:

$$S_i(t) = s_i(t) + \beta_i(t) \tag{1}$$

where the time interval is denoted by t, and the acoustic intensity measured at the i^{th} sensor due to a single vehicle is expressed as:

$$s_i(t) = \varsigma_i \sum \frac{a(t-\tau_i)}{d_{ij}} \tag{2}$$

The Euclidean distance between the i^{th} sensor and the target j is denoted by $d_{ij} = \parallel \bar{x}(t-\tau_i) - x_i) \parallel$. $\beta_i(t)$ carries a zero-mean additive white Gaussian (AWGN) noise random variable with a variance of the intensity of the vehicle σ_i^2. τ_i is the propagation delay of the acoustic signal from the target vehicle to the i^{th} sensor. x_i is a given $p \times 1$ vector, denoting the position vector of the i^{th} stationary sensor. ς_i is the sensor gain factor of the i^{th}acoustic sensor. T is the period between two consecutive discovery signals of the target. Let the time-dependent average energy measurements be denoted by $e_s(t)$ over the time interval, then we can make the following observation:

$$e_s(t) = S_i(t) + \varepsilon_i(t) \tag{3}$$

where $S_i(t)$, and $\varepsilon_i(t)$ are the signal, and noise energy, respectively. The background noise $\beta_i(t)$ has a χ^2 distribution, with the mean equals to σ_i^2, and the variance equals to $2\sigma_i^2/M$. M can be larger, for example, 40. Hence, the energy measured at the i^{th} sensor, $\varepsilon_i(t)$ can be approximately well with a Gaussian distribution, for example, $\varepsilon_{\mathrm{i}}(\mathrm{t}) \sim \mathrm{N}(\sigma_{\mathrm{i}}^2, \frac{2\sigma_{\mathrm{i}}^4}{M})$.

2.3. Localization error

In a real scenario, observations of acoustic signal processes are disturbed due to obstacles and noise. Naturally, we seek to find estimates of location with the smallest error c_v. When a node has location uncertainty, we characterize localization accuracy using a covariance bound that is similar to the formulation of the *Cramer-rao lower bound (CRLB)* of the variance [7]. To estimate the target position, it is necessary to define a measure of information utility, denoted by $\phi(\cdot)$. *CRLB* is defined as the inverse of the *Fisher information matrix* (FIM) [4], [7]. We can derive the utility based on the covariance c_v of the distribution. The determinant, $\det(c_v)$, is proportional to the volume of the region [4]. Hence, the information utility function for this approximation can be chosen as $\phi(\cdot) = -\det(c_v)$. The FIM for target detection is calculated as:

$$\begin{aligned} J &= -E\{[\tfrac{d}{d\bar{x}}(\tfrac{d}{d\bar{x}} \ln e_s(t)|\bar{x}])^T\} \\ &= E\{[\tfrac{d}{d\bar{x}}(e_s(t)|\bar{x}][\tfrac{d}{d\bar{x}}(e_s(t)|\bar{x}]^T\} \end{aligned} \tag{4}$$

where *E* is the expected value. According to target location coordinates, we have

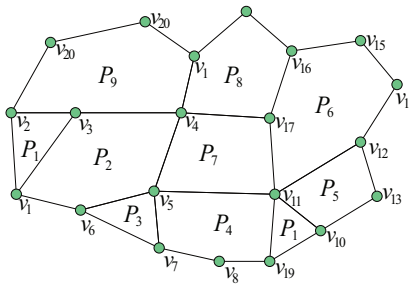

Figure 2. An example of the sensor network, demonstrating a polygon shaped neighborhood

$$J = \sum_{i=1}^{N} \frac{1}{2} E\{[\frac{d}{d\bar{x}}(e_s(t)|\bar{x}][\frac{d}{d\bar{x}}(e_s(t)|\bar{x}]^T\} \tag{5}$$

J^{-1} is the estimation error covariance matrix, which defines CRLB of the target localization error. The lower bound of the variance of the target location estimates can be expressed as:

$$c_v \geq J^{-1} \tag{6}$$

3. Polygon-based target tracking (PTT)

3.1. Localized polygon

In order to describe the polygonal traversal problem in the proposed PTT scheme, we can see an example of the polygon construction shown in Figure 2, with respect to Figure 1. We use polygons to describe the target moving path. The polygon is not necessarily a convex, but it must not be self-overlapping. Let a set of nodes in a polygon be $N_p = (v_1, v_2, ..., v_p)$, where $p \geq 3$. We illustrate with an example. Suppose a target is presently in P_2, then P_2 is called *active polygon* (P_c), and v_5 is an *active node* in P_2. In Figure 2, P_1 is a triangle, P_2 is a pentagon, and P_7 is a tetragon. Node v_5 in P_2 is aware of i) its own position, ii) the position of its active polygon neighbors v_6, v_1, v_3, and v_4, iii) the position of its adjacent neighbors v_4, v_{11}, v_7, and v_6, and iv) the position of the neighbors in adjacent P_2, P_3, P_4, and P_7 after deployment through intermediate nodes. Thus, v_5 stores information about 4 polygons that are adjacent to it in G-$\{v_5, v_4, v_{17}, v_{11}\}$,$\{v_5, v_{11}, v_{19}, v_8, v_7\}$, $\{v_5, v_7, v_6\}$, and $\{v_5, v_6, v_1, v_3, v_4\}$. Suppose the target is moving toward polygon P_7, then it is called a *forward polygon* (P_f). v_5's adjacent neighbors corresponding to the active polygon, with respect to the target position, are called *immediate neighbors*. Thus, node v_5 can have only two immediate neighbors that are v_4 and v_6 from the four adjacent neighbors in G. Either v_4 or v_6 becomes active as the target crosses edge v_5v_4, or edge v_5v_6. Suppose the target travels toward polygon P_7, it crosses v_5v_4, thus, we call v_5 and v_4 as *couple nodes* (N_c). The process of selecting the couple

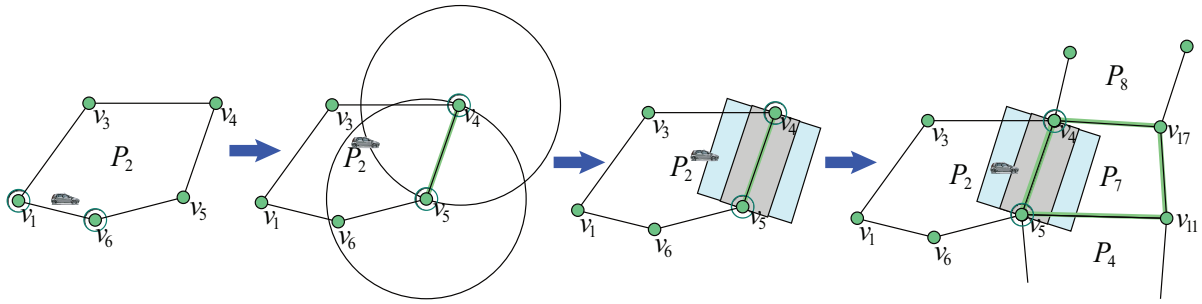

Figure 3. A simple scenario of the brink construction process

nodes is described in the later section. All the v_5's neighbor nodes in P_2, are called *polygon neighbors* N_a. The working area of v_5 covers all the edges between the adjacent neighbors, and itself. Thus, a node corresponds to a number of polygons (P_i). The size of a polygon is defined by the number of edges surrounding the polygon. The average size of a polygon is $\bar{P} \leq 2v_i/(v_i - e_i + 2)$, where v_i and e_i are the numbers of nodes and edges of a polygon, respectively. The relation between nodes, edges, and polygons is given as $p_i + v_i - e_i = 2$, where p_i is the number of polygons, according to Euler's formula [1]. This suggests that our scheme has cells for a planar graph, with as many edges as possible.

We need to mention an underlying issue of the polygonal region: When a target crosses over a polygon, not only the closed polygon is covered, but an extended area is also covered by the region according to each sensor's working area. In the example, the number of sensors is the same. The extended area can be effectively used if a target missing event has occurred. If any sensor node corresponding to the polygon can detect the target in the outside of the polygons, the polygon in the outside becomes the active polygon, and the sensor sends a message to the previous polygon. One of the main reasons or advantages of considering couple nodes in the polygons P_c and P_f, is to minimize the number of participating sensors in target tracking. For example, v_5 has 4 adjacent polygons with 9 neighbor nodes, and node v_4 has 4 adjacent polygons with 12 neighbor nodes. But, edge $e(v_5v_4)$ between the couple nodes has only 2 polygons with 5 neighbors. Thus, the number of sensor nodes can be minimized.

3.2. Brink construction

In this section, we introduce an edge construction algorithm, which is used to draw another conceptual polygon as a *critical region* by connecting an edge called a *brink* to the active polygon P_c. Now, the PTT problem turns into a critical region problem. Our goal is to find the brink, while the target is moving between the couple nodes in the network, that confirms the target leaving one polygon to another, which could work well against the target missing, and allow the system to work timely. We assume that some nodes may be damaged, or faulty in the network. Recently, Zhong et al [21] proposed a tracking method that converts the tracking problem into finding the shortest path in a graph, which is equivalent to optimal path matching (PM). However, the PM is based on node sequences through geographic face traversal, while the PTT is directly based on a dynamic moving path through a polygon sequence/traversal. Although we do not convert the tracking path into an optimal path, we reconstruct the polygon by using a brink; the tracking path should be created through the brink. The brink is constructed by the couple nodes that makes the polygon sequences. Nonetheless, the polygon sequence could be more effective in this tracking framework in terms of node failure/damage.

The couple nodes relay a joint-message to P_f that a target is approaching. However, the brink is an intersection of the two polygons. Before the target crosses over the brink, all the nodes in P_f receive the message in advance, and prepare to detect the target.

According to the properties of the localized polygon, after receiving the target discovery message in the forward polygon P_f, the edge of the forward polygon is mapped by the brink before the target arrives in the P_f. Thus, the target is in current P_c, meaning it is focused in the "follow spot". Each brink in the P_f has to be identified during the target's crossing over, as shown in Figure 3. When the target enters the spot, the couple nodes are aware of it. The spot is divided into a two-phase detection spot, namely, square spot, and rectangular spot. Here, we consider the brink to be mapped over the *X*-axis, as shown in Figure 4. Let D be the length of the brink, and i and k be the couple nodes, respectively. We suppose $D \propto d_{ik}$ and $\frac{D}{2} \leq r_s$. D is achieved from $(-D/2)$ to $(D/2)$. $D \leq 2r_s$, is a length of both square and rectangular spots. A $= D^2$ is for the total square spot, and A $= \frac{D}{2} \times D$ is for the total rectangular spot. We suppose the target is traveling through the square spot. Thus, when it touches the rectangular spot, a joint-message is broadcast to P_f. When the target leaves for P_f, P_f becomes the new active P_c, and the previous one becomes as inactive as normal polygon P_i. Besides, when the target moves away from the rectangular spot in P_f, all the

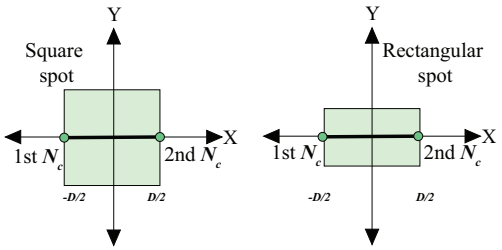

Figure 4. X-axis showing the brink crossing

brinks in the previous P_c should be removed. Variability of different parameters of the brink, such as brink length, local mean length, and local standard deviation, allow the couple nodes to identify the brink easier.

Let ρ and $\rho^{'}$ be the detection probability for the 2 phases, respectively, by a closest sensor that is a couple node to the target. These can be expressed as:

$$\begin{aligned} \rho &= \tfrac{1}{A} \int_{-D/2}^{D/2} e_s(N_c, j)dx \int_{-D/2}^{D/2} dy \\ \rho^{'} &= \tfrac{1}{A} \int_{-D/2}^{D/2} e_s(N_c, j)dx \int_{-D/4}^{D/4} dy \end{aligned} \tag{7}$$

Note that the values of ρ and $\rho^{'}$ completely rely on the length of the brink.

3.3. Optimal node selection algorithm (N_o)

The tracking of the target requires an optimal number of sensors in the network to aggregate data among the nodes. However, among the available sensors in the network, not all sensors provide useful information that improves the estimation. Furthermore, some information might be useful, but redundant. In this PTT scheme, we offer an optimal selection mechanism to choose the appropriate sensors.

After the brink is formed, using our observation model (Section 2.1) between the couple nodes, the nodes query and send a message to all the neighbors (N_a) corresponding to the forward polygon. The message contains the estimation of the target, and its own information. While receiving the message, each N_a combines its own measures of the target with the couple nodes' estimation, to compute its weight, whether it is about to be one of the new couple nodes using an optimal selection function, and then responds to the previous couple nodes by a bid [(e.g., ID, d_{ij}, etc.)]. When a node detects the target, it sends the bid to its immediate neighbors. It also receives a similar bid from the neighbors if both of its immediate neighbors detect the target, which then evaluates the received bids, and ranks them according to the weight of the bids. Then, it compares the weight of the bids with its own bid, and ranks them. It locally decides whether it should join in tracking the target, or withdraw itself from the tracking. If it is with the best weight, it can easily determine its couple node from the rank. In this way, we can select the best nodes closest to the target as the couple nodes with the best data. We use the optimal selection function as a mixture of both information usefulness, and energy cost [4]. Suppose the number of optimal nodes is $N_o(N_o \in N_p)$. The selection function is represented as:

$$\begin{aligned} \psi(\delta(\bar{x}|N_a, N_c)) = \alpha * \lambda_{use}(\delta(\bar{x}|N_a, N_c)) \\ - (1-\alpha) * \gamma_{\cos}(N_a, N_c) \end{aligned} \tag{8}$$

We describe the function as follows:

1. $\delta(x|N_a, N_c)$ is the estimate of the target, formed by each node and polygon neighbors.
2. $\lambda_{use}(\delta(x|N_d, N_a))$ is the information usefulness measure function given as:

$$\begin{aligned} \lambda_{use}(\delta(x|N_d, N_a)) &= \lambda_{use}(x_i, \bar{x}) \\ &= (x_i - x)^T c_v \end{aligned} \tag{9}$$

3. $\gamma_{\cos t}(N_a, N_c)$ is a function that refers the energy cost of communications between N_a, and previous N_c, thus, the geometric measure of the function is given as:

$$\gamma_{\cos}(N_a, N_c) = (x_i - x_c)^T (x_i - x_c) \tag{10}$$

4. α is the relative weight of the usefulness and cost.

Finally, the selection function (9) can be reduced by substituting (10) and (11) as follows:

$$\begin{aligned} \psi(\delta(x_c, x_i, \bar{x}) = \alpha * (x_i - x)^T c_v - (1-\alpha) \\ *(x_i - x_c)^T (x_i - x_c) \end{aligned} \tag{11}$$

A sensor node becomes one of the optimal nodes, based on the weight denoted by threshold N_{th} of the selection function. If $N_{th} < 1$, all the bids are chosen. However, we accept $N_o = 2$ for this tracking framework, where $N_o = mod(N_{th})$, i.e., selecting N_c. It is expected that the optimal node number should be chosen to be no more than the number of N_p. According to the different tracking tasks, the number of N_o can be changed through the sink broadcasting a message containing N_o, to the sensor networks. Although, the optimal number can be more, depending on the system demand $(N_o \leq N_p)$. Nevertheless, the optimal selection is very important, which not only impacts the accuracy of the tracking, but also the energy efficiency of the system, thus it impacts the lifetime of the whole network.

3.4. Tracking with polygon sequence

This section discusses the tracking process in the PTT scheme. The process is illustrated in Figure 4. In this polygon based framework for target tracking, a node has full polygon information after the network initialization, modeled in Section 3.1, and can estimate the cost of communication to the neighbor nodes. Initially, all the nodes in the sensor network are in the power-saving mode, waking up at a predefined period, and carrying out the sensing for a short time. However, in this framework, we presume a sensor node has three different states of operation, i.e., *active*,

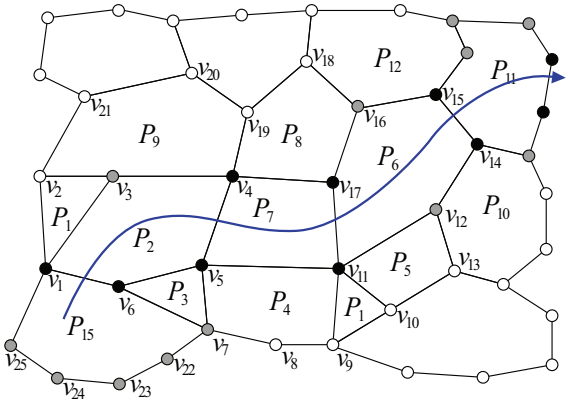

Figure 5. Using the polygon sequence to track the target

awaking, and *inactive* [2]. We consider that a sensor should be kept awake, so long as its participation is needed in a given task [3].

Once some nodes detect the target in a polygon P_i, the P_i becomes the active polygon P_c. At first, the sensor nodes that detect the target are required to broadcast this information. Whenever the couple nodes N_c are selected using the optimal selection algorithm, the detection probability ρ, or $\rho^{'}$, confirms that the target is about to cross the square or the rectangle phase. A join-request message is sent at the moment the couple nodes touch the rectangular phase to P_f, to join in target tracking. Before the target leaves the rectangular phase, the forward polygon is named by P_f. All the neighboring nodes in P_f receive the request message, and change their state to the *awaking* state, and then start sensing. When the target crosses the brink, another join-request message is sent to the neighbors in the previous polygon. After receiving the message, all the neighbors, except the previous couple nodes, return to the *inactive* state.

The couple nodes measure the difference in distance d_{ij} between two consecutive sensing results. All the results are measured by reducing CRLB covariance to achieve less error localization. Since the target travels across the square spot, and then the rectangular spot, accordingly, the distance is decreased, and the nodes are aware of it. If the target is moving away, the nodes are also aware that the distance is increased. If the target leaves the square spot for the same P_c, the couple nodes broadcast a message instantly in its route in the polygon P_c. Figure 5 illustrates the target moving path in the sensor network. The target is initially detected by sensor v_1 and v_6 (black shaded indicating the active nodes) in the polygon P_{15}, the rest of the corresponding nodes (grey shaded) in P_{15} are in the *awaking* state, and the rest of the nodes in the sensor network are in the *inactive* state when the target is in P_{15}. The target travels through the polygons. A sequence can be $P_{15} \rightarrow P_2 \rightarrow P_7 \rightarrow P_6 \rightarrow P_{11}$, and so on.

4. Simulation studies

We evaluated the performance of the PTT scheme via simulation. We have implemented it on the OMNet++ v3.3p1 simulation environment using Castalia simulator (http://castalia.npc.nicta.com.au/index.php). Our goals in conducting the simulation are as follows: (1) Studying the localization errors in different parameter settings, while considering the polygonal region, comparing the tracking accuracy of the PTT with existing protocols, while considering a noisy environment, (2) evaluating the observation model, (3) examining the brink construction probability, (4) finding the effect of the sensing noise, and finally (5) observing the performance when the number of sensor nodes is increased.

4.1. Simulation settings

The simulation has been performed within a $400m \times 400m$ square surveillance field in an area of interest (AoI). For simplicity, the position of the *N* (200) sensors is randomly uniform in a 2D square planar field hereafter. Thus, the sensor density should be large enough so that for any arbitrary sensing region, there are at least 2 sensors, which can monitor the region at a time. Throughout the simulation, any two sensor nodes can directly communicate via bi-directional wireless links, and their Euclidean distance is not greater than the communication range $(d_{ik} < r_c)$, and a target's position in the plane can be perfectly monitored by a sensor node if their Euclidean distance is also not greater than the sensing range $(d_{ij} < r_s)$.

Instead of considering all the possible combinations of r_c and r_s, we focus on the case of $r_c \geq 2r_s$ in the simulation. This specification of r_c and r_s holds for most commercially available sensors, such as, Berkeley Motes and Pyroelectric infrared sensors [19]. At the beginning of the simulation, the target shows up at a random position of the sensing area with the maximum acceleration $a_{\min}$= 5 m/s, and the maximum velocity $v_{\max}$ = 25 m/s.

4.2. Simulation results

1. Study of the localization error: We have performed the simulation to examine the target localization error. There is one sensor approximately in every $20 \times 20m^2$ sensor field. The predefined target locations are further perturbed randomly during the simulation over a square meter area. The background noise parameter is $\sigma = 0.5$. Four different sensor densities, i.e., 8, 12, 15, and 18 have been used. The mean and variance are computed. From simulation results, as shown in Figure 6(a), we observed that the mean values of these methods do not show any statistically significant bias, and hence, yield unbiased estimates.

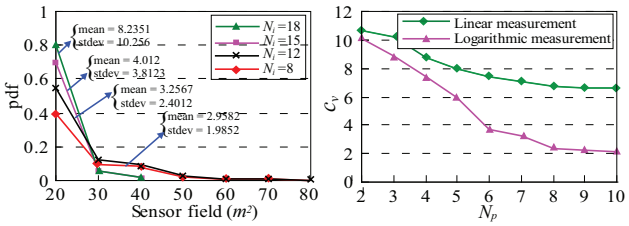

Figure 6. Average localization errors (a) based on the number of sensors, and (b) covariance based on the different measurements

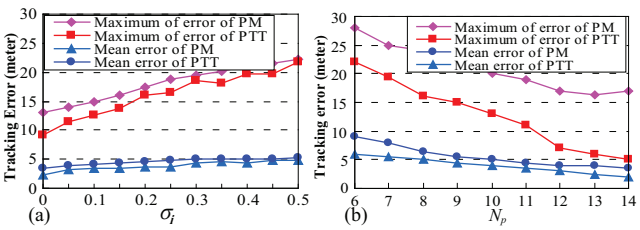

Figure 7. The effect of sensing noise to tracking errors

The histograms of the magnitudes of the localization error are plotted in Figure 6(a). Since the histogram can be regarded as an approximation of the probability density function (pdf), the mean and standard deviation of the magnitude of the localization errors are calculated.

2. Study of the sensor measurements: Figure 6(b) shows a plot of the number of sensors incorporated (N_p) versus the logarithm of the determinant of the error covariance (c_v) of the measurement model, and with the linear measurement model. Indeed, the error volume (covariance) under sensor selection criterion, using the logarithmic measurement model, is less than the error volume under the selection of the linear measurement model for the same number of sensors, except during the initial phase, or after all the sensors have been accounted for. The logarithmic measurement model shows a good level of accuracy.

3. Study of the sensing noise: We compare the tracking performance of the proposed PTT based on the dynamic moving path, with the optimal path matching (PM). The mean tracking error rate is defined in the PTT as an averaged error rate of all the nodes in the polygon, which is gathered from 100 simulation runs. The tracking accuracy is enhanced with an increasing number of sensor nodes. Figure 7(a) depicts the performance of different σ_i for the linear noise (Equation 3). It illustrates that the noise brings in some tracking error. PTT relatively decreases the noise compared to PM.

4. Study of the number of sensor nodes: We compare the PTT with PM under a different number of sensor nodes. Figure 7(b) shows that when the number of sensor nodes in a polygonal area increases, the tracking errors are lessened, and the polygon based tracking path has better performance compared to ML.

5. Study of the brink construction: In this set of simulation runs, we study the number of sensors in a polygon neighborhood, designed by the proposed tracking framework considering neighborhood size and brink length. Figure 8 (a) shows the divergence of the value of D and r_s as time goes on. The value of D relies on the value of r_s. Figure 8 (b) depicts the average length of all the brinks, and the average length of the brink in a polygon neighborhood. It is clearly seen that the length can be different on the basis of sensors deployment.

An observation is made in Figure 8(c) about the probability of the brink construction vs. polygonal area length, covered by the square and rectangular spot. In case of the target detection, the spot size can be a variant value. From the results, as shown in Figures 8, we can easily understand the probabilistic improvement, or degradation of detecting any target over a brink. The active polygon is only active upon the X-axis, so it totally depends on the length of the brink, with some specified parameters. The simulation results indicated that the target tracking greatly depends on the probability of the producing area.

5. Conclusion

We proposed a novel approach for mobile target tracking through polygonal neighborhood in wireless sensor networks. Tracking was modeled as a polygonal traversal problem in a graph. Besides the basic design, non-linear sensor measurements, and brink construction method were proposed for further enhancement of the system's accuracy. Meanwhile, we accomplished the sensor node selection that chooses a number of sensors optimally, without degrading the performance of the system. Evaluation results demonstrated that this tracking framework remarkably estimates a target's positioning area, and tolerates the bounded location errors. In addition, this work provided a general idea of how to follow an entity by creating a special area like "follow spot" when the entity enters or leaves a surveillance. Investigating the impact of target missing when there is a node failure, achieving a good trade-off between energy conservation and tracking quality in the proposed scheme, are our future work.

6 Acknowledgment

This work is supported by the Hunan Provincial Natural Science Foundation of China for Distinguished Young Scholars under Grant No. 07JJ1010, the National Natural Science Foundation of China for Major Research Plan under Grant No. 90718034, and the Program for Changjiang

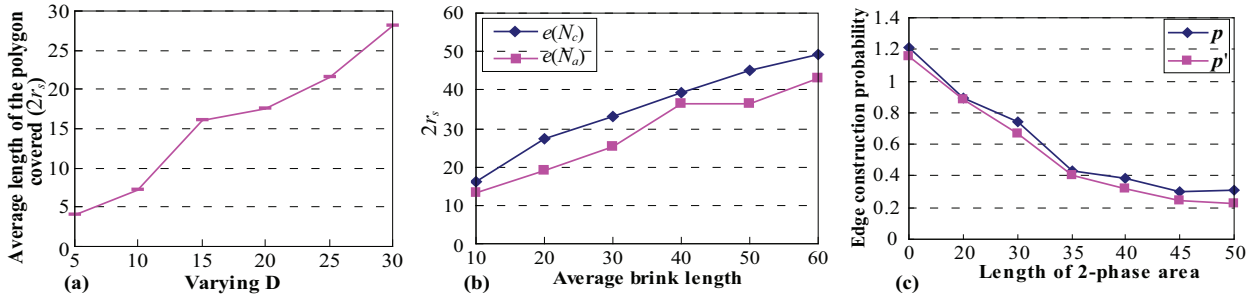

Figure 8. (a) Average brink length, (b) average area size, and (c) average length of the polygonal area of the portion of the target path covered by the couple nodes

Scholars and Innovative Research Team in University under Grant No. IRT0661.

References

[1] M. D. Berg, M. V. Kerveid, M. Overmars, and O. Schwarzkof. *Computational Geometry*. Springer, 1998.

[2] M. Z. A. Bhuiyan, G. Wang, and J. Wu. Target tracking with monitor and backup sensors in wireless sensor networks. *Proceedings of the 18th IEEE International Conference on Computer Communications and Networks (ICCCN 2009)*, San Francisco, CA, USA, 2009.

[3] Q. M. Chaudhari and E. Serpedin. A simple algorithm for clock synchronization in wireless sensor networks. *Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2007)*, pages 1–4, 2007.

[4] M. Chu, H. Haussecker, and F. Zhao. Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks. *International Journal of High Performance Computing Applications*, 16(3): 293–313, 2002.

[5] S. Datta, I. Stojmenovic, and J. Wu. Internal node and shortcut based routing with guaranteed delivery in wireless networks. *Cluster Computing*, Kluwer Academic Publishers, 5(2): 169–178, 2002.

[6] F. Hu, X. Cao, and C. May. Optimized scheduling for data aggregation in wireless sensor networks. *Proceedings of IEEE International Conference on Networking, Sensing and Control (ICNSC 2005)*, pages 557–561, 2005.

[7] M. Karan and R. Lobbia. Cramer-rao lower bound for single target tracking accuracy with coordinated turn maneuvers using range and bearing measurements. *Proceedings of the Sixth International Conference of Information Fusion (Fusion 2003)*, pages 911–918, Queensland, Australia, 2003.

[8] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker. Geographic routing made practical. *Proceedings of the Symposium on Network System Design and Implementation, USENIX Association*, pages 217–230, Berkeley, USA, 2005.

[9] L. E. Kinsler. *Fundamentals of Acoustics*. John Wiley and Sons Inc., 1982.

[10] M. Kuorilehto, M. Hannikainen, and T. D. Hamalainen. A survey of application distribution in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2005(5): 774–788, 2005.

[11] B. Leong, S. Mitra, and B. Liskov. Path vector face routing: Geographic routing with local face information. *Proceedings of 13th IEEE International Conference on Network Protocols (ICNP 2005)*, pages 147–158, 2005.

[12] C. Liu and J. Wu. Virtual-force-based geometric routing protocol in MANETs. *IEEE Transactions on Parallel and Distributed Systems*, 20(4): 433–445, 2009.

[13] M. A. Rajan, M. G. Chandra, L. C. Reddy, and P. Hiremath. Concepts of graph theory relevant to ad-hoc networks. *International Journal of Computers, Communications and Control*, 3(2008): 465–469, 2008.

[14] X. Sheng and Y.-H. Hu. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Transactions on Signal Processing*, 53(1): 44–53, 2005.

[15] A. Smith, H. Balakrishnan, M. Goraczkoet, and N. Priyantha. Tracking moving devices with the cricket location system. *Proceedings of the 2nd ACM International Conference on Mobile Systems (MobiSys 2004)*, pages 190–202, 2004.

[16] R. Stoleru, J. A. Stankovic, and S. Son. Robust node localization for wireless sensor networks. *Proceedings of the 4th workshop on Embedded networked sensors (EmNets 2007)*, pages 48–52, 2007.

[17] S. Suganya. A cluster-based approach for collaborative target tracking in wireless sensor networks. *Proceedings of the 1st International Conference on Emerging Trends in Engineering and Technology (ICETET 2008)*, pages 276–281, 2008.

[18] G. Wang, M. Z. A. Bhuiyan, and L. Zhang. Two-level cooperative and energy-efficient tracking algorithm in wireless sensor networks. *Wiley's Concurrency and Computation: Practice & Experience*, DOI: 10.1002/cpe.1503, September 2009.

[19] H. Zhang and J. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Wireless Ad Hoc and Sensor Network*, 1(1-2): 89–123, 2006.

[20] W. Zhang and G. Cao. Dynamic convoy tree-based collaboration for target tracking in sensor networks. *IEEE Transactions on Wireless Communications*, 3(5): 1689–1701, 2004.

[21] Z. Zhong, T. Zhu, D. Wang, and T. He. Tracking with unreliable node sequence. *Proceedings the 28th IEEE Conference on Computer Communications (INFOCOM 2009)*, Rio de Janeiro, Brazil, 2009.