Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 07-002

MCTA: Target Tracking Algorithm based on Minimal Contour in Wireless Sensor Networks

Jaehoon Jeong, Taehyun Hwang, Tian He, and David Du

January 26, 2007

$MCTA$: Target Tracking Algorithm based on Minimal Contour in Wireless Sensor Networks

Jaehoon Jeong, Taehyun Hwang, Tian He, and David Du
Department of Computer Science & Engineering
University of Minnesota
Minneapolis, MN 55455
Email: {jjeong,thwang,tianhe,du}@cs.umn.edu

***Abstract*—This paper proposes a minimal contour tracking algorithm ($MCTA$) that reduces energy consumption for tracking mobile targets in wireless sensor networks in terms of sensing and communication energy consumption. $MCTA$ conserves energy by letting only a minimum number of sensor nodes participate in communication and perform sensing for target tracking. $MCTA$ uses the minimal tracking area based on the vehicular kinematics. The modeling of target's kinematics allows for pruning out part of the tracking area that cannot be mechanically visited by the mobile target within scheduled time. So, $MCTA$ sends the tracking area information to only the sensor nodes within minimal tracking area and wakes them up. Compared to the legacy scheme which uses circle-based tracking area, our proposed scheme uses less number of sensors for tracking in both communication and sensing without target missing. Through simulation, we show that $MCTA$ outperforms the circle-based scheme with about 60% energy saving under certain ideal situations.**

***Index Terms*—Sensor Network, Target Tracking, Energy, Tracking Area, Mobile Target, Vehicle, Kinematics, Circle, Contour, Sensing, Communication, Optimization, and Minimization.**

I. INTRODUCTION

The energy efficiency is one of the important research issues in wireless sensor networks since it determines the lifetime of the sensor network deployed for the intended applications, such as environmental monitoring, area surveillance, and target tracking. Especially, in the target tracking application, the energy efficiency is the most important factor as it leads to the long-lived target tracking. In the target tracking setting, an energy-aware target tracking algorithm not only should guarantee the tracking of mobile targets (e.g., enemy tanks or vehicles), but also should maximize the sensor network lifetime using a minimum number of working sensor nodes. The *tracking area* is defined as the possible region where the mobile target can reach from its current position during some limited time. The legacy tracking scheme [1], [2] uses the circle-based tracking area for simplicity. Since the mobile target, such as vehicle, moves according to its vehicular kinematics [3], it is impossible for it to reach all the area of the tracking circle. We found that we can reduce the number of working sensor nodes in each tracking area if we use the vehicular kinematics that the mobile target moves according to. We try to prune out from the tracking circle the most unlikely region that the target cannot visit during some limited time. This makes the tracking area be a minimal-sized area based on the vehicular kinematics. Only the sensors within the minimal tracking area work for target tracking during some limited time. Thus, by updating the minimal tracking area containing the mobile target during the target's trajectory, the sensor network based on our scheme consumes less energy than the legacy scheme based on tracking circle. We call our minimal tracking area the *minimal contour*.

Our contributions in this paper are as follows:

- The modeling of tracking area based on the vehicular kinematics. Our tracking contour based on the vehicular kinematics is used to select a minimum set of working sensor nodes for some moment along with the target's trajectory.
- The optimization of tracking contour. We optimize the tracking contour in terms of energy cost by adjusting the lifetime of each contour according to the target's current speed.
- The minimization of communication energy consumption. We use both transmission power control and directional antenna to minimize the number of sensors that receive the tracking contour information and perform sensing.
- The considerations on measurement errors for mobile target's movement. Since the measurements can have some errors for vehicle's current position, speed, and direction, these measurement errors are considered to make a reasonably larger contour with some confidence interval.

The rest of this paper is organized as follows: Section II describes the problem formulation for mobile target tracking. Section III explains the minimal contour tracking algorithm. This section includes the modeling of tracking contour, the optimization of tracking contour, the minimization of communication energy consumption, and the measurement error handling. In Section IV, we discuss the implementation issue for our target tracking algorithm. In Section V, we show that our contour scheme outperforms the legacy scheme based on tracking circle through simulation. In Section VI, we compare our work with the related works. We summarize our work and shed our future work in Section VII.

II. PROBLEM FORMULATION

We propose an energy-aware target tracking algorithm based on tracking contour in order to maximize the lifetime of the sensor network. The tracking contour is constructed based on the vehicular kinematics, which allows a minimal number of sensors near to the target to work in both communication and sensing.

A. Assumptions and Definitions

We have a few assumptions as follows:

- The sensing range is a uniform-disk whose radius is r.
- The communication radius is adjustable by controlling RF transmission power [11], [12].
- The RF transmission angle is adjustable by using directional antenna [13]–[15].
- The localization scheme is provided for the sensor nodes in order to find the position, speed, and direction of the vehicle at any time [4], [5].

We define four terms as follows:

Definition 1. *Refresh Time***.** We define the lifetime of of the tracking area as *refresh time*. The old tracking area is replaced with the new tracking area according to the target's movement *every refresh time*.

Definition 2. *Tracking Circle***.** The *tracking circle* is the tracking area where the target can visit for its current position and speed during refresh time. The tracking circle's radius is the multiplication of target's *speed* and *refresh time*.

Definition 3. *Tracking Contour***.** The *tracking contour* is the tracking area where the target can visit for its current position, speed and direction during refresh time, considering the vehicular kinematics. It prunes out the most unlikely area from the *tracking circle*.

Definition 4. *Minimal Contour***.** The *minimal contour* is a tracking contour for a given target's speed that allows for the minimization of energy cost spent for target tracking.

B. Main Idea

Our main idea is to minimize the tracking area used to determine the neighboring sensors that participate in target tracking. The legacy scheme always uses a *tracking circle* surrounding the mobile target that is modeled as a *random walk*. Though this approach is simple, more than a half of the tracking area based on circle cannot be visited by the target within some limited time [3]. Our scheme uses the vehicular kinematics to prune out the most unlikely area where the target cannot visit within such small time. Our tracking contour's shape changes from a circle to a contour (e.g., cone-like shape) according to the target's movement state (i.e., stopping state and moving state). Our model for tracking contour is represented as a *polygon* approximately including the area where the target can reach during refresh time based on the vehicular kinematics. Figure 1 shows two tracking areas: (a) Tracking Circle and (b) Tracking Contour. Let $p = (x, y)$ be the target's position vector where x is x-coordinate and y is y-coordinate. Let $m = (v, \theta)$ be the target's movement vector where v is the target's speed and θ is its direction. We can see that the contour's area is always the subset of the circle's area. So, the contour can allow fewer sensor nodes to track the target than the circle; that is, only the sensor nodes within the contour whose area is smaller than the circle's perform sensing work, leading to energy saving.

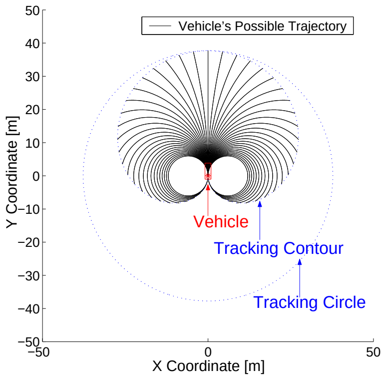


Fig. 1. Tracking Area: Tracking Contour versus Tracking Circle

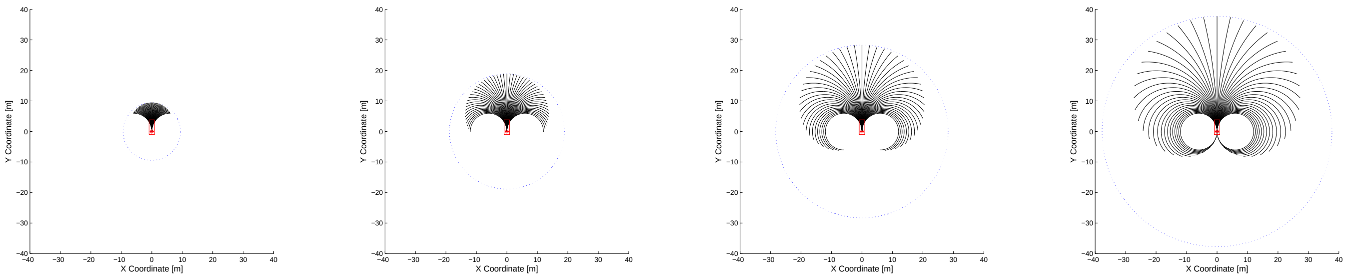
Figure 2 shows the possible trajectories of the vehicle according to *refresh time* where a new contour is generated for tracking every refresh time. Let *one turning time* be the time that is needed for the vehicle whose speed is v and whose turning angle is its maximum steering angle ϕ. Figure 2(a), Figure 2(b), Figure 2(c), and Figure 2(d) show the tracking area for $\frac{1}{4}$ turning time, that of $\frac{2}{4}$ turning time, that of $\frac{3}{4}$ turning time, and that of one turning time, respectively. The outer circle in each figure indicates the tracking area predicted by the legacy scheme based on circle. Thus, the tracking area is determined with *refresh time*, *vehicle speed*, and *turning angle*. Thus, since only sensor nodes which belong to the tracking contour smaller than the tracking circle need to turn on their sensing and communication devices, our scheme based on tracking contour can save more energy than the legacy scheme based on circle.

C. Design Goals

We have three design goals to minimize the energy consumption for target tracking: (a) the optimization of *refresh time* for minimal contour, (b) the minimization of *communication cost* in terms of the number of RF receiving sensors, and (c) each sensor's localized determination of its *warming-up time* and *finishing time* for sensing.

The *refresh time* determines the size of contour given the target's speed; that is, the bigger the refresh time is, the bigger the contour is. We need to use the optimal refresh time that leads to the minimal energy consumption for target tracking. This refresh time is selected as an optimal time, considering all the energy costs for tracking, such as communication cost, computation cost, and sensing cost.

The *RF transmission power control* and *directional antenna technology* are adapted for reducing the communication cost. Because the receiving power consumption is dominant factor in energy cost, we should reduce it. The RF transmission



(a) $\frac{1}{4}$ Turning Time (b) $\frac{2}{4}$ Turning Time (c) $\frac{3}{4}$ Turning Time (d) One Turning Time

Fig. 2. Tracking Contour's Shape according to Refresh Time

power control and directional antenna technology allow to save receiving power consumption.

When the sensor nodes turn on and turn off their sensing devices can be decided locally in order to save their energy with the target's movement information (i.e., position and speed), refresh time and their own position. Refer to Section III-F for details.

With the given number of sensor nodes, our objective is to maximize the sensor network lifetime to satisfy the following conditions:

- to guarantee the target tracking without missing and
- to use the minimal contour appropriate for the target's speed in terms of the energy cost in both communication and sensing.

III. MINIMAL CONTOUR TRACKING ALGORITHM (MCTA)

Assume that the sensor detecting the target can know the position and speed of the mobile target through the target localization scheme [4], [5]. We define a sensor node disseminating the tracking contour information as *root node*. When a sensor plays a role of root node, it broadcasts the movement information of a mobile target.

Algorithm 1 $Perform\_Tracking(contour\_info)$

```
$(t, p, v, \theta) \leftarrow Decapsulate\_Contour\_Information(contour\_info)$
{decapsulate the contour_info into the target's movement information}
$\Delta T \leftarrow Lookup\_Optimal\_Refresh\_time(v)$
{get the optimal refresh time from a look-up table}
$S \leftarrow Compute\_Minimal\_Contour\_Region(p, v, \theta, \Delta T)$
{compute the minimal contour's region with the minimal contour information sent from the root node with the contour's center position $p$, the target's speed $v$, the target's direction angle $\theta$, and the optimal refresh time $\Delta T$.}
$my\_position \leftarrow Get\_My\_Position()$
{$my\_position$ contains the coordinate of the this sensor node $(x, y)$}
$flag \leftarrow Am\_I\_Inside\_Minimal\_Contour(S, my\_position)$
if flag = TRUE then
    $Start\_Sensing(t)$
    {this sensor node warms up its sensing devices for sensing}
    $Rebroadcast(contour\_info)$
    {rebroadcast the new contour's information to neighbor sensor nodes}
end if
```

When the sensor node receives the broadcasted message containing the minimal contour information, it determines whether it belongs to the minimal contour or not. If the sensor is the member of the new contour, it warms up its sensing devices to prepare for the target tracking and relays the message to its neighbor sensor nodes. Otherwise, it just relays the message to its neighbors.

This section is organized as follows: Section III-A describes the vehicle's kinematics and formulates the motion process of the vehicle. Section III-B explains the modeling of tracking contour based on the vehicular kinematics. Section III-C discusses how to optimize the refresh time for the minimal tracking contour. Section III-E suggests how to expand the tracking contour under measurement errors in the target localization. Section III-F explains how the minimal contour is updated according to the vehicle's movement considering the energy saving related to sensor *warming-up* time.

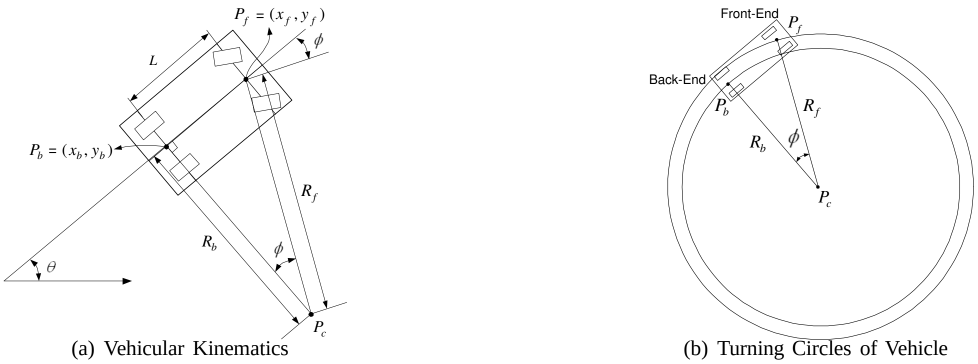
A. Modeling of Vehicle Motion

We assume that the mobile target is a four-wheeled vehicle. We can define the vehicle motion based on the vehicular kinematics [3], [9]. Figure 3(a) shows the front-end point P_f and the back-end point P_b of the vehicle that is turning right with steering angle ϕ. Assume that the vehicle's wheelbase is L that is the distance between the front-end point and back-end point. From the vehicle kinematics [3], we know that the front-end point P_f is moving on the circle whose radius is R_f and the back-end point P_b is moving on the circle whose radius is R_b like in Figure 3(b). R_f and R_b can be obtained by the following equations [3]:

$$R_f = \frac{L}{sin(\phi)} \tag{1}$$

$$R_b = \frac{L}{tan(\phi)} \tag{2}$$

We use P_b for vehicle's position and use R_b to make a turning circle. We can model the vehicle motion by a random vector $M = (X, Y, \Theta, V, \Phi, \dot{V})$ in $\mathbb{R}^6$ where (X, Y) is the position of the vehicle (P_b), Θ is the orientation (or direction), V is the speed, Φ is the steering angle, and $\dot{V}$ is the acceleration [9]. We assume that the initial state $M_0 = (X_0, Y_0, \Theta_0, V_0, \Phi_0, \dot{V}_0)$ is a Gaussian random vector. Let L be the wheelbase of the vehicle. Let s be the time when the vehicle has so far moved from its first detected time. The driving process that determines the vehicle motion is $(\Phi, \dot{V})$. We can compute



(a) Vehicular Kinematics

(b) Turning Circles of Vehicle

Fig. 3. Vehicular Kinematics for Tracking Contour

the remaining components of M by integrating the following stochastic differential equations over time $t \geq 0$:

$$dV_t = \dot{V}_t dt \tag{3}$$

$$d\Theta_t = \frac{V_s}{L} tan(\Phi) dt \tag{4}$$

$$\text{where} \quad V_{s+\Delta t} = V_s + \int_0^{\Delta t} \dot{V}_t \, dt \quad \text{and}$$
$$s \leftarrow s + \Delta t.$$

$$(dX_t, dY_t) = V_s(cos(\Theta_s), sin(\Theta_s))dt \tag{5}$$

$$\begin{aligned} \text{where} \quad & V_{s+\Delta t} = V_s + \int_0^{\Delta t} \dot{V}_t \, dt, \\ & \Theta_{s+\Delta t} = \Theta_s + \int_0^{\Delta t} \dot{\Theta}_t \, dt = \Theta_s + \int_0^{\Delta t} \frac{V_s}{L} tan(\Phi) \, dt, \\ \text{and} \quad & s \leftarrow s + \Delta t. \end{aligned}$$

We can update X_s and Y_s as follows:

$$\begin{aligned} X_{s+\Delta t} &= X_s + \int_0^{\Delta t} V_s cos(\Theta_s) \, dt \\ Y_{s+\Delta t} &= Y_s + \int_0^{\Delta t} V_s sin(\Theta_s) \, dt \\ s &\leftarrow s + \Delta t \end{aligned} \tag{6}$$

The driving processes Φ and $\dot{V}$ are Gaussian, satisfying the following equations:

$$\begin{aligned} d\Phi_t &= -\alpha \Phi_t \, dt + \sigma \, dC_t \\ d\dot{V}_t &= q \, dB_t \end{aligned} \tag{7}$$

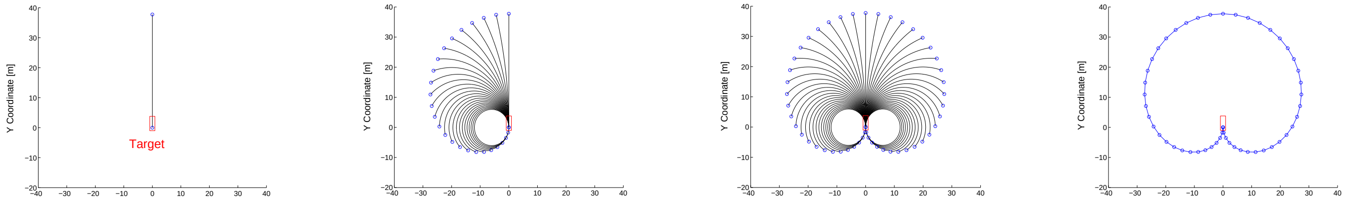
where (B, C) is a Brownian motion independent of M_0 such that $(B_0, C_0) = 0$. The constants α, σ, and q are chosen to suit particular vehicle motions.

We can consider the vehicle's backward motion. That is, when the vehicle stops, it can move backward. In this case, we can regard the vehicle's backward motion to be the same as the forward motion since it has the same motion equations above.

B. Modeling of Tracking Contour

We can make a tracking contour using the vehicular kinematics discussed in Section III-A. Let (X_0, Y_0) be the target's current position, Θ_0 be the target's direction, and V_0 be target's speed. Let ΔT be refresh time. Let $(X_{\Delta T}, Y_{\Delta T})$ be the target's position after ΔT. We divide target movement into three kinds: (a) Straight movement, (b) Left turning, and (c) Right turning. We can make a polygon representing the tracking contour with the three styles of movement. Figure 4 shows the procedure constructing the tracking contour. The straight movement gives two points in like Figure 4(a). The first point is the target's current position (X_0, Y_0). The second $(X_{\Delta T}, Y_{\Delta T})$ is the point away from (X_0, Y_0) by the distance that the target can go with its current speed and maximum acceleration. The point $(X_{\Delta T}, Y_{\Delta T})$ can be obtained from Eq. 5. Like in Figure 4(b), the left points can be obtained from Eq. 5 by changing the steering angle from 0 to maximum steering angle discretely to the left. In the same way, the right points can be obtained by changing the steering angle from 0 to maximum steering angle discretely to the right. The obtained points construct a polygon like Figure 4(d). This polygon is used by each sensor to determine whether it should work for tracking. Only the sensors inside the polygon work, and other sensors continue to be idle. The inside checking is done by *Ray Crossings* algorithm [7].

When the refresh time is less than one turning time of the target, the tracking contour guarantees the tracking of the moving target without missing. Since the tracking contour covers all the possible area visited by the mobile target, it guarantees the no-missing tracking. But when the refresh time is bigger than one turning time, it is very hard to represent a tracking contour less than the tracking circle. In fact, in most cases, as the optimal refresh time is less than one turning time through the optimization of refresh time, we need not worry about the case where the refresh time is bigger than one turning time. Besides, the fast moving target cannot make a sharp turn to the left or to the right with its maximum steering angle since the maximum turning makes the target be overturned.



(a) Case of Straight Movement (b) Case of Left Turning (c) Case of Right Turning (d) Completed Tracking Contour

Fig. 4. Construction Procedure of Tracking Contour

C. *Optimization of Refresh Time for Minimal Contour*

We need to use an optimal refresh time to let the contour-based tracking consume the minimum energy for target tracking. In this section, we shows how to optimize the refresh time according to the target's current speed.

Table I shows the notation of parameters used in this paper. The power consumption rates come from the Mica that is one of Berkeley Motes [10]. Let n be the number of minimal contours used for tracking given a target trajectory. Let m be the estimated number of sensor nodes per minimal contour given the sensor node density per unit area (i.e., ρ). The total energy cost E_{total} is the sum of all the operations required for tracking as follows:

TABLE I
NOTATION OF PARAMETERS

Parameter	**Description**
ΔT	Refresh time: unit is [sec]
v	Mobile target's speed: unit is [m/sec]
ϕ_{max}	Maximum steering angle: 25°
ϕ	Steering angle (or turning angle): unit is [°]
d	Average trajectory distance which is used to find out an optimal refresh time ΔT: unit is [m]
ρ	Sensor density that is the average number of sensor nodes per 1 m^2 in the surveillance field : unit is $[1/m^2]$
n	Number of minimal contours
R	Maximum communication range: 75[m]
P_{tx}	Energy cost of RF transmitting per second: 21[mW]
P_{rx}	Energy cost of RF receiving per second: 15[mW]
P_{comp}	Energy cost of computation per second: 16.5[mW]
P_{warm}	Energy cost of warming-up time for preparing sensing devices in each sensor node per second: 15[mW]
P_{work}	Energy cost of running sensing devices in each sensor node per second: 10[mW]
E_{total}	Total energy cost: unit is [mJ]
T_{tx}	Time cost of RF transmitting per hop for disseminating the minimal contour information: 0.2[sec]
T_{rx}	Time cost of RF receiving per hop for receiving the minimal contour information: 0.1[sec]
T_{comp}	Time cost of computation in a root sensor node for determining the minimal contour information: 0.02[sec]
T_{warm}	Time cost of warming-up time for preparing sensing devices in each sensor node: 0.1[sec]
T_{sense}	Time cost of minimum working time for sensing mobile target in each sensor node: 0.5[sec]
T_{work}	Time cost of working time for sensing mobile target in each sensor node during the minimal contour's lifetime: unit is [sec]
T_{total}	Total time cost: unit is [sec]

$$\begin{aligned} E_{total} =& (P_{comp}T_{comp} + P_{warm}T_{warm} + P_{work}T_{sense} \\ & + P_{work}T_{work})mn + P_{tx}T_{tx}n + P_{rx}T_{rx}n\rho\pi R^2 \end{aligned} \tag{8}$$

To optimize the refresh time ΔT minimizing the overall required energy for tracking a mobile target, we need to consider the following. Since the actual trajectory of a mobile target is unknown, we cannot see the number of minimal contours required for tracking the target within the surveillance field. So, we need to minimize the size of the minimal contour that is closely related to the energy spent during the general target tracking where the target has directional movement rather than random walk. For the reliable tracking, the refresh time ΔT should be no less than the sum of computation time, RF transmitting time, warming-up time, and minimum working time as follows:

$$\Delta T \geq T_{tx} + T_{comp} + T_{warm} + T_{sense} \tag{9}$$

The shape of the minimal contour is a function of ΔT, v, and ϕ_{max} as follows:

$$S = f(\Delta T, v, \phi_{max}) \tag{10}$$

Let d be an average target trajectory distance used to find out an optimal refresh time ΔT. We need to optimize Δt to minimize the total energy spent for the target tracking for the average trajectory distance; that is, we can formulate our problem as follows:

$$\Delta T \leftarrow arg \min_{t \in \mathbb{R}^+} Total\_Energy(t, v, d, \phi_{max}) \tag{11}$$

$$\text{where } T_{tx} + T_{comp} + T_{warm} + T_{sense} \leq t \leq \frac{d}{v}$$

In order to compute the total energy function $Total\_Energy$ of (t, v, d, ϕ_{max}), we need to find the minimal contour's shape S corresponding to t by Eq. 10. Next, we find the average number of sensor nodes given S and the sensor deployment distribution, such as a uniform distribution with m sensor nodes. Assume that a vehicle with wheelbase L can turn with

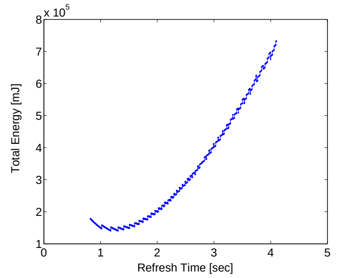


Fig. 5. Searching of Optimal Refresh Time considering Total Energy consumed for Tracking

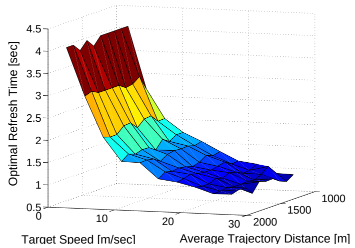


Fig. 6. Optimal Refresh Time according to Vehicle Speed and Average Trajectory Distance

steering angle ϕ with speed v. Figure 4(d) shows a minimal contour's shape as a polygon where v = 30 [km/h], L = 2.8 [m], and ϕ_{max} = 25°.

Algorithm 2 computes the optimal refresh time given the target's speed v, average trajectory distance d, and sensor density ρ by using the one-dimensional optimization, such as *Golden Section Search algorithm with parabolic interpolation* [6]. Note that there are many local minimums in the refresh time optimization like in Figure 5. The used searching algorithm should find out the global optimum among these local minimums. Note that the smallest contour made by the smallest allowable refresh time is not always the minimal contour in terms of energy cost. In Figure 5, about 1.305[sec] is the optimal refresh time.

Algorithm 2 $Find\_Optimal\_Refresh\_Time(v, d, \rho)$

```
$T_{min} \leftarrow T_{tx} + T_{comp} + T_{warm} + T_{sense}$
  {minimum time needed for minimal contour guaranteeing no missing of vehicle}
$T_{max} \leftarrow d/v$
  {minimum maximum time to cover the average trajectory distance by one minimal contour}
$\Delta T \leftarrow Searching(v, d, \rho, T_{min}, T_{max})$
  {perform a searching algorithm to find out a refresh time having a global minimum energy cost within the given range $(T_{min}, T_{max})$}
$return\ \Delta T$
  {$\Delta T$ is the optimal refresh time}
```

Algorithm 3 computes the total energy consumed for tracking given speed v, refresh time t, average trajectory distance d, and sensor density ρ by Eq. 8. Functions $Compute\_Contour\_Polygon()$ and $Polygon\_Area()$ are called to get the polygon approximating to the current contour and to compute the area of the contour polygon, respectively [7].

Algorithm 3 $Compute\_Total\_Enery(v, t, d, \rho)$

```
$n \leftarrow \lceil d/(vt) \rceil$
  {$\lceil x \rceil$ is the ceiling function of $x$}
$[X, Y] \leftarrow Compute\_Contour\_Polygon(v, t)$
  {this function returns the $X$ and $Y$ vectors representing the polygon approximating to the contour determined by $v$ and $t$ along with the maximum steering angle $\phi_{max}$}
$a \leftarrow Polygon\_Area(X, Y)$
  {this function returns the area of the polygon represented by $X$ and $Y$ vectors}
$m \leftarrow \lceil a\rho \rceil$
  {$m$ is the estimated number of sensors per unit area (i.e., $1m^2$)}
$E_{total} \leftarrow (P_{comp}T_{comp} + P_{warm}T_{warm} + P_{work}T_{sense} + P_{work}T_{work})mn + P_{tx}T_{tx}n + P_{rx}T_{rx}n\rho\pi R^2$
  {$E_{total}$ is computed with Eq. 8}
$return\ E_{total}$
  {$E_{total}$ is the total energy needed given refresh time $t$}
```

Figure 6 shows the optimal refresh time according to the vehicle's speed v and average trajectory distance d where the optimization of refresh time is done. Through Figure 6, we can see that the optimal refresh time is dominantly affected by the vehicle's speed, not the average trajectory distance. In $MCTA$, the optimal refresh time is chosen for a given vehicle speed from this graph that is stored as a form of *look-up table*.

D. Minimization of Communication Cost

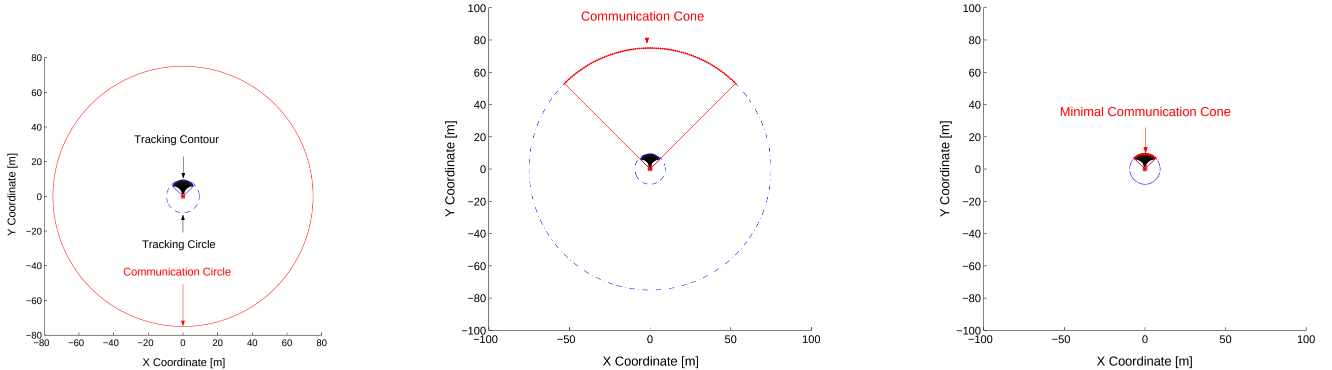
Since the communication cost is dominant factor in energy consumption in the target tracking, it is worthy to find out how to reduce such cost in our contour solution. Figure 7 shows the communication area for disseminating the contour information to the neighboring sensors. Figure 7(a) shows the communication area made by full transmission power. It shows three areas for tracking: (a) the communication circle whose radius is RF communication range, (b) the tracking contour, and (c) the tracking circle.

For the directional antenna for directional transmission, the root sensor sends the contour information only towards the sensors belonging to the current contour. The communication area can be reduced from the communication circle of Figure 7(a) to the communication cone of Figure 7(b). Through this directional transmission, we can modify the energy cost function of Eq. 8 as follows:

$$\begin{aligned} E_{total} =& (P_{comp}T_{comp} + P_{warm}T_{warm} + P_{work}T_{sense} \\ & + P_{work}T_{work})mn + P_{tx}T_{tx}n \\ & + P_{rx}T_{rx}n\rho(\pi R^2)\frac{D}{2\pi} \end{aligned} \tag{12}$$

where D is the angle of directional transmission in radians. D is selected as a value that can include our contour. The transmission area is the cone whose radius is the communication range and the internal angle is D. Thus, we can reduce the RF receiving cost by reducing the number of receiving sensors by $1 - \frac{D}{2\pi}$ times, that is, from $\rho(\pi R^2)$ to $\rho(\pi R^2)\frac{D}{2\pi}$.

We can minimize the communication cost by letting only the sensors within the tracking contour receive the contour



(a) Communication Circle by Full Tx Power

(b) Communication Cone by Tx Power Control

(c) Minimal Communication Cone by both Tx Power Control and Directional Antenna

Fig. 7. Communication Area for Contour Information Dissemination

information. For this purpose, we adopt the transmission (Tx) power control [11], [12] and directional antenna [13]–[15]. As a result, the communication cone in Figure 7(b) is reduced to the minimal communication cone like in Figure 7(c). Thus, the energy cost function can be modified as follows:

$$\begin{aligned} E_{total} =&(P_{comp}T_{comp} + P_{warm}T_{warm} + P_{work}T_{sense} \\ &+ P_{work}T_{work})mn + P_{tx}T_{tx}n \\ &+ P_{rx}T_{rx}n\rho(\pi(v\Delta T)^2)\frac{D}{2\pi} \end{aligned} \tag{13}$$

Finally, the gain of communication energy saving is from Eq.8 to Eq.13:

$$G = P_{rx}T_{rx}n\rho\pi[R^2 - (v\Delta T)^2\frac{D}{2\pi}] \tag{14}$$

for $R > v\Delta T$. Note that in the above formulas, we ignored another gain obtained from Tx power reduction for shorter communication range for simplicity.

E. Handling of Measurement Errors for Target Localization

The target localization is used to locate the most likely position of the mobile target with several sensor nodes that detected the mobile target at the same time [4], [5]. In order to estimate the mobile target's direction and speed, more than two localizations are needed where each localization provides a pair of the time and target position. We have assumed so far that the localization is performed to give these target's current position, speed and direction. However, in reality, since there are measurement errors in every localization scheme, we need to consider them to make a more realistic tracking contour.

These measurement errors can be modeled as noises that are *Gaussian* random variables [9]. The minimal contour can be expanded to guarantee containing the tracked target within the user-defined confidence interval such as 90%.

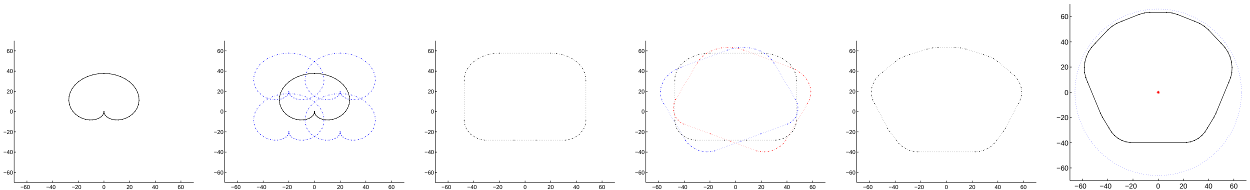
Let (X_t, Y_t, Θ) be the actual target movement vector where X_t is x-coordinate, Y_t y-coordinate, and Θ direction. Let $(\bar{X}_t, \bar{Y}_t, \bar{\Theta}_t)$ be a measurement of target movement where $\bar{X}_t$ is the measured x-coordinate, $\bar{Y}_t$ the measured y-coordinate, and $\bar{\Theta}_t$ the measured direction. We can see that this measurement has noise terms as follows:

$$\begin{aligned} \bar{X}_t &= X_t + \varepsilon_x \\ \bar{Y}_t &= Y_t + \varepsilon_y \\ \bar{\Theta}_t &= \Theta_t + \varepsilon_\theta \end{aligned} \tag{15}$$

where ε_x is a *Gaussian* random variable with $N(\mu_x, \sigma_x^2)$, ε_y with $N(\mu_y, \sigma_y^2)$, and ε_θ with $N(\mu_\theta, \sigma_\theta^2)$. Assume that the user-defined confidence interval is p. Let ξ_x be the offset from μ_x satisfying the confidence interval p. Let ξ_y be the offset from μ_y satisfying the confidence interval p. Let ξ_θ be the offset from μ_θ satisfying the confidence interval p. We will explain the construction procedure of tracking contour under measurement errors with Figure 8. First of all, like in Figure 8(a), we make a basic tracking contour according to Section III-B. Next, like in Figure 8(b), we make four worst-case contours considering ε_x and ε_y along with ξ_x and ξ_y. Next, like in Figure 8(c), we merge these five contours into a convex hull. Next, considering the direction error like in Figure 8(d), we rotate this convex hull to the left by $\varepsilon_\theta + \xi_\theta$ and also rotate this convex hull to the right by $\varepsilon_\theta + \xi_\theta$. These three convex hulls are merged into a bigger convex hull like in Figure 8(e) using *Graham's Algorithm* [7]. This one is our final tracking contour. It is still smaller than the tracking circle considering the same measurement errors like in Figure 8(f). Note that for the speed measurement error, we just use the greatest speed corresponding to the given confidence interval in the same way as the position error and direction error.

F. Update of Minimal Contour

The minimal contour follows the target's movement changing its refresh time based on the target's speed. The optimal contour size is also determined by the average trajectory distance used for the optimization of refresh time given the target's speed. So, we need to maintain the constant contour shape by changing the refresh time according to the target's current speed with the average trajectory distance of the targets observed so far. We can see that the refresh time means the



(a) Tracking Contour under No Measurement Error (b) Four Tracking Contours under Position Measurement Error (c) Convex Hull representing Merged Tracking Contour (d) Two Convex Hulls under Direction Measurement Error (e) Convex Hull representing Merged Tracking Contour (f) Comparison of Final Tracking Contour and Tracking Circle

Fig. 8. Construction Procedure of Tracking Contour Under Measurement Errors

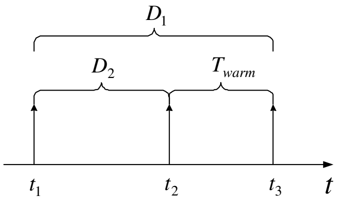


Fig. 9. Warmimg-up Starting Time of Sensing Devices

lifetime of the current contour when the sensors within it should continue to work for sensing the target.

Let T_{min} be the minimum overhead time to prepare for a new contour in the current contour that is the lower bound for the refresh time ΔT as in Eq. 9; that is, $T_{min} = T_{tx} + T_{comp} + T_{warm} + T_{sense}$. Before the target leaves the current contour, the next contour is prepared. That is, if some sensor that detects the target the overhead time T_{min} before the contour's lifetime ΔT expires, it broadcasts the target's position and movement information to its neighbor sensors. The neighbor sensors determine whether they will participate in sensing by performing Algorithm 1 or not. When the current contour's refresh time ΔT expires, the sensors turn off their sensing devices except for the sensors that continue to belong to the next contour.

The starting time of sensing devices is determined considering the movement information message's timestamp and the sensor devices' warming-up time. Table II shows the time variables needed to compute the warming-up starting time of sensing devices. We can get the warming-up starting time t_2 as follows:

$$t_2 = t_1 + (D_1 - T_{warm}) \tag{16}$$

where t_1 is the timestamp of the contour information message broadcasted by the current contour's root node, D_1 is the target's expected travel time from the root node to the computing sensor node, and T_{warm} is the sensing warm-up time.

IV. DISCUSSION

A. Computation of Tracking Contour along with Optimal Refresh Time

The complex computation related to optimal refresh time is done by some powerful sink node outside the sensor network. A table is constructed to have pairs of vehicle speed and corresponding optimal refresh time. Another table has polygons for tracking contour according to the pair of vehicle speed and optimal refresh time. These two tables are disseminated to the sensors in the sensor network, which are used for our target tracking algorithm in a distributed computing manner. So, since the expensive computation is done in the sink node, the computation cost in each sensor is not so high in comparison with circle-based tracking algorithm.

TABLE II
TIME VARIABLES FOR COMPUTATION OF WARMING-UP TIME

Parameter	**Description**
t_1	Time when the contour information was broadcasted by the root node
t_2	Time when the sensor node starts the warming-up of its sensing devices
t_3	Time when the sensing devices start the actual sensing
T_{warm}	Time needed for warming-up sensing devices in sensor node
D_1	Time needed so that the target can reach the sensor node earliest [17]; that is, $D_1 = l/v$ where l is the Euclidean distance between the target's starting position in the current contour and the sensor's position and v is the target's speed
D_2	Time difference between t_1 and t_2; that is, $D_2 = t_2 - t_1$

B. On-line Classification for Mobile Target

By observing the motion of the tracked target, such as the greatest turning angle so far, we can figure out a more appropriate motion process used for constructing a more optimal contour as in Section III-A. That is, we can use the accurate vehicle motion process according to the estimated vehicle type, such as two-wheeled vehicle, tricycle, four-wheeled vehicle, Reeds-Shepp car, and Dubins car [3].

V. PERFORMANCE EVALUATION

We model the sensor network including sensor and vehicle on the basis of SMPL simulation model along with Matlab where SMPL is one of the discrete event driven simulators [8], [16].

A. Simulation Analysis

We define the sensor network lifetime as the time until at least one sensor node among the sensor nodes on the surveillance field dies due to the energy exhaustion.

The simulation environment is as follows:

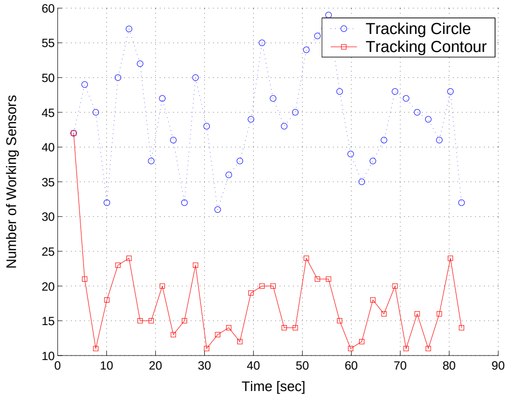
TABLE III
MEASUREMENT OF NUMBER OF SENSING SENSORS

Metric	**Contour (X_t)**	**Circle (Y_t)**	**Ratio**
$Area$	24.1$[m^2]$	137.8$[m^2]$	0.18
$Expectation(E)$	17	44	0.39
$Variance(Var)$	35.6	54.5	0.65

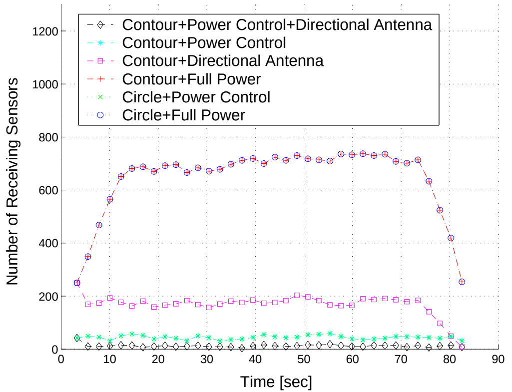
- 10,000 sensor nodes are uniformly deployed in the surveillance field of 500[m] × 500[m].
- The radius of communication is 75[m].
- The vehicle's speed is 30[km/h] and its maximum turning angle is 25°.

We simulated to know the number of sensing sensor nodes, the number of receiving sensor nodes, and the cumulative energy consumption according to the vehicle's movement with the tracking circle and tracking contour, respectively. Figure 10 shows these three kinds of performance comparison. Let X_t be the number of working sensor nodes at time t for the contour based scheme and Y_t be that at time t for the circle based scheme. Let $E[X_t]$ be the average number of working sensor nodes for the contour based scheme and $E[Y_t]$ be that for for the circle based scheme. Let $Var[X_t]$ and $Var[Y_t]$ be the variances of X_t and Y_t, respectively. Table III shows the comparison between two tracking schemes in terms of of the number of sensing sensor nodes in our simulation scenario. We can see that the ratio of the expected number of working (i.e., sensing) sensor nodes in contour based scheme to that in circle based scheme is about 0.25 time, equal to the area ratio (0.18) where the contour's area is 24.14$[m^2]$ and circle's area is 137.78$[m^2]$. Therefore, we can conclude that we can reduce the number of working sensor nodes with our minimal contour scheme, maximizing the sensor network lifetime. Note that in Figure 10(a), the number of sensing sensors at the first refresh time is the same in two schemes. The reason is that the tracking contour cannot have enough information for the vehicle's movement at first, so should use the tracking circle.

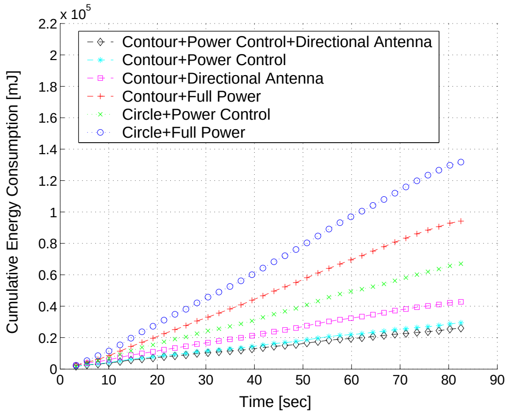
The cumulative energy consumptions for two schemes are shown in Table IV. When we do not use both RF transmission power control and directional antenna, the performance ratio of two tracking schemes is only 0.71; that is, the tracking contour can improve only 29% of the tracking circle's performance. The reason is that though 25% of sensors in tracking circle are used in tracking contour, the communication cost that is the major factor in energy cost is the same in two schemes from Eq. 8. To improve the performance in terms of energy cost, we need to use the RF transmission power control and directional antenna discussed in Section III-D. Like in Figure 10(b), we can reduce a large number of receiving sensor nodes with both tracking contour and two communication technologies. The tracking contour with two technologies can improve about 61% of the tracking circle that also uses the RF transmission power control. Note that it is good to use the tracking contour only with the RF transmission power control and without the directional antenna since it still brings the improvement of 56%.



(a) Number of Sensing Sensor Nodes Every Refresh Time



(b) Number of RF Receiving Sensor Nodes Every Refresh Time



(c) Cumulative Energy Consumption according to Time

Fig. 10. Performance Comparison between Tracking Circle and Tracking Contour

TABLE IV
COMPARISON OF CUMULATIVE ENERGY CONSUMPTION

Method	**Contour**	**Circle**
Maximum Transmission Range (i.e., Full Power)	94166$[mJ]$	131770$[mJ]$
Directional Antenna	42800$[mJ]$	N/A
RF Tx Power Control	29438$[mJ]$	67045$[mJ]$
RF Tx Power Control and Directional Antenna	25988$[mJ]$	N/A

VI. RELATED WORK

Aljadhai et al. proposed a resource allocation scheme based on predictive mobility in mobile wireless environments [17]. In their paper, the directionality probability was introduced to determine which cell the mobile target will visit next. The cell on the direction from the previous cell to the current cell is regarded as the most likely visited cell. Their scheme can be used for resource allocation in cellular networks having user's mobility profile, but cannot be used for the tracking of a mobile target whose mobility profile is unknown in the wireless sensor network. On the other hand, since our scheme considers all the possible tracking area where the mobile target can visit mechanically after some time, it guarantees the reliable tracking of the mobile target, such as vehicle, without its mobility profile.

Since the papers of [1], [2] model the mobile target as random walk, the mobile target can take any direction from the current position since the vehicle kinematics are ignored. So, the area where the mobile target cannot visit for some time belongs to the tracking area. On the other hand, since our scheme models the mobile target's movement based on the vehicular kinematics [3], only the area where the mobile target can visit mechanically belongs to the tracking area. As a result, we can reduce the number of working sensor nodes in each tracking area called the minimal contour for the energy efficiency. While the tracking algorithms in [1], [2] focus on the optimization of the reconfiguration of data collection tree for target tracking, this optimization for tree reconfiguration is out of scope in our paper. Their tracking algorithms can adopt our tracking contour in their tracking algorithm to reduce the number of working sensors.

The *RF transmission power control* is not only used to determine the neighboring sensor nodes that can hear the packet, but also to reduce the communication cost among clustered sensor nodes [11], [12]. In our tracking algorithm, the transmission power control not only can allow the number of RF receiving sensor nodes to be minimized, but also can reduce the transmit power for shorter transmission radius. The *directional antenna technology* is used for mobile ad hoc networks including sensor networks for the parallel communication in MAC protocol level [13]–[15]. Our tracking algorithm uses the directional antenna in order to reduce the number of RF receiving sensors.

VII. CONCLUSION

In this paper, we suggested a target tracking algorithm $MCTA$ using minimal tracking area called *tracking contour* that is based on the vehicular kinematics. $MCTA$ minimizes the number of working sensor nodes in terms of the communication and sensing energy cost during the mobile target's trajectory. We showed that the ratio of tracking contour's working sensor number to tracking circle's working sensor number is proportional to the ratio of the tracking contour's area to tracking circle's area. This indicates that the reduction of the tracking area leads to the communication and sensing energy saving. We optimize the refresh time for minimal contour according to the vehicle current speed. Also, in order to reduce the dissemination of tracking contour information within the tracking contour, we used the *RF transmission power control* and *directional antenna*, leading to the minimization of the number of RF receiving sensors. As our future work, we will implement our tracking algorithm in real sensor nodes (e.g., Mica [10]) and test it in our indoor testbed.

ACKNOWLEDGMENT

This work was supported by the Department of Computer Science and Engineering and Digital Technology Center at the University of Minnesota.

REFERENCES

[1] Wensheng Zhang and Guohong Cao, *Dynamic Convoy Tree-Based Collaboration for Target Tracking in Sensor Networks*, IEEE Transactions on Wireless Communications, Vol. 3, No. 5, September 2004.

[2] Wensheng Zhang and Guohong Cao, *Optimizing Tree Reconfiguration for Mobile Target Tracking in Sensor Networks*, IEEE Infocom, March 2004.

[3] Steven M. LaValle, *Planning Algorithms*, Cambridge University Press, 2006.

[4] Tian He, Chengdu Huang, Brian M. Blum, John A. Stankovic and Tarek Abdelzaher, *Range-Free Localization Schemes for Large Scale Sensor Networks*, ACM Mobicom, September 2003.

[5] Jeongkeun Lee, Kideok Cho, Seungjae Lee, Taekyoung Kwon and Yanghee Choi, *Distributed and energy-efficient target localization and tracking in wireless sensor networks*, Elsevier Computer Communications (COMCOM), Vol. 29, No. 13, pp. 2494–2505, August 2006.

[6] D. R. Kincaid and W. W. Cheney, *Numerical Analysis: the Mathematics of Scientific Computing*, Van Nostrand, 1991.

[7] Joseph O'Rourke, *Computational Geometry in C*, 2^{nd} Edition, Cambridge University Press, 1998.

[8] M. H. MacDougall, *Simulating Computer Systems: Techniques and Tools*, MIT Press, 1987.

[9] S. J. Maybank, A. D. Worrall and G. D. Sullivan, *Filter for Car Tracking Based on Acceleration and Steering Angle*, British Machine Vision Conference, 1996.

[10] Jason L. Hill and David E. Culler, *Mica: a Wireless Platform for Deeply Embedded Networks*, IEEE Micro, Vol. 22, Nov/Dec 2002.

[11] Vikas Kawadia and P. R. Kumar, *Principles and Protocols for Power Control in Wireless Ad Hoc Networks*, IEEE J. Sel. Areas Commun. (JSAC), Vol. 1, pp.76–88, January 2005.

[12] Vikas Kawadia and P. R. Kumar, *Power Control and Clustering in Ad Hoc Networks*, IEEE Infocom, March 2003.

[13] Young-Bae Ko, Vinaychandra Shankarkumar and Nitin H. Vaidya, *Medium Access Control Protocols Using Directional Antennas in Ad Hoc Networks*, IEEE Infocom, March 1999.

[14] Qingjiang Tian, Seema Bandyopadhyay and Edward J. Coyle, *Effect of Directional Antennas on Spatiotemporal Sampling in Clustered Sensor Networks*, IEEE WCNC, April 2006.

[15] Sensor Network Project for Directional Antenna, https://engineering.purdue.edu/IDEAS/SensorNtwks.html

[16] Matlab of the MathWorks, http://www.mathworks.com

[17] AbdulRahman Aljadhai and Taieb F. Znati, *Predictive Mobility Support for QoS Provisioning in Mobile Wireless Environments*, IEEE Journal on Selected Areas in Communications (JSAC), Vol. 19, No. 10, October 2001.