Noise Mitigation for Target Tracking in Wireless Acoustic Sensor Networks

Youngwon Kim An[1], Seong-Moo Yoo1, Changhyuk An[2], Earl Wells[1]
[1]Department of Electrical and Computer Engineering
University of Alabama, Huntsville, AL
yka0001@uah.edu, yoos@eng.uah.edu, wells@eng.uah.edu.
[2]PRA, Huntsville, AL, 35801
huggyan@gmail.com
*Corresponding author: *Seong-Moo Yoo*

Received December 30, 2012; revised March 9, 2013; accepted April 24, 2013; published May 31, 2013

Abstract

In wireless sensor network (WSN) environments, environmental noises are generated by, for example, small passing animals, crickets chirping or foliage blowing and will interfere target detection if the noises are higher than the sensor threshold value. For accurate tracking by acoustic WSNs, these environmental noises should be filtered out before initiating track. This paper presents the effect of environmental noises on target tracking and proposes a new algorithm for the noise mitigation in acoustic WSNs. We find that our noise mitigation algorithm works well even for targets with sensing range shorter than the sensor separation as well as with longer sensing ranges. It is also found that noise duration at each sensor affects the performance of the algorithm. A detection algorithm is also presented to account for the Doppler effect which is an important consideration for tracking higher-speed ground targets. For tracking, we use the weighted sensor position centroid to represent the target position measurement and use the Kalman filter (KF) for tracking.

***Keywords*:** Wireless sensor networks, Doppler effect, environmental noises, tracking, binary sensor, noise mitigation.

A preliminary version of this paper appeared in Int’l Conference on Computer Applications in Industry and Engineering, held in New Orleans, LA,USA, Nov 13-15, 2012. This version includes an improved mitigation algorithm and supporting implementation results.

http://dx.doi.org/10.3837/tiis.2013.05.013

1. Introduction

For tracking ground vehicles or underwater targets, acoustic Wireless Sensor Networks (WSNs) has been regarded to be useful due to recent advancement of micro sensors technology [1,18].

A number of tracking methods for a single ground target tracking have been proposed in the literature for WSNs type applications. Mechitov et al. [1] and Kim et al. [2] and Wang et al. [3], [4] used geometric methods for target position estimation. Ribeiro et al. [5] developed Kalman filter (KF) based recursive algorithms for distributed state estimate based on the sign of innovations (SOI). Recently, a number of authors [6,7,8] chose the particle filters (PF) over KF for target tracking in acoustic WSNs environments claiming that KF is not an optimal filter as the system and measurement process are not linear in real environments. For the application of PF to acoustic WSNs, PFs with lower computational complexity were proposed [14,15] and similar results were reported by other authors [10,11,12,13]. As an improvement over PF, Teng et al. [9] applied a variational filter (VF) for target tracking. The study of target tracking in acoustic WSNs has been extended to multiple target tracking in the areas of multiple-source localization [19], and multiple target detection and tracking [20,21].

All the previous studies mentioned above have limited application to use their detection and tracking algorithm in real acoustic WSNs environments. They use the power law for simulating target sound power detection by the sensors in the field and use the same power law for modeling the detection in their trackers. Their use of the power law excludes the Doppler effect in their detection simulation and makes the detection model in their tracker susceptible to mismatch with the detection by the sensors deployed in real environments [8]. The wireless acoustic sensors are not only subject to Doppler effect but also are prone to be disturbed by surrounding noises and thus the tracking results will be affected severely by the environmental noises. None of the previous studies mentioned above studied the noise and Doppler effect on acoustic WSNs target detection and tracking.

In order to remedy the shortfalls of the previous studies which use the power law detection model, we developed a realistic detection algorithm where the Doppler effect appears naturally depending on the target speed [16]. For target position measurement, we used the centroid of the detecting sensor positions weighted with detecting sound power. Our realistic detection model warrants no mismatch of detection between the tracker and the sensors deployed in the field. Our KF based tracker shows that computing speed is much faster than the speed of more sophisticated PF and VF but track accuracy is comparable with those trackers for linear and accelerated target motions. The detection model enabled us to study various environmental effects including Doppler effect, detection message delay and message collision at the fusion center, and different sampling time steps on track accuracy. Our study showed that the Doppler effect affects the track accuracy through the bias of target position measurement for higher target speed and demonstrated that the track accuracy is sensitive to these environmental effects.

Even though our previous study [16] made a significant progress for acoustic WSNs target tracking in realistic environments, the study did not reflect noisy military environments in the simulation. The wireless acoustic sensors will detect numerous noises from the surrounding environment and the sensors will send detection message to the fusion center whenever the detecting sound is higher than the sensor threshold regardless of the origin of the sound. The noises reported by the sensors adversely affect the centroid computation as a target position measurement and severely disrupt accurate target tracking. In order to deploy acoustic WSNs tracking system successfully in real environments, the development of noise mitigation algorithm is a requirement.

In this paper, we present how the environmental noises affect target detection and propose a new algorithm for environmental noise mitigation that works well for various target sound powers.

This paper is organized as follows: Section 2 describes the detection and tracking framework. It gives a detailed description of Doppler effect in the sensor detection simulation. This section is a summary of our previous paper [16] and is given for the completeness of this paper. In Section 3, we propose a new algorithm for environmental noise mitigation and show the mitigation results for various target sound powers. Section 4 discusses the implication of our results and future work.

2. Detection and Tracking Framework

Detailed description of our detection and tracking models that includes Doppler effect is given in our previous paper [16], but for the completeness of this paper a brief description of the models is shown below.

2.1 Description of Acoustic WSNs

We assume that the wireless sensors are distributed in a uniform grid of 50 by 50 with sensor separation (the distance between the sensors), Δ_s, being 25 meters in the sensing region. So the total number of sensors, N_s, deployed in the field is 2500. This distribution is reasonable for the detection and tracking of ground vehicles with approximate speeds of 20 m/sec and sensing range of 15 ~ 75 meters. Here, sensing range stands for the maximum distance from the target at which a sensor can detect the target sound above the sensor threshold. A target with higher sound power has longer sensing range than a target with lower sound power. Each sensor does not know the origin of detecting sound and sends detecting information to the fusion center when the detecting sound power is higher than its threshold value without communicating between sensors. Each sensor has its own unique ID that translates into its position.

2.2 Algorithm for Target Sound Detection with Doppler Effect

The target sound detection model with Doppler effect can be explained with **Fig. 1**, **Fig. 1** (a) shows the target position moving from left to right with a constant speed at times t_0, t_1, t_2, and t_3 and the sound wave propagation from respective target positions in the time step, Δt. At $t = t_0$, the target at $x=x_0$ starts to send sound and at $t = t_1$ the target is at $x = x_1$ and sound propagates by $d_0 = V_s \Delta t$ from x_0. Here V_s is the speed of sound. At $t = t_2$, the target moves to $x = x_2$ and sound from x_0 propagates by $d_0 = V_s \cdot 2\Delta t$ and the sound from x_1 propagates by $d_1 = V_s \cdot \Delta t$ At subsequent time steps, the target and sound propagate in similar fashion as at the previous time steps. Because the target speed is set to be more than half of the sound speed in the figure, the Doppler effect is clearly seen where the sound wavelength in the forward direction is shorter than that in the rear direction and the target position is shifted to forward from each center of the propagation.

Fig. 1 (b) shows target sound detection regions with shaded areas at each time step. If the sensors are distributed evenly and the sensing range, r_s, is set to be large to cover multiple sensors in the range, the sensors in the shaded areas of **Fig. 1** (b) detect target sound as long as the sound signal is higher than the threshold value of the sensors. We note that some sensors in the detection region receive the sound signal from more than one previous target positions. We see some gaps at $t = t_2$ and t_3 in which the sensors do not detect the sound. The gaps are due to the finite time step. The time step, Δt, is set to be 0.005 sec in our simulation.

For the simulation, target motion is described by the following discrete equations.

$$\begin{aligned} \vec{x}^t_{k+1} &= \vec{x}^t_k + \vec{v}^t_k \cdot \Delta t + 0.5 \cdot \vec{a}^t_k \cdot \Delta t^2 \\ \vec{v}^t_{k+1} &= \vec{v}^t_k + \vec{a}^t_k \cdot \Delta t \\ \vec{a}^t_{k+1} &= \vec{a}^t_k \end{aligned} \quad (1)$$

Here, $\vec{x}^t_k, \vec{v}^t_k, \vec{a}^t_k$ are target position, velocity and acceleration at a time index k.

The sound propagation from the target position of each previous time step can be expressed by the following equation.

$$\begin{aligned} d^2_{j,k} &= (x_{j,k} - x^t_j)^2 + (y_{j,k} - y^t_j)^2 \\ d_{j,k} &= (k-j) \cdot V_s \end{aligned} \quad (2)$$

V_s is the sound speed, j (= 1,2,…, k-1) is the index of previous time steps, $d_{j,k}$ is the sound propagation distance at the time of k from the target position at the previous time index j, $\vec{x}_{j,k}$ is the sound wave front position at the time of k propagating from the target position at previous time step j, and $\vec{x}^t_j$ is the target position at the previous time step j.

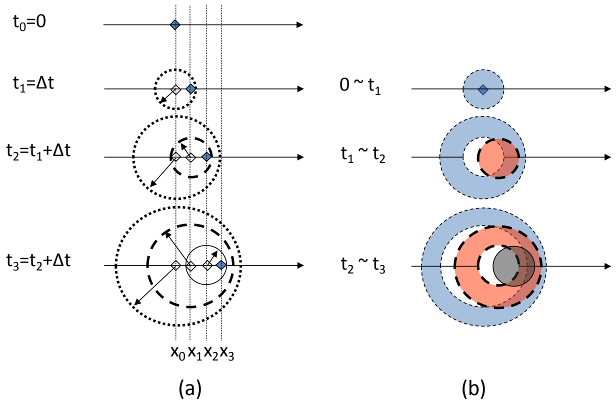

Fig. 1. (a) Sound wave propagation from a moving target. Target position and the center of each sound propagation at each time step are shown with solid and clear diamond respectively. Each arrow shows sound propagation from each propagation center. (b) Sound detection regions at each time step are shown with shades. Different shades are from different propagation centers.

The distance between a sensor and a target position at j can be expressed as follows.

$$d^2_{s,j} = (x_s - x^t_j)^2 + (y_s - y^t_j)^2 \quad (3)$$

where $\vec{x}_s$ is a sensor position and $d_{s,j}$ is the distance from the sensor of s to the target position at the previous time j. Any sensor that satisfies the following conditions at the time step of k can detect the target sound.

$$\begin{aligned} d_{j,k-1} &\le d_{s,j} \le d_{j,k} \\ P_s &= \sum_j P_t / d^\alpha_{s,j} \ge \gamma_s \end{aligned} \quad (4)$$

where P_s is the target sound power at the sensor of s, P_t is the sound power generated by the target, and γ_s is the sensor threshold value. The summation is over the previous time steps j which satisfies the first equation of (4). For our simulation we set $\alpha = 2$ but this can be changed depending on the environmental conditions in the sensor network region. We emphasize here that our tracking result is independent of any sound propagation model as long as the sound model describes sound propagation realistically.

2.3 Tracking Framework

For the passive acoustic sensors with no range measurement capability, the target position can be measured by computing the centroid of the detecting sensors. Two different centroids were considered, one is the unweighted (or geometric) and the other is the weighted centroid. The centroid weighted with detecting sound power is a better target position measurement because the centroid is closer to the true target position. The geometric centroid is the same as the power law detection model which was used by all the previous work mentioned in the introduction.

In our previous paper [16], we showed that the KF is near optimal for detection parameters used in the paper if we use the weighted centroid of detecting sensors as target position measurement.

The Doppler effect on the target position measurement through the centroids was studied in our previous study [16] for target speed 10 ~ 50 m/s and the sensing range of 30 ~ 75 meters. We computed the difference between the target true position ($\vec{x}_t$) and the centroid ($\vec{x}_c$) as

$$\vec{d}_c = \vec{x}_t - \vec{x}_c \tag{5}$$

The difference is considered as a measurement error.

In our previous study, we found that d_c from weighted centroid fluctuates around zero with time during track but the fluctuation center is a little bit shifted to positive with geometric centroid for target speed of 10 m/sec. For target speed 50 m/s, the measurement bias is significant for geometric centroid and the bias is noticeable even for weighted centroid. It was found that the measurement bias is unavoidable even for weighted centroid for target speed greater than 50m/sec. But as most of military ground vehicles have speed much less than 50m/sec we may say that the measurement is almost unbiased with the weighted centroid. We also found that Doppler effect affects the track accuracy through target position measurement bias. For target speed 50 m/s, track position error with the geometric centroid was found to be about four times higher than the position error with weighted centroid. For speed of 10 m/s, the geometric centroid gives the track error about 1.5 times higher than the error of the weighted centroid.

3. Environmental Noise Effects on Tracking

There are two types of noises, one is a random noise caused by sensor electronics and/or continuous noises from the sensing region which are distributed over the whole sensors in the field. The power distribution of this type of noise can be expressed as Gaussian or white noises. In order to filter this kind of noise, a threshold value is set on each sensor. If a detecting sound power is higher than the threshold, the sensor regards the sound as from a target and sends detection information to a fusion center. For simplicity of the problem, we assume that the detection probability is 100% if detecting power is above the threshold. The other type of noise is generated by, for example, passing animals or nearby plants in windy environments which impacts certain local sensors sporadically. These noises are assumed to be distributed over randomly selected sensors and the duration of the noise at each sensor varies from sensor to sensor. If the noise is higher than the sensor threshold, the sensor regards it as coming from a target and sends detection information to a fusion center. Thus, this type of noise will interfere with target sound and degrades the target detection and track accuracy severely. For accurate target tracking in real environments, this type of random noise should be filtered before tracking by the fusion center. In this section, we present an algorithm for mitigating this kind of environmental noises.

3.1 Random Distribution of Noises Over The Sensing Region

The environmental noise is assumed to be distributed randomly over the sensors and the duration of the noise on each sensor varies depending on the origin of noises. For the simulation, we assume that 1 ~ 6% of the sensors deployed in the field detect the noises at any time step and the duration of noises in each sensor varies in 0.1 ~ 1.0 sec. The noise duration is reasonable for the noises generated by some small animal passing, crickets chirping or foliage blowing. This environmental noise model is supported by [23] which performed a statistical analysis of environmental noise measurements and found that the noises are highly non-stationary and non-Gaussian and their duration is less than 1 sec.

The noise power is assumed to fluctuate randomly between one and two times of the threshold value as

$$\aleph_s = \gamma_s (1 + \Re) \tag{6}$$

where $\aleph_s$ is the noise power at a sensor s, γ_s is the sensor threshold value and $\Re$ is a random number generator uniformly distributed between 0 and 1. The set of sensors, *D*, that detect the noise are determined by the random number generator such that

$$D = \{\Re(s) < 0.06, s \in (1,2,...,N_s)\} \tag{7}$$

for 6% noise distribution for example. The total sound power detected by a sensor s is

$$\Im_s = P_s + \aleph_s \tag{8}$$

where P_s is the target sound detected by sensor s (see eq. (4)).

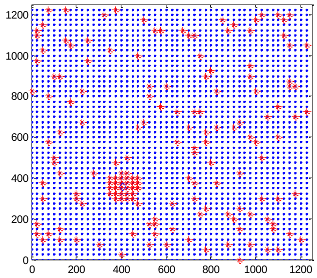

Fig. 2. The target (diamond) and the sensors (red) which detect target sound as well as the noise above the threshold. The blue dots are sensors in the grid form.

Fig. 2 shows a target (diamond) and the sensors (red) that detect the sound above the threshold. The sensors within the target sensing range detect target sound (some of them may also detect the noise). Some of the sensors that are outside the target sensing range detect the noise (red) and are distributed randomly over the sensing region. All the detecting sensors in red report their position to the fusion center which, then, computes the weighted centroid for target position measurement and feeds it to the tracking filter. If the fusion center computes the weighted centroid based on the reported positions as usual, the centroid will be very off from the target true position and the track will diverge. To mitigate the adverse noise effect on track, we need to develop a sophisticated denoising algorithm.

3.2 Quick Mitigation Solution

As a quick mitigation solution [17], we use the sensor position of a highest detecting power for target position measurement assuming that the sensor of highest detecting power is closest to the target position. This assumption may not be valid for targets with sensing range comparable or shorter than the sensor separation. We perform 10 Monte Carlo runs with the random noises for various sensing ranges.

Fig. 3 (a) shows the averaged position track error, <dx>, for each Monte Carlo run with 2% of noise distribution and the target sensing range of 30 meters. Track error fluctuates widely between 0.8 and 13 meters due to the noise over the 10 runs. Fig. 3 (b) shows averaged track error over 10 Monte Carlo runs for 2% and 6% noise distribution and for targets of various sensing ranges. The figure shows that the target sensing range should be equal or larger than 40 meters for accurate tracking with the quick solution.

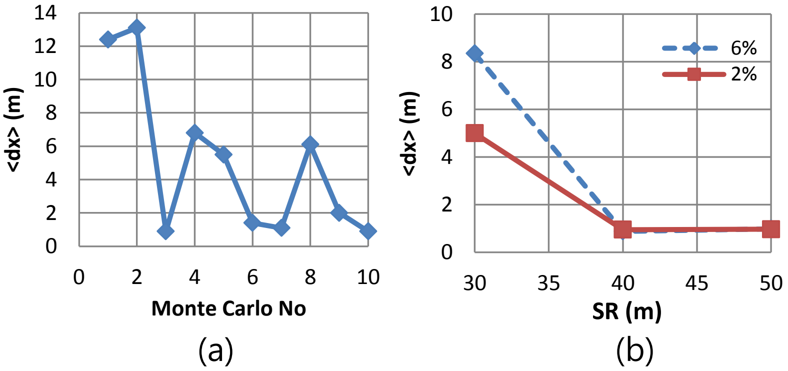

Fig. 3. (a) Averaged track error for each Monte Carlo run with 2% noises distribution and a target with sensing range (SR) 30m and speed V = 20 m/s. (b) Averaged track error over 10 Monte Carlo runs vs. SR for 2% and 6% of noise distribution. Target speed is 20 m/s.

For accurate tracking of targets with sensing range equal or less than the sensor separation we need to develop a better mitigation method. The new mitigation method we developed is described in the next section.

3.3 Algorithm for Random Noise Mitigation

The highly non-Gaussian and non-stationary nature of the noises require a new approach for the noise mitigation. Our denoising algorithm is based on the following assumptions on the noise. First, the noise is distributed randomly over the sensors (see Eq. (7)) and the random distribution changes every Δ_t sec. Second, the sensors that detect target sound are aggregated around the target and the target sound stays in the sensors more than Δ_t sec. For example, the noise generated by a small animal passing with speed 10 m/s and sensing range of 2 meters stays at a sensor less than 0.4 sec and the sound from a target of speed 20 m/s and sensing range of 30 meters stays in the sensor less than 3 sec. We also assume that sometimes the noise stays at a sensor longer time than the target sound. The denoising process we devised is as follows.

Step 1: The first step is to find the sensors that detect sounds higher than the sensor threshold and save the detection information in the following binary form.

$$F_k = (f_1, f_2, ..., f_{N_s})_k \tag{9}$$

where the subscript *k* stands for the time step, N_s is the total number of sensors distributed in the field.

$$\begin{aligned} f_{i,k} &= 1 \text{ if } \Im_i \geq \gamma_i \\ &= 0 \text{ otherwise} \end{aligned} \tag{10}$$

We save the detection information for three consecutive time steps and find the sensors that satisfy

$$f_{i,k-2} \cdot f_{i,k-1} \cdot f_{i,k} = 1 \tag{11}$$

We, then, save the sensor positions as

$$G_k = (\vec{x}_1, \vec{x}_2, ..., \vec{x}_m)_k \tag{12}$$

Here, *m* is the total number of sensors that satisfy Eq. (11) and $\vec{x}_j$ is jth sensor position . All the noises that stay less than $3\Delta_t$ sec at a sensor will be filtered out but some noises and target sound which stay longer than $3\Delta_t$ sec will survive the filtering. To filter out the residual noises, we take the following second step.

Step 2: This step is based on the assumption that the sensors that detect target sound aggregate around the target but the sensors that detect the noise are scattered randomly far from the target.

For a j^{th} sensor where $j \in \{1,2,…, m\}$, we compute the distance, $d_{j,l}$, between the j^{th} and a l^{th} sensors,

$$d_{j,l} = \left|\vec{x}_j - \vec{x}_l\right| \quad (l = 1,2,\ldots,m) \tag{13}$$

We, then, count the number of sensors, L_j, that satisfy

$$L_j = \{ (l=1,2,..., m): d_{j,l} > 3\Delta_s, \}, \tag{14}$$

where Δ_s is the sensor separation.
If $L_j > m/2$, then, the j^{th} sensor is excluded as noise, if not, the j^{th} sensor is included as a target detection

Step3: Most of the noises are filtered out through the two steps described above. But if the target sensing range is less than Δ_s the centroid computed after step 2 sometimes gives a wrong target position measurement by picking up a noise as target sound. To filter the noises further for target sensing range shorter than the sensor separation, we save the centroid computed at the previous time step, $d_{c,k\text{-}1}$, and compare it with the centroid of current time step, $d_{c,k}$. If

$$|d_{c,k\text{-}1}\text{-}d_{c,k}| > 3\Delta_s, \tag{15}$$

the fusion center discard $d_{c,k}$ and go to next time step, otherwise call KF with $d_{c,k}$ as a target position measurement. The psuedo code for the algorithm described above is given as follows.

Algorithm: Single Target Denoising

Inputs: Sensor positions $\vec{x}_s$ (s = 1,…,N$_s$), %N$_s$ ≡ total number of sensors deployed in the field.

```
1. for  k = 1:itmax %time step
2.   Target trajectory in time (Eq. (1)), noise model in time (Eq. (7))
3.   Call detection model (Eq. (4) and(8))
4.   Find the sensors that detect sound $\Im_s > \gamma_s$ and save the detection info in binary form as Eq.(10).
5.   Find the sensors that survive $3\Delta_t$ sec (satisfy eq. (11)) and save their positions as Eq. (12).
6.   for $j = 1 : m$  % $m$≡Total number of sensors that satisfy eq. (11).
7.     L($j$)=0
8.     for  $l = 1 : m$
9.       Compute distance between $j^{th}$ and $l^{th}$ sensors using eq. (13): $d_{j,l}$.
10.        if $d_{j,l} > 3\Delta_s$
11.          L($j$)=L($j$)+1
12.        end
13.    end
14.    if L($j$) ≦ $m$/2
15.      Save $\vec{x}_j$ in $X_k$  %k is the current time index
16.    end
17.  end
18.  Compute centroid $d_{c,k}$ from $X_k$
19.  if $X_{k-1}$ = { } go to 1
20.  if $|d_{c,k-1} - d_{c,k}| > 3\Delta_s$ go to 1
21.  else call KF with $d_{c,k}$ as an measurement
22.  $d_{c,k-1} = d_{c,k}$; $X_{k-1} = X_k$
23. end
```

3.3.1 Results of the Denoising

For testing the algorithm, we simulate the following three different noises depending on the noise duration at the sensors and the percentage of the sensors that detect the noises. Noise 1 has 0.1 sec duration at the sensors randomly selected from 6% of the sensors in the field. Noise 2 is a combination of the noise of 0.1 sec duration at the sensors randomly selected from 5% of the sensors and the noise of 0.5 sec duration at another randomly selected from 1% of the sensors. Noise 3 is a combination of the noise of 0.1 sec duration at the randomly selected from 5% of the sensors and the noise of 1 sec duration at another randomly selected from 1% of the sensors. MatLab is used for our simulation and the simulation setting for the sensor network is described in Section 2.1.

Fig. 4 shows how each step filters the noises for a target with sensing range 75 meters. **Fig. 4** (a) is the detecting sensors before filtering. After step1, many of the noises are filtered out as shown in **Fig. 4** (b). Step 2 filters out remaining noises and the sensors which detect target sound remain as shown in **Fig. 4** (c). As the target sensing range is much larger than the sensor separation, all the noises are filtered out after step2. If the sensing range is comparable or shorter than the sensor separation, step3 is needed to filter out remaining noises as will be shown in **Fig. 6**.

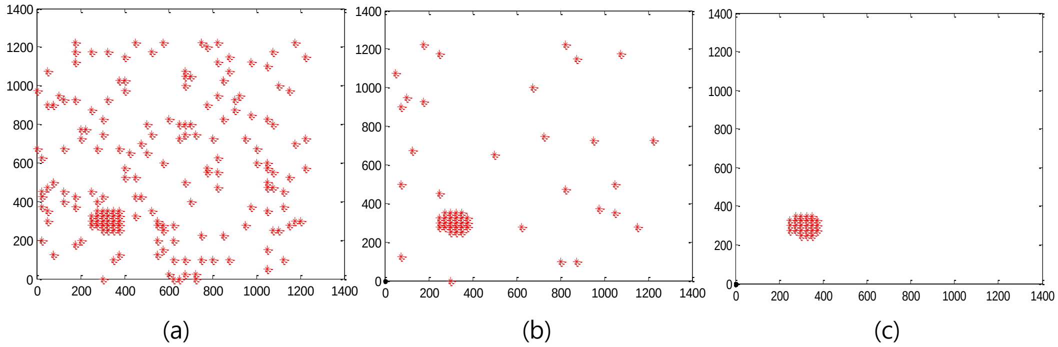

Fig. 4. Noise filtering by the algorithm. (a) Before noise filtering. (b) After the step 1 filtering. (c) After the step 2 filtering.

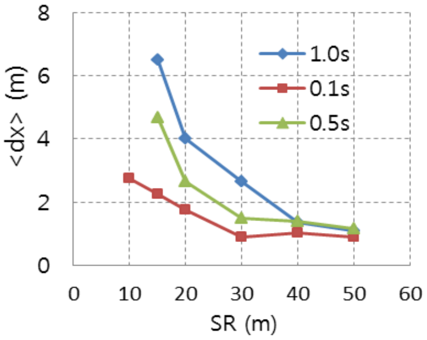

Fig. 5. Averaged track errors, <dx>, vs. sensing range (SR) for the three different noise durations and target speed V = 20 m/s.

For the three different noises mentioned above, we perform 10 Monte Carlo runs for sensing range SR = 10, 15, 20, 30, 40, and 50 meters and compute the averaged track position error, <dx>. The result is shown in **Fig. 5**. For the noise duration 0.1 sec, the algorithm works so well that <dx> is less than 3 meters for SR=10 m. But as the noise duration increases to 0.5 and 1 sec, a target with SR=10 m cannot be tracked because the target sound is too low to extract from the noises. For a target with SR=15 m, the algorithm filters out the noise and tracks the target with averaged track error 4.7 and 6.5 meter for noise duration 0.5 and 1 sec respectively but the track time interval increases up to 11 seconds at step3 of the algorithm.

Fig. 6 shows the position measurement error defined in Eq. (5) vs. tracking time for targets with (a) SR=40 m and (b) 15m and for Noise 3. The data was obtained after passing through Step3 of the algorithm. For SR=40 m, the averaged track time interval is 0.65 sec but the averaged track time interval increases to 3.3 sec (maximum interval is 11 sec) for SR=15 m. The figure shows that the step3 of the algorithm skips the track often when the target sensing range of 15 meters is shorter than the sensor separation of 25 meters.

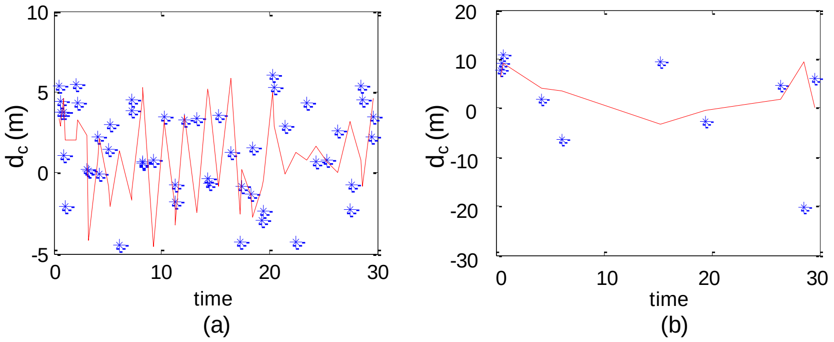

Fig. 6. The position measurement error, d_c, defined in eq. (5) at each track time for Noise 3. (a) SR=40 m. (b) SR=15 m.

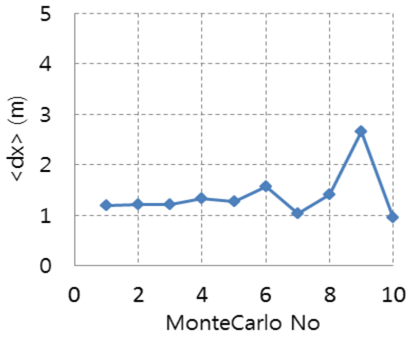

Fig. 7. Averaged position track error, <dx>, for each Monte Carlo run for a target with SR=30 m and V=20 m/s.

Fig. 7 shows the variation of the averaged track position error, <dx>, for each Monte Carlo run for SR=30m and Noise 3. The averaged track position error stays near 1.2 m through the 10 runs except 2.7 m at 9th run. The result is compared with **Fig. 3** (a) which shows wide fluctuation of <dx>, 50% of the runs have <dx> > 5 meters.

As the previous works deal only with Gaussian and stationary noises, it is not possible for us to compare their works with our algorithm which deals with non-Gaussian and non-stationary noises. For the computational complexity of denoising, we measured the denoising computing time for single run to be $1.2 \cdot 10^{-3}$ sec. The results shown in this section demonstrate that our noise mitigation algorithm works well for the three different noises we modeled. Further tests are underway with more diverse noise models and multiple targets to refine our noise mitigation algorithm.

4. Discussion

We have studied the target tracking in noisy WSN environments with acoustic sensors. The sensors are assumed to have no capability of communicating between sensors, no range measurement and no knowledge of the source of the detecting sound. The detecting sound can be from a target and/or from environmental noises. As long as the detecting sound is higher than the sensor threshold value, the sensor sends the detection information to the fusion center regardless of the origin of the sounds for further processing. Thus, the acoustic WSNs are prone to be suffered from various environmental noises for tracking target. For successful implementation of large scale acoustic WSNs in real world, the mitigation algorithms for these environmental noises should be developed.

The main difference between our work and the previous studies mentioned in the introduction is that we have developed a realistic model for target sound propagation and detection that accounts for the Doppler effect and studied various environmental effects including environmental noise mitigation. These environmental noises will be common in real battle field where the acoustic WSN will be deployed. We have proposed an algorithm for the environmental noise mitigation and tested the algorithm for various noise models. The mitigation algorithm is demonstrated to be effective even for target sound power sometimes lower than the sensor threshold value. It is found that noise duration at a sensor is a crucial factor that affects the success of the mitigation algorithm. Large scale implementation of WSNs in real world often experiences network faults and faces multiple targets crossing each other [22] . Detailed study for the environmental effects on multiple target tracking is currently under way including noise mitigation. The effect of network faults on target tracking will be investigated in the future.

References

[1] K. Mechitov, S. Sundresh, Y. Kwon, and G. Agha, “Cooperative Tracking with Binary Detection Sensor Networks,” *Technical Report UIUCDCS-R-2003-2379*, University of Illinois at Urbana-Champaign, September 2003.

[2] W. Kim, K. Mechitov, J. Choi, and S. Ham, “On Target Tracking with Binary Proximity Sensors,” in *Proc. of IEEE Fourth Int’l Symp. Information Processing in Sensor Networks*, Los Angeles, USA, pp. 301-308, April 2005.

[3] Z. Wang, E. Bulut, and B. Szymanski, “A Distributed Cooperative Target Tracking with Binary Sensor Networks,” in *Proc. of IEEE Int’l Conf. on Communications (ICC ‘08), Communication Workshop*, Beijing, China, pp. 306-310, May 2008.

[4] Z. Wang, , E. Bulut, and B. Szymanski, “Distributed Target Tracking with Imperfect Binary Sensor Networks,” in *Proc. of IEEE Global Telecommunications Conf. (Globecom ’08)*, New Orleans, USA, pp. 1-5, Nov. 2008.

[5] A. Ribeiro, G. B. Giannakis, and S. I. Roumeliotis, “SOI-KF: Distributed Kalman Filtering with Low-cost Communications Using the Sign of Innovations,” *IEEE Trans. Signal Processing*, vol. 54, no. 12, pp. 4782-4795, Dec. 2006. Article (CrossRef Link)

[6] P.M. Djuric, M. Vemula and M. Bugallo, “Signal Processing by Particle Filtering for Binary Sensor Networks,” in *Proc. of IEEE 11th Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop*, Taos Ski Valley, NM, USA, pp. 263-267, Aug. 2004.

[7] P.M. Djuric, M. Vermula, and M. Bugallo, “Target Tracking by Particle Filtering in Binary Sensor Networks,” *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2229-2238, June 2008. Article (CrossRef Link)

[8] N. Ahmed, M. Rutten, T. Bessell, S. S. Kanhere, N. Gordon, and S. Jha, “Detection and Tracking Using Particle-Filter-Based Wireless Sensor Networks,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 9, pp. 1332-1345, September 2010. Article (CrossRef Link)

[9] J. Teng, H. Snoussi, and C. Richard, “Decentralized Variational Filtering for Target Tracking in Binary Sensor Networks,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 10, October 2010. Article (CrossRef Link)

[10] D. Li, K. Wong, Y. Hu and A. Sayeed, “Detection, Classification and Tracking of Targets in Distributed Sensor Networks,” *IEEE Signal Processing Magazine*, vol. 19, issue 2, pp.17-29, March 2002.

[11] R. Brooks, P. Ramanathan, and A. M. Sayeed, “Distributed Target Classification and Tracking in Sensor Networks,” *Proc. of the IEEE*, vol. 91, no. 8, pp. 1163-1171, Aug. 2003. Article (CrossRef Link)

[12] A. Doucet, B. Vo, C, Andrieu, and M. Davy, “Particle Filtering for Multi-target Tracking and Sensor Management,” in *Proc. of Fifth Int’l Conf. Information Fusion*, Annapolis, USA, pp. 474-481, vol. 1, Nov. 2002.

[13] W. Chen, J. Hou, and L. Sha, “Dynamic Clustering for Acoustic Target Tracking in Wireless Sensor Networks,” *IEEE Transactions on Mobile Computing*, vol. 3 no. 3, pp. 258-271, July 2004. Article (CrossRef Link)

[14] J. H. Kotecha and P. M.. Djuric, “Gaussian Particle Filtering,” *IEEE Trans. Signal Processing*, vol. 51, no. 10, pp. 2592-2601, Oct. 2003. Article (CrossRef Link)

[15] J. H. Kotecha and P. M.. Djuric, “Gaussian Sum Particle Filtering,” *IEEE Trans. Signal Processing*, vol. 51, no. 10, pp. 2602-2612, Oct. 2003. Article (CrossRef Link)

[16] Y. K. An, S. -M. Yoo, C. An, and B. E. Wells, “Doppler Effect on Target Tracking in Wireless Sensor Networks,” in *press to Elsevier’s Communications*. http://dx.doi.org/10.1016/j.comcom.2013.01.002 Article (CrossRef Link)

[17] Y.K. An, S. -M. Yoo, C. An, and B. E. Wells, “Environmental Effects on Target Tracking in Wireless Sensor Networks,” in *Proc. of 25th Int’l Conference on Computer Applications in Industry and Engineering*, New Orleans, LA,USA, pp. 151, Nov, 2012

[18] G.Isbitiren and O. B. Akan, “Three-Dimensional Underwater Tracking With Acoustic Sensor Networks,” *IEEE Trans. On Vehicular Tech*, Vol. 60, No. 8, October, 2011.

[19] X. Sheng and Y. Hu, “Maximum Likelyhood Multiple-Source Localization Using Acoustic Energy Measurements with Wireless Sensor Networks”, IEEE Trans. On Signal Processing, Vol. 53, No. 1, pp 44-53, Jan. 2005 Article (CrossRef Link)

[20] M. Morelande and B. Moran, “Multiple target detection and tracking with a sensor network,” in *Proc. of 10th Int’l conf. on Information Fusion*, Canada, July, 2007

[21] G. H. Jajamovich and X. Wang, “Joint Multitarget Tracking and Sensor Localization in Collaborative Sensor Networks,” *IEEE Trans. Aerospace and Electronic Sys*, pp 2361-2375, Vol. 47, No. 4, Oct. 2011 Article (CrossRef Link)

[22] S. Oh, L. Schenato, P. Chen, and S. Sastry, “Tracking and Coordination of Multiple Agents Using Sensor Networks: System Design, Algorithms and Experiments,” in *Proc. of the IEEE*, Vol. 95, No. 1, pp 234-254, Jan. 2007 Article (CrossRef Link)

[23] A. Quach and K. Lo, “Automatic Target Detection using a Ground-based Passive Acoustic Sensor,” in *Proc. of the 1999 Conference on Information, Decision and Control*, pp. 187-192, 1999.

Youngwon Kim An received the BS and MS degree in physics from Seoul National University, Seoul, Korea in 1971 and 1974 respectively and Ph.D. degree in computer engineering from the University of Alabama in Huntsville in 2013. From 2007 to present, as a software engineer staff at Lockheed Martin Space System Company, she has been implementing tactical tracking algorithm. From 1999 to 2007, as a principal engineer at Northrop Grumman Corporation, she developed tactical software. From 1981 to 1999, she developed commercial/defense software as a software developer/consultant at NCR, CSC, Intergraph and SAIC. Her research interests include target tracking/track fusion and its application in wireless sensor networks. Dr. An is a member of IEEE.

Seong-Moo Yoo is an Associate Professor of Electrical and Computer Engineering at the University of Alabama in Huntsville (UAH). Before joining UAH, he was an Assistant Professor at Columbus State University, Columbus, Georgia – USA. He earned MS and PhD degrees in Computer Science at the University of Texas at Arlington. His research interests include computer network security, wireless network routing, and parallel computer architecture. He has co-authored over 90 scientific articles in refereed journals and international conferences. He is a senior member of IEEE and a member of ACM.

Changhyuk An received the BS and MS degree in physics from Seoul National University, Seoul, Korea in 1969 and 1974 respectively and Ph. D. degree in physics from the University of Tennessee, Knoxville, Tennessee in 1979. His research interest includes the tracking in wireless sensor network, the reflection and emission polarization signature modeling, hyper-spectral data compression, magnetic fusion plasma research, solar flare modeling/simulation, and clutter mitigation for missile defense. He published nearly 60 papers in refereed journals, conference proceedings, and military proceedings. Dr. An is a member of SPIE.

B. Earl Wells received the B.S., M.S. and Ph. D degree in electrical engineering from the University of Alabama in Tuscaloosa in 1983, 1988 and 1992, respectively. Since September 2005, he has served as Professor in Electrical and Computer Engineering Department of the University of Alabama in Huntsville, Huntsville, Alabama. Prof. Wells' research interests include real-time systems, computer architecture, and parallel processing. He has co-authored over 80 scientific articles in refereed journals and international conferences. Prof. Wells is a member of IEEE.