Wireless Sensor Network, 2012, 4, 264-272
http://dx.doi.org/10.4236/wsn.2012.411038 Published Online November 2012 (http://www.SciRP.org/journal/wsn)

A Practical Target Tracking Technique in Sensor Network Using Clustering Algorithm

Luke K. Wang, Chien-Chang Wu
Department of Electrical Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Chinese Taipei
Email: lwang@mail.ee.kuas.edu.tw, baseball160404@yahoo.com.tw

Received August 15, 2012; revised September 16, 2012; accepted September 27, 2012

ABSTRACT

Sensor network basically has many intrinsic limitations such as energy consumption, sensor coverage and connectivity, and sensor processing capability. Tracking a moving target in clusters of sensor network online with less complexity algorithm and computational burden is our ultimate goal. Particle filtering (PF) technique, augmenting handoff and K-means classification of measurement data, is proposed to tackle the tracking mission in a sensor network. The handoff decision, an alternative to multi-hop transmission, is implemented for switching between clusters of sensor nodes through received signal strength indication (RSSI) measurements. The measurements being used in particle filter processing are RSSI and time of arrival (TOA). While non-line-of-sight (NLOS) is the dominant bias in tracking estimation/accuracy, it can be easily resolved simply by incorporating K-means classification method in PF processing without any priori identification of LOS/NLOS. Simulation using clusters of sensor nodes in a sensor network is conducted. The dependency of tracking performance with computational cost versus number of particles used in PF processing is also investigated.

Keywords: Sensor Network; Handoff Scheme; Particle Filter; K-means Clustering; NLOS

1. Introduction

Sensor networks can be applied in a variety of areas such as target tracking, environment monitoring, military surveillance, medical applications, etc. [1,2]. However, the majority of sensor node platforms are operated using the low power 802.15.4 wireless technology, and its transmission range is extremely limited especially in an indoor environment [3]. The measurements used in the estimate of mobile locations in sensor network may include received signal strength, time difference of arrival (TDOA), time of arrival (TOA), and angle of arrival (AOA) [4]. Eventually, the above propagation measurement scenarios are divided into two categories, line-of-sight (LOS) and non-line-of-sight (NLOS). In multi-path propagation environments, particularly indoors or urban areas, the LOS path between nodes may be obstructed [5]. However, the NLOS propagation usually leads to a positive bias and causes a serious error in the results of tracking estimation [6].

Lots of attentions have been focused on the identification of LOS/NLOS condition and the mitigation of NLOS bias. A simple hypothesis test has been conducted to tell whether it's LOS or NLOS by the fact that the standard deviation of the range measurement of NLOS is presumably larger than the LOS' [6]. Using the individual measurement detection (IMD), basically a hypothesis test to identify whether an incoming measurement is LOS or NLOS and those NLOS ones being discarded, to do target tracking is proposed [7]; extended Kalman filter (EKF) algorithm is applied accordingly to do the target tracking job. The noise modeling of LOS/NLOS is formulated by a two-state Markov process, and the degree of contamination by NLOS errors is correlated with the transition probability of the Markov process. A disadvantageous effect has been indicated, the number of selected LOS measurements by IMD is different at each step, resulting in dimension validation of the reconstructed LOS measurement vector being dynamic; slow convergence rate is also appearing in the tracking results when using EKF. A modified Kalman filtering technique, adopting the modification at the measurement update stage, is introduced to tackle the NLOS identification/mitigation problem [8]. The NLOS positive bias is estimated directly throughout the constrained optimization method; no prior distribution knowledge of the NLOS error is needed, as claimed by the authors.

Our proposed tracking algorithm utilizes clusters of sensor network with handoff scheme in a heterogeneous wireless environment. A handoff scheme specified in IEEE 802.11 network is the process whereby a mobile station shifts its association from one access point (AP) to another [9]. When a mobile station moves out of the range of an AP and into another's, the handoff occurs during which there is an exchange of management frames [10].

On the other hand, clustering analysis is the method of unsupervised learning. It may include two parts, namely, partitional clustering and hierarchical clustering. Partitional clustering is a set of objects classified into K clusters without hierarchical structure [11]. The clustering method with PF can reduce the degradation of NLOS propagation efficacy in tracking estimation results. Meanwhile, handoff scheme is applied in the clusters of sensor network. In clusters of sensor network, each cluster has numbers of sensor nodes, including TOA and RSSI sensors. **Figure 1** shows the proposed architecture that illustrates an event of target tracking in clusters of sensor network.

The general approach to processing the LOS/NLOS signals in cellular communication network is to determine whether it's a LOS or NLOS condition. Even the fuzzy inference scheme is introduced to tell whether it is LOS or NLOS before any processing jobs can be done [10]. It is combined with adaptive Kalman filter to establish mobile location estimator; undoubtedly, system and computational complexities are increasing so tremendously, hindering the potentials in real-time applications.

We present an architecture which utilizes clustering analysis method with particle filter (PF) to track a moving target. Particle filter implements sequential Monte Carlo simulation based on a set of particles to construct prior density with associated weights for the approximation of posterior density.

The beneficial features of the proposed scenario are listed as bellows,

- Intuitive but feasible for real-time applications due to its low system and computational complexity attributes

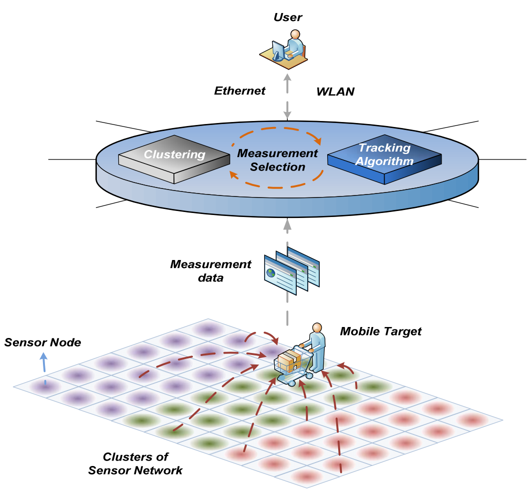


Figure 1. The proposed architecture in clusters of sensor network.

- No need to identify whether it is under LOS or NLOS condition prior to any processing of the target location estimation
- Only RSSI and TOA sensor measurements are required, practical and low-cost; fingerprinting scheme, the build-up of a "radio map" database [6], and extra hardware (sophisticated measurement devices) are not employed, always leading to a low complexity, cost-effective scenario.
- Both K-means clustering and hand-off schemes are incorporated into the particle filter, minimizing the system complexity to the most.

This analysis is set up as follows. The target's motion model and the associated measurement equation and NLOS propagation are described in Section 2. Particle filter is described in Section 3. Section 4 discusses the proposed tracking algorithm which includes the handoff scheme. The proposed tracking algorithm with clustering method is derived in Section 5. The simulation and performance analysis are presented in Section 6. Section 7 includes the conclusion.

2. Motion Models

We assume a mobile target's movement is on a two-dimensional (2-D) plane. Besides, the measurement of signal propagation with LOS/NLOS conditions may be modeled as a two-state Markov process if the performance at transient stage is under investigation.

2.1. Target Model

The moving target's state vector is defined as $x_{1,k} = \left[x_k \ \dot{x}_k \ y_k \ \dot{y}_k\right]^{\mathrm{T}}$, consisting of position and velocity at a time instant k, where $()^T$ stands for transpose operation of matrix. The target's motion is modeled as

$$\mathbf{x}_{1,k} = A_1 \mathbf{x}_{1,k-1} + w_{1,k-1} \tag{1}$$

where

$$A_1 = I_2 \otimes A_s$$

$$A_s = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}$$

and I_2 is the 2 × 2 identity matrix; ⊗ is the Kronecker product operator; A_1 is the state transition matrix; T_s is the sampling time; and $w_{1,k-1}$ is a zero mean white Gaussian noise process with covariance matrix Q_1, *i.e.*, $w_{1,k-1} \sim N\left(0, Q_1\right)$. The covariance matrix Q_1 is

$$Q_1 = E\left[w_{1,k} w_{1,k}^{\mathrm{T}}\right] = q_1 I_2 \otimes Q_s$$

$$Q_s = \begin{bmatrix} T_s^3/3 & T_s^2/2 \\ T_s^2/2 & T_s \end{bmatrix}$$

where q_1 is a scalar which determines the intensity of the process noise and E[ ] is the expectation.

An alternative motion model may be described as below,

$$\boldsymbol{x}_{2,k} = A_2 \boldsymbol{x}_{2,k-1} + w_{2,k-1} \tag{2}$$

Here $\boldsymbol{x}_{2,k} = \left[x_k \ \dot{x}_k \ \ddot{x}_k \ y_k \ \dot{y}_k \ \ddot{y}_k \right]^{\mathrm{T}}$; $\left(x_k, y_k \right)$ represents the position of the target; $\left(\dot{x}_k, \dot{y}_k \right)$ and $\left(\ddot{x}_k, \ddot{y}_k \right)$ denote the velocity and acceleration of the moving target along the x and y directions, respectively. The transition matrix A_2 is given by

$$A_2 = I_2 \otimes \begin{bmatrix} 1 & T_s & T_s^2/2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$$

Here $w_{2,k-1} \sim N\left(0, Q_2\right)$. The covariance matrix Q_2 is available by transforming a continuous-time stochastic target model into an equivalent discrete-time model [12]:

$$Q_2 = q_2 I_3 \otimes Q_{2s}$$

$$Q_{2s} = \begin{bmatrix} T_s^5/20 & T_s^4/8 & T_s^3/6 \\ T_s^4/8 & T_s^3/3 & T_s^2/2 \\ T_s^3/6 & T_s^2/2 & T_s \end{bmatrix}$$

Here q_2 is a scalar determining the intensity of the process noise.

2.2. Measurement Model

Assume that our sensor node is stationary, and its state vector is

$$s_i^j = \left[s_{x,i}^j \ s_{y,i}^j \right]$$

the ith sensor node in the jth cluster zone. The measurement equation corresponding to TOA data can be formulated as:

$$c \times \tau_i = D_{TOA} = \left\| \boldsymbol{x}_k - s_i^j \right\| \tag{3}$$

where c is the propagation speed of the transmitted signal; $\mathbf{x}_k$ is the state vector of the moving target; and τ_i is the propagation time. The measurement equation with NLOS propagation error is shown as below,

$$z_k = h\left(\boldsymbol{x}_k\right) + \mathrm{NLOS}_k + v_k . \tag{4}$$

The measurement function $h(\bullet)$ models the TOA through the ith sensor; v_k is the measurement noise process independent of $w_{1,k-1}$, $w_{2,k-1}$, and any other sensor noise source; and the measurement noise is modeled by a zero mean Gaussian white noise $N(0,R)$. NLOS_k is the NLOS propagation error at the sampling instant k, modeled by a two modes/states Markov process [13]. Therefore the propagation error, NLOS_k, is

$$\mathrm{NLOS}_k = \begin{cases} 0, & \text{if LOS present} \\ nlos, & \text{if NLOS present.} \end{cases} \tag{5}$$

Here the random variable *nlos* has been modeled by a large scale of covariance value, usually hundred of meters, of statistical distribution. An alternative measurement model may be employed [12],

$$z_k = h\left(\boldsymbol{x}_k\right) + b_k \cdot nlos + G_k n_k \tag{6}$$

where

$$b_k = \begin{cases} 0, & \text{if LOS presents} \\ 1, & \text{if NLOS presents} \end{cases}$$

$$G_k = \begin{cases} \sigma_m, & \text{if LOS presents} \\ \left(\sigma_m + \sigma_{\mathrm{NLOS}}\right)^{0.5}, & \text{if NLOS presents} \end{cases}$$

b_k is a binary sequence, modeled by a two-state, LOS and NLOS, discrete-time Markov chain process and σ_m is the typical standard deviation of LOS.

3. Particle Filter Processing

Particle filter is based on sequential importance sampling (SIS) to estimate the system state with some numbers of particles with their associated weightings. Particle filtering can be arranged to process the nonlinear and non-Gaussian systems, and it has become an important alternative to the extended Kalman filter (EKF) [14]. There are five processing stages in the implementation of PF, *i.e.*, initialization, particle propagation, weighting computation, resampling, and estimation [14]. First of all, particles are drawn from a proposal distribution at the initialization step, where each particle possesses initial weights. Then, the particle propagation is based on the EKF to produce next state's distribution. At the weighting computation step, particle weights are set equal to the ratio of probability distributions from the proposal distribution. The update equation of particle weighting can be shown as [14]

$$w_k^i \propto w_{k-1}^i \frac{p\left(z_k \middle| x_k^i\right) p\left(x_k^i \middle| x_{k-1}^i\right)}{q\left(x_k^i \middle| x_{k-1}^i, z_k\right)} \tag{7}$$

where the quantity $(\bullet)^i_k$ denotes it is the ith particle at kth sampling time instant; $q(\bullet)$ is an importance density, which can generate particles x^i; and the likelihood function $p\left(z_k \middle| x_k^i\right)$ is formulated by multivariate Gaussian distribution,

$$p\left(z_k \middle| x_k\right) = \frac{1}{\sqrt{(2\pi)^n \left|R_k\right|}} \exp\left(-\frac{1}{2}\left(z_k - \hat{z}_k\right)^T R_k^{-1} \left(z_k - \hat{z}_k\right) \right) \tag{8}$$

where $\hat{z} = h\left(\hat{x}_k\right)$ with (^) standing for estimate value. Here we utilize resampling strategy to eliminate particles with low weightings, and duplicates particles with larger weighting. At the final step, it needs to calculate the center of gravity from a group of samples. The PF algorithm is summarized as below,

Particle Filter Algorithm

Step 1. Initialization

Draw initial samples $x_0^i \sim q(x_0)$.

Set the weighting of particles, w_0^i, equal to $1/N$ where N is the number of particles used in PF processing.

Step 2. Particle Propagation

Predict the next state of particles and update each particle by using EKF technique.

Step 3. Weighting Computation

Update the weightings with likelihood function $w_k^i \propto w_{k-1}^i p\left(z_k \middle| x_k^i\right)$.

Normalize the weighting for each particle

$$\tilde{w}_k^i = \frac{w_k^i}{\sum_{i=1}^{N} w_k^i}.$$

Step 4. Resampling

if $N_{\text{eff}} < N_{\text{threshold}}$

$$\left[\left\{x_k^i, w_k^i\right\}_{i=1}^{N}\right] = \text{resampling}\left[\left\{x_k^j, w_k^j\right\}_{j=1}^{N}\right]$$

else

$$\left\{x_k^i, w_k^i\right\}_{i=1}^{N}$$

end

where $N_{\text{eff}}\left(= N\big/\left(1+\text{var}\left(w_k^i\right)\right)\right)$ is the effective number of particles and var(•) stands for taking the variance while $N_{threshold}$ is the prescribed threshold, usually chosen as 2/3 of the particle number.

Step 5. Estimation

Calculate the mean of particles with their associated weightings, $\hat{x}_k \approx \sum_{i=1}^{N} w_k^j x_k^i$.

4. Tracking Algorithm

The handoff scheme is applied to clusters of sensor network, using the received signal strength measurements generated by the moving target, and each cluster of RSSI sensors provides the signal strength indications to switch on/off the handoff.

For formulating the signal strength measurements, let α_k^j be the signal strength received by a moving target from jth RSSI sensor at the kth sampling instant. The received signal strengths can be modeled as a function of the distance plus a logarithmic of the shadowing component [15], *i.e.*,

$$\alpha_k^j = p_k^j + \beta_k^j \text{ dbm} \tag{9}$$

$$p_k^j = \mu_j - 10\gamma \log d_k^j \tag{10}$$

where p_k^j is the local signal power at the kth sampling instant; γ is the path loss index; μ_j is a constant determined by transmitted power, wavelength, and antenna gain of the jth RSSI sensor; d_k^j is the distance between the moving target and the jth RSSI sensor; and β_k^j is the logarithm of the shadow fading, modeled by a zero mean Gaussian random process.

Handoff is triggered only if the current signal strength drops below a user-defined threshold Δ, and any other RSSI sensor's strength is stronger than that of the current one. Here E_k^1 is defined as the event with handoff being triggered, and E_k^0 is the non-handoff situation [15,16], that is,

$$\begin{cases} E_k^1 & \text{if } \alpha_r \geq \alpha_c \\ E_k^0 & \text{if } \alpha_r < \alpha_c \end{cases} \tag{11}$$

where α_c is the received signal strength at moving target due to current RSSI sensor and α_r is the received power at moving target owning to a reference RSSI sensor other than the current one. The conditional cost function is therefore defined

$$C\left(\alpha_r \middle| E_k^1\right) = \begin{cases} 1, & \text{if } \alpha_c < \Delta \\ 0, & \text{if } \alpha_c \geq \Delta \end{cases} \tag{12}$$

Handoff will be activated starting from the current RSSI sensor to the candidate one, when the cost function $C\left(\alpha_r \middle| E_k^1\right)$ is equal to 1 [15]. If both Equations (11) and (12) are not satisfied, it means that there is no handoff needed.

The candidate cluster of sensor network is chosen by handoff decision, and measurements can be expressed as $z_k = z_k^j$. Here the measurements follow Equation (4), and the given TOA measurement would be used in particle filter processing. Through PF processing, likelihoods have to be changed accordingly with handoff decision for each particle, $p\left(z_k \middle| x_k\right) = p\left(z_k^j \middle| x_k^i\right)$.

5. Tracking with K-Means Algorithm

The use of K-means method is to cluster the measurement residual to be used in PF. The flowchart of tracking algorithm with clustering method is shown in **Figure 2**.

5.1. K-means Clustering Method

K-means clustering is an unsupervised learning, computationally efficient for large datasets with numeric. Initially, K samples, serving as the initial centroids, are chosen at random from the whole sample space to approximate the centroids of initial clusters. The cluster centroid is typically the mean of the data in the cluster.

Let $\boldsymbol{y} = \{y_1, y_2, \cdots, y_L\}$ be the dataset; m_l is the centroid of cluster C_l with N_l data points. The calculation of the centroid of clusters is described as below,

$$m_l = \frac{1}{N_l} \sum_{\boldsymbol{y} \in C_l} \boldsymbol{y}, l = 1,2,\cdots,k. \tag{13}$$

where k is the number of clusters. First, initialize the

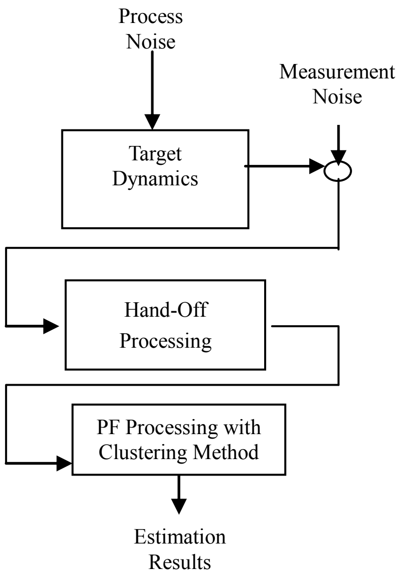


Figure 2. The block diagram of tracking estimation with clustering analysis.

number of centroids k, specified by user and indicating the desired number of clusters. Each data point is assigned to the nearest cluster centroid. After assigning all data points, we recalculate the position of the k centroids. After all, the whole process iteratively updates the centroids until no substantial changes of positions of all k centroids for each cluster. K-means clustering process is directed by an objective function. The sum of the squared error function is often served as an objective function,

$$J = \sum_{l=1}^{k} \sum_{y \in C_L} \| \boldsymbol{y} - m_l \|^2. \tag{14}$$

Here J is the sum of squared error of measurement residual data. Note that the number of clusters k has to be selected first, due to the fact that choosing a different k will results in different values for J.

5.2. Tracking with Clustering Method

The processing of particle evolution in PF actually updates each particle by using EKF algorithm. After performing the clustering of the measurement residual, we choose one of the dataset clusters, having the smallest average value. Here the dataset cluster is the part of measurement residual dataset, and each cluster is chosen by K-means clustering algorithm, which can be expressed as Equations (17) and (18). The chosen dataset is utilized in extended Kalman filter, the intermediate step of the PF algorithm. The proposed method of K-means clustering with extended Kalman filter is described as follows:

1) *Prediction*

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} \tag{15}$$

$$P_{k|k-1} = AP_{k-1|k-1}A^{\mathrm{T}} + Q \tag{16}$$

2) *Residual clustering*

$$e_k = z_k - h\left(\hat{x}_{k|k-1}\right) \tag{17}$$

$$\varepsilon_k = k\_\mathrm{means}\left(e_k\right) \tag{18}$$

3) *Correction*

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \arg\mathrm{Min}\left[e_k\left(\varepsilon_k\right)\right] \tag{19}$$

$$K_k = P_{k|k-1}H^{\mathrm{T}}\left(HP_{k|k-1}H^{\mathrm{T}} + R\right)^{-1} \tag{20}$$

$$P_{k|k} = \left[I - K_k H\right]P_{k|k-1} \tag{21}$$

where H is the Jacobian of measurement matrix which role is to correctly propagate the relevant component of the measurement information for the Kalman gain K_k [2,17]; e_k is the data of measurement residual for clustering analysis; ε_k is the clustering dataset for correction step; $P_{k|k-1}$ is the error covariance matrix for the state x, processing at the time k given the prior value, P_{k-1}; and *k_means*(•) is the function of K-means clustering processing. The favorable clustering analysis can classify the measurement residuals with NLOS noise propagation, although possibly eliminating the residual data of LOS condition.

6. Simulation

To examine the applicability of the proposed tracking algorithm, simulations are performed on a two-dimensional plane with 9 clusters of sensor network system. The target is moving along a random trajectory with varying velocity and acceleration; hence, the target can move freely to any cluster. The positions of TOA sensors, s_1, s_2, $\cdots$, s_n, are randomly deployed, and each cluster contains 20 TOA sensors. Besides, the coordinates of RSSI sensors are assumed to be at C^1(70,50), C^2(330,310), C^3(330,310), C^4(–190,320), C^5(–190,–220), C^6(70,400), C^7(70,–300), C^8(–270,50), and C^9(420,50) in each cluster, respectively. In addition, each TOA sensor may have chance to involve NLOS propagation. Here the NLOS propagation errors are modeled as a random variable, following the statistical distribution defined in Section 2. **Figure 3** is showing a realization of the random NLOS error distribution at a TOA sensor.

6.1. Results of Tracking Estimation

The motion model for simulation follows Equation (2), including position, velocity, and acceleration as the system state. The simulation parameters are summarized in **Table 1**.

The initial coordinates for the simulated and true (or actual) moving targets are $\hat{x}_o = \left[500, -10, 0, 400, -25, 0\right]^{\mathrm{T}}$

and $x_o = [410, -20, 2, 500, -15, 1]^T$, respectively. The initial error covariance is defined by a 6-by-6 diagonal matrix $P_0 = \mathrm{diag}\{1000, 100, 10, 1000, 100, 10\}$, and the measurement covariance matrix R is defined as a 20-by-20 diagonal matrix, $R = \mathrm{diag}\{25, 25, \cdots, 25\}$. **Figure 4** shows the entire picture of the simulation scenario, including sensors positions and the actual and estimated trajectories. **Figure 5** illustrates the variation of received signal strengths, which are relayed to the activations of handoff events. Here these received signals are measured by RSSI sensors, where the position of sensors C^1, C^2, $\cdots$, C^9 are located at each cluster of sensors, respectively. Hence, with the variation of signal strength at each cluster, the handoff event will be triggered based on Equations (11) and (12).

According to the occurrences of handoff events, the switching among the clusters in the sensor network are shown in **Figure 6**. Noticed that the handoffs were triggering at those time periods around 50 and 130, switching back and forth due to the crossings of the cluster boundaries (#3, #4, and #9). **Figure 7** illustrates the state tracking of position, velocity, and acceleration. **Figure 8** depicts the RMSE values of estimations along *x*- and *y*-axis, including position, velocity, and acceleration.

Table 1. Simulation Parameters.

Parameters	Definition	Remark
$T_s = 1/5$	Sampling period	200ms
$T = 150$	# of iteration	Dynamic model
$\mu_j = 0$	Transmission power	RSSI
$\gamma = 2$	Path loss	RSSI
$\Delta = -40$	Threshold	Handoff scheme
$i = 20$	Number of sensors	At each cluster of the sensor network
$N = 50$	Particle numbers	Particle filter
$k = 2$	Number of states in Handoff	K-means clustering
$j = 9$	Number of clusters	Sensor network

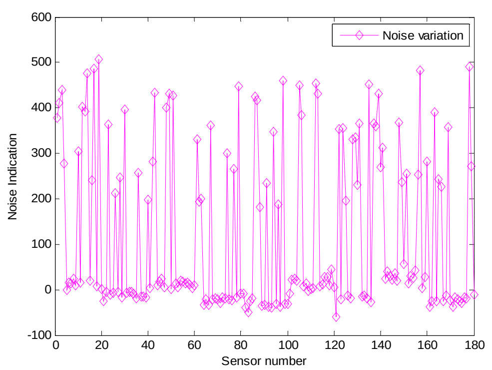


Figure 3. The variations of noise levels (m) with randomly toggling between LOS and NLOS conditions.

6.2. Performance Analysis

During the simulation, we found that using the tracking algorithm in combination with K-means clustering eventually reduces the estimation error of position, velocity, and acceleration substantially. The comparison of root-mean-square errors (RMSE) are shown in **Figure 9**.

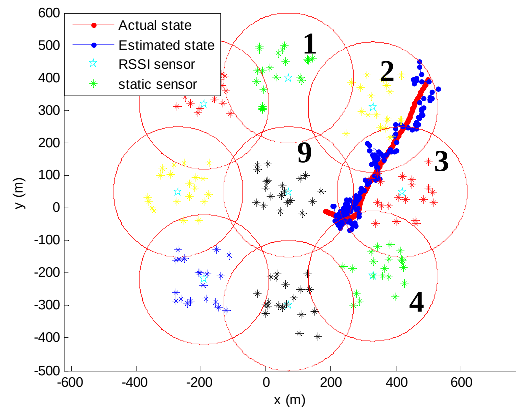


Figure 4. The actual and estimated trajectories of the moving target.

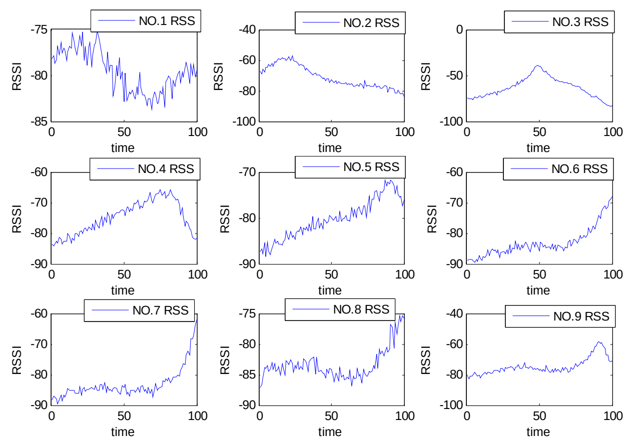


Figure 5. The signal strength at RSSI sensors from each cluster of network.

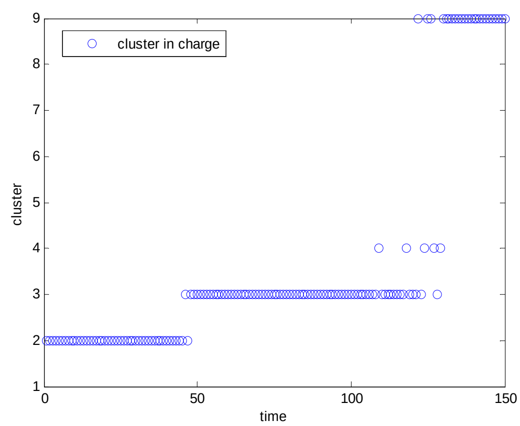


Figure 6. Switching history (the designated cluster vs. time).

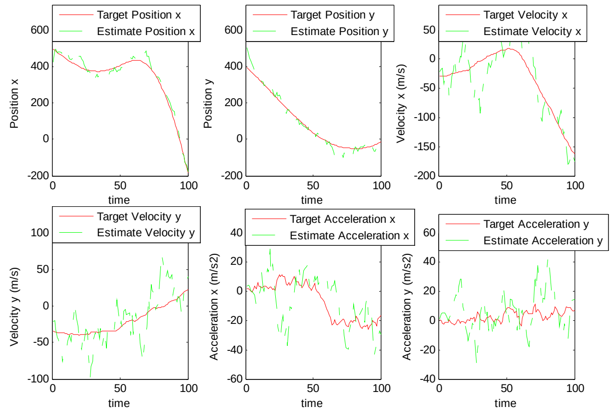


Figure 7. The estimations of acceleration, velocity, and position along x and y coordinates.

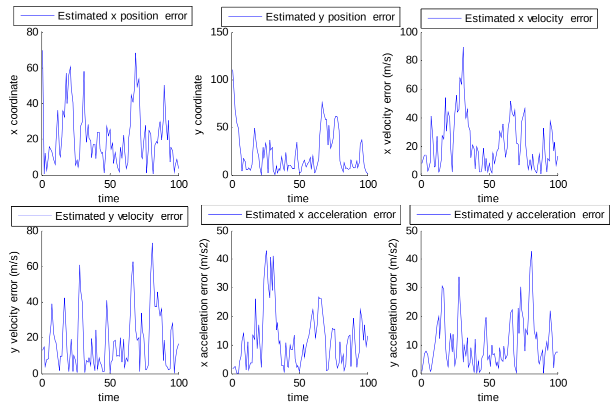


Figure 8. The RMSE values for the estimation errors of position, velocity, and acceleration.

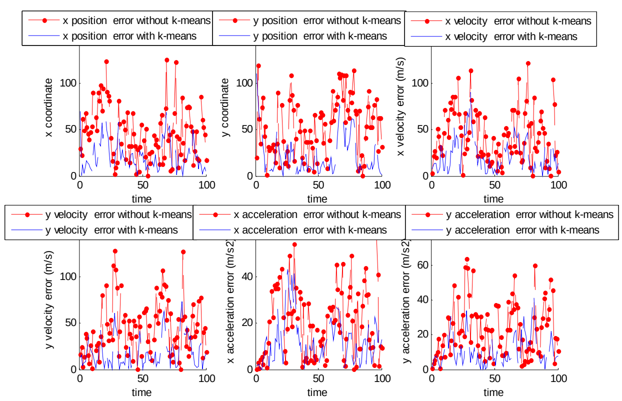


Figure 9. Comparing the RMSE values for the estimation errors of position, velocity, and acceleration with and without the augmentation of K-means clustering algorithm.

In **Figure 9**, the performance of tracking algorithm with K-means clustering is outperforming the one without clustering. The comparisons of error distributions due to the scenarios with and without the introduction of K-means clustering algorithm are plotted in **Figure 10**. The average estimation errors after conducting 20 Monte Carlo runs are denoted by Error1, solely with PF processing, whereas Error2 is incorporating with K-means clustering method.

The probability that Error1 is strictly greater than Error2 is 75% in average, while 25% is the case, Error1 < Error2. Similarly, assuming the possibility that Error1 will be strictly less than two times of Error2, the results are shown in **Figure 11** in the estimations of position, velocity, and acceleration. The robustness of the proposed PF and K-means clustering technique to mitigate the NLOS effect is verified as shown in both **Figures 10** and **11**.

As proposed above, the tracking algorithm is established by particle filter, and particle numbers are set by user. Different particle numbers may affect the filtering accuracy and processing speed for target tracking. Hence, we choose five sets of particle numbers to investigate the performance of tracking accuracy; there are 10, 50, 100, 500, and 1000 particles used for performance analysis. Environments for simulation are surrounded with either LOS or NLOS propagation.

We simulate the tracking estimation in a surrounded LOS propagation environment, and plot the means of estimation errors versus different numbers of particles, as shown in **Figure 12**. The results, with a surrounded NLOS propagation environment, are shown in **Figure 13**. In addition, **Figure 14** illustrates the performance with the employment of K-means clustering.

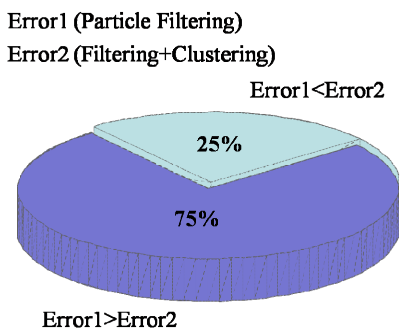


Figure 10. The probability of error distributions.

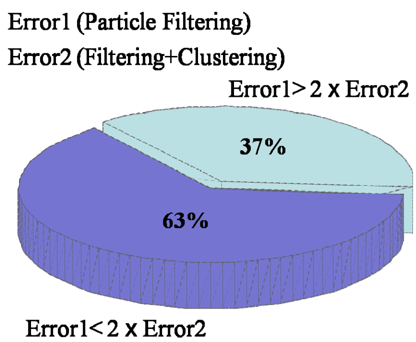
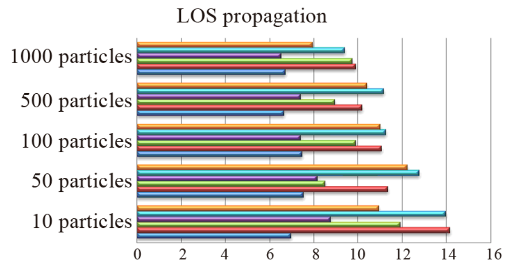
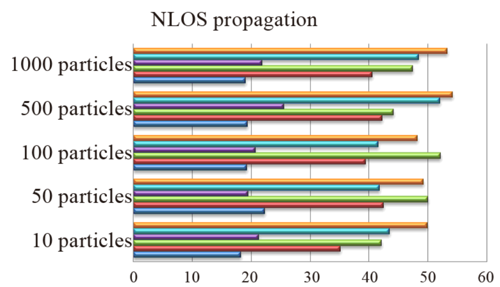


Figure 11. The probability of error distributions with the assumption Error1 is less than two times of Error2.



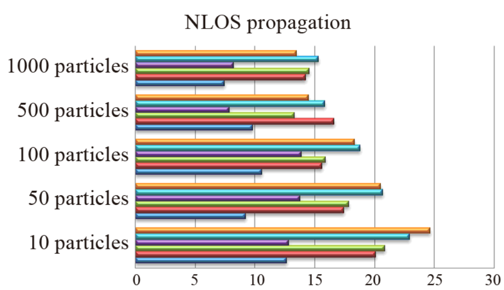
	10 particles	50 particles	100 particles	500 particles	1000 particles
x-axis position	10.9439	12.2317	11.0108	10.4087	7.9404
x-axis velocity	13.9616	12.7805	11.2445	11.1779	9.4037
x-axis acceleration	8.7646	8.1579	7.4238	7.4311	6.5316
y-axis position	11.9274	8.5246	9.9069	8.9654	9.7459
y-axis velocity	14.1546	11.3401	11.0665	10.1731	9.9087
y-axis acceleration	6.9858	7.5477	7.4651	6.6503	6.711

Figure 12. The results, estimation errors versus numbers of particles, using solely particle filtering in a surrounded LOS propagation environment.



	10 particles	50 particles	100 particles	500 particles	1000 particles
x-axis position	49.9661	49.2846	48.1913	54.1573	53.3286
x-axis velocity	43.4733	41.8349	41.5752	52.0958	48.4216
x-axis acceleration	21.3256	19.5778	20.7743	25.6271	21.9349
y-axis position	42.1214	50.0817	52.1323	44.2144	47.4691
y-axis velocity	35.2211	42.4538	39.4701	42.2123	40.5311
y-axis acceleration	18.3326	22.2981	19.3266	19.4063	19.1158

Figure 13. The results, estimation errors versus numbers of particles, using solely particle filtering in a surrounded NLOS propagation environment.



	10 particles	50 particles	100 particles	500 particles	1000 particles
x-axis position	24.6508	20.5456	18.3168	14.4855	13.4896
x-axis velocity	22.9567	20.6677	18.8074	15.8587	15.3321
x-axis acceleration	12.8406	13.8118	13.8985	7.8624	8.2434
y-axis position	20.8904	17.8349	15.9136	13.3067	14.5598
y-axis velocity	20 1223	17.4479	15.5954	16.6464	14.2717
y-axis acceleration	12.6818	9.2299	10.5755	9.823	7.4452

Figure 14. The results, estimation errors versus numbers of particles, using particle filtering with K-means clustering in a surrounded NLOS propagation environment.

As the simulation results shown from **Figure 12** to **Figure 14**, each figure is simulated with 1000 time instants, and we compute its associated estimation errors, position, velocity, and acceleration.

As position, velocity, and acceleration are estimated, not too much benefit can we expect in a LOS propagation environment when we vary the number of particles in PF processing. The number of particles we choose is 50, being attractive in real-time tracking applications. The estimation errors gradually reduce with the increaseing of particle numbers. On the contrary, situated in a NLOS environment, substantial improvement of estimation errors are illustrated in **Figure 14** with K-means clustering scenario; eventually, the trade-off among the increment of particle numbers, estimation errors, and computational load is accomplished via the use of moderate number of particle, *i.e.*, 50, and the augmentation of K-means clustering scheme in the particle filter processing job.

7. Conclusions

Sophisticated and high system/computational complexity algorithms are always proposed to mitigate the NLOS effect and estimate the mobile/target location. In this article, we propose a simple and feasible generic tracking algorithm to track the moving target in clusters of sensor network. The proposed tracking algorithm is the technique that adds handoff decision to the ordinary tracking algorithm, based on TOA and RSSI measurements; the handoff decision is implemented on clusters of sensor network.

Besides, K-means clustering is utilized, and it combines with particle filter to reduce the NLOS propagation effect. Finally, the proposed algorithm can accomplish higher accuracy in tracking estimation for sure.

Simulations illustrate that the estimation results of tracking trajectory is well predicted, even around the NLOS propagation environment. This analysis applies to any motion modes, even with varying acceleration. Moreover, we also compare the results of tracking algorithm with and without K-means clustering in statistics. Through the performance analysis, it demonstrates that the proposed tracking algorithm may find potentials in real-time tracking/localization applications as the particle numbers used are reducing to as low as 50.

8. Acknowledgements

This work was partially supported by the National Science Council of Taiwan, Republic of China under contract NSC 99-2221-E-151-022.

REFERENCES

[1] P. M. Djuric, M. Vemula and M. F. Bugallo, "Tracking

with Particle Filtering in Tertiary Wireless Sensor Networks," *IEEE International Conference on Acoustics, Speech and Signal Processing*, 18-23 March 2005, pp. 757-760. doi:10.1109/ICASSP.2005.1416119

[2] S. Maheswararajah and S. Halgamuge, "Mobile Sensor Management for Target Tracking," *2nd International Symposium on Wireless Pervasive Computing*, San Juan, 5-7 February 2007, pp. 506-510. doi:10.1109/ISWPC.2007. 342656

[3] J. Wu, H. Xiong, J. Chen and W. Zhou, "A Generalization of Proximity Functions for K-means," *7th IEEE International Conference on Data Mining*, Omaha, 28-31 October 2007, pp. 361-370.doi:10.1109/ICDM.2007.59

[4] J. F. Liao and B. S. Chen, "Robust Mobile Location Estimator with NLOS Mitigation Using Interacting Multiple Model Algorithm," *IEEE Transactions on Wireless Communications*, Vol. 5, No. 11, 2006, pp. 3002-3006. doi:10.1109/TWC.2006.04747

[5] S. S. Moghaddam, V. T. Vakilim and A. Falahati, "New Handoff Initiation Algorithm (Optimum Combination of Hysteresis & Threshold Based Methods)," *52nd IEEE VTS-Fall VTC of Vehicular Technology Conference*, Vol. 4, 2000, pp. 1567-1574. doi:10.1109/AMS.2009.11

[6] C.-D. Wann, "Kalman Filtering for NLOS Mitigation and Target Tracking in Indoor Wireless Environment," In: V. Kordic, Ed., *Kalman Filter*, Intech Publication, 2010, pp. 16-33.

[7] L. Yi, S. G. Razul, Z. Lin and C.-M. See, "Target Tracking in Mixed LOS/NLOS Environments Based on Individual TOA Measurement Detection," *IEEE Sensor Array and Multichannel Signal Processing Workshop*, Jerusalem, 4-7 October 2010, pp.153-156. doi:10.1109/SAM.2010.5606720

[8] W. Kei and L. Wu, "Mobile Location with NLOS Identification and Mitigation Based on Modified Kalman Filtering," *Sensors*, Vol. 11, No. 2, 2011, pp. 1641-1656. doi:10.3390/s110201641

[9] S. Venkatesh and R. M. Buehrer, "NLOS Mitigation Using Linear Programming in Ultrawideband Location-Aware Networks," *IEEE Transactions on Vehicular Technology*, Vol. 56, No. 5, 2007, pp. 3182-3198. doi:10.1109/TVT.2007.900397

[10] J. F. Liao and B. S. Chen, "Adaptive Mobile Location Estimator with NLOS Mitigation Using Fuzzy Inference Scheme," *8th International Symposium on Communications*, Kaohsiung, Taiwan, 2005.

[11] B. S. Gukhool and S. Cherkaoui, "Handoff in IEEE 802.11p-Based Vehicular Networks," *IFIP International Conference on Wireless and Optical Communications Networks*, Cairo, 28-30 April 2009, pp. 1-5.

[12] J. F. Liao and B. S. Chen, "Robust Mobile Location Estimator with NLOS Mitigation Using Interacting Multipke Model Algorithm," *IEEE Transactions on Wireless Communications*, Vol. 5, No. 11, 2006, pp. 3002-3006. doi:10.1109/TWC.2006.04747

[13] C. C. Tseng, K. H. Chi, M. D. Hsieh and H. H. Chang, "Location-Based Fast Handoff for 802.11 Networks," *IEEE Communications Letters*, Vol. 9, No. 4, 2005, pp. 304-306. doi:10.1109/LCOMM.2005.04010

[14] S. Arulampalam, S. Maskell, N. Gordon and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, 2002, pp. 174-188. doi:10.1109/78.978374

[15] Z. Yang and X. Wang, "Joint Mobility Tracking and Handoff in Cellular Networks via Sequential Monte Carlo Filtering," *IEEE Transactions on Signal Processing*, Vol. 51, No. 1, 2003, pp. 269-281. doi:10.1109/TSP.2002.806580

[16] R. Prakash and V. Veeravalli, "Adaptive Hard Handoff Algorithm," *IEEE Journal on Selected Areas in Communications*, Vol. 18, 2000, pp. 2456-2464. doi:10.1109/49. 895049

[17] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," Technical Report, University of North Carolina, New Caledonia, 2002.

Nomenclature

$\boldsymbol{A}_i$, $i = 1, 2$	State transition matrix
b_k, $k = 1, 2$	Binary sequence (LOS or NLOS)
E_k^i, $i = 0, 1$	Handoff/non-Handoff event
NLOS_k	Measurement error at time instant k
$p(\bullet\|\bullet)$	Conditional probability distribution
$q(\bullet)$	Importance density
$\boldsymbol{X}_{i,k}$, $i = 1, 2$	Target state vector at time instant k
W_k^i	Weighting associated with ith particle at time instant k