A Study on Particle Filter based on KLD-Resampling for Wireless Patient Tracking

Nga Ly-Tu*
School of Computer Science and Engineering, International University VNUHCM, Vietnam

Thuong Le-Tien, Linh Mai
Department of Electrical & Electronics Engineering, University of Technology-VNUHCM, Vietnam

(Received: April 29, 2016 / Revised: October 10, 2016 / Accepted: December 6, 2016)

ABSTRACT
In this paper, we consider a typical health care system via the help of Wireless Sensor Network (WSN) for wireless patient tracking. The wireless patient tracking module of this system performs localization out of samples of Received Signal Strength (RSS) variations and tracking through a Particle Filter (PF) for WSN assisted by multiple transmit-power information. We propose a modified PF, Kullback-Leibler Distance (KLD)-resampling PF, to ameliorate the effect of RSS variations by generating a sample set near the high-likelihood region for improving the wireless patient tracking. The key idea of this method is to approximate a discrete distribution with an upper bound error on the KLD for reducing both location error and the number of particles used. To determine this bound error, an optimal algorithm is proposed based on the maximum gap error between the proposal and Sampling Important Resampling (SIR) algorithms. By setting up these values, a number of simulations using the health care system’s data sets which contains the real RSSI measurements to evaluate the location error in term of various power levels and density nodes for all methods. Finally, we point out the effect of different power levels vs. different density nodes for the wireless patient tracking.

Keywords: SIR, KLD-Resampling, RSS, Health Care Systems

* Corresponding Author, E-mail: ltnga@hcmiu.edu.vn

1. INTRODUCTION
The current challenge in health care system is to estimate physical parameters associated to the patient (Mann et al., 2004) because of increasingly aging population combined with nursing staff shortages and decreasingly the capacity of hospital (Ko et al., 2010). Wireless patient tracking/monitoring provides flexible and powerful patient surveillance through wearable devices anytime and anywhere by Ren et al. (2010).

The core of health care system is an automatic supervision of patients within the nursing institute based on the lightweight system via Wireless Sensor Network (WSN). Patient’s supervision includes two major services: tracking the location of patients and monitoring the patient’s status. In the first service, patients need urgent staff intervention, the exact knowledge of their location plays an important factor to reach them in a short period of time. On the other hand, the last service, the status of patients in critical situation must be continuously available to the medical staff, when patients can roam around the hosting institution’s premises. Many kinds of information on the patient’s status are continually collected and automatically detected such as moving characteristics, heartbeat, etc.

Several researchers such as Redondi et al. (2010), Lim et al. (2010) and Evennou et al. (2006) introduced...
the LAURA (LocAlization and Ubiquitous monitoRing of paAtients) system, with Personal Monitoring System (PMS) and Personal Localization and Tracking System (PLTS) to monitor/track the wireless patient localization based on Received Signal Strength (RSS). In case of moving targets, Lim et al. (2010) introduced gradient descent to determine the wireless patient position when the distances from the patient to the anchor nodes computed via Signal-to-Distance Mapping (SDM).

Most research in WSN had emphasized the use of PFs for solving many problems. Nguyen et al. (2013) derived a new method for evaluation of the number of particles in WSN. Li et al. (2013 and 2015) introduce KLD-resampling for solving problems in WSN under the benchmark model of maneuvering in a two-dimensional plane. Ly-Tu et al. (2014) introduced the ability of KLD-resampling PF to evaluate the number of particles used for non-linear problems.

Recently, by incorporating these PF methods and gradient descent to avoid wall crossing under known environment’s map for improving wireless patient tracking via the help of WSN. Redondi et al. (2010) and Evennou et al. (2006) introduced PF based on Sampling Importance Resampling (SIR) to enhance the quality of tracking. While Ly-Tu et al. (2015) introduced the KLD-resampling PF for wireless patient tracking by the given upper bound error with fixed probability so as to create a sample set near the high-likelihood region. By setting up a specific upper bound error for the resampling parameter, we found that this technique further reduces the location error of patient but maintains the proper KLD-resampling (Li et al., 2015) for health care systems because the variation of RSS measurement values is the diminished. The upper bound error of the resampling parameter (Ly-Tu et al., 2015) is only a special case of Remark in Section 3.

The main contributions of this paper are summarized as follows. First, we extend the underlying system (Ly-Tu et al., 2015) by proposing an algorithm to obtain the upper bound error of resampling parameter based on the maximum gap error between the proposed algorithm and SIR method for various power levels and density nodes. Second, given the strong contribution of the selected upper bound values of resampling parameter and weight function method, we use them in the proposal for assessing the impact of power levels of self RSSI vs. different density nodes to track wireless patient location.

The paper is organized as follows. Introduction to a problem solving framework such as the patient localization based on gradient descent method, the improved localization via PF, and error criteria are given in Section 2. All related schemes, namely, SIR and our method are presented in Section 3. All experimental results based on MATLAB for wireless patient tracking are shown in Section 4. Finally, we conclude the paper in Section 5.

2. BACKGROUND

As has been said, we focus on the wireless patient tracking in health care system. Lim et al. (2010) discussed about the problem context and its location data requirements are given in the first section. The wireless patient localization by gradient descent method is shown in the next section. Final, the last section describes PFs serving to enhance the wireless patient tracking by Evennou et al. (2006) and Redondi et al. (2010).

2.1 Problem

Evennou et al. (2006) introduced a typical health care system, as shown in Figure 1, to assemble the required data from the patients and distribute the same data via a central controller. This system contains the personal monitoring system (PMS), the personal localization and tracking system and a lightweight and flexible network architecture (NA). The PMS is composed of small-wearable devices geared with sensing and wireless transmission capabilities to monitor patient’s status and transmit the comprehensible information to the central controller via the help of the NA.

The accuracy of localization calculates on the site of the anchor nodes and on their density, for instance, an average localization error of just a few meters is evaluated via real experiments.

Regardless of the fundamental network infrastructure, localization is performed adopting analogous processing methods. Under fixed and known the placement of the anchor nodes, RSS measurements are gathered between pairs of nodes which related to the inter-node distances. Indeed, a RSS indicator (RSSI), which addressed by non-parametric or parametric methods, is supplied by off-the-shelf devices without the demand of ded-
located hardware.

The first method, namely non-parametric method or fingerprinting which is introduced by Lim et al. (2010) and Evennou et al. (2006), is described to avoid the explicit modeling of the RSSI-distance relationship. The collection of fingerprint databases, which is implemented during a training phase, has many fingerprint vectors. Each vector consists of the RSSI values measured between a client node at a specific site and the anchor node. A client node forms a vector of RSSI measurements between itself and each anchor node and it compares this vector with the pre-computed fingerprints in the database. Final, its position is estimated based on the placement of the most similar fingerprints. This method suffers from a non-negligible setup cost which is related to the construction of the fingerprint database. Furthermore, it is not robust in face of changes of the environment due to invalid the affection of RF propagation and pre-computed fingerprints.

The second method, which is called parametric method, explicitly assumes the propagation model’s knowledge that involves the RSSI metrics $S$ (expressed in dBm units) and the inter-node distance $d$. Patwari et al. (2005) introduced an effortless dependency as the following model

$$S = S_0 - 10\alpha \log_{10} \frac{d}{d_0} + \nu,$$  \hspace{1cm} (1)

where $S_0$ is the RSSI metrics measured between two nodes, $d_0$ meters apart, and $\alpha$ indicates the factor in path loss for indoor environments. The effects of shadowing and multipath environments is represented by the noise term $\nu$ with a Gaussian random variable $N(0, \sigma^2)$, whereas the value of standard deviation $\sigma_v$ depends on the characteristics of the specific environment. Barsocchi et al. (2009) introduced this method to solve the localization problem by computing the position that maximizes the likelihood with respect to the model in (1). The extension of this model is to include the signal attenuation caused by walls. The advantage of this approach is to restrict the system-setup cost by estimating based on RSSI measurements between anchor nodes. Meanwhile, Lim et al. (2010) also introduced a similar approach via applied the propagation model in (1) under implicit assumptions, but the distance between a client and anchor node is modeled as a linear combination of the RSSI measurements between the client and all the nodes. The construction of these RSSI measurements based on this way is called the RSSI-to-distance linear mapping. The disadvantage of this approach is to setup a zero-configura-

Because the noise influences RSSI measurements, temporal averaging is used over a time window. The length of the window enables to define a trade-off between the RSSI measurement’s accuracy and the system delay. As a rule of thumb, every 200ms each node is collected and 3-5 RSSI measurements are updated.

In case of the moving patient, the position of the wireless patient is tracked over time so as to improve the quality of localization. To this end, Evennou et al. (2006) applied a simply PF method which enables to merge the RSSI measurements together with the typical movement patterns of humans in indoor environments. This way can avoid wall crossing because of incorporating this a-priori information and PF method.

### 2.2 Indoor Localization Via Gradient Descent

Let us denote $\chi_i \in \mathbb{R}$, $i=1, ..., M$ be the position of an anchor node, while $M$ indicates M anchor nodes. Each anchor node obtains a vector $s_i = [s_{i1}, s_{i2}, ..., s_{iM}]^T$ of RSSI measurements exchanging data with the other nodes, where $s_{ij}$ is the RSSI relative to the signal emitted by the $j$-th anchor node. Then, let us define $S = [s_1, s_2, ..., s_M] \in \mathbb{R}^{M \times M}$ be the overall RSSI measurements among all anchor node pairs. In general, $S$ is not symmetric in radio links; diagonal entries of $S$ contain the self-RSSI values.

Let us define $d_i = [d_{i1}, d_{i2}, ..., d_{iM}]^T$ with $d_{ij}$ is the Euclidean distance between anchors $i$ and $j$, in which case of the distance vector, and the corresponding matrix $D = [d_1, d_2, ..., d_M] \in \mathbb{R}^{M \times M}$ is symmetric and has zero diagonal entries. Note that these are the solely parameters that require to be manually obtained during the system calibration phase, as they depend on the specific hardware. A linear relationship between the RSSI measurements and the logarithm of the inter-node distance by Lim et al. (2010) is followed

$$\log(D) = TS$$ \hspace{1cm} (2)

where $T \in \mathbb{R}^{M \times M}$ is the signal-to-distance mapping and each row vector is represented as a linear combination of the columns of $S$, weighted by the elements of the $i$-th row $T_i$'. The matrix $T$ is calculated by least square method as

$$T = \log(D)S( SS^T)\hspace{0.5cm}^{-1} .$$ \hspace{1cm} (3)

Once the signal-to-distance mapping is determined, the client node’s localization proceeds collecting RSSI measurements between itself and its adjacency anchor nodes in the vector $\hat{s}$. Then, the corresponding distance vector $\hat{d}$ can be computed as
\[
d = \exp(Ts). \quad (4)
\]

The location based on the obtained distance vector \( \hat{d} \) is estimated owing to apply a gradient descent algorithm by Lim et al. (2010). It is exploited in LAURA system to initialize tracking.

2.3 Improving Localization Via Particle Filtering

The paper focuses on improving the wireless patient tracking in LAURA PLT (Personal Localization and Tracking) module by Redondi et al. (2010 and 2013) to smooth down the functions of the RSSI variations due to PF.

In case of static conditions and fixed client nodes, the localization algorithm described above achieves slightly perfect when robust RSSI measurements can be obtained by means of the time averaging. On the other hand, when the node is attached to the patient’s body, in which case of moving client nodes, severe short-time RSSI fluctuations are observed. As such, the accuracy of the estimated location decreases, and artificial motion discontinuities are detected.

The localization accuracy is improved with a help of incorporating a-priori knowledge about the moving node and the geometry of the environment. In this system, a PF to track the position of the client node over time is used. At each time instant \( t \), the state of the node is represented by the vector \( x(t) = [x(t), v(t)]' \), where \( x(t) \in \mathbb{R}^2 \) indicates the position of the node and \( v(t) \in \mathbb{R}^2 \) its velocity.

The sequence of previous states and all available observations are given. It is defined \( z_i(t) = [x_i(t), v_i(t)]' \), PF estimates the posteriori probability density function of the state \( z \) at time \( t \) with \( k = 1, ..., N \) and \( p_k(t) \) is a weighted set of particles. The state of the target node is then point-wise estimated as:

\[
\begin{align*}
\hat{x}(t) &= (1-\alpha_x)(\hat{x}(t-1)+\hat{v}(t-1)\Delta_t)+\alpha_x \sum w_i(t)\hat{x}(t), \\
\hat{v}(t) &= (1-\alpha_v)(\hat{v}(t-1)+\alpha_v(\hat{x}(t)-\hat{x}(t-1))). \quad (5)
\end{align*}
\]

where \( \Delta_t \) is the sampling period, and \( \alpha_x, \alpha_v \) are the adaption rates which are tuned experimentally based on the node dynamics.

For the problem at hand, PF alternates the execution of two steps: prediction and update. Firstly, these assumptions of system will be discussed. Then, prediction step will be shown in the next part. Finally, how to update process will be mentioned in the last.

2.3.1 Assumptions

The location of a target node \( \hat{x}(0) \) is the first thing at startup which can be computed by using the estimated vector \( \hat{d} \) through a gradient descent method which minimizes the following cost function:

\[
\hat{x}(0) = \arg \min_x \frac{1}{2} \sum_{i=1}^N \theta_i \left( \| x - x_i \| - \hat{d} \right)^2. \quad (6)
\]

where \( \hat{d}_i \) is the estimated distance between the target node and the \( i \)-th anchor and \( \theta_i \) is a weight factor computed as

\[
\theta_i = \frac{\hat{d}_i^{-2}}{\sum_{i=1}^M \hat{d}_i^{-2}}. \quad (7)
\]

By differentiating equation (6) with respect to \( x \) it is possible to obtain an iterative equation used to update the estimated \( \hat{x} \):

\[
\hat{x}^{(k+1)} = \hat{x}^{(k)} + \alpha \sum_{i=1}^N \theta_i \left( 1 - \frac{\hat{d}_i}{\| \hat{x}^{(k)} - x_i \|} \left( \hat{x}^{(k)} - x_i \right) \right). \quad (8)
\]

with \( \alpha \) is set to 0.1; the initial estimate \( \hat{x}^{(0)} \) is equal to the location of the nearest anchor node, and then \( z_i(t) = [\hat{x}, \hat{v}]' \), \( \forall p \).

Secondly, since the initialization of the particle filter might potentially locate the target outside the building, the geometry leverage's knowledge of the environment to constrain the initialization point to lie inside the building.

2.3.2 Prediction

At each step, the new state vector for the \( p \)-th particle is computed. The following dynamic model constructed upon kinematic equations is employed.

\[
\begin{align*}
\hat{x}_p(t) &= \hat{x}_p(t-1) + v_p(t-1) \Delta_x + \xi_x, \\
\hat{v}_p(t) &= \hat{v}_p(t-1) + \xi_v
\end{align*} \quad (9)
\]

with \( \xi_x \) and \( \xi_v \) are samples drawn from zero-mean Gaussian random variables with variance, respectively, \( \sigma_x^2 \) and \( \sigma_v^2 \), which provide the system with a diversity of hypotheses. If the geometry of the environment is known, it can be incorporated in the dynamic model, e.g. by assigning \( w_p = 0 \) to those particles that crossed a wall.

2.3.3 Update

After the prediction step, the weight of the \( p \)-th particle is updated based on the RSSI measurements collected at time \( t \). Firstly, the vector \( \hat{d} \) as described in (4) is obtained. Then, we assign a weight computed as

\[
w_p(t) = w_p(t-1) \exp \left( \frac{-1}{2} \sum_{i=1}^M w_i \left( \| \hat{x}_p(t) - x_i \| - \hat{d}_i \right)^2 \right). \quad (10)
\]

where \( \hat{d}_i \) is the estimated distance, and \( \| \hat{x}_p(t) - x_i \| \) is the current distance of the particle from the \( i \)-th anchor.
In order to avoid the critical situation, the weight distribution of the particles in (10) is tested. When all particles are trapped within a room while the client node had moved outside,
\[ \sum_{p=1}^{N} w_p(t) > \tau, \]
(11)
where \( \tau \) is a threshold value that we set equal to \( 10^{-5} \) in our experiments. If the test fails, particles will not correct tracking the client node. Thus, the PF is re-initialized as described below. Otherwise, the particle weights are normalized such that \( \sum_{p=1}^{N} w_p(t) = 1 \) and the client node’s position is determined based on (5).

Working with PFs as customary, a resampling step will perform if the weight distribution is severely skewed, such that has just a few particles with non-negligible weight. In this paper, two kinds of resampling steps, namely SIR and KLD-resampling, will be discussed in the next step.

2.4 Error Criteria

Let us define \( Er \) to be the error for measuring the performance of the residuals which are considered the difference between the observed and predicted values.
\[ Er = \sum_{i=1}^{K} (\hat{x}_i - x_i) \]
(12)
where \( \hat{x}_i \) and \( x_i \) are respectively the predicted location and the observed location at the \( i \)-th observation, and \( K \) is the number of observations.

3. SIR AND PROPOSAL ALGORITHMS

In this section, SIR PF is reviewed and our proposal (KLD-resampling PF) is presented.

3.1 SIR Algorithm

The concept of the auxiliary PF has been introduced by a number of researchers such as Redondi et al. (2010 and 2013) and Arulampalam et al. (2002). A PF is also known as the bootstrap filter, Monte Carlo technique, condensation algorithm, interacting particle approximations and survival of the fittest. The key idea is to represent the required posterior density function by a set of random samples (particles) with associated primary weights, and to compute the estimates based on the samples and primary weights. As the number of samples becomes very large, the Monte Carlo characterization is the closest equivalent for representation of the posterior probability function, and the solution approaches the optimal Bayesian. Schön et al. (2010) introduced the sequential importance sampling (SIS) algorithm, shown in Algorithm 1 for the PF, includes a resampling step at each instant as described in detail in the reference. The SIS algorithm uses the important density, which is a proposed density to represent another one that cannot be exactly computed, that is the sought posterior density in the present case. Hence, samples are drawn from the important density instead of the actual density. The degeneracy phenomenon is known as a common problem with the SIS PF, which all but one particle has negligible primary probability function, and the solution approaches the optimal Bayesian.

Algorithm 1 Pseudo-code of SIS

1: procedure SIS \( \{x_p^i, w_p^i\}_{i=1}^{N} \) 
2: \( c = 0 \) \( \triangleright \) Initialise the first CDF 
3: for \( i = 2: N \) do 
4: \( c_i = c_{i-1} + w_p \) \( \triangleright \) Construct others 
5: end for 
6: \( u_i \sim U[0, N-1] \) \( \triangleright \) Draw a starting point 
7: for \( j = 1: N \) do 
8: \( u_j = u_i + N^{-1}(j-1) \) \( \triangleright \) Move along the CDF 
9: end for 
10: \( t = \text{SUM}\left\{w_p^i, 1_{i=1}^{N}\right\} \) \( \triangleright \) Calculate total weight 
11: for \( i = 1: N \) do 
12: \( w_p^i = t^{-1}w_p^i \) 
13: while \( (u_j > c_i) \) do 
14: \( i' = i \) \( \triangleright \) Increase \( i \) 
15: end while 
16: \( x_p^i = x_p \) \( \triangleright \) Assign sample 
17: \( w_p^i = N^{-1} \) \( \triangleright \) Assign weight 
18: \( i' = i \) \( \triangleright \) Assign parent 
19: end for 
20: end procedure

Algorithm 2 Pseudo-code of SIR PF

1: procedure SIR \( \{x_p^i, w_p^i\}_{i=1}^{N} \) 
2: for \( i = 1: N \) do 
3: \( x_p^i = p(x_{p_{i-1}}) \) \( \triangleright \) Draw sample 
4: \( w_p^i = p(x_{p_{i-1}}) \) \( \triangleright \) Calculate weight 
5: end for 
6: \( t = \text{SUM}\left\{w_p^i, 1_{i=1}^{N}\right\} \) \( \triangleright \) Calculate total weight 
7: for \( i = 1: N \) do 
8: \( w_p^i = t^{-1}w_p^i \) \( \triangleright \) Normalise weight 
9: end for 
10: \( \triangleright \) Applied Alg.1 
11: end procedure
weight after a few states. Thus, a large computational effort is devoted to updating particles whose contribution to the approximation of the posterior density function is almost zero. This problem can be overcome by increasing the number of particles or more efficiently by approximately selecting the important density. In addition, the use of resampling technique is recommended to avoid the degeneracy of the particles by Arulampalam et al. (2002), as Algorithm 1. The pseudo-code of SIR PF is shown in Algorithm 2.

3.2 Proposal Algorithm

In this subsection, we discuss two problems. First, we describe our KLD-resampling for applied PF. Second, we present a way to determine the optimal bound values of resampling parameter for our method, as shown in Algorithm 4. Because our proposal which relates to KLD-sampling, thus we briefly overview this method.

Fox et al. (2003) introduced KLD-sampling to determine the number of samples such that, with probability $1-\delta$, the error between the true posterior and the sample-based approximation is less than $\varepsilon$. KLD is used to show how to determine the number of samples so that distance between the sample-based maximum likelihood estimate and the true posterior does not exceed a pre-specified threshold.

The KLD between the proposal ($q$) and ($p$) distributions is defined in discrete form as

$$d_{KL}(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

with $W(x) = p(x)/q(x)$. The required number $N_{r}$ of samples is determined as follows

$$N_{r} = \frac{1}{2\varepsilon} \chi_{\varepsilon \cdot 1-I-\delta}^2,$$  \hspace{1cm} (14)

where $l$ is the number of bins with support, the quantizes of Chi-square distribution is computed as

$$P\left(\chi_{\varepsilon \cdot 1-I-\delta}^2 \leq \chi_{\varepsilon \cdot 1-I-\delta}^2\right) = 1 - \delta$$ \hspace{1cm} (15)

Based on Wilson-Hilferty transformation by Johnson and Kotz (1994) to compute the approximation of $\chi_{\varepsilon \cdot 1-I-\delta}^2$ in (14) is expressed as

$$N_{r} = \frac{l-1}{2\varepsilon} \left(1 - \frac{2}{9(l-1)} + \frac{2}{\sqrt{9(l-1)z_{\varepsilon \cdot 1-I-\delta}}}ight)^3,$$ \hspace{1cm} (16)

where $z_{\varepsilon \cdot 1-I-\delta}$ is the upper quartile of the standard normal distribution.

The undesirable results above are caused by the number of particles needed to approximate a discrete distribution with an upper bound $\varepsilon$ on the KLD. In other words, the KLD-sampling method leads to statistical bounds of the approximation quality of samples that are actually drawn from the proposal rather than the true posterior distributions. Therefore, the mismatch between the true and the proposal distribution is ignored.

3.2.1 KLD-resampling

To avoid the mismatch from KLD-sampling, the total number of required particles in (16) is applied in the resampling process better than sampling one. Li et al. (2013 and 2015) introduced this method which divides the particles of the posterior distribution into bins and count the number $l$ of bins which at least one particle is resampled to determine the total number of particles to resample. Therefore, the required number $N_{r}$ in (16) is rewritten as

$$N_{r_{min}} = \min \left\{ N_{min} \cdot \text{cell} \left(\frac{l-1}{2\varepsilon} \left(1 - \frac{2}{9(l-1)} + \frac{2}{\sqrt{9(l-1)z_{\varepsilon \cdot 1-I-\delta}}}ight)^3\right) \right\}.$$ \hspace{1cm} (17)

The Pseudo-code of proposal is shown in Algorithm 3.

3.2.2 Finding the upper bound error

We propose an algorithm to find the upper bound error as shown in Algorithm 4. Let us denote $\varepsilon_{i}$ to be the upper bound error of $i$ th, $E_{\varepsilon_{i}}^{p_{a}}$ and $E_{\varepsilon_{i}}^{m}$ which are respectively the error of proposal at $\varepsilon_{i}$ and that of SIR; and $\varepsilon_{i}$ (line 11) is the set of the upper bound value that fulfils the condition of Remark (line 10). Let us define $\Delta E_{i} = [\Delta E_{\varepsilon_{i}}^{p_{a}}, \ldots, \Delta E_{\varepsilon_{i}}^{m}]$ with $\Delta E_{\varepsilon_{i}}^{p_{a}} = E_{\varepsilon_{i}}^{m} - E_{\varepsilon_{i}}^{p_{a}}$ is the gap of error value between SIR and proposal at $\varepsilon_{i}$. Algorithm 4 provides the optimal bound value of proposal.
Remark: if $\Delta E > 0$ (lines 10-13), then a value $\varepsilon_{\text{opt}}$ (line 16) exists to maximize the function $\Delta E$.

4. SIMULATION RESULTS

We conducted a series of simulations via the data sets, which are used from publicly available by ANT Lab, to determine the optimal bound value of resampling parameter in (16) based on Algorithm 4. Next, the performance of tracking error, various power levels of self-RSSI by Hongliang et al. (2009) and the gap error for all methods are evaluated under four density nodes by Redondi et al. (2010 and 2013) and Ly-Tu et al. (2015). The network architecture described by Lim et al. (2010) which consists of MicaZ and IRIS motes operated by the TinyOS.

All simulation results are run on PC Core i5-2400 @3.10GHz, 4.00GB RAM and MATLAB 2012a (7.14.0.739). Furthermore, all parameters assignment for this system is given in Table 1.

The test was repeated for different density nodes and different power levels. To discard the bias related to a specific anchor node, the choosing estimation in random subset of anchor nodes was repeated for each density node.

The results in Table 2 show the optimal bound values with various power levels of self-RSSI in term of four density nodes for our proposal. For example, at power level of self-RSSI (-15dBm), the bound values for four various density nodes are respectively about 0.3, 0.95, 0.75 and 0.9 (see column 7 in Table 2). First, we set up these values for our proposed method in Figure 3 to compare the cumulative density function of the tracking error for Gradient descent, SIR with 200 particles, SIR with 400 particles and proposal with 200 particles is shown in Figure 2. Finally, all these values in Table 2 are applied to evaluate the performance of gap error vs. four various density nodes for all these methods and examine thoroughly the impact of density nodes on the whole power levels for our proposal in Figure 3 and Figure 4, respectively.

### Table 1. Parameters assignment of LAURA system

<table>
<thead>
<tr>
<th>Lines</th>
<th>Variable symbols</th>
<th>Meaning</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M</td>
<td>The number of anchor nodes</td>
<td>[5; 10; 15; 20]</td>
</tr>
<tr>
<td>2</td>
<td>exp_day</td>
<td>Data sets is retrieved (ANT Lab, 2015)</td>
<td>’9 aprile 2’</td>
</tr>
<tr>
<td>3</td>
<td>N_max</td>
<td>Maximum number of subsets to test for each density</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>$\alpha_s$</td>
<td>Adaption rate in (5)</td>
<td>0.95</td>
</tr>
<tr>
<td>5</td>
<td>$\alpha_s$</td>
<td>Adaption rate in (5)</td>
<td>0.8</td>
</tr>
<tr>
<td>6</td>
<td>$\tau_s$</td>
<td>Threshold value in (11)</td>
<td>$10^{-5}$</td>
</tr>
<tr>
<td>7</td>
<td>N_max</td>
<td>The maximum number of samples in Alg.3</td>
<td>200</td>
</tr>
<tr>
<td>8</td>
<td>$\varepsilon$</td>
<td>Upper bound error in Alg.4</td>
<td>See in Table 2</td>
</tr>
<tr>
<td>9</td>
<td>$\delta$</td>
<td>Fixed probability in Alg.3</td>
<td>0.01</td>
</tr>
</tbody>
</table>

### Table 2. Performance of finding the optimal bound algorithm

<table>
<thead>
<tr>
<th>Nodes</th>
<th>Power levels of self-RSSI [dBm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.65 0.3 0.85 0.5 0.4 0.3 0.85</td>
</tr>
<tr>
<td>10</td>
<td>0.8 0.7 0.8 0.85 0.85 0.95 0.9</td>
</tr>
<tr>
<td>15</td>
<td>0.15 0.85 0.9 0.95 0.3 0.75 0.75</td>
</tr>
<tr>
<td>20</td>
<td>0.6 0.9 0.55 0.1 0.35 0.9 0.95</td>
</tr>
</tbody>
</table>
The comparison of the cumulative density function of the tracking error for Gradient descent, SIR with 200 particles, SIR with 400 particles and proposal with 200 particles in case of power level (-15dBm) for four density nodes (5nodes, 10nodes, 15nodes and 20nodes) is illustrated in Figure 2. It is observed that the tracking error of each density node, at errors in 80% of the cases (Pr < 80%), for Gradient descent method achieves below 5m, around 4m and below 2.5m are respectively in Figure 2a, Figure 2b, and Figure 2cd. As enabling the PFs, the localization accuracy of our proposed is better than that of SIRs and Gradient descent methods. For example, at Pr = 0.8 in Figure 2c, it verifies that the curve of the localization accuracy for our proposal with 200 particles enhances respectively about 0.11 and around 0.37 that of SIR with 200 particles and SIR with 400 particles. It also verifies that our proposal not only reduces the number of used particles because of the selected upper bound error with fixed probability of resampling parameter, but also improves the average patient localization error compared with SIRs PF in clinical environments. Specially, as increasing the number of nodes our curve can improve the performance of the tracking error compared with others and satisfy the accuracy error from 3m to 5m by Farid et al. (2013).

We further study the relationship between the gap error and various power levels for all methods under different density nodes is shown in Figure 3. At 5 nodes as illustrated in Figure 3a, overall particles used of proposal is almost equal to the number of particles used of SIR, it confirms the statistical gaps error of the proposal and SIR are respectively about 0.8, 0.38, 0.22, 0.2, 0.04, 0.04.

Figure 2. Cumulative density function of the tracking error in case of power level (-15dBm)
(a) 5nodes (b) 10nodes (c) 15nodes (d) 20nodes.
0.26, and 0.36 when changing power levels of self-RSSI from -25dBm to 0dBm, respectively. In addition, as increasing the number of particles for SIR to twice, our method performs slightly about 0.03, 0.11, 0.04, 0.1, 0.06, 0.05, and 0.19 when compared this method. Similar to Figure 3a, when changing the density nodes from 0.02 nodes/m² to 0.08 nodes/m², the proposal also performs with traditional others.

Before reaching a conclusion, it is important to examine thoroughly the impact various power levels on the tracking error for our proposal from 0.02nodes/m² to 0.08nodes/m² as illustrated in Figure 4. First, Figure 4d shows the performance of tracking error, the all curves of our proposal converge and reach below 2m at Pr = 80%. On the other hand, with 20 nodes or 0.08nodes/m², the proposal can apply to any power levels or can use with a single transmit-power information. Moreover, the proposal in case of 15 nodes fulfils hard accuracy criteria only two power levels which are 0dBm and -3dBm. Figure 4a and Figure 4b illustrate to find power levels for improving tracking error from 3m to 5m in others density nodes.

Figure 3. Gap error vs. various power levels of self-RSSI for all methods (a) 5nodes (b) 10nodes (c) 15nodes (d) 20nodes.
5. CONCLUSIONS

In this study, a KLD-resampling under the optimal bound error of resampling parameter is introduced to smooth down the fluctuations of RSS variations. Ultimately, the wireless patient tracking is enhanced using WSN with a PF system structure that is designed by these parameters. Furthermore, in using these bound error values, this technique reduces the number of particles used and produces accurate patient location in real environment. We conduct a large set of simulations to study the tracking error, the relationship between the gap error and various power levels, and the effect of different density nodes under the whole power levels. For future work, we will assess PFs based on an efficient KLD-resampling algorithm on board. Future works, we will analyse the problem under probability to show the great contribution in the slope of the mean number of particles used.

ACKNOWLEDGMENT

This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number C2015-28-03. The authors would like to thank the International University of the VNU-HCM for its support and the anonymous reviewers for their valuable comments.
REFERENCES


Barsocchi, P., Lenzi, S., Chessa, S., and Giunta, G. (2009), Virtual calibration for rss-based indoor localization, with ieee 802.15.4, IEEE International Conference on Communications, 1-5.


Ren, Y., Pazzi, R., and Boukerche, A. (2010), Monitoring patients via a secure and mobile health care system, Wireless Communications, 17(1), 59-65.