AdaMap: Adaptive Radiomap for Indoor Localization

Zhiqi Yang, Yongcai Wang(✉), and Lei Song

Institute for Interdisciplinary Information Sciences (IIIS), Tsinghua University, Beijing 100084, People's Republic of China
wangyc@tsinghua.edu.cn

Abstract. In wireless networks, radiomap (also known as fingerprinting) based locating techniques are commonly used to cope the diverse fading signatures of radio signal, in which probabilistic or static radiomaps are trained in offline phase. A challenging problem of radiomap locating is that the radiomap can be outdated when environments change. Reconstruction of radiomap is time consuming and laborious. In this paper, we exploit the inter-beacon radio signal strength (RSS) to construct adaptive radiomap (AdaMap) by an online self-adjusted linear regression model. The distinct feature of AdaMap is that not only the radio signatures at the training locations vary with the online inter-beacon RSS measurements, but also the coefficients of the model are self-adjusted when the environments change significantly, so that AdaMap is highly adaptive to the environment changes. The proposed schemes are evaluated by extensive simulations, with comparisons to the state of art of the radiomap wireless localization methods. The results showed that AdaMap presented dramatical advantages in preserving positioning accuracy when the environments changed over time.

1 Introduction

Nowadays, considerable research efforts were dedicated to develop wireless networks as indoor location infrastructure. Access points (AP) or other wireless nodes which have been widely deployed in the indoor environments are treated as reference points of locating, called *beacons*. A target to be located measures the received signal strength (RSS) from the beacons to infer its own location. *Propagation model regarding distance* and the *signature of signal strength* are two ways of utilizing the RSS. The former way is to infer distance by RSS, which is generally inaccurate for the complex fading nature of RF signals in the physical environments; the latter one is to adopt a patten-matching based approach [1,2], which contains two phases. In the offline phase, the RSS signatures of a set of training positions are measured to construct a database, which is called radiomap. In the online phase, a target takes online RSS measurement to search in the radiomap to find a position that has the least RSS-distance (distance in RSS space) to the measurement, which will be treated as the position estimation of the target.

S. Papavassiliou and S. Ruehrup (Eds.): ADHOC-NOW 2015, LNCS 9143, pp. 134–147, 2015.
DOI:10.1007/978-3-319-19662-6_10

Radiomap based locating method can achieve meter level positioning accuracy when the radiomap is trained to be fine-grained (for example in 1 m resolution). But fine-grained radiomap positioning method is challenged by two key problems: (1) training the radiomap of fine-grained locations is laborious, especially when the area of interest (AOI) is large; (2) the laboriously trained radiomap is easily outdated when the environment changes. The variations of weather, temperature, movement of objects all can deviate the real-time radio signatures from the offline trained radiomap. But reconstruction the radiomap is time-consuming and laborious.

To overcome these challenges, designing adaptive radiomap mechanism to update the radiomap online is therefore critically important for the radiomap method to be valuable in practice. Existing works have investigated this problem mainly from two ways: (1) using additional reference points, and (2) designing adaptive radio-map model using inter-beacon RSS measurements.

(1) *Using additional reference points:* LANDMARC [3] used the active RFID and training data learned from online sources to perform indoor localization. The main advantage of LANDMARC is that it improves locating accuracy by utilizing the reference tags. LEMT [4] used additional reference points (with known positions) to online collect radio signatures and proposed model-tree method to construct piece-wise linear regression radio-map which can be online calibrated. Reference [5] used radio signatures learned from reference points to online calculate the channel models constants to adapt the radio-map model. Reference [6] proposed to use RFID sensors and environment sensors to help the location systems to adapt to the environment dynamics. These methods all need additional reference points or sensors to provide online assistance for radio-map adaptation.

(2) *Adaptive radio-map model by measuring inter-beacon RSS:* To relieve the needs for additional reference points or sensors, the closest related work in [7] proposed to use the beacons themselves as reference points, because most of current beacons (such as WiFi or sensor nodes) have both transmitting and receiving capabilities. Linear regression type radiomap was trained for each training location by using the inter-beacon RSS as a reference. We will introduce the details in Sect. 2. Another important approach is to use hybrid method to improve the reliability of radio-map based locating [25]. We focus on pure radio-map based solutions.

However, in existing adaptive radiomap model as stated in [7], the coefficients of linear regression model will remain constant after training without further adapting to the environments online. The principle of radiomap adaptivity is only to substitute the online measured inter-beacon RSSs into the static functions to infer the RSS signatures of specific locations. Since the function coefficients are non-adaptive, such model is partially adaptive, because when environments change dramatically, the linear regression model, i.e., the trained coefficients may become out-of-date, which fails to capture the changes of radio signatures. Therefore, how to design adaptive radio-map model to make not only the radio signatures at the training locations vary with the online inter-beacon

RSS measurements, but also to online adjust the coefficients of the trained functions autonomously is critically important.

To achieve this, this paper presents AdaMap, which uses the inter-beacon RSSs to online justify the disparity between the trained model and the online RSS measurements. It updates the coefficients of the model when noteworthy disparity between the trained model and the online measurements is detected. Algorithms are designed to check the disparity and to update the coefficients adaptively. We further investigated the distance and angle metrics for selecting reference beacons to make the online model updating more efficient and more accurate. We show by extensive simulations and hardware experiments that AdaMap provides adaptivity to the environment changes, which helps to provide dramatically better online positioning performances than the other radiomap locating methods.

The remainder of this paper is organized as follows: in Sect. 2, we introduce radiomap model and related works. In Sect. 3, we present our methodologies of AdaMap. Extensive simulation results are shown in Sect. 4. Final section is conclusion and future work.

2 Background and Problem Model

2.1 Radiomap Locating Method

Radiomap locating is essentially a *pattern-matching* based approach, which offline learns the RSS signatures of the a set of locations to construct a *radio-map* and online searches in the radio-map to find the location whose RSS fingerprint best matches the measured RSS at the location to be determined [1,2,9]. In offline phase, k training locations are selected in the sensing field, which are denoted by $\mathbf{L} = \{l_1, l_2, ..., l_k\}$. Suppose there are n beacons (WiFi APs or wireless sensors) in the sensing field, which are denoted by $\mathbf{B} = \{b_1, b_2, ..., b_n\}$. In the training phase, the RSS values of all beacons at each training location l_i will be measured over a period of time. So that a signal signature vector of location l_i is constructed as $\mathbf{r}_i = \{r_{i,1}, r_{i,2}, r_{i,n}\}$, where $r_{i,j}$ is inferred deterministically or probabilistically by the RSS values measured from beacon b_j at location l_i. When only the mean value of RSS is considered, $r_{i,j}$ represents the average RSS value [9]. When signature distribution is considered [10–12], $r_{i,j}$ can be probabilistic density function (pdf) of RSS. The signature vectors of all training locations are stored as a database, called *radiomap*, denoted by $\mathbf{R} = \{r_1, r_2, ..., r_k\}$.

In the online positioning phase, a mobile target measures its current RSS vector $\mathbf{s} = s_1, s_2, ..., s_n$ and finds the best match (Euclidean distance in signal space) of $\mathbf{s}$ in $\mathbf{R}$ to estimate its location. In mean value type radio-map, matching can be conducted by Nearest Neighbor algorithm [9]. In probabilistic radio-maps, maximum likelihood estimate and Bayesian estimation were applied to improve the positioning accuracy [11].

2.2 Radio-Map Adaptation by Inter-Beacon Signatures

The radiomap can be outdated when environment changes, but re-training radiomap will be time-consuming and laborious. Existing work investigated radiomap adaptation methods [3,4,7,13]. Most of these methods use additional reference points (with known locations), sensors or other resources to help handle the RSS dynamics. Instead of using the additional reference points or additional sensors, [7] proposed to use the beacons themselves as reference points to construct linear regression model, which can relieve the additional hardware costs. This method explores the facts that some off-the-shelf WiFi stations can measure RSSs from other beacons. We introduce this method briefly before presenting our solution.

Let's consider there are n beacons $b_1, b_2, \cdots, b_n$ in the network and let l be a training location. In offline phase, to train the signal signature of beacon i at location l, i.e., $d_{i,l}$, the inter-beacon RSS vector $\{r_{i,1}^t, r_{i,2}^t \cdots, r_{i,n}^t\}$ are measured over a period of time, where $r_{i,j}^t, j \neq i$ represents the RSS of beacon i measured by beacon j at time t. Then a linear regression model is built by assuming the RSS at location l, i.e., $d_{i,l}^t$ is a linear combination of the inter-beacon RSSs vector:

$$d_{i,l}^t = \alpha_{i,1}^l r_{i,1}^t + \alpha_{i,2}^l r_{i,2}^t + \cdots + \alpha_{i,n}^l r_{i,n}^t + \alpha_{i,n+1}^l. \tag{1}$$

So that by the measurements from 1 to T, the following equation can be constructed to calculate the coefficient vector. Least square estimation can be applied and the equation can be made overdetermined by improving T.

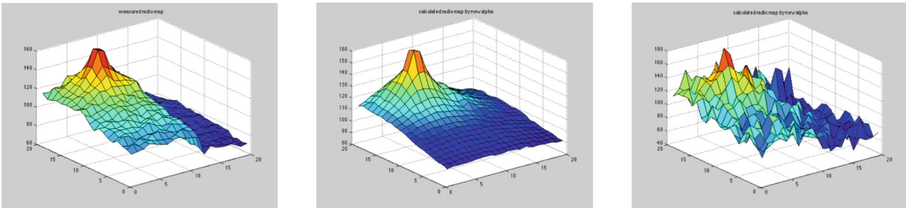
$$\begin{bmatrix} r_{i,1}^{(1)} & \cdots & r_{i,n}^{(1)} & 1 \\ \vdots & \cdots & \vdots & \vdots \\ r_{i,1}^{(T)} & \cdots & r_{i,n}^{(T)} & 1 \end{bmatrix} \begin{bmatrix} \alpha_{i,1}^l \\ \vdots \\ \alpha_{i,n}^l \\ \alpha_{i,n+1}^l \end{bmatrix} = \begin{bmatrix} d_{i,l}^1 \\ \vdots \\ d_{i,l}^T \end{bmatrix} \tag{2}$$

So that the RSS fingerprint at each location l is no longer a static value or a vector, but an adaptive value calculated by the linear regression function. In online phase, the inter-beacon RSS $\{r_{i,1}^t, r_{i,2}^t, \cdots, r_{i,n}^t\}$ are measured, and are substituted into the linear regression function to calculate the RSS value at location l, which is denoted by $d_{i,l}^c$. Since the inter-beacon RSS values $\{r_{i,1}^t, r_{i,2}^t, \cdots, r_{i,n}^t\}$ change adaptively with the environments, the changes are embodied by the linear regression model to make $d_{i,l}^c$ be adaptive to the environment.

However, in this linear regression model (LRM), once the linear coefficients are calculated in offline phase, the coefficients will not change in online phase. When environment changes slowly, the self-adaptivity of LRM could keep the effectiveness of the radio-map model to provide reasonable locating accuracy. While in case striking environment change happens, the linear regression model may lose the correctness in modeling the RSS signatures. We call this *partial-adaptivity* problem.

Figure 1 uses an example to show the *partial-adaptivity* problem, i.e., ineffectiveness of the linear regression model in case the environment changes dramatically. We simulate a block effect on the experimental environment, i.e., a big block is placed in the sensing field. Any signal that passes through the block would have 20*db* signal attenuation.

Figure 1 presents online calculated radiomap by LRM with same measured radiomap while facing two different environment scenarios. Compared to the same measured radiomap from Fig. 1(a), which is accurate, Fig. 1(b) shows that when environment doesn't change, the self-adaptivity of LRM can ensure reasonable locating accuracy. While Fig. 1(c) shows that when facing block effect, which means striking environment changes, LRM fails to perform locating and the calculated radiomap differs much from measured radiomap in Fig. 1(a).



(a) Measured Radiomap

(b) Online Calculated Radiomap, when environment doesn't change

(c) Online Calculated Radiomap, when environment changes by block effect

Fig. 1. Block effect on LRM model locating

3 Adaptive Radiomap Model

3.1 Overview

To solve above problems, we present AdaMap, an efficient and flexible online radiomap adaptation model. It not only trains the linear regression model to make the radio signatures be adaptive to environment dynamics, but also can online update the coefficients of the linear model by judging whether the calculated inter-beacon radiomap differs much from the measured one. The overview of the AdaMap model is shown in Fig. 2.

In offline phase, we apply LRM to calculate offline coefficients by solving equation (2). In online phase, the trained offline coefficients are utilized to calculate the RSS signature of each location. These calculated signatures are then compared with the measured RSS signature of the target. The location whose signature best matches the measured RSS signature of the target will be determined as the real-time position of the target.

Different from existing approaches, Adamap designs a judgment step to determine whether the trained coefficients by LRM are outdated. If the coefficients are detected to be outdated, efficient algorithms to update the coefficients of LRM model were designed. The updated coefficients make the LRM be adaptive

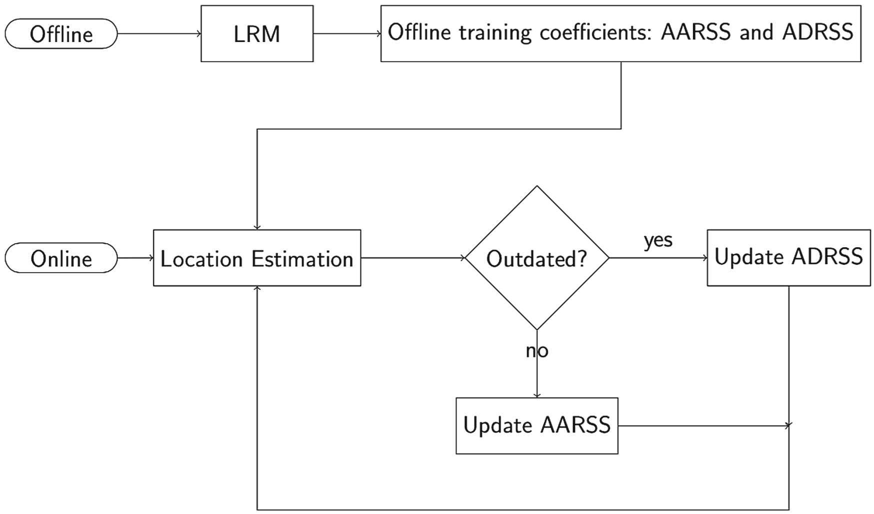


Fig. 2. Flowchart of AdaMap.

to the environment changes. The whole process ensures that the calculated RSS signatures at each location be adapt to the environment dynamics by two ways: (1) LRM model; (2) coefficients updating scheme of LRM model.

To present the coefficient updating scheme of Adamap clearly, we define AARSS as short for AP-to-AP RSS:

$$r^c_{i,j} = \sum_{v=1, v\neq i}^{n} \beta_{i,j,v} r^m_{i,v} + \beta_{i,j,n+1} \cdot 1 \tag{3}$$

$r^c_{i,j}$ is the calculated RSS value from beacon i to beacon j. It is represented by the linear combination of the measured RSS values from beacon i to other beacons. β is the coefficient vector, and the superscript c and m means calculated and measured RSS respectively.

We define ADRSS as short for AP-to-Device RSS, which indicates the beacon to device's location RSS values.

$$d^c_{i,l} = \sum_{j=1, j\neq i}^{n} \alpha_{i,j,l} r^m_{i,j} + \alpha_{i,n+1,l} \cdot 1 \tag{4}$$

$d^c_{i,l}$ is the calculated RSS value at location l of beacon i, which is represented by the linear combination of inter-beacon RSS values. α in (4) represents the coefficients. The superscript c and m represents calculated and measured value respectively.

3.2 AdaMap Framework

AdaMap is designed based on above two linear regression models. It contains two phases: (1) offline training and (2) online adapting.

Step 1: Offline Training. Compared to traditional LRM model [7], in offline phase, in addition to the beacon-to-device LRM model as given in Eq. (4), the AP-to-AP LRM model shown in Eq. (3) is also trained. The matrix form of these two LRM models are as following:

$$\begin{bmatrix} d_{i,l}^{1} \\ . \\ d_{i,l}^{T} \end{bmatrix} = \begin{bmatrix} r_{i,1}^{1} \; r_{i,2}^{1} \; ... \; r_{i,n}^{1} \; 1 \\ . \\ r_{i,1}^{T} \; r_{i,2}^{T} \; ... \; r_{i,n}^{T} \; 1 \end{bmatrix} \begin{bmatrix} \alpha_{i,1,l} \\ . \\ \alpha_{i,n,l} \\ \alpha_{i,n+1,l} \end{bmatrix} \tag{5}$$

$$\begin{bmatrix} r_{i,j}^{1} \\ . \\ r_{i,j}^{T} \end{bmatrix} = \begin{bmatrix} r_{i,1}^{1} \; r_{i,2}^{1} \; ... \; r_{i,n}^{1} \; 1 \\ . \\ r_{i,1}^{T} \; r_{i,2}^{T} \; ... \; r_{i,n}^{T} \; 1 \end{bmatrix} \begin{bmatrix} \beta_{i,j,1} \\ . \\ \beta_{i,j,n} \\ \beta_{i,j,n+1} \end{bmatrix} \tag{6}$$

The could be simplified as $\mathbf{D_{i,l}} = \mathbf{A}\alpha_{\mathbf{i,l}}$ and $\mathbf{R_{i,j}} = \mathbf{B}\beta_{\mathbf{i,j}}$. By using least square method, we can calculate the offline ADRSS coefficients and AARSS coefficients as (7) and (8).

$$\alpha_{\mathbf{i,l}} = (\mathbf{A^T A})^{-1}\mathbf{A^T D_{i,l}} \tag{7}$$

$$\beta_{\mathbf{i,j}} = (\mathbf{B^T B})^{-1}\mathbf{B^T R_{i,j}} \tag{8}$$

Note that $\alpha_{\mathbf{i,l}}$ is the ADRSS coefficient vector of beacon i at location l. $\beta_{\mathbf{i,j}}$ is the AARSS coefficient vector from beacon i to beacon j.

Step 2: Online Adapting. In online phase, $r_{i,j}^{c}$ is calculated by (3) and $r_{i,j}^{m}$ is measured in real-time. Then we judge whether $r_{i,j}^{c}$ is close to $r_{i,j}^{m}$ to decide whether to update the coefficient vector or not.

(1) **If $r_{i,j}^{c}$ is close to $r_{i,j}^{m}$** ($||r_{i,j}^{c} - r_{i,j}^{m}|| \leq H$), where H is a threshold, we consider the ADRSS model (4) is still effective, which don't need to be updated. We can calculate the RSS signature at location l of beacon i by (4).

(2) **If $r_{i,j}^{c}$ differs greatly from $r_{i,j}^{m}$** ($||r_{i,j}^{c} - r_{i,j}^{m}|| > H$), which means that the trained LRM model is outdated. Then we firstly collect the inter-beacon RSS measurements to update the inter-beacon coefficients β by (8). The updated coefficient vector is denoted by β'. Then we show how to update α.

$$d_{i,l}^{c} = \sum_{j=1, j\neq i}^{n+1} \alpha_{i,j,l} \left\{ \sum \beta'_{i,j,v} r_{i,v}^{m} \right\} = \sum_{j=1, j\neq i}^{n+1} \alpha'_{i,j,l} r_{i,j}^{m} \tag{9}$$

By simple mathematical derivation, we can conclude the updated coefficients α' as shown in (10).

$$\alpha'_{i,j,l} = \sum_{k=1, k\neq i, k\neq j}^{n+1} \alpha_{i,k,l} \beta'_{i,k,j} \tag{10}$$

From the derivation, we can see clearly that the ADRSS coefficients α can be updated online.

Step 3: Device Location Estimation. Finally we can calculate the estimated location $d_{i,l}^c$ with updated coefficients and inter-beacon RSS values, as in the traditional methods.

3.3 Algorithm Properties

The adaptive model solves the problem that linear regression model coefficients cannot adapt online. The algorithm pseudo code is written as Algorithm 1. The computation complexity of Algorithm 1 can be analyzed as follows. Solving the least square estimation has complexity $O(n^3)$, n is beacon number. In online phase, the outloop has complexity $O(n^2)$, updating coefficients inside the loop has complexity $O(nm)$, m is training location number. Therefore, the overall complexity of this algorithm is $O(n^2(n^3 + nm))$. It is affordable, since n is generally not large.

Input: $\{r_{i,j}^m\}$ and $\{d_{i,l}^m\}$ at running times $t = 1, 2, \cdots, T$, AP number n and location number k, threshold H
Use $\{r_{i,j}^m\}$ and $\{d_{i,l}^m\}$ to construct $B_{i,j}$, A_i, and $R_{i,j}$ within training time T
Calculate AARSS training coefficients $\beta = (B^T B)^{-1} B^T R$
Calculate ADRSS training coefficients $\alpha = (A^T A)^{-1} A^T D$
while *(system is running)* **do**
  for $i = 1; i \leq n; i++$ **do**
    for $j = 1; j \leq n, j \neq i; j++$ **do**
      Calculate $r_{i,j}^c$ with $B_{i,j}$ and $\beta_{i,j}$
      if $\left\| r_{i,j}^c - r_{i,j}^m \right\|_2 < H$ **then**
        update ADRSS coefficients $\alpha_{i,l}$
      end
      else
        Construct $B_{i,j}$ and $R_{i,j}$, *both at time* t, and update AARSS coefficients: $\beta_{i,j} = (B_{i,j}^T B_{i,j})^{-1} B_{i,j}^T R_{i,j}$
      end
    end
    Calculate location estimation $d_{i,l}^c$ with updated coefficients
  end
end

Algorithm 1. Algorithm for AdaMap Model

4 Performance Evaluation

Simulations were conducted to evaluate the performance of AdaMap compared with simple linear regression model (LRM). More specifically, the locating accuracy and robustness of AdaMap locating against block effect and temperature effect were evaluated in this section.

4.1 Simulation Settings

We conducted simulation in MATLAB environment. Area is partitioned into grids, and a number of beacons are deployed randomly in the area, they all possess the same initial transmission power, and environment noise follows a gaussian distribution. The target moves in this area, and its transmission power can be received by beacons. In our experiment, the parameters are set as Table 1. We use 20∗20 area size, the beacon number is set as 10, initial transmission power is set as 150. The gaussian noise variance is set as 2. The target follows a sine wave path in the area. Besides, in order to verify the performance convincingly, the 10 APs are deployed with random location generation algorithm in the area.

Table 1. Setting of simulation

Parameters	Value
Area size	20*20
Beacon number	10
Transmission power	150
Noise variance	2
Target trajectory	Sine wave in the area

4.2 Environment Dynamics

To verify our adaptive algorithm performance on environment dynamics, we consider the 3 following cases:

(1) **Free Space:** environment doesn't change, the signal transmission follows the propagation model [9]. The model is shown as Eqn (11).

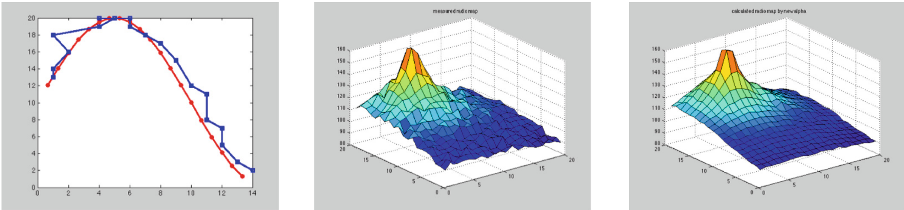
$$r = p_t - p_0 - 20 * \log(d) + \sigma \tag{11}$$

In this formula, p_t is transmitter's initial power, p_0 is constant term, d is the distance between transmitter and receiver, σ is environment noise which follows normal gaussian distribution, r is the power received from transmitter.

(2) **Block Effect:** we simulate a big block moving in a constant velocity inside the sensing area. The block would cause specific amount of signal attenuation for signals passing through the block, and we use *blockeffect* to define this value loss;

(3) **Temperature Effect:** the temperature would cause the transmission powers of all the beacons decrease slowly, and we use *powerdecrease* to express the decrease value.

In the latter two cases, AdaMap and LRM model would be applied to compare the locating performance.

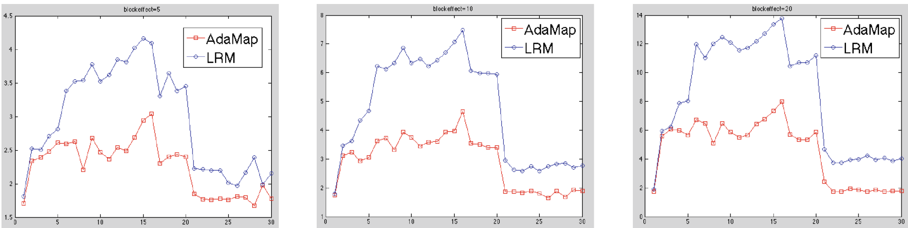


(a) Tracking performance (b) Measured Radiomap (c) Online Calculated Radiomap

Fig. 3. When environment doesn't change

4.3 Simulation Results

Environment Doesn't Change. Fig. 3 presents LRM model locating performance when environment doesn't change. In Fig. 3(a), the red line is the real path, and blue line is estimated path. In our simulation, the real path is a sine wave in the area, in each time slot, we apply LRM model to calculate the most possible estimated location. We can see that the tracking performance is quite good when the environment doesn't change, Fig. 3(b) and(c) shows that the online calculated radio map matches the measured radiomap quite well, which means that LRM performs reasonably well in free space.



(a) be=5 (b) be=10 (c) be=20

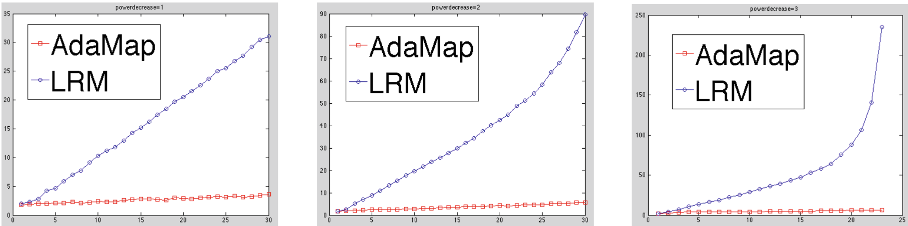
Fig. 4. Block effect radiomap loss: AdaMap vs. LRM

Block Effect and Temperature Effect. We define *radiomapLoss* to present the similarity between measured radiomap and online calculated radiomap by AdaMap and LRM models at each time slot. Suppose measured radiomap is denoted as 2D matrix **A**, online calculated radiomap is denoted as 2D matrix **B**, the *radiomapLoss* is defined as (12).

$$radiomapLoss = \frac{sum(abs(A-B))}{row * column} \tag{12}$$

The definition shows that if *radiomapLoss* is smaller, the measured radiomap and online calculated radiomap is more similar, which means more locating accuracy.

Figure 4 shows the comparison of radiomap loss between AdaMap and LRM when facing block effect. Red line and blue line are online radiomap calculated by AdaMap and LRM respectively. Figure 4(a), (b) and (c) present the radiomap loss when *blockeffect(be)* equals 5, 10 and 20 dB respectively. Let's compare the $radiomapLoss$ for the two models in each figure. When $blockeffect$ equals $5, 10$ and 20, the AdaMap's calculated $radiomapLoss$ ranges from $[1.5, 3]$, $[1, 5]$ and $[2, 8]$ approximately, while LRM's calculated $radiomapLoss$ ranges from $[1.5, 4.5]$, $[1, 8]$ and $[2, 14]$ approximately. As the $blockeffect$ increases, both LRM and AdaMap's $radiomaoLoss$ increases, LRM model can partially adapt to the increasing block effect, so that the range is not that wide, while AdaMap's range is less than LRM in all scenarios, which means that when facing block effect, AdaMap performs better locating accuracy when compared to LRM.

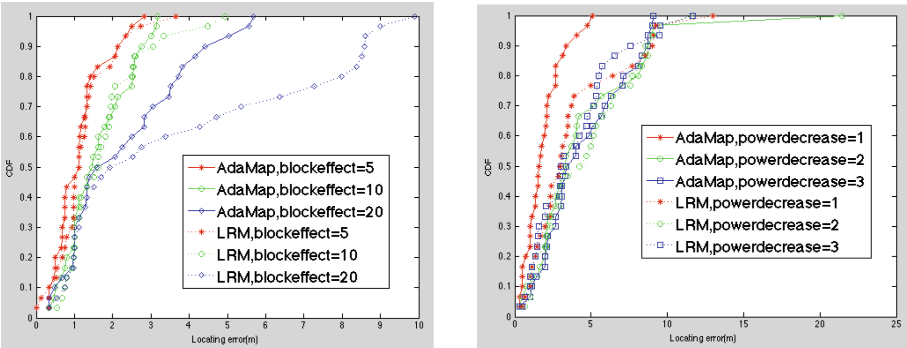


(a) pd=1 (b) pd=2 (c) pd=3

Fig. 5. Power transmission decrease radiomap loss: AdaMap vs. LRM

Figure 5 shows the comparison of radiomap loss between AdaMap and LRM when facing power transmission decrease. Figure 5(a), (b) and(c) present the radiomap loss when $powerdecrease(pd)$ equals 1, 2 and 3 respectively. Let's compare the $radiomapLoss$ for the two models in each figure. When $powerdecrease$ equas $1, 2$ and 3, the AdaMap's calculated $radiomapLoss$ ranges from $[1.7, 3.6]$, $[1.7, 5.7]$ and $[1.7, 6.5]$, while LRM's calculated $radiomapLoss$ ranges from $[1.7, 31.1]$, $[1.7, 89.7]$ and $[1.7, 235.0]$ approximately. The power transmisson decrease effect obviously affect LRM's locating accuracy, the $radiomapLoss$ increases almost exponentially, which means the failure of locating by LRM, while AdaMap's adaptivity presents more stable(only increases from 3.6 to 6.5 with worst case) and less radiomap loss when facing power transmission decrease, which means our AdaMap model successfully achieves better locating accuracy.

Figure 6 shows the locating performance comparison between AdaMap model and LRM model. Block effect CDF errors are presented in Fig. 6(a). The solid lines and dotted lines are AdaMap CDF errors and LRM CDF errors respectively, the comparison shows that as $blockeffect$ increases, AdaMap error increases smaller than LRM and achieve better locating accuracy. Temperature effect CDF errors are presented in Fig. 6(b). The result shows that although $powerdecrease$ decreases AdaMap locating accuracy because of inherent effect, AdaMap CDF errors are still smaller than LRM.



(a) block effect (b) temperature effect

Fig. 6. AdaMap model vs. LRM model

In summary, these simulation results verified the locating efficiency of AdaMap model. They show that better satisfactory locating accuracy can generally be obtained by AdaMap model compared with LRM model.

5 Conclusion and Future Work

Considerable research efforts were dedicated to develop wireless networks as indoor location infrastructure, we have investigated current locating methods, in which radiomap based locating method can achieve meter level positioning accuracy when the radiomap is trained to be fine-grained. However, the challenging problem is that the radiomap would be outdated when environment changes. The reconstruction would be time-consuming and laborious. In this paper, we present AdaMap, a novel radio map model to deal with environment changes. We exploit the inter-beacon RSS to construct adaptive radiomap based on an online self-adjusted linear regression model. AdaMap not only trains the linear regression model to make the radio signatures be adaptive to environment changes, but also can online update the coefficients according to online inter-beacon RSS measurements. The updating algorithms ensures the efficiency and accuracy of AdaMap locating. The simulation results that shows our proposed method can efficiently make the trained radiomap model online adapt to the environment dynamics and the signal dynamics. It can help improve the locating accuracy of radiomap based algorithms without the pain of recalibration of the radio maps. In the future, work should focus on the online adaptive algorithm optimization and complete model performance evaluation in real systems.

Acknowledgments. This work was supported by in part by National Natural Science Foundation of China Grant 61202360, 61073174, 61033001, 61061130540, the Hi-Tech research and Development Program of China Grant 2006AA10Z216, and the National Basic Research Program of China Grant 2011CBA00300, 2011C-BA00302.

References

1. Haque, I.T., Assi, C.: Profiling-based indoor localization schemes (2013)
2. Scholl, P.M., et al. Fast indoor radio-map building for RSSI-based localization systems. In: 2012 Ninth International Conference on Networked Sensing Systems (INSS), IEEE (2012)
3. Ni, L.M., et al.: LANDMARC: indoor location sensing using active RFID. Wireless Netw. **10**(6), 701–710 (2004)
4. Yin, J., Yang, Q., Ni, L.M.: Learning adaptive temporal radio maps for signal-strength-based location estimation. IEEE Trans. Mob. Comput. **7**(7), 869–883 (2008)
5. Bernardos, A.M., Casar, J.R., Tarro, P.: Real time calibration for rss indoor positioning systems. In: 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE (2010)
6. Chen, Y.-C., et al.: Sensor-assisted wi-fi indoor location system for adapting to environmental dynamics. In: Proceedings of the 8th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems. ACM (2005)
7. Lo, C.-C., Hsu, L.-Y., Tseng, Y.-C.: Adaptive radio maps for pattern-matching localization via inter-beacon co-calibration. Pervasive Mob. Comput. **8**(2), 282–291 (2012)
8. El-Kafrawy, K., et al.: Propagation modeling for accurate indoor WLAN RSS-based localization. In: 2010 IEEE 72nd Vehicular Technology Conference Fall (VTC 2010-Fall), IEEE (2010)
9. Bahl, P., Padmanabhan, V.N.: RADAR: An in-building RF-based user location and tracking system. In: Proceedings Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000), Vol. 2, IEEE (2000)
10. Roos, T., et al.: A probabilistic approach to WLAN user location estimation. Int. J. Wirel. Inf. Netw. **93**, 155–164 (2002)
11. Dieter, F., et al.: Bayesian filtering for location estimation. IEEE Pervasive Comput. **2**(3), 24–33 (2003)
12. Youssef, M.A., Agrawala, A., Udaya Shankar, A.: WLAN location determination via clustering and probability distributions. In: Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, (PerCom 2003). IEEE (2003)
13. Yin, J., Yang, Q., Ni, L.: Adaptive temporal radio maps for indoor location estimation. In: Third IEEE International Conference on Pervasive Computing and Communications, (PerCom 2005). IEEE (2005)
14. Roberts, B., Pahlavan, K.: Site-specific RSS signature modeling forWiFi localization. In: IEEE Global Telecommunications Conference, GLOBECOM. IEEE (2009)
15. Atia, M.M., Noureldin, A., Korenberg, M.J.: Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks. IEEE Trans. Mob. Comput **12**(9), 1774–1787 (2013)
16. Pahlavan, K., Levesque, A.H.: Wireless Information Networks, vol. 95. Wiley, New York (1995)
17. Sharma, P., et al.: KARMA: Improving WiFi-based indoor localization with dynamic causality calibration. In: 2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), IEEE (2014)

18. Pan, S.J., et al.: Adaptive localization in a dynamic wifi environment through multi-view learning. In: Proceedings of the National Conference on Artificial Intelligence, 22(2). AAAI Press, MIT Press, London, Cambridge (1999, 2007)
19. Liu, H., et al.: Survey of wireless indoor positioning techniques and systems. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **37**(6), 1067–1080 (2007)
20. Gu, Y., Lo, A., Niemegeers, I.: A survey of indoor positioning systems for wireless personal networks. Commun. Surv. Tutorials IEEE. **11**(1), 13–32 (2009)
21. Yang, Z., Wu, C., Liu, Y.: Locating in fingerprint space: wireless indoor localization with little human intervention. In: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking. ACM (2012)
22. Rai, A., et al.: Zee: zero-effort crowdsourcing for indoor localization. In: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking. ACM (2012)
23. Sujak, B., et al.: Indoor propagation channel models for WLAN 802.11 b at 2.4 GHz ISM band. In: Asia-Pacific Conference on Applied Electromagnetics, APACE 2005. IEEE (2005)
24. Ji, Y., et al.: Impact of building environment on the performance of dynamic indoor localization. In: IEEE Annual Wireless and Microwave Technology Conference, WAMICON 2006. IEEE (2006)
25. Geng, X., Wang, Y., et al.: Hybrid radio-map for noise tolerant wireless indoor localization. In: 2014 IEEE 11th International Conference on Networking, Sensing and Control (ICNSC), 233–238, April 2014
26. Wu, C., et al.: WILL: Wireless indoor localization without site survey. IEEE Trans. Parallel Distrib. Syst. **24**(4), 839–848 (2013)
27. Ji, Y., Player, R.: A 3-D indoor radio propagation model for WiFi and RFID. In: Proceedings of the 9th ACM International Symposium on Mobility Management and Wireless Access. ACM (2011)