Fingerprinting Localization in Wireless Networks Based on Received Signal Strength Measurements:

A case study on WiMAX networks

Mussa Bshara, *Member, IEEE,* Umut Orguner, *Member, IEEE,*

Fredrik Gustafsson, *Senior Member, IEEE,*

and Leo Van Biesen, *Senior Member, IEEE*

Abstract

This paper considers the problem of fingerprinting localization in wireless networks based on received signal strength (RSS) observations. First, the performance of static localization using power maps (PM) is improved with a new approach called base station strict (BS-strict) methodology which emphasizes the effect of BS identities in the classical fingerprinting. Second, dynamic motion models with and without road network information are used to further improve the accuracy via particle filters. The likelihood calculation mechanism proposed for the particle filters is interpreted as a soft version (called BS-soft) of the BS-strict approach applied in the static case. The results of the proposed approaches are illustrated and compared on an example whose data were collected from a WiMAX network in a challenging urban area in the Brussels capitol city.

Index Terms

Mussa Bshara and Leo Van Biesen are with the Department of Fundamental Electricity and Instrumentation, Vrije Universiteit Brussel, Brussels, Belgium (e-mail: {mbshara,lvbiesen}@vub.ac.be).

Umut Orguner and Fredrik Gustafsson are with Department of Electrical Engineering, Linköping University, 581 83, Linköping, Sweden (e-mail: {umut,fredrik}@isy.liu.se).

Fingerprinting, GPS, GSM, navigation, LBS, path loss model, positioning, positioning accuracy, power maps, received signal strength, road network information, RSS, SCORE, time of arrival, WiMAX.

I. INTRODUCTION

There are several ways to position a wireless network user. GPS is the most popular way, its accuracy meets all the known location-dependent applications requirements. The main problems with GPS, in addition to the fact that the user's terminal must be GPS-enabled, are the high battery consumption, the limited coverage and the latency. Furthermore, GPS performs poorly in urban areas near high buildings and inside tunnels. Another way to position a user is to rely on the wireless network itself by using the available information like Cell-ID which has been used widely in GSM systems despite its limited accuracy [1]. Using other network resources (information) like received signal strength (RSS), time of arrival (TOA) or time-difference of arrival (TDOA) gives better accuracy, but requires making measurements by the wireless terminal (terminal side measurements) or by the network (network side measurements) or by both [2, 3]. From this point on, we are going to refer to these measurements as network measurements regardless of where these measurements have been conducted. Some of these measurements are hard to obtain like TOA which needs synchronization and some are easy to obtain like RSS measurements. Many localization approaches depending on network measurements have been proposed in GSM networks and sensor networks. Most of the work focused on range measurements depending on TOA, TDOA observations and RSS observations, see surveys [2, 4, 5] and the references therein. These approaches can improve the localization accuracy achieved by using the Cell-ID. The basic idea in RSS based localization is to compare all measured RSS values to a model of RSS for each position and then determine the position that gives the best match. The two most common models are the general exponential path loss model and a dedicated power map constructed off-line for the region of interest. The first alternative is the most common strategy and the simplest to deploy. The exponential path loss model is known as the Okumura-Hata model [6, 7], and in a log power scale it says that the RSS value decreases linearly with distance to the antenna. This is quite a crude approximation, where the noise level is high and further depends on multipath and non-line of sight (NLOS) conditions. In [8] the authors used this alternative to track a target and proposed using different path loss exponents for the links

between the terminal and the base stations. The proposed method achieved higher localization accuracy than the conventional localization methods that use the same path loss exponent for all the links. Also the authors of [9] proposed using RSS statistical log-normal model and sequential Monte Carlo localization (MCL) technique to get better localization accuracy. The log-normal model was also used in [10] to estimate mobile location, and the authors tried to mitigate the influence of the propagation environment by using the differences of signal attenuations.

The second alternative is to determine the RSS values in each point and save this in a database (i.e., a map). This can be done using off-line measurement campaigns, adaptively by contribution from users or using cell planning tools. The advantage of this effort is a large gain in signal to noise ratio and less sensitivity to multipath and NLOS conditions. The set of RSS values that are collected for each position in the map from various BSs is called the *fingerprint* for that location. The idea of matching observations of RSS to the map of the previously measured RSS values is known as *fingerprinting*. It proved to provide better performance than the first alternative [1]. In [11] and [12], the authors used RSS information in fingerprinting positioning to improve the accuracy obtained by the log-normal model. The authors of [13] used fingerprinting to overcome the inconveniences related to the use of TOA, angle of arrival (AOA) and RSS log-normal model for positioning.

In this paper, we propose to use fingerprinting localization depending on RSS-based observations for positioning and tracking in wireless networks. We first consider classical fingerprinting and, based on the BS identities, we propose a method to increase fingerprinting performance. The new method emphasizes the effects of the BS identities in classical fingerprinting and it is called BS-strict method. Then, the use of dynamic motion models is suggested for further improvement. In this regard, we use particle filters [14, 15, 16] with both unconstrained and road-constrained motion models. The simultaneous use of the motion models and the road network information has shown to yield quite good estimation performance. The special likelihood calculation mechanism that the paper suggests for the dynamic case, called BS-soft method, is also interpreted as a soft version of the BS-strict methodology proposed for the static case. We present our results along with remarks on WiMAX networks which were the main motivation and the illustrative case study for this research. However, our results equally apply to other type of networks. The importance of the contributions of this paper can be summarized as follows:

- The proposed approaches yield direct methodologies for RSS-based localization balancing

the effects of measured RSS values and the BS identities. Increasing the effect of BS identities in location estimation is especially significant when the signal to noise ratio in the RSS values is low and the effects of multipath and fading are dominant.

- Dynamic localization using particle filters gives a seamless integration of fingerprinting type approaches with dynamical motion models and road network information.

We also argue that the approaches considered in the paper meet the requirements of most location-dependent applications.

The paper is organized as follows: The measurement modeling methodologies for the RSS measurements are summarized in Section II. The main building blocks of the proposed methods, which are different likelihood calculation mechanisms, are given as separate algorithms in Section III and Section V for the static and dynamic estimation cases respectively. These algorithms are used in their corresponding positioning and tracking methods and their performances are illustrated in Sections IV and VI respectively. Conclusions are drawn in Section VII.

II. MODELLING RSS MEASUREMENTS FOR FINGERPRINTING

In general, the received signal r_t at the time instant t can be expressed as

$$r_t = a_t s_{t-\tau} + v_t. \tag{1}$$

Here, s denotes the transmitted (pilot) signal waveforms, a_t is radio path attenuation, τ is the distance dependent delay and v_t is a noise component. A WiMAX modem does not readily provide information for time delay based localization, so we focus on the path loss constant a_t. This value is averaged over one or more pilot symbols to give a sampled RSS observation

$$z_k = h(x_k^p) + e_k, \tag{2a}$$

$$y_k = \begin{cases} z_k, & \text{if } z_k \geq y_{min}, \\ \text{NaN}, & \text{if } z_k < y_{min}. \end{cases} \tag{2b}$$

where k is the sample index (corresponding to time instant $t = t_0 + kT$ where t_0 and T are the time of the first sample ($k = 0$) and the sampling period respectively), x_k^p is the position of the target and NaN stands for Not a Number representing a "non-detection" event. This expression includes one deterministic position dependent term $h(x_k^p)$ including range dependence, and e_k is

the noise that includes fast and slow fading. We also explicitly model the detector in the receiver with the threshold y_{min}, since too weak signals are not detected.

The classical model of RSS measurements is based on the so-called Okumura-Hata model [6, 7] which is given as

$$\text{OH model:} \quad z_k = P_{BS} - 10\alpha \log_{10}(\|p_{BS} - x_k^p\|_2) + e_k. \tag{3}$$

where P_{BS} is transmitted signal power (in dB); α is the path loss exponent; e_k is the measurement noise and p_{BS} is the position of the antenna; the standard $\| \cdot \|_2$ norm is used. This model has been used in many proposed localization algorithms [2, 17]. Though being a global and simple model, there are several problems associated with using it:

- The transmitted power needs to be known, which requires a protocol and software that allows higher layer of applications to access this information.

- The position of the antenna needs to be known. This requires first building a database, and second that the user application can access the identification number of each antenna connected to the model. Third, the operators in some countries consider the position of their antennas to be classified.

- The path loss constant needs to be known, while it, in practice, depends on the local environment.

An alternative model is based on a local power map (LPM), which is obtained by observing the measurement y_k over a longer time and over a local area. Each LPM item is then computed as a local average

$$\text{LPM model:} \quad \hat{z}(x) = \hat{\mathrm{E}}(y) = \hat{\mathrm{E}}(h(x) + e) \tag{4a}$$

$$\hat{h}(x) = \begin{cases} \hat{z}(x), & \text{if } \hat{z}(x) \geq y_{min}, \\ \text{NaN}, & \text{if } \hat{z}(x) < y_{min}. \end{cases} \tag{4b}$$

where the operator $\hat{\mathrm{E}}$ denotes the corresponding averaging. LPM provides a prediction of the observation (2) in the same way as the OH model in (3). However, the LPM should be considered more accurate since it implicitly takes care of the LOS/NLOS problems that are difficult to handle [18]. The LPM model also partially includes the effects of slow and fast fading. The total effect can be approximated as a gain in SNR with a factor of ten compared to the OH model, see [2].

The collection of averaged measurements $\hat{h}(x)$ for the same position in a single vector gives us the *fingerprint* $\hat{\mathbf{h}}(x)$ for that position i.e.,

$$\hat{\mathbf{h}}(x) \triangleq \begin{bmatrix} \hat{h}_1(x) & \hat{h}_2(x) & \ldots & \hat{h}_{N_{BS}}(x) \end{bmatrix}^T \tag{5}$$

where N_{BS} is the number of BSs and $\hat{h}_j(x)$ is the averaged measurement from the jth BS at the position x. The advantage of collecting fingerprints in a database is that prior knowledge of antenna position, transmitted power or path loss constant is not needed, enabling mobile-centric solutions. The price for this is the cumbersome task to construct the LPM. Here, three main alternatives are plausible:

1) Collect the fingerprints during an off-line phase. Measurements to be stored have to be collected from all possible places where the target can be and under various weather conditions at different times in the area under study. This method gives the most accurate database but it is money and time consuming.

2) Use the principle of *wardriving* [19], where the users on-line contribute to the LPM. The idea is that users with positioning capabilities (for instance GPS) report their position and observations (2) to a database [20, 21], which is used for positioning other users.

3) Predict the fingerprints using Geographical Information System (GIS) planning tools [2]. Using the radio propagation formulas to predict the RSSs is not as accurate as measuring them because it is not possible to model all the propagation effects. As a result, the predicted data is not as accurate as the measured one but it is quite easy to obtain.

In this study, the first method was adopted, and the WiMAX RSS values have been collected from all the possible roads in the area under study (we assume that the target or the user is using the public road network) during an off-line phase. The LPM has been formed from this database as follows.

- N_{LPM} different grid points denoted as $\{p^i \triangleq [x^i, y^i]^T\}_{i=1}^{N_{LPM}}$, where x^i and y^i denote the x and y coordinates of the ith point respectively, have been selected on the road network. A maximum distance of 10m has been left between these LPM points.

- For each piece of data that has been collected, the closest LPM grid point has been found.

- For each LPM grid point i, the vector $\hat{\mathbf{h}}^i$ (called "RSS vector" or fingerprint) is formed

such that

$$\hat{\mathbf{h}}^i = \left[\begin{array}{cccc} \hat{h}_1^i & \hat{h}_2^i & \ldots & \hat{h}_{N_{BS}}^i \end{array}\right]^T \tag{6}$$

where $\hat{h}_j^i$ is the mean of the RSS data from the jth BS assigned to the ith LPM grid point. If there is no RSS data from the jth BS assigned to the ith LPM grid point, we set $\hat{h}_j^i = \text{NaN}$ representing a non-detection. Note that each fingerprint (or RSS) vector $\hat{\mathbf{h}}^i = \hat{\mathbf{h}}(p^i)$, is a representative of the expected RSSs at the position p^i.

The measured RSS values at the time of localization are then collected in another RSS vector $\mathbf{y}$ which is defined as

$$\mathbf{y} = \left[\begin{array}{cccc} y_1 & y_2 & \ldots & y_{N_{BS}} \end{array}\right]^T \tag{7}$$

where the values y_j are equal to the measured RSS values from jth BS or equal to NaN when there is no value measured (no detection). The localization can then be done by defining distance measures between the measurement vector $\mathbf{y}$ and the map RSS-vectors $\hat{\mathbf{h}}^i$. In this study, we will denote such measures in the form of *likelihoods* $p(\mathbf{y}|\hat{\mathbf{h}}^i)$ of the measurement vector $\mathbf{y}$ given the RSS vector $\hat{\mathbf{h}}^i$ which represents an *hypothesis* about the position of the target (i.e., p^i). Note that this notational selection makes sense in the case of dynamic localization where probabilistic arguments appear quite frequently. However, even in the static localization, use of such a symbol for the distance measures, in spite of the fact that there is no stochastic reasoning in their definition most of the time, emphasizes the similarity of the problems in both cases. How to define the likelihoods is not straightforward and forms the backbone of localization. Once they are defined, the localization procedure in fingerprinting can be posed mathematically as the maximum likelihood estimation problem given below.

$$\left[\begin{array}{c} \hat{x} \\ \hat{y} \end{array}\right] = p^{\hat{i}} \tag{8}$$

$$\hat{i} = \arg \max_{1 \le i \le N_{LPM}} p(\mathbf{y}|\hat{\mathbf{h}}^i) \tag{9}$$

where $\hat{x}$ and $\hat{y}$ are the estimated x and y-coordinates of the target.

III. LIKELIHOOD DEFINITIONS FOR STATIC ESTIMATION

In defining the likelihoods used for classical (static) fingerprinting (given in (8) and (9)), if the vectors $\mathbf{y}$ and $\hat{\mathbf{h}}^i$ did not have NaN values, than any norm (or norm-like functions) would do the

job. The same would be true in the case where the places of NaN values and the non-NaN values would match in the two vectors. However, it is quite unlikely that this condition is satisfied in any real application. The classical way of defining the likelihood function is as given in the following algorithm, [1, 12].

   Algorithm 1: Classical Fingerprinting

Ignore the NaN values and compute the likelihood as the distance between the two (sub-)vectors, i.e.,

$$p(\mathbf{y}|\hat{\mathbf{h}}^i) \triangleq \|\Gamma^i\|^{-1} \tag{10}$$

where $\Gamma^i \triangleq [\gamma_1^i, \gamma_2^i, \ldots, \gamma_{N_{BS}}^i]^T$ is the vector whose elements are defined as

$$\gamma_j^i \triangleq \begin{cases} y_j - \hat{h}_j^i, & y_j \neq \text{NaN}, \quad \hat{h}_j^i \neq \text{NaN} \\ 0, & \text{otherwise} \end{cases}. \tag{11}$$

The norm $\| \cdot \|$, although its effects, most of the time, might be negligible, can be selected to be any valid norm or distance. In our paper, for the comparisons, the standard $\| \cdot \|_2$ norm is used.

   On the other hand, the non-matching NaN values, as is going to be shown in this paper, carry valuable information that should not be neglected in the localization. The information given by them can be summarized for two different cases as given below:

- When the measurement vector $\mathbf{y}$ has NaN value for some BS (this means that the receiver did not get any RSS measurement from that BS) the hypotheses $\hat{\mathbf{h}}^i$ that have a value for that BS are unlikely. In other words, positions p^i that are far from the BS are more likely.

- When the measurement vector has a value for some BS (this means that the receiver has got a RSS measurement from that BS) the hypotheses $\hat{\mathbf{h}}^i$ that does not have a value for that BS (these are the RSS vectors $\hat{\mathbf{h}}^i$ that have a NaN value for that BS) are unlikely, i.e., the positions p^i that are close to the BS are more likely.

Usage of this, in a way, negative information in localization to different extents is the main theme of our paper. The localization hypotheses $\hat{\mathbf{h}}^i$ having non-matching NaN values which we call as *non-matching hypotheses* are punished by our proposed methods. Two different likelihood calculation mechanisms (hence measurement models) are proposed for the static and dynamic estimation cases respectively. The static estimation case involves no assumption of temporal correlation of the estimated position values and therefore requires the full extent of the

punishment of the non-matching hypotheses. Consequently, we call the likelihood calculation mechanism proposed for this case as the *BS-strict* approach. The dynamic estimation case, on the other hand, makes use of a dynamic motion model for the estimated position values, which enables the positioning algorithm to accumulate information from consecutive measurements. This requires a softer version of the BS-strict approach in the sense that it allows for the survival of the unlikely hypotheses between consecutive times. Hence, we call the proposed algorithm for this dynamical case as the *BS-soft* approach.

We delay the stochastic derivation of BS-soft approach to Section V and give below the BS-strict approach which is going to be used in the static estimation in Section IV.

Algorithm 2: BS-strict

This approach calculates the likelihoods in the same way as Algorithm 1, but this time the elements γ_j^i of the vector Γ^i are defined as

$$\gamma_j^i \triangleq \begin{cases} y_j - \hat{h}_j^i, & y_j \neq \text{NaN}, \quad \hat{h}_j^i \neq \text{NaN} \\ 0, & y_j = \text{NaN}, \quad \hat{h}_j^i = \text{NaN} \\ \infty, & \text{otherwise} \end{cases} \quad . \tag{12}$$

Notice that the infinite punishment given to the non-matching NaN values in Algorithm 2 results in the elimination of the corresponding hypotheses because their likelihood will vanish. Any likelihood based method using Algorithm 2, therefore, will search for the *strict* match of the NaN and non-NaN values in the two compared RSS vectors. This methodology will then increase the effects of the BS identities in the estimation process. The methods based on this algorithm can be more robust than the ones using the classical algorithm which relies only on the measured RSS values. This is because the measured BS identities are much more reliable than the actual measured RSS values under a significant range of effects like weather, NLOS and fading.

IV. FINGERPRINTING LOCALIZATION: THE STATIC CASE

For our study, the power maps of all available sites in the measurement area shown in Figure 1 have been generated and plotted in Figure 2. In the following sub-sections, fingerprinting as defined in (8) and (9) is applied to RSSI and SCORE values where the likelihoods $p(\mathbf{y}|\hat{\mathbf{h}}^i)$ involved are calculated by either the classical method or BS-strict approach defined in Section III.
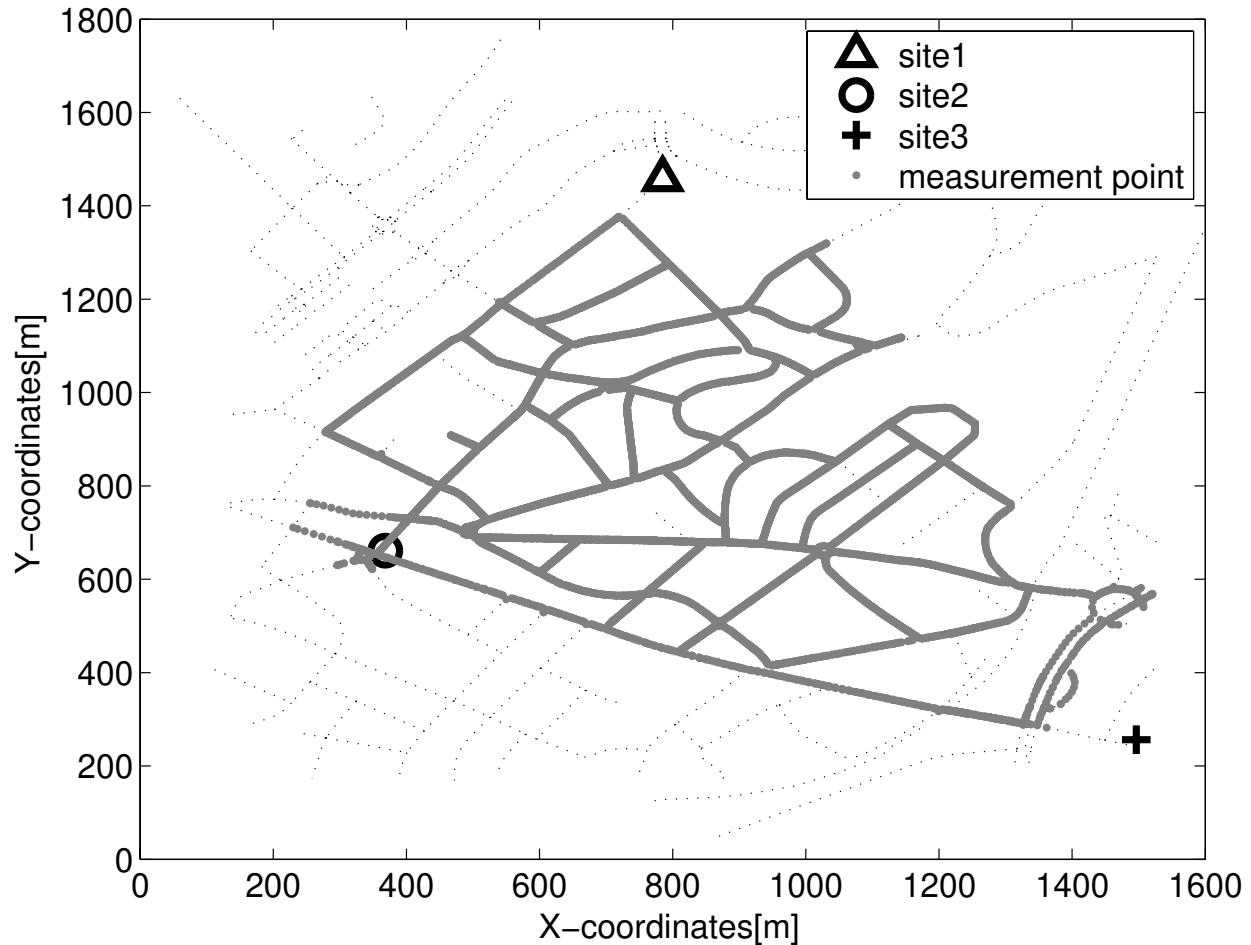
Fig. 1: The area under study (the measurement area). The average distance between two sites is about 1150 m.

A. Fingerprinting using RSSI values

In this section we suppose that the user can measure accurately (the same accuracy as the power map) the received power (RSSI values). This can be done (and has been done for this paper) using special calibrated modems with extra software installed and the measurements have to be collected off-line, because only one channel can be measured at a time. Currently, it is not practical to use such modems in applications but the purpose of using them in this study is to check the possible achievable accuracy in case the user can make such measurements. The validation data set was obtained using the trajectory shown in Figure 3 and used to position a user. The two mentioned approaches were applied: the classical fingerprinting (Algorithm 1)
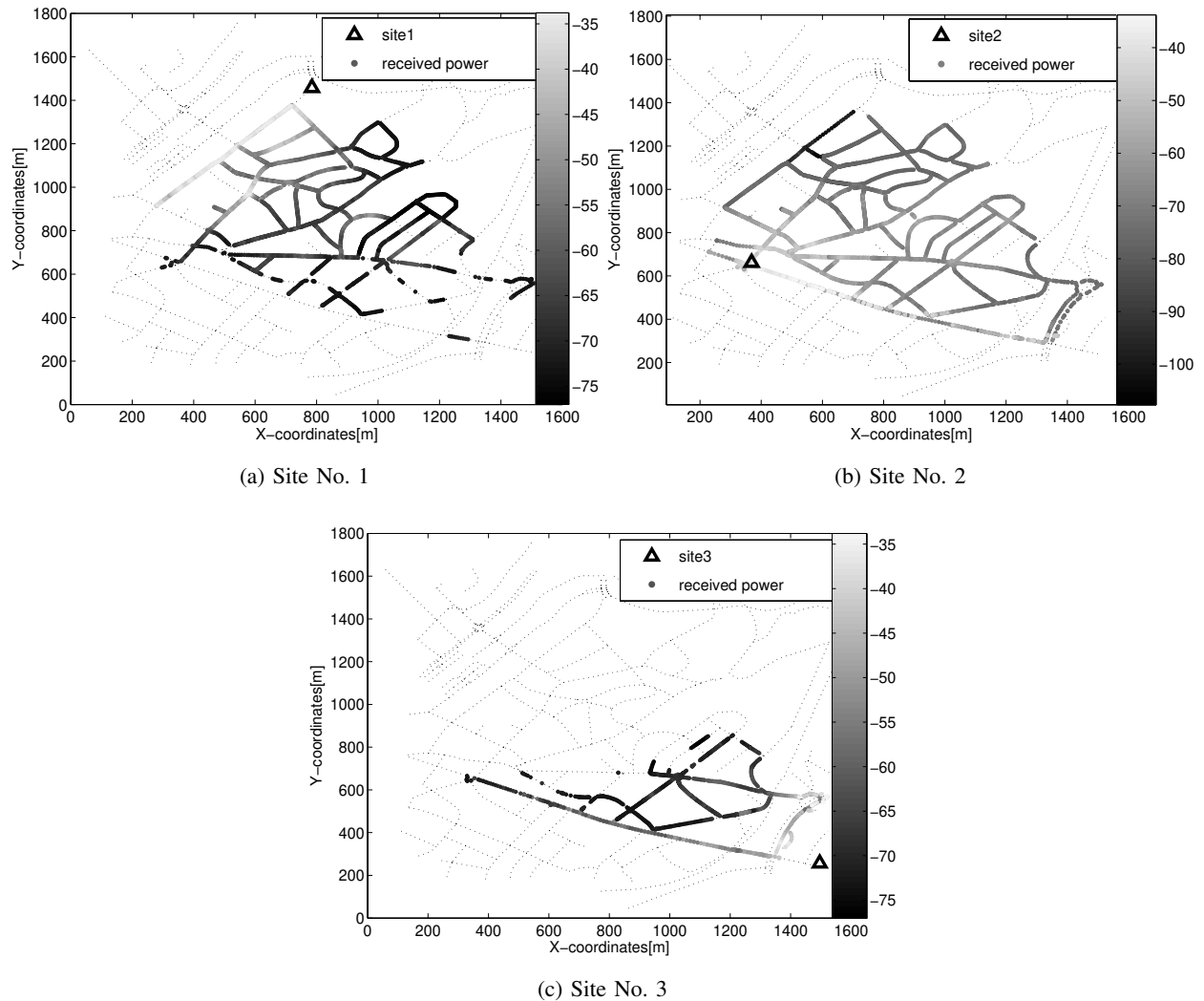
(a) Site No. 1

(b) Site No. 2

(c) Site No. 3

Fig. 2: The power maps of the three WiMAX sites.

and the BS-strict fingerprinting (Algorithm 2). The results are shown in Figure 4. The strict BS fingerprinting approach's performance was significantly better than the classical one and this is due to the fact that the BS number is more robust against the noise than RSS values, i.e., the same BS number will be obtained regardless of the presence of a strong noise or not, but different RSS values will be collected.
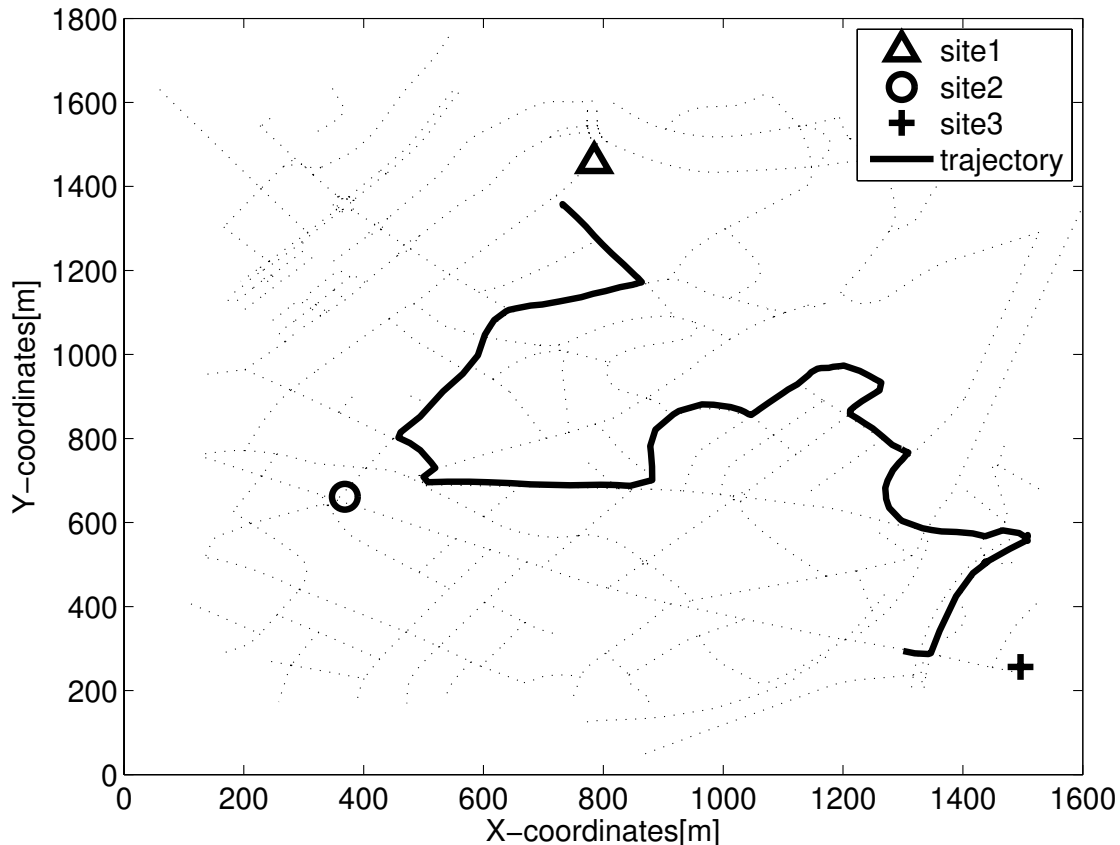
Fig. 3: The used target trajectory.

B. Fingerprinting using SCORE values

The SCORE values are used by the standard WiMAX modems to evaluate the connection quality between the subscriber station (SS) and the available BSs and they can be collected without adding any extra software or hardware to the modem. The advantage of using the SCORE values is the possibility to obtain them for all the available BSs simultaneously, but the disadvantage lies in their low accuracy compared to RSSI values. The relation between SCORE values and RSSI values is given, according to the information provided by modem manufacturer, by (13):

$$SCORE = (RSSI - 22) - (0.08 \times AvgViterbi). \tag{13}$$

where, AvgViterbi value is computed statistically from the Viterbi decoder. This adds an extra challenge for localization services, since even though the performance of the decoder is important

for handover decisions, it is only nuisance for localization. Measurements were collected using the same trajectory to validate the two fingerprinting approaches (using the same database built using RSSI values). Figure 4 shows the cumulative distribution function (cdf) of the positioning error. Two observations can be made:

1) Using the SCORE values gives less positioning accuracy than using the RSSI values. This is logical because the SCORE values are less accurate than RSSI values.

2) The impact of using the BS-strict approach is larger in the case of SCORE values. The SCORE values are subject to bigger changes than the RSSI values because SCORE values do not depend only on the received power but also on the quality of the signal determined by the Viterbi decoder.
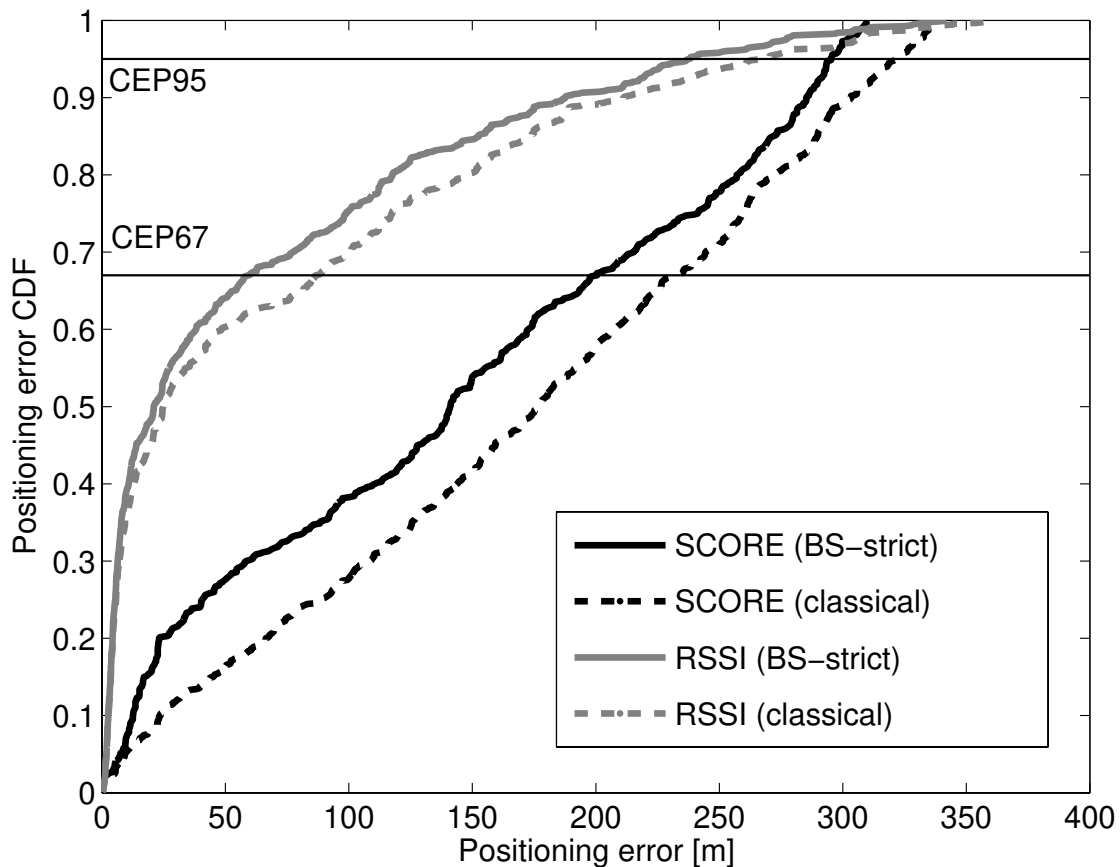

Fig. 4: The positioning error CDFs. The two fingerprinting approaches were used (the classical and the BS-strict) with the available measurements (RSSI and SCORE).

V. LIKELIHOOD DEFINITIONS FOR DYNAMIC ESTIMATION

In static estimation, there is no temporal correlation between the consecutively made estimations. In other words, once a measurement $\mathbf{y}_{t_k}$ is collected at time t_k and an estimate $\hat{x}_k^p$ of the target position is obtained, in the next time step t_{k+1}, the whole procedure is repeated by using only $\mathbf{y}_{t_{k+1}}$ and the new estimate $\hat{x}_{k+1}^p$ is independent of what $\hat{x}_k^p$ is. In such a case, the use of the information stored in the measurement $\mathbf{y}_{t_k}$ to its full extent is reasonable because by doing this,

- we extract most out of a single measurement.
- even if we make a mistake in the current estimation, the estimation errors cannot accumulate and affect the subsequent estimations.

Consequently, the negative information (i.e., non-detection events or NaN values) in the measurements $\mathbf{y}$ have been used to eliminate some positioning hypotheses completely in Algorithm 2.

On the other hand, the dynamical estimation methods, which use models to take advantage of the correlated information in consecutive position estimations, get their power from the accumulation of the information in the algorithm along the time. Therefore, the survival of different hypotheses about the position values is important in such methods for information gathering process which enables higher estimation performance. Moreover, the complete elimination of some hypotheses (like the assignment of infinite cost to non-matching hypotheses in Algorithm 2) can result in error accumulation in a recursive procedure because an hypothesis deletion can never be compensated in the future even if some contrasting evidence appears. Thus, assigning still higher but finite costs to non-matching hypotheses and hence allowing them (or some of them) to survive is more suitable in dynamic estimation procedures. Since such a cost assignment procedure make the hypothesis punishment in a softer way than Algorithm 2 by assigning finite costs to non-matching hypotheses (compared to the infinite punishment in Algorithm 2 which results in the hypothesis elimination), we call the resulting methodology as the "soft" approach. Below we give such a soft likelihood calculation mechanism to be used in a dynamic estimation method. The algorithm that we will present is based on the following simple assumptions:

1) The elements $\{y_j\}_{j=1}^{N_{BS}}$ of the measurement vector $\mathbf{y}$ are conditionally independent given the database RSS vector $\hat{\mathbf{h}}^i$.

2) Matching not-NaN values in the measurement and RSS values satisfy

$$y_j = \hat{h}_j^i + e_j \quad \text{if } y_j \neq \text{NaN and } \hat{h}_j^i \neq \text{NaN} \tag{14}$$

where $e_j \sim p_{e_j}(.)$ represents measurement noise for the jth BS.

Using the first assumption, the likelihood $p(\mathbf{y}|\hat{\mathbf{h}}^i)$ can be written as

$$p(\mathbf{y}|\hat{\mathbf{h}}^i) = \prod_{j=1}^{N_{BS}} \beta_{ij} \tag{15}$$

where $\beta_{ij} \triangleq p(y_j|\hat{h}_j^i)$ is the individual likelihood for the jth BS. The different combinations that appear in the analysis due to NaN values are considered separately below:

- If $y_j \neq \text{NaN}$ (we get a measurement from the jth BS) and $\hat{h}_j^i \neq \text{NaN}$ (the ith hypothesis has an LPM data for jth BS), we have by assumption 2

$$\beta_{ij} = p_{e_j}(y_j - \hat{h}_j^i) \tag{16}$$

where $p_{e_j}(.)$ can be selected considering the application requirements. A simple choice is to set

$$\beta_{ij} = \mathcal{N}(y_j; \hat{h}_j^i, \sigma_j^2) \tag{17}$$

where $\mathcal{N}(y_j; \hat{h}_j^i, \sigma_j^2)$ denotes a normal density with mean $\hat{h}_j^i$ and standard deviation σ_j evaluated at y_j. This corresponds to $p_{e_j}(.) = \mathcal{N}(.; 0, \sigma_j^2)$. If the number of data points averaged for an LPM grid point are greater than e.g. 10, then, by the central limit theorem, this Gaussian likelihood seems to be the most appropriate selection. The standard deviation σ_j is a user selected parameter that could change from base-station to base-station.

- If $y_j = \text{NaN}$ (we do not get any measurement from the jth BS) and $\hat{h}_j^i \neq \text{NaN}$, then we have

$$\beta_{ij} = P(y_j < y_{min}|\hat{h}_j^i) \tag{18}$$

$$= P(e_j < y_{min} - \hat{h}_j^i) \tag{19}$$

$$= cdf_{e_j}(y_{min} - \hat{h}_j^i) \tag{20}$$

where $cdf_{e_j}(x) \triangleq \int_{-\infty}^{x} p_{e_j}(x)dx$ is the cumulative distribution function of e_j. Here, while passing from (19) to (20), we assumed that e_j is a continuous random variable (i.e., no discontinuity in its cumulative distribution function). The probability density function

appears in the calculation again as a design parameter. Notice here that, although it is the same density that is required in the previous case, the density $p_{e_j}(.)$ can be selected differently for each case for design purposes. In fact, as observed from several preliminary experiments, the Gaussian selection as in the previous case gives too much (exponential) punishing for the non-matching hypotheses (i.e., hypotheses corresponding to $\hat{\mathbf{h}}^i$ for which $\hat{h}_j^i \neq \text{NaN}$). Such a selection would therefore yield a hard approach which is similar to BS-strict algorithm. Therefore, another selection has been made which leads to the softer result

$$\beta_{ij} = \mu \left| \frac{\hat{h}_j^i}{y_{min}} \right|, \tag{21}$$

where $\mu \leq 1$ is a constant design parameter. This selection, in fact, corresponds to a uniform density for e_j between the values y_{min} and $-y_{min}$ when $\mu = 0.5/y_{min}$. Notice that we always have $y_{min} < \hat{h}_j^i \leq 0$ in our paper [1] and therefore, $0 \leq \beta_{ij} < 1$. Since we do not get a measurement from the jth BS, we punish the hypotheses that have LPM values for that BS and note that the larger the LPM value (i.e., power) the more the punishment is, i.e., β_{ij} is smaller.

- If $y_j \neq \text{NaN}$ and $\hat{h}_j^i = \text{NaN}$ (for the ith hypothesis, we do not have any data for the jth BS), then a similar analysis would be

$$\beta_{ij} = p(y_j | \hat{h}_j^i < y_{min}) \tag{22}$$

$$= \frac{P(\hat{h}_j^i < y_{min} | y_j) p(y_j)}{P(\hat{h}_j^i < y_{min})} \tag{23}$$

which requires the prior likelihood $p(y_j)$ and probability $P(\hat{h}_j^i < y_{min})$ which are hard to obtain. A straightforward approximation can be

$$\beta_{ij} \approx P(\hat{h}_j^i < y_{min} | y_j) \tag{24}$$

which is simple to calculate in a way similar to (21) but has been seen to be giving low performance in preliminary simulations. The reason for this has been investigated and is

---

[1]In fact, the data collected for our paper (i.e., $\{\hat{h}_j^i\}_{i=1}^{N_{LPM}}$ for $j = 1, \ldots, N_{BS}$) satisfied this assumption but in general, the collected data need not satisfy it. This is, however, not a restriction because one can always find the quantity $\bar{h} \triangleq \max_{1 \leq j \leq N_{BS}} \max_{1 \leq i \leq N_{LPM}} \hat{h}_j^i$ and subtract it from all the data and the online measurements when they are collected to obtain an equivalent data and measurements that satisfy the assumption for a value of y_{min}.

found to be that the term calculated using (24) can sometimes be much larger than the terms calculated for the hypotheses that actually have a (non-NaN) value for that BS. We are going to illustrate our argument on the following example case: Suppose that $y_j \neq$ NaN (i.e., we have collected a measurement from the jth BS) and i_1 and i_2 are two positioning hypotheses such that $\hat{h}_\ell^{i_1} = \hat{h}_\ell^{i_2}$ for $\ell \neq j$. Suppose also that $\hat{h}_j^{i_1} = $ NaN (for the i_1th hypothesis, we do not have any data for the jth BS) and $\hat{h}_j^{i_2} \neq$ NaN (for the i_2th hypothesis, we have a data for the jth BS). We would like to calculate the punishing terms (likelihoods) $\beta_{i_1 j}$ and $\beta_{i_2 j}$ corresponding to these two hypotheses. Since the hypothesis i_1 is non-matching (in terms of only the jth BS, i.e., $y_j \neq \hat{h}_j^{i_1}$) and the hypothesis i_1 is matching (in terms of only the jth BS, i.e., $y_j = \hat{h}_j^{i_1}$), we expect that the punishment for i_1 to be more than the one for i_2, that is, the inequality

$$\beta_{i_1 j} \leq \beta_{i_2 j} \tag{25}$$

must be satisfied. Note that, since $\beta_{i_2 j}$ depends on $\hat{h}_j^{i_2}$ and y_j via (17), it can be arbitrarily small. Therefore, if $\beta_{i_1 j}$ is selected irrespective of the $\beta_{i_2 j}$ values for the matching hypotheses, it is a strong possibility that $\beta_{i_1 j}$ would happen to be much higher than $\beta_{i_2 j}$ and hence a non-matching hypothesis will be promoted instead of the matching ones. In fact, in the preliminary simulations using (24), this caused the matching hypotheses to be discarded. Therefore, for this case, we (give up the formula (24) and) propose the following likelihood calculation method

$$\beta_{ij} = \min_{m \in \mathcal{M}_j} \beta_{mj} \tag{26}$$

where the set $\mathcal{M}_j$ is given as

$$\mathcal{M}_j \triangleq \{i | \hat{h}_j^i \neq \text{NaN}\}. \tag{27}$$

The likelihood (26) always satisfies the condition (25) and hence the non-matching hypotheses are punished more than or equal to the matching ones. One can actually replace the punishment factor with any smaller value. Notice that when there are no hypotheses which have values for the BS (i.e., the set $\mathcal{M}_j$ is empty), arbitrary punishing or (24) can be applied.

- If $y_j = $ NaN and $\hat{h}_j^i = $ NaN, then, since the vectors are matching for jth BS, one can set $\beta_{ij} = 1$.

The algorithm outlined above is summarized in the following from an implementation point of view:

Algorithm 3: BS-soft

Suppose the current available hypotheses are shown as $\{\hat{\mathbf{h}}^i\}_{i=1}^{N_h}$ where N_h represents the number of hypotheses.

- Calculate the quantities α_{ij} for $i = 1, \ldots, N_h$ and $j = 1, \ldots, N_{BS}$ as

$$\alpha_{ij} = \begin{cases} \mathcal{N}(y_j; \hat{h}_j^i, \sigma_j^2), & y_j \neq \text{NaN}, \hat{h}_j^i \neq \text{NaN} \\ \text{NaN}, & \text{otherwise} \end{cases}. \tag{28}$$

- Calculate the quantities β_{ij} for $i = 1, \ldots, N_h$ and $j = 1, \ldots, N_{BS}$ using $\{\alpha_{ij}\}$ as

$$\beta_{ij} = \begin{cases} 1, & y_j = \text{NaN}, \hat{h}_j^i = \text{NaN} \\ \mu \left| \dfrac{\hat{h}_j^i}{y_{min}} \right|, & y_j = \text{NaN}, \hat{h}_j^i \neq \text{NaN} \\ \min_m \alpha_{mj}, & y_j \neq \text{NaN}, \hat{h}_j^i = \text{NaN} \\ \alpha_{ij}, & y_j \neq \text{NaN}, \hat{h}_j^i \neq \text{NaN} \end{cases} \tag{29}$$

where only numeric values are considered in the minimization.

- Calculate the likelihoods $\{p(\mathbf{y}|\hat{\mathbf{h}}^i)\}_{i=1}^{N_h}$ from $\{\beta_{ij}\}$ as

$$p(\mathbf{y}|\hat{\mathbf{h}}^i) = \prod_{j=1}^{N_{BS}} \beta_{ij} \tag{30}$$

for $i = 1, \ldots, N_h$.

The punishment terms in the likelihood calculation can be thought of as a softened version of the BS-strict approach considered previously in this section. In a way, by assigning lower weights to hypotheses that do not match the measurement, one lowers their effect in the overall estimate instead of discarding them completely (similar to BS-strict) which can be quite harmful in dynamic approaches.

VI. FINGERPRINTING LOCALIZATION: THE DYNAMIC CASE

For the positioning methods used in Section IV, one does not consider the time information (stamps) available with the measurements. When the target is localized with good accuracy for one measurement, in the next measurement when the user is possibly quite close to the previous location (because only a small amount of time has passed), the previous accurate localization

is completely discarded and a new localization is done based on the new measurement. This is one type of static target localization and the dynamic information coming from the fact that the user does not move much between consecutive measurements is not used. One of the ways to use this extra information in localization is to use a dynamic model for the target (user) position given as:

$$\mathbf{x}_{t_{k+1}} = f_{t_{k+1},t_k}(\mathbf{x}_{t_k}, w_{t_{k+1},t_k}) \tag{31}$$

where

- $\mathbf{x}_{t_k} \in \mathbb{R}^{n_x}$ is the state of the target at time t_k,
- $w_{t_{k+1},t_k} \in \mathbb{R}^{n_w}$ is the process noise representing the uncertainty in the model between time instants t_k and t_{k+1}. If the process noise term is selected to be small, this means that the target model is known with good accuracy and vice versa.
- $f_{t_{k+1},t_k}(.,.)$ is in general a nonlinear function of its arguments.

This type of models is generally used in target tracking [22, 23] to model target motion dynamics. At each time instant t_k, we get a measurement $\mathbf{y}_{t_k}$ which is related to the state of the target as

$$\mathbf{y}_{t_k} = h(\mathbf{x}_{t_k}) + v_{t_k} \tag{32}$$

where

- $h(.)$ is, in general, a nonlinear function. In our application, it is the power map whose information is collected off-line. The likelihoods $p(\mathbf{y}_{t_k}|\mathbf{x}_{t_k})$ will be formed from the power map using Algorithm 3. The details will be given below in sub-section VI-B2.
- v_{t_k} is the measurement noise representing the quality of our sensors.

The state estimation with this type of probabilistic model, given by (31) and (32), is a mature area of research [24, 25]. The optimal solution when the functions $f(.)$ and $h(.)$ are linear and the noise terms w_{t_{k+1},t_k} and v_{t_k} are Gaussian is the well-known Kalman filter [26]. Some small nonlinearities can be handled by approximate methods such as the extended Kalman filter (EKF) [27] and the methods called as sigma-point Kalman filters [28], of which the unscented Kalman filter (UKF) [29, 30] is one type, have been shown to be suitable for a much larger class of nonlinearities (see the extensive work [31]). These approaches are possible alternatives in the cases where the posterior density of the state is unimodal. On the other hand, if one assumes that the user is moving on the road, the state density would be highly multi-modal,

which can be approximated with a single Gaussian distribution quite poorly. Complicating the facts, the measurement function $h(.)$ that is represented by the power map is highly nonlinear and furthermore it is discontinuous. Therefore, in this work we are going to use the relatively recent algorithms in the literature called particle filters [14, 15, 16]. Two particle filters are used to track the target (user). The first one exploits the target dynamic information (motion model) only, and the second filter makes use of the public road information map in addition to the dynamic information. We call these filters *off-road* and *on-road* particle filters for obvious reasons. Knowing that the user is on the public road network is a valuable information to position him/her. The *TeleAtlas* maps have been used as assisting data in addition to the measured data [32].

A. Particle Filter

Particle filters are recursive implementation of Bayesian density recursions [14, 15, 16]. The main aim in the method, as in many Bayesian methods, is to calculate the posterior density of the state $\mathbf{x}_{t_k}$ given all the measurements $\mathbf{y}_{t_{1:k}} \triangleq \{\mathbf{y}_{t_1}, \mathbf{y}_{t_2}, \ldots, \mathbf{y}_{t_k}\}$; i.e. we calculate the density $p(\mathbf{x}_{t_k}|\mathbf{y}_{t_{1:k}})$. While doing this, the particle filter approximates the density $p(\mathbf{x}_{t_k}|\mathbf{y}_{t_{1:k}})$ with a number of state values $\{\mathbf{x}_{t_k}^{(i)}\}_{i=1}^{N_p}$ (called particles) and their corresponding weights $\{\eta_{t_k}^{(i)}\}_{i=1}^{N_p}$ (called particle weights) i.e.,

$$p(\mathbf{x}_{t_k}|\mathbf{y}_{t_{1:k}}) \approx \sum_{i=1}^{N_p} \eta_{t_k}^{(i)} \delta_{\mathbf{x}_{t_k}^{(i)}}(\mathbf{x}_{t_k}) \tag{33}$$

Then, at each time step, the particle filter needs to calculate the particles and weights $\{\mathbf{x}_{t_k}^{(i)}, \eta_{t_k}^{(i)}\}_{i=1}^{N_p}$ from the previous particles and weights $\{\mathbf{x}_{t_{k-1}}^{(i)}, \eta_{t_{k-1}}^{(i)}\}_{i=1}^{N_p}$. We are going to use the basic particle filtering algorithm which is called *bootstrap filter* and which was proposed first in [33]. At each step of the algorithm, one can calculate the conditional estimate $\hat{\mathbf{x}}_{t_k}$ and the covariance P_{t_k} of the state as

$$\hat{\mathbf{x}}_{t_k} \triangleq \sum_{i=1}^{N_p} \eta_{t_k}^{(i)} \mathbf{x}_{t_k}^{(i)}, \tag{34}$$

$$P_{t_k} \triangleq \sum_{i=1}^{N_p} \eta_{t_k}^{(i)} \left[\mathbf{x}_{t_k}^{(i)} - \hat{\mathbf{x}}_{t_k} \right] \left[\mathbf{x}_{t_k}^{(i)} - \hat{\mathbf{x}}_{t_k} \right]^T. \tag{35}$$

It is possible to calculate other types of point estimates like maximum a posteriori (MAP) estimates etc. [34] from the particles and the weights of the posterior state density, however, this

would require a kernel smoothing of the particles in general [35]. Note that the particle filter described above is one of the simplest and computationally cheapest algorithms among more complicated ones as given in [36] and in [14]. In the following subsection, we will describe the specific models and parameters that are used in the two different (off-road and on-road) implemented particle filters.

B. Implementation Details of the Particle Filters

We implemented two different bootstrap particle filters using different target motion models but with the same measurement model (i.e., likelihood).

1) State Models: The first particle filter (called off-road filter) uses a classical (nearly) constant velocity model with state $\mathbf{x}_k = \left[p_{k+1}^x, p_{k+1}^y, v_{k+1}^x, v_{k+1}^y \right]^T$ where variables p and v denote the position and velocity of the the target respectively. The motion model is given

$$
\begin{bmatrix} p_{k+1}^x \\ p_{k+1}^y \\ v_{k+1}^x \\ v_{k+1}^y \end{bmatrix} = \begin{bmatrix} \mathbf{I}_2 & T_{k+1}\mathbf{I}_2 \\ \mathbf{0} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} p_k^x \\ p_k^y \\ v_k^x \\ v_k^y \end{bmatrix} + \begin{bmatrix} \frac{T_{k+1}^2}{2}\mathbf{I}_2 \\ T_{k+1}\mathbf{I}_2 \end{bmatrix} w_{k+1} \tag{36}
$$

where w_k is a two dimensional white Gaussian noise with zero mean and covariance $5^2\mathbf{I}_2$ and $\mathbf{I}_n$ is the identity matrix of dimension n. $T_{k+1} = t_{k+1} - t_k$ is the difference between consecutive time stamps of the measurements.

The second particle filter (called on-road filter) makes use of the road database information. The literature is abundant with a large number of publications on target tracking with road network information. Although the early studies used multiple model (extended) Kalman filter based approaches [37, 38, 39], the particle filters, in a short time, have proved to be one of the indispensable tools in road constrained estimation [40, 41]. This is confirmed in the large number of publications on the subject like [42, 43, 44, 45, 46, 47], which appeared only in the last five years. Our approach here considers a single reduced-order on-road motion model with a bootstrap filter. The state of the particle filter is denoted by $\mathbf{x}_k^r$ where r stands for emphasizing road-information, and it is given as $\mathbf{x}_k^r = [p_k^r, v_k^r, i_k^r]^T$ where the scalar variables p_k^r, v_k^r denote the position and speed values of the target on the road segment, which is identified by the integer

index i_k^r. The following model is used for the dynamics of $\mathbf{x}_k^r$.

$$
\begin{bmatrix} p_{k+1}^r \\ v_{k+1}^r \\ i_{k+1}^r \end{bmatrix} = f^r \left(\begin{bmatrix} \mathsf{p}_{k+1}^r \\ \mathsf{v}_{k+1}^r \\ i_k^r \end{bmatrix}, \mathcal{I}_{RN}, w_{k+1}^{r_d} \right) \tag{37}
$$

where

$$
\begin{bmatrix} \mathsf{p}_{k+1}^r \\ \mathsf{v}_{k+1}^r \end{bmatrix} = \begin{bmatrix} 1 & T_{k+1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_k^r \\ v_k^r \end{bmatrix} + \begin{bmatrix} \frac{T_{k+1}^2}{2} \\ T_{k+1} \end{bmatrix} w_{k+1}^{r_c}. \tag{38}
$$

The continuous process noise $w_k^{r_c}$ is a scalar white Gaussian acceleration noise with zero mean and $0.2 m/s^2$ standard deviation. The predicted position and speed values p_{k+1}^r, v_{k+1}^r might not be on the road segment indicated by i_k^r. The function $f^r(.)$, therefore, projects the values p_{k+1}^r, v_{k+1}^r into the road segment denoted by i_{k+1}^r. If there is more than one candidate for the next road segment index i_{k+1}^r due to the junctions, the function also selects a random one according to the value of the discrete on-road process noise term $w_{k+1}^{r_d} \in \{1, 2, \ldots, N_r(\mathbf{x}_k^r)\}$ where $N_r(\mathbf{x}_k^r)$ is the number of possible road segments that the target with on-road state $\mathbf{x}_k^r$ might go in the following T_{k+1} seconds.
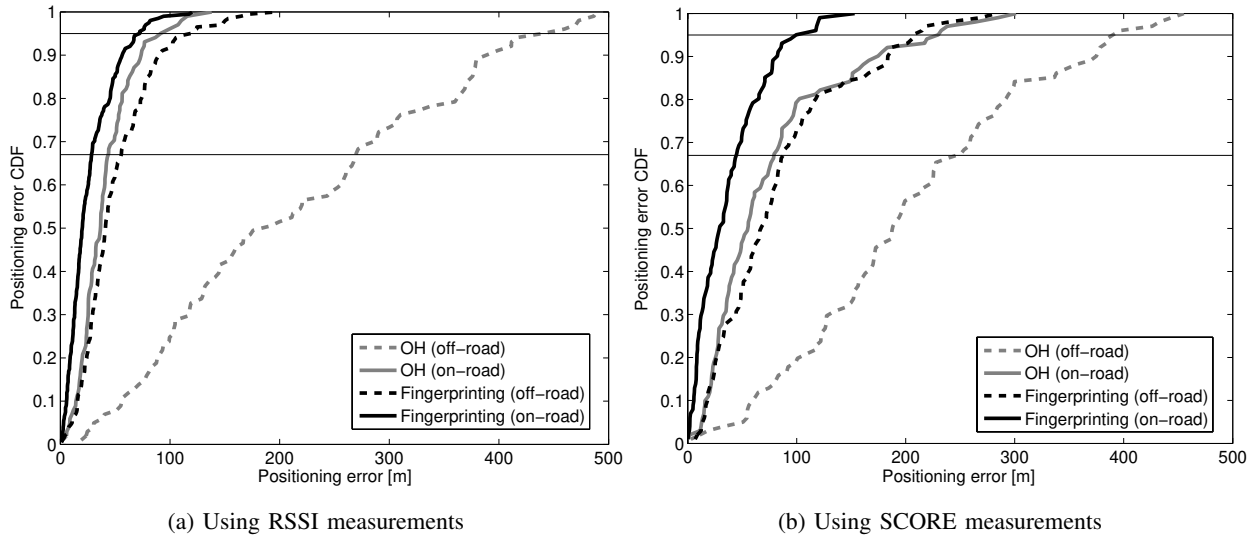

(a) Using RSSI measurements   (b) Using SCORE measurements

Fig. 5: Positioning error CDFs for the proposed fingerprinting approach and the conventional approach (based on the OH-model).

(a) Using RSSI measurements
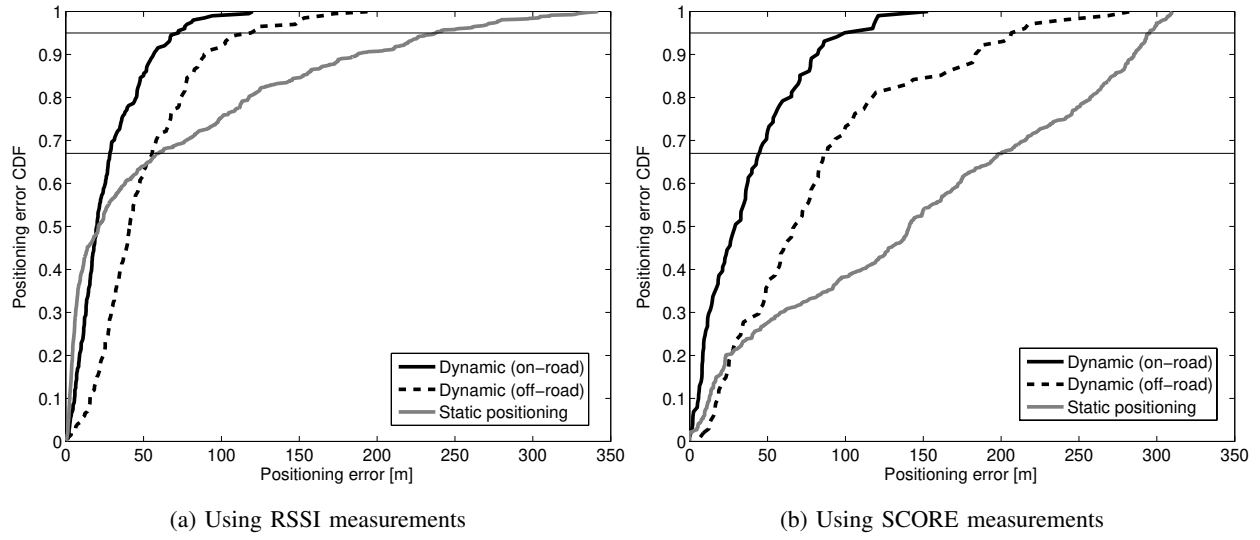
(b) Using SCORE measurements

Fig. 6: Positioning error comparison between dynamic positioning (on-road and off-road) and static positioning.

2) Likelihoods: The measurement model is the same for both particle filters. At a single time instant t_k, the measurement vector is in the following form

$$\mathbf{y}_{t_k} = \begin{bmatrix} y_1 & y_2 & \cdots & y_{N_{BS}} \end{bmatrix}^T \tag{39}$$

as has also been given in Section II. The likelihood value $p(\mathbf{y}_{t_k}|\mathbf{x}_{t_k}^{(i)})$ is calculated using the LPM as given in the following algorithm:

Algorithm 4: Calculation of $p(\mathbf{y}_{t_k}|\mathbf{x}_{t_k}^{(i)})$

- Calculate the distance of the particle to all of the LPM grid points as

$$d_j = \|p_{t_k}^{(i)} - p^j\|_2 \tag{40}$$

  where $p_{t_k}^{(i)}$ denotes the vector composed of the position components of $\mathbf{x}_{t_k}^{(i)}$.

- Find the closest point in the LPM to the particle position as

$$\hat{j} = \arg \max_{1 < j < N_{LPM}} d_j. \tag{41}$$

- Calculate $p(\mathbf{y}_{t_k}|\mathbf{x}_{t_k}^{(i)})$ as

$$p(\mathbf{y}_{t_k}|\mathbf{x}_{t_k}^{(i)}) = \begin{cases} p(\mathbf{y}_{t_k}|\hat{\mathbf{h}}^{\hat{j}}), & \text{if } d_{\hat{j}} \leq d_{threshold} \\ p(\mathbf{y}_{t_k}|\bar{\mathbf{h}}), & \text{otherwise} \end{cases} \tag{42}$$

TABLE I: Parameter values used for Algorithm 3 for likelihood calculation.

Parameter	Name	Value
σ_j	Measurement Noise Covariance	7
μ	Scaling constant	1
y_{min}	Minimum detectable RSS	-100
N_h	Number of hypotheses	N_p

where $p(\mathbf{y}_{t_k}|\hat{\hat{\mathbf{h}}}^j)$ and $p(\mathbf{y}_{t_k}|\bar{\mathbf{h}})$ are calculated using Algorithm 3 whose specific parameters are given in Table I. In (42), $\bar{\mathbf{h}}$ denotes a N_{BS}-vector with all elements equal to NaNs. $d_{threshold}$ is a user selected distance threshold that determines the largest distance between a particle and an LPM grid point that the LPM grid point can be used to calculate the likelihood of the particle. This is going to be especially important in the offroad particle filter where the particles can frequently go outside of the area of interest. In this case, using $p(\mathbf{y}_{t_k}|\bar{\mathbf{h}})$ instead of $p(\mathbf{y}_{t_k}|\hat{\hat{\mathbf{h}}}^j)$ punishes such a particle implicitly. We selected $d_{threshold} = 100m$ in our simulations.

3) Initialization: Particle filters were initialized with a large Gaussian spread of particles with mean at the true positions and zero velocities; i.e.

$$\begin{bmatrix} p_0^{x,(i)} & p_0^{y,(i)} & v_0^{x,(i)} & v_0^{y,(i)} \end{bmatrix}^T \sim \mathcal{N}(.,m_0,P_0) \tag{43}$$

for $i = 1, \ldots, N_p$ where

$$m_0 \triangleq \begin{bmatrix} \bar{p}_0^x & \bar{p}_0^y & 0 & 0 \end{bmatrix}^T, \tag{44}$$

$$P_0 \triangleq \mathrm{diag}\left(\begin{bmatrix} 100^2 & 100^2 & 10^2 & 10^2 \end{bmatrix}\right). \tag{45}$$

Here, $[\bar{p}_0^x, \bar{p}_0^y]$ is the true target coordinates at time t_0 and the operator $\mathrm{diag}(.)$ forms a diagonal matrix from the elements of the input vector. The results that have been obtained in our study does not change with different initial distribution selection as long as the initial distribution covers the true target position with some probability mass. The initial Gaussian density given above has position standard deviation of 100m's which is in a way an indirect assumption of a prior information for the initial target position with that quality. It is unfortunately not possible to distribute the particles to the whole area of study initially and then start the estimation. This is because in such a case, the percentage of the probability mass that is spread around the

true position would be too small. Therefore, a suggestion for the general case, where no prior information of the initial target position is available, can be to initialize the particles around an initial estimate obtained by the static fingerprinting with the first collected measurement.

In the off-road particle filter, we use the initial particles directly. On the other hand, in the on-road particle filter, which always needs particles that are on the road network, the corresponding particles are obtained by projecting the ones defined above onto the road network.

In order to compare our fingerprinting based bootstrap filters, we have implemented two additional (onroad and offroad) bootstrap PFs that use only the OH-model in (3) for likelihood calculation. For this purpose, we have estimated transmitted powers (P_{BS}) and measurement variances for each BS and the path loss exponent α using the least squares method with our previously collected data (that has been used for forming the LPM). The estimation results for our fingerprinting based bootstrap filters and OH-model based bootstrap filters are shown in Figure 5. Notice that using SCORE or RSSI measurements with OH model in the offroad filter gives almost the same results because the dominating model errors like fading overcomes the effect of the accuracy of the measurements and the difference is no longer visible. In the onroad case the difference is more evident. The fingerprinting approach reduces the effect of modeling errors and therefore, the quality of the measurements gains more importance in the results. The performance of the fingerprinting methodology in dynamic filtering exceeds the OH-model based approach significantly. The performance gain with fingerprinting is overwhelming in the offroad case but still visible in the onroad filters, especially with the SCORE measurements where SNR is lower. It is remarkable that onroad OH-model based PF is almost equivalent to the offroad fingerprinting based filter in terms of estimation errors, which clearly illustrate the effect of strong modeling capability of the fingerprinting approach.

As a last point, we make a comparison between the results of the dynamic and the static cases, which are depicted in Figure 6. A very interesting observation is that in the high accuracy parts of the RSSI case, static approach makes better estimations than the dynamic ones although the dynamic estimation algorithms, in the overall results, are seen to be much more robust. Note that there is about 10m's performance loss in the RSSI based dynamic onroad filter compared to the static result. We attribute this difference to the fact that the static estimation calculates a maximum likelihood (ML) estimate whereas the dynamic onroad filter calculates a mean square estimate (MSE). Since there are about 10m's between the LPM grid points and since the particle

filter calculates the likelihood of a particle as the likelihood of the closest LPM point, there can appear many particles with the same weights in a 5m radius. Calculating the average of these particles which may be biased towards one side of the optimal result due to the road constraints can give an error of about 5m's. Considering the error terms added by averaging over all the particles, we can expect an error of about 10m's in the result compared to the ML based static approach which would directly give the position of most likely LPM grid point when SNR is high (like the case with RSSI). Calculation of the MAP estimate in the particle filter can be an alternative for this problem. In the offroad case, since the particles are even more separated we can expect this lower performance effect (under high SNR) to be more visible. Further, we think that the lack of the road network information also makes the estimates of the offroad filter suffer from low prior information compared to static estimates which are always constrained to the road segments. Note that with the SCORE measurements which represents a more practical low SNR case, there are no similar important sufferings. In the global behavior (95% lines), the performance gains with the dynamic approaches make it clear that these methods should be preferred when highly robust estimators are required. The results show that for 95% lines, the positioning accuracy improvement caused by the motion model comparing to the static case is about 33 % when SCORE values are used, and about 50% when RSSI values are used. The localization accuracy improvement achieved by using the road information comparing to the dynamic case is about 50% in case of SCORE values and about 40% in case of RSSI values, which indicates the strong effect of the road network information on the localization accuracy.

VII. CONCLUSION

This paper has discussed the use of fingerprinting positioning in wireless networks based on the RSS measurements: RSSI and SCORE with specific remarks on WiMAX networks. The introduced work has been divided into two main parts as static and dynamic localization. In the latter, the information of the target's motion model was used with and without road information. In both of the approaches, the effect of BS identities, which are more robust to propagation effects, on the estimates has been increased via designing specific likelihood calculation mechanisms. The results obtained show that fingerprinting positioning is a strong and robust approach to overcome the RSS's high variability. The positioning accuracy obtained by using the motion model and the road network information is notable. The accuracy improvement

was very promising and new location-dependent applications could be seen in the horizon. The positioning accuracy achieved by using fingerprinting positioning approach with motion model and road information can therefore be seen as a further step toward more accuracy demanding applications and new type of LBSs.

ACKNOWLEDGMENTS

REFERENCES

[1] Cello consortium report. [Online]. Available: http://www.telecom.ntua.gr/cello/documents/CELLO-WP2-VTT-D03-007-Int. pdf

[2] F. Gustafsson and F. Gunnarsson, "Possibilities and fundamental limitations of positioning using wireless communication networks measurements," *IEEE Signal Process. Mag.*, vol. 22, pp. 41–53, 2005.

[3] G. Sun, J. Chen, W. Guo, and K. Liu, "Signal processing techniques in network-aided positioning: A survey of state-of-the-art positioning designs," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 12–23, July 2005.

[4] S. Gezici, Z. Tian, B. Giannakis, H. Kobayashi, and A. Molisch, "Localization via ultra-wideband radios," *IEEE Signal Process. Mag.*, vol. 22, pp. 70–84, 2005.

[5] D. Li and Y. Hu, "Energy-based collaborative source localization using acoustic microsensor array," *Journal of Applied Signal Processing*, pp. 321–337, 2003.

[6] Y. Okumura, E. Ohmori, T. Kawano, and K. Fukuda, "Field strength and its variability in VHF and UHF land-mobile radio service," *Rev. Elec. Commun. Lab.*, vol. 16, pp. 9–10, 1968.

[7] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Trans. Veh. Technol.*, vol. 29, no. 3, pp. 317–325, Aug. 1980.

[8] J. Shirahama and T. Ohtsuki, "RSS-based localization in environments with different path loss exponent for each link," *Vehicular Technology Conference*, pp. 1509–1513, 2008.

[9] W. D. Wang and Q. X. Zhu, "RSS-based Monte Carlo localisation for mobile sensor networks," *IET Communications*, pp. 673–681, 2008.

[10] D.-B. Lin and R.-T. Juang, "Mobile location estimation based on differences of signal attenuations for GSM systems," *IEEE Trans. Veh. Technol.*, vol. 54, no. 4, pp. 1447–1454, Jul. 2005.

[11] O. Sallent, R. Agusi, and X. Cavlo, "A mobile location service demonstrator based on power measurements," *Vehicular Technology Conference*, vol. 6, no. 1, pp. 4096–4099, Sep. 2004.

[12] K. K. C. Takenga, "Mobile positioning based on pattern-matching and tracking techniques," *ISAST Transactions on Communications and Netwotking*, vol. 1, no. 1, pp. 529–532, Aug. 2007.

[13] A. Taok, N. Kandil, S. Affes, and S. Georges, "Fingerprinting localization using ultra-wideband and neural networks," *Signals, Systems and Electronics*, vol. 54, no. 4, pp. 529–532, Aug. 2007.

[14] A. Doucet, S. J. Godsill, and C. Andrieu, "On sequential simulation-based methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[15] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*.   Springer Verlag, 2001.

[16] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[17] A. Heinrich, M. Majdoub, J. Steuer, and K. Jobmann, "Real-time path-loss position estimation in cellular networks," in *Proceedings of International Conference on Wireless Networks (ICWN'02)*, Jun. 2002.

[18] K.-T. Feng, C.-L. Chen, and C.-H. Chen, "GALE: an enhanced geometry-assisted location estimation algorithm for NLOS environments," *IEEE Trans. Mobile Comput.*, vol. 7, no. 2, pp. 199–213, Feb. 2008.

[19] K. Jones and L. Liu, "What where wi: An analysis of millions of wi-fi access points," May 2007.

[20] K. Jones, L. Liu, and F. Alizadeh-Shabdiz, "Improving wireless positioning with look-ahead map-matching," Aug. 2007.

[21] S. Byers and D. Kormann, "802.11b access point mapping," *Communications of the ACM*, vol. 46, no. 5, pp. 41–46, May 2003.

[22] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*.   New York: Wiley, 2001.

[23] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*.   Norwood, MA: Artech House, 1999.

[24] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*.   Englewood Cliffs, NJ: Prentice Hall, 1979.

[25] P. R. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification and Adaptive Control*.   Englewood Cliffs, New Jersey: Prentice Hall, 1986.

[26] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 34–45, Mar. 1960.

[27] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*.   NY, USA: Academic Press, 1970.

[28] R. Van Der Merwe and E. Wan, "Sigma-point Kalman filters for probabilistic inference in dynamic state-space models," in *Proceedings of the Workshop on Advances in Machine Learning*, 2003.

[29] S. Julier, J. Uhlmann, and H. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Autom. Control*, vol. 45, no. 3, pp. 477–482, Mar. 2000.

[30] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[31] R. Van Der Merwe, "Sigma-point Kalman filters for probabilistic inference in dynamic state-space models," Ph.D. dissertation, Oregon Health & Science University, 2004.

[32] Digital mapping and solutions-teleatlas. [Online]. Available: http://www.teleatlas.com

[33] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "A novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings on Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, Apr. 1993.

[34] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*.   New York, USA: John Wiley & Sons, 1968, vol. I.

[35] H. Driessen and Y. Boers, "MAP estimation in particle filter tracking," Apr 2008.

[36] M. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, June 1999.

[37] T. Kirubarajan, Y. Bar-Shalom, K. R. Pattipati, and I. Kadar, "Ground target tracking with variable structure IMM estimator," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 1, pp. 26–46, Jan. 2000.

[38] P. J. Shea, T. Zadra, D. Klamer, E. Frangione, and R. Brouillard, "Improved state estimation through use of roads in ground tracking," in *Proceedings of Signal and Data Processing of Small Targets*, vol. 4048.   SPIE, 2000, pp. 312–332.

[39] ——, "Precision tracking of ground targets," in *Proceedings of Aerospace Conference*, vol. 3.   IEEE, 2000, pp. 473–482.

[40] M. S. Arulampalam, N. Gordon, M. Orton, and B. Ristic, "A variable structure multiple model particle filter for GMTI tracking," in *Proceedings of International Conference on Information Fusion*, vol. 2, Jul. 2002, pp. 927–934.

[41] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. London: Artech House, 2004, ch. 10.

[42] M. Ulmke and W. Koch, "Road-map assisted ground target tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 3, pp. 1264–1274, Oct. 2006.

[43] Y. Cheng and T. Singh, "Efficient particle filtering for road-constrained target tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 4, pp. 1454–1469, Oct. 2007.

[44] O. Payne and A. Marrs, "An unscented particle filter for GMTI tracking," in *Proceedings of Aerospace Conference*, vol. 3, Mar. 2004, pp. 1869–1875.

[45] L. Hong, N. Cui, M. Bakich, and J. R. Layne, "Multirate interacting multiple model particle filter for terrain-based ground target tracking," *IEE Proc.-Control Theory Appl.*, vol. 153, no. 6, pp. 721–731, Nov. 2006.

[46] G. Kravaritis and B. Mulgrew, "Variable-mass particle filter for road-constrained vehicle tracking," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, 2008.

[47] M. Ekman and E. Sviestins, "Multiple model algorithm based on particle filters for ground target tracking," in *Proceedings of International Conference on Information Fusion*, Jul. 2007.