Fingerprinting Localization in Wireless Networks Based on Received-Signal-Strength Measurements: A Case Study on WiMAX Networks

Mussa Bshara, *Student 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 (PMs) is improved with a new approach called the 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 with an example whose data were collected from a WiMAX network in a challenging urban area in the capitol city of Brussels, Belgium.**

***Index Terms*—Fingerprinting, Global Positioning System (GPS), Global System for Mobile Communications (GSM), location-based service (LBS), navigation, path loss model, positioning, positioning accuracy, power maps (PMs), received signal strength (RSS), road network information, SCORE, time of arrival (TOA), 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, limited coverage, and 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 the cell ID, which has widely been used in Global System for Mobile Communications (GSM) systems, despite its limited accuracy [1]. Using other network resources (information) like the 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), 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 works focused on range measurements depending on TOA, TDOA, and RSS observations; see surveys [2], [4], and [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 (PM) constructed offline for the region of interest. The first alternative is the most common strategy and is the simplest to deploy. The exponential path loss model is known as the Okumura–Hata (OH) model [6], [7], and in a log power scale, it says that the RSS value linearly decreases with the 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 (BSs). The proposed method achieved higher localization accuracy than the conventional localization methods that use the same path loss exponent for all the links. Furthermore, the authors of [9] proposed using an RSS statistical lognormal model and a sequential Monte Carlo localization technique to get better localization accuracy. The lognormal model was also used in [10] to estimate the mobile location, and the authors tried to mitigate the influence of the propagation environment by using the differences in signal attenuations.

The second alternative is to determine the RSS values in each point and save these in a database (i.e., a map). This can be done using offline measurement campaigns adaptively by contribution from users or by using cell planning tools. The advantage of this effort is a large gain in the SNR 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*, which proved

Manuscript received January 27, 2009; revised July 3, 2009. First published August 18, 2009; current version published January 20, 2010. The work of U. Orguner and F. Gustafsson was supported in part by the SSF Strategic Research Center MOVIII and in part by the Vinnova/FMV TAIS project ARCUS. The review of this paper was coordinated by Dr. Y. Gao.

M. Bshara and L. Van Biesen are with the Department of Fundamental Electricity and Instrumentation, Vrije Universiteit Brussel, 1050 Brussels, Belgium (e-mail: mbshara@vub.ac.be; lvbiesen@vub.ac.be).

U. Orguner and F. Gustafsson are with the Department of Electrical Engineering, Linköping University, 581 83 Linköping, Sweden (e-mail: umut@isy.liu.se; fredrik@isy.liu.se).

Digital Object Identifier 10.1109/TVT.2009.2030504

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 lognormal model. The authors of [13] used fingerprinting to overcome the inconveniences related to the use of the TOA, the angle of arrival, and the RSS lognormal 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 improve fingerprinting performance. The new method emphasizes the effects of the BS identities in classical fingerprinting, and it is called the BS-strict method. Then, the use of dynamic motion models is suggested for further improvement. In this regard, we use particle filters (PFs) [14]–[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 this paper suggests for the dynamic case, which is called the 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 types of networks. The importance of the contributions of this paper can be summarized as follows.

1) 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 particularly significant when the SNR in the RSS values is low and the effects of multipath and fading are dominant.
2) Dynamic localization using PFs gives a seamless integration of fingerprinting-type approaches with dynamical motion models and road network information.

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

This 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 Sections III and 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. Modeling 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 the 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, and therefore, 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

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

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

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 "nondetection" 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 signals that are too weak are not detected.

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

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

where P_{BS} is the transmitted signal power (in decibels), 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]. Although it is a global and simple model, there are several problems associated with using it.

1) The transmitted power needs to be known, which requires a protocol and software that allows a higher layer of applications to access this information.
2) The position of the antenna needs to be known. This requires first building a database. Second, it requires that the user application be able to 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.
3) The path loss constant needs to be known, while, in practice, it depends on the local environment.

An alternative model is based on a local PM (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

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

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

where the operator $\hat{\text{E}}$ denotes the corresponding averaging. LPM provides a prediction of the observation (2) in the same way as the OH model in (3) does. However, the LPM should be considered to be more accurate since it implicitly takes care of the line-of-sight/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 with 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.,

$$\hat{\mathbf{h}}(x) \triangleq [\,\hat{h}_1(x) \quad \hat{h}_2(x) \quad \cdots \quad \hat{h}_{N_{\mathrm{BS}}}(x)\,]^T \tag{5}$$

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 the antenna position, transmitted power, or path loss constant is not needed, enabling mobile-centric solutions. The price for this is the cumbersome task of constructing the LPM. Here, three main alternatives are plausible.

1) Collect the fingerprints during an offline phase. The 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 time consuming and expensive.
2) Use the principle of *wardriving* [19], where the users contribute online 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 to position other users.
3) Predict the fingerprints using Geographical Information System planning tools [2]. Using the radio propagation formulas to predict the RSS values is not as accurate as measuring them because it is not possible to model all the propagation effects. As a result, the predicted data are not as accurate as the measured ones, but they are quite easy to obtain.

In this paper, 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 offline phase. The LPM has been formed from this database as follows.

1) N_{LPM} different grid points denoted as $\{p^i \triangleq [x^i, y^i]^T\}_{i=1}^{N_{\mathrm{LPM}}}$, 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 10 m has been left between these LPM points.
2) For each piece of data that has been collected, the closest LPM grid point has been found.
3) For each LPM grid point i, the vector $\hat{\mathbf{h}}^i$ (called the "RSS vector" or fingerprint) is formed such that

$$\hat{\mathbf{h}}^i = [\,\hat{h}_1^i \quad \hat{h}_2^i \quad \cdots \quad \hat{h}_{N_{\mathrm{BS}}}^i\,]^T \tag{6}$$

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 are no RSS data from the jth BS assigned to the ith LPM grid point, we set $\hat{h}_j^i = \mathrm{NaN}$, representing a nondetection. Note that each fingerprint (or RSS) vector $\hat{\mathbf{h}}^i = \hat{\mathbf{h}}(p^i)$ is a representative of the expected RSS values 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

$$\mathbf{y} = [\,y_1 \quad y_2 \quad \cdots \quad y_{N_{\mathrm{BS}}}\,]^T \tag{7}$$

where the values y_j are equal to the measured RSS values from the jth BS or are 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 paper, 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 a *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 quite frequently appear. However, even in the static localization, the 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 mathematically be posed as the maximum-likelihood (ML) estimation problem given as follows:

$$\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} = p^{\hat{i}} \tag{8}$$

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

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, then any norm (or normlike functions) would do the job. The same would be true in the case where the places of NaN values and 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].

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

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

The norm $\|\cdot\|$ (although, most of the time, its effects 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 nonmatching 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.

1) When the measurement vector $\mathbf{y}$ has a 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, the positions p^i that are far from the BS are more likely.

2) When the measurement vector has a value for some BS (this means that the receiver has got an 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.

The use of this (in a way) negative information in localization to different extents is the main theme of this paper. The localization hypotheses $\hat{\mathbf{h}}^i$ having nonmatching NaN values, which we call *nonmatching hypotheses*, are punished by our proposed methods. Two different likelihood calculation mechanisms (and, 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 nonmatching 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 the *BS-soft* approach.

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

2) Algorithm 2—BS-Strict: This approach calculates the likelihoods in the same way as Algorithm 1 does, but this time, the elements γ_j^i of the vector Γ^i are defined as

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

Notice that the infinite punishment given to the nonmatching 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 will therefore 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

In this paper, the PMs of all available sites in the measurement area shown in Fig. 1 have been generated and plotted in Fig. 2. In the following sections, fingerprinting as defined in (8) and (9) is applied to the RSS index (RSSI) and SCORE values where the likelihoods $p(\mathbf{y}|\hat{\mathbf{h}}^i)$ involved are calculated by either the classical method or the BS-strict approach defined in Section III.

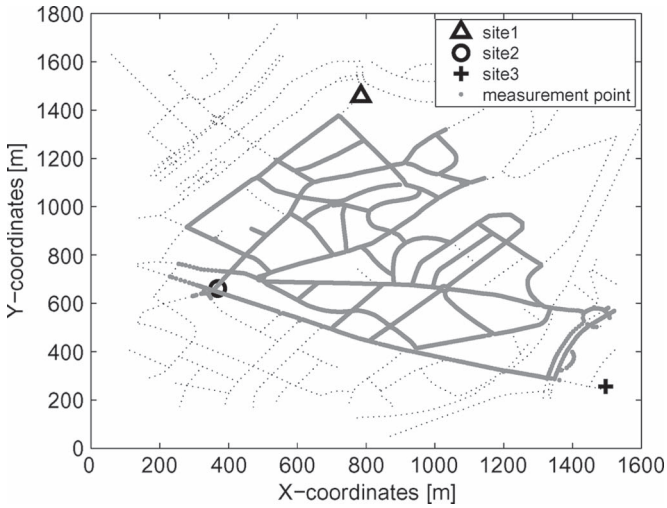

Fig. 1. Area under study (the measurement area). The average distance between two sites is about 1150 m.

A. Fingerprinting Using RSSI Values

In this section, we suppose that the user can accurately measure (the same accuracy as the PM) the received power (RSSI values). This can be done (and has been done in this paper) using special calibrated modems with extra software installed, and the measurements have to be collected offline, 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 paper 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 Fig. 3 and was used to position a user. The two mentioned approaches were applied: the classical fingerprinting (Algorithm 1) and the BS-strict fingerprinting (Algorithm 2). The results are shown in Fig. 4. The BS-strict fingerprinting approach's performance was significantly better than the classical one 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 strong noise, but different RSS values will be collected.

B. Fingerprinting Using SCORE Values

The SCORE values are used by the standard WiMAX modems to evaluate the connection quality between the subscriber station 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 of simultaneously obtaining them for all the available BSs, but the disadvantage lies in their low accuracy compared with RSSI values. The relation between SCORE values and RSSI values

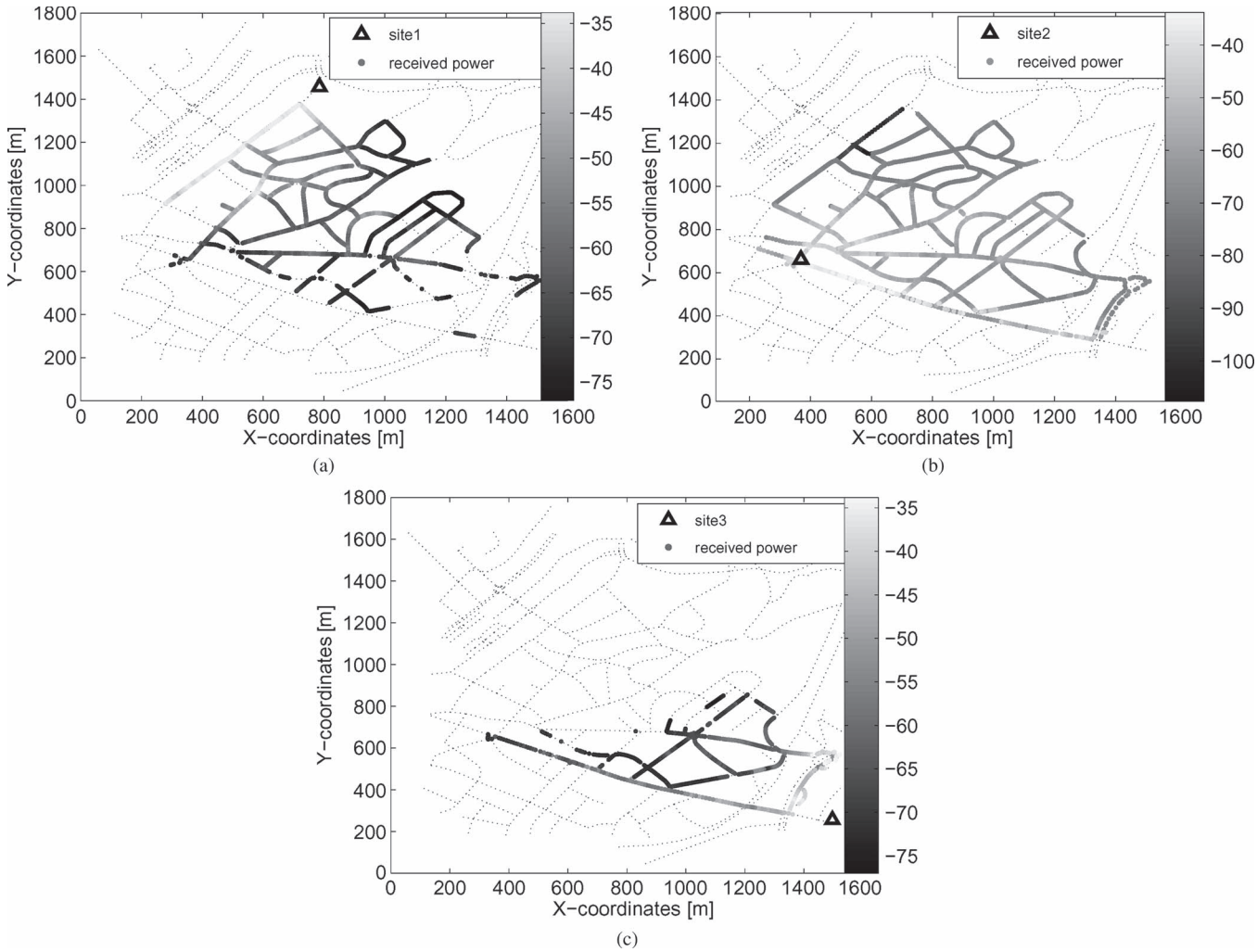

Fig. 2. PMs of the three WiMAX sites. (a) Site 1. (b) Site 2. (c) Site 3.

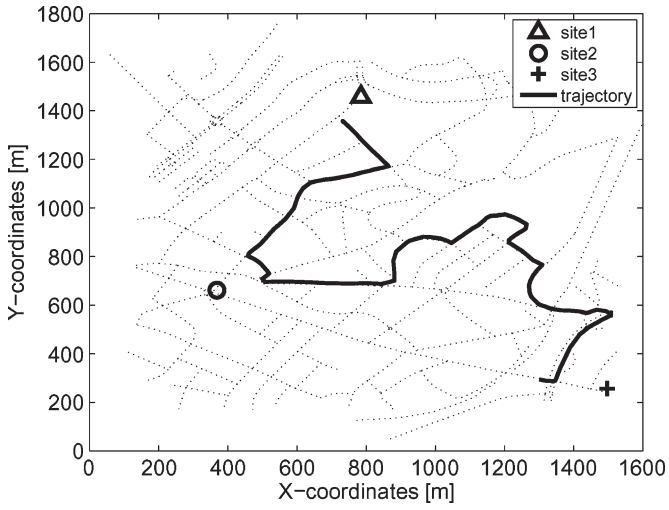

Fig. 3. Used target trajectory.

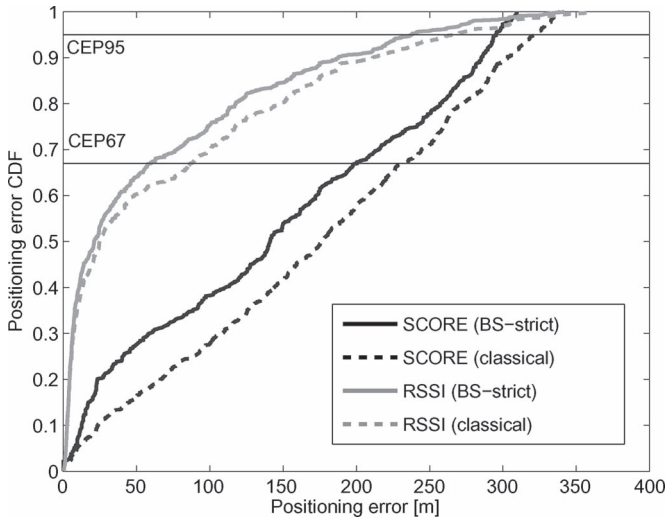

Fig. 4. Positioning error cdf's. The two fingerprinting approaches were used (the classical and the BS-strict) with the available measurements (RSSI and SCORE).

is given, according to the information provided by the modem manufacturer, by

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

where the AvgViterbi value is statistically computed 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 a nuisance for localization. Measurements were collected using the same trajectory to validate the two fingerprinting approaches (using the same database built using RSSI values). Fig. 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 the SCORE values depend on not only the received power but the quality of the signal determined by the Viterbi decoder as well.

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 fullest extent is reasonable because by doing this, we achieve the following.

1) We extract most out of a single measurement.
2) 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., nondetection events or NaN values) in the measurements $\mathbf{y}$ has been used to completely eliminate some positioning hypotheses 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 the information-gathering process, which enables higher estimation performance. Moreover, the complete elimination of some hypotheses (like the assignment of infinite cost to nonmatching hypotheses in Algorithm 2) can result in error accumulation in a recursive procedure because a hypothesis deletion can never be compensated for in the future, even if some contrasting evidence appears. Thus, assigning still higher but finite costs to nonmatching hypotheses, hence allowing them (or some of them) to survive, is more suitable in dynamic estimation procedures. Since such a cost assignment procedure makes the hypothesis punishment softer than that in Algorithm 2 by assigning finite costs to nonmatching hypotheses (compared with the infinite punishment in Algorithm 2, which results in the hypothesis elimination), we call the resulting methodology as the "soft" approach. In the following, 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 $\{y_j\}_{j=1}^{N_{\mathrm{BS}}}$ of the measurement vector $\mathbf{y}$ are conditionally independent, given the database RSS vector $\hat{\mathbf{h}}^i$.
2) Matching non-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 the measurement noise for the jth BS.

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

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

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 separately considered as follows.

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

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

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

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

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 is greater than, e.g., ten, 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 BS to BS.
2) If $y_j =$ NaN (we do not get any measurement from the jth BS) and $\hat{h}_j^i \neq$ NaN, we then have

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

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

$$= \mathrm{cdf}_{e_j}\left(y_{\min} - \hat{h}_j^i\right) \tag{20}$$

where $\mathrm{cdf}_{e_j}(x) \triangleq \int_{-\infty}^{x} p_{e_j}(x)dx$ is the cdf 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 cdf). The probability density function appears in the calculation again as a design parameter. Notice here that, although it is the same density as that required in the previous case, the density $p_{e_j}(.)$ can be selected differently in 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)

punishment for the nonmatching 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 that is similar to the 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 this 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 greater the punishment is, i.e., β_{ij} is smaller.

3) 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

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

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

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

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

which is simple to calculate in a way similar to (21) but has been seen to give low performance in preliminary simulations. The reason for this has been investigated and is 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 $y_j \neq \text{NaN}$ (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 $\hat{h}_j^{i_1} = \text{NaN}$ (for the i_1th hypothesis, we do not have any data for the jth BS) and $\hat{h}_j^{i_2} \neq \text{NaN}$ (for the i_2th hypothesis, we have 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 nonmatching (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 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 nonmatching 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 (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

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

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

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

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

1) 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.

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

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

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

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

where only numeric values are considered in the minimization.

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

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

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

[1] In fact, the data collected in this paper (i.e., $\{\hat{h}_j^i\}_{i=1}^{N_{\text{LPM}}}$ for $j = 1, \ldots, N_{\text{BS}}$) 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 $\bar{h} \triangleq \max_{1 \leq j \leq N_{\text{BS}}} \max_{1 \leq i \leq N_{\text{LPM}}} \hat{h}_j^i$ and subtract it from all the data and the online measurements when they are collected to obtain equivalent data and measurements that satisfy the assumption for a value of $y_{\min}$.

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

1) $\mathbf{x}_{t_k} \in \mathbb{R}^{n_x}$ is the state of the target at time t_k.
2) $w_{t_{k+1},t_k} \in \mathbb{R}^{n_w}$ 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*.
3) $f_{t_{k+1},t_k}(.,.)$ 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 $\mathbf{y}_{t_k}$ that is related to the state of the target as

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

where we have the following.

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

The state estimation with this type of probabilistic model, which is 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 [27], and the methods called sigma-point Kalman filters [28], of which the unscented Kalman filter [29], [30] is one type, have been shown to be suitable for a much larger class of nonlinearities (see the extensive work in [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 multimodal, which can quite poorly be approximated with a single Gaussian distribution. Complicating the facts, the measurement function $h(.)$ that is represented by the PM is highly nonlinear, and furthermore, it is discontinuous. Therefore, in this paper, we are going to use the relatively recent algorithms in the literature called PFs [14]–[16]. Two PFs 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* PFs for obvious reasons. Knowing that the user is on the public road network is valuable information for the positioning of the user. The *TeleAtlas* maps have been used as assisting data, in addition to the measured data [32].

A. PF

PFs are the recursive implementation of Bayesian density recursions [14]–[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 PF 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.,

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

Then, at each time step, the PF 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 a *bootstrap filter* that was first proposed 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 [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 PF described is one of the simplest and computationally cheapest algorithms among the more complicated ones, as given in [36] and [14]. In the following section, we will describe the specific models and parameters that are used in the two differently (off-road and on-road) implemented PFs.

B. Implementation Details of the PFs

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

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

$$\begin{bmatrix} p^x_{k+1} \\ p^y_{k+1} \\ v^x_{k+1} \\ v^y_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_2 & T_{k+1}\mathbf{I}_2 \\ \mathbf{0} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} p^x_k \\ p^y_k \\ v^x_k \\ v^y_k \end{bmatrix} + \begin{bmatrix} \frac{T^2_{k+1}}{2}\mathbf{I}_2 \\ T_{k+1}\mathbf{I}_2 \end{bmatrix} w_{k+1} \quad (36)$$

where w_k is a 2-D 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 PF (called an 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 approaches based on multiple-model (extended) Kalman filters [37]–[39], the PFs, 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]–[47], which appeared only during the last five years. Our approach here considers a single reduced-order on-road motion model with a bootstrap filter. The state of the PF is denoted by $\mathbf{x}^r_k$, where r stands for emphasizing road information, and it is given as $\mathbf{x}^r_k = [p^r_k, v^r_k, i^r_k]^T$, where the scalar variables p^r_k and v^r_k denote the position and speed values of the target on the road segment, which is identified by the integer index i^r_k. The following model is used for the dynamics of $\mathbf{x}^r_k$:

$$\begin{bmatrix} p^r_{k+1} \\ v^r_{k+1} \\ i^r_{k+1} \end{bmatrix} = f^r \left(\begin{bmatrix} \mathbf{p}^r_{k+1} \\ \mathbf{v}^r_{k+1} \\ i^r_k \end{bmatrix}, \mathcal{I}_{RN}, w^{r_d}_{k+1} \right) \quad (37)$$

where

$$\begin{bmatrix} \mathbf{p}^r_{k+1} \\ \mathbf{v}^r_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T_{k+1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p^r_k \\ v^r_k \end{bmatrix} + \begin{bmatrix} \frac{T^2_{k+1}}{2} \\ T_{k+1} \end{bmatrix} w^{r_c}_{k+1}. \quad (38)$$

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

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

$$\mathbf{y}_{t_k} = [\, y_1 \quad y_2 \quad \cdots \quad y_{N_{\rm BS}} \,]^T \quad (39)$$

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

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

1) Calculate the distance of the particle to all of the LPM grid points as

$$d_j = \left\| p^{(i)}_{t_k} - p^j \right\|_2 \quad (40)$$

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

2) Find the closest point in the LPM to the particle position as

$$\hat{j} = \arg \max_{1<j<N_{\rm LPM}} d_j. \quad (41)$$

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

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

where $p(\mathbf{y}_{t_k}|\hat{\mathbf{h}}^{\hat{j}})$ 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 an $N_{\rm BS}$ vector with all elements being equal to NaN. $d_{\rm threshold}$ is a user-selected distance threshold that determines the largest distance between a particle and an LPM grid point at which the LPM grid point can be used to calculate the likelihood of the particle. This is going to be particularly important in the off-road PF 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 $p(\mathbf{y}_{t_k}|\hat{\mathbf{h}}^{\hat{j}})$ implicitly punishes such a particle. We selected $d_{\rm threshold} = 100$ m in our simulations.

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

4) Initialization: PFs were initialized with a large Gaussian spread of particles with mean at the true positions and zero velocities, i.e.,

$$\left[\, p^{x,(i)}_0 \quad p^{y,(i)}_0 \quad v^{x,(i)}_0 \quad v^{y,(i)}_0 \,\right]^T \sim \mathcal{N}(., m_0, P_0) \quad (43)$$

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

$$m_0 \triangleq [\, \bar{p}^x_0 \quad \bar{p}^y_0 \quad 0 \quad 0 \,]^T \quad (44)$$

$$P_0 \triangleq \mathrm{diag}\left([\, 100^2 \quad 100^2 \quad 10^2 \quad 10^2 \,]\right). \quad (45)$$

Here, $[\bar{p}^x_0, \bar{p}^y_0]$ is the true target coordinates at time t_0, and the operator $\mathrm{diag}(.)$ forms a diagonal matrix from the elements of

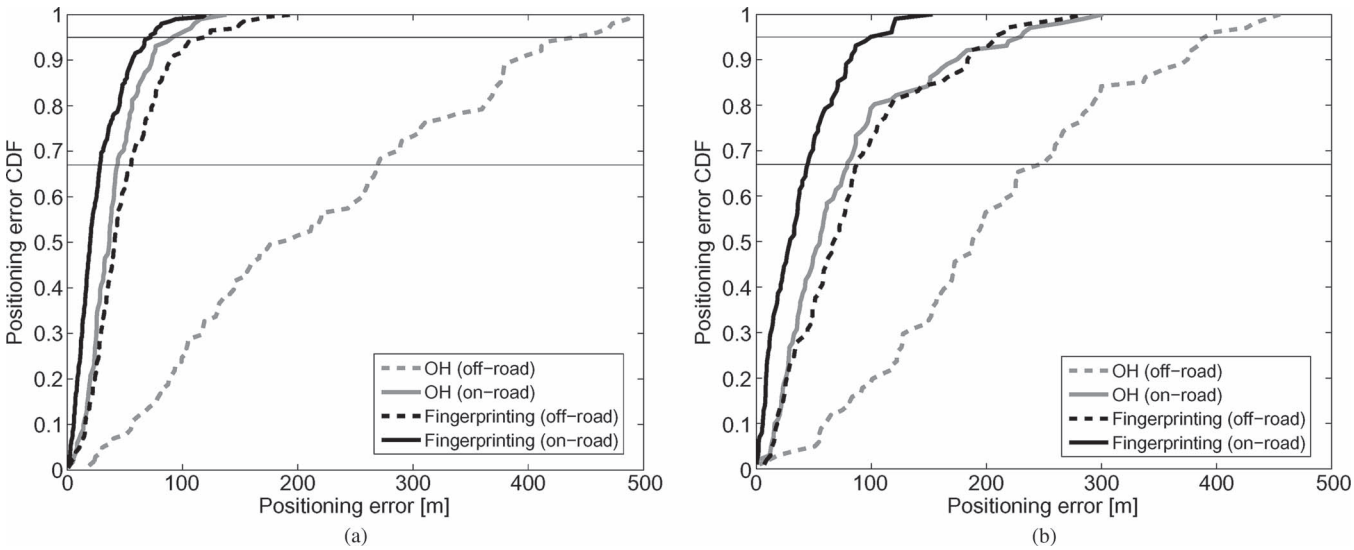

Fig. 5. Positioning error cdf's for the proposed fingerprinting approach and the conventional approach (based on the OH model). (a) Using RSSI measurements. (b) Using SCORE measurements.

the input vector. The results that have been obtained in this paper 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 has a position standard deviation of 100 m, which is, in a way, an indirect assumption of prior information for the initial target position with that quality. It is unfortunately not possible to initially distribute the particles to the whole area of study 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 PF, we directly use the initial particles. On the other hand, in the on-road PF, which always needs particles that are on the road network, the corresponding particles are obtained by projecting the ones defined earlier onto the road network.

To compare our fingerprinting-based bootstrap filters, we have implemented two additional (on-road and off-road) 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 (which 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 Fig. 5. Notice that using SCORE or RSSI measurements with the OH model in the off-road filter gives almost the same results because the dominating model errors like fading overcome the effect of the accuracy of the measurements, and the difference is no longer visible. In the on-road 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 significantly exceeds that of the OH-model-based approach. The performance gain with fingerprinting is overwhelming in the off-road case but is still visible in the on-road filters, particularly with the SCORE measurements where the SNR is lower. It is remarkable that the on-road OH-model-based PF is almost equivalent to the off-road fingerprinting-based filter in terms of estimation errors, which clearly illustrate the effect of the 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 Fig. 6. A very interesting observation is that, in the high-accuracy parts of the RSSI case, the 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 a 10-m performance loss in the RSSI-based dynamic on-road filter compared to the static result. We attribute this difference to the fact that the static estimation calculates an ML estimate, whereas the dynamic on-road filter calculates a mean square estimate. Since there are about 10 m between the LPM grid points and the PF 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 5-m radius. Calculating the average of these particles, which may be biased toward one side of the optimal result due to the road constraints, can give an error of about 5 m. Considering the error terms added by averaging over all the particles, we can expect an error of about 10 m in the result compared with the ML-based static approach, which would directly give the position of the most likely LPM grid point when the SNR is high (like the case with RSSI). The calculation of the MAP estimate in the PF can be an alternative for this problem. In the off-road case, since the particles are even more separated, we can expect this lower performance effect (under a high SNR) to be more visible. Furthermore, we think that the lack of road-network information also makes the estimates of the off-road filter suffer from low prior information

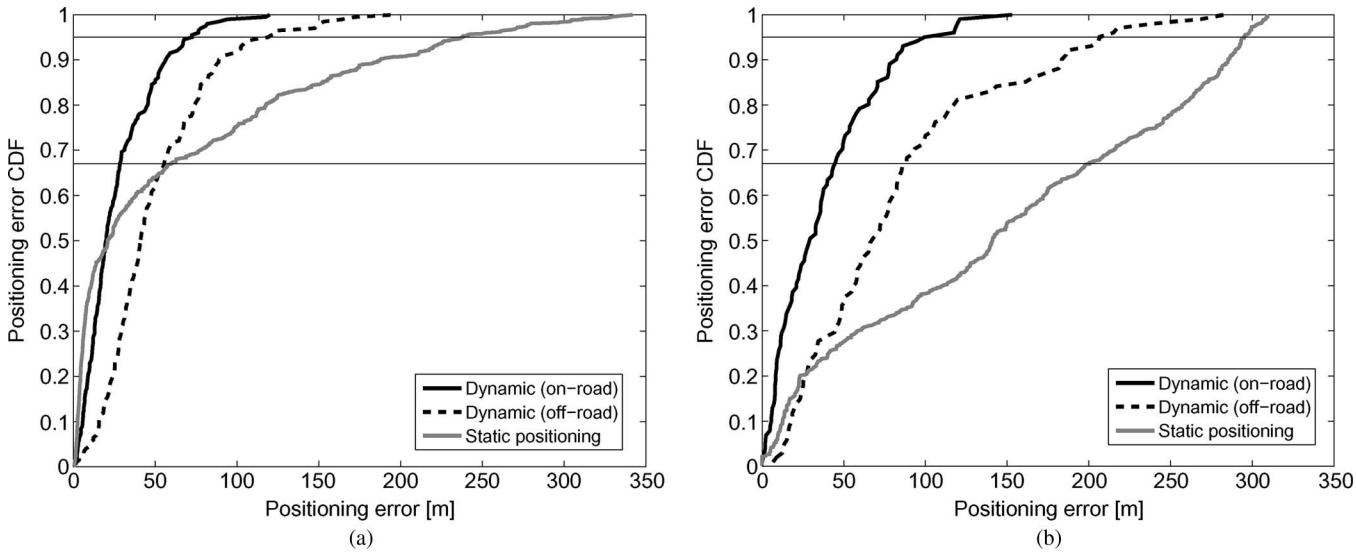

Fig. 6. Positioning error comparison between (on-road and off-road) dynamic positioning and static positioning. (a) Using RSSI measurements. (b) Using SCORE measurements.

compared with 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 compared with 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 compared with the dynamic case is about 50% in the case of SCORE values and about 40% in the 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, i.e., RSSI and SCORE, with specific remarks on WiMAX networks. The introduced work has been divided into two main parts: static localization and dynamic localization. In the latter, the information of the target's motion model was used with and without road information. In both 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 for overcoming 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 the fingerprinting-positioning approach with the motion model and road information can therefore be seen as a further step toward more accuracy-demanding applications and new types of location-based services.

References

[1] Cello Consortium Rep. [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, no. 4, pp. 41–53, Jul. 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, Jul. 2005.
[4] S. Gezici, Z. Tian, B. Giannakis, H. Kobayashi, and A. Molisch, "Localization via ultra-wideband radios," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 70–84, Jul. 2005.
[5] D. Li and Y. Hu, "Energy-based collaborative source localization using acoustic microsensor array," *J. Appl. Signal Process.*, vol. 2003, pp. 321–337, Jan. 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. Elect. 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. VT-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," in *Proc. Veh. Technol. Conf.*, 2008, pp. 1509–1513.
[9] W. D. Wang and Q. X. Zhu, "RSS-based Monte Carlo localisation for mobile sensor networks," *IET Commun.*, vol. 2, no. 5, pp. 673–681, May 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," in *Proc. Veh. Technol. Conf.*, Sep. 2004, vol. 6, pp. 4096–4099.
[12] K. K. C. Takenga, "Mobile positioning based on pattern-matching and tracking techniques," *ISAST Trans. Commun. Netw.*, 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," in *Proc. Signals, Syst. Electron.*, Aug. 2007, pp. 529–532.
[14] A. Doucet, S. J. Godsill, and C. Andrieu, "On sequential simulation-based methods for Bayesian filtering," *Stat. Comput.*, vol. 10, no. 3, pp. 197–208, 2000.
[15] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. Berlin, Germany: 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 *Proc. ICWN*, 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," in *Proc. IEEE Int. Conf. PORTABLE*, May 2007, pp. 1–4.
[20] K. Jones, L. Liu, and F. Alizadeh-Shabdiz, "Improving wireless positioning with look-ahead map-matching," in *Proc. 4th Annu. Int. Conf. MobiQuitous*, Aug. 2007, pp. 1–8.
[21] S. Byers and D. Kormann, "802.11b access point mapping," *Commun. 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, NJ: 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*. New York: Academic, 1970.
[28] R. Van Der Merwe and E. Wan, "Sigma-point Kalman filters for probabilistic inference in dynamic state-space models," in *Proc. Workshop Adv. Mach. Learn.*, 2003, p. 377.
[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, Sci. Univ., Portland, OR, 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," *Proc. Inst. Elect. Eng.—Radar Signal Process.*, vol. 140, no. 2, pp. 107–113, Apr. 1993.
[34] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*, vol. I. New York: Wiley, 1968.
[35] H. Driessen and Y. Boers, "MAP estimation in particle filter tracking," in *Proc. IET Semin. Target Tracking Data Fusion: Algorithms Appl.*, Apr. 2008, pp. 41–45.
[36] M. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. Amer. Stat. Assoc.*, vol. 94, no. 446, pp. 590–599, Jun. 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 *Proc. SPIE Signal Data Process. Small Targets*, 2000, vol. 4048, pp. 312–332.
[39] P. J. Shea, T. Zadra, D. Klamer, E. Frangione, and R. Brouillard, "Precision tracking of ground targets," in *Proc. IEEE Aerosp. Conf.*, vol. 3, 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 *Proc. Int. Conf. Inf. Fusion*, Jul. 2002, vol. 2, pp. 927–934.
[41] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. London, U.K.: 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 *Proc. IEEE Aerosp. Conf.*, Mar. 2004, vol. 3, 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," *Proc. Inst. Elect. Eng.—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 J. Adv. Signal Process.*, vol. 2008, p. 321 967, Jan. 2008.
[47] M. Ekman and E. Sviestins, "Multiple model algorithm based on particle filters for ground target tracking," in *Proc. Int. Conf. Inf. Fusion*, Jul. 2007, pp. 1–8.

Mussa Bshara (S'09) received the Bachelor's degree in electrical engineering from Damascus University, Damascus, Syria, and the M.Sc. degree in signal processing and information security from the Beijing University of Posts and Telecommunications, Beijing, China. He is currently working toward the Ph.D. degree with the Department of Fundamental Electricity and Instrumentation, Vrije Universiteit Brussel, Brussels, Belgium.

His research interests include localization, navigation and tracking in wireless networks, signal processing, wireless communications, power line communications, and x digital subscriber line (xDSL) technologies.

Umut Orguner (S'99–M'07) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Middle East Technical University, Ankara, Turkey, in 1999, 2002, and 2006, respectively.

From 1999 to 2007, he was with the Department of Electrical and Electronics Engineering, Middle East Technical University, as a Teaching and Research Assistant. Since January 2007, he has been a Postdoctoral Associate with the Division of Automatic Control, Department of Electrical Engineering, Linköping University, Linköping, Sweden. His research interests include estimation theory, multiple-model estimation, target tracking, and information fusion.

Fredrik Gustafsson (S'91–M'93–SM'05) received the M.Sc. degree in electrical engineering and the Ph.D. degree in automatic control from Linköping University, Linköping, Sweden, in 1988 and 1992, respectively.

Since 2005, he has been a Professor of sensor informatics with Department of Electrical Engineering, Linköping University. From 1992 to 1999, he held various positions in automatic control, and from 1999 to 2005, he had a professorship in communication systems. He is a Cofounder of the companies NIRA Dynamics and Softube, developing signal processing software solutions for the automotive and the music industry, respectively. He is currently an Associate Editor for the *EURASIP Journal on Applied Signal Processing* and the *International Journal of Navigation and Observation*. His research interests are in stochastic signal processing and adaptive filtering and change detection, with applications to communication, vehicular, airborne, and audio systems. His work in the sensor fusion area involves the design and implementation of nonlinear filtering algorithms for localization and navigation and tracking of all kinds of platforms, including cars, aircraft, spacecraft, unmanned aerial vehicles, surface and underwater vessels, cell phones, and film cameras for augmented reality.

Dr. Gustafsson was elected as a member of the Royal Academy of Engineering Sciences in 2007. He received the Arnberg prize by the Royal Swedish Academy of Science in 2004. He was an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2000 to 2006.

Leo Van Biesen (SM'96) received the Electro-Mechanical Engineer and the Doctoral (Ph.D.) degrees from the Vrije Universiteit Brussel (VUB), Brussels, Belgium, in 1978 and 1983, respectively.

Currently, he is a Full Senior Professor with the Department of Fundamental Electricity and Instrumentation, VUB. He teaches courses on fundamental electricity, electrical measurement techniques, signal theory, computer-controlled measurement systems, telecommunication, physical communication, and information theory. His current interests are signal theory, PHY layer in communication, time-domain reflectometry, wireless communications, x digital subscriber line (xDSL) technologies, and expert systems for intelligent instrumentation.

Dr. Van Biesen was the Chairman of the International Measurement Confederation (IMEKO) TC-7 from 1994 to 2000 and the President Elect of IMEKO from 2000 to 2003 and has been the Liaison Officer between the IEEE and IMEKO. He was the President of IMEKO from 2003 to September 2006. He is currently the Chairman of the Advisory Board of IMEKO as the immediate past President. He is also member of the board of Federation des Ingenieurs de Telecommunication des Communautées Européens Belgium and Union Radio-Scientifique International Belgium.