Particle filters for location-aware services

Mark A. Cusack and Simon R. Maskell

Advanced Processing Centre, QinetiQ ltd, Malvern WR14 3PS, UK
{cusack,smaskell}@signal.qinetiq.com

Abstract. We present a novel method for determining the position of a mobile computer user based on the intensity of the signals received from a collection of wireless network access points. Our scheme uses a combination of Bayesian filtering and route planning techniques that account for the geometry of the building, to determine the probable trajectory of a user moving through an indoor environment. Our particle filter scheme has applicability beyond wireless network-based localisation, and could be used to improve the accuracy and precision of other positioning technologies. The approach has the potential to provide reliable indoor positioning information to support location-aware services.

1 Introduction

The provision of services to mobile users based on their location represents the archetypal pervasive computing application. Within such applications, the position of the user provides the context in which tailored services can be offered to meet the information needs of the individual.

Typically, location-based services focus on providing information regarding people, places and things [1]. For example, a location-based service might offer information on the whereabouts of friends and family; it might provide a weather report based on current location; or it could report the position of a stolen item such as a mobile phone or car.

In addition to providing information, a person's location can also be used to provide an additional layer of security and safety in the form of authentication and access control of services [2]. For example, in a hospital environment, confidential medical records might only be available to healthcare workers inside the hospital building, or an x-ray machine might only operate when the radiographer is in a position that is suitably shielded.

The principal problem to be addressed in systems that provide location-based services is the accurate positioning of people with respect to each other, physical locations or devices. Depending on the application, the position of the user may be required with an accuracy ranging from centimetres to hundreds of metres. A number of technologies can provide positioning information over these length scales. In the context of outdoor environments, global positioning systems (GPS) and cellular phone networks can be used to provide a person's location with an accuracy of ≈ 1m and ≈ 100m, respectively.

For positioning indoors, neither GPS nor cellular-based techniques are suitable, due respectively to the weakness of the signal and poor accuracy. Instead, a range of other devices and techniques have been developed for the purposes of positioning people within buildings. Active sensing systems based on ultrasound, infrared, laser range finding, and both narrow band and ultra wide band radio frequencies (RF) have been developed. In addition, passive sensing technologies for positioning which rely on radio frequency identification (RFID) technology, cameras and physical contact have also been investigated. Hightower and Borriello have recently surveyed positioning technologies [3].

One of the most popular methods for determining the location of a wireless-enabled computer user exploits the characteristics of the wireless network itself. Various researchers have recently developed techniques for people localisation using the signal intensities[1] received from IEEE 802.11b access points [4, 5]. The reasons for the popularity of this approach are that it can exploit (and add value to) an existing wireless network, and that the cost of the sensing infrastructure (the access points) is relatively low when compared to many other indoor positioning technologies.

However, wireless local area network (WLAN) based systems also possess a number of disadvantages. First, they require the creation of a model of the propagation of RF throughout the area in which the position of the user is to be determined. In general, indoor environments severely affect the propagation of wireless signals, and the presence of multi-path effects and interference means that a realistic wireless propagation model cannot be generated analytically. Instead the model must be created empirically, by conducting a site survey of RF signal strength. Depending on the complexity and size of the environment, the process of surveying signal intensities can be time consuming and error-prone.

Second, the propagation of RF in the 2.4 GHz band is particularly susceptible to absorption by people, which can cause the propagation model to deviate from the ground truth at different times of the day. More generally, changes to the physical environment may need to be reflected in the propagation model, if positioning accuracy is to be preserved. Changing the orientation of the antenna on the client wireless card can also produce deviations in the received signal strength from the model.

Finally, the received signal strength also depends on the hardware used and the number and positions of the access points. Changes to any of these can impact the position estimate.

Given the assumption that the wireless infrastructure and the gross features of the environment remain constant, the problem becomes one of how to minimise the impact of the significant random errors in the measured signal strength. A number of Bayesian filtering approaches have been applied to the problem of reducing the measurement noise in WLAN-based positioning systems. The application of the Kalman filter and its extended variants has been demonstrated previously [6, 7]. Recently, the more generally applicable particle filter approach

[1] We will use the terms signal strength and intensity interchangeably to refer to these quantities.

has been used to improve accuracy [8, 9]. Particle filters have the advantage of being able to model non-Gaussian noise and multi-modal probability distributions, which leads to flexible and more robust error correction.

In this paper, we describe our technique for establishing the position of a wireless-enabled Personal Digital Assistant (PDA) user, based on an analysis of the received signal intensities and Bayesian filtering. Our scheme differs from previous applications of Bayesian filtering to the problem of WLAN-based geolocation through its use of route planning and obstacle avoidance algorithms during the probabalistic prediction phase.

We begin by describing a novel approach to determining an initial estimate of the user's location. We then examine how to reduce the uncertainties associated with our position estimate using a Kalman filter. Finally we compare the results using a Kalman filter with those using a particle filter technique that uses a route planning algorithm to account for the presence of walls and doors in a building.

2 WLAN-based localisation

The principal method for establishing location using wireless networks is the nearest neighbour in signal space (NNSS) approach [4]. During the positioning cycle, a candidate set of N signal strengths are received from the available access points. The vector of these signal strengths is I. I is compared to entries in a database containing recorded signal strengths measured at known locations, x^j; x^j is a vector (of two values) consisting of the Cartesian position corresponding to the entry in the database. Typically, these locations may be spaced several metres apart. This database represents the results of a previous RF survey in the physical area of interest. The signal strength vector at location x^j is $I\left(x^j\right)$. The mobile user's location is estimated to be at the position corresponding to the database entry that minimizes the Euclidian distance in the abstract signal space, i.e. the minimum of $|I - I\left(x^j\right)|$:

$$\langle x \rangle = \arg\min_{x^j} |I - I\left(x^j\right)| \tag{1}$$

A further refinement of the estimate can be made by using weighted averages of multiple nearest neighbours in signal space [4, 6]. The weights in this averaging are then the inverse of the distance, such that the estimate of the position is:

$$\langle x \rangle = \frac{\sum_{j \in J_n} f_I\left(I, I\left(x^j\right)\right) x^j}{\sum_{j \in J_n} f_I\left(I, I\left(x^j\right)\right)} \tag{2}$$

where $j \in J_n$ means that the sum is over the n nearest neighbours and $f_I\left(I, I\left(x^j\right)\right)$ is the weight, which is:

$$f_I\left(I, I\left(x^j\right)\right) = \frac{1}{|I - I\left(x^j\right)|} \tag{3}$$

Note that if J_n is simply the nearest neighbour in signal space, $j^\star$, then equation (2) becomes:

$$\langle x \rangle = x^{j^\star} \tag{4}$$

which is exactly the same as equation (1).

2.1 Likelihood approach

The weights, $f_I\left(I, I\left(x^j\right)\right)$, can be interpreted as probabilities or likelihoods, which makes it possible to motivate an alternative scheme for deriving a position estimate from the measured intensities.

Using a weight that is the inverse of the Euclidean distance in signal space will result in an equal weighting being given to each of the signals received in the positioning cycle. This can result in a lack of robustness; using equation (1) or equation (3), if some of the received signals imply that the position is near (in signal space) to one entry in the database and the other signals imply that the position is near another entry then the estimate can be prone to jumping between the two corresponding positions. It is possible to use a different form of the weight to counter this effect, namely a Student-T distribution, in which case:

$$f_I^\star\left(I, I\left(x^j\right)\right) = \prod_{k=1}^{N}\left(1 + \frac{\left(I^k - \bar{I}^k\left(x^j\right)\right)^2}{\mu{\sigma_s}^2}\right)^{-\frac{\mu+1}{2}} \tag{5}$$

where the product over k is over the available access points. I^k is then the intensity corresponding to the kth available access point and $I^k\left(x^j\right)$ is the corresponding intensity for the jth entry in the database. σ_s is the uncertainty over the signal strength; it is the standard deviation between the measured intensity and the intensity near a point in the database predicted at a position. μ is a parameter governing the uncertainty over this value. Since this approach is robust, we found we could simply use all the points in the database as the n nearest neighbours:

$$\langle x\rangle = \frac{\sum_j f_I^\star\left(I, I\left(x^j\right)\right)x^j}{\sum_j f_I^\star\left(I, I\left(x^j\right)\right)} \tag{6}$$

We used $\sigma_s = 3$ and $\mu = 5$ throughout and we will refer to this approach as the "likelihood" method for deriving position estimates from the measured intensities.

2.2 Interpolation

One can extend these ideas to try to visualise the information encoded in each of the received intensities in isolation. To do this, we need to be able to model how the intensity changes with position. So, there is a need to interpolate between the positions in the database to predict intensities for a grid of candidate points.

To perform this interpolation, we use a weighted estimate of the intensities in the database to get an estimate of the intensity at a position, x:

$$\langle I^k(x)\rangle = \frac{\sum_j f_x\left(x, x^j\right) I^k\left(x^j\right)}{\sum_j f_x\left(x, x^j\right)} \quad (7)$$

where

$$f_x\left(x, x^j\right) = \mathcal{G}\left(x; x^j, \sigma_p^2 \times \mathbf{I}_2\right) \quad (8)$$

where $\mathbf{I}_2$ is a 2×2 Identity matrix and $\mathcal{G}(x; \langle x\rangle, C)$ is a Gaussian distribution for x with a mean of $\langle x\rangle$ and a covariance of C:

$$\mathcal{G}(x; \langle x\rangle, C) = \frac{1}{\sqrt{|2\pi C|}} \exp\left(-\frac{1}{2}(x - \langle x\rangle)^T C^{-1} (x - \langle x\rangle)\right) \quad (9)$$

where x^T is the transpose of a vector x and C^{-1} and $|C|$ are respectively the inverse and the matrix determinant of a matrix, C.

This makes it possible to derive an estimate of the kth signal strength at any point (and so interpolate between the points in the database). σ_p is another design parameter which models the concept of *geometric proximity* and so controls the extent of the influence of the entries in the database in terms of geometric distance: reducing σ_p results in the interpolated intensities moving closer to the entries in the database when near to the corresponding position; increasing σ_p results in the interpolated intensities forming a smoother surface. We found an appropriate compromise was achieved using $\sigma_p = 3$ m.

The likelihood of the measured intensity from the kth access point is then $p\left(I^k|x\right)$, which is assumed to be a Gaussian distribution with mean $\langle I^k(x)\rangle$ and standard deviation, σ_s, such that:

$$p\left(I^k|x\right) = \mathcal{G}\left(I^k; \langle I^k(x)\rangle, \sigma_s^2\right) \quad (10)$$

Values of $p\left(I^k|x\right)$ can be calculated for a grid of values, x, and for each of the available access points. The result is that the surfaces defined by $p\left(I^k|x\right)$ are a visualisation of the likely positions given the measured signal strengths in isolation.[2]

The values derived at each point can then be multiplied across the available access points to obtain fused values of $p(I|x)$ for the grid of values of x:

$$p(I|x) = \prod_{k=1}^{N} p\left(I^k|x\right) \quad (11)$$

$p(I|x)$ can then be used to visualise the likely positions given all the measured signal strengths.

[2] A pedantic author might emphasise that $p\left(I^k|x\right)$ is a likelihood function parameterised by x; it is not a probability distribution and is not the posterior $p\left(x|I^k\right)$. Hence, it is really a tool for visualising how likely the measured signal is at different positions and should not strictly be interpreted as the probable positions given the measured signal strengths. These quantities would only be equivalent if the prior on the position was uniform over the visualised region. These issues are not explored further in this paper.

2.3 Location estimation results

The architecture for our prototype positioning system consists of a Web service, a PDA (IPAQ), and a wireless network comprising five IEEE 802.11b access points distributed throughout an office environment. The PDA captures signal intensities and sends them over the wireless network to the Web service using the simple object access protocol (SOAP). The Web service generates an estimate of the position using equation (6). The Web service can provide this location estimate to other Web services, which in turn can supply tailored services to the PDA user based on their location. The aim is to then use Bayesian filtering and route planning operations on the sequence of these initial estimates to reduce the impact of measurement noise; this will be described in detail in section 3. Here, it suffices to say that position estimation is performed server-side because of the complex nature of Bayesian filtering and the limited computational resources on the PDA.

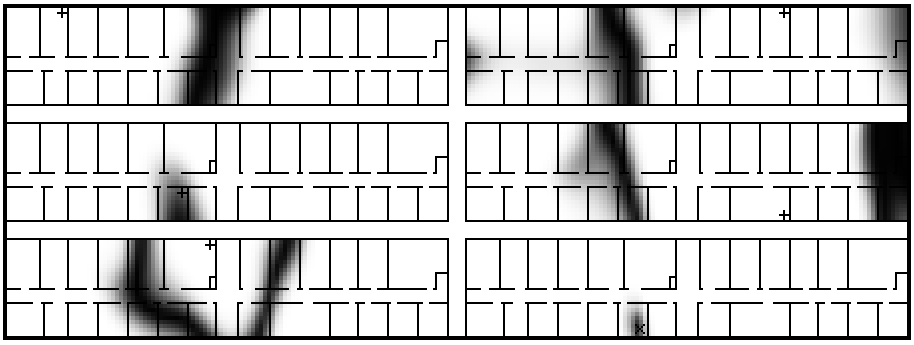

Fig. 1. Representations of the likely positions of a wireless PDA in an office environment. The darker regions indicate likely positions for the PDA. The figure shows these likely positions calculated from the perspective of each of the five access points individually (each labelled '+' in the corresponding image). The combined probability is shown in the lower right of the figure, and the user's actual position is indicated by '×'.

The process of determining the probable location of a PDA user in the office environment is visualised in figure 1. The figure shows six plan views of the same office space, with lines indicating the walls. The lower right section of the figure visualises the likely position of the user, as $p(I|x)$, which is calculated according to equation (11). This likelihood is highly localised within a single office near to the user's actual position, which is labelled by '×'. As described previously, this likelihood is calculated at each point as the product of the likelihoods perceived from each of the five access points. The other five sections of the figure visualise the likely positions given the signal strength for each of the access points in isolation, as $p\left(I^k|x\right)$. The five access points are indicated by a '+'. The five likelihoods were calculated using equation (10).

The shape of the likelihoods for each of the wireless access points hints at the role that the access points play in locating a user. Each access point acts as a range finding sensor, localising the PDA to the circumference of a circle centred on the access point. This is best illustrated in the bottom left section of figure 1, which shows the user's likely position as being confined to a band around the access point. The presence of walls and other obstructions tends to distort the shape of the likelihood from a circle.

In figure 2 we present the results of experiments in tracking the position of a mobile user moving through an office building. The figure shows the positioning error and the cumulative probabilities for two different position estimation techniques: the well-known NNSS approach; and our likelihood approach which was described in section 2.1. The results are derived from one hundred signal samples taken during ten traversals of the same path between two offices. The positioning errors represent the deviation of the estimated position from the ground truth. The ground truth is represented by a series of waypoints at which samples of signal strength are taken and an estimate of position is computed. The NNSS values were obtained using a weighted average of the nearest three neighbours in signal space.

In our tests, position estimates obtained using the likelihood approach have a higher probability of being accurate to within 1 m of the true position, when compared to estimates derived using the NNSS approach: a probability of 0.35 for the likelihood approach; and a probability of 0.18 in the case of the NNSS scheme.

In the sections that follow, we describe how the positioning errors can be reduced by the application of Bayesian filtering and route planning, and provide the results of tracking experiments performed with a Kalman filter and a particle filter. First, however, we provide a brief overview of Bayesian tracking theory.

3 Tracking as sequential Bayesian inference

Tracking can be posed as the sequential Bayesian inference problem. The state of the target contains all the pertinent information relating to the target, which might just be the position of the target, but could also contain information relating to the velocity of the target or other quantities of interest. The uncertainty over the target's state, x_t, resulting from the history of the incoming stream of estimates of position (derived from the wireless signal intensities), $z_{1:t}$, is represented using the probability distribution $p\left(x_t|z_{1:t}\right)$. As each new measured position, z_t, is received, this distribution is sequentially updated using the model for the dynamics of the target, $p\left(x_t|x_{t-1}\right)$ and the likelihood $p\left(z_t|x_t\right)$:

$$\underbrace{p\left(x_t|z_{1:t}\right)}_{\text{New track}} \propto \int \underbrace{p\left(z_t|x_t\right)}_{\text{Likelihood}} \underbrace{p\left(x_t|x_{t-1}\right)}_{\text{Dynamics}} \underbrace{p\left(x_{t-1}|z_{1:t-1}\right)}_{\text{Previous track}} dx_{t-1} \tag{12}$$

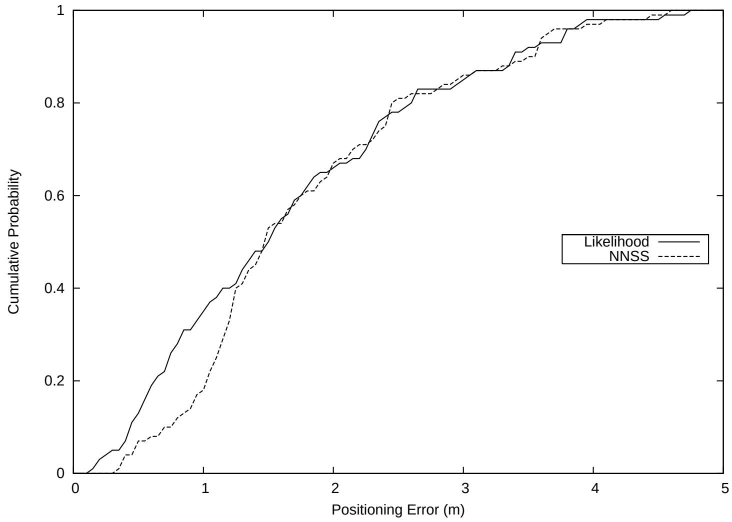

Fig. 2. Cumulative probabilities and positioning errors for tracks determined using the nearest neighbour in signal space approach and our likelihood approach which is described in section 2.1.

We now describe in moderate detail how this generic process is conducted when using a Kalman filter and a particle filter. We suggest that, if the reader is more interested in the results than the details of the generic implementation, the reader should skip these subsections and move on to section 4.

3.1 Kalman filter

If the dynamics and likelihood are (or can be approximated as) linear and Gaussian (so parameterised by a set of matrices) then the track is parameterised using a mean and covariance matrix[3]. A Kalman filter can then be used to recursively calculate these quantities.

So, the dynamics and likelihood are assumed to be of the following form:

$$p\left(x_t|x_{t-1}\right) = \mathcal{G}\left(Fx_{t-1}, Q\right) \tag{13}$$

$$p\left(z_t|x_t\right) = \mathcal{G}\left(Hx_t, R\right) \tag{14}$$

where z_t is a measurement of position derived from the received intensities, I_t, and where $\mathcal{G}\left(m, C\right)$ is a Gaussian distribution with mean m and covariance C. These models are therefore parameterised by four matrices; F, Q, H and R.

[3] If the models are linear and Gaussian then the Kalman filter is the optimal algorithm to use. If the models used are approximated as being linear and Gaussian then the algorithm is often referred to as an Extended Kalman filter and is an approximate algorithm.

This is equivalent to having:

$$x_t = Fx_{t-1} + \omega_t \tag{15}$$
$$z_t = Hx_t + \epsilon_t \tag{16}$$

where

$$\omega_t \sim \mathcal{G}(0, Q) \tag{17}$$
$$\epsilon_t \sim \mathcal{G}(0, R) \tag{18}$$

and where $x \sim p(x)$ means that x is a sample drawn from the distribution $p(x)$.

The process begins with some initial estimate of the state $m_{0|0}$ and some associated covariance, $C_{0|0}$. At time t, the Kalman filter stores an estimate of the state, $m_{t|t}$, and an associated covariance matrix, $C_{t|t}$. The Kalman filter then operates by using the matrices parameterising the models to update these two quantities in the light of the received measurement, z_t. Specifically, the recursive operation of the Kalman filter is as follows:

$$m_{t|t-1} = Fm_{t-1|t-1} \tag{19}$$
$$C_{t|t-1} = FC_{t-1|t-1}F^T + Q \tag{20}$$
$$m_{t|t} = m_{t|t-1} + P_{xz}P_{zz}^{-1}\left(z_t - Hm_{t|t}\right) \tag{21}$$
$$C_{t|t} = C_{t|t-1} - P_{xz}P_{zz}^{-1}P_{xz}^T \tag{22}$$

where

$$P_{xz} = HC_{t|t-1} \tag{23}$$
$$P_{zz} = HC_{t|t-1}H^T + R \tag{24}$$

A fuller description of Kalman filter-based tracking algorithms may be found in [10].

3.2 Particle filter

The particle filter adopts a different approach and uses the diversity of a set of (a few hundred) particles to convey the uncertainty over the track. Each particle represents a candidate state for the target, x_t^i, and also has an associated weight, w_t^i. The particles are initially sampled according to the prior on the state, which could be the Gaussian distribution with the mean and covariance used to initialise the Kalman filter ($m_{0|0}$ and $C_{0|0}$). At each subsequent iteration, the particles are propagated through some convenient proposal distribution, $q(x_t|x_{t-1}, z_t)$, which moves each particles from where they hypothesised that the target was to some candidate positions for where the target might be:

$$x_t^i \sim q\left(x_t|x_{t-1}^i, z_t\right) \tag{25}$$

The weight then reflects the disparity between the true distribution and the proposal that was used to propagate the particles and is calculated as follows using evaluations of the probability distributions:

$$w_t^i \propto w_{t-1}^i \frac{p\left(x_t^i|x_{t-1}^i\right) p\left(z_t|x_t^i\right)}{q\left(x_t^i|x_{t-1}^i, z_t\right)} \tag{26}$$

where the weights are normalised across the particles to sum to unity.

The advantage of this approach is that the proposal distribution can be of a convenient (usually Gaussian) form, but evaluations of the models update the weights; the (potentially approximate) models are not restricted to be linear and Gaussian as with the Kalman filter. The impact is that any information that would be lost in approximating the models as linear and Gaussian (which one is forced to do when using an Extended Kalman Filter) can be used by the particle filter. This information can be critical to performance and, when it is, the particle filter can offer improved performance over the Kalman filter.

The disparity between the proposal distribution and the true track results in the weights becoming skewed—one of the particles eventually has a weight of one and all the others have a weight of zero. To alleviate this effect, the samples are probabilistically replicated and discarded on the basis of the weights such that the total number of particles remains constant. This can be carried out at each time step or a diagnostic can be used to determine when to carry out this *resampling* step.

The particle filter is the sequential Bayesian inference analogue of evolutionary algorithms, with which the reader may be more familiar: the sampling step plays the role of the mutation step, the weight evaluation corresponds to the cost calculation and the resampling is reminiscent of the selection step. A fuller description of particle filtering is available in [11].

4 Application to wireless localisation

4.1 Kalman filter

If we assume that the position of the PDA at a point in the future is approximately where the PDA was at a recent point in the past then it is appropriate to model the movement with a Gaussian distribution; the difference in position is a sample from a Gaussian distribution. Since the position has two components, the Gaussian distribution is bivariate. Such a model is known as a random walk model and is equivalent to defining the matrices parameterising the model as follows:

$$F = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{27}$$

$$Q = \begin{bmatrix} T & 0 \\ 0 & T \end{bmatrix} q \tag{28}$$

where T is the time step between measurements and q is a quantity that determines the scale of the random perturbations. We used a popular algorithm based on the Kalman filter, known as the interacting multiple model (IMM) [10]. The IMM automates the selection of the value of q from a number of candidates. We used two candidate values: $q = 0.1$ m^{2}s^{-1} and $q = 0.001$ m^{2}s^{-1}. The fact that Q, which defines the scale of the random perturbations is scaled by T means that the size (variance) of the perturbations scales (linearly) with the time between measurements; if the time between measurements happens to be large, the uncertainty will be larger than if this time is short. This is intuitively appealing.

The likelihood used by the Kalman filter is that derived from equation (6). In that we are trying to estimate position from a sequence of measurements of position in the presence of noise, we can define the matrices as follows:

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{29}$$

$$R = \begin{bmatrix} \sigma_z^2 & 0 \\ 0 & \sigma_z^2 \end{bmatrix} \tag{30}$$

where σ_z is the standard deviation of the position estimate derived from the received intensity. It therefore represents the uncertainty associated with the position estimate. We used $\sigma_z = 3$ m.

So, operation of the filter consists of substituting equations (28) to (30) into equations (20) to (24) and so deducing an estimate of the position of the PDA as $m_{t|t}$.

4.2 Particle filter

The particle filter uses a dynamic model based on the shortest path between the sampled positions at the two time steps, x_{t-1}^i and x_t^i. The shortest path is calculated using the A^* search algorithm [12], which accounts for the geometry of the environment in which tracking is being performed. The distance travelled is then assumed to be a Gaussian distribution with mean zero. So, in calculating the weight update, $p\left(x_t^i|x_{t-1}^i\right)$ is taken to be:

$$p\left(x_t^i|x_{t-1}^i\right) = 2\mathcal{G}\left(d\left(x_t^i, x_{t-1}^i\right), qT\right) \tag{31}$$

where $d\left(x_t^i, x_{t-1}^i\right)$ is the distance calculated by the A^* search algorithm. The 2 is present to ensure $p\left(x_t^i|x_{t-1}^i\right)$ integrates to one given that distances are constrained to be greater or equal to zero. This choice of model is equivalent to the random walk model used above if no walls are present. This enables the particles' weights to reflect the effect of movement constraints imposed by the presence of walls.

The particle filter uses the same model as the Kalman filter for the measurement process:

$$p\left(z_k|x_k\right) = \mathcal{G}\left(Hx_k, R\right) \tag{32}$$

The proposal distribution used is the random walk model outlined in the previous section:

$$x_t \sim q\left(x_t^i | x_{t-1}^i, z_t\right) = \mathcal{G}\left(Fx_{t-1}^i, Q\right) \tag{33}$$

So, for each particle, operation of the particle filter consists of simulating values of x_t^i from $\mathcal{G}\left(Fx_{t-1}^i, Q\right)$ and then substituting equations (31) and (32) into equation (26) to calculate the particle weights. Resampling is then used if necessary.

5 Results of Bayesian filtering

We have applied Kalman and particle filters to the estimates of position generated by the likelihood approach (see section 2.1). Figure 3 shows the probability of obtaining an estimate of position for a given error. The figure allows for a comparison between the accuracy obtained using the likelihood approach, a Kalman filter, a particle filter that exploits the A^* search algorithm, and the particle filter with route planning disabled.

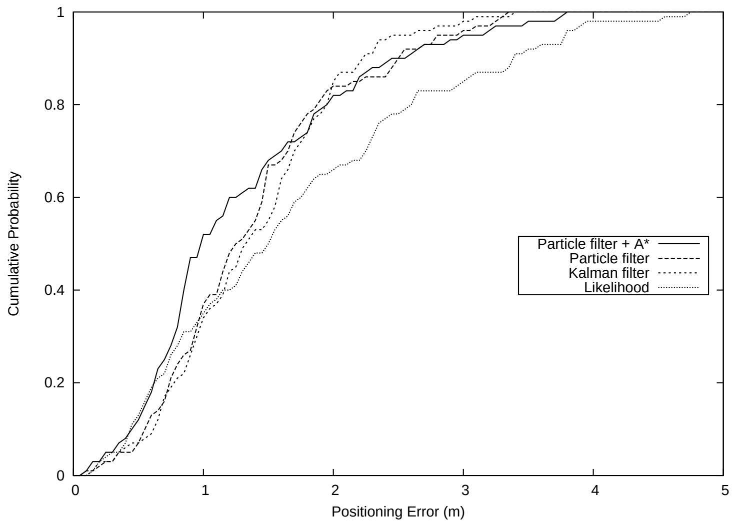

Fig. 3. Cumulative probabilities and positioning errors for tracks determined using the Kalman filter, particle filter and particle filter combined with A^* route planning. The positioning errors for the likelihood approach are also given for comparison.

In general, both the particle filter methods and the Kalman filter outperform the likelihood method. The particle filter approach that accounts for the geometry of the building offers a probability of 0.55 of obtaining an estimate of position that is accurate to within 1 m. A particle filter without the route planning functionality provides a similar level of accuracy with a probability of 0.35. Clearly there are benefits to be gained from a treatment of the geometry of the environment in which positioning is taking place.

The results for the Kalman filter and the particle filter are similar in the case of no route planning. Any differences between the two graphs in figure 3 are likely to be due to the randomness of the algorithm used in the particle filter approach and our choice of the number of particles. Over a repeated set of experiments one would expect the results of the Kalman filter and the particle filter to converge, in the absence of route planning.

The particle filter approach has an advantage over the Kalman filter because the application of the A^* search algorithm to the particle filter method is easier to implement and interpret. To illustrate, in our approach the A^* algorithm is used to supply the length of the navigable path between a particle's previous position and its currently measured position. This path length is determined for all particles. We interpret this path length as a probability according to equation (31), and the longer the path length, the lower the confidence there is in that particle's proposed trajectory. This means that the importance of certain trajectories will tend to diminish with respect to others, leading to more credible position estimates. It is less clear how such a scheme could be applied in the case of the Kalman filter.

6 Summary and discussion

We have described a new approach to generating an estimate of a wireless PDA user's position, based on the signal strength received from IEEE 802.11b access points. Our likelihood method effectively treats each access point as a range finding sensor, providing the probable location of the user along the circumference of a circle centred on the access point, but distorted by the presence of walls. We have shown that our likelihood approach provides a higher level of positioning accuracy than the traditional NNSS approach.

We have attempted to increase the accuracy of our tracking scheme further by applying Bayesian filtering techniques to estimates of position generated using our likelihood method. In general, Bayesian filtering provides improved accuracy by accounting for the dynamics of the user and the position estimate at the previous time step, in addition to the position estimate at the current time. We have extended the Bayesian technique to account for the layout of the environment by employing the A^* search algorithm which acknowledges the existence of walls and other obstructions during route planning. Thus, estimates of the position of the user are restricted to corridors, rooms and doorways. Our results show that an improved estimate of position can be obtained by using a combined particle filter/A^* approach.

While wireless positioning combined with Bayesian filtering does provide a reasonable degree of accuracy, a more dependable way of establishing position may be required for certain applications. If people are to trust location-based services enough to use them, then they need to be sure that their location is known to a sufficient degree of accuracy and precision. To address the need for improved accuracy, a strength-in-depth approach has been advocated [13], in which a number of positioning technologies are combined to provide a fused estimate of the position of the user. A simple example of this might be the placement of a short range RFID reader within a corridor. If an RFID tag were to be attached to the user or the PDA, then the position of the user as they walked past the RFID reader could be established unambiguously, and this information used to calibrate the wireless location service.

A great deal of time and effort is required to generate the initial survey of received signal strengths as a function of position. It may be possible to lower the sampling density of the survey by relying on interpolated values of signal strength generated by the method described in section 2.2. Interpolation of values may provide a better initial estimate of position when the sample density is low, when compared to the NNSS method. Attempts have also been made to use learning algorithms to help reduce the number of survey points required for positioning [8]. In this approach, signal strength data is collected by people as they walk through a building and the patterns received are used to form the basis of a motion model of the person. A less comprehensive database of received signal strengths can be exploited, since it is the motion model that is primarily responsible for estimating the position of the person.

The use of particle filters makes it possible to exploit the measured intensities directly using an approach similar to that used in section 2.3 to visualise the data. However, our initial efforts indicate that this necessitates some development of the interpolation scheme used. This is the subject of current work.

7 Acknowledgements

We should like to thank Jeremy Baxter for providing the A^* route planner, Martin Evans, Rob Wright and Mark Briers for providing valuable assistance with the tracking software and Chris Booth and Stéphane Lo Presti for many useful discussions and suggestions. This work is sponsored by the UK Department of Trade and Industry's Next Wave Technologies and Markets Programme.

References

1. T. Kindberg et al. People, places, things: Web presence for the real world. *Technical Report HPL-2000-16, Internet and Mobile Systems Laboratory, HP Laboratories Palo Alto*, 2000.
2. S. Creese, M. Goldsmith, B. Roscoe, and I. Zakiuddin. Authentication for pervasive computing. *In Proceedings of the First International Conference on Security in Pervasive Computing*, February 2003.
3. J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 34:57–66, August 2001.
4. P. Bahl and V. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. *In Proceedings of IEEE Infocom 2000*, pages 775–784, March 2000.

5. J. Small, A. Smailagic, and D.P. Siewiorek. Determining user location for context aware computing through the use of a wireless LAN infrastructure. 2000.
6. I. Guvenc, C.T. Abdallah, R. Jordan, and O. Dedeoglu. Enhancements to RSS based indoor tracking systems using Kalman filters. *In Proceedings of GSPx & International Signal Processing Conference*, March 2003.
7. J. Latvala, J. Syrjärinne, S. Niemi, and J. Niittylahti. Patient tracking in a hospital environment using wireless stations and extended Kalman filtering. *In Proceedings of the 1999 Middle East Conference on Networking*, 1999.
8. M. Berna, B. Sellner, B. Lisien, S. Thrun, G. Gordon, and F. Pfenning. A learning algorithm for localizing people based on wireless signal strength that uses labeled and unlabeled data. *In Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, August 2003.
9. S.M. Siddiqi, G.S. Sukhatme, and A. Howard. Experiments in Monte-Carlo localization using WiFi signal strength. *In Proceedings of the International Conference on Advanced Robotics (ICAR '03)*, June 2003.
10. Y. Bar-Shalom and X-R Li. *Multitarget-Multisensor Tracking: Principles and Techniques.* YBS, 1995.
11. S. Arulampalam, S.R. Maskell, N.J. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 2001.
12. P. Winston. *Artificial Intelligence.* Addison-Wesley, 1992.
13. J. Hightower, D. Fox, and G. Borriello. The location stack. *Technical Report IRS-TR-03-008, Intel Research, Seattle*, 2003.