A cellular positioning system based on database comparison - The hidden Markov model based estimator versus the Kalman filter

Trond Nypan¹ and Oddvar Hallingstad²

UniK – University Graduate Center, PO Box 70, N-2027 Kjeller, Norway, ¹trond@unik.no, ²oh@unik.no

ABSTRACT
Different techniques for calculating the position of user terminals in cellular networks have been proposed. No method has yet proven superior in terms of accuracy, cost, system impact and coverage. We consider database comparison of measured location sensitive parameters to be a promising positioning technique. Due to ambiguity and signal degradation, the position estimates from the comparison process must be filtered. In this paper we discuss two different filtering approaches. One is based on hidden Markov modeling and the other is based on the Kalman filter. We present the two methods and compare their ability to estimate position data of a database comparison process.

1. INTRODUCTION
Positioning of user terminals in cellular networks is typically done using techniques based on time-of-arrival measurements, like network assisted GPS [1] and observed time difference of arrival [2]. Other methods are based on network available pieces of information like cell-id, cell-sector, timing advanced and received-signal-strength [3]. Hybrid methods using different selections of these parameters, often referred to as enhanced cell-id methods, have also been investigated. A major problem for most positioning techniques is shadowing of the line-of-sight (LOS) signal. In fact no method has yet proven superior in terms of accuracy, coverage, system impact or cost [4].

A technique for user terminal positioning, based on database comparison methods, has been proposed in [5,6]. The main idea of this technique is to store location sensitive parameters of radio signals, measured along streets covered by the cellular communication network, in a database. Later, a moving user terminal can compare its measurements with the ones in the database. In this way, primary position estimates of the user terminals are calculated. Problems with ambiguity and reproducibility of the location sensitive parameters, see section 3, suggest the use of a secondary estimation procedure in order to achieve better accuracy and more robustness [7].

Figure 1 gives an overview of the main processing steps of the positioning system. The location sensitive measurements, denoted Meas. in the figure, are input to the database comparison process. The resulting cost-functions (CFs) are then input to the primary estimation process, which outputs the minimum of the CF as primary position estimate, denoted $y_p$. The position estimates, $y_p$, are then filtered by a Kalman filter (KF), or a hidden Markov filter (HMF). In our research we have used those two methods for filtering the primary position estimates. They both use modeling for signal noise removal. In [7] we presented results using the Kalman filter and in [8] we presented results using the hidden Markov filter. In this paper we focus on the two filtering methods and compare their ability to estimate position data of a database comparison process.
2. SIGNAL MODELING

A real world process generally produces observable outputs, which can be characterised as signals. A problem of fundamental interest is characterising such real world signals in terms of signal models. Broadly one can divide the types of signal models into the class of deterministic models, and the class of stochastic models. Deterministic models generally exploit some known specific properties of the signal, e.g., that the signal is a sine wave or a sum of exponentials, etc. The second broad class of signal models is a set of statistical models in which one tries to characterise also the statistical properties of the signal. Examples of such statistical models include Poisson, Markov, and hidden Markov.

In this paper we concentrate on two methods for secondary estimation. The first method views the observed signal, $y_p$, (see Figure 1) as a probabilistic function of distinct states. The physical system, which we denote $\lambda$, generating the observations is modelled using a hidden Markov model (HMM) [9], which we denote $\lambda^*$, and this is used to construct the hidden Markov filter. HMM has previously been used for positioning by database comparison of the received-signal-level by O. Kennemann [10], S. Mangold, and S. Kyriazakos [11]. The second method models the physical system, $\lambda$, as a first-order ordinary vector difference equation. The observation process is modelled using a vector algebraic equation. The filter design model, which we denote $\lambda^{**}$, is used to construct a KF. This estimator performs optimal estimation, but only when specific conditions are fulfilled [12].

3. DATABASE COMPARISON

In our research we have used the channel impulse response (CIR) for comparison [13]. In most of the coverage area of cellular networks there is a non line-of-sight path between the base-transceiver-station (BTS) and the user terminal. The main propagation mechanism is therefore scattering from the surface of obstacles and diffraction around them. In practice, energy arrives via several paths and a multipath situation is said to exist at the receiver. When measured by a wideband receiver, the CIR yields an estimate of the number of multiple propagation paths as well as their relative delay and strength. The uniqueness of the CIR depends on the topography of the area and the system bandwidth, e.g. built-up and hilly areas experience more distinct CIRs than flat areas when measured by a wideband receiver. The CIR is measured both by the user terminal and the BTS in dedicated mode.

The processing steps of the database comparison are depicted in Figure 2. Basically, the CIRs used as measured input to the comparison procedure are processed identically to the ones in the database, except that averaging is performed over 0.5 seconds in time instead of over 4 meters in distance. Further explanation is given in [7]. At time step $l$, each of the averaged CIR vectors, denoted $h_l$, is compared with the database, denoted $U$, and a vector cost-function (CF), denoted $d(h_l, U)$, is calculated.

Figure 2. The steps of the pattern recognition process. $H$ is the measurement matrix and $U$ is the database. The vector $d(h_l, U)$ is the cost-function (CF). The output is position, $y_p$, and variance, $r_p$.

A CF displays the similarity between the measured CIR and the ones in the database along the road. One example is shown in Figure 3. The CF is calculated according to the Euclidian metric formula, see [14] section 4.6, at every time step $l$. Primary position estimates, $y_p(l)$, are calculated by choosing the distance corresponding to the minimum of the cost-function, i.e. the least Euclidian distance, at every time step $l$. The width and the magnitude of the minimum can then be
used to calculate the position variance, \( r_p(l) \), at each instant of time, \( l \). This process is subject to tuning, and accurate estimation of the variance is unfortunately difficult to achieve. Note that only the KF and not the HMF uses the estimated variance, \( r_p(l) \), as input.

The actual location of the CF in Figure 3 is 94 meters, but we see that the minimum is at about 105 meters, yielding an error of about 11 meters in this case. This is due to problems with the reproducibility of the measured location sensitive parameters. We define the reproducibility as the reproducibility of the measured location sensitive parameter. This means that the user terminal measurements are very similar at two or more different locations. Both these effects are problematic for the positioning process.

4. HIDDEN MARKOV MODELS

HMM has been used in a wide range of applications, e.g. bioscience, control, communication, image, speech, and signal processing. One main application of HMM is speech recognition systems. Characteristic for speech recognition is that the speed of the different speakers is variable. It is thus necessary to compress or expand time in order to match measurements with a recorded database. This feature is normally referred to as dynamic time warping (DTW). In our database comparison system a similar situation occurs. The database records are coupled with discrete positions along streets in the coverage area, but the user terminals to be located have different velocities.

![Figure 4. A left-right Markov chain with 5 states and state transitions](image)

A discrete Markov process (chain) may be described as a system being in one of a set of \( N \) distinct states, \( s_1, s_2, ..., s_N \), as illustrated in Figure 4 (where \( N = 5 \) for simplicity). At regularly spaced discrete times, the system undergoes a change of state (possibly back to the same state) according to a set of transition probabilities associated with the state. We denote the time instants associated with state changes as \( l = 1, 2, ... \) and we denote the actual state at time \( l \) as \( q(l) \). The state transition probabilities, \( a_{ij} \), is on the form

\[
a_{ij} = P(q(l+1) = s_j | q(l) = s_i), \quad i, j \in \{1, 2, ..., N\} \tag{1}
\]

which is the probability that the model will be in state \( s_j \) at time \( l+1 \) if it was in state \( s_i \) at time \( l \), where \( N \) is the number of states. This stochastic process is called an observable Markov model if the output of the process is mapped one-to-one to the states. In our system set-up we use a hidden Markov model, which we denote \( \lambda^* (A, B, \pi) \), where the observed output, \( y_p(l) \), is viewed as a probabilistic function of the state [15]. In our case each state is a position interval along the road. The observation symbol probability distributions, \( b_{ij} \), is on the form

\[
b_{ij} = P(y_p(l) = s_j | q(l) = s_i), \quad i, j \in \{1, 2, ..., N\} \tag{2}
\]

which is the probability of measuring (observing) state \( s_j \) when the model is in state \( s_i \) at time \( l \). The initial state distribution, \( \pi_i \), is on the form

\[
\pi_i = P(q(1) = s_i), \quad i \in \{1, 2, ..., N\} \tag{3}
\]

which is the probability that the model will be in state \( s_i \) at time \( l=1 \).

The transition probability distributions, denoted \( A = [a_{ij}] \), are estimated from the speed distribution of vehicles in the coverage area. The observation symbol probability distributions, denoted \( B = [b_{ij}] \), are estimated directly from cost-functions of an earlier comparison process of the same streets.

Position estimation is performed at time step \( l \), by finding the “optimal” state sequence associated with the given observation sequence, which we denote \( Y = [y_p(1), y_p(2), ..., y_p(l)] \). Our optimality criterion is simply to choose the states \( q(l) \) which are individually most likely, at each time step \( l \). This optimality criterion maximises the expected number of correct individual states. Note that we have not used the more complex Viterbi algorithm, which calculates the most probable state sequence, in our processing.
5. THE KALMAN FILTER

Since the publication of the well-known article by R.E. Kalman [16] in 1960 the Kalman filter has been used in a wide range of applications. Especially in areas like object tracking, economics and navigation the Kalman formulation and solution of the estimation problem has been used with success. The approach is to model the physical system as a discrete dynamic system driven by white noise in the form

\[ x(l+1) = \Phi(l)x(l) + \Gamma(l)w(l) \]  

(4)

where \( l \) represents discrete time, \( x \) is the process state vector, \( \Phi \) is the state transition matrix, \( \Gamma \) is the process noise matrix and \( v \) is the (white) process noise vector. The deterministic control vector \( u \) is here omitted. The initial conditions \( x(0) \sim N(x(0), P(0)) \) and process noise covariance matrix, \( Q_d \), where \( v(l) \sim N(0, Q_d) \), is known. The discrete observations of the process are modelled in the form

\[ y(l) = D(l)x(l) + w(l) \]  

(5)

where \( y(l) \) is the observations at discrete time steps \( l \), \( D \) is the measurement matrix and \( w \) is the (white) measurement noise. The measurement covariance matrix, \( R_d(l) \), where \( w(l) \sim N(0, R_d(l)) \) and

\[ R_d(l) = \begin{bmatrix} r_p(l) & 0 \\ 0 & r_v(l) \end{bmatrix} \]  

(6)

which is calculated in real time as described below. We assume \( x(0), w(l_1), \) and \( w(l_2) \) \( \forall l_1, l_2 \) to be uncorrelated. This constitutes the KF design model, which we denote \( \lambda^s = \{ \Phi, \Gamma, D, Q_d, R_d, x(0), P(0) \} \).

The states of our KF are position, velocity, and acceleration. Acceleration is modelled as a first order Markov process. The parameters of this process are tuned by considering acceleration limitations of a typical vehicle. In addition we model the coloured measurement noise on position and velocity as first order Markov processes. The parameters of these processes are tuned by considering the nature of the database comparison process and the driving pattern of vehicles along the road, respectively.

The discrete KF is constructed from the filter design model, \( \lambda^s \), which is described in detail in [7]. A position update in the KF from time \( t = t_0 \) to \( t = t+1 \) is based on the current state estimate of \( x \), predictions of the next state, and prediction-corrections using measurement inputs.

The input of the Kalman filter is the primary position estimates, \( y_{p,n} \) and the estimated variance, \( r_{p,n} \), see Figure 1. Speed is input as a virtual measurement, which is based on the average speed of vehicles in the area in which the positioning is performed. Due to the ambiguity of the database comparison process the error of the primary position estimation process is not Gaussian. Within a small area however, the noise may be approximated to be Gaussian. The search for the minimum in the next instant of time is then limited to a short distance around the previous minimum. The risk is that the KF no longer “sees” the correct minimum. This suggests the use of multi-hypothesis-tracking procedures in which several KFs are an integral part. In such procedure each of the different hypotheses, i.e. KFs, receives different measurements based on local least-Euclidean-distance points on the cost-function. Unfortunately such estimation algorithms are relative complex and thus demand much processing power.

6. COMPARING THE HMM BASED ESTIMATOR AND THE KF

Our aim is to develop a simple and robust secondary estimation procedure yielding the best possible accuracy for our positioning system. The challenge of the secondary estimation depends on the behaviour of the primary estimation output, \( y_p \). This behaviour is affected by the ambiguity and the reproducibility of the location dependent measurements. A low degree of ambiguity means that most of the time we observe only one distinct minimum on the cost-function, close to the true location of the user terminal. A high degree of ambiguity corresponds to the existence of many minima of almost equal magnitude on the cost-function. We have a high reproducibility factor when the cost-function minimum close to the true location of the user terminal is almost zero, and vice versa we have a low degree of reproducibility when the minimum of the true location is far from zero.

With not much ambiguity and a high factor of reproducibility, the Gaussian approximation of the primary position estimates is valid. Unfortunately, using only the CIR for comparison as we have done yields a more rugged cost-function and the Gaussian assumption is not valid anymore. In such cases a single KF will not improve the overall accuracy of the primary position estimates, as can be seen from the results in [7]. The HMF on the other hand works with any distribution of
primary position estimates with respect to the street
elements. This as what we refer to as the
observation symbol probability distribution \( B \) in
section 4 and this matrix is trained using real data of
a previous comparison process.

In general there will be a difference in speed of user
terminals performing measurements along a street
and the effective speed of the database
measurements. In the KF case the user terminal
speed is estimated from (1) the position estimates,
(2) the average-speed pattern of user terminals in
the street and (3) the limitations on the acceleration.
In practice this means that if a user terminal in a car
suddenly changes its speed pattern, e.g. because of
a queue, in an area where the primary position
estimates are bad, the secondary KF position
estimates are likely to drift far out from the true
location. However, if the pattern of speed changes
can be predicted, e.g. during daily rush hours, this
error can be modelled and the KF based positioning
system would still work well. This requires close
monitoring of the traffic pattern and an interface to
the positioning system. We did not use any such
traffic information but tried to model coloured
behaviour of the speed and acceleration as first-
order Markov processes. Unfortunately this model
was insufficient in order to yield significant
reductions of the KF estimation error. The KF
estimator is relatively sensitive to error in the speed
modeling and the ability to perform DTW is poor
using an estimator set up with only one filter.

The HMF performs the secondary estimation in a
probabilistic manner as described in section 4. A
new position estimate is based on estimating the
most probable state sequence, where the last state in
this sequence is the new position estimate. This
enables multiple-hypothesis evaluation in a simple
manner. Say the user terminal moves down one of
the streets coming out of a junction. One hypothesis
can then be created for every street going out from
the junction by assuming each of the different
streets to be the last state in the input sequences.
The output yielding the highest probability is then
chosen as the position estimate. A similar approach
can be used in the initial phase of an estimation
process were normally the a priori position estimate
has relatively low accuracy. Several HMF’s with
different starting positions can then be realised and
the correct one should yield the highest probability.

As far as the HMF is concerned, there is always a
possibility that a state transition simply returns the
user terminal back to the same state, i.e. because it
is stationary or travels at a lower speed than the
effective speed of the database. In this manner the
difference of speed between the user terminal
measurements and the effective speed of the
database is compensated for. The speed of the user
terminal is modelled by the transition probability
distributions, \( A = [a_{ij}] \), where each state has
assigned a speed distribution. These speed
distributions can be trained using real data or, like
we did, by performing educated guessing. The
HMF is thus relatively insensitive to minor errors in
the speed distributions of the states. An important
fact is that any arbitrary speed distribution can be
empirically estimated and easily implemented in
the HMF set up. This approach to speed modelling
is an import advantage over the KF.

7. RESULTS

We present the positioning results using the HMF
system set up described in [8]. The trial was
performed using measurements from about 12 km
of streets in urban and suburban parts of Munich.
The cumulative error distributions of the primary
and secondary position process for 100 percent of
the time are depicted in Figure 5. The error using
the HMF is less than 24.2 meters in 67 percent of
the time and within 71.3 meters 95 percent of the
time.

The result of the KF trail, presented in [7], showed
that the KF was only able to increase the accuracy
of the primary position estimates 64 percent of the
time. We are thus not able to present both the KF
and the HMF cumulative error distributions
together.

![Figure 5. Cumulative error distribution using HMF of the positioning process of urban and suburban areas of Munich.](image-url)
8. CONCLUSION

The challenge of Kalman filtering is the estimation of the model parameters. Especially when the speed is not measured, as in our case, the position estimate is likely to drift to extreme values. Correct modeling of the speed is important in order to achieve adaptation in the form of DTW, which is crucial for the secondary estimation procedure in a positioning system based on database comparison.

If we compare the HMF and the KF, the HMF parameters are fewer and easier to estimate, the estimator seems to be less sensitive to errors in the model parameters and multi-hypothesis processing is much simpler. We conclude that the HMF approach is more suited to perform position estimation of a database comparison process.

9. REFERENCES