Mobility Tracking in Cellular Networks with Sequential Monte Carlo Filters

Lyudmila Mihaylova*, **David Bull***, **Donka Angelova**** **and Nishan Canagarajah***
* Department of Electrical and Electronic Engineering, University of Bristol, UK,
** Bulgarian Academy of Sciences, 25A Acad. G. Bonchev St, 1113 Sofia, Bulgaria

mila.mihaylova@bristol.ac.uk, dave.bull@bristol.ac.uk , donka@bas.bg, nishan.canagarajah@bristol.ac.uk

Abstract – *This paper considers mobility tracking in wireless communication networks based on received signal strength indicator measurements. Mobility tracking involves on-line estimation of the position and speed of a mobile unit. Mobility tracking is formulated as an estimation problem of a hybrid system consisting of a base state vector and a modal state vector. The command is modelled as a first-order Markov process which can take values from a finite set of acceleration levels. In order to cover the wide range of acceleration changes, a set of acceleration values is predetermined. Sequential Monte Carlo algorithms – a particle filter (PF) and a Rao-Blackwellised particle filter (RBPF) are proposed and their performance evaluated over a synthetic data example.*

Keywords: mobility tracking, Monte Carlo methods, wireless networks, hybrid systems, Rao-Blackwellisation, Singer model

1 Introduction

Mobility tracking is one of the most important features of wireless cellular communication networks. Data from two types of stations are usually used: *base stations* which position is known and *mobile stations* (or mobile users) which location and dynamic motion is being estimated.

Mobility tracking techniques can be divided in two groups [1]: methods in which the position, speed, and acceleration are estimated versus conventional geo-location techniques, which only estimate the position coordinates. Approaches for mobility tracking rely on Kalman filtering [1, 2, 3, 1], hidden semi-Markov models [4, 5, 3] and sequential Monte Carlo filtering [6]. Two types of measurements can be used: pilot signal strengths from different base stations measured at the mobile unit and the corresponding propagating times.

The Kalman-filtering algorithms developed in [1] for real-time tracking of the location and dynamic motion of a mobile station in a cellular network have all limitations and advantages coming from the Kalman filtering framework: necessity of linearisation and the coming from this inaccuracies. The two algorithms proposed in [1] use the pilot signal strengths from neighbouring base stations, i.e., the *Received Signal Strength Indication (RSSI)*, although they are suitable for signal measurements such as time-of-arrival (TOA) information. The mobility model is linear driven by a discrete command process that determines the mobile station's acceleration. The command process is modelled as a semi-Markov process over a finite set of acceleration levels. The first algorithm consists of an averaging filter for processing pilot signal strength measurements and two Kalman filters, one to estimate the discrete command process and the other to estimate the mobility state. The second algorithm employs a single Kalman filter without prefiltering the measurements and is able to track a mobile station even when a limited set of pilot signal measurements is available. Both of the proposed algorithms can be used to predict future mobility behaviour, which can be used to resource allocation applications.

Yang and Wang [6] developed a Monte Carlo algorithm for joint mobility tracking and hard handoff detection in cellular networks. In their work mobility tracking involves on-line estimation of the location and speed of the mobile, whereas *handoff detection* involves on-line prediction of the pilot signal strength at some future time instants. The optimal solution of both problems is prohibitively complex due to the nonlinear nature of the system.

In this paper we focus on mobility tracking based on signal strength measurements. In contrast to previous works [1, 2, 7, 6] mobility tracking in cellular networks is formulated here as an estimation problem of *hybrid systems* which are systems with a base state vector and a mode (modal) state vector. The base states are continuously evolving, whilst the modal states can undergo abrupt changes. This formulation together with the sequential Monte Carlo approach provides us with a powerful tool for mobility tracking. A particle filter and a Rao-Blackwellised particle filter are developed and their performance investigated.

The outline of the paper is as follows. Section 2 contains the problem formulation. Section 3 presents the mobility state and observation models. The mobility tracking and prediction within Bayesian framework is given in section 4. A particle filter for mobility estimation in wireless cellular networks is presented in section 4 and a Rao-Blackwellised particle filter is designed in Section 5. Their performance evaluation is given in Section 7. Conclusions and ongoing research issues are highlighted in the last Section 8.

2 Problem Formulation

We consider the mobility tracking in cellular networks within the sequential Monte Carlo framework. The dynamics of the mobility unit is described by the equation

$$\boldsymbol{x}_k = f(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k(\boldsymbol{m}_k), \boldsymbol{w}_k(\boldsymbol{m}_k)), \tag{1}$$

where $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ is the system *base* state, $\boldsymbol{m}_k \in \mathbb{R}^{n_m}$ is the *modal* state, $\boldsymbol{u}_k \in \mathbb{R}^{n_u}$ specifies the command process, and $\boldsymbol{w}_k \in \mathbb{R}^{n_x}$ is the state noise, with $k \in \mathbb{N}$ being the discrete time and $\mathbb{N}$ is the set of natural numbers. The measurement equation is in the form

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k, \boldsymbol{v}_k), \tag{2}$$

where $\boldsymbol{z}_k \in \mathbb{R}^{n_z}$ is the observation, and $\boldsymbol{v}_k \in \mathbb{R}^{n_v}$ is the measurement noise. Functions $f(.)$ and $h(.)$ are nonlinear in general.

Assume that the observations are taken at discrete time points $T.k$, with a discretisation time step T. A mobile user may have abrupt and unexpected changes in acceleration $\boldsymbol{u}_k$ caused by different reasons such as traffic lights, road turns. On the other hand, the acceleration of the mobile is highly correlated. In order to model these both sides, following [6, 2, 7] we model the moving user as a dynamic system driven by a command $\boldsymbol{u}_k = (u_{x,k}, u_{y,k})'$ and a correlated random acceleration $\boldsymbol{r}_k = (r_{x,k}, r_{y,k})'$ at time k, i.e. the total acceleration is $\boldsymbol{a}_k = \boldsymbol{u}_k + \boldsymbol{r}_k$ (see Fig. 1).

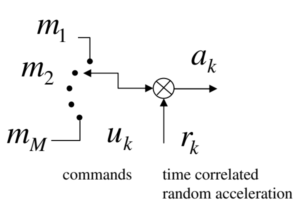

Fig. 1: Structure of the mobility acceleration chain

The purpose is to estimate the current state of the moving object on the basis of the measurements. Since the measurement function is highly nonlinear and the measurement errors are big, we propose a solution to the problem within the sequential Monte Carlo framework.

3 Mobility State and Observation Models

Different state mobility models were previously used in cellular networks such as the constant acceleration model [6] and Singer-type models [3, 1]. In this paper we choose a discrete-time Singer model [1] because it captures correlated accelerations and allows for prediction of position, speed, and acceleration of mobile users. Originally proposed by Singer [8] for tracking targets in military systems, the Singer model has served as a basis for developing many effective maneuver models with various applications (see [9] for a detailed survey), including for user mobility patterns. In the original Singer model the command input is supposed to be a Markov process, which has a time autocorrelation, whilst the Singer-type model from [1] includes a command process in explicit form.

For the sake of conciseness here we present directly the discrete-time form of this Singer-type model. The derivation of the discrete-time model from the corresponding continuous-time model is given in [1]. The state of the moving mobile at time instant k is defined by the vector $\boldsymbol{x}_k = (x_k, \dot{x}_k, \ddot{x}_k, y_k, \dot{y}_k, \ddot{y}_k)'$ where x_k and y_k specify the position, $\dot{x}_k$ and $\dot{y}_k$ specify the speed, and $\ddot{x}_k$ and $\ddot{y}_k$ specify the acceleration in the x and y directions in a two-dimensional grid.

The motion of the mobility user can be described by the equation

$$\boldsymbol{x}_k = \boldsymbol{A}(T,\alpha)\boldsymbol{x}_{k-1} + \boldsymbol{B}(T,\alpha)\boldsymbol{u}_k + \boldsymbol{w}_k, \tag{3}$$

where $\boldsymbol{u}_k = (u_{x,k}, u_{y,k})'$ is a discrete-time command process, the respective matrices in (3) are of the form

$$\boldsymbol{A}(T,\alpha) = \begin{pmatrix} \tilde{\boldsymbol{A}} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \tilde{\boldsymbol{A}} \end{pmatrix}, \boldsymbol{B}(T,\alpha) = \begin{pmatrix} \tilde{\boldsymbol{B}} & \boldsymbol{0}_{3\times1} \\ \boldsymbol{0}_{3\times1} & \tilde{\boldsymbol{B}} \end{pmatrix}, \tag{4}$$

$$\tilde{\boldsymbol{A}} = \begin{pmatrix} 1 & T & a \\ 0 & 1 & b \\ 0 & 0 & e^{-\alpha T} \end{pmatrix}, \tilde{\boldsymbol{B}} = \begin{pmatrix} c \\ \alpha a \\ \alpha b \end{pmatrix}, \tag{5}$$

with $a = (-1 + \alpha T + e^{-\alpha T})/\alpha^2$, $b = (1 - e^{-\alpha T})/\alpha$, $c = (1 - \alpha T + \frac{\alpha^2}{2}T^2 - e^{-\alpha T})/\alpha^2$. The random process $\boldsymbol{w}_k$ is a 6×1 vector, T is the discretisation period, and α is the reciprocal of the autocorrelation of the acceleration time constant. Since $\boldsymbol{w}_k$ is a white noise, $E[\boldsymbol{w}_k \boldsymbol{w}_{k+i}] = 0$, for $i \neq 0$. The covariance matrix $\boldsymbol{Q}$ of $\boldsymbol{w}_k$ is $\boldsymbol{Q} = 2\alpha\sigma_1^2 \boldsymbol{I}_2 \otimes \boldsymbol{Q}_1(T)$, where $\boldsymbol{I}$ denotes the unit matrix, $\otimes$ the Kronecker product. The errors in x_k and y_k direction are assumed the same (see [1] for details) and are characterised with the standard deviation σ_1. The matrix $\boldsymbol{Q}_1(T)$ is symmetric, with dimension 3×3 and entries having the form:

$$\begin{aligned} q_{11} &= (1 - e^{-2\alpha T} + 2\alpha T + 2\alpha^3 T^3/3 - 2\alpha^2 T^2 \\ &\quad -4\alpha T e^{-\alpha T})/(2\alpha^5), && (6) \\ q_{12} &= (e^{-2\alpha T} + 1 - 2e^{-\alpha T} + 2\alpha T e^{-\alpha T} && (7) \\ &\quad -2\alpha T + \alpha^2 T^2)/(2\alpha^4), && (8) \\ q_{13} &= (1 - e^{-2\alpha T} - 2\alpha T e^{-\alpha T})/(2\alpha^3), && (9) \\ q_{22} &= (4e^{-\alpha T} - 3 - e^{-2\alpha T} + 2\alpha T)/(2\alpha^3), && (10) \\ q_{23} &= (e^{-2\alpha T} + 1 - 2e^{-\alpha T})/(2\alpha^2), && (11) \\ q_{33} &= (1 - e^{-2\alpha T})/(2\alpha). && (12) \end{aligned}$$

The unknown command processes $u_{x,k}$ and $u_{y,k}$ are modelled as a first-order Markov chain that takes values from a set of acceleration levels $\mathcal{A} = \{m_1, \ldots, m_M\}$, and the process $\boldsymbol{u}_k$ takes values from the set $\mathcal{M} = \mathcal{A}_x \times \mathcal{A}_y$, with transition probabilities $\boldsymbol{\pi_{ij}} = P(\boldsymbol{u}_k = \boldsymbol{m}_j | \boldsymbol{u}_{k-1} = \boldsymbol{m}_i)$ and initial probability distribution $\mu_{i,0} = P\{\boldsymbol{m} = \boldsymbol{m}_i\}$ for modes $\boldsymbol{m}_i \in \mathcal{A}$ such that $\mu_{i,0} \geq 0$ and $\sum_{i=1}^{M} \mu_{i,0} = 1$.

3.1 Observation Model

A commonly used model [1, 6] in cellular networks for the distance between a mobile and a given base station (BS) relies on the received signal strength indication (RSSI), which is the average of the pilot signal strength received at the mobiles. Denote $z_{k,i}$ the observation, the RSSI signal received by a given mobile from the i-th BS with coordinates (a_i, b_i) at time k. The RSSI can be modelled as a sum of two terms: one due to path loss, and another due to shadow fading. Fast fading is neglected assuming that a low-pass filter is used to attenuate the Rayleigh or Rician fade. Therefore, the RSSI (measured at dB) that the mobile unit receives from a particular BS i at time k, can be modelled as the following function

$$z_{k,i} = z_{0,i} - 10\eta \log_{10}(d_{k,i})^{1/2} + v_{k,i}, \tag{13}$$

where $z_{0,i}$ is a constant determined by the transmitted power, wavelength, antenna height, and gain of cell i; η is a slope index (typically $\eta = 2$ for highways and $\eta = 4$ for microcells in a city); $d_{k,i} = \sqrt{(x_k - a_i)^2 + (y_k - b_i)^2}$ is the distance between the mobile unit and the base station; (a_i, b_i) is the position of the i-th base station; $v_{k,i}$ is the logarithm of the shadowing component, which is found to be a zero mean, stationary Gaussian process with standard deviation $\sigma_{v,i}$, typically from 4 – 8 dB [2]. The shadowing component can considerably worsen the estimation process as it is shown in [3, 1]. This difficulty can be overcome by pre-filtering the measurements (e.g. by an averaging filter) in order to reduce the observation noise.

To locate the mobile station in a two-dimensional plane, three distance measurements to neighboring BSs are sufficient. The necessary data are available in GSM systems where within regular intervals the mobile samples the forward signal levels of six neighbour links. For the considered problem the observation vector consists of the three largest RSSI denoted $z_{k,1}, z_{k,2}, z_{k,3}$. Hence, the measurement equation is of the form

$$\boldsymbol{z}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_{i,k}, \tag{14}$$

with $\boldsymbol{h}(\boldsymbol{x}_k) = (h_1(\boldsymbol{x}_k), h_2(\boldsymbol{x}_k), h_3(\boldsymbol{x}_k))'$, $h_i(x_{i,k}) = z_{0,i} - 10\eta \log(d_{k,i})$ a measurement vector $\boldsymbol{z}_k = (z_{k,1}, z_{k,2}, z_{k,3})'$, shadowing components $\boldsymbol{v}_{k,i} = (v_{k,1}, v_{k,2}, v_{k,3})'$ assumed to be uncorrelated both in time and space, and having Gaussian distribution, $v_{k,i} = \mathcal{N}(0, \sigma_v^2)$.

4 Mobility Tracking and Prediction within Bayesian Framework

Now we look at the real-time estimation of the mobility of a user within the Bayesian framework. Since the command process $\boldsymbol{u}$ is unknown, a hybrid particle $\mathbf{x}_k = \{\boldsymbol{x}_k, m_k\}$ is considered that fully characterises the target state and mode. The mobility state $\boldsymbol{x}_k$ can be evaluated at each time instant from the conditional probability density function $p(\mathbf{x}_k|\boldsymbol{z}_{1:k})$ and a set of measurements $\boldsymbol{z}_{1:k} \triangleq \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_k\}$ up to time instant k according to the Chapman-Kolmogorov equation

$$p(\mathbf{x}_k|\boldsymbol{z}_{1:k-1}) = \int_{\mathbb{R}^{n_x}} p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{k-1}|\boldsymbol{z}_{1:k-1}) d\mathbf{x}_{k-1}. \tag{15}$$

After the arrival of the measurement $\boldsymbol{z}_k$ at time k, the posterior state probability density function (pdf) can be updated via Bayes' rule

$$p(\mathbf{x}_k|\boldsymbol{z}_{1:k}) = \frac{p(\boldsymbol{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\boldsymbol{z}_{1:k-1})}{p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1})}, \tag{16}$$

where $p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1})$ is a normalising constant. The analytical solution to the above equations is very difficult and intractable. We utilise the Monte Carlo technique [10] which has proven to be very suitable and powerful for dealing with nonlinear system dynamics.

The Monte Carlo approach relies on a sample-based construction of these probability density functions. Multiple particles (samples) of the variables of interest are generated, each one associated with a weight characterising the quality of a specific particle. An estimate of the variable of interest is obtained by the weighted sum of particles. Two major stages can be distinguished : *prediction* and *update*. During the prediction each particle is modified according to the state model, including the addition of random noise in order to simulate the effect of the noise on the variable of interest. In the update stage, each particle's weight is re-evaluated based on the new sensor data. The *resampling* procedure is dealing with the elimination of particles with small weights and replicates the particles with higher weights.

4.1 A Particle Filter for Mobility Tracking

The developed particle filter (PF) is based on multiple models for the unknown acceleration $\boldsymbol{u}$. Denote with N the number of particles of the PF. A detailed scheme of the filter is given in *Table 1*.

Table 1. A particle filter for mobility tracking

Initialisation

1. $k = 0$, for $j = 1, \ldots, N$,
   generate samples $\{\boldsymbol{x}_0^{(j)} \sim p(\boldsymbol{x}_0), m_0^{(j)} \sim P_0(m)\}$,
   where $P_0(m)$ are the initial mode probabilities for the accelerations and set initial weights $W^{(j)} = 1/N$.

Prediction Step

2. For $k = 1, 2, \ldots,\quad j = 1, \ldots, N$,
   generate samples
   $\boldsymbol{x}_k^{(j)} = \boldsymbol{A}(T, \alpha)\boldsymbol{x}_{k-1}^{(j)} + \boldsymbol{B}(T)\boldsymbol{u}^{(j)}(m_k^{(j)}) + \boldsymbol{w}_k^{(j)}$,
   where $\boldsymbol{w}_k^{(j)} \sim \mathcal{N}(0, \boldsymbol{Q}^{(j)}(0, m_k^{(j)}))$,
   $m_k^{(j)} \sim \{\pi_{\ell m}\}_{\ell=1}^{M}$, $m = 1, \ldots, M$ for $\ell = m_{k-1}^{(j)}$;

Measurement Update: evaluate the importance weights

3. On the receipt of a new measurement, compute the weights

$$W_k^{(j)} = W_{k-1}^{(j)} \mathcal{L}(\boldsymbol{z}_k|\boldsymbol{x}_k^{(j)}). \tag{17}$$

   The likelihood $\mathcal{L}(\boldsymbol{z}_k|\boldsymbol{x}_k^{(j)})$ is calculated from (14) $\mathcal{L}(\boldsymbol{z}_k|\boldsymbol{x}_k^{(j)}) \sim \mathcal{N}(\boldsymbol{h}(\boldsymbol{x}_k^{(j)}), \sigma_v)$.

4. Normalise the weights, $\hat{W}_k^{(j)} = W_k^{(j)} / \sum_{j=1}^{N} W_k^{(j)}$.

   Output

5. The posterior mean $E[\boldsymbol{x}_k|\boldsymbol{z}_{1:k}]$

$$\hat{\boldsymbol{x}}_k = E[\boldsymbol{x}_k|\boldsymbol{z}_{1:k}] = \sum_{j=1}^{N} \hat{W}_k^{(j)} \boldsymbol{x}_k^{(j)}. \tag{18}$$

   Calculate posterior mode probabilities

6. $P(m_k = \ell|\boldsymbol{z}_{1:k}) = \sum_{j=1}^{N} 1(m_k^{(j)} = \ell)\hat{W}_k^{(j)}$,
   where $1(.)$ is an indicator function such that
   $1(m_k = \ell) = 1$, if $m_k = \ell$,
   and $1(m_k = \ell) = 0$ otherwise.

Compute the effective sample size

7. $N_{eff} = 1/\sum_{j=1}^{N}(\hat{W}_k^{(j)})^2$,

Selection step (resampling) if $N_{eff} < N_{thresh}$

8. Multiply/ suppress samples $\{\boldsymbol{x}_k^{(j)}, m_k^{(j)}\}$ with high/ low importance weights $\hat{W}_k^{(j)}$, in order to obtain N new random samples approximately distributed according to the posterior state distribution. The residual resampling algorithm [11, 12] is applied. This is a two step process making use of sampling-importance-resampling scheme.
   * For $i = 1, \ldots, N$, set $W_k^{(j)} = \hat{W}_k^{(j)} = 1/N$.

9. Set $k \rightarrow k + 1$ and return to step 2.

---

5 A Rao-Blackwellised Particle Filter for Mobility Tracking

A major drawback of the particle filtering over sensor networks is that it might become prohibitively expensive when a large number of particles is used. The mobility tracking algorithm has to possess such computational complexity that allows an on-line implementation. The complexity can be reduced and the estimation accuracy improved by a procedure called Rao-Blackwellisation [13, 14, 15, 16, 17, 18, 19].

Rao-Blackwellisation is a technique improving particle filtering by analytically marginalising out some of the variables (linear, Gaussian) from the joint posterior distribution, and then the linear part of the system model is estimated by a Kalman filter (KF), an optimal estimatior, whilst the nonlinear part is estimated by a PF. This leads to the fact that a KF is attached to each particle. In the mobility tracking problem the positions of the mobile unit are estimated with a PF, the speeds and accelerations with a KF. Since the measurement equation is highly nonlinear, the particle filter is used to approximate this distribution. After the estimation of the positions, these estimates are given to the KF as measurements. As a result of the marginalisation, the variance of the estimates is reduced compared to the standard PF.

Rather similar to the Rao-Blackwellisation approach is the mixture Kalman filtering approach proposed by Chen and Liu [20] where the system is represented by a linear conditional dynamic model and this way the problem is solved by multiple Kalman filters run with the Monte Carlo sampling approach. A formulation of the Rao-Blackwellisation problem is done also in [21, 22] in a more different way compared to [15].

The mobility model (1)-(2) is rewritten in the form

$$\begin{pmatrix} \boldsymbol{x}_k^{pf} \\ \boldsymbol{x}_k^{kf} \end{pmatrix} = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{A}^{pf} \\ \boldsymbol{0} & \boldsymbol{A}^{kf} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_{k-1}^{pf} \\ \boldsymbol{x}_{k-1}^{kf} \end{pmatrix} + \begin{pmatrix} \boldsymbol{B}_u^{pf} \\ \boldsymbol{B}_u^{kf} \end{pmatrix} \boldsymbol{u}_k + \begin{pmatrix} \boldsymbol{B}_w^{pf} \\ \boldsymbol{B}_w^{kf} \end{pmatrix} \boldsymbol{w}_k, \tag{19}$$

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k^{pf}) + \boldsymbol{v}_k, \tag{20}$$

where $\boldsymbol{x}^{pf}$ (pf short for particle filter) and $\boldsymbol{x}^{kf}$ (kf short for Kalman filter) is a partition of the state vector with $\boldsymbol{w}$ assumed Gaussian. Assume that equations (19)-(20) have the same properties like equations (1)-(2). Since the noise $\boldsymbol{w}_k$ is Gaussian,

$$\boldsymbol{w}_k = \begin{pmatrix} \boldsymbol{w}_k^{pf} \\ \boldsymbol{w}_k^{kf} \end{pmatrix} \in \mathcal{N} \sim (\boldsymbol{0}, \boldsymbol{Q}), \boldsymbol{Q} = \begin{pmatrix} \boldsymbol{Q}^{pf} & \boldsymbol{M}^{pf} \\ (\boldsymbol{M}^{pf})' & \boldsymbol{Q}^{kf} \end{pmatrix}. \tag{21}$$

Instead of directly estimating the pdf $p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k})$, with the entire state vector, consider the pdf $p(\boldsymbol{x}_k^{pf}, \boldsymbol{x}_k^{kf}|\boldsymbol{z}_{1:k})$. Using the Bayes rule, this pdf can be factorised into two parts according to

$$p(\boldsymbol{x}_k^{pf}, \boldsymbol{x}_k^{kf}|\boldsymbol{z}_{1:k}) = p(\boldsymbol{x}_k^{kf}|\boldsymbol{x}_k^{pf}, \boldsymbol{z}_{1:k})p(\boldsymbol{x}^{pf}|\boldsymbol{z}_{1:k}). \tag{22}$$

Since the measurements $\boldsymbol{z}_{1:k}$ are conditionally independent on $\boldsymbol{x}_k^{kf}$, the probability $p(\boldsymbol{x}_k^{kf}|\boldsymbol{x}_k^{pf}, \boldsymbol{z}_{1:k})$ can be written as

$$p(\boldsymbol{x}_k^{kf}|\boldsymbol{x}_k^{pf}, \boldsymbol{z}_{1:k}) = p(\boldsymbol{x}_k^{kf}|\boldsymbol{x}_k^{pf}). \tag{23}$$

Consider now the system

$$\begin{aligned} \boldsymbol{x}_k^{kf} &= \boldsymbol{A}^{kf}\boldsymbol{x}_{k-1}^{kf} + \boldsymbol{B}_u^{kf}\boldsymbol{u}_k + \boldsymbol{B}_w^{kf}\boldsymbol{w}_k^{kf}, \\ \boldsymbol{z}_k &= \boldsymbol{A}^{pf}\boldsymbol{x}_{k-1}^{kf} + \boldsymbol{B}_u^{pf}\boldsymbol{u}_k + \boldsymbol{B}_w^{pf}\boldsymbol{w}_k^{pf}, \end{aligned} \tag{24}$$

where $\boldsymbol{z}_k = \boldsymbol{x}_k^{pf} - f(\boldsymbol{x}_k^{pf})$. Since the system (24) is linear and Gaussian, the optimal solution is provided by the KF. We can assume a Gaussian form of the pdf (23), i.e.

$$p(\boldsymbol{x}_k^{kf}|\boldsymbol{x}_k^{pf}) \sim \mathcal{N}(\hat{\boldsymbol{x}}_{k|k-1}^{kf}, \boldsymbol{P}_{k|k-1}^{kf}), \tag{25}$$

where the estimate vector $\hat{\boldsymbol{x}}_{k|k-1}^{kf}$ and the corresponding covariance matrix $\boldsymbol{P}_{k|k-1}^{kf}$ are calculated by the Kalman filter.

The mobility model is a Singer-type model which accounts for correlations between the state vector components. Hence, we cannot assume that the process noise $\boldsymbol{w}^{pf}$ is uncorrelated with $\boldsymbol{w}^{kf}$, i.e. $\boldsymbol{M} \neq \boldsymbol{0}$.

The second pdf from (22) can be written recursively [14]

$$p(\boldsymbol{x}_k^{pf}|\boldsymbol{z}_{1:k}) = \frac{p(\boldsymbol{z}_k|\boldsymbol{x}_k^{pf})p(\boldsymbol{x}_k^{pf}|\boldsymbol{x}_{1:k-1}^{pf})}{p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1})}p(\boldsymbol{x}_{1:k-1}^{pf}|\boldsymbol{z}_{1:k-1}). \tag{26}$$

Due to the nonlinear measurement equation we apply a PF to solve (26). The weights are recursively calculated based on the likelihoods $p(\boldsymbol{z}_k|\boldsymbol{x}_k^{pf,(j)})$. The particles will be sampled according to $p(\boldsymbol{x}_k^{pf,(j)}|\boldsymbol{x}_{1:k-1}^{pf,(j)})$. Using the state equation for the $\boldsymbol{x}^{pf}$ from (19) and having in mind (25), the prediction step in the particle filter can be done as follows

$$\boldsymbol{x}_{k+1}^{pf,(j)} \sim \mathcal{N}(\boldsymbol{x}_k^{pf,(j)} + \boldsymbol{A}^{pf}\hat{\boldsymbol{x}}_{k|k-1}^{kf,(j)} + \boldsymbol{B}_u^{pf}\boldsymbol{u}_{k+1}^{(j)}, \\ \boldsymbol{A}^{pf}\boldsymbol{P}_{k|k-1}^{kf,(j)}(\boldsymbol{A}^{pf})' + \boldsymbol{B}_w^{pf}\boldsymbol{Q}^{pf,(j)}(\boldsymbol{B}_w^{pf})'. \quad (27)$$

For each particle, one Kalman filter estimates $\boldsymbol{x}_{k+1|k}^{kf,(j)}$, $j = 1,\dots,N$. It should be noted that the prediction of the nonlinear variables is used to improve the estimates of the linear state variables.

Table 2 presents the developed Rao-Blackwellised Particle Filter (RBPF).

Table 2. A Rao-Blackwellised PF for mobility tracking

Initialisation

1. $k = 0$, for $j = 1,\dots,N$, generate samples $\{\boldsymbol{x}_0^{pf,(j)} \sim p(\boldsymbol{x}_0^{pf}), m_0^{(j)} \sim P_0(m)\}$, where $P_0(m)$ are the initial mode probabilities for the accelerations. Initialise the Kalman filters by $\{\hat{\boldsymbol{x}}_{0|-1}^{kf,(j)} \sim \mathcal{N}\left(\hat{\boldsymbol{x}}_{0|-1}^{kf}, \boldsymbol{P}_{0|-1}^{kf}\right)\}$ and set initial weights $W_0^{(j)} = 1/N$.

Particle Filter Prediction Step

2. For $j = 1,\dots,N$,
   Predict the particles

$$\boldsymbol{x}_{k+1}^{pf,(j)} = \mathcal{N}(\boldsymbol{x}_k^{pf,(j)} + \boldsymbol{A}^{pf}\boldsymbol{x}_{k|k-1}^{kf,(j)} + \boldsymbol{B}_u^{pf}\boldsymbol{u}(m_{k+1}^{(j)}), \\ \boldsymbol{A}^{pf}\boldsymbol{P}_{k|k-1}^{kf}(\boldsymbol{A}^{pf})^T + \boldsymbol{B}_w^{pf}\boldsymbol{Q}_w^{pf}(\boldsymbol{B}_w^{pf})'), \quad (28)$$

   where $m_{k+1}^{(j)} \sim \{\pi_{\ell m}\}_{\ell=1}^M$ for $\ell = m_k^{(j)}$;

3. *Update step of the Kalman filters*

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}^{kf}(\boldsymbol{A}_k^{pf})'(\boldsymbol{S}_k)^{-1}, \quad (29)$$

$$\boldsymbol{P}_{k|k}^{kf} = \boldsymbol{P}_{k|k-1}^{kf} - \boldsymbol{K}_k\boldsymbol{A}^{pf}\boldsymbol{P}_{k|k-1}^{kf}, \quad (30)$$

$$\boldsymbol{S}_k = \boldsymbol{A}^{pf}\boldsymbol{P}_{k|k-1}^{kf}(\boldsymbol{A}^{pf})' + \boldsymbol{B}_w^{pf}\boldsymbol{Q}^{pf}(\boldsymbol{B}_w^{pf})', \quad (31)$$

   For $j = 1,2,\dots,N$

$$\boldsymbol{x}_{k|k}^{kf,(j)} = \boldsymbol{x}_{k|k-1}^{kf,(j)} + \boldsymbol{K}_k(\boldsymbol{z}_k^{(j)} - \boldsymbol{A}^{pf}\hat{\boldsymbol{x}}_{k|k-1}^{kf,(j)}), \quad (32)$$

   where $\boldsymbol{z}_k^{(j)} = \boldsymbol{x}_{k+1}^{pf,(j)} - \boldsymbol{x}_k^{pf,(j)}$.

4. *Prediction step of the Kalman filters*

$$\boldsymbol{P}_{k+1|k}^{kf} = \boldsymbol{D}\boldsymbol{P}_{k|k}^{kf}\boldsymbol{D}' + \boldsymbol{B}_w^{kf}\bar{\boldsymbol{Q}}^{kf}(\boldsymbol{B}_w^{kf})', \quad (33)$$

   where

$$\boldsymbol{C} = \boldsymbol{M}'(\boldsymbol{Q}^{pf})^{-1}, \quad (34)$$

$$\boldsymbol{D} = \boldsymbol{A}^{kf} - \boldsymbol{C}\boldsymbol{A}^{pf}, \quad (35)$$

$$\bar{\boldsymbol{Q}}^{kf} = \boldsymbol{Q}^{kf} - \boldsymbol{M}'(\boldsymbol{Q}^{pf})^{-1}\boldsymbol{M}, \quad (36)$$

   For $j = 1,2,\dots,N$

$$\boldsymbol{x}_{k+1|k}^{kf,(j)} = \boldsymbol{D}\hat{\boldsymbol{x}}_{k|k}^{kf,(j)} + \boldsymbol{C}\boldsymbol{z}_k^{(j)} + \boldsymbol{B}_u^{kf}\boldsymbol{u}(m_{k+1}^{(j)}). \quad (37)$$

Measurement Update: evaluate the importance weights

5. Compute the weights

$$W_{k+1}^{(j)} = W_k^{(j)}\mathcal{L}(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}^{pf,(j)}). \quad (38)$$

   The likelihood $\mathcal{L}(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}^{pf,(j)})$ is calculated from (14) $\mathcal{L}(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}^{pf,(j)}) \sim \mathcal{N}(\boldsymbol{h}(\boldsymbol{x}_{k+1}^{pf,(j)}), \sigma_v)$.

6. Normalise weights, $\hat{W}_{k+1}^{(j)} = W_{k+1}^{(j)} / \sum_{j=1}^N W_{k+1}^{(j)}$.
7. *Output*

$$\hat{\boldsymbol{x}}_{k+1}^{pf} \approx \sum_{i=1}^N \hat{W}_{k+1}^{(j)}\boldsymbol{x}_{k+1}^{pf,(j)}, \quad (39)$$

$$\hat{\boldsymbol{x}}_{k+1}^{kf} \approx \sum_{i=1}^N \hat{W}_{k+1}^{(j)}\hat{\boldsymbol{x}}_{k+1|k}^{kf,(j)}, \quad (40)$$

   Calculate posterior mode probabilities

8. $P(m_{k+1} = \ell|\boldsymbol{z}_{1:k+1}) = \sum_{j=1}^N 1(m_{k+1}^{(j)} = \ell)\hat{W}_{k+1}^{(j)}$,
   where $1(.)$ is an indicator function such that
   $1(m_{k+1} = \ell) = 1$, if $m_{k+1} = \ell$, and
   $1(m_{k+1} = \ell) = 0$ otherwise.

   Selection step (resampling)

9. If $N_{eff} < N_{thresh}$ resample $\{\boldsymbol{x}_{k+1}^{pf,(j)}, \boldsymbol{x}_{k+1/k}^{kf,(j)}, m_{k+1}^{(j)}\}$ in the same way as in the PF from Table 1.

10. Set $k \rightarrow k+1$ and return to step 2.

Note that in the KF update and prediction step the filter gain $\boldsymbol{K}_k$, the predicted and estimated covariance matrices, $\boldsymbol{P}_{k|k}^{kf}$, $\boldsymbol{P}_{k-1|k}^{kf}$ are calculated once, which affords to reduce the computational load.

5.1 Mobility Prediction

Based on the approximation of the filtering distribution $p(\mathbf{x}_k|\boldsymbol{z}_{1:k})$ we seek to estimate the r-step ahead prediction distribution ($r \geq 2$). In a general prediction problem we are interested in computing the posterior r-step ahead prediction distribution $p(\mathbf{x}_{k+r}|\boldsymbol{z}_{1:k})$ given by ([21, 6])

$$p(\mathbf{x}_{k+r}|\boldsymbol{z}_{1:k}) = \int_{\mathbb{R}^{n_x}} p(\mathbf{x}_k|\boldsymbol{z}_{1:k}) \left[\prod_{i=k+1}^{k+r} d\mathbf{x}_{k:k+r-1}\right], \quad (41)$$

where $\mathbf{x}_{k+r} = \{\boldsymbol{x}_k, \boldsymbol{u}_k, \dots, \boldsymbol{x}_{k+r}, \boldsymbol{u}_{k+r}\}$. The integrals in (41) can be evaluated using the evolution equation (3), resp. (19). Then the solution to the r step ahead prediction can be given by performing the steps from *Table 3*.

Table 3. r step ahead prediction

- For $i = 1, \dots, r$, For $j = 1, 2, \dots, N$, sample $\boldsymbol{x}_{k+i}^{(j)} = \boldsymbol{A}(T, \alpha)\boldsymbol{x}_{k+i-1}^{(j)} + \boldsymbol{B}(T)\boldsymbol{u}^{(j)}(m_{k+i}) + \boldsymbol{w}_{k+i}^{(j)}$, where $\boldsymbol{w}_{k+i}^{(j)} \sim \mathcal{N}(0, \boldsymbol{Q}^{(j)}(0, m_{k+i}^{(j)}))$, $m_{k+i}^{(j)} \sim \{\pi_{\ell m}\}_{\ell=1}^{M}$, $m = 1, \dots, M$ for $\ell = m_{k+i-1}^{(j)}$;

Then the predicted state estimate of the mobile unit is equal to

$$\hat{\boldsymbol{x}}_{k+r/k} = \sum_{j=1}^{N_{mc}} W_k^{(j)} \boldsymbol{x}_{k+r/k}^{(j)}. \tag{42}$$

6 Performance Evaluation

The developed Monte Carlo algorithms are evaluated over a conventional hexagon cellular network (similar to those in [6]). It is supposed that a map of the cellular network is available and the centre coordinates of the base stations are known.

7 Synthetic Data Example

The simulated service area contains 64 base stations with cell radius of 2 km, as shown in Fig. 2. The mobile can move to any cell of the network with varying speed and acceleration. The discrete-time command processes $u_{x,k}$ and $u_{y,k}$ can change within the range $[-7\ m/s^2, 7\ m/s^2]$. These commands $u_{x,k}$ and $u_{y,k}$ are assumed in the filters to be independent Markov processes, each of them taking values between the following acceleration levels

$$\mathcal{A}_x = \{0,\ 0,\ 1,\ -1,\ 0,\ 1,\ -1,\ 1,\ -1,\ 2.5,\ -2.5,\ 2.5,\ -2.5,\ 5,\ -5,\ -5,\ 5\}, \tag{43}$$

$$\mathcal{A}_y = \{0,\ 1,\ 0,\ 0,\ -1,\ 1,\ 1,\ -1,\ -1,\ 2.5,\ -2.5,\ -2.5,\ 2.5,\ 5,\ -5,\ 5,\ -5\}, \tag{44}$$

in units of $[m/s^2]$. The simulated trajectory of the mobile is generated according to the mobility model (3) and with this trajectory the RSSI signals are randomly generated according to the observation equation (14) with different noises for each simulation run. The randomness of the RSSI comes from the randomness of the shadowing component. At any sampling time, the observed RSSI signal is chosen to be the three largest signal powers among all 64 BSs in the network. The simulation parameters are summarised in *Table 4*. Additionally we take into account that the estimated speed $\hat{v}$ and acceleration $\hat{a}$ can not exceed certain physical limits, i.e. $\hat{v} = \sqrt{\hat{\dot{x}}^2 + \hat{\dot{y}}^2} \in [0, 45]\ [m/s]$, $\hat{a} = \sqrt{\hat{\ddot{x}}^2 + \hat{\ddot{y}}^2} \in [-5, 5]\ [m/s^2]$. The particles that are outside these intervals are eliminated (set to zero).

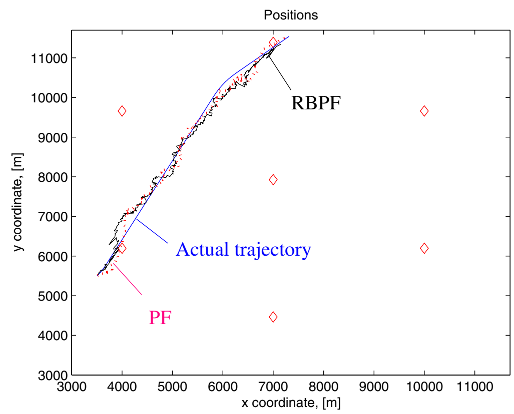

Fig. 2: Centres of the base stations, the actual trajectory of the mobility unit, estimated trajectories by the PF and RBPF from a single realisation

Table 4. Simulation parameters

Discretisation time step T	0.5s
Correlation coefficient α	0.95
Path loss index η	2
Base station transmission power $p_{0,i}$	0
Covariance σ_w^2 of the noise $\boldsymbol{w}_k$ in (3)	1.95 m/s^2
Transition probabilities $p_{i,i}$	0.1
Initial mode probability vector $\mu_{i,0}$	$[1/i],\ i = 1, \dots, 17$
Number of particles	$N = 2000$
Threshold for resampling	$N_{thresh} = N/10$
Number of Monte Carlo runs	$N_{mc} = 50$
Covariance σ_v^2 of the noise $v_{i,k}$	3 dB

After partitionning the state vector according to (19) within the RBPF scheme the respective matrices of the PF and KF are of the form

$$\boldsymbol{A}^{pf} = \begin{pmatrix} T & a & 0 & 0 \\ 0 & 0 & T & a \end{pmatrix}, \boldsymbol{B}_u^{pf} = \begin{pmatrix} c & 0 \\ 0 & c \end{pmatrix}, \boldsymbol{B}_w^{pf} = \boldsymbol{I}_2,$$

$$\boldsymbol{Q}^{pf} = \begin{pmatrix} q_{11} & 0 \\ 0 & q_{11} \end{pmatrix}, \boldsymbol{M} = \begin{pmatrix} q_{12} & q_{13} & 0 & 0 \\ 0 & 0 & q_{12} & q_{13} \end{pmatrix},$$

$$\boldsymbol{A}^{kf} = \begin{pmatrix} 1 & b & 0 & 0 \\ 0 & e^{-\alpha T} & 0 & 0 \\ 0 & 0 & 1 & b \\ 0 & 0 & 0 & e^{-\alpha T} \end{pmatrix}, \boldsymbol{B}_u^{kf} = \begin{pmatrix} \alpha a & 0 \\ \alpha b & 0 \\ 0 & \alpha a \\ 0 & \alpha b \end{pmatrix},$$

$$\boldsymbol{B}_w^{kf} = \boldsymbol{I}_4,$$

$$\boldsymbol{Q}^{kf} = \begin{pmatrix} q_{22} & q_{23} & 0 & 0 \\ q_{23} & q_{33} & 0 & 0 \\ 0 & 0 & q_{22} & q_{23} \\ 0 & 0 & q_{33} & q_{33} \end{pmatrix},$$

where $q_{ij},\ \ i, j = 1, 2, 3$ have the form (6)-(12).

The estimated and actual trajectories of the mobile unit over a single realisation are given in Fig. 2. Figs. 3-4 show

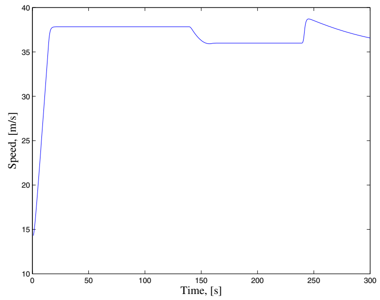

Fig. 3: Speed of the moving unit

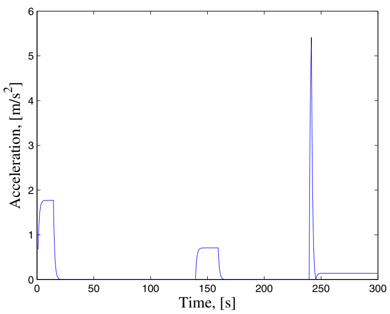

Fig. 4: Acceleration of the moving unit

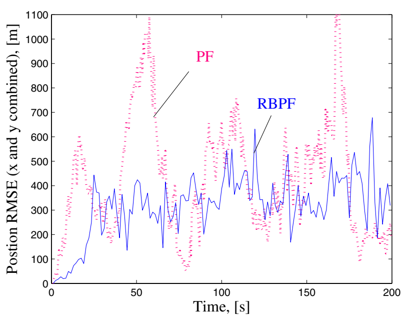

Fig. 5: RMSE of x and y positions combined

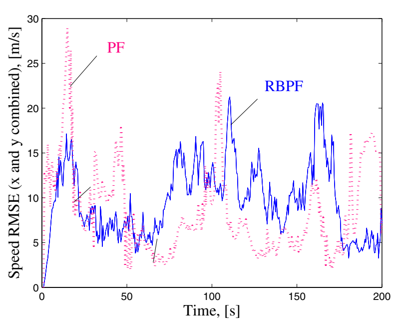

Fig. 6: RMSE of x and y speeds combined

respectively the speed and acceleration of the testing scenario. Short-time manoeuvers are followed by uniform motions.

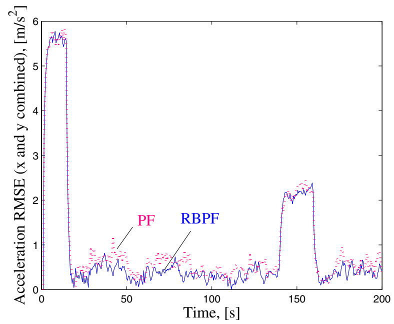

Fig. 7: RMSE of x and y accelerations combined

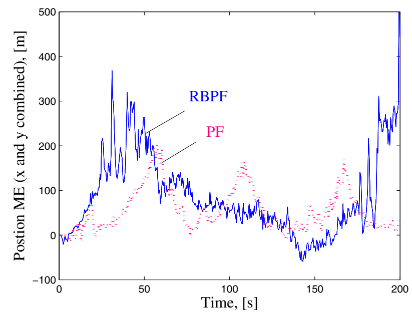

Fig. 8: ME of x and y positions combined

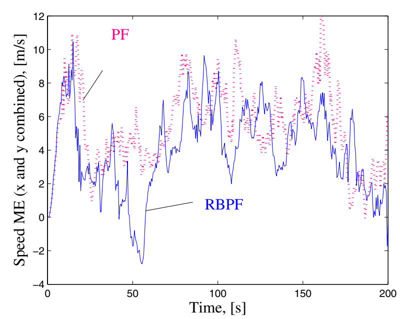

Fig. 9: Speed ME (x and y combined)

The root-mean-square error (RMSE) [23]

$$RMSE = \sqrt{\frac{1}{N_{mc}} \sum\nolimits_{m=1}^{N_{mc}} [(\hat{x}_k - x_k)^2 + (\hat{y}_k - y_k)^2]} \tag{45}$$

is used to assess the closeness of the estimated trajectory $\{\hat{x}_k, \hat{y}_k\}$ to a given trajectory $\{x_k, y_k\}$ over $N_{mc} = 50$ Monte Carlo runs. This position RMSE is presented in Fig. 5. RMSEs are calculated also for the estimated speed and acceleration components (Figs. 6-7). Both the PF and RBPF have shown a reliable tracking performance. The mean errors (MEs) combined in both coordinates are shown in Figs. 8-10. The results show that the computational time of the RBPF is reduced about 35% with respect to the PF.

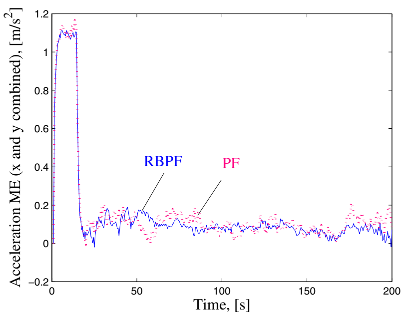

Fig. 10: Acceleration ME (x and y combined)

8 Conclusions

Two sequential Monte Carlo algorithms - a particle filter and a Rao-Blackwellised particle filter were developed. They can be used to make efficient mobility tracking and prediction in wireless networks. Their performance has been evaluated over a synthetic data example. The Rao-Blackwellised particle filter allows to decrease the computational complexity by reducing the number of required particles and shows slightly superior results to the particle filter in terms of accuracy.

Open issues for future research are investigation of the Monte Carlo framework for mobility tracking in ad hoc mobile wireless networks with different propagation models, different scenarios with single and multiple mobile units.

Acknowledgements

This research is supported by the the UK MOD Data and Information Fusion Defence Technology Center and in part by the Bulgarian Foundation for Scientific Investigations under grants I-1202/02 and I-1205/02.

References

[1] Z. R. Zaidi and B. L. Mark. Real-time mobility tracking algorithms for cellular networks based on Kalman filtering. *IEEE Trans. on Mobile Computing*, 2005.

[2] T. Liu, P. Bahl, and I. Chlamtac. Mobility modeling, location tracking, and trajectory prediction in cellular networks. *IEEE Journ. on Special Areas in Communications, Special Issue on Wireless Access Broadband Networks*, 16(6):922–936, Aug. 1998.

[3] B. Mark and Z. Zaidi. Robust mobility tracking for cellular networks. In *Proc. of the IEEE Intl. Communications Conf.*, pages 445–449, May 2002.

[4] H. Kobayashi, S.-Z. Yu, and B. Mark. An integrated mobility and traffic model for resource allocation in wireless networks. In *Proc. of the 3rd ACM Intl. Workshop on Wireless Mobile Multimedia*, pages 39–47. ACM Press, 2000.

[5] S.-Z. Yu and H. Kobayashi. A hidden semi-Markov model with missing data and multiple observation sequences for mobility tracking. *Signal Process.*, 83(2):235–250, 2003.

[6] Z. Yang and X. Wang. Joint mobility tracking and hard handoff in cellular networks via sequential Monte Carlo filtering. In *Proc. of the Conf. on Computer Communications, IEEE Infocom*, New York, June 2002.

[7] Z. R. Zaidi and B. L. Mark. Mobility estimation based on an autoregressive model. *IEEE Trans. on Vehicular Techn.*, 2005.

[8] R. Singer. Estimating optimal tracking filter performance for manned maneuvering targets. *IEEE Trans. on Aerosp. and Electr. Systems*, 6:473– 483, July 1970.

[9] X. R. Li and V. Jilkov. A survey of maneuveuvering target tracking. Part I: Dynamic models. *IEEE Trans. on Aerospace and Electr. Systems*, 39(4):1333–1364, 2003.

[10] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. on Signal Proc.*, 50(2):174–188, 2002.

[11] J. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

[12] E. Wan and R. van der Merwe. *Kalman Filtering and Neural Networks*, chapter 7. the Unscented Kalman filter 7, pages 221–280. Wiley Publishing, September 2001.

[13] G. Casella and C. Robert. Rao-Blackwellisation of sampling schemes. *Biometrica*, 83(1):84–94, 1996.

[14] P.-J. Nordlund and F. Gustafsson. Sequential Monte Carlo filtering techniques applied to integrated navigation systems. In *Proc. of the American Control Conf.*, pages 4375–4380, 2001.

[15] F. Gustafsson, F. Gunnarsson, N. Bergman, and U. Forssell. Particle filters for positioning, navigation and tracking. *IEEE Trans. on Signal Processing*, 50(2):425–437, 2002.

[16] P.-J. Nordlund. *Sequential Monte Carlo Filters and Integrated Navigation*. Ph.D. thesis, Linköping University, Sweeden, 2002.

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

[18] T. Schön, F. Gustafsson, and P.J. Nordlund. Marginalized particle filters for nonlinear state-space models. *IEEE Trans. on Signal Processing*, Nov. 2005.

[19] S. Maskell, M. Rollason, N. Gordon, and D. Salmond. Efficient particle filtering for multiple target tracking with application to tracking in structural images. *Image and Video Computing*, 21(10):931–939, 2003.

[20] R. Chen and J. Liu. Mixture Kalman filters. *J. Royal Stat. Soc. B*, 62:493–508, 2000.

[21] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[22] C. Andrieu and A. Doucet. Particle filtering for partially observed Gaussian state-space models. *J. of R. Statist. Soc. B.*, 64(4):827–836, 2002.

[23] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley and Sons, 2001.