Abstract - This paper addresses the problem of position localisation of mobile nodes in ad hoc wireless networks based on received signal strength indicator measurements. Node mobility is modelled as a linear system driven by a discrete command Markov process. Self-localisation of mobile nodes is performed via an Interacting Multiple Model filter consisting of a bank of Unscented Kalman filters (IMM-UKF). Estimation of the mobility state, which comprises the position, speed and acceleration of the mobile nodes is accomplished. The performance of the IMM-UKF filter is investigated and compared to a multiple model particle filter (MM PF) by Monte Carlo simulation.

Keywords: localisation, wireless networks, Monte Carlo methods, multiple model estimation, Unscented Kalman filter

1 Introduction

The movement patterns of mobile users play an important role in performance analysis of wireless computer and communication networks in which nodes may move freely within an area. The structure of ad hoc wireless networks is dynamically changeable over time which complicates the network control and management tasks. It is very important for this type of network to localise the node positions and movement [1, 2, 3], as the transmitter range is generally fairly small with respect to the size of the area. Self-localisation [4] involves the combination of absolute location information (e.g., obtained from a Global Positioning System (GPS)) with relative distance information (e.g. distance measurements between sensors) over regions of the network. It is also desirable to minimise the amount of inter-sensor communications.

There are many methods for self-localisation, one class of which is based on signal measurements and their statistical models (see, e.g., the surveys [1, 2, 5, 6]). These methods rely upon the signal time-of-arrivals, time difference of arrivals, angle-of-arrivals or received signal strengths and they vary in their complexity and accuracy. In this paper we consider the self-localisation of mobile nodes in wireless ad hoc networks using received signal strength indicator (RSSI) measurements. Node mobility is modelled with a linear system model, with multiple acceleration modes, which are driven by a discrete Markov process. Due to the fact that the control process of the mobile node is unknown and we have multiple acceleration levels, the Interacting Multiple Model approach [7] is suitable for the considered problem. We implemented it as a bank of Unscented Kalman filters, that affords avoiding linearisation of the highly nonlinear measurement equations. The IMM-UKF is compared with a multiple model particle filter for mobile nodes self-localisation. In [8] multiple model particle filtering techniques for mobility tracking of users in cellular networks are developed. This approach relies on prior information about the position of the base stations, which information is not available in ad hoc systems. Hence, the approach proposed in [8] cannot be directly used for ad hoc networks.

Previous approaches to mobile nodes localisation depend on the type of the ad hoc network [6]: indoor or outdoor. In the indoor sensor network the localisation can be performed with beacons (fixed or moving) or it is beacon free. In outdoor applications GPS systems are mainly used. Many localisation techniques rely on Kalman filtering [9, 10], Monte Carlo techniques [11], including nonparametric belief propagation [4] and knowledge of the connectivity between the nodes. Some works consider the case when nodes can communicate between each other which is not always possible because communications are energy-consuming.

Nevertheless that different algorithms for self-localisation have been proposed in the literature, this is an open and active research area, where many questions from theoretical and practical point of view are unsolved. The difficulties are coming from the change-
able network topology, need of communications between the nodes under limited resources (energy, bandwidth), necessity to work under noisy data and overcome losses.

Although mobile ad hoc networks have particularities making them unique, many of the models and some techniques applied to cellular networks can also be used in wireless networks. In a cellular network, the coordinates of the base stations are known and used for mobility unit tracking, while most of nodes in ad hoc networks are mobile. In the framework of ad hoc networks, only signal measurements including relative distances between nodes are available. In [9] local coordinate systems are assigned to each node and relative distances between the nodes are calculated. As noticed in [9], absolute position coordinates can be obtained by assuming knowledge of the coordinates of at least three mobile nodes.

The contributions of this paper are in the application of target tracking techniques to the problem of mobile nodes self-localisation. The localisation problem in ad hoc networks has particularities, making it more difficult than the typical target tracking problems.

This paper is structured as follows. Section 2 formulates the considered problem for self-localisation of mobile nodes. Section 3 describes the mobility and observation models. Section 4 presents the developed Interacting Multiple Model algorithm consisting of a bank of Unscented Kalman Filters (IMM-UKF) for self-localisation. A multiple model particle filter (MM PF) for mobile nodes self-localisation is designed in section 5. The performance of both filters is investigated in Section 6. Finally, Section 7 discusses the results and outlines open issues for future research.

2 Self-localisation of mobile nodes

We consider the two-dimensional (2-D) problem of cooperative self-localisation [2] of mobile nodes. The vector \( \mathbf{X} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \) of positions of \( n \) mobile nodes is estimated given \( r \) known reference coordinates \( \mathbf{X}^r = \{(x_{n+1}, y_{n+1}), (x_{n+2}, y_{n+2}), \ldots, (x_{n+r}, y_{n+r})\} \) and pair-wise measurements \( \{X_{i,j}\} \), where \( X_{i,j} \) is a measurement between devices \( i \) and \( j \). In this paper apart from the mobile nodes positions, we estimate their speeds and accelerations. We focus our attention on cases in which mobile sensor nodes receive distance measurements from a subset of reference sensor nodes in the network. This includes applications in which each sensor is equipped with a wireless transceiver and the distance is estimated by received signal strength indicators or time delay of arrival between sensor locations. We consider a sensor deployment architecture as described in [3], Figure 1. Received signal strength indicators are processed in a central information processor (CIP) through a low-power communication network. We allow unknown-location devices to make measurements with known-location references. In cooperative localisation, unknown-location devices might be allowed to make measurements with other unknown-location devices. The additional information gained from the measurements between pairs of unknown-location devices might enhance the accuracy and robustness of the localisation system. However, apart from the increased energy necessary for communications, the complexity of the localisation algorithms also increases. In this paper we limit our consideration to the case without communications between the mobile nodes.

3 Mobility and observation models

3.1 Mobility model for ad hoc networks

Different state mobility models were previously used in ad hoc and cellular wireless networks such as random walk and constant acceleration models [12] and Singer-type models [13, 14, 15, 1]. Any model suggested in the target tracking literature is also plausible for localisation, see, e.g., [1, 16]. In this paper we choose a discrete-time Singer-type model [17] because it captures correlated accelerations and allows for prediction of position, speed, and acceleration of mobile users. This higher-order model affords decreasing the estimation error, although mobility models that do not comprise the acceleration can be used too. Originally proposed by Singer [18] for tracking targets in military systems, the Singer model has served as a basis for developing many effective manoeuvre models with various applications (see [16] for a detailed survey), including for user mobility patterns. In the original Singer model there is no control process and the acceleration is considered as a random process, which has a time autocorrelation, whilst the Singer-type model from [14] includes a command process in explicit form. Yang and Wang consider a simpler form of this mobility model [17]. In our previous paper [8] we investigated the Singer-type model adopted for mobility tracking by [14]. However, we found that the model of [17] gives better results and at the same time is simpler. In this paper we adopt their model.
The mobility tracking model is based on the dynamic model [17] representing the dynamics of the mobile unit. The state of the moving mobile at time instant \( k \) is defined by the vector, \( x_k = (x_k, \dot{x}_k, y_k, \dot{y}_k) \) 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 space. The motion of the mobility user can be described by the equation

\[
x_k = A(T, \alpha)x_{k-1} + B_u(T)u_k + B_w(T)w_k,
\]

where \( u_k = (u_x, u_y, u_v, u_e) \) is a discrete-time command process, the respective matrices in (1) are of the form

\[
A(T, \alpha) = \begin{pmatrix} \dot{A} & 0_{3 \times 3} \\ 0_{3 \times 1} & \dot{A} \end{pmatrix}, B_i(T) = \begin{pmatrix} \dot{B}_i \\ 0_{3 \times 1} \end{pmatrix}, B_v = \begin{pmatrix} \dot{B}_v \end{pmatrix}.
\]

The subscript \( i \) in the matrix \( B(T) \) in (2) stands for \( u \) or \( w \) respectively. The random process \( w_k \) is a \( 2 \times 1 \) vector and \( T \) is the discretisation period. The parameter \( \alpha \) is the reciprocal of the manoeuvre time constant and thus depends on how long the manoeuvre lasts. Since \( u_k \) is a white noise, \( E[w_k w^T_{k+j}] = 0, \) for \( j \neq 0 \). The covariance matrix \( Q \) of \( w_k \) is \( Q = \sigma_v^2 I \), where \( I \) denotes the unit matrix and \( \sigma_v \) is the standard deviation.

The unknown command processes \( u_{x,k} \) and \( u_{y,k} \) are modelled as a first-order Markov chain that take values from a set of acceleration levels \( M_x \) and \( M_y \). The process \( u_k \) takes values from the set \( M = M_x \times M_y = \{m_1, \ldots, m_l\} \), with transition probabilities \( \pi_{ij} = P(u_k = m_j | u_{k-1} = m_i) \), \( i, j = 1, \ldots, l \) and initial probability distribution \( \mu_{i,0} = P(m_i | m) \) for modes \( m_i \in M \) such that \( \mu_{i,0} \geq 0 \) and \( \sum_{i=1}^l \mu_{i,0} = 1 \).

Consider a typical situation ([9], Figure 2) for ad hoc networks, where several nodes form a cluster within which one of the nodes is a reference node, with known coordinates (e.g., they can be obtained by a GPS). Two other reference nodes are needed to uniquely define the directions to the local coordinate system. In the particular ad hoc scenario depicted in Figure 2, the algorithm is localising \( n = 3 \) mobile nodes simultaneously. The state vector includes the states of three neighbour nodes \( N_1, N_2, \) and \( N_3 \), i.e.

\[
s_k = (x_{k,1}, x_{k,2}, x_{k,3})^T.
\]

### 3.2 Observation model

In a wireless ad hoc network, the distance between the mobile and a reference (reachable) node can be inferred from the received signal strength indicator (RSSI) or pilot signal of the node. The RSSI, measured in dB, received at the mobile node \( N_i \) from the node \( N_j \) with coordinates \( (x_{j,k}, y_{j,k}) \) at time \( k \) is given by [9]

\[
z_{ij,k} = \kappa_j - 10\gamma \log_{10}(d_{ij,k}) + v_{i,k},
\]

where \( \kappa_j \) is a constant determined by the transmitted power, wavelength, and gain of the node \( N_j \), \( \gamma \) is the slope index (typically \( \gamma = 2 \) for highways and \( \gamma = 4 \) for microcells in a city), \( v_{i,k} \) is the logarithm of the shadowing component which is found to be a zero mean, stationary Gaussian process with standard deviation \( \sigma_v \) (from 4-8 dB), and \( d_{ij,k} \) is the distance between the mobile nodes \( N_i \) and \( N_j \)

\[
d_{ij,k} = \sqrt{(x_{j,k} - x_{i,k})^2 + (y_{j,k} - y_{i,k})^2}.
\]

All nodes in the group send their pilot signal strength measurements (reference strength indicator signals) to the reference nodes. In the case, with \( n \) mobile nodes, and \( r \) fixed nodes, having known positions and without communications between the fixed nodes, the overall observation vector is then of the form

\[
z_k = \{z_{ij,k}\}_{i,j=1}^L,
\]

where \( L = n \times r \) is the number of measurements. In the developed solutions we do not consider communications between the mobile nodes, hence we exclude the measured distances between the mobile nodes. The vector form of the observation equation (5) is

\[
z_k = h(s_k) + v_k.
\]

where the noise \( v_k \) consists of the path loss and the shadowing component.

### 4 Interacting Multiple Model - Unscented Kalman filter (IMM-UKF) for localisation

This section presents a solution to the self-localisation problem of mobile nodes, based on an Interacting Multiple Model - Unscented Kalman Filter (IMM-UKF). The rationale for using the IMM approach is due to the fact that the command processes (the accelerations) of the mobile nodes are unknown, which brings the necessity of multiple models. Since the problem is with a highly nonlinear measurement model, the UKF is a suitable individual filter for the bank of filters. Below we describe the UKF as it is implemented for self-localisation. Since the command process in the update equation (1) is unknown, we assume a set of acceleration levels \( M_x \times M_y \) and \( l \) UKFs (described in Table 1) take part in the IMM-UKF algorithm, each with different level \( u_{i,k} \) of the command process.
The Unscented Kalman filter (UKF) relies on the unscented transformation [19, 20], a method for calculating the statistics of a random variable which undergoes a nonlinear transformation. Consider propagating the random state vector \( \mathbf{x} \) (with dimension \( n_x \)) of each mobile node through the nonlinear measurement function \( \mathbf{z} = h(\mathbf{x}) \). Assume that \( \mathbf{x} \) has mean \( \hat{\mathbf{x}} \) and covariance matrix \( \mathbf{P} \). To calculate the statistics of \( \mathbf{z} \), a matrix \( \mathbf{X} \) of \( 2n_x + 1 \) sigma points \( X_i \) is formed. These sigma points \( X_i \) are propagated through the time update equation. The predicted sigma points \( \mathbf{X}_i,k/k-1 \) propagated through the measurement function give the matrix \( \mathbf{Z}_k,k-1 \) of the transformed points. Finally, similarly to the Kalman filter, the Kalman gain \( \mathbf{K}_k \), the state estimate \( \hat{\mathbf{x}} \) and the corresponding covariance matrix \( \mathbf{P} \) are updated by (26)-(28). The UKF for localisation is given as Algorithm 1. Please note that each UKF is working with a separate level \( \mathbf{u} \) for the command processes \( j = 1, 2, \ldots, l \).

Algorithm 1. The Unscented Kalman filter for localisation

<table>
<thead>
<tr>
<th>I. Initialisation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \hat{x}_0 = E[\mathbf{x}_0] ), ( \mathbf{P}_0 = E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)'] ).</td>
</tr>
<tr>
<td>For ( k = 1, 2, \ldots ),</td>
</tr>
<tr>
<td>II. Calculate sigma points:</td>
</tr>
<tr>
<td>( \mathbf{X}<em>k,k-1 = [\hat{x}</em>{k-1}, \hat{x}<em>{k-1} + \gamma \sqrt{\mathbf{P}</em>{k-1}}, \hat{x}<em>{k-1} - \gamma \sqrt{\mathbf{P}</em>{k-1}}] ),</td>
</tr>
<tr>
<td>( \sqrt{\mathbf{P}_{k-1}} ) is a Cholesky factor,</td>
</tr>
<tr>
<td>( \gamma = \sqrt{n_x + \lambda}, \lambda = \alpha^2(n_x + \kappa) - n_x ),</td>
</tr>
<tr>
<td>( 1 \leq \alpha \leq 1 - \frac{1}{2n_x} ), ( \kappa = 3 - n_x )</td>
</tr>
<tr>
<td>III. Time update of the sigma points ( \mathbf{X}_i,k/k-1 ) using the state update equation (1):</td>
</tr>
<tr>
<td>( \mathbf{X}_i,k/k-1 = \mathbf{A}(T)\alpha \mathbf{X}_i,k-1 + \mathbf{B}(u)\mathbf{u}_k ),</td>
</tr>
<tr>
<td>( \hat{x}<em>{k,k-1} = \sum</em>{i=0}^{2n_x} W^{(m)}_{i} \mathbf{X}_i,k/k-1 ),</td>
</tr>
<tr>
<td>( P_{k,k-1} = Q + \sum_{i=0}^{2n_x} W^{(c)}_{i} [\mathbf{X}<em>i,k/k-1 - \hat{x}</em>{k,k-1}] [\mathbf{X}<em>i,k/k-1 - \hat{x}</em>{k,k-1}]' ),</td>
</tr>
<tr>
<td>( \mathbf{Z}_k,k-1 = h(\mathbf{X}_i,k/k-1, \mathbf{X}_i,k-1) ),</td>
</tr>
<tr>
<td>( \tilde{z}<em>{k,k-1} = \sum</em>{i=0}^{2n_x} W^{(m)}_{i} \mathbf{Z}_i,k/k-1 ),</td>
</tr>
<tr>
<td>where the measurement sigma points ( \mathbf{Z}_i,k/k-1 ) are calculated from the observation equation (5).</td>
</tr>
</tbody>
</table>

| IV. Measurement update equations: |
| \( P_{z_k,z_k} = R + \sum_{i=0}^{2n_x} W^{(c)}_{i} [\mathbf{Z}_i,k/k-1 - \tilde{z}_{k,k-1}] [\mathbf{Z}_i,k/k-1 - \tilde{z}_{k,k-1}]' \), |
| \( P_{z_k,x_k} = \sum_{i=0}^{2n_x} W^{(c)}_{i} [\mathbf{X}_i,k/k-1 - \hat{x}_{k,k-1}] [\mathbf{Z}_i,k/k-1 - \tilde{z}_{k,k-1}]' \), |
| \( \mathbf{K}_k = P_{z_k,x_k} P_{z_k,z_k}^{-1} \), |
| \( \hat{x}_{k,k} = \hat{x}_{k,k-1} + \mathbf{K}_k(\mathbf{z}_k - \tilde{z}_{k,k-1}) \), |
| \( P_{k,k} = (I - \mathbf{K}_k P_{z_k,x_k}) P_{k,k-1} \), |
| where the weights are: |
| \( W^{(m)}_0 = \lambda/(n_x + \lambda) \), |
| \( W^{(c)}_0 = \lambda/(n_x + \lambda) + (1 - \alpha^2 + \beta) \), |
| \( W^{(c)}_i = 1/2(n_x + \lambda), i = 1, \ldots, 2n_x \). |

5 Multiple model particle filtering for localisation

The aim of particle filtering is to evaluate the posterior probability density function (pdf) \( p(\mathbf{x}_k|\mathbf{Z}^k) \) of the state vector \( \mathbf{x}_k \in \mathbb{R}^{n_x} \), given a set \( \mathbf{Z}^k = \{ \mathbf{z}_{1:k} \} \) of sensor measurements up to time \( k \). The particle filtering approach relies on a particle-based construction to represent the state pdf. Multiple particles (samples) of the state are generated, each one associated with a weight which characterises 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 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 state. In the update stage, each particle’s weight is re-evaluated based on the new data. An inherent problem with particle filters is degeneracy, the case when only one particle has a significant weight. A resampling procedure helps to avoid degeneracy by eliminating particles with small weights and replicating the particles with larger weights.

Since the command process of the mobile nodes is unknown, this imposes the necessity of designing a multiple model particle filter for localisation. Given the set \( \mathbf{M} \) covering well the possible command values, the unknown commands are supposed to evolve as a first-order Markov chain. The mobile nodes can be localised by the MM PF, presented as Algorithm 2. Additionally it is taken into account in (10) that the speed of the mobile nodes cannot exceed certain limits, i.e. the maximum speed \( v_{\text{max}} \). The particles for the base state are generated according to the transition prior, by (9) and the particles of the mode states by (11). The PF relies on randomly generated samples, whereas the sigma points of the UKF are deterministically chosen so that they exhibit certain properties, e.g. have a given mean and covariance. The UKF is formulated for Gaussian distributions of the noises, whereas the PF has the advantage to work with arbitrary distributions.

Algorithm 2. A multiple model particle filter for localisation of mobile nodes in wireless networks

<table>
<thead>
<tr>
<th>I. Initialisation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k = 0 ), for ( i = 1, \ldots, N ), generate samples ( { \mathbf{x}_0^{(i)} \sim p(\mathbf{x}_0), \mathbf{m}_0^{(i)} \sim p(\mathbf{m}) } ),</td>
</tr>
<tr>
<td>where ( p(\mathbf{m}) ) are the initial mode probabilities for the accelerations and set initial weights ( W^{(i)}_0 = 1/N ).</td>
</tr>
</tbody>
</table>

II. For \( k = 1, 2, \ldots \),

1. Prediction Step
   For \( i = 1, \ldots, N \), generate samples
   \( \mathbf{x}_k^{(i)} = \mathbf{A}(T, \alpha)\mathbf{x}_{k-1}^{(i)} + \mathbf{B}_u(T)\mathbf{u}^{(i)} + \mathbf{B}_w(T)\mathbf{w}_k^{(i)} \),
   (9)
   where \( \mathbf{w}_k^{(i)} \sim \mathcal{N}(0, \mathbf{Q}) \), and under the speed constraint:
   \( \text{if } V = \sqrt{\mathbf{x}_k^{(i)}(2)^2 + \mathbf{x}_k^{(i)}(5)^2} > v_{\text{max}}, \)
   \( \alpha_c = \arctan(\mathbf{x}_k^{(i)}(5)/\mathbf{x}_k^{(i)}(2)), \)
   \( \mathbf{x}_k^{(i)}(2) = V_{\text{max}} \cos(\alpha_c), \)
   (10)
\[ x_k^{(i)}(5) = V_{\text{max}} \sin(\alpha_k), \]

end

\[ m_k^{(i)} \sim \{ \pi_m \}_{m=1}^M, \quad m = 1, \ldots, M \text{ for } \ell = m_{k-1}^{(i)} \] (11)

**Measurement Update:** evaluate the importance weights

2. for \( i = 1, \ldots, N \), on the receipt of a new measurement, compute the weights

\[ W_k^{(i)} = W_{k-1}^{(i)} \mathcal{L}(z_k|x_k^{(i)}). \] (12)

The likelihood \( \mathcal{L}(z_k|x_k^{(i)}) \) is calculated using (5)-(7)

\[ \mathcal{L}(z_k|x_k^{(i)}) \sim \mathcal{N}(h(x_k^{(i)}), \sigma_v). \]

3. for \( i = 1, \ldots, N \), normalise the weights,

\[ W_k^{(i)} = W_k^{(i)}/\sum_{j=1}^N W_k^{(j)}. \]

**Output**

4. The posterior mean

\[ \hat{x}_k = \sum_{i=1}^N W_k^{(i)} \hat{x}_k^{(i)}. \] (13)

**Calculate posterior mode probabilities**

5. for \( i = 1, \ldots, N \),

\[ P(m_k = \ell|Z^k) = \sum_{i=1}^N 1(m_k^{(i)} = \ell) \hat{W}_k^{(i)}, \]

where \( 1(.) \) is an indicator function such that

\[ 1(m_k = \ell) = 1, \text{ if } m_k = \ell, \]

\[ 1(m_k = \ell) = 0 \text{ otherwise}. \]

**Compute the effective sample size**

6. \( N_{\text{eff}} = 1/\sum_{i=1}^N (\hat{W}_k^{(i)})^2. \)

**Selection step (resampling)** if \( N_{\text{eff}} < N_{\text{thresh}} \)

7. Multiply/ suppress samples \( \{ x_k^{(i)}, m_k^{(i)} \} \) with high/low importance weights \( \hat{W}_k^{(i)} \), in order to obtain \( N \) new random samples approximately distributed according to the posterior state distribution. The residual resampling algorithm [21, 20] is applied. This is a two step process making use of sampling-importance-resampling scheme.

* For \( i = 1, \ldots, N \), set \( \hat{W}_k^{(i)} = \hat{W}_k^{(i)} = 1/N. \)

### 6 Performance evaluation

Similarly to the scenario presented in [2], we consider three mobile nodes with three reference nodes. Simulation results for localisation of three mobile nodes are given in Figures 3 and 4, obtained respectively with the IMM-UKF and the PF. The actual speeds of the mobile nodes are shown in Figure 5. Respective pilot signal strengths are used between the mobile nodes and reference nodes. Random mobile trajectories were generated with the dynamic state equation (1)-(3). Three IMM-UKF are run for localising each individual mobile node. Each IMM-UKF utilises measured RSSI, i.e. for mobile node 1 these are measurements with respect to the three fixed nodes. Also three MM particle filters are run under the same conditions. The discrete-time command processes \( u_{x,k} \) and \( u_{y,k} \) can change within the range \([-5 \text{ m/s}^2, 5 \text{ m/s}^2]\). The command process \( u_k \) in the filters is assumed to be a Markov chain, taking values between the following acceleration levels

\[ M = M_x \times M_y = \{(0,0), (3.5,0), (0,3.5), (0,-3.5), (-3.5,0)\}, \]

in units of \([\text{m/s}^2]\). The parameters of the algorithms are given in Table 1. The combined position root-

![Figure 3: Actual trajectories of the mobile nodes versus their estimated trajectories (with the IMM-UKF).](image)

![Figure 4: Actual trajectories of the mobile nodes versus their estimated trajectories (with the MM PF).](image)
The IMM-UKF combined RMSE for both position coordinates is shown in Figure 6. The respective RMSEs obtained with the MM PFs are given in Figures 7. The speed RMSE of the filters are presented respectively in Figures 8 and 9.

Table 1. Simulation parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discretisation time step $T$</td>
<td>0.1 [s]</td>
</tr>
<tr>
<td>Correlation coefficient $\alpha$</td>
<td>0.8</td>
</tr>
<tr>
<td>Path loss index $\eta$</td>
<td>3</td>
</tr>
<tr>
<td>Transmission power $\kappa_{ij}$</td>
<td>90</td>
</tr>
<tr>
<td>Covariance $\sigma_{w_i}^2$ of the noise $w_k$ in (1)</td>
<td>$0.5^2$ [m/s$^2$]</td>
</tr>
<tr>
<td>Maximum speed $V_{\text{max}}$</td>
<td>51 [m/s]</td>
</tr>
<tr>
<td>Initial mode probabilities $\mu_i, i = 1, \ldots, 5$</td>
<td>1/5</td>
</tr>
<tr>
<td>Number of particles of the PF $N$</td>
<td>500</td>
</tr>
<tr>
<td>Threshold for resampling $N_{\text{thresh}} = N/10$</td>
<td>$N_{\text{res}} = 100$</td>
</tr>
<tr>
<td>Covariance $\sigma_{v_i}^2$ of the noise $v_{i,k}$</td>
<td>[4$^2$] dB</td>
</tr>
</tbody>
</table>

The algorithms have shown similar speed estimation accuracy. The MM PF and IMM-UKF accuracy obtained for two of the mobile nodes is comparable, with respect to the third node, the MM PF accuracy is slightly smaller than those of the IMM-UKF. We have found that the MM PF is more sensitive to the maneuvers, and to the mutual positions of the fixed and mobile nodes than the IMM-UKF.

The computational complexity is another important issue that we investigated. The ratio between the computational time of the three MM PFs estimating the states of the three mobile nodes, each with $N = 500$ particles, and the computational time of the three IMM-UKF is 90:1. When the number of the particles is $N = 100$ the ratio is 19:1. So, for this localisation problem the IMM-UKF seems to be a more suitable approach than the MM PF especially because it is much faster than the MM PF.

In the case when the mobile node communicate with each other, i.e. when the distances between them are measured and used in the IMM-UKF algorithm for localisation the complexity grows considerably. The reason is that with the multiple levels of command inputs, many predicted distances between the nodes are avail-
able, and this complexity is difficult to handle. This poses the necessity to ‘fuse’ somehow the predicted distances for each mobile node with the different control levels (5 in our case). The MM PF has an advantage in this case. The problem with the increasing number of combinations due to many control levels, can be easily circumvented in the MM PF. An extended state vector can be formed comprising the states of all mobile nodes. The problem of excessive communications between the nodes is avoided in the PF MM. The price to pay is the increased computational time, but at least the problem is solvable in an easy way.

Current investigations are going on the calculation of the Cramér-Rao lower bound of the algorithms and with different mobility models. The Cramér-Rao lower bound gives a lower bound on the variance attainable by an unbiased estimator and is a valuable characteristic for the practice.

7 Conclusions

This paper proposes a solution to the self-localisation problem of mobile nodes in wireless sensor networks. An IMM-UKF and a particle filter are developed and their performance is investigated. The IMM-UKF is faster than the particle filter, and experiments with a relatively small number of particles (up to 500) have shown that the multiple model particle filter accuracy is comparable to the accuracy of the IMM-UKF for some nodes and slightly worse than those of the IMM-UKF for other mobile nodes. This different accuracy of the MM PF is due to its sensitivity to the mutual positions of the fixed and mobile nodes.

The technique developed can be applied in several applications, such as GPS-free position localisation of nodes in wireless networks, for localisation in moving vehicles and robots. The algorithms developed here can be useful in scenarios where the location information for the mobile nodes is supporting basic network functions.

Acknowledgements

This research is supported by the UK MOD Data and Information Fusion Defence Technology Centre and in part by the Bulgarian Foundation for Scientific Investigations, I-1202/02, I-1205/02 and Ml-1506/05.

References


