# Chapter XVI
# Experiences in Data Processing and Bayesian Filtering Applied to Localization and Tracking in Wireless Sensor Networks

**Junaid Ansari**
*RWTH Aachen University, Germany*

**Janne Riihijärvi**
*RWTH Aachen University, Germany*

**Petri Mähönen**
*RWTH Aachen University, Germany*

## ABSTRACT

*The authors discuss algorithms and solutions for signal processing and filtering for localization and tracking applications in Wireless Sensor Networks. Their focus is on the experiences gained from implementation and deployment of several such systems. In particular, they comment on the data processing solutions found appropriate for commonly used sensor types, and discuss at some length the use of Bayesian filtering for solving the tracking problem. They specifically recommend the use of particle filters as a flexible solution appropriate for tracking in non-linear systems with non-Gaussian measurement errors. They also discuss in detail the design of some of the indoor and outdoor tracking systems they have implemented, highlighting major design decisions and experiences gained from test deployments.*

## INTRODUCTION

In this chapter we focus on the practical aspects of implementing Wireless Sensor Network (WSN) based localization and tracking solutions. We discuss different design decisions, such as the types of signals used for localization and how to preprocess and filter the sensor readings before applying the complete localization and tracking algorithms. Mobility tracking and localization are multifaceted problems, which have been studied for a long time in different contexts. Many potential applications in the domain of WSNs require such capabilities. The need for mobility tracking is inherent in many surveillance, security and logistic applications. Hence, the development of robust and low-cost techniques has also high practical interest in the industrial context. Vast literature exists on these topics, and especially theoretical underpinnings are relatively well established by now. In this chapter we will focus on explaining some of the practical issues for engineers who are interested in implementing tracking solutions. In particular, we also introduce particle filters, which are often better suited than Kalman filters and their variants to the real-world situations where non-linearities and/or non-Gaussian errors are relatively commonly encountered.

We shall use the following terminology throughout this chapter. By *localization,* we mean the determination of the location of the object at a single time instant either in relative or in absolute coordinates. Sometimes this is performed using *ranging,* that is, by obtaining distance estimates to nodes or devices at known locations. In applications where localization in terms of a global coordinate system is necessary we assume the presence of, for example, differential GPS enabled anchor nodes, which then provide global coordinates for the sensor nodes which use local ranging or localization techniques. Finally, by *tracking* we mean the estimation of the trajectory of an object based on sequential measurements.

Due to space limitations, this chapter is naturally providing only a starting point for someone being interested in deploying wireless sensor based tracking systems. Our aim is to provide enough theoretical background and references to enable similar work by others, but not to provide a comprehensive survey of the field. Moreover, we have provided a number of examples and less emphasized practical lessons from our own work with *deployed experimental networks.* We have drawn our knowledge from a number of networks and deployments, and particularly from a large outdoor vehicular tracking network that was semi-commercially deployed with ca. 100 surveillance nodes, and from indoor tracking networks which have also been using inertial tracking methods. The outdoor network was designed for target tracking and surveillance in the context of SMAUG-project (Ansari *et al.* 2007a) which considered tactical purpose networks. The designed indoor network, which has also common components with the SMAUG-network, was designed for asset tracking and ubiquitous computing purposes, tracking assets and doctors in hospitals being a good example application. Hence, our selection of topics has been heavily influenced by practical projects and from our experience on what are the key issues to be highlighted and take into account when building real deployed systems.

The rest of this chapter is structured as follows: We begin by a short general discussion on major localization techniques followed by an overview of different signal types used in localization. We also discuss different sensor types available for detecting such signals. After these preliminaries, we discuss at some length different data processing and filtering solutions, specifically highlighting common application scenarios and implementation considerations. The techniques considered range from filtering that can be carried out on individual sensor nodes all the way to Bayesian filters for data fusion and for solving the tracking problem using individual localizations as inputs. We then give rather detailed accounts on some of the systems we have developed applying the principles outlined. We discuss both the systems for indoor localization (for asset tracking, localization of terminals or tracking users) as well as for tracking vehicles.

## TECHNIQUES FOR LOCALIZATION AND RANGING

Before going into the discussion on data processing and filtering, we shall briefly recall the basic techniques for ranging and localization. For further details of these techniques the reader is referred to the previous chapters of the present volume. Our focus here will be on approaches with relevance for the tracking systems discussed in-depth later in this chapter.

In general localization and ranging can be performed either actively or passively. In active schemes, the listener or the target object transmits a signal solely for the detection process. The listener associates this signal with the position of the object in a given coordinate space. On the other hand, passive localization is performed based on the already present signal in the environment. Active schemes provide more control and flexibility in positioning and generally result in higher accuracy than the passive schemes. The passive systems have their relevance in environments where the detection process is to be done without being noticed, for example in hostile environments or where the localization is to be performed without altering the already existing infrastructure. Examples of systems with such constraints include traffic monitoring and surveillance systems.

For ranging, the main techniques considered in the following are Time of Arrival (ToA), Time Difference of Arrival (TDoA), and techniques based on signal strengths. In Time of Arrival method, also known as Time-of-Flight (ToF), the time duration for a particular signal in propagating from a transmitter to a receiver is measured. Using the propagation speed and the traveling time, the distance between the transmitter and the receiver is computed. The main limitation of ToA is that it requires strict synchronization between the sender and the receiver clocks. TDoA removes this restriction by employing two different types of signals with different propagation speeds. Most commonly used combination of signals is the Radio Frequency (RF) and acoustic signal pair. If $t_d$ is the time between the arrival of a RF and an acoustic signal, then the Line-Of-Sight (LOS) distance *d* can be computed as

$$d = \frac{t_d}{\frac{1}{speed_{RF}} - \frac{1}{speed_{acoustic}}}.$$

TDoA requires two different kinds of transceivers and is applicable only to active localization systems. The primary limiting factors in ranging accuracy with TDoA are the accuracy of estimation of these delays, clock resolution of the receiver, and changes in signal propagation speeds caused by environmental effects. Examples of ToA-based schemes utilizing audible sound have been developed by Simon *et al.* (2004) and Kuckertz *et al.* (2007) whereas an example of a TDoA-based scheme has been given by Whitehouse and Culler (2003). However, in TDoA-based schemes, ultrasound is usually preferred because of its highly directional properties and due to the fact that it causes no irritation to human ears. The MIT Cricket system is the most famous example here (Priyantha, 2005). Since radio waves cannot travel in water, ultrasonic based positioning systems are very promising in underwater sensor networks as well. Classically, ultrasonic based localization and tracking has been used for over half a century in Sound Navigation and Ranging (SONAR) (Drumheller, 1987).

Techniques based on received signal strength form another important family of passive localization and ranging techniques. Radio waves are one of the popular methods used in signal strength based schemes as virtually all the devices communicating over wireless use radio waves. The distance may be computed from the received signal strength or techniques utilizing signal strength maps can be used for localization. For examples of systems based on these approaches see Bahl and Padmanabhan (2000) and Lorincz and Welsh (2007). The major limiting factor for the accuracy of ranging and localization systems based on radio waves is the complexity of the radio propagation environment. However, radio based signal strength methods are quite popular in sensor networks because no additional hardware is required for localization purpose as every sensor node is already equipped with a radio for communication.

A localization technique not directly relying on ranging is Angle of Arrival (AoA). Instead of distance estimates, triangulation is used to determine the position of the target object in the coordinate space based on the angle of the received signal using an antenna array. Since this method requires a calibrated antenna array, it is usually too expensive and complicated to be used in sensor networks. However, in Enhanced-911 system (Federal Communications Commission), the mobile phone uses RF time of flight and AoA of the signals from the mobile phone to the base station towers in order to find the mobile phone's location.

Another passive localization technique of importance especially in vehicular tracking is obtained by observing local strength of the magnetic field. A magnetic object or an object with significant permeability disturbs the Earth's magnetic lines of forces. This property is used in the magnetometers to estimate the direction and the "magnetic content" in the object to be localized. Typical magnetometers are based on flux-gate, magneto-resistivity and Hall Effect. These are able to detect a changing magnetic field caused by a moving ferromagnetic object, rotation of a magneto-sensitive object or due to a changing electric current. These characteristics can be employed passively in a variety of applications like traffic monitoring, border surveillance applications etc. Magnetic sensors are not only used in detecting the presence of ferromagnetic objects for proximity but also for distance ranging in highly calibrated systems (Arora *et al.*, 2006).

Finally, different optical and infrared (IR) sensors can be effectively used for localization purposes as well. Such techniques are especially appropriate for various surveillance applications and military usage. Both passive and active approaches are possible, depending on the requirements and the constraints of the scenario being considered.

## DATA PRE-PROCESSING

In the previous section we described the basic localization and ranging techniques together with their characteristic features and requirements relevant for our applications. We shall now briefly cover the main approaches to data processing on sensor nodes used in various parts of the signal processing chain. Again, our focus is on methods we have found most useful when implementing tracking solutions.

### Data Calibration

Sensor readings generally depend on environmental factors such as light intensity, temperature, humidity, etc. The dependencies can be linear as well as non-linear. The process of compensating these effects is known as calibration. Depending on the type of sensor and system requirements, calibration can take place in various stages of the system lifecycle. Often calibration is part of the assembly process, but especially if the environmental factors affect the performance significantly, online calibration becomes necessary. As an example, we consider the TDoA based distance ranging system using a pair of radio frequency and ultrasonic signals. Since the ultrasonic waves are significantly affected by temperature, necessary adjustments are required in the speed of the sound waves for accurate distance ranging. It is convenient if the dependency relationship can be approximated through a mathematical relationship as it allows a sensor node to simply perform a calculation to calibrate the data instead of maintaining large statistical tables, which may not be possible because of the limited available memory. For instance, the temperature dependency of the speed of ultrasonic signal can be expressed as

$$v_{ultrasound} = 331.4 + 0.6T_c \text{ [m/s]} ,$$

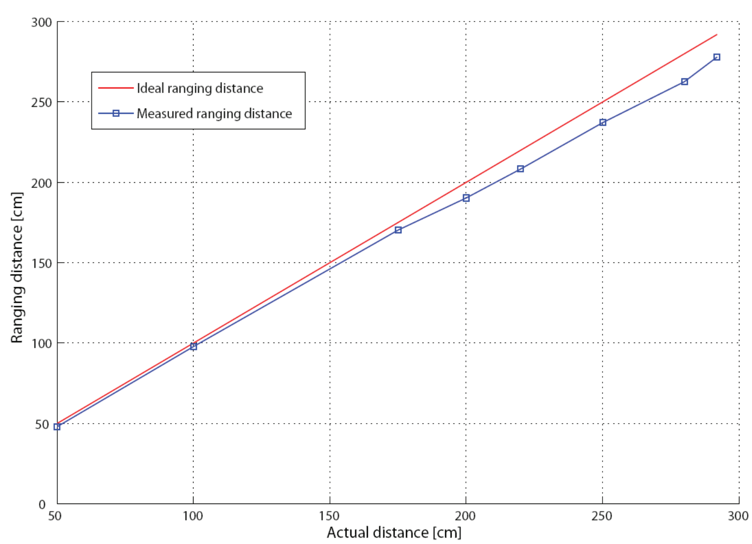
where $T_c$ is the temperature on Celsius scale.

Furthermore, as we will explain in the later sections, the ranging error grows over the real distance in the case of TDoA based ranging using the pair of radio and ultrasonic signals. In this case, appropriate re-adjustments in the distance estimates are performed, which are referred to as post-calibration.

## Compensation of Unwanted Constants and Long-Term Drift

In many situations, sensor readings show a constant signal level and the signal of interest overrides this constant level. In these situations, either a differential reading is useful which automatically nullifies the DC level or the constant offset needs to be subtracted explicitly. Occasionally it is more useful to subtract the constant offset before the digitization of the signal so that the constant offset does not reduce the dynamic range of the ADC. In many of the situations, the constant offset depends on the sensor surroundings or post deployment conditions for a sensor network. For instance, magnetometer readings always possess some DC offsets depending upon the ferromagnetic content in the surroundings or the readings from the infrared sensors have a constant offset because of the ambient temperature.

*Figure 1. The deviation of the measured ranging distance over real distance, obtained from Cricket System. A post-calibration scheme can be applied to compensate the distance dependence errors.*



All the magnetometer based compasses require some kind of offset compensation and calibration. A good practice in sensor board design is to include a signal conditioning circuitry that can dynamically compensate the environmental effects. This can be achieved by introducing software controllable circuit elements. A common approach is to use a digital potentiometer based voltage divider to generate similar offset at an input of a differential amplifier so that the resulting static differential signal becomes zero. This differential signal is then amplified.

In many sensors, a drift is developed over long periods of time. This is due to the changes in environmental conditions or the degradation of sensors. Changes in environmental conditions may, for instance, include the day and night effects, temperature and humidity variations and pressure differences, the changes in the Earth's magnetic field strength due to its rotation etc. Long-term drifts restrict the capabilities of sensors completely or partially and need to be eliminated. In an outdoor vehicle tracking scenario, using the Honeywell's HMC1022 magnetometer sensors (Honeywell, 2008a) it was observed that over periods of eight hours, the sensors gradually develop a high enough drift that the output signal starts to saturate. This drift can be compensated by using an exponentially moving average filter (Ansari *et al.*, 2007). The coefficient of the filter needs to be set according to the dynamic characteristics of the signal.

Finally, it is important to note that certain sensors develop anomalies over extended periods of time and lose their sensitivity. Some of the effects are due to the mechanical characteristics and others are due to the intrinsic measuring property. For example, the anisotropic magnetometer sensors have magneto-resistive elements that remain identical in the absence of magnetic field and get aligned in the presence of an external magnetic field. When the sensor is exposed to strong external magnetic field over extended periods of time, the sensor gets magnetized and loses its sensitivity. This can be de-magnetized using a high current pulse, using a circuitry as described in (Honeywell, 2008a).

## Filtering of Data

In many cases, the data obtained from the sensors has more information content than desired. This unwanted signal is regarded as noise and is normally filtered out before amplification and digitization. The choice of the filter is highly related to the type of noise in the sensor and the signal characteristics like bandwidth, variance, amplitude, phase etc. Filter selection is also dependent upon the degree of the required accuracy, the computational ability and the available memory at a sensor node. There is a filtering delay associated with each type of filter, which is generally dependent upon the computational complexity and the filter taps, i.e., number of samples required to generate an output. In this context, iterative filtering algorithms are very attractive because of their easy realization with low computational overhead, especially in resource constrained embedded devices like sensor nodes. In the following, we describe some of the commonly used filters in sensor networks and highlight the noise types the filters are effective in dealing with.

In many cases, the sensor readings contain spurious noise samples, which are uncorrelated with their adjacent samples. These noise samples can easily be filtered out using *median filters*. For instance, Cricket system results in some spurious distance estimates bearing no relationship with their neighbours. An effective way to get rid of the unwanted samples is by applying a moving median filter (Priyantha, 2005). *Lowpass filtering* is a popular smoothing technique which can be used to attenuate the noise inherent in many types of waveforms in the measurement dataset. It lets through the lower frequencies and attenuates the higher frequencies. The cut-off frequency is chosen to be compatible with the sampling rate of the ADC and the desired band of frequencies of measured signal. Lowpass filters can be categorized into two main classes based on how they operate on their impulse response:

- **Finite impulse response (FIR):** An FIR filter is usually implemented by using a series of delays, multipliers, and adders to create the filter's output. FIR filters do not use any feedback and are generally easy to implement.
- **Infinite impulse response (IIR):** An IIR filter uses feedback to keep more historical information active in the calculation. Its impulse response is infinite in duration. IIR filter might not be stable compared to FIR filters due to the feedback.

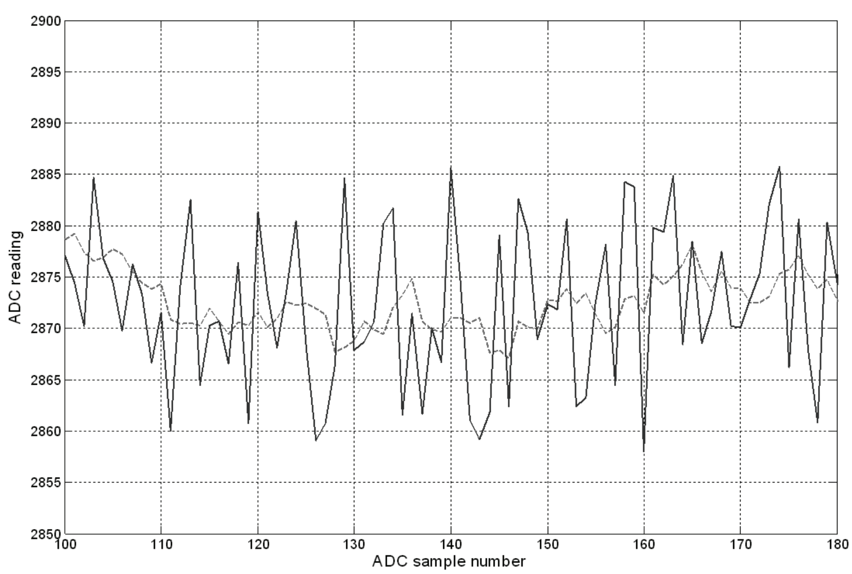
Owing to the limited memory and computational constraints, the filter coefficients used on the sensor nodes should be integer numbers and the order/complexity of the filter should be low. The operation of an 8-tap FIR filter is illustrated in the Figure 2, where the continuous line is the original measurement and the dashed line is the smoothed data.

A *moving average filter* is used to smooth out fluctuations in the data. It is implemented simply by calculating the sum of the measurements over a time window divided by the number of samples within the window. The *Exponentially Weighted Moving Average (EWMA)* approximates an arithmetic moving average by shifting the current estimate for the average by a constant multiple of the latest measurement. This can be expressed as

$$y \leftarrow \alpha z + (1 - \alpha) y,$$

where $z$ is the new measurement, $y$ is the variable containing the approximation to the average, and $\alpha$ is a parameter between zero and one. Note that every past value of $z$ in the time series is contained in

*Figure 2. Results of applying low pass filtering. The blue curve (solid line) represents the raw data sampled from the ADC and the red curve (dashed line) shows the smoothened version after applying a low-pass-filter.*



each new result $y$, but older measurement values get exponentially weighted to insignificance as the series progresses. The advantage of EWMA filter over the regular moving average one is obviously that memory for storing only a single variable is needed.

## BAYESIAN FILTERING AND DATA FUSION

The techniques presented in the previous section are absolutely necessary for obtaining accurate individual samples from sensors of various types. We shall now focus on techniques which can be utilized to estimate the behavior of a *system* being measured based on individual localization or ranging samples. We begin with a rather general introduction to sequential Bayesian filtering as a probabilistic tool to reason about the time-evolution of the system. We focus specifically on *particle filters* due to their wide range of applications, and capability to cope with non-linearities and non-Gaussian measurement noises. Our experience with practical tracking systems has led to the conclusion that there are many situations where non-linearities and/or non-Gaussian errors are quite significant. Later on we shall discuss on the applications of these principles for various localization and tracking systems. For further details, we refer the reader to Arulampalam *et al.* (2002) and Van Trees and Bell (2007), on which the following exposition is also based. Another insightful survey on Bayesian filtering in location estimation is given in Fox *et al*. (2003).

### Basic Framework

We denote the state of the system under study at time $t_k$ by $x_k$, where $k$ indexes the time instances at which either measurements become available or prediction of the system state is desired. In general $x$ can be an arbitrary vector, although in localization and tracking applications it is almost invariably the location of the object being tracked (or a Cartesian product of multiple location vectors in case of multi-object localization and tracking is performed). We focus only on the discrete-time case since this seems to cover all the major applications in the present field. We assume that a *system model* of the form

$$x_t = f(x_{t-1}, s_{t-1})$$

is known, where $f$ is a function of the state vector and the so-called *process noise* $s$. In general $f$ can be a non-linear function of its arguments, although the linear case is obviously of interest as well. We further assume that *measurements* on the system behavior are given by

$$z_k = h(x_k, n_k),$$

where $h$ can again be non-linear, and $n$ is the model for *measurement noise*. The noise processes are usually assumed to be independent and identically distributed, and we shall do so here as well.

### Bayesian Solution

We shall now apply Bayesian approach to the system and measurement models to obtain a recursive set of equations as a general solution to the state estimation problem. We assume that the initial state of the system is known in the form of a probability density $p(x_0)$. Our interest is on estimating the posterior probability of the state of the system given all the measurement information, that is, obtaining a law for $p(x_k \mid z_{1:k})$, where the shorthand $z_{1:k} = \{z_i \mid i = 1,\dots,k\}$ is used.

The usual way of writing down the solution for the Bayesian filtering problem is by means of two steps performed at each update time: *prediction* and *update*. In the prediction phase, the so-called Chapman-Kolmogorov equation is used to write the probability density function in the form

$$p(x_k \mid z_{1:k-1}) = \int p(x_k \mid x_{k-1}) p(x_{k-1} \mid z_{1:k-1}) dx_{k-1}.$$

The first term of the integrand is known from the system model and the second term is known from the previous time step. As the $k^{\text{th}}$ measurement is performed, the estimate of the state is updated by applying the Bayes rule, namely by computing

$$p(x_k \mid z_{1:k}) = \frac{p(z_k \mid x_k) p(x_k \mid z_{1:k-1})}{p(z_k \mid z_{1:k-1})}.$$

The likelihood function for the measurements is obtained from the measurement model, and the normalizing constant can be calculated from

$$p(z_k \mid z_{1:k-1}) = \int p(z_k \mid x_k) p(x_k \mid z_{1:k-1}) dx_k.$$

The above equations give the optimal Bayesian solution to the filtering problem in a form of recursive formulae. Unfortunately, the exact computation of the above integrals is not feasible in the most general case. We shall outline two approaches that have been found especially useful in practice. First is the simplification of the above equations by assuming simple form of system and measurement models. This leads to the famous *Kalman filter* and its variants. The second approach is driven by Monte Carlo techniques. The probability densities in the above equations can be approximated by a large sum of delta functions ("particles") turning the integrals into finite sums. This leads to very powerful approximate solutions of the filtering problem by means of *particle filters*.

## Kalman Filter and its Major Variants

The simplest special case of the filtering problem is obtained by assuming that both the system model and the measurement model are linear functions, and that both noise models in the system description are Gaussian. The system model thus becomes

$$x_k = F_k x_{k-1} + s_{k-1},$$

where *F* is now a matrix. Similarly, the measurement model becomes

$$z_k = H_k x_k + n_k.$$

Let now $N(x \mid m, Q)$ denote the Gaussian density with mean *m* and covariance matrix *Q*. It can be shown that in this case the Bayesian filtering equations are reduced to

$$p(x_{k-1} \mid z_{1:k-1}) = N(x_{k-1} \mid m_{k-1|k-1}, P_{k-1|k-1}),$$
$$p(x_k \mid z_{1:k-1}) = N(x_k \mid m_{k|k-1}, P_{k|k-1}),$$
$$p(x_k \mid z_{1:k}) = N(x_k \mid m_{k|k}, P_{k|k}),$$ and

where

$$m_{k|k-1} = F_k m_{k-1|k-1}$$
$$P_{k|k-1} = Q_{k-1} + F_k P_{k-1|k-1} F_k^T$$
$$m_{k|k} = m_{k|k-1} + K_k (z_k - H_k m_{k|k-1})$$
$$P_{k|k} = P_{k|k-1} - K_k H_k P_{k|k-1}$$
$$S_k = H_k P_{k|k-1} H_k^T + R_k$$
$$K_k = P_{k|k-1} H_k^T S_k^{-1}$$

and $Q$ and $R$ denote the covariances of the process and measurement noises, respectively. These are the equations defining the Kalman filter, yielding the optimal solution in the case of linear system with Gaussian noises. These equations can be used to approximate the non-linear case as well by linearization (that is, by approximating the system and measurement models by their Taylor series and discarding the non-linear terms). This results in the *Extended Kalman filter* (EKF). The accuracy of the results obtained from EKF depends heavily on the approximation error made, and the structure of the true shape of the posterior density. As the Kalman filter always results in Gaussian posterior, even the EKF will have limited validity in case of highly skewed or multimodal true posteriors. Another limitation of Kalman filter is the assumption of Gaussian process and measurement noises. As we have seen above the measurement noise for many real-world sensors is non-Gaussian, calling for more advanced solutions to the filtering problem.

## Particle Filters

The key observation needed in the following is that any probability density can be approximated by a sum of delta-functions of the form

$$p(x) \approx \sum_{i=1}^{N} w^i \delta(x - x^i),$$

where $\{x^i\}$ are called *particles* and $\{w^i\}$ are *weights* summing to one. The accuracy of the approximation is mainly dependent on the shape of $p$ and the total number of particles $N$. The weights are introduced to facilitate sampling of the points $\{x^i\}$. Usually one chooses an *importance density q* for which generating samples is easy and makes weights proportional to the ratio of the true distribution and $q$. In the case of Bayesian filtering using the above representation in the recursive formulation for the density

$$p(x_k \mid z_{1:k}) \approx \sum_{i=1}^{N} w_k^i \delta(x_k - x_k^i)$$

leads to the update rule

$$w_k^i \propto w_{k-1}^i \frac{p(z_k \mid x_k^i) p(x_k^i \mid x_{k-1}^i)}{q(x_k^i \mid x_{k-1}^i, z_k)}$$

for the weights, where $q$ is a suitably chosen importance density (we shall return to the question of choosing $q$ momentarily). This approach, called *Sequential Importance Sampling* (SIS) can be chosen to lead to the exact solution to the filtering equations in the limit of large $N$.

The SIS algorithm is relatively simple to implement, and has turned out to be practical in a number of applications. It does have, however, certain shortcomings the awareness of which is important. First of these is the so-called *degeneracy problem* manifesting itself in the majority of particles having weights approaching zero. This can be solved by *resampling*, that is, generating a new collection of particles and weights by sampling from the particle cloud approximation of the posterior distribution with weights giving the probabilities of the individual particles occurring in the sample.

Let us now briefly discuss the selection of the importance density $q$. The simplest choice would be to put

$$q(x_k \mid x_{k-1}^i, z_k) = p(x_k \mid x_{k-1}^i),$$

that is, choosing the importance density to simply be the prior density. The update step for weights is then obviously reduced to multiplication by the likelihood of the measurements, which is not only simple conceptually, but also implementation-wise. The drawback of this choice is that if the prior is highly concentrated, most of the particles will obtain very small weights and the system becomes rapidly degenerate. It also does not take into account the latest measurement, potentially accentuating the degeneracy problem. The remedies for this degeneracy are either substantial increase in the number of particles used, or selection of a more appropriate importance density. Choice between these alternatives has to be ultimately made based on the particulars of the problem at hand, and the computational resources available. For a survey and comparison of different importance densities, see Simandl and Straka (2007).

Numerous variants of the basic particle filter concept discussed above have been presented in the literature. For an overview of some of the most promising of these, see Arulampalam *et al.* (2002). However, for the applications in WSN-based tracking systems we have found the presented solutions quite satisfactory. In the following sections we discuss in more detail two applications of particle filters, namely on improving accuracy of Cricket-type TDoA systems as well as performing data fusion in vehicular tracking. For examples of use of particle filters in solving tracking problems in other systems (such as Ad Hoc and cellular networks), see, for example, Mihaylova *et al.* (2007), Gustafsson *et al.* (2002), Thrun (2002) and Olama *et al.* (2006). Relevant techniques from robotics research are given in Howard (2006). For an earlier application of Bayesian filtering to Cricket-system using Extended Kalman Filters see Smith *et al.* (2004).

## INDOOR SYSTEMS FOR LOCALIZATION

Many WSN-based localization systems have been developed for indoor applications as well as for outdoor scenarios. These use one of the basic principles for localization as explained in the previous sections and have their own application-specific design requirements in terms of functionality, scalability, performance and hardware. Representative examples for indoor localization systems include IR based systems like Active Badge (Want *et al.*, 1992), radio signal based systems like MoteTrack (Lorincz and Welsh, 2007), ultrasonic based systems like Active Bat (Ward, Jones and Hopper, 1997) and combination of radio and sonic waves like Cricket system (Priyantha, 2005). Active Badge uses pulse width modulated IR signals and can give an accuracy of 6m. Although, IR signals inherently suffer from ambient interferences, Versus Information Systems Ltd. developed commercial tags for locating doctors in hospitals and medical centers. Active Bat sender-tags transmit an ultrasonic pulse, which is received by a mesh of receivers mounted on the walls and ceilings in order to perform distance ranging. Trilateration is applied on the distance estimates to obtain position of the transmitting tag. MoteTrack uses radio signal signature to estimate the position of an object. It first requires establishing a calibrated signal signature model of the environment, which is obtained by measurements at a number of nodes from known geographical points. In the following we present Cricket System (Priyantha, 2005) in detail as a practical example of an indoor localization system. In the later sub-sections, we would discuss how improved accuracy can be obtained from Cricket System by applying Bayesian filtering framework and by complementing it with inertial sensors.

## Cricket System

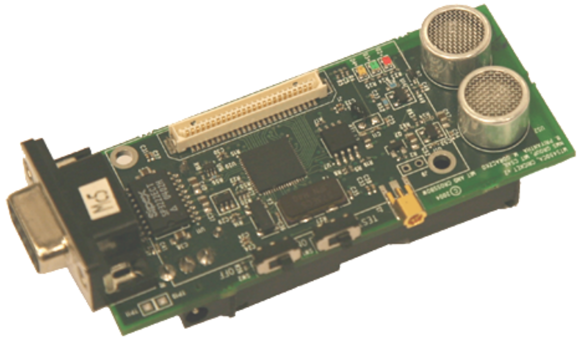
Cricket system is an indoor localization system, which uses inexpensive wireless sensor nodes (c.f. Figure 3) mounted at known positions. These nodes act as active beacons and transmit a pair of RF and ultrasonic signals. A Cricket node mounted on the target object, receives these signals from various beacons and applies TDoA based distance ranging. Since the atmospheric temperature has a significant influence to the speed of sound, the distance estimates obtained through the TDoA scheme must be calibrated to the ambient temperature conditions as described in the earlier section on Data Calibration. These distances are then used to find the spatial position of the object using lateration. Cricket system is able to determine the location of the target object within a few centimeters of accuracy.

## Limitations of the Cricket System

Let us look in detail what are the different constraints posed by the Cricket system:

- Cricket nodes work correctly only if a LOS path between a listener and a beacon node exists. In many sparsely deployed cases, this results in localization blind spots.
- Cricket nodes have a limited range of approximately 11m. The range depends upon the sensitivity of the ultrasonic receiver module, the detection threshold and most importantly on the power level of the transmitted ultrasonic pulse.
- Cricket distance ranging error increases as the actual distance between the listener and the beacon node is increased. The error also grows-up if the angle between the faces of listener and the beacon nodes increases. This has to do with the radiation characteristics of the ultrasonic transmitter and receiver. Beyond certain angles at a particular distance, Cricket nodes cannot compute distance estimates. The coverage over higher angles is limited to smaller distances. Please refer to Figure 4 for distance ranging dependency of the Cricket system on the actual distance and the angle between a transmitting and receiving node.

*Figure 3. Cricket node*

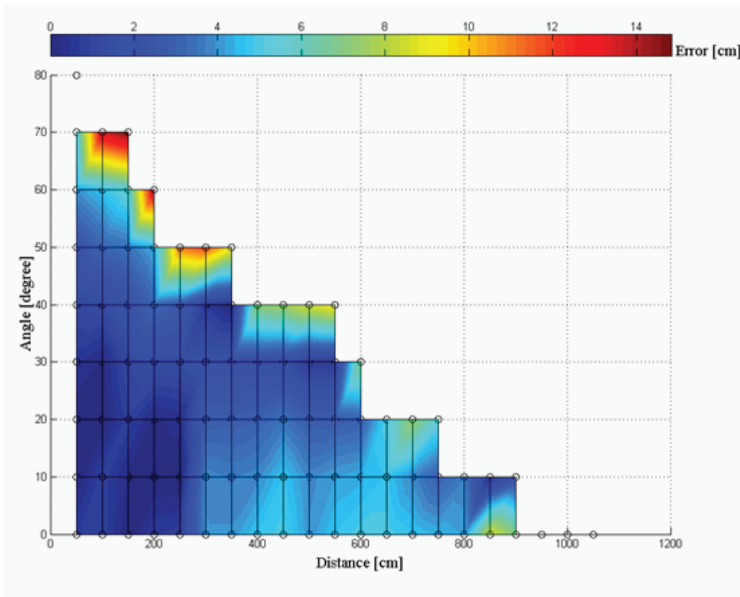


The localization performance of the Cricket system is dominated by the performance of the deployed ultrasonic sensors and their directivity and gain characteristics. When beaconing and listening nodes are faced directly to each other, the system can work properly up to 10 or 11 m. However, if the angle between beaconing and listening nodes increases, the measurements get worse and worse before it ceases to work at all. The larger the angle, the smaller is the operating range of the Cricket system. Near this limit of the maximum operating range, the measurements also get less stable and less accurate. However, for shorter distances, the system provides accurate and reliable distance measurements.

The system design includes various types of delays besides the signal propagation time, which limits the number of beacons transmitted per second and so the possible position updates per second. These delays are inserted in order to avoid radio and ultrasonic mutual interferences from different nodes. The constraints on the maximum number of beacon signals that can be transmitted per second are mostly imposed by the initial delay that has been put to let any stray ultrasonic pulse die down before the transmission of the beacon signal.

At a particular time instant, there needs to be at least four distance estimates to uniquely compute the position of the object in 3-dimentional space. From the ranged distance alone, the listener node is unable to determine the angle to the target Cricket node and therefore cannot calibrate it. A set of Cricket nodes may first apply lateration to determine the position of the target node as well as the angles from each listener node to the target node. Depending upon the angles, post-calibration can be applied on the ranged distances and localization can be performed again. The post-calibration process attempts to compensate the ranging anomalies due to the varying strengths of the ultrasonic transmission pattern at different real distances and angles. Despite the computational effort induced by the process, the post-calibration mechanism can in practice be iterated a few times to achieve higher accuracies. We also enhanced the Cricket system by filtering out non-coherent and spurious distance estimates known as outliers by using a moving median filter (c.f. Filtering of Data above). We found out that besides the low computational complexity, the choice of a moving median filter is very appropriate because of its high robustness, making the likelihood of sporadic noisy distance estimates very low. Furthermore, because of mobility the distance estimates change and can easily be adapted by a moving median filter. The pre-processing leads to better location estimates from the Cricket system (Ansari *et al., 2007b)* and as a consequence results in more accurate tracking outputs.

*Figure 4. Plot of the distance ranging error for distances between 50 cm and 1050 cm and angles between 0 degree and 90 degrees*




## Applying Bayesian Framework on Cricket Localization

As described in the previous section, Bayesian techniques can be applied on localization system to achieve performance gains. In the following we will describe how particle filtering can be combined to the Cricket system for minimizing the positioning error (Ansari *et al., 2007b)*.

In the particle filtering setup, the system maintains a set of parameters as a state vector. By modeling the system dynamics appropriately, a given state vector can be used to predict the state vector at the next measurement time step. The filtering framework carries out new measurements to correct the uncertainties/anomalies in the predicted state vector. Appropriate noise models are introduced to incorporate the uncertainties in the measurements and the system models. The system and measurement models may also consist of multiple models and depending upon the case, switching from one to other can be made.

In the following, we will give an example how particle filtering framework be applied on the Cricket system based localization of a toy train. We use this particular example due to its repeatability as well as non-trivial dynamics. However, the framework developed is general, and other dynamics and noise distributions for different motion dynamics can easily be included for applications in other scenarios. The toy train rail track is shown in Figure 5. The motion consists of two types of dynamics and is modeled by Constant Velocity (CV) and Constant Turn (CT) models along the straight and curved paths, respectively.

The state vector at any time instant $T_k$ consists of the coordinate position ($x_k$, $y_k$) and the corresponding velocity components ($\dot{x}_k$, $\dot{y}_k$) of the object. The state vector, $\mathbf{X}_k$, can be expressed as $\mathbf{X}_k = [x_k\ y_k\ \dot{x}_k\ \dot{y}_k]^T$. The system model can be written as

$$\mathbf{X}_{k+1} = \mathbf{F}_k \mathbf{X}_k + \mathbf{G}_k \mathbf{p}_{k,}$$

where $\mathbf{F}_k$ represents the transition state space matrix representing the motion dynamics of the train, $\mathbf{G}_k$ is the noise input matrix and $\mathbf{p}_k$ is the noise vector. Here,

$$F_k^{(CV)} = \begin{bmatrix} 1 & 0 & T_k & 0 \\ 0 & 1 & 0 & T_k \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad F_k^{(CT)} = \begin{bmatrix} 1 & 0 & \dfrac{\sin(\omega T_k)}{\omega} & \dfrac{\cos(\omega T_n) - 1}{\omega} \\ 0 & 1 & \dfrac{1 - \cos(\omega T_k)}{\omega} & \dfrac{\sin(\omega T_n)}{\omega} \\ 0 & 0 & \cos(\omega T_k) & -\sin(\omega T_k) \\ 0 & 0 & \sin(\omega T_k) & \cos(\omega T_k) \end{bmatrix}$$

where ω represents the turn rate and has opposite direction along the two opposite curved sections. The noise process $\mathbf{p}_k$ is assumed to have Gaussian distribution. The $\mathbf{G}_k$ is given by

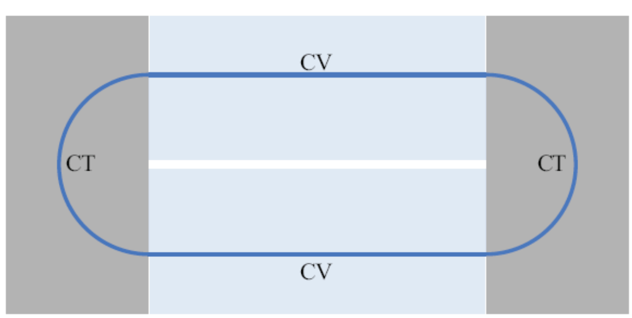
$$G_k = \begin{bmatrix} \dfrac{T_k^2}{2} & 0 \\ 0 & \dfrac{T_k^2}{2} \\ T_k & 0 \\ 0 & T_k \end{bmatrix}.$$

The measurement model is given by

$$\mathbf{Z}_\mathbf{k} = \mathbf{H}\mathbf{X}_k + \mathbf{n}_{k\,,}$$

where $\mathbf{Z}_k$ is the measurement vector, $\mathbf{H}$ is the measurement matrix and $\mathbf{n}_k$ represents the measurement noise process. The measurement system provides the coordinate position of the object and the angle information obtained from the digital compass, CMPS03. The digital compass is calibrated and gives

*Figure 5. Trajectory of the toy train which exhibits two kinds of motion dynamics. It can be represented by Constant Velocity (CV) model along the straight sections while Constant Turn (CT) model along the curved sections of the path.*



the absolute angle information. The measurement model is therefore expressed as $\mathbf{H}(\mathbf{X}_k) = [x_k\ y_k\ \theta]^T$. The measurement noise characteristics $\mathbf{n}_k$ are found using approximation on a statistically big enough set of localization and angle measurements. In the above framework, there exist two motion models and it is necessary to switch from one model to the other appropriately. This is generally achieved by maintaining a mode transition or regime switching matrix. We refer the reader to Arulampalam *et al.* (2002) and Ristic *et al.* (2004) for a detailed discussion on multi-modal Bayesian filtering. This approach can be in general visualized as parallel banks of particle filters as shown in Figure 6.

In our simplified case with the availability of absolute angle information from the digital compass, we could easily decide to switch from one motion model to another by observing the change in the absolute angle. More complicated model-switching logic would, of course, be necessary if such absolute information were not available, or the readings obtained from the sensor had significant inaccuracies. Especially in the latter case the accuracy of the models employed becomes critical. Noisy measurements with incorrect noise model can cause significant problems or delays in identifying correct dynamics. Also, if the behavior of the system deviates significantly from the dynamics selected the performance of the framework will, of course, be poor.

The particle filtering framework now estimates the posterior probability $p(\mathbf{X}_k \mid \mathbf{Z}_k)$ by Monte Carlo integration. After applying the above described framework on the toy train, we obtained the tracking results as shown in Figure 7. The plus signs in the figure indicate the location measurements and the continuous line indicates the trajectory estimated by the tracking algorithm for repeated number of rounds. For 358 location estimates in three complete rounds, the average RMSE (Root Mean Squared Error) is found to be approximately 2.8 cm, which is certainly an improvement over the localization obtained from Cricket system alone (A. Smith *et. al.*, 2004). Bayesian filtering can be applied to other systems as well for indoor localization in sensor networks and has shown to result in improved accuracy.

*Figure 6. Multiple motion models and switching logic*

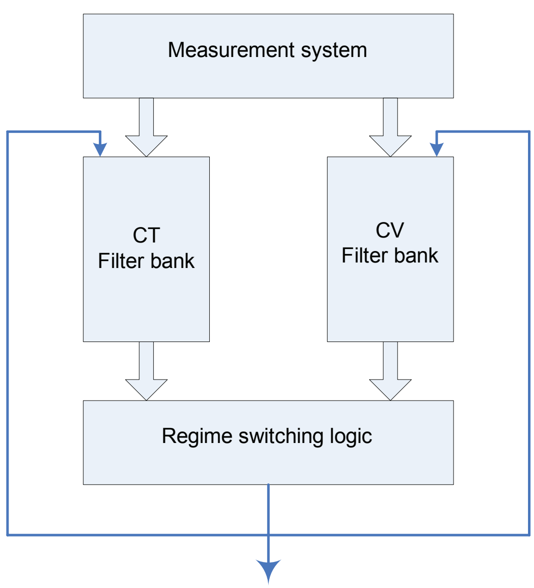
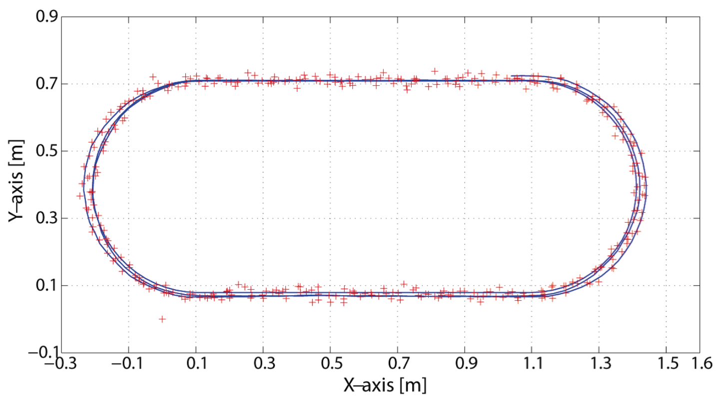


*Figure 7. Tracking performance of the toy train after applying particle filtering based technique on the Cricket localization system (adapted from Ansari et al., 2007b; © 2007 IEEE).*




## Inertial Navigation

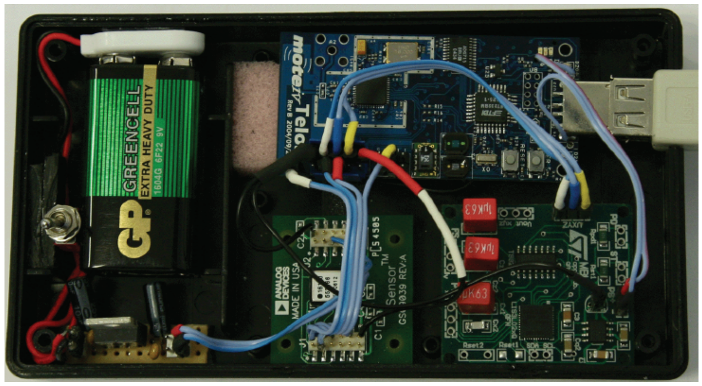
Inertial navigation can be utilized for tracking the position and orientation of a moving object without requiring any external frame of reference. Inertial navigation has outdoor as well as indoor applications. It has been widely used in the field of aviation, military systems and robotics. An inertial navigation system essentially consists of gyroscopes and accelerometers. By simply applying principles of Newtonian Physics on the accelerometer and gyroscope readings, relative position and orientation information of the moving object can be achieved. Accelerometer readings are used to estimate the displacement along different axes and the gyroscope readings give the relative change in the orientation along different directions. By combining the information from the accelerometers and gyroscopes, smoothened trajectory of the moving object can be obtained. The relative positioning information can be mapped to an appropriate reference scale by applying a single rigid transformation. One of the biggest problems of the inertial navigation is the error accumulation. Therefore, the error needs to be compensated from time to time.

Besides other application areas, inertial navigation has also a good potential in sensor networks. This is mainly owing to the design of very low power inertial sensors that can easily be attached to sensor nodes. One such example is the inertial navigation sensor board prototype (Popa *et al.*, 2008). It consists of STMicroelectronics' LIS3L02AQ accelerometer and Analog Devices' ADIS16100 gyroscope connected to a TelosB sensor node as shown in Figure 8.

## Indoor Human Tracking Using Inertial Sensors

In order to track a person using inertial sensors, the speed and direction of the motion are required. Human motion is characterized by several acceleration components along various directions and the displacement cannot simply be found out by integrating the acceleration components twice. However, there is a unique vertical acceleration spike associated with each single foot-step as a person walks. For slower and smoother steps, the acceleration components along other axes are less visible. Another peculiar characteristic of human motion is the foot-step size, which remains constant for a particular

*Figure 8. Inertial Navigation sensor node platform (from Popa et al., 2008; © 2008 IEEE)*




individual in his/her normal motion. Based on the inter-peak distance of the acceleration signature and the step-size for a particular person, his/her speed and hence displacement can be computed.

One of the advantages of inertial navigation is its independence from external factors. However, inertial navigation systems tend to accumulate errors over a period of time and therefore, periodic corrections are necessary. The position corrections can be applied using another system.

Cricket system has coverage problems owing to the line-of-sight requirements but it has relatively high accuracy. One logical approach is to combine Cricket system with inertial navigation. The combined solution can solve the coverage problem by relying on the inertial navigation and the error accumulation can be eliminated by periodically correcting the position using the Cricket system measurements. Figure 9 shows the localization output for Cricket system and a hybrid system, which combines inertial navigation with Cricket system. It can be observed that Cricket system alone has a high accuracy but it suffers from coverage problem owing to the strict line-of-sight requirements. Inertial navigation accumulates errors over time and the output for an arbitrary human motion is not very accurate. The combination of the Cricket system and inertial navigation has better accuracy than achieved by inertial navigation alone and the problem of blind spots is also eliminated. The improved coverage of the combined system is illustrated by the larger number of combined localization estimates compared to the ones obtained from the Cricket system alone.

## LOCALIZING AND TRACKING VEHICLES

We shall now move from indoor to outdoor localization systems, focusing especially on vehicular localization and tracking. In particular we report on our work on the design and implementation of a complete, large-scale modular sensor network hardware/software platform for target tracking applications (Ansari *et al.*, 2007a). In this work, the objective was to design a flexible software platform that could be adapted into a variety of scenarios, and to prototype it on a likewise adaptable hardware platform targeted for passive tracking scenarios. In the resulting design the software platform is carefully separated from the hardware by various abstraction layers, and due to the modular design various filtering, data processing and communication solutions can be used in the platform according to the needs of the particular application. The overall hardware design is flexible as well, in the sense of not being confined to tracking applications with a particular target object in mind. Combination of magnetometers and

*Figure 9. Localization output for Cricket System, inertial navigation and the combined system for human motion on an arbitrary path (from Popa et al., 2008; © 2008 IEEE)*

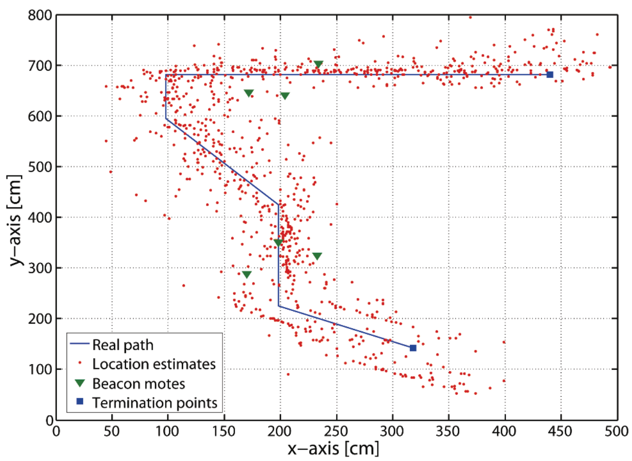
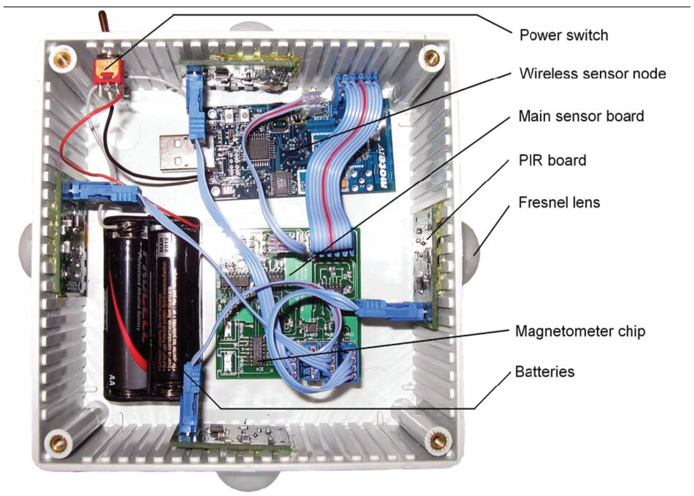


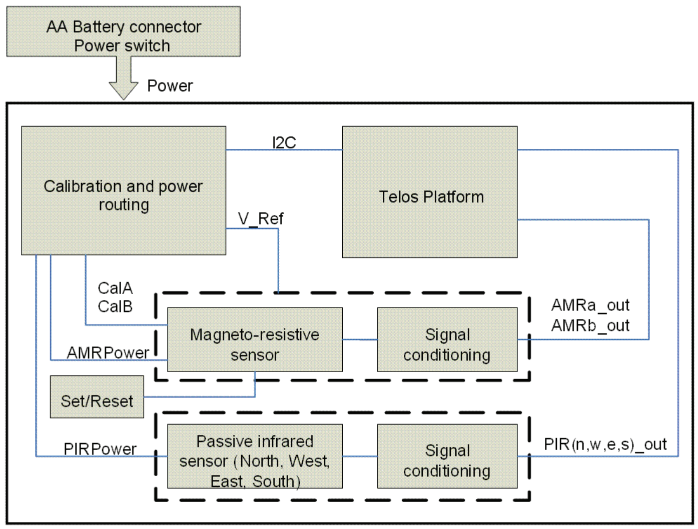

*Figure 10. Sensor node platform for vehicular tracking consisting of the passive infrared sensors and magnetometer attached to TelosB (from Ansari et al., 2007a; © 2007 IEEE)*




passive infrared sensors makes the platform very versatile, and we enhanced the platform further by developing automatic calibration features to remove the influence of environmental changes.

The sensor nodes used in the tracking system were built using the TelosB platform from Moteiv Inc. extended with a substantial amount of customized hardware. In particular, the nodes feature a sensor board consisting of two types of sensors namely passive infrared (PIR) sensors and anisotropic magneto-resistive (AMR) sensors, as shown in Figure 10. PIR sensors detect the differential thermal energy signal rather than absolute values and are therefore highly suitable for tracking applications. AMR sensors generate an output voltage proportional to the magnetic field strength. A moving ferromagnetic object disturbs Earth's magnetic field and causes the AMR sensors to generate an output signal, which is used for detection purposes as explained below. Low power consumption is one of the key objectives in the

*Figure 11. Circuit block diagram of the sensor node platform used for vehicular tracking (from Ansari et al., 2007a; © 2007 IEEE)*




design. The energy efficient design includes individual power control of the PIR and AMR sensors. The data pre-processing includes multiple phases: The output signal of the sensors is amplified using two-stage instrumentation amplifiers before feeding it to the *ADC* of the TelosB platform. The circuit also includes high frequency noise suppression filters as part of the signal pre-processing implemented in the circuitry. In order to demagnetize the AMR sensors, we included a provision for external set/reset circuitry, which we always used before any large scale deployment setups in order to compensate the long term drift and loss of sensitivity of the AMR sensors. The loss of sensitivity and drifts are developed owing to the alignment of ferro-sensitive cells inside the sensor chip when exposed to high magnetic field over extended periods of time as described in the above discussion on Compensation of Unwanted constants and Long-term Drift.

The packaged sensor node is shown in Figure 10 together with the block diagram of its architecture (c.f. Figure 11). A set of four PIR sensors (called as North-West-South-East) are mounted orthogonally for a complete 360-degree field of view. Fresnel lenses are used to increase the sensing range but at the same time not losing the beam-width below 90 degrees per PIR sensor. A combination of two axis magnetometer (AMRa and AMRb) enables the sensor node to detect moving ferromagnetic objects in a field. The amplitude level from the AMR sensors is highly dependent on its orientation with respect to Earth's magnetic field, the ferromagnetic material content in the surroundings and obviously on the strength of Earth's magnetic lines of force. For the optimum swing of the signal, the amplified output of the magnetometer to be fed to the *ADC* should be at the midscale (around 1.5V). This is done by tuning the resistance of a digital potentiometer, and hence adjusting the voltage levels at one of the inputs of the second stage amplifier. This calibration phase (c.f. section on Data Calibration) is a recursive process. Firstly, a set of 10 AMR samples are taken and the mode value is calculated. The underlying reason is that the probability of an outlier is very low. The mode value is converted into voltage and is checked whether it lies within a small window around the mid-scale voltage. Otherwise, the value of the potentiometer is increased or decreased accordingly by sending appropriate commands to the digital potentiometer over the $I^2C$ bus. The process is repeated till the voltage assumes a mid-scale value. This process not only prevents clipping of the signal but also enables the sensor node to calibrate automatically to any environmental condition.

The software developed for the platform also follows a layered and modular design to ensure flexibility. Transfer of the measurement data to gateway nodes, time synchronization, calibration etc. are handled in middleware tailored for the system, but mainly consisting of standard protocol solutions. Sensor readings themselves are processed with an FIR low-pass filter to reduce noise prior to further processing. The FIR filter is chosen with a cut-off frequency as depicted in the signal of interest. The filter components implemented in software are also designed for maximum reusability, with filter parameters such as type, order, cut-off frequency and filter coefficients all being adjustable according to the characteristics of the incoming raw sensor readings. Another major signal processing aspect implemented in the software is adaptation to changes in the environment. Changes in ambient conditions and Earth's magnetic field cause, for example, the magnetometer readings to drift over time. A computationally less intensive EWMA filter implementation is used to track the baseline of the magnetometer readings, which is subtracted from the filtered measurements before further processing is applied. This way a calibrated, zero-offset and non-clipped signal is obtained, which can be processed for the potential object detection purposes.

After applying the data processing techniques, the readings from AMR and PIR sensors are given as inputs to the vehicle detection algorithm running on each node. The algorithm relies on impulse integration with adjustable thresholds as illustrated in Figure 12 for the AMR sensor. The filled area between this curve and the thresholds (dotted lines) represents the integrated impulse value. When the AMR readings return to the area between the thresholds, a dwell timer is started (horizontal line). If the readings cross the threshold before this timer expires, the timer is reset and the integration of impulse continued (crosses terminating the thick timer line). When the dwell timer expires (upward arrow), the impulse value is compared against a pre-set value. Too small impulses are considered as noise, and detection is not signaled (cross over the upward arrow).

*Figure 12. Illustration of the vehicle detection algorithm using a magnetometer (from Ansari et al., 2007a; © 2007 IEEE)*

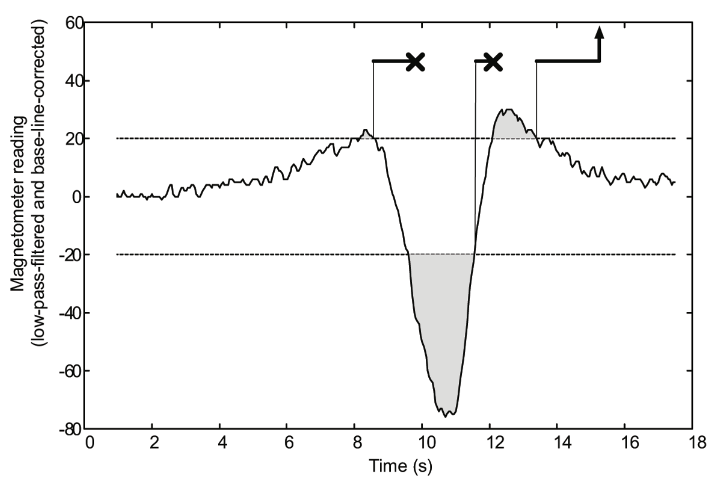
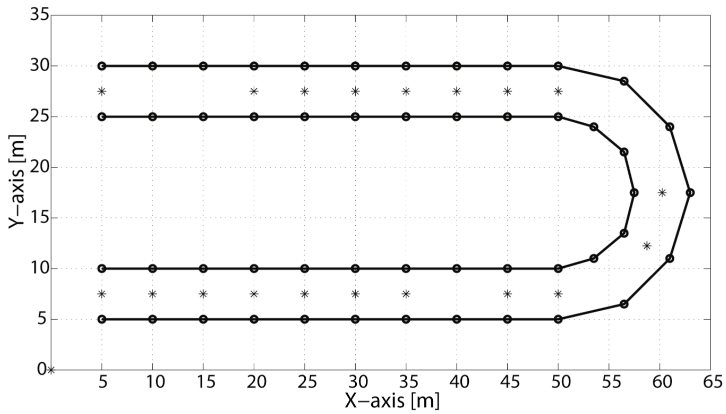


*Figure 13. Vehicle tracking output (from Ansari et al., 2007a; © 2007 IEEE)*




All the detection events from the individual sensor nodes are gathered on a powerful node responsible for further data processing. Rejection of false positives and data fusion in general is performed at this stage. Even though extensive filtering is applied in various parts of the system occasional noise spikes unavoidably still cause false detections on individual nodes, as will sudden changes in environmental conditions. Also detections from PIR sensors are not necessarily all related to vehicles, but might also arise from animals or humans passing by. Neighborhood-based fusion of the readings from both the AMR and PIR sensors from the whole sensor field results, however, in accurate detection. Figure 13 illustrates the performance of the system on a small test track where a single vehicle was moving at a speed of approximately 50 km/h. The dots indicating localized detections were confirmed to be highly accurate. The figure also shows that the system did not, however, reach 100% accuracy as some of the events were missed. The underlying cause of this appears to be the residual unreliability in the communications stack rather than the algorithms described above. With a more aggressive data protection in the network (such as multipath routing) we expect our design to yield highly accurate vehicle detection and tracking performance even in very dynamic environmental conditions.

We have also worked on distributed localization and tracking systems for vehicles utilizing particle filters for data fusion. Our focus has been especially on fusing tracking information obtained from radars with local positioning estimates of the vehicles obtained via vehicle-to-vehicle communications. Particle filters are an especially appropriate solution for such a scenario due to the non-linearities in the system model, highly non-Gaussian measurement noises of radars, as well as due to the need to fuse sensor readings coming from diverse sources. For further details we refer the reader to Riihijärvi *et al.*, 2005.

## CONCLUSION

In this chapter we discussed some of the algorithms and solutions that can be applied for data processing and filtering in wireless sensor networks deployed for localization and tracking applications. Following a short overview of the different basic localization and ranging solutions commonly applied in the field, we gave a short account of the basic data filtering algorithms, such as median and lowpass filters, as well as different flavors of averaging. We then focused on Bayesian filtering as a method of choice for solving the tracking problem based on individual measurements. We advocate the use of particle filters due to their ability to handle non-linearities and non-Gaussian measurement noises, both of which appear to be common features in tracking scenarios. The increase in computational complexity induced by particle filters is not very high compared to, for example, usual Kalman filters, and the improvement in accuracy and robustness is significant. They are also not methodologically more difficult to implement or understand than other major Bayesian filter types. Finally, we have discussed the design of indoor and outdoor localization and tracking systems we have implemented based on the data processing and filtering solutions discussed above.

## REFERENCES

Ansari, J. *et al*. (2007a). Flexible Hardware/Software Platform for Tracking Applications. In *Proceeding of VTC 2007*, Dublin, Ireland.

Ansari, J., Riihijärvi, J., & Mähönen, P. (2007b). Combining Particle Filtering with Cricket System for Indoor Localization and Tracking Services. In *Proceeding of PIMRC 2007*, Athens, Greece.

Arora, A., Dutta, P., Bapat, S., Kulathumani, V., Zhang, H., Naik, V., Mittal, V., Cao, H., Demirbas, M., Mohamed, G., Choi, Y-R., Herman, T., Kulkarni, S., Arumugam, U., Nesterenko, M., Vora, A., & Miyashita, M. (2006). A line in the sand: A wireless sensor network for target detection, classification and tracking. *Computer Networks*, *46*(5), 605–634.

Arulampalam, S., Maskell, S., Gordon, N., & T. Clapp, (2002). A Tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 50(2), 174-188.

Bahl, P., & Padmanabhan, V. (2000). RADAR: An In-Building RF Based User Location and Tracking System. In *Proceedings of the IEEE INFOCOM 2000*,Tel Aviv, Israel.

Drumheller, M. (1987). Mobile robot localization using sonar. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *9*(2), 325–332.

Federal Communications Commision´s Enhanced 911 Wireless Services. http://www.fcc.gov/pshs/services/911-services/. Retrieved: 08.10.2008.

Fox, D., Hightower, J., Liao, L., Schulz, D., & Borriello, G. (2003). Bayesian filters for location estimation. *IEEE Pervasive Computing*, *2*(3), 24–33.

Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., & Nordlund, P.-J. (2002). Particle Filters for Positioning, Navigation, and Tracking. *IEEE Transactions on Signal Processing*, *50*(2), 425-437.

Honeywell (2008). *HMC1022 data sheet.* www.ssec.honeywell.com/magnetic/datasheets/hmc1001-2_1021-2.pdf,", Retrieved: 08.10.2008.

Howard, A. (2006). Multi-robot Simultaneous Localization and Mapping using Particle Filters. *The International Journal of Robotics Research*, *25*(12), 1243-1256.

Kuckertz, P., Ansari, J., Riihijärvi, J., & Mähönen, P. (2007). Sniper Fire Localization using Wireless Sensor Networks and Genetic Algorithm based Data Fusion. In *Proceedings of IEEE MILCOM 2007*, Orlando, Florida, USA.

Lorincz, K., & Welsh, M. (2007). Motetrack: A robust, decentralized approach to RF-based location tracking. In *Personal and Ubiquitous Computing, Special Issue on Location and Context-Awareness, 11*(6), 489-503.

Mihaylova, L., Angelova, D., Honary, S., Bull, D., Canagarajah, C., & Ristic, B. (2007). Mobility Tracking in Cellular Networks Using Particle Filtering. *IEEE Transactions on Wireless Communications*, *6*(10), 3589-3599.

Olama, M., Djouadi, S., & Pendley, C. (2006). Position and Velocity Tracking in Mobile Cellular Networks Using the Particle Filter. *In Proceedings of IEEE WCNC 2006*, Las Vegas, USA.

Polastre, J., Szewczyk, J., & Culler, D. (2005). Telos: enabling ultra-low power wireless research. *In Proceedings of Information Processing In Sensor Networks 2005*, Los Angeles, USA.

Popa, M., Ansari, J., Riihijärvi, J., & Mähönen, P. (2008). Combining Cricket System and Inertial Navigation for Indoor Human Tracking. *In Proceeding of WCNC 2008*, Las Vegas, USA.

Priyantha, N. (2005). *The Cricket Indoor Location System*. doctoral dissertation. Massachusetts Institute of Technology, Boston, USA.

Riihijärvi, J., Vázquez Martí, C., & Mähönen, P. (2005). Enhancing Radar Tracking with Car-to-Car Communication. In *Proceedings of IEEE SCVT'05*, Enschede, the Netherlands.

Ristic, B., Arulampalam, S., & Gordon, N. (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications.* Artech House, Norwood, Massachusetts, USA.

Simandl, M., & Straka, O. (2007). Sampling Densities of Particle Filter: A Survey and Comparison. *In Proceedings of the 2007 American Control Conference*, New York, USA.

Simon, G., Maroti, M., Ledeczi, A., Balogh, G., Kusy, B., Nadas, A., Pap, G., Sallai, J., & Frampton, K. (2004). Sensor Network-Based Countersniper System. In *Proceedings of the 2nd international conference on Embedded Networked Sensor Systems*, Baltimore, USA.

Smith, A., Balakrishnan, H., Goraczko, M., & Priyantha, N. (2004). Tracking Moving Devices with the Cricket Location System. *In Proceedings of the ACM MobiSys 2004*, Boston, USA.

Thrun, S. (2002). Particle Filters in Robotics. In *Proceeding of Uncertainty in AI.*

Van Trees, H., & Bell, K. L. (Eds.) (2007, August). *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking.* Wiley-IEEE Press.

Want, R., Hopper, A., Falcao, V., & Gibbons, J. (1992). The active badge location system. *ACM Transactions on Information Systems*, *10*(1), 91-102.

Ward, A., Jones, A., & Hopper, A. (1997). A new location technique for the active office. *IEEE Personnel Communications*, *4*(5), 42–47.

Whitehouse, K., & Culler, D. (2003). Macro-calibration in Sensor/Actuator Networks. *Mobile Networks and Applications Journal (MONET), Special Issue on Wireless Sensor Networks*, *8*(4), 463-472.