Bayesian Approach for Indoor Pedestrian Localisation

Ivan Spassov, EPFL ENAC TOPO
Michel Bierlaire, EPFL SB IMA ROSO
Bertrand Merminod, EPFL ENAC TOPO

Conference paper STRC 2006

STRC | **6th Swiss Transport Research Conference**
Monte Verità / Ascona, March 15. – 17. 2006

Bayesian Approach for Indoor Pedestrian Localisation

Ivan Spassov
EPFL ENAC TOPO
Lausanne - Switzerland

Phone: 021.693.2751
Fax: 021.693.5740
email: ivan.spassov@epfl.ch

Michel Bierlaire
EPFL SB IMA ROSO
Lausanne - Switzerland

Phone: 021.693.2537
Fax: 021.693.5570
email: michel.bierlaire@epfl.ch

Bertrand Merminod
EPFL ENAC TOPO
Lausanne - Switzerland

Phone: 021.693.2754
Fax: 021.693.5740
email: bertrand.merminod@epfl.ch

March 2006

Abstract

The principal concept of navigation is to start from a known (initial) position and to ensure a continued and reliable localisation of the user during his/her movement. The initial position of the trajectory is usually obtained via GPS or defined by the user.
Consider a pedestrian navigation system which contains a GPS receiver and a set of inertial sensors, connected with a map database. In the urban environment and indoors the localisation depends entirely on the measurements from the inertial sensors. The trajectory is defined in a local coordinate system and with an arbitrary orientation. The problem to solve is to determine the user's location using the map database and inertial measurements of the navigation system.
The idea behind our approach is to find the location and orientation of the trajectory and thus the user's location. The proposed solution associates the user's trajectory with the map database applying statistical methods in combination with map-matching.
Similar geometric forms must be identified in both the trajectory and the link-node model. The trajectory, defined by a set of consecutive points, is transformed to a set of lines thanks to a dedicated *motion model*.
In this research we propose a solution based on statistical methods where the history of the route and actual measurements are treated at the same time. The determination of the absolute position is entirely represented by its probability density function (PDF) in the frame of Bayesian inference. Following this approach the posterior estimation of the user's location can be calculated using prior information and actual measurements. Because of the non-linear nature of the estimation problem, non-linear filtering techniques like *particle filters* (Sequential Monte Carlo methods) are applied.

Keywords

Bayesian – Sequential Monte Carlo – pedestrian – localisation – indoors – map-matching – map database

1. Introduction

The principal concept of navigation is to start from a known (initial) position and to ensure a continued and reliable localisation of the user during his/her movement. The initial position of the trajectory is usually obtained via GPS, and described by so-called absolute coordinates in an Earth-fixed system. This positioning technique requires optimal conditions and undisturbed satellite signals, which is not the case in a city or indoors. Alternatively, the user may define the start of the trajectory, but normally he/she does not know the own absolute position.

Consider a pedestrian navigation system which contains a GPS receiver and a set of inertial sensors to capture different characteristics of the user's movement, thus representing the trajectory by a set of points. Such a navigation system could be connected with a map database. By saying *map database* we consider mainly the well-known link-node model, which represents the street network of some region as planar graph. For indoor navigation, a similar link-node model is applied to represent the network of all corridors and passageways in the building. For instance, each link connects two nodes and each node is defined by its coordinates in the national coordinate system. Thus the link-node model is defined in an absolute coordinate system.

In the urban environment and indoors the localisation depends entirely on the measurements from the inertial sensors. The position of each step is determined as a function of the previous position and relative measurements like speed and turning rate. Thus the trajectory is defined in a local coordinate system and with an arbitrary orientation. The problem to solve is to determine the user's location using the map database and inertial measurements of the navigation system.

The idea behind our approach is to find the location and orientation of the trajectory and thus the user's location. Our proposed solution associates the user's trajectory with the map database applying statistical methods in combination with map-matching.

First of all, we need to answer the question: what are the characteristics of a trajectory that could be associated with some elements of the map database? Similar geometric forms must be identified in both the trajectory and the link-node model. We consider that the map database cannot be modified, so a solution needs to be found in the modification of the trajectory. Since the trajectory is defined by a set of consecutive points, this set must be transformed to a set of lines before searching an association with the link-node model of the map database. That is possible thanks to a dedicated *motion model*. The motion model consists in a number of functions capable to detect different characteristics of the pedestrian trajectory like straight walk, turn, stop, etc. With the help of speed and direction constraints, the essential movements of the user's trajectory can be detected. After the application of the motion model the trajectory is represented as a polygon where each edge is a straight walk pass and each vertex is a turn. The link-node model of the map database cannot be modified, instead it can be preanalysed so only the critical nodes like turns and crossings need to be considered.

After the pre-analysis of the link-node model and the modification of the trajectory, we have two data sources and we must associate similar details from both data sources. The trajectory can be considered as the history of the route and its last point as the actual position of the user.

In this research we propose a solution based on statistical methods where the history of the route and actual measurements are treated at the same time. The determination of the user's location is entirely represented by its probability density function (PDF) in the frame of Bayesian inference. Following this approach the posterior estimation of the user's location can be calculated using prior information and actual measurements. Because of the non-linear nature of the estimation problem, non-linear filtering techniques like *particle filters* (Sequential Monte Carlo methods) are applied.

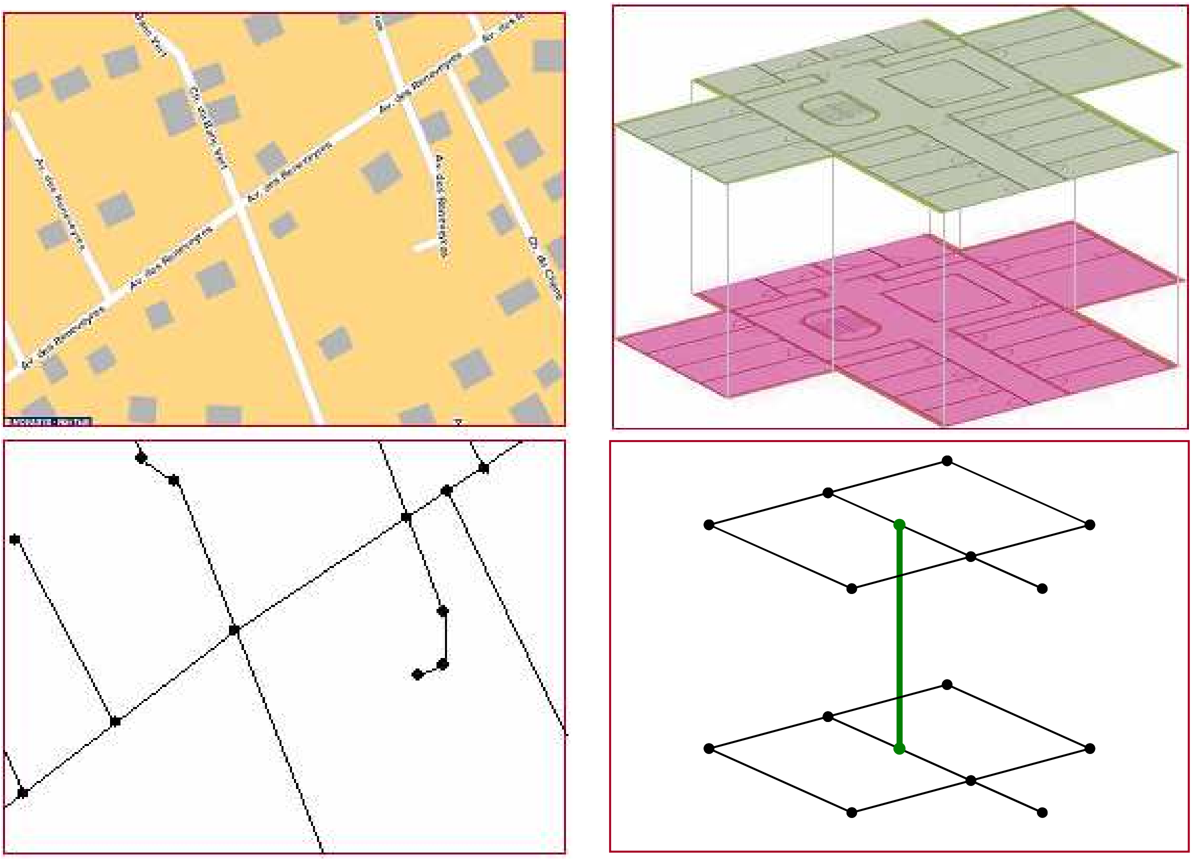
Outline: In sections 2 and 3 the map database and the pedestrian tajectory will be presented. The problem formulation and the association of both datasources will be discussed in section 4. Section 5 presents the numerical solution of the estimation followed by tests of the algorithms in section 6 and conclusions in section 7.

2. Database model

Generally the geographical database contains information on the position, dimensions, capacity, functionality, etc. of the geographical objects. For the purposes of the navigation process the connections between these objects are of interest. In the urban areas these connections are defined by the street network, which is represented by a planar graph [Bernstein, Kornhauser 1996]. The streets are defined by links or arcs and the crossings – by nodes (Fig.1a). That representation of the street network is named *link-node model* .

In order to create the map database for the building the same link-node model is used [Büchel, 2003]. This model includes all connections like corridors and passageways. The start and the end of each link is defined by nodes. Links are assumed to coincide with the axis of the corridor (Fig.1b). Each link connects two nodes and each node is known with its coordinates in the national coordinate system. Thus the link-node model of the building is absolutely defined. Using the node coordinates different properties of the links, like length and azimuth could be computed. An important property of the building datamodel is that the vertical connections are considered. The elevators and staircases are represented as links connecting two nodes from different floors. In the context of indoor pedestrian navigation the database could be constituted by the limits of the building. However, it could be connected with the street network database or with the database of other buildings. Based on that graph representation there exist algorithms for computing the shortest path between two points of interest.

Figure 1: Street network and building represented by the link-node model



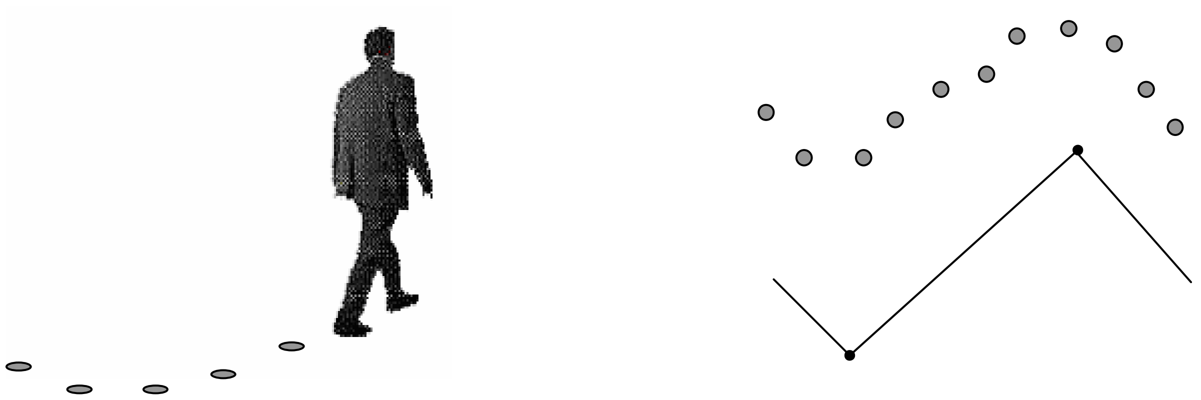
(a) (b)

In the building normally one needs the path from one room to another. For that reason, in the link-node model, the doors of the rooms are considered as nodes. Thus the map database becomes huge. Moreover for the process of localisation discussed here the doors are not of interest. That imposes a pre-analysis of the link-node model of the map database so as to consider only the critical nodes like turns and crossings.

3. Pedestrian trajectory

We consider a personal navigation system that contains a GPS receiver and a set of sensors that capture different characteristics of the human walk. Since the problem to tackle is pedestrian localisation indoors, we'll focus on the use of inertial sensors only. Sensors like accelerometers and gyroscopes are capable to measure the speed and the turn rate of a moving body. While the person walks measurements are made on each step [Ladetto, 2002]. Thus the pedestrian trajectory is represented as a set of successive points (Fig.2a). Every point is registered with the turn rate, speed and time. Thanks to the speed and time the distance between each pair of consecutive points is computed.

Figure 2: The set of successive points

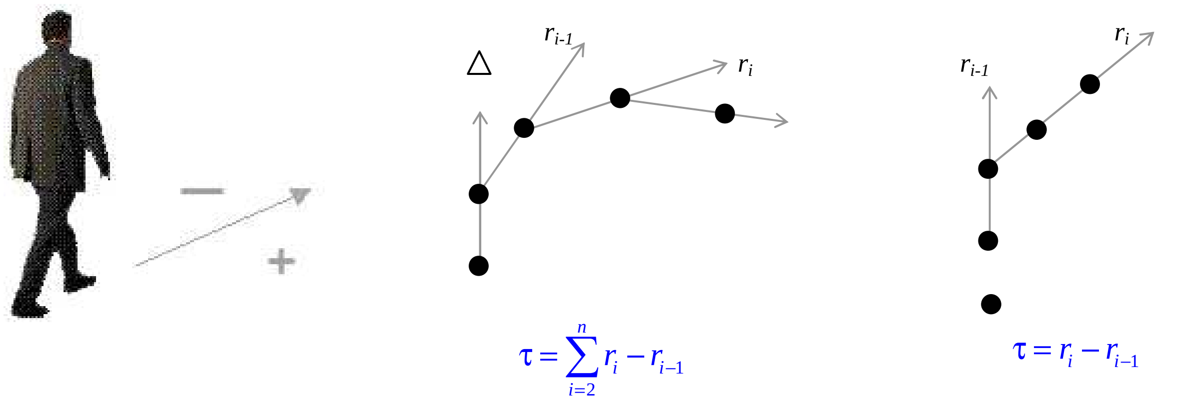


(a) (b)

This set of points is in a local coordinate system with arbitrary orientation. In order to determine the location of the user we need to find the location and orientation of that trajectory. The proposed solution associates the user's trajectory with the map database. So we need to identify similar geometric forms in both the trajectory and the link-node model, a process known as *map-matching* . At this stage the set of points forming the trajectory can not be associated to the link-node model. The natural way to proceed is to transform the set of points to a polygon (Fig.2b). Thus the trajectory will be generalized so as to distinguish between straight walks and turns. To perform this transformation an algorithm is developed that applies a dedicated *motion model*.

We assume that the trajectory is defined in a local coordinate system with origin in the first position. The construction of the polygon is based on the detection of the turns of the trajectory. With every step new values of the distance and the turn rate become available. The bearing of each step with respect to the previous one can be computed. Depending on whether the person makes a right or a left turn the change in orientation (bearing) has a positive or a negative value (Fig.3a). Thus the algorithm can determine when the person enters and leaves the turn. The person can make a turn spread over several steps (Fig.3b) or a sharp change of direction in one step only (Fig.3c). In both cases the change of direction τ is computed.

Figure 3: Detection of the turns



(a) (b) (c)

The movement is considered as a turn if $|\tau| \geq 18°$, which is an empirically derived threshold. In the polygonal representation of the trajectory the turns are defined as nodes. In Figure 3c the node coincides with the position where the direction has been changed. For the case of Figure 3b we could not make the same decision, because the turn and the trajectory would not be represented correctly. Instead we can determine the position of a pivot point (marked with Δ on the figure) and the node in that case will coincide with it.

Based on that method the algorithm can detect different movements of the person like a turn or half-turn, a straight walk and a stop. The modification of the user's trajectory with respect to the motion model transforms the upcoming set of inertial measurements to a set of polygonal parameters (distances and angles).

Since the user walks through the corridors in the building, the trajectory (the polygon) that he performs is considered as a part of the link-node model of the building.

Now we have two sources of data, the map database and the polygonal representation of the user's trajectory. The problem to solve is to associate the data from both sources, i.e. map-matching. That is to find the placement of the polygon in the contents of the link-node model of the building. Then the last edge of the polygon represents the user's location in the building. So we need to select from the map database the link associated to the last polygon edge. While the database has a finite number of elements, the polygon is updated with a distance and an angle periodically. Every time the polygon is updated an estimation of the user's location will be performed. The estimation relies on prior information (the trajectory, actual measurements and database) that could be used to compute a posterior estimation via the Bayesian inference.

4. Problem formulation

The walking person is considered as a dynamic system, whose trajectory is modified with respect to the motion model. The evolution of that dynamic system is defined by the following state space model:

$$x_t = f\left(x_{t-1}, u_{t-1}\right) \tag{1a}$$

$$y_t = h\left(x_t, x_{t-1}\right) + z_t \tag{1b}$$

with the following elements

x_t	state vector
u_t	motion input
y_t	measurement vector
z_t	measurement error
$h(x_t, x_{t-1})$	dimensions of x_t and x_{t-1} according to the database

The state vector x_t represents the location (the link) in moment t. The dynamic process is discretized regarding the motion model, so an estimation is made every time the new measurements are available. The measurement vector $y_t = (l_t, \alpha_t)^T$ includes the distance and angle of movement detected by the motion model. The measurement noise e_t is assumed Gaussian. The history of all states up to moment t is defined by $X_t=\{x_0 , x_1 ,\ldots, x_t\}$, respectively $Y_t=\{ y_1 , y_2 ,\ldots, y_t\}$ defines the history of the measurements up to moment t. The problem to solve is to estimate x_t using the set of all available measurements Y_t .

From a Bayesian viewpoint this sequential estimation problem demands the computation of the posterior density $p(X_t|Y_t)$. We assume that the state follows a first order Markov process:

$$p(x_t|x_{t-1}, x_{t-2},\ldots, x_0) = p(x_t|x_{t-1}), \quad \text{and} \quad p(x_0|x_{-1}) = p(x_0) \tag{2}$$

So if we compute the *marginal* of the posterior density $p(x_t|Y_t)$, also known as *filtering density*, there is no need to keep the complete history of the states [Doucet et al., 2001]. Often in sequential estimation algorithms, the measurements are assumed to be independent given the states:

$$p(y_t|x_t, A) = p(y_t|x_t) \tag{3}$$

In our case such an assumption will not be reasonable because we consider that the trajectory made up to moment *t-1* holds additional information, which is critical for the estimation.

Considering the state space model and assumptions made, the filtering density is estimated:

$$\begin{aligned} p(x_t|Y_t) &= \frac{p(Y_t|x_t)\,p(x_t)}{p(Y_t)} \\ &= \frac{p(y_t, Y_{t-1}|x_t)\,p(x_t)}{p(y_t, Y_{t-1})} \\ &= \frac{p(y_t|Y_{t-1}, x_t)\,p(Y_{t-1}|x_t)\,p(x_t)}{p(y_t|Y_{t-1})\,p(Y_{t-1})} \\ &= \frac{p(y_t|Y_{t-1}, x_t)\,p(x_t|Y_{t-1})\,p(Y_{t-1})\,p(x_t)}{p(y_t|Y_{t-1})\,p(Y_{t-1})\,p(x_t)} \\ &= \frac{p(y_t|Y_{t-1}, x_t)\,p(x_t|Y_{t-1})}{p(y_t|Y_{t-1})} \end{aligned} \tag{4}$$

Here $p(x_t|Y_{t-1})$ is called *prior* of the state at moment *t*. It is obtained by using the state space model (1) and the Chapman-Kolmogorov equation:

$$p(x_t|Y_{t-1}) = \int p(x_t|x_{t-1})\,p(x_{t-1}|Y_{t-1})dx_{t-1} \tag{5}$$

considering the assumption that the state follows a first order Markov process. The transition density $p(x_t|x_{t-1})$ is defined by the evolution of the dynamic system (1a).

In (4) the *likelihood function* $p(y_t|Y_{t-1},x_t)$, is defined by the measurement model and the known statistics of the measurement noise e_t . The *evidence* $p(y_t|Y_{t-1})$ has function of a normalizing constant. Thus $p(x_t|Y_t)$ can be computed recursively in two stages: *prediction* and *update*.

- Prediction

$$p(x_t|Y_{t-1}) = \int p(x_t|x_{t-1})\,p(x_{t-1}|Y_{t-1})dx_{t-1} \tag{6}$$

- Update

$$p(x_t|Y_t) = \frac{p(y_t|Y_{t-1}, x_t)\,p(x_t|Y_{t-1})}{p(y_t|Y_{t-1})} \tag{7}$$

where $p(y_t|Y_{t-1}) = \int p(y_t|x_t)\,p(x_t|Y_{t-1})dx_t$

$p(x_t|Y_{t-1})$ is the prior from (4)

$p(y_t|Y_{t-1}, x_t)$ is the likelihood function

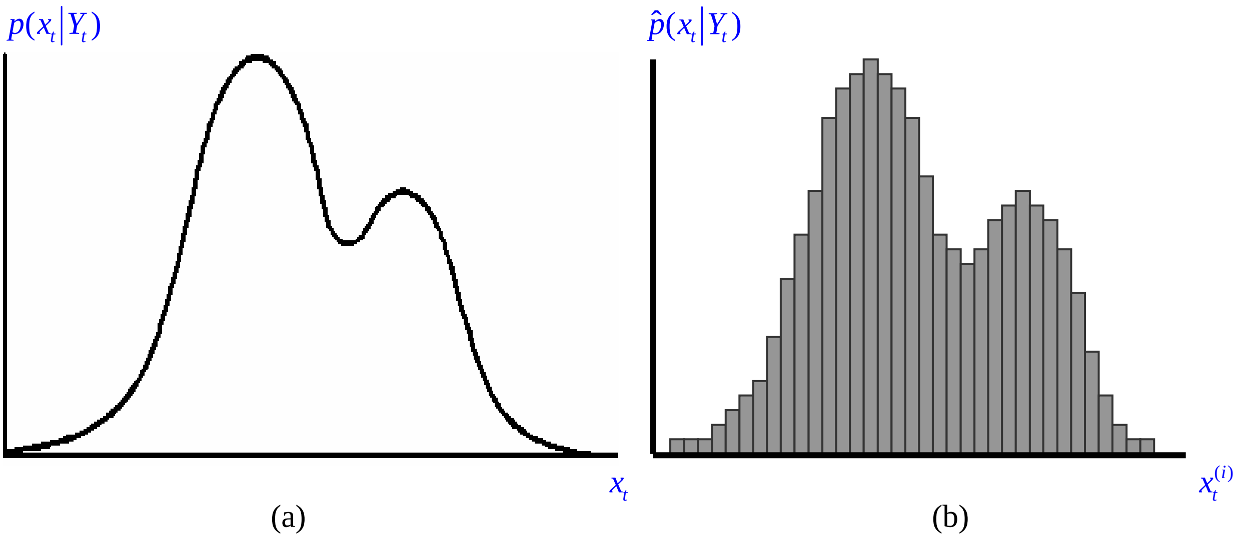
Bayes' theorem says that the posterior probability is *proportional to* the product of the prior probability and the likelihood function. To describe this relationship $p(y_t|Y_{t-1})$ is defined as a normalizing constant by which that product is divided.

That conceptual formulation of the problem, based on the Bayesian inference cannot be determined analytically [Arulampalam, 2001]. The solution can be achieved by applying Sequential Monte Carlo (SMC) methods also known as particle filters.

5. Particle filtering

Particle filtering is defined as a sequential process for estimation of the states (parameters or hidden variables) of a system when new sets of observations become available. The principle of the SMC methods is to discretize a given density using a great number of samples also known as particles [Antonini et al.] (Fig. 4).

Figure 4: Discretization of $p(x_t|Y_t)$ using N samples



This operation transforms the intractable integrals of the Bayesian solution (6) and (7) into tractable discrete sums of weighted samples [Doucet, 2001]:

$$p(x_t|Y_{t-1}) = \sum_{x_{t-1}\in L} p(x_t|x_{t-1})\,p(x_{t-1}|Y_{t-1})$$

$$p(y_t|Y_{t-1}) = \Sigma\, p(y_t|x_t)\,p(x_t|Y_{t-1})$$

where

$$p(x_t|x_{t-1}) = \begin{cases} 1 & \text{,if } x_t \text{ is a neighbour of } x_{t-1} \\ 0 & \text{,if not} \end{cases}$$

$p(x_{t-1}|Y_{t-1})$ is the estimation at instant $t-1$

L is the set of all the links in the map database

If we note the samples as $x_t^{(i)}$, and their weights as $w_t^{(i)}$, where $i=\{1,2,\ldots,N\}$, N ? 0 the posterior density $p(x_t|Y_t)$ could be approximated by (8) where $\delta(x_t - x_t^{(i)})$ is the Dirac delta measure [Gustafsson et al., 2002]

$$\hat{p}(x_t|Y_t) = \frac{\sum_{i=1}^{N} w_t^{(i)} \delta(x_t - x_t^{(i)})}{\sum_{i=1}^{N} w_t^{(i)}} \tag{8}$$

In our case $p(x_t|Y_t)$ is already discretized itself in accordance with the link-node model of the map database where each link is considered as a particle with associated weight. The weight $w_t^{(i)}$ reflects the probability that the i^{th} particle represents the user's location at moment t. The aim is to recursively compute $w_t^{(i)}$ applying prediction and update steps of the filtering process.

On each iteration of the process prior data is used to evaluate the posterior density. Initially at $t=0$ there is no prior data because no measurements are made on the user's walk and there is no information about the trajectory. At that moment we consider that the person could be anywhere in the building. Thus $p(x_t|Y_t)$ is defined by a uniform distribution where $w_0^{(i)} = w_0^{(j)}$, $i \neq j$ which is described by:

$$x_0^{(i)} : p(x_0), \quad w_0^{(i)} = \frac{1}{N}, \quad i = 1, \ldots, N \tag{9}$$

At moment $t+1$ a new set of information on the trajectory is available. The posterior density $p(x_{t+1}|Y_{t+1})$ is now calculated using (8). That is, the weights $w_{t+1}^{(i)}$ are updated according to:

$$w_{t+1}^{(i)} = p(y_{t+1}|Y_t, x_{t+1}^{(i)}) w_t^{(i)} \tag{10}$$

and normalized by:

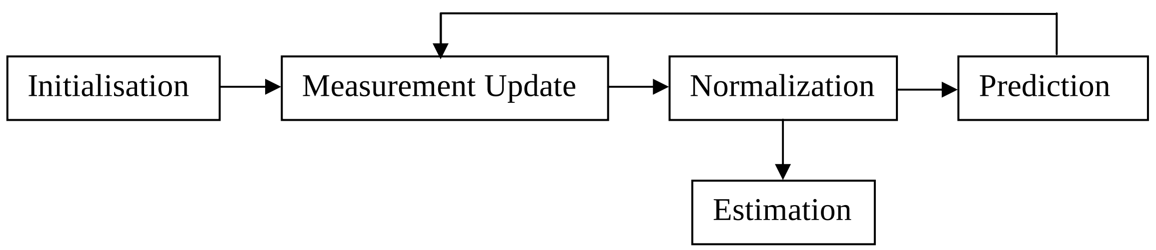
$$\overline{w}_{t+1}^{(i)} = \frac{w_{t+1}^{(i)}}{\sum_{i=1}^{N} w_{t+1}^{(i)}} \tag{11}$$

To calculate the likelihood function $p(y_{t+1} \,|\, Y_t, x_{t+1}^{(i)})$ in (10) the data of the measurement vector $y_t = (l_t, \alpha_t)^T$ is confronted to the data of the map database.

The location is estimated by the sample with maximal weight, noted as x_t. Note that at moment t several samples could have maximal weight thus representing the location in different places on the map. In the next iterations the additional information on the trajectory will help to solve this ambiguity. Finally, with the convergence of the filter only one sample will have a maximal weight of 1.

The prediction step chooses a new set of samples by giving a weight 1 to the neighbour samples of x_t and a weight 0 to the rest of the samples. Then the algorithm turns to the update step. A flow chart of the particle filtering algorithm is given as Figure 5.

Figure 5: The particle filter implementation

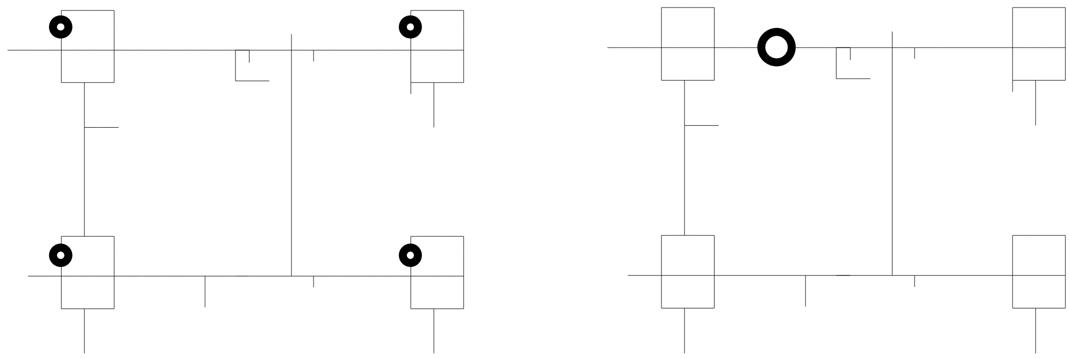


$w_0^{(i)} = \frac{1}{N}$ $i = 1,\dots,N$	$y_t = \begin{pmatrix} l_t \\ \alpha_t \end{pmatrix}, \quad h_{x_t^{(i)},x_t^{(i-1)}} = \begin{pmatrix} l_{x_t^{(i)}} \\ \alpha_{x_t^{(i)},x_t^{(i-1)}} \end{pmatrix}$ $w_t^{(i)} = \left(y_t - h_{x_t^{(i)},x_t^{(i-1)}}\right) w_{t-1}^{(i)}$ $i = 1,\dots,N$	$\bar{w}_t^{(i)} = \frac{w_t^{(i)}}{\sum_i^N w_t^{(i)}},$ $x_t = x^{(i)}_{\bar{w}_t^{(i)} MAX}$ $i = 1,\dots,N$	$w_{t+1}^{(i)} = \begin{cases} 1 & ,x_t \to x_{t+1}^{(i)} \\ 0 & ,\text{otherwise} \end{cases}$ $i = 1,\dots,N$

6. Tests and Discussions

A large map database for the entire campus of the EPFL[1] has been created to support the management of the buildings. Recently, the data structure was improved to implement new functionalities such as shortest path computation and guidance[2]. This database is used in our tests. As a navigation system we have used the Personal Navigation Module (PNM), developed by Vectronix AG[3]. The module is attached on the back side on the user's belt. The measurements are saved on a pocket PC. Both algorithms, the modification of the trajectory and the particle filter, are writen in MS Visual Basic and run in post-treatment mode.

Figure 6: Localisation on the link-node model. User's location is marked by



(a) (b)

Figure 6 presents an extract of the corridor network of the second floor of the civil engineering building. Because of its almost symmetric geometry, this part of the network allows to test the resolution of ambiguity. In Figure 6a the ambiguity is not solved yet, that is the polygon, walked so far, could be found on several places on the link-node model. In Figure 6b, after the new information has come the filter converges. That is, the unique place of the polygon has been found on the link-node model, thus determining the user's location.

The common constraint is that the trajectory has to be performed on places covered by the map database. That is, the person must not leave the area represented by the link-node model. With a normal walk, the user's location is determined after several iterations of the algorithm. Since the localisation is refered to a link from the database its precision depends on the length

[1] http://plan.epfl.ch

[2] http://www.camptocamp.com

[3] http://www.vectronix.ch

of that link. The key idea in the Monte Carlo simulation is to discretize the posterior density by a set of weighted samples. Generally the number of the samples has to be very large in order to approximate the real density. In our case the discretized posterior density $p(x_t|Y_t)$ is represented using all links of the database. Thus we work with the entire density involving all the samples into the computation at every moment. This is not an issue in post-processing, but a real-time implementation will impose restrictions.

7. Conclusions and perspectives

A solution for indoor pedestrian localisation is proposed in this research. The method is based on the Bayesian inference solved by particle filtering and applying map-matching techniques. A dedicated motion model is used to transform the user's trajectory into a polygon in order to associate it with the link-node model of the map database. Using inertial measurements only, the process of localisation is entirely autonomous and gives promising results. That method of localisation can be applied to many pedestrian navigation tasks. In particular, it suits the needs of fire-brigades and security services.

The future efforts in this research will point to at the modeling of more sophisticated movements of the person. Special attention will be payed to vertical movements (e.g. taking the stairs). The real-time implementation of the process is another challenge that will be addressed.

References

Antonini, G., Bierlaire, M., Weber, M. (to appear) Discrete choice models of pedestrian walking behavior, Transportation Research Part B (accepted for publication).

Büchel, D. (2003) Méthodes de guidage applicables au plan d'orientation de l'EPFL, *Travail de diplôme*, Laboratoire de topométrie, EPFL, Suisse

Ladetto, Q. (2002) Capteurs et algorithmes pour la localisation autonome en mode pédestre**,** *PhD thesis No.2710*, Laboratoire de topométrie, EPFL, Suisse

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, vol. 50, No.2

Doucet, A., de Freitas, N. and Gordon, N. (2001) Sequential Monte Carlo Methods in Practice, New York: Springler-Verlag.

Arulampalam, S., Maskell, S., Gordon, N., Clapp, T. (2001) A Tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking, ©QinetiQ Ltd, ©DSTO

Bernstein, D. and Kornhauser, A. (1996) An Introduction to Map Matching for Personal Navigation Assistants**,** New Jersey TIDE Center, New Jersey Institute of Technology