Indoor Localization by Particle Map Matching

Karim El Mokhtari[1]* , Serge Reboul[2], Jean-Bernard Choquel[2],
Benaissa Amami[1], Mohammed Benjelloun[2]
[1] Laboratoire d'Informatique, Système et Télécommunications,
Abdelmalek Essaadi University, Faculty of Sciences and Techniques,
Ancienne Route de l'Aéroport, Km 10, Ziaten. BP: 416. Tangier, Morocco
[2] Laboratoire d'Informatique Signal et Image de la Côte d'Opale,
Littoral Côte d'Opale University, Univ. Lille Nord de France,
50, rue Ferdinand Buisson BP 719 - 62228 Calais Cedex, France
Email: karim@elmokhtari.com

***Abstract*—This article presents the implementation of an indoor localization approach that combines map matching and a circular particle filter defined in a Bayesian framework. The technique relies only on velocity and heading observations coupled with a map of the road network. No prior knowledge of the initial position is given. A circular distribution is used to match the vehicle's heading with the roads direction. This allows to detect turns and provide a more accurate position estimate. The algorithm is assessed with a synthetic dataset in a real context.**

Keywords—Indoor localization, Particle filter, Map matching, Circular Filter

I. INTRODUCTION

Indoor localization has gained an increasing importance in the research area on positioning and tracking. Over the last decade, a lot of efforts have been devoted to indoor localization problems. One of the widely known approaches is fingerprinting where a pre-recorded radio map of the zone of interest is used to compute the best matching of the position. This matching uses the model constructed in a previous training phase. Many wave sources have been experimented like WiFi [1], [2], [3], GSM [4], [5] and sound signals [6]. Other techniques known as "Propagation based" compute distances through the calculation of the Radio Signal Strength Indications (RSSI) but are generally less reliable as they are compromised by interferences, multipath and shadowing [7].

Other approaches use a pre-deployed active beacon infrastructure at known locations with technologies like Bluetooth [8], infrared [9], ultrasound [10], RFID [11] or FM [12]. These short-range signals, however, need an important effort of deployment and a constant labor of monitoring and maintenance.

Indoor navigation has also been driven by the advancement in the technology of imaging and camera sensors (*e.g.* CCD). The authors in [13] use panoramic landscape images to localize pedestrians using 3D models. In [14], images of points markers taken from a phone camera are used to compute the location indoors.

In this research paper, we are particularly interested in the problem of localizing a vehicle operating in a logistic area such as a warehouse. In this kind of applications, the vehicle navigates in a predefined road network. However, the existence of metallic structures and obstacles makes difficult the use of technologies based on radio or light waves because of frequent non line-of-sight, multiple reflections and shadowing. In addition, the constant change in the area content configuration (movement of boxes and goods) may decrease significantly the accuracy of fingerprinting.

One of the common techniques applied in such conditions is inertial navigation along with a dead reckoning algorithm, however the proprioceptive sensors are not reliable as they their accuracy drifts quickly through time. The fusion of positioning data with the map of the area can bring a solution to this problem. In fact, the localization quality could be improved by the additional layer of information on the road network. This process, known as map matching, identifies the estimated position with the map information according to known criteria in order to improve the estimation accuracy. These criteria can be geometric, such as proximity of the estimated position to the road [15], [16] or topological taking into account the structure of the road network [17]. This last kind of approaches proved to be more accurate since it remedies to many limitations of the geometric algorithms [18]. Other approaches have been applied with different degrees of success such as probabilistic modeling [19], [20], fuzzy logic [21], [22] or belief theory [23]. Refer to [18] for a comparative study of the different map matching algorithms.

The method that we propose in this paper can be classified in the topological map matching category. The vehicle uses only two sensors to calculate its position: speed and heading. The initial position of the vehicle is supposed to be completely unknown. At first, there's a seeking phase that aims to reduce the positioning error to an acceptable level by identifying sequentially the most probable areas of the vehicle's existence. Once the position is found, the method tracks the vehicle mainly by detecting the turn maneuvers. The algorithm can switch back to the seeking phase if it looses track of the correct vehicle's position.

The algorithm uses a particle filter defined in the circular domain. It was initially described in our previous paper [24] as a backup positioning method during GPS masking outdoors and showed interesting performances. This method relies mainly on the vehicle's heading which has the advantage of being an absolute measurement. The aim is to match the measured heading to the roads direction on the map by using a circular statistical distribution.

The structure of this article is as follows. The principle of the circular estimation and the presentation of the localization method are described in Section 2. The experimental setup and results are discussed in Section 3. A conclusion is found in the last section.

II. Particle Circular map matching

A. Circular Bayesian estimation

When estimating angle observations, the von Mises distribution is used because it is considered the equivalent of the normal distribution in the circular domain. The pdf of a von Mises distribution of the variable θ is defined by [25]:

$$f_{CN}(\theta;\mu,\kappa) = \frac{1}{2\pi I_0(\kappa)}\, e^{\kappa\ \cos(\theta-\mu)} \tag{1}$$

μ is the mean of the distribution and κ its concentration parameter. κ is the equivalent of the inverse of a Gaussian distribution variance. I_0 is the modified Bessel function of the first kind and order zero. Figure 1 clearly shows that when the mean μ is not centered on the measurement interval, in this case $[0, 2\pi[$, the bell shape of the distribution is adjusted to the 2π interval to take into account the periodic nature of angle.

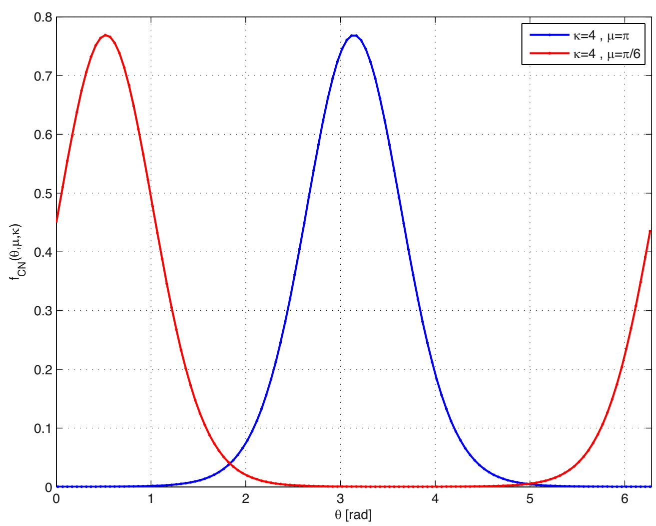

Fig. 1. The pdf of a von Mises distribution

B. Principle of localization

To localize a vehicle using its velocity and direction, we use a dead reckoning system described by the following state model:

$$x(t) = x(t-1) + v(t)\ \Delta t\ \cos(\theta(t)) + \nu_x \tag{2}$$
$$y(t) = y(t-1) + v(t)\ \Delta t\ \sin(\theta(t)) + \nu_y \tag{3}$$

where $v(t)$ and $\theta(t)$ are the velocity and the direction respectively considered as commands. Δt is the time step. ν_x and ν_y are centered Gaussian noises.

To include the map data, the state model is augmented by an additional variable $s(t)$ that stands for the road segment identity. We suppose here that the map database is a set of linear segments identified by their identity index, the coordinates of their ends and their direction β with respect to the horizontal axis. The augmented state model is then:

$$\begin{aligned} x(t) &= x(t-1) + v(t)\ \Delta t\ \cos\left(\beta\left(s(t-1)\right)\right) \\ &\quad + \nu_x \cos(\beta(s(t-1))) \end{aligned} \tag{4}$$
$$\begin{aligned} y(t) &= y(t-1) + v(t)\ \Delta t\ \sin(\beta(s(t-1))) \\ &\quad + \nu_y \sin(\beta(s(t-1))) \end{aligned} \tag{5}$$
$$s(t) = s(t-1) + \nu_r \tag{6}$$

where ν_r is a discrete uniform noise on the road segments index connected to the current segment $s(t)$. It models the error on the segment identification.

The measurement equation is defined by:

$$\phi(t) = \beta(s(t)) + \nu_\phi \tag{7}$$

where ν_ϕ is the measurement noise assumed to follow a von Mises circular distribution. $\phi(t)$ is an observation of direction provided by the on-board magnetometer.

Therefore, we can consider a recursive filter that uses the state equations 4, 5 and 6 for the state prediction. The measurement equation 7 is used to correct the road segment identity. In this case, the estimate of $s(t)$ would be the value that minimizes the filter's innovation $\phi(t) - \beta(s(t))$.

This system has a strong non-linearity in its state model and measurement equations. Therefore we propose to implement it with a circular particle filter. The circular framework is motivated by the angular observations that vary in the interval $]-\pi, \pi]$ [26].

C. Map matching Implementation

The particle map matching algorithm is used with velocity and direction observations [24]. The circular particle filter estimates the vehicle's position on a digital map by matching the vehicle's direction with the roads direction on the map. The particles represent the posterior distribution of the state vector. Each particle has a weight proportional to its probability.

In our scenario, the initial position of the vehicle is unknown, thus, the initial set of particles is generated along all the segments. In the prediction step, we propagate each particle on the road segment at a distance $d = v(t)\Delta t$ from its last position. $v(t)$ is the velocity and Δt the time between two measurements. When a particle reaches the end of a segment, we choose randomly among the segments connected to the current one and we place the particle on it.

In the update step, each particle is affected a weight that is the likelihood of the observed direction $\phi(t)$ compared to the predicted direction $\beta(s(t))$. Consequently a particle that exists on a segment whose direction is similar to the vehicle's direction receives a more important weight and has a higher probability to live.

The implementation of the filter is described as follows:

1) *Initialization*:
Since we have no prior knowledge on the initial position of the vehicle, the particles are placed on all segments separated by the same distance. In case a prior knowledge is available, more particles should

be placed on the considered area since particles stand for probable spots of the vehicle's position.

2) *Prediction*:
Each particle moves following the state model equations described by (4), (5) and (6). It is identified by its coordinates (x_k^i, y_k^i) and the road segment on which it exists S_k^i:

$$\begin{aligned} x_k^i &= x_{k-1}^i + v_{k-1}\ \Delta t\ \cos(\beta(S_{k-1}^i)) \\ &\quad + \eta_k^i\ \cos(\beta(S_{k-1}^i)) \end{aligned} \tag{8}$$

$$\begin{aligned} y_k^i &= y_{k-1}^i + v_{k-1}\ \Delta t\ \sin(\beta(S_{k-1}^i)) \\ &\quad + \eta_k^i\ \sin(\beta(S_{k-1}^i)) \end{aligned} \tag{9}$$

$$S_k^i = \psi(S_{k-1}^i, x_k^i, y_k^i) \tag{10}$$

where:

- η_k^i is a random noise following a centered uniform distribution of variance η_Q. It models the uncertainty on velocity v_{k-1}.
- S_{k-1}^i is the segment index of the particle i at instant $k-1$. The particles are not corrected in position and are propagated according to the noise on the command $v_{k-1}\ \Delta t$.
- $\psi(.)$ is a function that decides whether particle i will stay on the same road segment S_{k-1}^i or not. The segment change occurs when the particle reaches the segment end, *i.e.* the new calculated position (x_k^i, y_k^i) is located outside the current segment S_{k-1}^i. In this case, the function $\psi(.)$ selects randomly one of the segments connected to the current segment S_{k-1}^i in the movement direction.

3) *Update*:
In order to estimate the current road identity, we calculate the weight of each particle with the following von Mises likelihood distribution:

$$\tilde{w}_k^i = w_{k-1}^i f_{CN}\left(\phi_k; \beta(S_k^i), \kappa_P\right) \tag{11}$$

This weight $\tilde{w}_k^i$ is higher for particles moving on roads whose direction $\beta(S_k^i)$ is close to the vehicle's direction ϕ_k. The concentration parameter κ_P controls the particles dispersion around the direction $\beta(S_k^i)$. The adjustment of κ_P is crucial for the algorithm, indeed, a more selective f_{CN} (high values of κ_P) restricts the particles only on the highly probable segments which incurs the risk to uncover some potentially interesting areas, mainly turns. In the other hand, low values of κ_P allow the particles to explore less probable areas which delays the convergence of the algorithm and increases unnecessarily the computational cost.

The weights are normalized as follows:

$$w_k^i = \frac{\tilde{w}_k^i}{\sum_{j=1}^N \tilde{w}_k^j} \tag{12}$$

4) *Position estimation*:
The following weighted sum of the particles gives the position coordinates $(\hat{x}_k^p, \hat{y}_k^p)$:

$$\hat{x}_k^p = \sum_{i=1}^N w_k^i x_k^i \quad ; \quad \hat{y}_k^p = \sum_{i=1}^N w_k^i y_k^i \tag{13}$$

This position is then projected on the nearest road segment $\hat{s}_k$ to generate the estimate of the vehicle's position $(\hat{x}_k, \hat{y}_k)$. Where $\hat{s}_k$ is the estimated road segment index.

5) *Particle resampling*:
We resample the particles according to the Sequential Importance Resampling (SIR) method described in [27] with an effective number of particles defined as follows:

$$N_{eff} = \frac{1}{\sum_{i=1}^N {w_k^i}^2} \tag{14}$$

In the resampling process, we keep unchanged the position (x_k^i, y_k^i) and the segment identity S_k^i of each particle.

Finally, the particle map matching filter at instant k is defined with the following algorithm:

for *i=1:N* **do**
  Draw particle:
  $(x_k^i, y_k^i) \sim g((x_k, y_k)|(x_{0:k-1}^i, y_{0:k-1}^i))$
  Draw segment identity: $S_k^i \sim \psi(s_k|(x_k^i, y_k^i))$
  Compute the particle's weight:
  $\tilde{w}_k^i = w_{k-1}^i\ f_{CN}(\phi_k|\beta(S_k^i))$
end
Compute $\hat{x}_k^p = \sum_{i=1}^N w_k^i x_k^i$; $\hat{y}_k^p = \sum_{i=1}^N w_k^i y_k^i$, for $i = 1, \ldots, N$
Find the nearest segment to $(\hat{x}_k^p, \hat{y}_k^p)$ $\rightarrow$ segment identity estimate $\hat{s}_k$
Project $(\hat{x}_k^p, \hat{y}_k^p)$ on $\hat{s}_k$ $\rightarrow$ position estimate $(\hat{x}_k, \hat{y}_k)$
Normalize weights: $w_k^i = \tilde{w}_k^i / \sum_{i=1}^N \tilde{w}_k^i$, for $i = 1, \ldots, N$
if $1/\sum_{i=1}^N (w_k^i)^2 < N_T$ **then**
  Resample the particles with the SIR algorithm
  Homogenize weights: $w_k^i = 1/N$, for $i = 1, \ldots, N$
end

Algorithm 1: Particle map matching algorithm

Where $g(.)$ is a uniform law, $f_{CN}(.)$ a von Mises law and $\psi(.)$ the function that predicts the segment identity defined by (10).

III. Experimentation

We consider a vehicle such as forklift that moves inside a covered warehouse. The movement is restricted to the road network depicted in figure 2. The segments represent the different corridors and can be vertical or horizontal. Each segment holds a number between 1 to 26.

The vehicle is equipped with two sensors : an odometer and a digital compass for velocity and heading measurement respectively. In this experimentation, we suppose that the vehicle starts from an unknown position then moves through the road network, when it reaches the end of a segment, it chooses randomly to turn left or right. The maximum velocity authorized for a forklift in a warehouse is 8mph or 3.58m/s, we choose a constant velocity of 2m/s. We assume that the error on the velocity measurement is Gaussian and biased. The distribution standard deviation is set to 0.6 m/s and the bias is assumed to be following a uniform distribution between 0.5 m/s and -0.5 m/s.

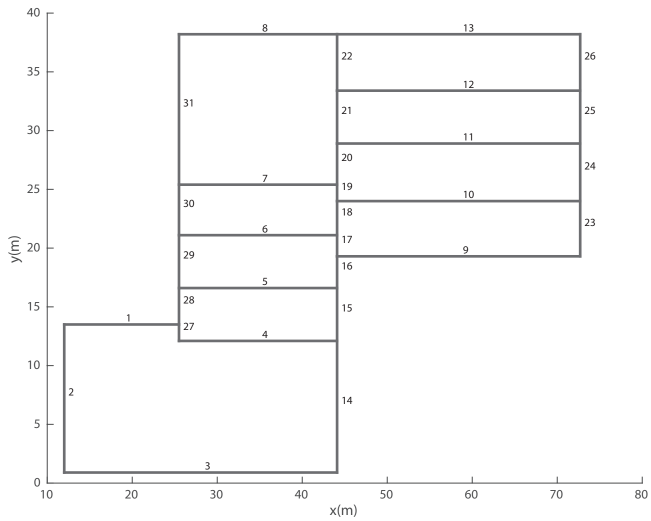

Fig. 2. Warehouse Road network

We run 100 realizations of the algorithm with a random starting point and trajectory. The duration of a simulation is 200s (3min20s). We suppose that the algorithm is in the seeking phase when the positioning error (distance between the real and the estimated position) is above 2m. When it falls below this threshold, the algorithm is in the tracking phase.

First, we are interested in the error of positioning over time for different values of κ_P. This error is depicted in Figure 3. We observe that the error is high in the seeking phase at the start then falls to a quasi-constant value for each κ_P. We also show in Table I the global positioning error which is the mean of the errors for all the simulation interval. We can easily observe that $\kappa_P = 0.5$ and $\kappa_P = 1$ are the best settings. Indeed they have the smallest global positioning error (around 3.3m) and their positioning error falls below 2m after 80s and 120s respectively. For the other values of κ_P the positioning error never reaches 2m because the particles are too dispersed (low κ_P) or too compressed (high κ_P).

κ_P	**10.0**	**5.0**	**1.0**	**0.5**	**0.1**
Global positioning error (m)	13.4	9.1	3.3	3.4	7.2

TABLE I. GLOBAL POSITIONING ERROR AS A FUNCTION OF κ_P

A second important measure of the algorithm's performance is the probability of correct segment identification. We compare the segment number of the real and estimated positions. We affect a score of one when they are the same and zero otherwise. For 100 Monte Carlo runs, we compute the probability of correct segment identification that equals the mean of these scores for each instant. We show in figure 4 this probability for each value of κ_P. We observe that for $\kappa_P = 0.5$ and $\kappa_P = 1$ the probability increases and stays above 0.8 after 120s. For $\kappa_P = 1$, the performances are better during the seeking phase. Table II shows the global probability of correct segment identification which is the mean of the probability during the whole simulation interval of 200s. We also notice that the best value is around 0.7 and is obtained for $\kappa_P = 0.5$ and $\kappa_P = 1$.

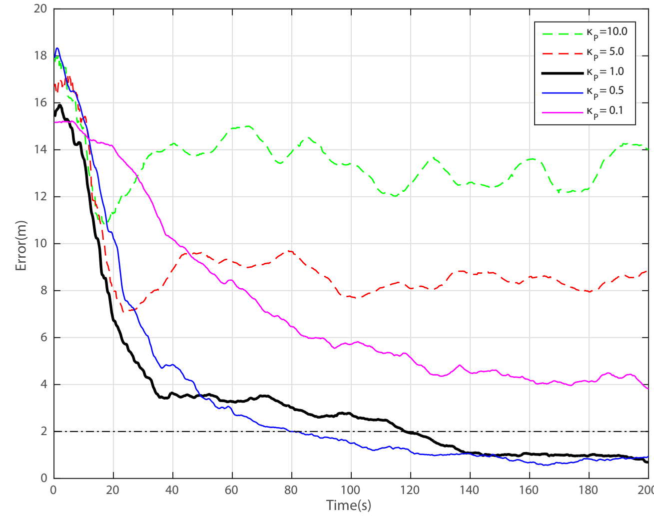

Fig. 3. Position Error over time for different values of κ_P

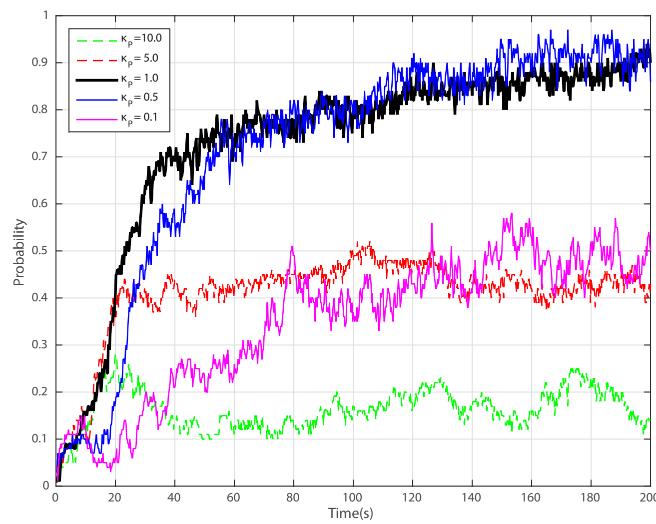

Fig. 4. Probability of Correct segment identification over time for different values of κ_P

κ_P	**10.0**	**5.0**	**1.0**	**0.5**	**0.1**
Probability of indentification	0.16	0.41	0.73	0.72	0.35

TABLE II. GLOBAL PROBABILITY OF SEGMENT IDENTIFICATION AS A FUNCTION OF κ_P

In Figures 5 to 8, we illustrate the evolution of the algorithm during the seeking phase until the correct position is reached for $\kappa_P = 1$. In the initial state (Figure 5), the particles are uniformly distributed on all the segments. In Figure 6, the cloud of particles moves to the right after a movement of the vehicle in this direction. After several turns, we notice in Figure 7 that the particles begin to concentrate on the most probable segments mainly around the real position. Finally, after 54s of movement (Figure 8), most of the particles with less probability are eliminated and recreated around the correct position. The algorithm is now in the tracking phase.

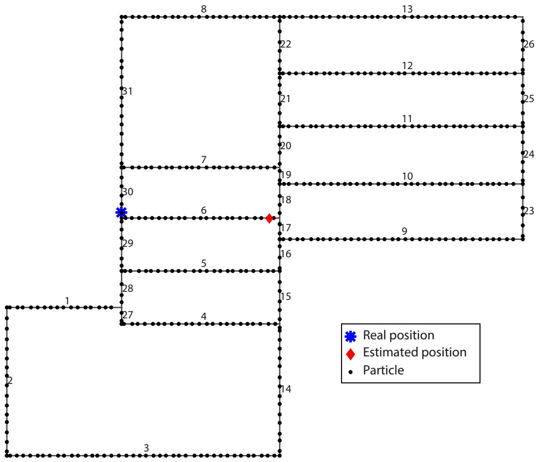

Fig. 5. Initial sate of the algorithm

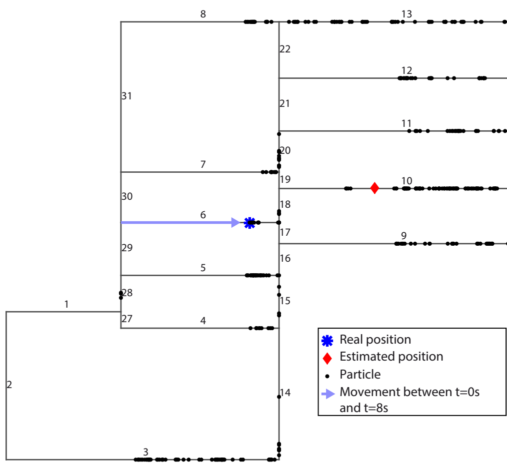

Fig. 6. Evolution of the algorithm between t=0s and t=8s

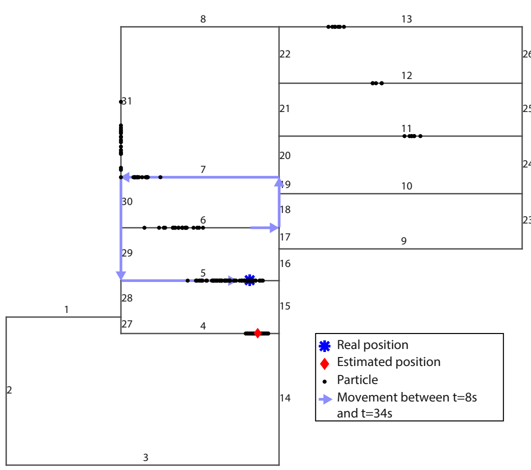

Fig. 7. Evolution of the algorithm between t=8s and t=34s

IV. Conclusion

In this paper we present an approach of indoor localization based on a topological map matching algorithm and a circular particle filter defined in a Bayesian framework. The studied scenario is the localization of a vehicle moving in a covered warehouse on a predefined road network. The vehicle's position is observed with measurements of velocity and heading. The particles represent the posterior distribution of the vehicle's position in a segment. They are propagated by the velocity and their weight calculated by the circular distribution of von Mises that matches the direction of movement with the direction of the road segments.

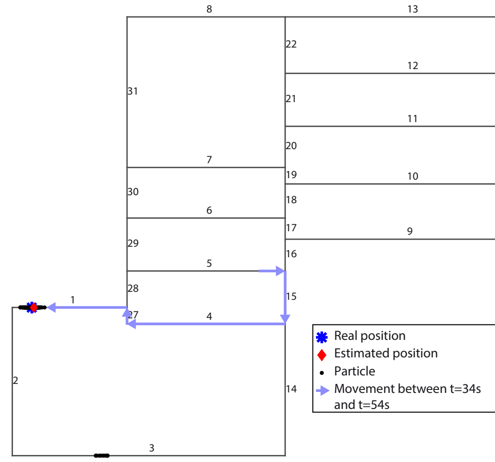

Fig. 8. Evolution of the algorithm between t=34s and t=54s

The experimental results show that after a seeking phase where the positioning error is high, the proposed algorithm is able to localize the vehicle after 80s of movement in average with a positioning error less than 2m. The algorithm switches to the tracking phase where the error is reduced progressively to 1m with a probability of correct segment identification of around 0.9.

Future work will involve the fusion of this method with a second approach of indoor localization like fingerprinting in order to improve the performances mainly reducing the seeking phase duration.

References

[1] C. Beder and M. Klepal, "Fingerprinting based localisation revisited - a rigorous approach for comparing RSSI measurements coping with missed access points and differing antenna attenuations", In 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2012.

[2] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Lightweight map matching for indoor localization using conditional random fields", In The International Conference on Information Processing in Sensor Networks (IPSN'14), Berlin, Germany, 2014.

[3] A. Haeberlen, E. Flannery, A. M. Ladd, A. Rudys, "D. S. Wallach, and L. E. Kavraki. Practical robust localization over large-scale 802.11 wireless networks", In Proc. Intl. Conference on Mobile Computing and Networking (MOBICOM), 2004.

[4] V. Otsason, A. Varshavsky, A. L. Marca, and E. de Lara. "Accurate GSM indoor localization", In Proc. 7th Intl. Conf. on Ubiquitous Computing (UbiComp), 2005.

[5] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason "GSM indoor localization", Pervasive and Mobile Computing, vol. 3, no. 6, pp. 698-720, 2007.

[6] S. P. Tarzia, P. A. Dinda, R. P. Dick, and G. Memik, "Indoor localization without infrastructure using the acoustic background spectrum", In Proc. Intl. Conference on Mobile Systems, Applications and services (MobiSys), pp.155-168, 2011.

[7] Poovendran, R., Wang, C., Sumit R., “Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks”, Springer, 2006.

[8] R. Bruno and F. Delmastro, “Design and analysis of a bluetooth-based indoor localization system”, In Personal Wireless Communications, Lecture Notes in Computer Science, 2003.

[9] R. Want, A. Hopper, V. Falc ao, and J. Gibbons, “The active badge location system”, ACM Trans. Inf. Syst., 10, January 1992.

[10] P. Lazik and A. Rowe, “Indoor pseudo-ranging of mobile devices using ultrasonic chirps”, In Proc. of the 10th ACM Conference on Embedded Network Sensor Systems, SenSys’12, pp. 99-112, New York, USA, 2012.

[11] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, “Landmarc: indoor location sensing using active RFID”, Wireless networks, 10 (6), pages 701-710, Springer, 2004.

[12] A. Matic, A. Popleteev, V. Osmani, and O. Mayora-Ibarra, “Fm radio for indoor localization with spontaneous recalibration”, Pervasive Mob. Comput., 6, December 2010.

[13] Y. Miyazaki, T. Kamiya, “Pedestrian Navigation System for Mobile Phones Using Panoramic Landscape Images”, In Proc. of SAINT’06, IEEE Computer Society, page 102-108, 2006.

[14] A. Mulloni, D. Wagner, D. Barakonyi, I. Schmalstieg, “Indoor Positioning and Navigation with Camera Phones”, IEEE Pervasive Computing, 8(2), pages 22-31, 2009.

[15] D. Bernstein and A. Kornhauser, “Map matching for personal navigation assistants”, Proc. of the 77th annual meeting of the Transportation Research Board, Washington D.C, (1998) 11-15.

[16] C.E. White, D. Bernstein and A.L. Kornhauser, “Some map matching algorithms for personal navigation assistants”, Transportation Research Part C, 8 (2000) 91-108.

[17] J.S. Greenfeld, “Matching GPS observations to locations on a digital map”, Proc. of the 81st Annual Meeting of the Transportation Research Board - Washington D.C., (2002) 554-559.

[18] M. Hashemi and H. A. Karimi, “A critical review of real-time map-matching algorithms: Current issues and future directions”, Computers, Environment and Urban Systems, vol.48, pp.153-165, 2014.

[19] Y. Zhao, “Vehicle Location and Navigation System”, Artech House, Inc., MA., (1997).

[20] M. A. Quddus, “High Integrity Map Matching Algorithms for Advanced Transport Telematics Applications”, Ph.D. dissertation, Imperial College London UK, (2006).

[21] S. Syed and M.E. Cannon, “Fuzzy logic-based map matching algorithm for vehicle navigation system in urban canyons”, Proc. of the Institute of Navigation (ION) national technical meeting, 26-28 January, California, USA, (2004).

[22] M.A. Quddus, R.B. Noland and W.Y Ochieng, “A high accuracy fuzzy logic based map matching algorithm for road transport”, Journal of Intelligent Transportation Systems, 10(3) (2006) 103-115.

[23] G. Nassreddine, F. Abdallah and T. Denœux, “Map matching algorithm using belief function theory”, Proc. of the 11th International Conference on Information Fusion (FUSION ’08), (2008) 995-1002.

[24] K. El Mokhtari, S. Reboul, M. Azmani, J.B. Choquel, S. Hamdoune, B. Amami, M. Benjelloun, “A map matching algorithm based on a particle filter”, International Conference on Multimedia Computing and Systems (ICMCS’14), (2014) 723-727.

[25] S.R. Jammalamadaka and A. SenGupta, “Topics in Circular Statistics”, New Jersey: World Scientific Publ., (2001).

[26] G. Stienne, S. Reboul, M. Azmani, J. B. Choquel and M. Benjelloun, “A multi-temporal multi-sensor circular fusion filter”, Information Fusion, 18 (2014) 88-100.

[27] S. Arulampalam,S. Maskell, N. Gordon and T. Clapp, “A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking”, IEEE Transactions on Signal Processing, 50 (2002) 174-188.