A Regularised Particle Filter for Context-Aware Sensor Fusion Applications

Enrique Martí, Jesús García and José Manuel Molina

{emarti,jesus.garcia}@inf.uc3m.es, molina@ia.uc3m.es

Abstract: Particle Filters are the most suitable filtering techique for some problems where the prediciton and update models are extremely non-linear. However, they suffer some problems as sample depletion which can drastically reduce their performance. There are multiple solutions to this problem. Some of them make assumptions that invalidate the filter for the most difficult scenarios. Some others increase the computational cost far beyond the bounds of real time applications. Context is a very important source of information for those systems that must work flawlessly in changing scenarios, but it introduces strong nonlinearities and uncertainties that filtering algorithms must deal with. This paper analyzes the performance and robustness of a recently developed regularisation technique for particle filters. The proposed scenarios include a navigation problem where a map is used to provide contextual information, because the final target for the particle filter is a mobile robot able to navigate both indoors and outdoors.

1 Introduction

The discipline of Data Fusion is progressively becoming more concerned about context information. This concept embraces a wide variety of factors that are either unforeseen or which can change over time following an unpredictable pattern. In the run for reliable autonomous systems, capable to work flawlessly even for long time periods and under unforeseen conditions, it is of the uttermost importance to build "intelligent" systems which can adapt their functioning to the circumstances in order to deliver the best performance. Context information is the basis for these capabilities.

This work is focused in the more concrete discipline of Sensor Fusion. Taking as starting point the companion paper [MGM11a], where we performed a theoretical analysis about how context information can be applied to a Sensor Fusion system, this paper presents new advances in the experimental part.

The core of our original navigation solution was implemented as a Particle Filter (PF) with loose coupling integration of received information. Amongst the reasons for selecting this tool, we can cite its capability for dealing with non-linear prediction and sensing models. Another important point is that some applications as navigation —specially indoors— result in multimodal probability distributions for the state vector. This discards many filtering techniques such as the Extended Kalman Filter (EKF) [WB95] or the members of Sigma-Point Kalman Filters (SP-KF) [MW04] (e.g the Unscented Kalman Filter (UKF),

erschienen im Tagungsband der INFORMATIK 2011
Lecture Notes in Informatics, Band P192
ISBN 978-3-88579-286-4

[Sim97]). Many extensions based on banks of such filters have been proposed to deal with the multimodality. However, under cases of very non-linear models or process/measure noises, the only effective solutions are the different families of PFs.

The last argument in favor of this Bayesian inference tool is that it requires minimal efforts for integrating new types of sensor measures. As suggested in our previous work [MGM11a], some forms of context knowledge can be fed to the filtering algorithms directly in the shape of sensor measures. This means that by using a PF, some context information will be straightforwardly integrated instead of requiring the specific mechanism which other tools would do.

Nonetheless, PFs are not free of problems. Some problems with special features show what has been called "sample depletion", which is a degeneracy of the filter caused by deficiencies in the algorithm. Under these circumstances, the filter fails to maintain a population of particles that actually describe the real state probability distribution.

Most of the solutions to sample depletion belong to the family of "regularisation techniques". Regularisation consists in switching from the particle-based discrete representation of probability distributions to a continuous formulation, so that the deficiencies that led to the empoverishment of the population get solved. The experimental part of this work aim to validate a novel regularisation scheme which was presented in [MGM11b]. In the previous paper, a literature review was performed, and the theoretical foundations of the algorithm were sent. and performed some general tests, but

Next section will briefly introduce to the problem of sample depletion, and will present the proposed regularisation algorithm. The following four sections are experiments for validating our proposal, including one which analyze the time complexity of the tested regularisation algorithms. The last of the experiments (section 7) tries to reproduce the target scenario for the implemented system: a mobile robot that uses an Inertial Measure Unit (IMU), a positioning sensor and context information to perform navigation. Finally, conclusions are shown and discused in section 8.

2 Regularised Particle Filters

The population of a PF uses a "set of weighted deltas" to approximate the continuous probability distribution of the filtered system state. Arriving observations provide new information that is incorporated as a change in the weight of the particles. The variance of the population grows unbounded over time because of the process noise and other factor. As a result, many particles will represent part of the state space with null interest, and most of the weight of the population will be concentrated in a few samples —a situation comparable to having a PF with a smaller number of particles.

Resampling tries to solve this problem by arranging the samples that compose the population, so that its descriptive power is not degraded over time. With this purpose samples with marginal weight are discarded, and their place is occupied by new samples located in a more interesting place of the state space. A simple and efficient method is to select existing samples with a probability proportional to their associated weight. These samples

erschienen im Tagungsband der INFORMATIK 2011
Lecture Notes in Informatics, Band P192
ISBN 978-3-88579-286-4
weitere Artikel online:
http://informatik2011.de/519.html

can be cloned and perturbed with a random sample of process noise to give a brand new particle distributed according to the expected state posterior probability distribution.

Particle Filters approximate the optimal Bayesian estimation with an accuracy proportional to the employed number of particles. As a rule of thumb, we can say that the approximation will be good enough when the space between neighbor particles has "smooth" properties from the Bayesian point of view. This includes the effects of observations and application of process noise.

Sample depletion is a side effect when the previous rule does not hold. The following list analyzes some of the concrete causes of sample depletion, which will be tested in the experimental section:

- The measurement noise (observation model accuracy) is extremely small compared with the variance of the state pdf. On update, the measures can either invalidate every particle, or concentrate all the weight of the population in very few samples. Under these circumstantes, the resampling step does not have enough information to regenerate the population properly. This problem is analyzed in section 4, and can be seen in figure 4.
- If plant noise is too small compared with particle density in the state space, resampling through particle multiplication is not viable. The introduced noise sample barely varies the new particles, and the space between neighbor samples is never filled. As a result the particles will collapse in dense clusters around the original multiplied samples, causing the filter to diverge in a few cycles. This case will be treated in section 5.

Existing literature provides a number of varied solutions to the resampling deficiency which leads to sample depletion. Some of them calculate a continuous analytical expression for the probability distribution described by particles, a procedure commonly termed as "regularisation". The book [RAG04] goes further and discerns between "regularisation" and "local linearization" techniques, depending on how the analytical expression is obtained, as well as how is it used.

Regularisation techniques approximate the cloud of particles through a weighted sum of kernels —usually Epanechnikov, or the computationally cheaper Gaussian function—, and use that continuous representation to generate the new population. The appropriate set of kernels can be found either by fitting them to the population, for example using an Expectation-Maximization (EM) algorithm. An example is Kernel PFs [CRM03]. Other approach is to attach a kernel to each particle, which is used in [Ans05] for object tracking in video.

Local linearization techniques approximate the population using a bank of Kalman-like filters. An example is the already cited Unscented Particle Filter (UPF) [MDdFW00], which uses an UKF per particle that, in words of the authors, will propagate the sufficient statistics for each particle. The difference with the more simple regularisation schemes is that prediction and/or update can be performed directly on the bank of filters. This last idea is used in [MW04] to refine the idea of the UPF, reducing the computational burden of running N UKFs. In order to achieve such a substantial performance gain without

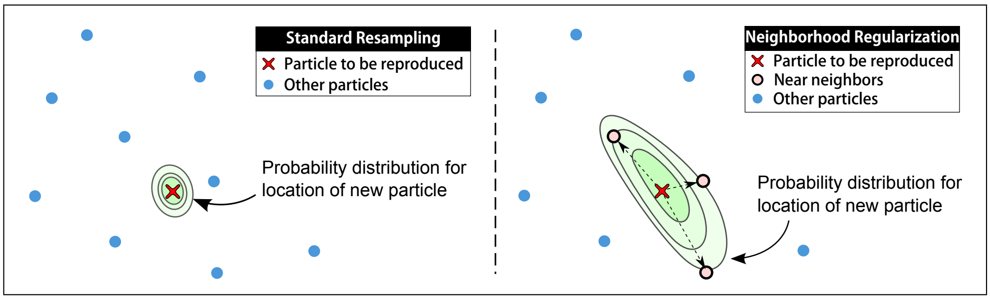

Figure 1: Neighborhood-based regularization distributes new samples more evenly in the state posterior distribution, based on information taken from local particles

loosing capbilities, the population (N particles), process noise and measure noise must be simplified to a sum of G, P and M Gaussians respectively. Assuming that $G \cdot P \cdot M \ll N$, its is computationally cheaper to perform prediction and measure update steps applying Kalman equations to all the $G \cdot P \cdot M$ combinations of Gaussian components, rather than sampling particles from them and applying the usual PF process. Particles are used only during resampling.

3 Proposal and compared algorithms

This section reviews our proposal: the Neighborhood-Based Regularization (NBR-PF) method that was introduced in [MGM11b]. It will be compared with an already existing regularisation scheme, the Gaussian Sum Particle Filter (GS-PF) [YWSM10]. It combines a PF with a Gaussian Mixture Model (GMM) for the resampling stage, obtaining results over UKF and SIR-PF.

In our algorithm the resampled particles are generated according to the expected smoothed distribution, which is calculated in a case-by-case fashion using individual samples in a vicinity —instead of fitting a continuous probability distribution to the population. The NBR algorithm obtains samples from the original population by applying a traditional resampling technique —stratified, systematic, residual resampling—. The resulting particles are moved around in their surrounding space, so that they are more effectively distributed in the expected posterior state distribution, as shown in figure 1. This approach ensures that not only the "macroscopic" features of the original probability distribution are respected, but also that the finer detail is preserved. We encourage to review the original paper for additional details.

The presented algorithm smooths the space between particles using local information, so that zones with a high likelihood (that will usually be densely populated) will offer a finer detail. On the other hand, areas of low interest (described by just a few particles) will suffer a more aggresive smoothing. This approach deals right with uneven particle densities and multimodal distributions, whithout incurring on artificial increments of population

ISBN 978-3-88579-286-4

variance.

The main drawback of this technique is the increased computational load. While sampling a whole population has linear complexity —$\mathcal{O}(n)$, since it requires constant time for each particle—, the algorithm proposed in this paper is near-quadratic, given that generating a new particle from an original candidate involves searching its nearer neighbors. However, a complexity of $\mathcal{O}(n^2)$ is less than the $\mathcal{O}(n^3)$ of some SP-PF. The speed comparison between the proposed NBR-PF and the GS-PF is covered in the next sections.

4 Time series

In [MDdFW00], a synthetic problem is proposed to measure the robustness of filtering algorithms against sample depletion. The unidimensional system evolves according to the following formulation:

$$x_{t+1} = 1 + sin(\omega \pi t) + \phi_1 \cdot x_t + \nu_t \tag{1}$$

Where $\omega = 4e - 2$ and $\phi_1 = 0.5$ are parameters of the series, and $\nu_t \sim \mathcal{G}(3, 2)$ is the process noise. The observation model is:

$$y_t = \begin{cases} \phi_2 \cdot x_t^2 + n_t, & t \leq 30 \\ \phi_3 \cdot x_t - 2 + n_t, & t > 30 \end{cases} \tag{2}$$

With $\phi_2 = 0.2$ and $\phi_3 = 0.5$, and $n_t \sim \mathcal{N}(0, 1e - 5)$ the measure noise.

Montecarlo algorithms are tested with a population of 200 particles. The large variance of the process noise forces the particles to spread over a very large area (over 20 units on the Y axis). On update, the measures are so accurate that in most cases even the nearest particle will lay at several standard deviations from the observation. Under this circumstance, the filter will not be able to distinguish between particles “near” the true state and those that are too far away. This is because all the weights will be zero because of the numeric limits of floating point representation.

This effect can be seen in figure 4. In spite that the population of particles is properly distributed and the density of particles is quite high around the real state (left plot), the narrow observation error makes difficult for particles to fall close in observation space. Right plot shows that particle density (samples represented by blue circumferences) is clearly insufficient.

This problem has no straightforward solution, as increasing the number of particles can be countermeasured by further reductions of observation noise. Local linearization techniques, as [MDdFW00] (that uses Unscented Particle Filters) can deal with this problem. However, this is not a valid approach in the general case without further assumptions or simplifications, given than the Unscented Particle Filter requires the observation noise to be Gaussian.

Our proposal does not avoid this depletion problem properly. In fact, the three compared approaches (SIR-PF, NBR-PF and GS-PF) obtain a similar MSE of about 10 units. These

erschienen im Tagungsband der INFORMATIK 2011
Lecture Notes in Informatics, Band P192
ISBN 978-3-88579-286-4
weitere Artikel online:
http://informatik2011.de/519.html

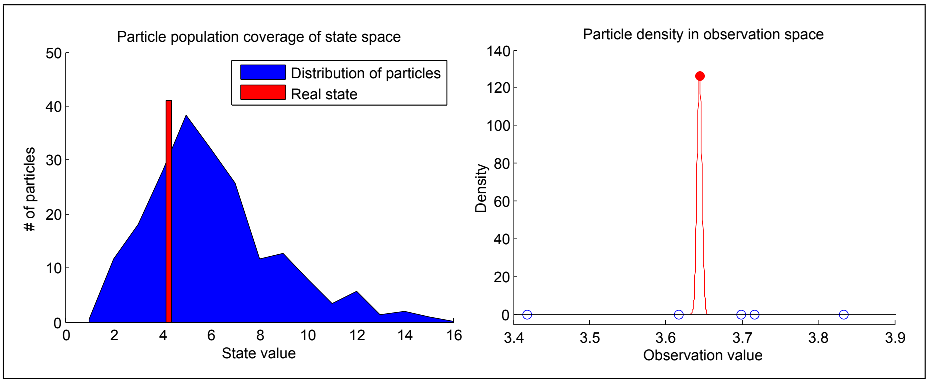

Figure 2: Sample depletion problem caused by very accurate measures (too peaked observation model error)

numbers do not match with the $MSE = 0.4$ anounced in [MDdFW00] for the SIR-PF, so it is difficult to establish a direct comparison with other algorithms. Nonetheless, we can conclude that the proposed regularisation technique delivers the same performance of SIR-PF and Gaussian Mixture Particle Filter.

5 UAV navigation

The second problem consists in estimating the position, speed and attitude of an aerial vehicle equiped with an Inertial Measure Unit (IMU) and a GPS sensor [CDPV06].

We have developed a basic simulation process that generates realistic flight trajectories from an input consisting on forces and angular momenta in the body frame of the air vehicle. The simulator is based in MATLAB$^{\text{TM}}$Aerosim Aeronautical Simulation Block Set, and employs the equations of a rigid body with six degrees of freedom (6DoF in advance) and inertia.

The test trajectories correspond to the most relevant situations in Air Traffic Control, such as straight flight at constant speed, straight flight with longitudinal acceleration and race-tracks (a rectangle with two semicircles attached to its shorter sides, performed during the waiting time before landing in order to fit with the time scheduled).

IMU measures are interpreted as control inputs that are used in the prediction phase, because they describe how the state has to be propagated, i.e. how the state changes. The prediction equation is defined as: $x_{t+1} = f(x_t, q_t, u_t, \Delta t)$, where q_t is the process noise, $u_t = \begin{bmatrix} acc_t & \omega_t \end{bmatrix}$ describes the control input composed by IMU measures from accelerometer and gyroscope, and Δt is the time elapsed since last prediction/update performed over the PF.

This problems exposes the second cause of sample empoverishment cited in 3, this time caused by a defficiency of the resampling process in SIR-PFs. The standard strategy for re-

ISBN 978-3-88579-286-4

Table 1: Time spent in resampling

	SIR-PF	Auxiliary PF	Regularized PF	GS-PF
Time (in seconds)	8.46 s	251.2 s	72.8 s	78 s
Part of total time	2.6%	44.2%	18.7%	19.7%

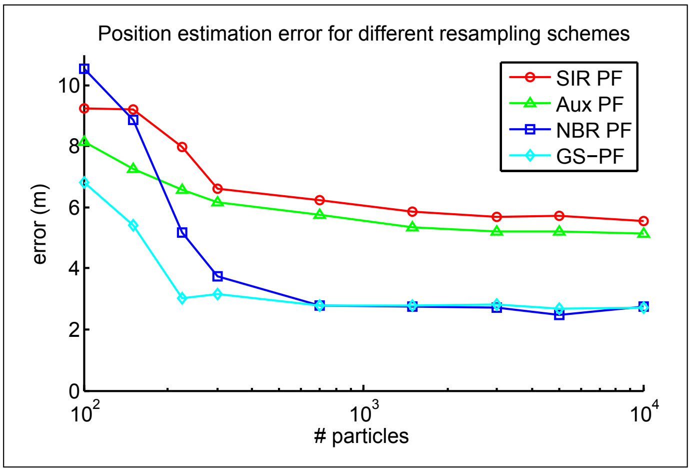

Figure 3: Average position estimation error over number of particles for different resampling strategies)

sampling, which consists on multiplying particles proportionally to their weight, produces almost perfect clones that do not fill the space between near samples. Improvement over standard PF is clear in this problem. Figures 3,4 show that the proposed NBR-PF performs worse than GS-PF with a low number of particles, while above 500 particles both behave similarly.

Table 1 completes that one in [MGM11b] with the results for the GS-PF. It shows raw time and percentage of total execution time spent in resampling when filtering a 400 second long trajectory using a population of 1500 particles. While being clearly more expensive than SIR-PF, they perform much better under every circumstance.

6 Complexity analysis

Regarding the performance results in the las section, it is important to note that the state posterior in the UAV problem is very simple due to the smoothness of the trajectories and the observations having Gaussian noise. As a result, the GS-PF does only require 1-2 components to approximate distribution, and the resampling times are comparable to those of the NBR-PF.

In this section, we conduct some experiments to find the empirical time complexity of the

erschienen im Tagungsband der INFORMATIK 2011
Lecture Notes in Informatics, Band P192
ISBN 978-3-88579-286-4
weitere Artikel online:
http://informatik2011.de/519.html

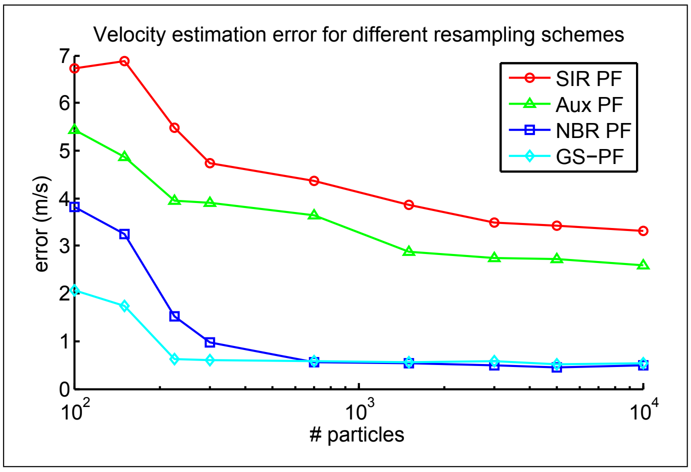

Figure 4: Average speed estimation error over number of particles for different resampling strategies)

different resampling algorithms. In the case of SIR-PF and Auxiliary PF, the resampling time is linear in the number of particles. The tests focus on the regularisation strategies: GS-PF and NBR-PF.

The GS-PF has to find on each cycle a sum of gaussian components representing the population of particles. The usual algorithm can be alleged to be nearly linear in the number of components $\mathcal{O}(k)$ —where k is the number of components of the Gaussian Mixture Model. However, our implementation determines the optimal number of components automatically, starting from one and incrementing it as long as the new number of components appears to describe better the set of particles using Akaike Information Criterion [Aka74]. This process takes the same time as solving the problem once with $k \cdot (k/2) \approx \mathcal{O}(k^2)$ components. Implementing a more advanced search scheme can reduce the time to $\mathcal{O}(k \cdot \ln(k))$.

NBR-PF complexity is only related with the number of particles, remaining unaffected by the abruptness of the state posterior. However, its running time is expected to be quadratic on the size of the population, mainly due to the complexity of searching the neighbors of the particles to be resampled. As remarked in our previous paper, this factor is not easy to reduce because space partitioning techniques are effective only when $n \gg 2^D$, being n the number of samples and D the dimensionality of the explored space.

Figure 5 describes resampling time of NBR and GS strategies. It also contains running time of GS-PF when the number of components k is known beforehand —optimal but unreal case—. In the case of a GS-PF algorithm with automatic selection of number of components using an advanced search scheme, the time is expected to be fall slightly above that of NBR-PF. The numbers have been calculated using a dataset that makes the —rather realistic— assumption that the required number of particles is directly proportional to the complexity of the state posterior. This complexity is related with the number of components needed by the GS-PF to represent the posterior. In our tests the amount of

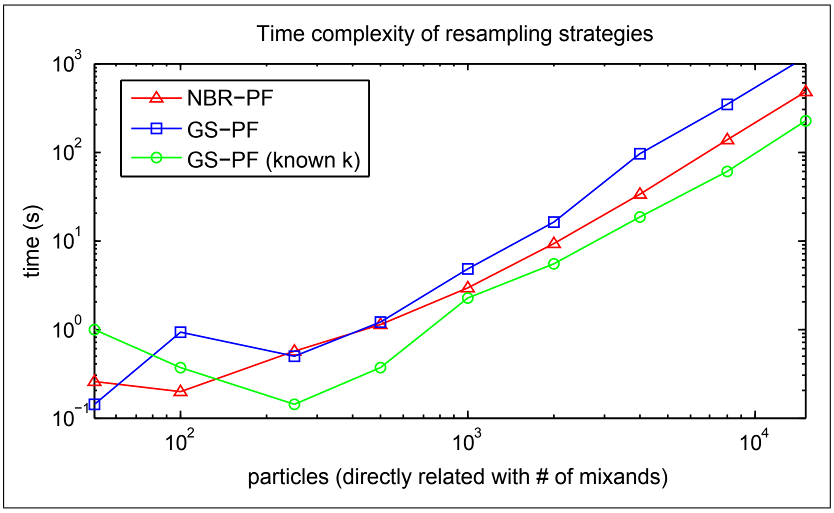

Figure 5: Resampling running time for different posterior sizes and complexities

mixands varies from 1 in the case of 50 particles to 15 when the population is composed by 15000 samples.

7 Ground vehicle map navigation

The last experiment is a solution for 2D navigation of an autonomous vehicle, using an IMU, a positioning device (currently a GPS) and a map of the terrain that provides basic context information. The vehicle is a wheeled robotic platform with two degrees of freedom —translation along its longitudinal axis and rotation—, although the state vector includes the possibility of lateral translation: $x = (pos_x pos_y vel_x vel_y \alpha)$ where α is the attitude of the platform.

Prediction model makes use of IMU measures, and the update step integrates the information provided by either positioning systems and the map. These types of “measures” arrive asynchronously, but the PF does not impose any restriction on the subject.

The experiments in this paper have been reproduced in laboratory from both simulated and real data. The real data has been obtained in controlled experiments where the robot was been equipped with a GPS sensor with meter-level precision and an inertial measurement unit (IMU). With simulation purposes, GPS measures are assumed to suffer a random gaussian-like noise with standard deviation of 1 meter. Figure 6 shows a sample trajectory with obstacles . The baseline navigation algorithm relies in a Particle Filter which performs loosely coupled fusion of the two proposed sensors. This approach will be compared with a similar system that also includes the information of a map. The map is used to discard particles that move into a wall, as done in [EMNR05].

erschienen im Tagungsband der INFORMATIK 2011
Lecture Notes in Informatics, Band P192
ISBN 978-3-88579-286-4

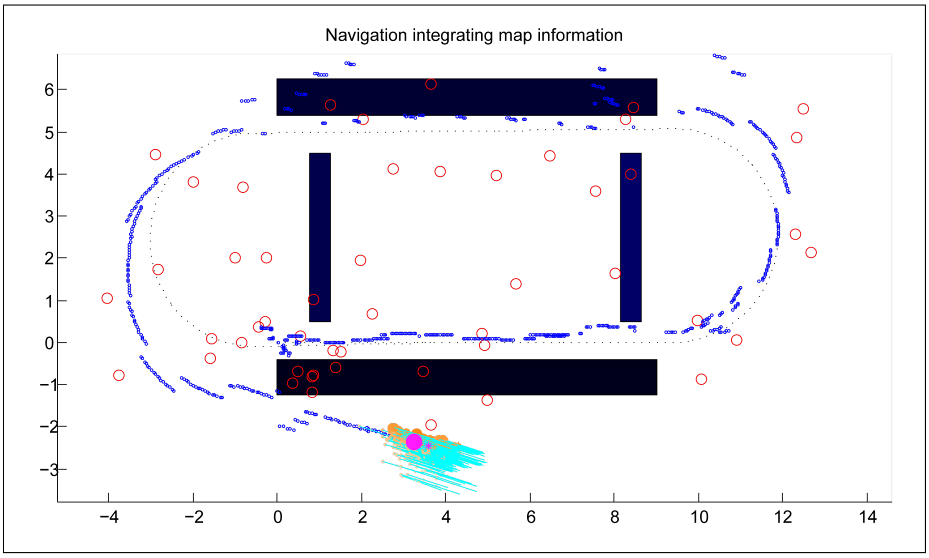

Figure 6: Navigation with a map providing context information. The walls anihilate particles that tend to bias the prediction)

The performance of SIR-PF, NBR-PF and GS-PF is comparable, showing an average 60% improvement in location accuracy over raw GPS estimates. The only difference resides in execution time, as expected. In the sample trajectory, the GS-PF never requires more than 5 components —maybe due to the simplicity of the map—, making it slightly faster than the proposed resampling algorithm. This is expected to change in further experiment with larger maps.

8 Conclusions

This work joins two lines of research we started some time ago: low level filtering algorithms and context aware sensor fusion. The recently introduced Neighborhood Regularisation scheme for Particle Filters [MGM11b] is extensively tested. It is compared with another modern regularisation algorithm, demonstrating a similar performance with very moderate computational requirements. When the state posterior is unimodal and reasonably Gaussian-shaped, GS-PF is faster and requires a lower number of samples. However, in some problems as indoor navigation the computational cost of NBR-PF is better bounded than other approaches as GS-PF. Regarding the conclusions about introducing context in sensor fusion processes, this paper represents a step forward in the construction and validation of the architecture proposed in [MGM11a].

erschienen im Tagungsband der INFORMATIK 2011
Lecture Notes in Informatics, Band P192
ISBN 978-3-88579-286-4

References

[Aka74] Hirotugo Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[Ans05] R. Ansari. Kernel particle filter for visual tracking. *IEEE Signal Processing Letters*, 12(3):242–245, March 2005.

[CDPV06] F Caron, E Duflos, D Pomorski, and P Vanheeghe. GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects. *Information Fusion*, 7(2):221–230, 2006.

[CRM03] D Comaniciu, V Ramesh, and P Meer. Kernel-based object tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(5):564 – 577, 2003.

[EMNR05] F Evennou, F Marx, E Novakov, and Rd. Map-aided indoor mobile positioning system using particle filter. *2005 IEEE Wireless*, 2005.

[MDdFW00] Rudolph Van Der Merwe, Arnaud Doucet, Nando de Freitas, and Eric Wan. The Unscented Particle Filter. Technical report, Cambridge University Engineering Department, 2000.

[MGM11a] Enrique D. Martí, Jesús García, and José M. Molina. Context-awareness at the service of Sensor Fusion Systems: Inverting the Usual Scheme. In *Proceedings of the ISCIF Conference*, 2011.

[MGM11b] Enrique D. Martí, Jesus García, and José M. Molina. Neigborhood-based Regularization of Proposal Distribution for Improving Resampling Quality in Particle Filters. In *14th International Conference on Information Fusion (Fusion 2011)*, page 7, Chicago, USA, 2011.

[MW04] Rudolph Van Der Merwe and Eric Wan. Sigma-Point Kalman Filters For Integrated Navigation, 2004.

[RAG04] Branko Ristic, Sanjeev Arulampalam, and Neil Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech Print on Demand, 2004.

[Sim97] Jeffrey K. Uhlmann Simon J. Julier. A New Extension of the Kalman Filter to Nonlinear Systems. 1997.

[WB95] Greg Welch and Gary Bishop. An Introduction to the Kalman Filter, 1995.

[YWSM10] Shaodong Yang, Desheng Wen, Jing Sun, and Junyong Ma. *Gaussian sum particle filter for spacecraft attitude estimation*. IEEE, 2010.

erschienen im Tagungsband der INFORMATIK 2011
Lecture Notes in Informatics, Band P192
ISBN 978-3-88579-286-4