# A PROBABILISTIC MULTIMODAL ALGORITHM FOR TRACKING MULTIPLE AND DYNAMIC OBJECTS

**MARTA MARRÓN, ELECTRONICS. ALCALÁ UNIV. SPAIN marta@depeca.uah.es**
**MIGUEL A. SOTELO, ELECTRONICS. ALCALÁ UNIV. SPAIN sotelo@depeca.uah.es**
**J. CARLOS GARCÍA, ELECTRONICS. ALCALÁ UNIV. SPAIN. jcarlos@depeca.uah.es**

## ABSTRACT

The work presented is related to the research area of autonomous navigation for mobile robots in unstructured, heavily crowded, and highly dynamic environments. One of the main tasks involved in this research topic is the obstacle tracking module that has been successfully developed with different kind of probabilistic algorithms. The reliability that these techniques have shown estimating position with noisy measurements make them the most adequate to the mentioned problem, but their high computational cost has made them only useful with few objects. In this paper a computational simple solution based on a multimodal particle filter is proposed to track multiple and dynamic obstacles in an unstructured environment and based on the noisy position measurements taken from sonar sensors.

**KEYWORDS:** Extended particle filters, multiple objects tracking, multimodal probability distributions, multiple hypotheses, dynamic unstructured and crowded environments.

## 1. INTRODUCTION

The origin of the probabilistic estimators arrives quite early on the fifties, with the idea of representing the state vector to predict its probability distribution, and applying this idea not only to the area of tracking in robot navigation. There were many advantages in the so-called Bayesian method (also known as Sequential MonteCarlo) from the stochastic ones: the system model should not necessary be linear, and the noise coupled to measurements should not necessarily be Gaussian.

The standard particle filter (PF) is a sampling weighted representation of the Bayesian filter, where each one of the samples taken from the continuous probabilistic distribution is called particle. The set of particles must be independent and identically distributed to achieve a correct approximation to the continuous distribution, but this can be easily solved by using a large enough number of randomly acquired particles. These techniques are not extensively used until the end of the 90s in the area of interest [1], with the introduction of a selection step in the PF loop to avoid the degeneration of the algorithm with time [2]. The idea consists on selecting (or resampling) and multiplying the particles with high importance weights and rejecting the rest. Different alternatives for this part were also designed [3].

To achieve a multiple objects tracker, different options have also been designed during the last years [4]. An initial solution is to use a standard PF to track each object but this is not efficient as it does not work with a dynamic number of objects. Some other solutions include an association among the detected objects and the particles of the filter over the time (JPDAF) [5]. This is not done in the application of interest as long as these techniques are not the most appropriated here.

## 2. DESCRIPTION OF THE DESIGNED ALGORITHM

The dynamic and multi-object tracker designed for the application mentioned is based on a PF, thus in the following paragraphs both the standard algorithm and the improvements made will be described.

## 2.1 The standard PF

The main loop of a standard PF at time t starts with a set $S = \{s_i / i = 1..N\}$ of random particles representing the posterior distribution of the state vector to be estimated $p(\vec{x}_{t-1}|\vec{y}_{t-1})$ at the previous time step (t-1). These particles are propagated by the system dynamics to obtain a new set $S'$ that represents the prior distribution of the state vector at time t, $p(\vec{x}_t|\vec{y}_{t-1})$. The weight of each particle $W = \{w_i / i = 1..N\}$ is then obtained based on the comparison of the measured output vector and the estimated one based on the prior estimations. Applying the selected resample scheme, a new set $S''$ is obtained with the most probable particles that will constitute the new $p(\vec{x}_t|\vec{y}_t)$ at time t.

The functionality of the algorithm is shown in figure 1. See [6] for a detailed explanation.

1. *Initialize (only at t=0):* Obtain the sample set $S$ with N particles from the posterior distribution $p(\vec{x}_{t-1}|\vec{y}_{t-1})$ (for t-1=-1, this will be done with the prior distribution $p(\vec{x}_0)$ )
2. *Propagate:* Obtain the new prior set $S'$ with the model of the state vector to estimate:
$$\vec{x}_t = f(\vec{x}_{t-1}, \vec{v}_{t-1}) \quad \Rightarrow \quad S' = f(S, V), \qquad (1)$$
where $\vec{v}_{t-1}$ is the noise signal associated to the dynamics of the state vector model, and from which V is obtained as a vector of standard random variables with the statistics that define $\vec{v}_{t-1}$.
3. *Importance Sample:* Calculate the importance weights of $S'$ set from the likelihood $p(\vec{y}_t|\vec{x}_t)$:
$$W_t = W_{t-1} \cdot \frac{p(\vec{y}_t|\vec{x}_t) \cdot p(\vec{x}_t|\vec{x}_{t-1})}{q(\vec{x}_t|\vec{x}_{0..t}, \vec{y}_{0..t})} \quad \Rightarrow \quad W_t = W_{t-1} \cdot p(\vec{y}_t|\vec{x}_t), \qquad (2)$$
where $q(\vec{x}_t|\vec{x}_{0..t}, \vec{y}_{0..t})$ is the best proposal approximation of the posterior distribution. In most works this best approximation is substituted by the prior one, so the Eq. (2) can be simplified.
4. *Resample*: Multiply/reject samples of $S'$ with high/low importance weights respectively to obtain a posterior distribution $p(\vec{x}_t|\vec{y}_t)$ represented by a new set $S''$ with N particles.
5. *Output:* The final estimated vector state is usually obtained calculating the mean of the posterior distribution.
6. *Go to step 2…:* The sample set $S''$ from the posterior distribution $p(\vec{x}_t|\vec{y}_t)$ at time t is then used as the new set $S$ for the algorithm implementation in the next step, at time t+1.

*Figure 1. Description of the standard PF functionality.*

## 2.2 The multimodal PF

The standard PF estimates quite well the evolution of any kind of single object defined by its model, but as it was already mentioned at the introduction of this paper, it has not been designed to track a multiple and variable number of them. To do so, different solutions depending on the final application have been proposed as it has already been explained in the introduction. The most interesting of them for the work proposed in this paper is the one presented in [7], because with a single probability distribution a variable number of objects can be tracked with high reliability and with no need of doing a previous association between the different measurements and the particles of the distribution.

The most important innovations that were implemented at [7] to adapt a standard PF to a multimodal estimator are the following:

***Re-initialization:*** In the standard PF a new appearing object is not going to be considered unless its state vector is close enough to an already existing one, as the output vector does not modify directly the sample set of the estimated state vector itself but only their importance weights.
To solve this problem a re-initialization of the sample set $S$ at each time step has to be done, inserting on it M samples directly from the output vector. With this modification, information from the new environmental configuration $p_m(\vec{x}_{t-1}|\vec{y}_{t-1})$ is combined with the posterior distribution $p_p(\vec{x}_{t-1}|\vec{y}_{t-1})$ to obtain a new expression for it:

$$p(\vec{x}_{t-1}|\vec{y}_{t-1}) = \gamma \cdot p_m(\vec{x}_{t-1}|\vec{y}_{t-1}) + (1-\gamma) \cdot p_p(\vec{x}_{t-1}|\vec{y}_{t-1}), \qquad (3)$$

where $\gamma$ is the factor that weights the distribution association up, and that is fixed by the relation between the M samples inserted directly from the output vector measured at t-1 and the total number of samples (N) in the particle set $S$ ($\gamma = {}^{M}\!/_{N}$).
With this new initialization the single probability distribution will adapt itself over time to finally represent simultaneously the state vector of all the different objects that exist in the environment at each time, without the risk that the particles related to new objects in $S$ disappear in the resample stage.

***Resample:*** To insert the new M particles as mentioned, the resample phase is also modified. In this case, only N-M samples have to be selected from the N existing at the $S'$ sample set. The resampling process, as well as the rest of the PF algorithm is for the rest equal to that one of the standard PF.

This version of the extended PF works quite well if all objects are sensed with more or less the same accuracy, but the authors of [7] explain that if it does not occur (as it can happen easily working with ultrasound sensors) the sample set may degenerate as the related weights can be much larger for some objects than for some others. To solve this problem different improvements have been made to this extended PF in the algorithm proposed in this paper.

## 2.3 The clustering algorithm

A clustering algorithm has been designed to organize the measurements that come directly from the sonar in k detected objects. This process is based on a standard kmeans algorithm [8], but some improvements have been included to adapt it to its specific use:

***Standard kmeans:***

a. Select randomly k centroids for the clusters.
b. While the distance from each measurement is not minimum to its assigned cluster centroid.
c. For 1 to all measurements: assign it to the cluster whose centroid is the nearest.
d. If the distance from the measurement to any centroid is bigger than a limit, create a new cluster, whose centroid is the measurement itself (k=k+1).
e. Recalculate all cluster centroids using the mean.
f. If a cluster is empty or has very few members it is erased (k=k-1).

***Cluster movement estimation:*** Instead of assigning randomly the initial centroids, they are obtained from the previous clustering process, so that the algorithm is shorter as the clusters to find are slightly predefined. The movement of the cluster can be estimated calculating the dynamics of its centroid.

***Cluster candidate:*** When a new cluster is created (it has not been calculated from an initial centroid) it is converted into a candidate that is not validated to be useful in the probabilistic algorithm until it is possible to follow its evolution with its related dynamics for a variable number of times. The same process is used to remove a cluster. This method ensures the robustness of the probabilistic estimator to spurious measurements and so increments its reliability. The information obtained from the clustering algorithm is used in different parts of the PF loop incrementing the possibilities and reliability of the probabilistic estimator, as follows:

***At the re-initialization step:*** With a cluster organization it is possible to select the measurements to be inserted in the prior distribution $p(\vec{x}_{t-1}|\vec{y}_{t-1})$ at the re-initialization step, according to their preliminary object assignation (in general M/k measurements from each cluster). As newly inserted particles are chosen randomly from groups with high level concentration of measurements their likelihood is very high from the beginning. This fact prevents from situations in which particles related to objects poorly sensed are erased from the multimodal distribution at the resampling step, as occurred in [7]. The M/k particles to be inserted from each cluster are completed with some others randomly selected from its history buffer, which contains measurements assigned to each cluster in previous time steps and that are not very distant from its actual centroid. New particles taken from the history buffer make the estimation more stable.

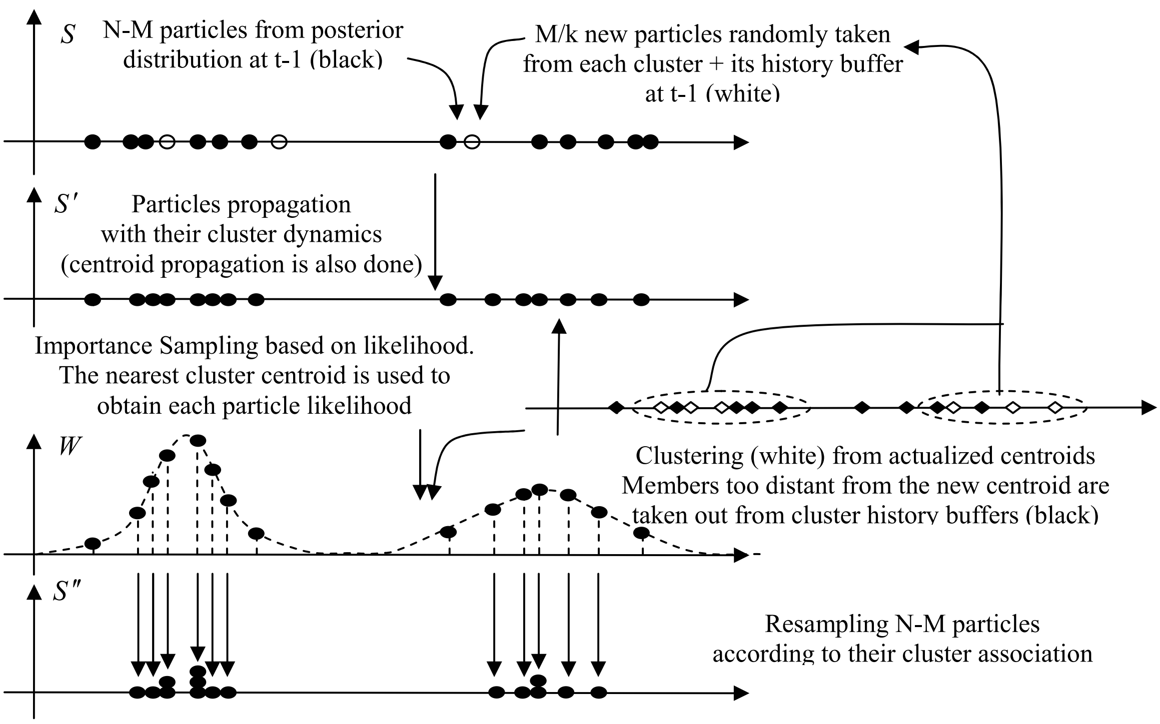


*Figure 2. Description of the functionality of the extended PF designed.*

***At the propagation and importance sampling step:*** The cluster structure is used to make particles evolution and likelihood calculus according their one most similar cluster model and output vector respectively. With this method the predicted sample set $S'$ is going to be very close to the real state vector, obtaining high values for the likelihood function $p(\vec{y}_t|\vec{x}_t)$ at the importance sampling step, and thus improving the robustness and reliability of the global estimator.

***At the resampling step:*** The cluster information can be used to do a dynamic assignment of the M-N particles to resample among the k different clusters detected, and according to their likelihood too. This fact also prevents from the situations of objects poorly measured whose related particles are removed from the posterior distribution, as previously mentioned.

***At the output step:*** The clustering algorithm is executed again at the end of the extended PF, and this time over the $S''$ sample set, using the centroids of the obtained clusters as the final estimated state vector, that is the output of the extended PF at each time step. Figure 2 shows the functionality of the extended PF designed, including the clustering algorithm.

## 3. RESULTS

Different tests have been developed with a robotic platform (from ActivMedia Robotics), in a dynamic environment. The robot has 16 sonar all around its body with a 3m range. In one of the tests the robot has been wandering around by an unstructured environment with diverse obstacles appearing in the scene, in a manual driving mode. Next figures 3 and 4 show different moments of the experiment, each one with four graphics with the following meaning:

- ***Left upper corner:*** actual 3D density function of the occupied space represented by the particle set in the x-y space.
- ***Right upper corner:*** representation of the robot actual situation in the environment. The circles show validated clusters centroids generated by the output clustering process.
- ***Left lower corner:*** accumulated representation of the centroids from the output clusters, all along the covered path. Different identified clusters are represented with different shapes for their centroid point. The darker line shows the robot position all along the path.
- ***Right lower corner:*** actual histogram of the particle set state vector components. The obstacle positioning model is based on the state vector: $\vec{x}_t = [x_t \quad y_t \quad vx_t \quad vy_t]$, and only its two first components are shown here. The different groups that can be distinguished at the histograms would result in different clusters (different distinguished obstacles) at the output clustering process.

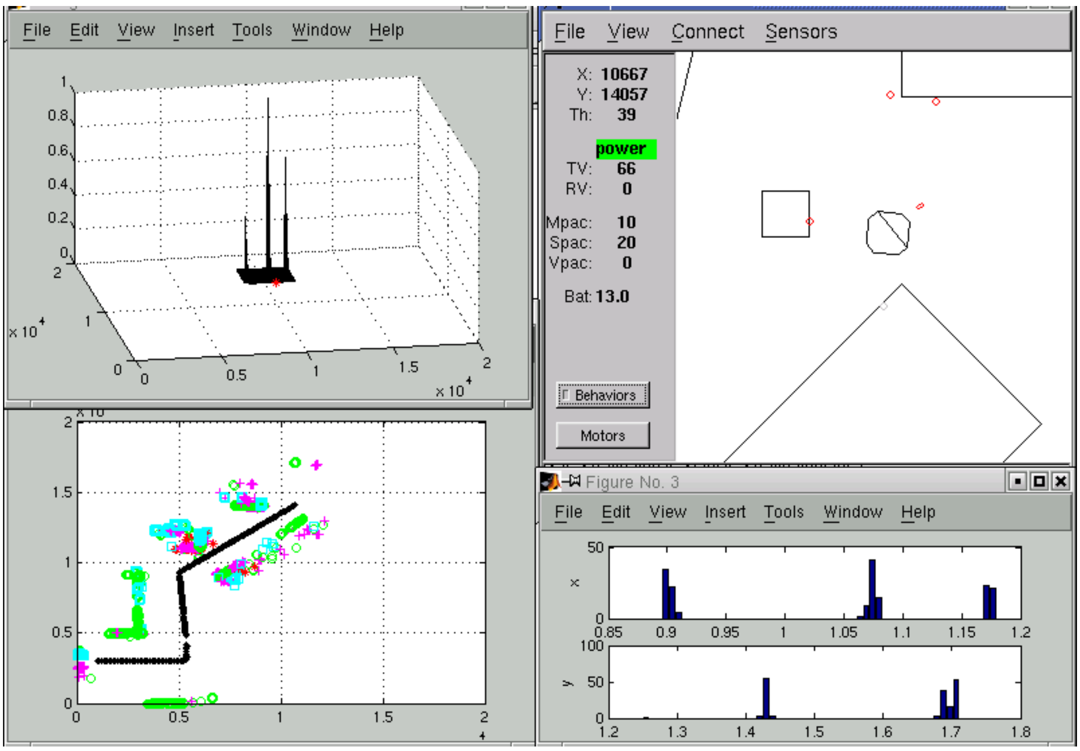


*Figure 3. Results from the probabilistic tracker developed.*

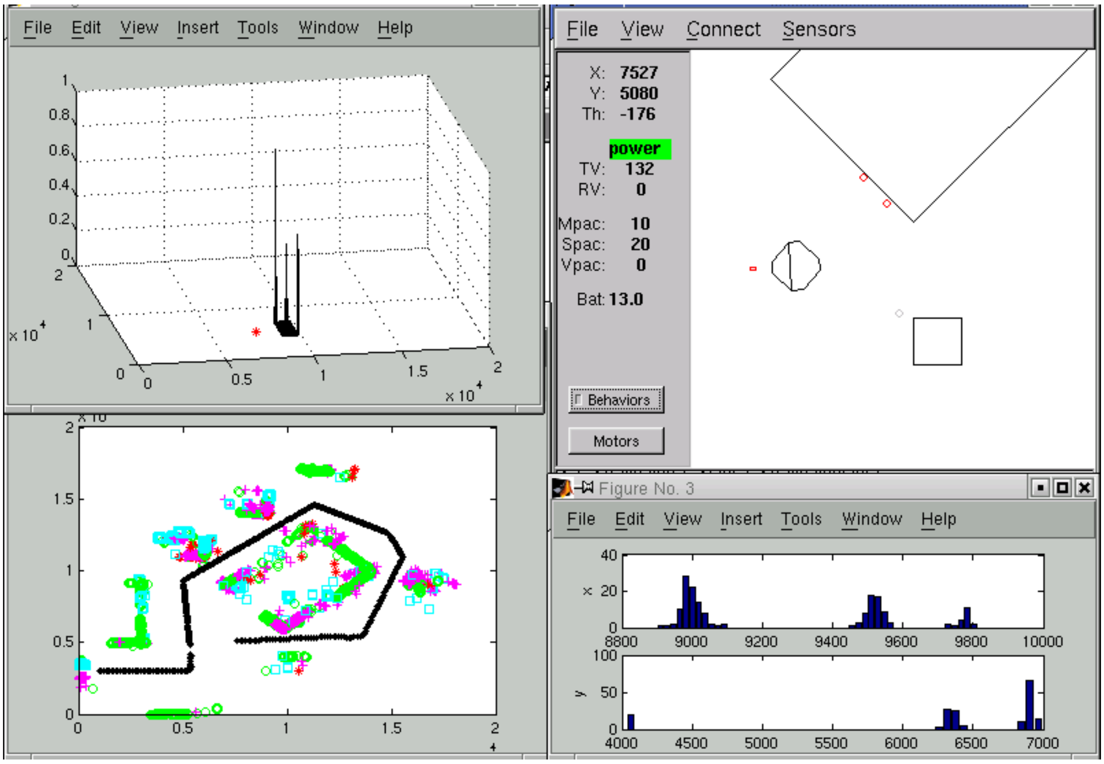


*Figure 4. Results from the probabilistic tracker developed.*

## 4. ACKNOWLEDGEMENTS

This work has been financed by the Spanish administration (Consejería de Educación de la Comunidad de Madrid) through Research Project “Enigma”, 07T/51/2003 and was developed during a researching period spent at the Centre of Autonomous Robots in KTH (Sweden) sponsored by the Spanish State Secretary of Education and University.

## 5. REFERENCES

[1] Isard M. & Blake A. "Condensation: conditional density propagation for visual tracking". *International Journal of Computer Vision*, 29 (1), pp. 5-28. 1998.

[2] Gordon N.J., Salmond D.J, & Smith A.F.M. “Novel Approach to Nonlinear/Non-gaussian Bayesian State Estimation”. *IEE Proceedings-F*, 140(2), pp:107-113. 1993.

[3] van der Merwe R., de Freitas N., Doucet A., Wan E. “The Unscented Particle Filter”. Advances in Neural Information Processing Systems. 13, November. 2001.

[4] Orton M. & Fitzgerald W. “A Bayesian Approach to Tracking Multiple Targets using Sensor Arrays and Particle Filters”. Technical Report, Cambridge University Engineering Dept. CUED/F-INFENG/TR.403.

[5] Schulz D., Burgard W., Fox D. & Cremers A.B. “Tracking Multiple Moving Objects with a Mobile Robot”. Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2001.

[6] Doucet A. de Freitas J.F.G. & Gordon N. *Sequential MonteCarlo methods in practice*. Springer-Verlag. 2000.

[7] Koller-Meier E.B. & Ade F. “Tracking Multiple Objects using a Condensation Algorithm”. *Journal of Robotics and Autonomous* Systems, pp. 93-105, 2001.

[8] Kanungo T., Netanyahu N.S. & Wu A.Y. “An Efficient k-Means Clustering Algorithm: Analysis and Implementation”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 24, nº7, July. 2002.