Parallel Particle Filters for Multiple Target Tracking

Sebbagh Abdennour and Tebbikh Hicham
Automatic and Computing Laboratory of Guelma (LAIG), 8 Mai 1945 Guelma University, Algeria

Abstract: *The Multiple Targets Tracking (MTT) problem is addressed in signal and image processing. When the state and measurement models are linear, we can find several algorithms that yield good performances in MTT problem, among them, the Multiple Hypotheses Tracker (MHT) and the Joint Probabilistic Data Association Filter (JPDAF). However, if the state and measurement models are nonlinear, these algorithms break down. In this paper we propose a method based on particle filters bank, where the objective is to make a contribution for estimating the trajectories of several targets using only bearings measurements. The main idea of this algorithm is to combine the Multiple Model approach (MM) with Sequential Monte Carlo methods (SMC). The result from this combination is a Nonlinear Multiple Model Particle Filters algorithm (NMMPF) able to estimate the trajectories of multiple targets.*

Keywords: *MM approach, MTT, particle filtering.*

Received August 21, 2014; accepted December 21, 2014; Published online December 23, 2015

1. Introduction

Multiple Targets Tracking (MTT) problems have received a wide attention in literature [3, 5, 6]. It is a special kind of dynamic state estimation problem. To perform MTT the observer has at his disposal some measurements collected from different sensors. In fact, the estimation of the states in MTT problem requires the assignment of the measurements to multiple targets. Generally, MTT presents two basic problems: the estimation and data association.

Many problems in science require estimation of the state of a system that changes over time using a sequence of noisy measurements made on the system. The most widely used estimator for nonlinear systems is the Extended Kalman Filter (EKF). However, in the case of a highly nonlinearities, EKF suffers from some limitations that make it inapplicable

In order to, accurately estimate states or measurements models of nonlinear targets with non Gaussian noises, it is necessary to substitute the EKF by a nonlinear one. This nonlinear filter is called Sequential Monte Carlo (SMC) methods or particle filtering methods. The basic idea of this filter is to propagate a weighted set of particles to estimate the Probability Density Function (PDF) of the state conditioned on the observations, and updating particles weights using Bayes formula. More details are found in [19, 21]. SMC methods have been used in very different areas for Bayesian filtering and can be applied under very general hypotheses, under different names: the bootstrap filter for target tracking [12] and the condensation algorithm in computer vision [15] are two examples among others.

In the case of linear state and measurement models, we can find several algorithms that yield good performances in MTT problem, among them, the Multiple Hypotheses Tracker (MHT) [5] and the Joint Probabilistic Data Association Filter (JPDAF) [1, 6]. However, if the state and measurement models are nonlinear, these algorithms break down. In this paper, another algorithm based on particle filter bank for MTT is proposed and is called Nonlinear Multiple Model Particle Filters algorithm (NMMPF) [19]. The main idea of this algorithm is based on running several parallel filter bank at the same time and is obtained by the combination between Multiple Model (MM) approach and SMC methods. It results from this a bank of particle filters and each filter is designed to track one target.

This work is organized as follows. In section 2, we present the mathematic formulation of an aircraft motion models in horizontal plan. In section 3, we describe the particle filter bank: Starting by SMC methods with adaptive resampling and finishing by the combination with MM approach. It results from this: A NMMPF algorithm for tracking multiple targets. Finally, in section 4 we present the simulation results obtained by the application of NMMPF algorithm in MTT problem to track three targets.

2. Problem Formulation

In Air Traffic Control (ATC), civilian aircraft has two basic modes of flight: Uniform motion and maneuver. More details can be found in [2, 4, 19, 22].

The aircraft flight in (X, Y) plane can be modeled by:

1. A constant velocity model for the uniform motion In discrete time, the constant velocity model with noise is given by [17]:

$$X_{k+1} = \begin{pmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{pmatrix} X_k + \begin{pmatrix} 0.5T^2 & 0 \\ T & 0 \\ 0 & 0.5T^2 \\ 0 & T \end{pmatrix} V_k \quad (1)$$

Where T: Is the sampling time, X_k: Is the state vector including the position and velocity of the aircraft, defined as:

$$X = [¶ \quad \dot{¶} \quad ¦ \quad \dot{¦}]' \quad (2)$$

With ¶ and ¦ denoting the orthogonal coordinates of the horizontal plane, and V_k is a zero-mean Gaussian white noise representing the accelerations, with an appropriate covariance $Q = \begin{pmatrix} ±_X^2 & 0 \\ 0 & ±_Y^2 \end{pmatrix}$.

2. A nearly constant-speed turn model for a maneuver the turn of a civilian aircraft is characterized by constant turn rate and constant speed and known as Coordinated Turn (CT) [2, 17]. The CT model is a nonlinear one if the turn rate is not a known constant. By adding the turn rate Ω to the state vector $X = [¶ \quad \dot{¶} \quad ¦ \quad \dot{¦}]'$, results from this a new state vector defined as follows:

$$X = [¶ \quad \dot{¶} \quad ¦ \quad \dot{¦} \quad \Omega]' \quad (3)$$

The nearly constant-speed turn is defined as follows [2, 4, 17]:

$$X_{k+1} = \begin{pmatrix} 1 & \frac{sin\, t_k T}{t_k} & 0 & -\frac{1-cos\, t_k T}{t_k} & 0 \\ 0 & cos\, t_k T & 0 & -sin\, t_k T & 0 \\ 0 & \frac{1-cos\, t_k T}{t_k} & 1 & \frac{sin\, t_k T}{t_k} & 0 \\ 0 & sin\, t_k T & 0 & cos\, t_k T & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} X_k + \begin{pmatrix} 0.5T^2 & 0 & 0 \\ T & 0 & 0 \\ 0 & 0.5T^2 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{pmatrix} V_k \quad (4)$$

Note that, V_k in Equation 1 and V_k in Equation 4 have not the same dimension. The equation of observations is given according to:

$$Y_k = arctan\left(\frac{¦ = r}{¶ = `}\right) + W_k \quad (5)$$

Where (X, Y): Is the position of the target and W_k is gaussian noise with covariance $±_w^2$.

3. Particle Filters Bank

3.1. General Framework

We suppose that r is the number of targets. The state vector that we have to estimate is made by concatenating the state vector of each target model at time k, $X_k = \left(X_k^1, ..., X_k^r\right)$ follows the state Equation 8 decomposed in r partial equations [10, 13, 14]:

$$X_k^i = F_k^i\left(X_{k-1}^i, V_k^i\right) \quad i = 1, ..., r \quad (6)$$

(V_k^i) and $(V_k^{i'})$ are supposed white noises, independent for $i \neq i'$. The measurement vector at time k is denoted by $y_k = (y_k^1, ..., y_k^{m_k})$ and given by:

$$Y_k^j = H_k^i(X_k^i, W_k^j) \quad (7)$$

Where m_k is the available measurements at time k, with m_k can be different to r. Again, the noises (W_k^j) and $(W_k^{j'})$ are only supposed to be white noises, independent for $j \neq j'$.

3.2. SMC Methods

The dynamic system is represented by the stochastic process $(X_k) \in R^{n_x}$ whose evolution is given by the state equation [10, 13, 20, 21]:

$$X_k = F_k(X_{k-1}, V_k) \quad (8)$$

The objective is to estimate the state vector (X_k) at discrete times via the measurement equation:

$$Y_k = H_k(X_k, W_k) \quad (9)$$

Where the functions F_k and H_k are not assumed linear and the processes $(V_k) \in R^{n_v}$ and $(W_k) \in R^{n_w}$ are assumed independent white noises.

The original particle filter, which is called the bootstrap filter [8, 14, 20], estimates the densities $(L_k)_k$ by a weighted sum of N particles.

In the prediction step the particles of the set S_{k-1} are propagating according to Equation 8. During the correction step the weights of particles are updated using Equation 10:

$$L_k(x_k) = \frac{p(Y_k = y_k | X_k = x_k) p(X_k = x_k | Y_{0:k-1} = y_{0:k-1})}{\int_{R^{n_x}} p(Y_k = y_k | X_k = x_k) p(X_k = x_k | Y_{0:k-1} = y_{0:k-1}) dx} \quad (10)$$

In the most general setting of SMC methods [9, 11, 23] for more details, the displacement of particles is obtained by sampling from an appropriate density f. These two steps called Sequential Importance Sampling algorithm (SIS) which is given in Algorithm 1. The degeneracy is the principal drawback of the SIS algorithm; most of particles become dispose of very small weights and the others keep high weights. To remedy this drawback, we need another step to eliminate the particles of smaller weights; this latter is called the resampling step.

Algorithm 1: SIS algorithm.

Initialisation: $\begin{cases} s_0^n \sim p(X_0) \\ μ_0^n = 1/N \end{cases} \quad n = 1, ..., N.$

For $k = 1, ..., T_{end}$:

Proposal: *Sample* s_k^n *from*

$f(X_k | X_{k-1} = s_{k-1}^n, Y_k = y_k)$ *for* $n = 1, ..., N$.

$$Weighting \begin{cases} compute\ un-normalized\ weight: \\ \tilde{μ}_k^n = μ_{k-1}^n \dfrac{p(s_k^n | s_{k-1}^n) l_k(y_k; s_k^n)}{f(s_k^n | s_{k-1}^n, y_k)} \quad for\ n = 1, ..., N \\ normalize\ weights:\ μ_k^n = \dfrac{\tilde{μ}_k^n}{\sum_{n=1}^N \tilde{μ}_k^n} \quad for\ n = 1, ..., N \end{cases}$$

Return $\hat{E}g(X_k)=\sum_{n=1}^{N}\mu_k^n g(s_k^n)$.

After the resampling step the new particles will have new weights equal to $1/N$ [12].

But before this step, it will be necessary to calculate the effective sample size $\hat{N}_{eff}=1/\sum_{n=1}^{N}(\mu_k^n)^2$ to measure the degeneracy of the algorithm [16, 18].

As advocated in [9], the resampling step can be done only if $\hat{N}_{eff}<N_{threshold}$. This enables the particle set to better learn the process and to keep its memory during the interval where no resampling occurs. As shown in Algorithm 2 [9, 14, 16, 18, 24].

Algorithm 2: Sequential Importance with Adaptive Resampling (SIAR).

Initialisation: $\begin{cases} s_0^n \sim p(X_0) \\ \mu_0^n = 1/N \end{cases} \quad n=1,\ldots,N.$

For k= 1, …, T_{end}:

Proposal: *Sample* $\tilde{s}_k^n$ *from*

$f(X_k \mid X_{k-1}=s_{k-1}^n, Y_k=y_k) \quad for \quad n=1,\ldots,N.$

$$\textit{Weightin} \quad \begin{cases} \textit{compute un-normalized weight}: \\ \tilde{\mu}_k^n=\mu_{k-1}^n \dfrac{p(\tilde{s}_k^n \mid s_{k-1}^n) l_k(y_k;\tilde{s}_k^n)}{f(\tilde{s}_k^n \mid s_{k-1}^n, y_k)} \quad for\ n=1,\ldots,N. \\ \textit{normalize weights}: \mu_k^n=\dfrac{\tilde{\mu}_k^n}{\sum_{n=1}^{N}\tilde{\mu}_k^n} \quad for\ n=1,\ldots,N. \end{cases}$$

Return $\hat{E}g(X_k)=\sum_{n=1}^{N}\mu_k^n g(\tilde{s}_k^n)$.

Calculate $\hat{N}_{eff}=\dfrac{1}{\sum_{n=1}^{N}(\mu_k^n)^2}$.

Resampling: *if* $N_{eff}<N_{threshold}$ *then*: $\begin{cases} s_k^n \sim \sum_{d=1}^{N}\mu_k^d \delta_{\tilde{s}_k^d}. \\ \mu_k^n = 1/N. \end{cases}$

For n= 1, …, *N*, else $s_k^n=\tilde{s}_k^n$.

3.3. MM Approach

In this approach the system is one of r models [2, 7, 19], it starts with the prior probabilities of each model to obtain the corresponding posterior probabilities.

The prior probability that M_i is in mode i is:

$$P(M_i|Y_0)=\mu_i(0) \quad i=1,\ldots,r \tag{11}$$

Where Y_0: Is the prior information and $\sum_{i=1}^{r}\mu_i(0)=1$.

3.3.1. Model Probabilities

The posterior probability that the model i is being correct can be calculated recursively as [2] for more details:

$$\mu_i(k) \triangleq P(M_i \mid Y_k) \tag{12}$$

$$\mu_i(k)=P(M_i \mid Y_{k-1}, y(k))=\frac{p(y(k) \mid M_i, Y_{k-1})P(M_i \mid Y_{k-1})}{p(y(k) \mid Y_{k-1})} \tag{13}$$

Or

$$\mu_i(k)=\frac{p(y(k) \mid M_i, Y_{k-1})\mu_i(k-1)}{\sum_{i=1}^{r} p(y(k) \mid M_i, Y_{k-1})\mu_i(k-1)} \quad i=1,\ldots,r \tag{14}$$

Which $p(y(k)| M_i, Y_{k-1})$ is the likelihood function, this latter is obtained under Gaussian assumptions by:

$$\Lambda_i(k) \triangleq p[y(k) \mid Y_{k-1}, M_i] = N[\gamma_i(k); 0; S_i(k)] \tag{15}$$

Where γ_i and S_i are the innovation and its covariance. More details in [2, 5, 18].

3.3.2. Combined Estimate

We find in the output of each filter the state estimate $\hat{X}^i$, the covariance P^i and the likelihood function Λ_i. After the initialization, the filters run recursively in the same time [2, 25].

The models probabilities $\mu_i(k)$ are updated using their likelihood functions Λ_i.

These latters are used to obtain the mean estimate from the elemental estimates:

$$\hat{X}(k|k)=\sum_{i=1}^{r}\mu_i(k)\hat{X}^i(k|k) \tag{16}$$

The associated covariance is also obtained by using a weighted average:

$$P(k|k)=\sum_{i=1}^{r}\mu_i(k)\left(P^i(k|k)+\left[\hat{X}^i(k|k)-\hat{X}(k|k)\right]\cdot\left[\hat{X}^i(k|k)-\hat{X}(k|k)\right]^T\right) \tag{17}$$

MM approach with r filters bank is given in Figure 1.

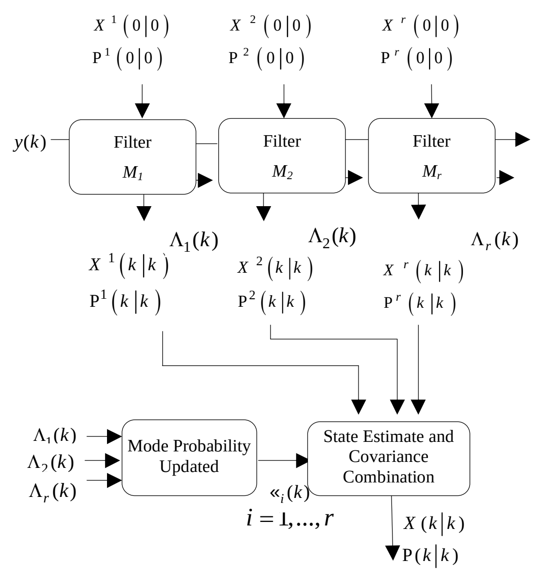


Figure 1. MM approach with r filters bank.

3.4. NMMPF Algorithm

To study accurately the MTT problem, we present the proposed algorithm called NMMPF algorithm [19].

The combination between MM approach and SMC methods is its basic idea. It consists of running at the same time r particle filters based on different motion models. For each measurement, we combine the states of the r particle filters weighted by a probability factor of each filter to obtain an aggregate state estimate for one target. Algorithm 3 presents a general description of NMMPF algorithm for multiple targets.

Algorithm 3: NMMPF algorithm for MTT.

State and covariance initialization filters: $\begin{cases} X_0^i \\ P_0^i \end{cases}$ *for i*=1, ..., *r*

Initialization of models probabilities:

For j=1, …, m_k *then* $\pi_i^j(0)=1/r$ *i*=1, ..., *r*

For k=1, …, *end*

Model-matched filtering (using a bank of particle filtering)

For i=1, …, *r*

Initialization: $\begin{cases} s_0^{n,i} = p(X_0^i) \\ \mu_0^n = 1/N \end{cases}$ *for* $n=1,...,N$

For j=1, …, m_k

Proposal: *sample* ($s_k^{n,i}$) *from*

$$f\left(X_k^i \middle| X_{k-1}^i = s_{k-1}^{n,i}, Y_k^j = y_k^j\right) \text{ for } n=1,...,N$$

Weighting:

$$\begin{cases} \text{compute un-normalized weights:} \\ \tilde{\mu}_k^{n,j} = \mu_{k-1}^{n,j} \dfrac{p(s_k^{n,i} \mid s_{k-1}^{n,i}) l_k(y_k^j; s_k^{n,i})}{f(s_k^{n,i} \mid s_{k-1}^{n,i}, y_k^j)} \quad \text{for } n=1,...,N \\ \text{normalize weights:} \\ \mu_k^{n,j} = \dfrac{\tilde{\mu}_k^{n,j}}{\sum_{n=1}^{N} \tilde{\mu}_k^{n,j}} \quad \text{for } n=1,...,N. \end{cases}$$

Return $E(g(X_k^i)_j) = \sum_{n=1}^{N} \mu_k^{n,j} g(X_k^i)_j$

Calculate $N_{eff}^j = 1/\sum_{n=1}^{N} (\mu_k^{n,j})^2$

Resampling if $N_{eff}^j \langle N_{threshold}$

Model probability update

For j=1, …, m_k

$$\Lambda_i^j(k) \overset{a}{=} p\left(y^j(k) \middle| Y_{k-1}^j, M_i\right) = p\left(\nu_i^j(k)\right) = N\left(\nu_i^j(k); 0; S_i^j(k)\right)$$

$$\pi_i^j(k) = \frac{\Lambda_i^j(k)\pi_i^j(k-1)}{\sum_{i=1}^{r} \Lambda_i^j(k)\pi_i^j(k-1)} \quad i=1,...,r$$

State estimate and covariance combination

For j=1, …, m_k

$$\hat{X}^j(k|k) = \sum_{i=1}^{r} \pi_i^j(k)\hat{X}_j^i(k|k)$$

$$P_j(k|k) = \sum_{i=1}^{r} \pi_i^j(k)\left(P_j^i(k|k) + \left[\hat{X}_j^i(k|k) - \hat{X}^j(k|k)\right]\cdot\left[\hat{X}_j^i(k|k) - \hat{X}^j(k|k)\right]^T\right)$$

The combination between MM approach and particle filtering methods requires the calculation of additional conditional probabilities $P(M(k))=M_i|Y_{k-1}$ which allow an adequate mixing of the modal estimates to produce an aggregate state and covariance estimates.

4. Simulation Results

In order to, evaluate the effectiveness of NMMPF algorithm in MTT, a simulated example is presented in this section using the Matlab simulation. We consider the following scenario where three targets follow:

- *Target* 1: A constant velocity model without acceleration noise.
- *Target* 2: A constant velocity model with acceleration noise.
- *Target* 3: A nearly constant-speed turn model or (CT model).

Initial states vectors that contain positions and velocities of the three targets are:

$$(X_0^1) = \begin{pmatrix} 3000 \\ 10 \\ 500 \\ 5 \end{pmatrix}; \quad (X_0^2) = \begin{pmatrix} 500 \\ 10 \\ 3000 \\ 5 \end{pmatrix}; \quad (X_0^3) = \begin{pmatrix} 2000 \\ 10 \\ 1500 \\ 5 \\ 0.00223 \end{pmatrix}$$

With $\sigma_X = \sigma_Y = 0.05 M/s^2$.

We suppose that each measurement is related to one target according to Equation 5 at each time T=4s with σ_w=0.02rad (about 1.5degree). The Targets trajectories and their bearings are plotted in Figures 2 and 3.

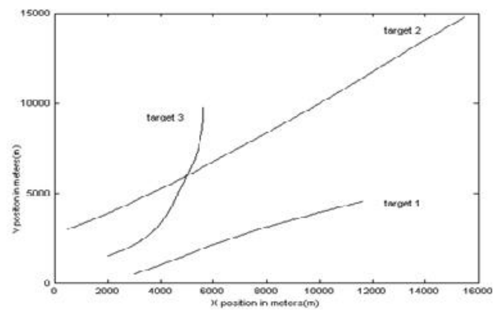


Figure 2. Real targets trajectories.

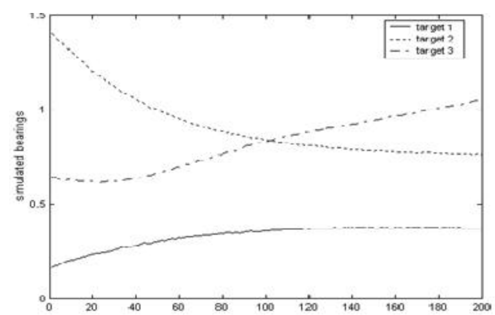


Figure 3. Simulated bearings of three targets in radians.

In order to, track these targets, it has been necessary to put in place a bank of filters composed from three particle filters, their mean vectors and covariance matrixes initialization according to a Gaussian law are:

$$(X_0^1)_{mean} = \begin{pmatrix} 2800 \\ 11 \\ 600 \\ 4 \end{pmatrix}; \quad (X_0^2)_{mean} = \begin{pmatrix} 450 \\ 9 \\ 3200 \\ 4.5 \end{pmatrix}; \quad (X_0^3)_{mean} = \begin{pmatrix} 2100 \\ 11 \\ 1400 \\ 4 \\ 0.00323 \end{pmatrix}$$

$$P_0^1 = P_0^2 = \begin{pmatrix} 100 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0.01 \end{pmatrix}$$

$$P_0^3 = \begin{pmatrix} 100 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0 & 0.0001 \end{pmatrix}$$

For each measurement, the initial probabilities of the three particle filters are:

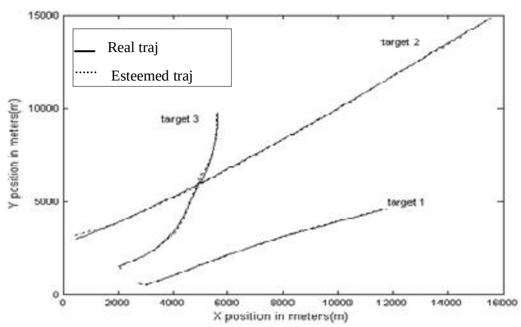
$$\alpha_i^1 = 1/3; \ \alpha_i^2 = 1/3; \ \alpha_i^3 = 1/3; \ i = 1, 2, 3$$

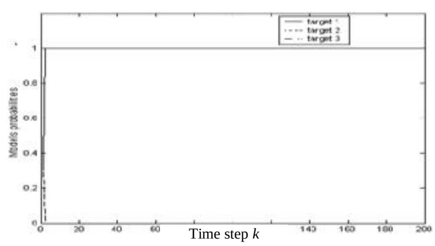
That is to say: The three particle filters models have the same chance to be chosen in the beginning.

The application of the NMMPF algorithm in this example has been done with:

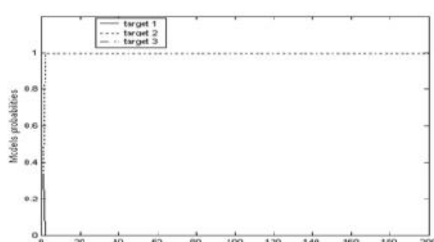
- 100 Monte Carlo runs.
- Particles number N=500.
- Resampling threshold $N_{threshold}$=0.8N.

In order to measure the accuracy of NMMPF algorithm in tracking, we have chosen the Root Mean Square Error (RMSE) as the performance measure of this algorithm.



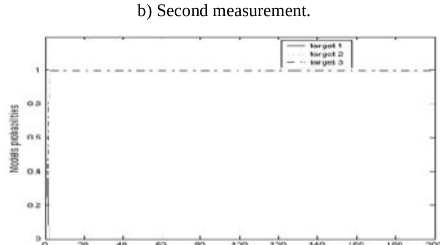


a) First measurement.



Time step k

b) Second measurement.



Time step k

c) Third measurement.

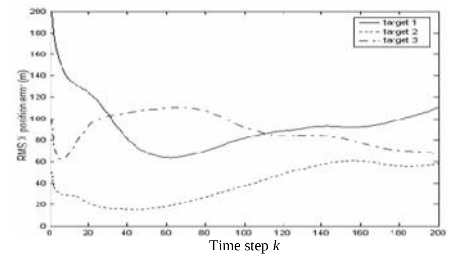
Figure 5. Models probabilities.

Figure 4. Targets trajectories (real and esteemed).

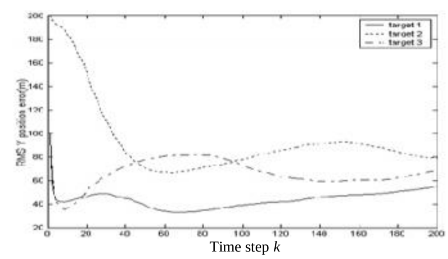
From Figure 4 it appears that the esteemed trajectories are indistinguishable from the real ones.

The power of NMMPF algorithm appears in the assignment of the measurements to their right targets, it needs to one or two samples to assign each measurement to the right target and these results are plotted in Figure 5-a, b and c.

(X, Y) position RMSE, (X, Y) velocity RMSE and angular velocity RMSE are plotted in Figure 6-a and b, Figures 7-a, b and 8. These smallest RMSE demonstrate and confirm that the NMMPF algorithm is a good estimator for MTT problem and prove the results plotted in Figure 4.

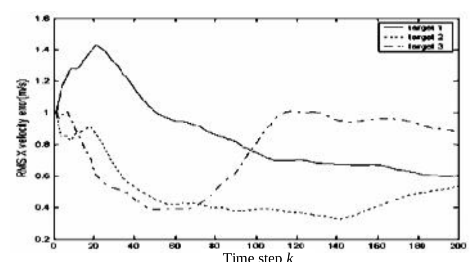


a) X Position error.

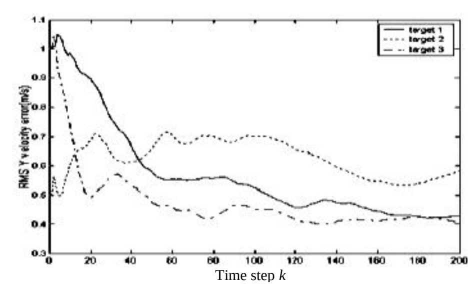


b) Y Position error.

Figure 6. RMS positions errors.



a) X Velocity error.



b) Y Velocity error.

Figure 7. RMS velocities errors.

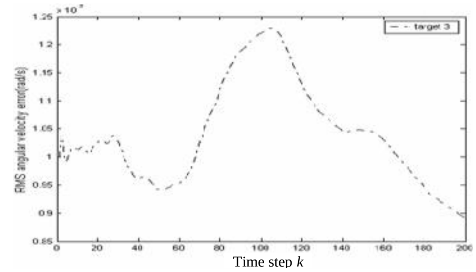


Figure 8. RMS angular velocity error (rad/s).

Figures 9 and 10 present the real and the esteemed trajectories of the three targets for N=1000 particles and N=2000 particles respectively.

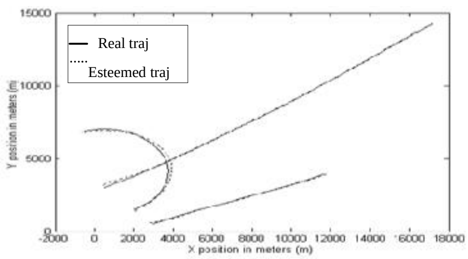


Figure 9. Targets trajectories (real and esteemed) for N=1000 particles.

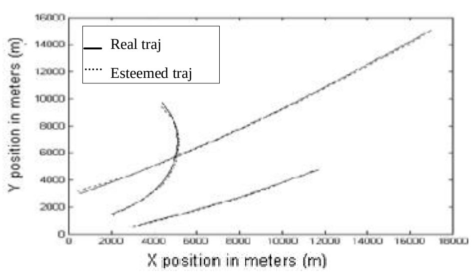


Figure 10. Targets trajectories (real and esteemed) for N=2000 particles.

Generally, when we increase the particles number N, the estimation error decreases, but the time calculation for each iteration or time step k increases. Table 1 presents the necessary time for one iteration when we execute the NMMPF algorithm by PC Pentium IV, 3.40GHz with N=500, N=1000 and N=2000.

Table 1. Time calculation for one iteration of NMMPF Algorithm.

Particles Number *N*	**500**	**1000**	**2000**
Time of One Iteration of NMMPF Algorithm in Milliseconds (ms)	420 ms	945 ms	2300ms

A biased initialisation by the mean vectors:

$$(X_0^1)_{mean}=\begin{pmatrix}3500\\11\\1000\\4\end{pmatrix};\ (X_0^2)_{mean}=\begin{pmatrix}1000\\9\\2500\\4.5\end{pmatrix};\ (X_0^3)_{mean}=\begin{pmatrix}2500\\11\\1000\\4\\0.00323\end{pmatrix}$$

Gives us the results plotted in Figure 11.

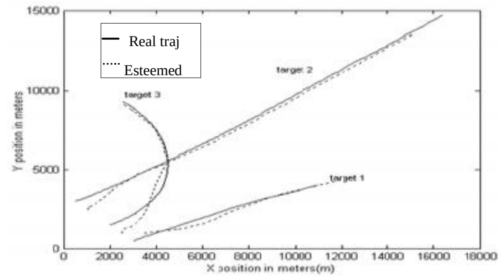


Figure 11. Targets trajectories (real and esteemed) with biased initialisation, for N=500 particles.

We can say that the initialisation is still a main problem for particle filters and all suboptimal filters.

From these results we can say that while maintaining good tracking performance, the NMMPF algorithm is a pertinent solution to nonlinear MTT problem. The RMSE formula is given by Equation 18.

$$RMSE(X_j)=\sqrt{\frac{\sum_{k=1}^{j}\left(X_{real}(k)-X_{esteemed}(k)\right)^2}{j}} \qquad (18)$$

$$for\ j=1,\ldots,200$$

5. Conclusions

Under the combination technique between MM approach and SMC methods, we have presented in this work a NMMPF algorithm for an efficient tracking of three targets with nonlinear state and/or measurement models and non Gaussian noises. The SMC methods estimate the targets states vectors, where the MM approach plays the assignment task of measurements to their targets models.

In the case of linear MTT, the JPDAF and the PMHT estimators are the leaders because of the Kalman filter optimality. However, if the process and/or measurement models are nonlinear, the EKF is no longer optimal and presents several drawbacks. We substitute the EKF by the SMC methods to overcome these drawbacks and to combine them with MM approach. The results simulations of the used example demonstrate that the NMMPF algorithm tracks accurately the targets trajectories and also assigns each measurement to the right target model.

References

[1] Bar-Shalom Y. and Tse E., “Tracking in a Cluttered Environment with Probabilistic Data Association,” *Automatica*, vol. 11, no. 5, pp. 451-460, 1975.

[2] Bar Shalom Y., Li X., and Thiagalingam K., *Estimation with Applications to Tracking and Navigation*, USA, 2001.

[3] Blom H. and Bloem E., “Tracking Multiple Targets by Joint Combinations of IMM and PDA,” *in Proceeding of the 42nd IEEE*

Conference on Decision and Control, pp. 2965-2970, 2003.

[4] Boers Y. and Driessen H., "A Multiple Model Multiple Hypothesis Filter for Markovian Switching Systems," *Automatica*, vol. 41, no. 4, pp. 709-716, 2005.

[5] Chang K. and Bar-Shalom Y., "Joint Probabilistic Data Association for Multi target Tracking with Possibly Unresolved Measurements and Maneuvers," *IEEE Transaction of Automation and Control*, vol. 29, no. 7, pp. 585-594, 1984.

[6] Chen B. and Tugnait J., "Tracking of Multiple Maneuvering Targets in Clutter using IMM/JPDA Filtering and Fixed-lag Smoothing," *Automatica*, vol. 37, no. 2, pp. 239-249, 2001.

[7] Djouadi M., Sebbagh A., and Berkani D., "A Nonlinear Algorithm for Manoeuvrings Target Visual-based Tracking," *IEEE Proceedings of the 2nd International Conference on intell Sens and Infor Proc, ICISIP*, Chennai, India, pp. 61-66, 2005.

[8] Doucet A., "Convergence of sequential Monte Carlo Methods," *Technical Report*, Signal Processing Group, Department of Engineering, University of Cambridge, 2000.

[9] Doucet A., "On Sequential Simulation-based Methods for Bayesian Filtering," *Technical Report*, Department of Engineering. University of Cambridge, 1998.

[10] Doucet A., De Freitas N., and Gordon N., *Sequential Monte Carlo Methods in Practice*, New York: Springer, pp. 525- 532, 2001.

[11] Gordon N., "A Hybrid Bootstrap Filter for Target Tracking in Clutter," *IEEE Transaction on Aerospace and Electronic Systems*, vol. 33, no. 1, pp. 353-358, 1997.

[12] Gordon N., Salmond D., and Smith A., "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *in Proceedings of F-Radar and Signal Processing*, pp. 107-113, 1993.

[13] Hue C., Le Cadre J., and Pérez P., "Sequential Monte Carlo Methods for Multiple Target Tracking and Data Fusion," *IEEE Transaction on Signal Processing*, vol. 50 no. 2, pp. 309-325, 2002.

[14] Hue C., Le Cadre J., and Pérez P., "Tracking Multiple Objects with Particle Filtering," available at: http://www.irisa.fr/vista/Papers/2002_aes_hue.pdf, last visited 2002.

[15] Isard M. and Blake A., "Condensation–Conditional Density Propagation for Visual Tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5-28, 1998.

[16] Kong A., Liu J., and Wong W., "Sequential Imputation Method and Bayesian Missing Data Problems," available at: http://stat.rutgers.edu/home/rongchen/papers/KongLiuWong.pdf, last visited1994

[17] Li X. and Bar-Shalom Y., "Design of an Interacting Multiple Model Algorithm for Air Traffic Control Tracking," *IEEE Transactions on Control Systems Technology*, vol. 1, no. 3, pp. 186-194, 1993.

[18] Liu J., "Metropolized Independent Sampling with Comparison to Rejection Sampling and Importance Sampling," *Statistics and Computing*, vol. 6, no. 2, pp. 113-119, 1996.

[19] Sebbagh A. and Tebbikh H., "Nonlinear Multiple Model Particle Filters Algorithm for Tracking Multiple Targets," *Archives of Control Sciences*, vol. 21, no. 1, pp. 37-60, 2011.

[20] Van Der Merwe R., Doucet A., De Freitas N., and Wan E., "The Unscented Particle Filte," *Technical Report*, Cambridge University Engineering Department, 2000.

[21] Vermaak J., Godsill S., and Perez P., "Monte Carlo Filtering for Multi-Target Tracking and Data Assoication, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 309-332, 2005.

[22] Wang H., Kirubarajan T., and Bar Shalom Y., "Precision Large Scale Air Traffic Surveillance Using an IMM Estimator with Assignment," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 1, pp. 255-266, 1999.

[23] Yang N., Tian W., Jin Z., and Zhang C., "Particle Filter for Sensor Fusion in a Land Vehicle Navigation System," *Measurement Science and Technology*, vol. 16, no. 3, pp. 677-681, 2005.

[24] Zahidul I., Chi-Min O., and Chil-Woo L., "Effect of Resampling Steepness on Particle Filtering Performance in Visual Tracking," *The International Arab Journal of Information Technology*, vol. 10, no. 1, pp. 102-109, 2013.

[25] Zheng T., Chao S., and Zongwei L., "The Tracking Algorithm for Maneuvering Target Based on Adaptive Kalman Filter," *The International Arab Journal of Information Technology*, vol. 10, no. 5, pp. 453-459, 2013.

Sebbagh Abdennour is a Doctoral candidate in Electritechnic and Automatic Department of 08 May 1945 Guelma University, Algeria. He received his automatic engineer degree from the University of Annaba in 2000 and his Magister degree from the Polytechnic military school (EMP) of Algiers, Algeria in 2004. His research interests include automatic control and tracking targets.

Tebbikh Hicham is a professor in automatic and signal processing of 08 may 1945 Guelma University, Algeria. He received his electronic engineer degree from the University of Annaba in 1984 and his doctorate degree from the I.N.P of Grenoble, France in 1988. His research interests include modeling and fractional control systems, modeling and control of hybrid dynamic systems and automatic pattern recognition. He was in charge of several scientific and administrative responsibilities including Chair of the Scientific Council of the Faculty of Science and Technolog. He also participated, as a member on several national science and technology committees including the Sector Standing Committee (CSP) of the MESRS from 2009 to 2012, the Central Committee of the new technologies of information and communication of the MESRS from 2002. Currently he is a director of automatic and computing laboratory of guelma (LAIG).