# A Fast JPDA-IMM-PF based DFS Algorithm for Tracking Highly Maneuvering Targets

Mohand Saïd DJOUADI

*Laboratoire robotique & productique, Ecole militaire polytechnique, Bp :17 Bordj El Bahri 16111 Alger, Algérie.*
Email: *msdjouadi@yahoo.fr*

Yacine Morsly

*Laboratoire robotique & productique, Ecole militaire polytechnique, Bp :17 Bordj El Bahri 16111 Alger, Algérie.*
Email: *ymorsly@yahoo.fr*

Daoud BERKANI

*Electrical* & Computer Engineering, *Ecole Nationale Polytechnique, 10, Avenue Hassen Badi, BP.182, 16200 El Harrach Alger, Algérie.*
Email: *dberkani@hotmail.com*

Abstract— **In this paper, we present an interesting filtering algorithm to perform accurate estimation in jump Markov nonlinear systems, in case of multi-target tracking. With this paper, we aim to contribute in solving the problem of model-based body motion estimation by using data coming from visual sensors. The Interacting Multiple Model (IMM) algorithm is specially designed to track accurately targets whose state and/or measurement (assumed to be linear) models changes during motion transition. However, when these models are nonlinear, the IMM algorithm must be modified in order to guarantee an accurate track. In order to deal with this problem, the IMM algorithm was combined with the Unscented Kalman Filter (UKF) [6]. Even if the later algorithm proved its efficacy in nonlinear model case; it presents a serious drawback in case of non Gaussian noise. To deal with this problem we propose to substitute the UKF with the Particle Filter (PF). To overcome the problem of data association, we propose the use of an accelerated JPDA approach based on the depth first search (DFS) technique [12]. The derived algorithm from the combination of the IMM-PF algorithm and the DFS-JPDA approach is noted DFS-JPDA-IMM-PF.**

*Index Terms—* **Estimation, Kalman filtering, Particle filtering JPDA, Multi-Target Tracking, Visual servoing, data association.**

## I. INTRODUCTION

This paper hope to be a contribution within the field of visual-based control of robots, especially in visual-based tracking [3]; tracking maneuvring targets, which may themselves be robots, is a complex problem, to ensure a good track when the target switches abruptly from a motion model to another is not evident. Because of the complexity and difficulty of the problem, a simple case is considered. The study is restricted to 2-D motions of a point, whose position is given at sampling instants in terms of its Cartesian coordinates. This point may be the center of gravity of the projection of an object into a camera plane, or the result of the localisation of a mobile robot moving on a planar ground.

Several of maneuvering target tracking algorithms are developed. Among them, the interacting multiple model (IMM) method based on the optimal Kalman filter, yields good performance with efficient computation especially when the measurement and state models are linear with Gaussian noise. However, if the later are nonlinear and/or non Gaussian noise, the standard Kalman filter should be substituted, in our study we choose the Particle Filter (PF). The algorithm derived from this combination is called IMM-PF. The other problem treated in this paper, is about the data association. Effectively, at each sample time, the sensor (camera) present, several measures and observations, coming from different targets; the problem is how to affect each measure to the correct target, to deal with this problem we choose an accelerated version of the JPDA algorithm based on the depth first search (DFS). The algorithm derived from the combination of DFS-JPDA and the non linear IMM algorithms is noted DFS- JPDA-IMM-PF.

The paper is organized as follows. In section II the mathematical formulation of 2-D motion is presented. In section III we describe the IMM algorithm PF based. In section IV we present the DFS algorithm and than in section V we present DFS-JPDA-IMM-PF algorithm. In section VI we present and discuss the results of simulations. Finally in section VII we draw the conclusion.

## II. MATHEMATICAL FORMULATION OF 2-D MOTION

The mathematical formulation of 2-D motion used is mainly inspired from Danes, Djouadi, and al in [4]. They make the hypothesis that the measurements are only the 2-D Cartesian coordinates of the moving point.

Let s(.) denote the curvilinear abscissa of M over time onto its trajectory, the origin of curvilinear abscissae is set arbitrarily. Functions x(.) and y(.), represent the Cartesian coordinates of M. The measurement equation may be written as:

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \underline{h}\left(\underline{s}(t), \underline{p}(t)\right) \qquad (1)$$

Where $\underline{p}(.)$ is a parameter vector function of minimal size.
We can see that equation (1) is independent of the type of the motion of M onto its trajectory.
The state equation could be written as:

$$\dot{\underline{X}}(t) = A\underline{X}(t) \qquad (2)$$

with $\underline{X}(t) = \begin{pmatrix} \underline{s}(t) \\ \underline{p}(t) \end{pmatrix}$

A equals $\begin{pmatrix} A_s & 0 \\ 0 & 0 \end{pmatrix}$, with $A_s$ the $n$ x $n$ zero matrix with ones added on its first upper diagonal, and 0 the matrices of convenient sizes. The continuous time state equation (2) is linear time invariant and independent of M's trajectory, except on the sizes of $\underline{s}(.)$ and $\underline{p}(.)$. Moreover, it may be shown that the fundamental matrix $\mathbf{F}$ involved its exact discretization at the period $T$ takes the form

$$\mathbf{F} \overset{\Delta}{=} \exp(AT) = \sum_{i=0}^{n-1} \frac{(AT)^i}{i!}.$$

The dynamic and measurement noises are supposed to be stationary, white and Gaussian, non inter-correlated with known covariances.

*A. Canonical motion equations*

The point M is supposed to move on straight or circular trajectories at constant or uniformly time-varying speed (constant speed or constant acceleration). Those motions belong to the set of the possible behaviours of a non-holonomic robot whose wheels are driven at constant velocities or accelerations.

*1) Output equations*: One minimal description of a straight line is defined by the vector $\underline{p} = (\alpha, d)^T$ shown in figure 1(a), which is related to Plucker coordinates. Concerning a circular trajectory one minimal description is defined by the vector $\underline{p} = (R0, x_0, y_0)^T$ shown in figure 1(b). The origin of curvilinear abscissa is uniquely defined from those parameterizations.



(a) Line
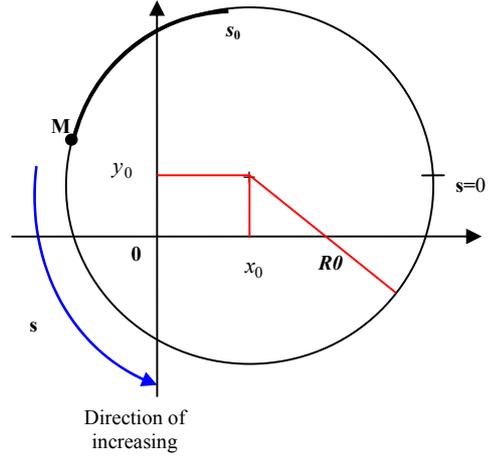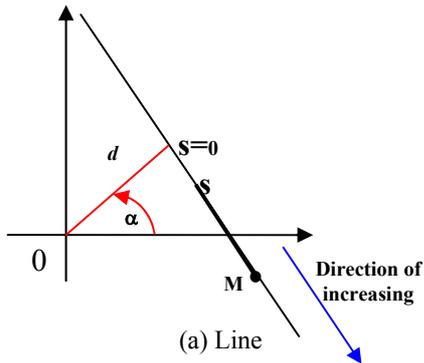


(b) Circle
Fig.1. Trajectory Parameterization

The output equations are as follows (trajectory parameter are considered time-invariant):

**Straight Line:** $z(k) \overset{\Delta}{=} \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} = \begin{pmatrix} d\cos\alpha + s(k)\sin\alpha \\ d\sin\alpha - s(k)\cos\alpha \end{pmatrix} + v(k)$ (3)

**Circle:** $z(k) \overset{\Delta}{=} \begin{pmatrix} x(k) \\ y(k) \end{pmatrix} = \begin{pmatrix} x_0 + R0\cos\dfrac{s(k)}{R0} \\ y_0 + R0\sin\dfrac{s(k)}{R0} \end{pmatrix} + v(k)$ (4)

with $s(k)$ distance covered by the target and $v(\cdot)$ measurement noise *with density $d_{v(k)}(v)$*.

*2) State Equations*

*Constant velocity* $\begin{cases} s(k+1) = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} s(k) + w_{\underline{s}}(k) \\ \underline{p}(k+1) = \quad\quad \underline{p}(k) + w_{\underline{p}}(k) \end{cases}$ (5)

*Constant acceleration*

$\begin{cases} \underline{s}(k+1) = \begin{pmatrix} 1 & T & \dfrac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{pmatrix} \underline{s}(k) + w_{\underline{s}}(k) \\ \underline{p}(k+1) = \quad\quad \underline{p}(k) + w_{\underline{p}}(k) \end{cases}$ (6)

where the random vectors
$x(0) = \left( \underline{s}(0)^T, \underline{p}(0)^T \right)^T$, $w(\cdot) = \left( w_{\underline{s}}(\cdot)^T, w_{\underline{p}}(\cdot)^T \right)^T$

and $\quad s(k) = \begin{bmatrix} s(k) \\ \dot{s}(k) \\ \ddot{s}(k) \end{bmatrix}$ vector of the point M dynamics

$\underline{p}(k)$ Trajectory parameters vector

## III. THE IMM-PARTICLE FILTER ALGORITHM

The basic idea is to combine the IMM approach [1], with a particle filter one. In the derivation of the standard IMM filter, a merging and filtering process are defined. We adopt a regularized particle filter for this filtering step, and perform the merging step on the probability densities, represented by a Gaussian mixture. One consequence of the discrete nature of the approximation of the a posteriori density is that it cannot directly be applied to an IMM framework as it is used in [1].

To obtain a good continuous approximation of the a posteriori density, we use a regularized version of the bootstrap filter as first reported in [8,9] for tracking targets in clutter. In this hybrid version of the bootstrap filter, the probability density function, that has been computed as a point mass probability density on a number of grid points in the state space, is fitted to a continuous probability density function that is a sum of a prefixed number of Gaussian density functions. Moreover, by using a hybrid type of sampling filter as an alternative for direct resampling, degeneracy in the effective number of particles is avoided [8,9]. The main advantages of the new method that we propose here are:

- ✓ the method is able to deal with nonlinearities and non-Gaussian noise in a mode;
- ✓ the method uses a fixed number of particles in each mode, independent of the mode probability.

### *Algorithm*

Let a system be described by the equations:

$$x(k) = f[x(k-1), k-1, M(k)] + w[k-1, M(k)]$$
$$z(k) = h[x(k), k, M(k)] + v[k, M(k)] \tag{7}$$

The process noise and the measurement noise are possibly mode-dependent. Their densities are denoted by: $d_{w[k,M(k)]}(w)$ and $d_{v[k,M(k)]}(v)$.

Where $M(k)$ denotes the model at time $k$. It's a finite state Markov process tacking values in $\{M_j\}_{j=1}^r$, according to a Markov transition probability matrix $p$ assumed to be known.

The probability density of the initial state is known, $x(0) \sim P_0(x)$. Define the information up to and including time step k as:

$$Z(k) = \{z(1), \dots, z(k)\}$$

The filtering problem that has to be solved is:
Given a realization of Z(k) associated with (7) compute p(x(k)|Z(k)); i.e. the conditional probability density of the state x(k) given the set of measurements Z(k).

A cycle of the IMM algorithm could be summarized in four steps:

### ✓ *Interaction stage*

Compute Mixing probabilities

$$\mu_{i/j}(k-1/k-1) = \frac{1}{c_j} p_{ij} \mu_i(k-1) \tag{8}$$

Compute Normalizing factors

$$c_j = \sum_{i \in M} p_{ij} \mu_i(k-1) \tag{9}$$

Compute A priori probability density in mode $j$

$$\hat{p}_0^j(x_{0j}(k-1)/Z(k-1)) = \sum_{i \in M} \hat{p}^i(x_i(k-1/Z(k-1))) * \mu_{i/j}(k-1/k-1) \tag{10}$$

### ✓ *Filtering stage*

$\forall j \in M$ draw N samples $\bar{x}_l^j(k-1)$ according to $\hat{p}_0^j(x_{0j}(k-1)/Z(k-1))$

The predicted samples are:

$$\hat{x}_j^l(k) = f(\bar{x}_j^l(k-1), k-1, j) + w^l(k-1, j) \tag{11}$$

Where $w^l(k-1, j)$ are samples obtained from $d_{w(k-1,j)}(w)$

The predicted output

$$z_j^l(k/k-1) = h(\hat{x}_j^l(k), k, j) \tag{12}$$

The probability weight

$$\bar{q}_j^l(k) = d_{v(k,j)}(z(k) - \hat{z}_j^l(k/k-1)) \tag{13}$$

Normalizing

$$\tilde{q}_j(k) = \sum_{l=1}^N \bar{q}_j^l(k) \tag{14}$$

Normalized probability masses

$$q_j^l = \frac{\bar{q}_j^l(k)}{\tilde{q}_j(k)} \tag{15}$$

Mean of the state over the sample set

$$\bar{x}_j(k) = \sum_{l=1}^N q_j^l \hat{x}_j^l(k) \tag{16}$$

Covariance of the state over the sample set

$$\hat{P}_j(k) = \sum_{l=1}^N q_j^l(\hat{x}_j^l(k) - \bar{x}_j(k))(\hat{x}_j^l(k) - \bar{x}_j(k))^T \tag{17}$$

From the conditional probability density function for the state in mode $j$ based on a mixture of N Gaussian densities

$$\hat{P}_N^j(x_j(k)/Z(k)) = \sum_{l=1}^N q_j^l N(\hat{x}_j^l(k), v_j \hat{P}_j(k)) \tag{18}$$

Where $\nu_j = 0.5 N^{-2/d_j}$, and $d_j$ is the dimension of the state space.

We obtain the probability density function for the state in mode $j$ after mixture reduction, i.e. based on a mixture of $N_r \leq N$ Gaussian densities.

$$\hat{p}^j\left(x_j(k)/Z(k)\right) = \sum_{l=1}^{N_r} q_j^{r,l} N\left(\hat{x}_j^{r,l}(k), \nu_j \hat{P}_j^r(k)\right) \qquad (19)$$

The mean of predicted output over the sample set

$$\overline{h}_j(k) = \sum_{l=1}^{N} h\left(\hat{x}_j^l(k), k, j\right) \qquad (20)$$

Residual covariance over the sample set

$$\hat{S}_j(k) = \sum_{l=1}^{N} q\left(h(\hat{x}_j^l(k), k, j) - \overline{h}_j(k)\right)\left(h(\hat{x}_j^l(k), k, j) - \overline{h}_j(k)\right)^T$$

$$\qquad (21)$$

Innovations

$$\gamma_j^l(k) = z(k) - h\left(\hat{x}_j^l(k), k, j\right) \qquad (22)$$

Probability density function for the innovations

$$\hat{p}^j\left(\gamma_j(k)/Z(k)\right) = \sum_{l=1}^{N} q_j^l N\left(0, \hat{S}_j(k)\right) = N\left(0, \hat{S}_j(k)\right)$$

$$\qquad (23)$$

Likelihoods

$$L_j^l(k) = N\left(\gamma_j^l(k); 0, \hat{S}_j(k)\right) \qquad (24)$$

Mode probabilities

$$\mu_j(k) = \frac{1}{c} L_j(k) c_j \qquad (25)$$

Where

$$c = \sum_{j \in M} L_j(k) c_j \qquad (26)$$

✓ **Combination stage**

The a posteriori conditional probability density function for the stae

$$\hat{p}(x(k)/Z(k)) = \sum_{j \in M} \hat{p}^j\left(x_j(k)/Z(k)\right) \mu_j(k) \qquad (27)$$

## IV. DFS ALGORITHM

Let $m$ and $n$ be the numbers of measurements and targets, respectively, in a particular cluster. The computational cost of data association increases exponentially with $m$ and $n$. The efficiency of the algorithm used in the generation of the data association hypothesis is particularly important when $m$ and $n$ are large. In order to develop an efficient algorithm to generate all data association hypotheses, a mathematical model is developed for data association. One of the most used models for a combinational problem is called *exhaustive search with constraints* [13].

In the context of tracking multitarget, data association can be modeled as an exhaustive search with a set of proper notations. Let $X_j \left(j = 1,2,...,m\right)$ denote measurement $j$. The value of $X_j$ belongs to a set $Z_j$. The value of $Xj$ identifies the target which is hypothesized to be associated with measurement $j$. For example, $X_j = 2$ implies that measurement $j$ is hypothesized to be associated with target 2. There, the set $Z_j$ is defined by:

$$Z_j = \left\{t \mid w_{jt} = 1\right\}, \quad j = 1,2,...,m \text{ and } t = 1,2,...,n.$$

Where $w_{jt}$ takes two values, 1 if measurement $j$ is associated to target $t$, 0 else.

Based on the validation matrix $\Omega(w_{jt})$, data association hypotheses [14] are generated subject to two restrictions: (1) each measurement can have only one origin, and (2) no more than one measurement originates from a target.

In a JPDA scenario, the above two constraints can be easily translated into the language of exhaustive search problem for data association. Usually, an $m$-tuple, $\left(X_1, X_2,...,X_p,...,X_q,...,X_m\right)$, is a solution the these two constraints are satisfied:
1. If $p \neq q$, $X_p \neq 0$, and $X_q \neq 0$, then $X_p \neq X_q$,
2. If $X_p = X_q$ and $p \neq q$, then $X_p = X_q = 0$.

All data association hypotheses can be generated by solving the exhaustive search problem considered above.

Here we use the specialized DFS algorithm proposed in [12] for the generation of data association hypotheses.
In general, in exhaustive search problem, no solutions are known in advance. However, in the problem of data association, a solution which is always known, is (0,0,…,0). The other solutions can be generated systematically from various valid combinations of non-zero values of the elements. For more precision see [12].

## V. DFS-JPDA-IMM-PF ALGORITHM

The principle of the JPDA algorithm is the computation of probabilities association for each track and new measurement. These probabilities are then used as weighting coefficients in the formation of the averaged state estimate, which is used for updating each track. For a better description of the JPDA algorithm, see [2,5].

The combination of the JPDA and the IMM-PF algorithms done as follows. A single set of validated measurements for JPDA-IMM-PF is obtained by considering the intersection $Z_k$, of $r$ sets of measurements corresponding to individual models:

$$Z_k = \bigcap_{j=1}^{r} Z_k^j$$

Where $Z_k^j$ represents the set of validated measurements under the assumption that model **j** is effective. The combined likelihood functions for the **r** modes of the IMM-PF algorithm are computed as in [6].

The prior mixed state estimates for model **j** and the validation regions for individual models are also computed as in [2,6]. The new mode probabilities, output state estimates, and corresponding error covariances are obtained as in [2,6].

## VI. SIMULATIONS AND RESULTS

In this section, we perform some simulations to evaluate our algorithm (DFS-JPDA-IMM-PF).

The motion models considered are: - constant velocity on straight line ($M_1$), -constant acceleration on straight line ($M_2$), - constant velocity on circle ($M_3$), - constant acceleration on circle ($M_4$).

To explore the capability of our JPDA-IMM-UKF algorithm to track maneuvring targets, various scenarios are considered; among of them we select the typical case of three highly maneuvring targets with crossing trajectories.

We assume that the target is in a 2-D space and its position is sampled every T=1s. we run the DFS-JPDA-IMM-PF with 1000 samples in each mode.
- The probability transition matrix of four models is

$$p = \begin{bmatrix} 0.97 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.97 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.97 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.97 \end{bmatrix}$$

- The initial probability of selecting a model is 0.25, that's to say, at the start all models have the same chance to be selected.
- The curvilinear abscissa *s* (.) remains continuous even if a trajectory jump occurs.

### A. Considered scenario

We consider that we have to track simultaneously three maneuvring targets. In order to complicate the scenario, we suppose that the targets follow during there movements, crossing trajectories.

#### a)    Target 1(black):

The target starts moving according to model $M_1$ until the 50th sample when an abrupt trajectory change occur and still moving according to this during 50 samples (switching from model $M_1$ to $M_3$).

#### b)    Target 2 (blue):

The target starts moving according to model $M_3$ until the 50th sample when an abrupt acceleration about 0.2 m/s$^2$ occur and still moving according to this during 50 samples (switching from model $M_3$ to $M_4$).

#### c)    Target 3 (green):

The target starts moving according to model $M_1$ until the 50th sample when an abrupt acceleration about 0.2 m/s$^2$ occur and still moving according to this during 50 samples (switching from model $M_1$ to $M_2$).

#### d)    Target 4 (red):

As the target 3 ,the target 4 starts moving according to model $M_1$ until the 50th sample when an abrupt acceleration about 0.2 m/s$^2$ occur and still moving according to this during 50 samples (switching from model $M_1$ to $M_2$).
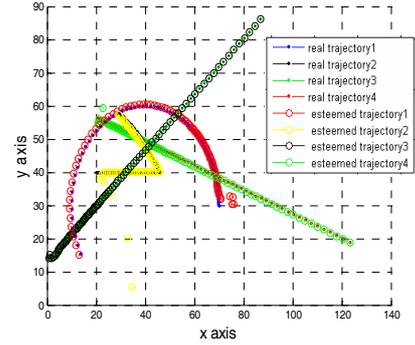


Fig.2. Real and Esteemed Trajectories
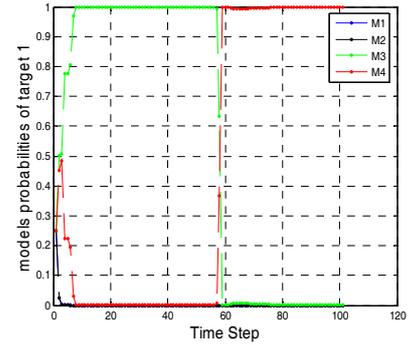


Fig.3. Models Probabilities for target 1
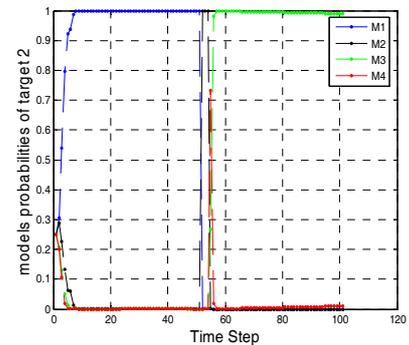


Fig.4. Models Probabilities 2
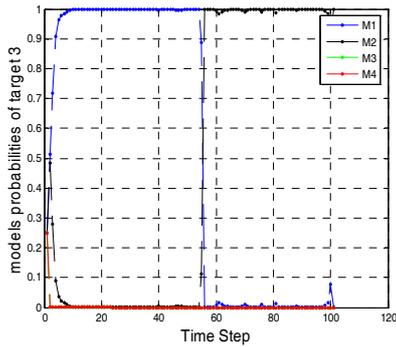
Fig.5. Models Probabilities 3
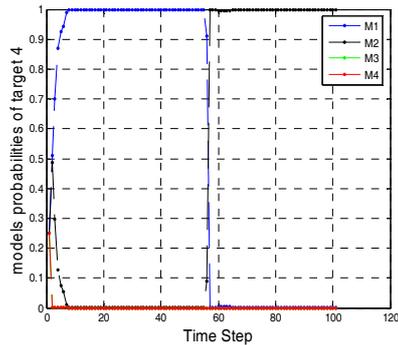


Fig.6. Models Probabilities 4
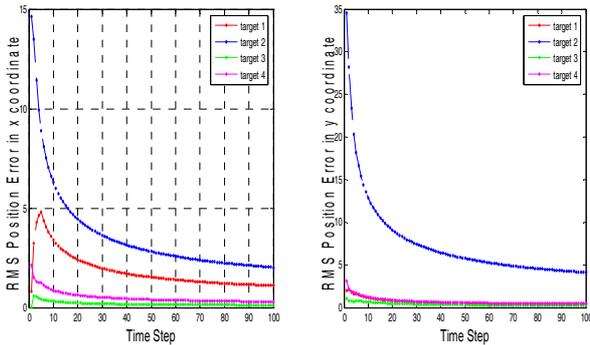


Fig.7. RMS x and y position error
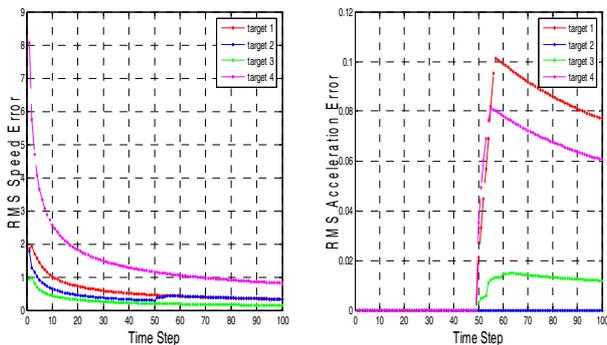


Fig.8. RMS Acceleration and Speed Error

*B. Results interpretation*:

Figure 2 shows that the esteemed and the real trajectory for the three targets are superposable and almost identical even if an abrupt change occurs on the tracked target dynamic. This result is confirmed by the figures (3,4,5,6,7,8), from this we can say that the tracker based IMM-PF algorithm is a pertinent solution to the problem of visual-based tracking highly maneuvering targets. In the Other hand figure 2 shows also that the data association is correctly done even if the trajectories cross each other. This should permit us to say that the JPDA algorithm computes perfectly and its combination with the IMM-PF algorithm (DFS-JPDA-IMM-PF) would be an efficient solution to the problem of highly maneuvering multi-target visual-based tracking.

## VII. CONCLUSION

The model-based body motion estimation by using data coming from visual sensors still an open problem on which we try to provide a contribution. In this paper we presented a nonlinear algorithm which attempts to track efficiently a highly maneuvering target whose trajectory and/or dynamic could change abruptly, and the noise distribution is not necessary Gaussian; the algorithm proposed is noted IMM-PF. To extend this algorithm to multi-target case, we combined the later with a fast version of the JPDA algorithm noted DFS-JPDA to ensure good data association.

Simulations show that the DFS-JPDA-IMM-PF is a good investment while we are asked to track a highly maneuvrable targets whose measurement and/or state models present a strong nonlinearities, and the noises are not Gaussian and when there different trajectories cross each other.

## REFERENCES

[1] Y.Bar-Shalom and T.E.Fortman, "Tracking and Data Association," *Mathematics in Science and Engineering*, volume 179, Academic Press, 1988.

[2] Y.Bar-Shalom and X. R. Li, "Estimation and Tracking, *Principles, Techniques and Software*," *Artech House*, Boston, MA (USA), 1993.

[3] B.Espiau, F.Chaumette, and P.Rives, " A new approach to visual servoing in robotics,"*IEEE Transactions on Robotics and Automation,* 8(3):313-326, June 1992.

[4] P.Danes, M.S.Djouadi, D.Bellot, "A 2-D Point-Wise Motion Estimation Scheme for Visual-Based Robotic Tasks," *7th International Symposium on Intelligent Robotic Systems (SIRS'99)*, Coimbra (Portugal), 20-23 July 1999, pp.119-128

[5] Y.Bar-Shalom and X. R. Li, "Multi-target Multi-sensor Tracking: Principles and Techniques," *Storrs, CT* , YBS Publishing, 1995.

[6] M.S.Djouadi, A.Sebbagh, D.Berkani, ''IMM-UKF and IMM-EKF Algorithms for Tracking Highly Maneuvring Target,'' *Archive of Control Sciences*, Vol: 1, issue: 1, 2005.

[7] M. Hadzagic, H. Mchalska, A. Jouan, " IMM-JVC and IMM-JPDA for closely maneuvering targets," IEEE, 2001.

[8] Y. Boers, JN. Driessen, " Interacting multiple model particle filter", *IEE Proc- Radar Sonar Navig, Vol.150, No. 5,* October 2003.

[9] S. McGinnity, G.W. Irwin, "Multiple model bootstrap filter for maneuvering target tracking," *IEEE Trans. Aerosp. Electron. Syst.,2000, 36, (3), pp. 1006–1012.*

[10] S. McGinnity, G.W. Irwin, "Manoevering target tracking using a multiple-model bootstrap filter," in Doucet, A., de Freitas, N. and Gordon, N. (Eds.): 'Sequential Monte Carlo methods in practice' (Springer, New York, 2001), pp. 247–271

[11] N.J. Gordon,"A hybrid bootstrap filter for target tracking in clutter," IEEE Trans. Aerosp. Electron Syst., 1997, 33, (1), pp. 353–358

[12] B.Zhou, N.K. Bose, " Multitarget tracking in clutter: Fast Algorithm for Data Association," *IEEE trans. Aero*. Elect. Vol.29, N°.2 April 1993.

[13] E.M. Reingold, J. Neivergelt, N. Deo, " Combinational Algorithms, Theory and Practice," *Prentice-Hall*, 1977.

[14] T.E. Fortmann, Y. Bar-Shalom, M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, OE-8 , July 1983.