International Journal of Engineering and Technology Vol. 1, No.3, August, 2009
ISSN: 1793-8236

Multiple Maneuvering Targets Tracking Using Kalman and Real-Time Particle Filter A Comparison

S.Vasuhi[1,] V. Vaidehi[2,] Midhunkrishna P.R[3]

Abstract - **In this paper, a comparison between thetwo algorithms for tracking multiple maneuvering targets in heavy clutter is done. First one is by using Multiple Hypothesis Tracking (MHT) and nonlinear non-Gaussian Kalman filter and the second one is by combining MHT and Real-Time Particle Filter (RTPF). The main difficulty in multiple maneuvering targets tracking is the nonlinearity associated with target states. The multiple target's motion modes in highly non-linear states are detected by using Multiple Hypothesis Tracking (MHT). In MHT, hypothetical tracks are generated, so the computational burden increases exponentially with number of tracks. So the 1-backscan MHT algorithm is a good alternative because its having good tracking performance and limitation of computation time. The nonlinear non-Gaussian Kalman filter is used to track the target with high maneuver rate and also it gives less probability of missing the target. Tracking by Real-time particle filter (RTPF) uses all sensor information even when the filter update rate is below than that of sensors. In RTPF each posterior is represented as mixture of sample sets, where each mixture component integrates one observation arriving during a filter update. RTPF eliminate the problem of filter divergence due to an insufficient number of independent samples.**

Index Terms— **Multiple Hypothesis Tracking, nonlinear non-Gaussian Kalman filter, RTPF, tracking of multiple maneuvering Targets.**

I. INTRODUCTION

In multiple targets tracking (MTT) the main objective is to partition the sensor data into sets of observations, or tracks, produced by the same source. Once tracks are formed and confirmed, the number of targets can be estimated and quantities, such as target velocity, future predicted position, and target classification characteristics, can be computed for each track. An important distinction when comparing MTT processing methods is between batch and recursive methods. Batch processing techniques represent the ideal situation where no information is lost due to preprocessing because all observations are processed together. On the other hand, by

using recursive methods, processing is done at each scan using data received on that scan to update the results of previous processing [1]. The tracking of a maneuvering targets is a highly nonlinear and challenging problem that involves, at every time instant, the estimation not only of the unknown state (composed of position, velocity and acceleration of the target) in the dynamic model that describes the evolution of the target, but also the underlying model that accounts for the regime of movement [2].

If the standard sequential processing approach is taken, the most likely combination will be chosen after each data set is received. Using the Multiple Hypothesis Tracking (MHT) approach, a number of candidate hypotheses will be generated and evaluated later as more data are received. Thus, the capability of using later measurements to aid prior correlation decisions is allowed. However, the method is recursive so that data sets only need be processed as they are received. The non-Gaussian Kalman filter proposed here seems to be optimal under the minimum-mean-square error (MMSE) criterion for non-Gaussian problem. The non-Gaussian linear DSS model, in which the PDFs of the system initial state, system noise, and the posterior state PDFs are modeled by the Gaussian mixture model (GMM), was assumed. Using the property that any PDF can be approximated by a mixture of finite number of Gaussians, a recursive method based on the MMSE estimator for GMM-distributed random vector was derived. This algorithm estimates the posterior PDF of the system state by the GMM, and therefore it can be effectively used for maneuvering target tracking. The main difficulty in real-time recursive estimation is the mismatch between incoming sensor data rate and the filter update rate. In usual case the filter will discard the sensor information which arrives during update process. In Real-time particle filter instead of discarding sensor readings, it distributes the samples among the different observations arriving during a filter update.[5].Hence, RTPF represents densities over the state space by mixtures of sample sets. Each mixture components are assigned with a weight corresponding to probability density function so as to minimize the approximation error introduced by the mixture representation.

II. PROBLEM FORMATION

Suppose that there are N targets and the target set is denoted by $T_N = \{1, 2, ..., N\}$. For the target r (r ϵ T_N), its dynamic equation and measurement equation are denoted by the equations (1) and (2) respectively [3].

Manuscript received June 5, 2009.
S.Vasuhi, Lecturer, Department of Electronics Engg. MIT-Anna University, Chrompet, Chennai-600 044, INDIA
V. Vaidehi, Professor, Department of Electronics Engg MIT-Anna University, Chrompet, Chennai-600 044, INDIA.
Midhunkrishna P, Student-ME (AVIONICS), Department of Electronics Engg., MIT-Anna University, Chrompet, Chennai-600 044, INDIA

$$x_k^r = F_{k-1} x_{k-1}^r + G_{k-1} v_{k-1}^r \quad (1)$$

$$z_k^r = H_k x_k^r + w_k^r \quad (2)$$

Where, x_k^r is the state vector of target r at time k and z_k^r is the measurement vector of target r at time k. F_{k-1} and G_{k-1} are the system transition matrix and the input matrix at time k−1 respectively. H_k is the measurement matrix. v_{k-1}^r is a non Gaussian driving noise for maneuvering target r and w_k^r is a zero-mean white Gaussian measurement noise vector. The initial state vector x_{-1}^r, v_{k-1}^r and w_k^r of target r are independent random process GMM distributed [4]. For all these targets, the transition process from model i to model j is governed by Markov chain whose transition probability P ij is known.

$$x_{-1}^r \quad GMM(a_{xj}^r[-1], m_{xj}^r[-1], \Gamma_{xj}^r[-1]; j = 1, ..., M)$$
$$v_{-1}^r \quad GMM(a_{vn}^r[-1], m_{vn}^r[-1], \Gamma_{vn}^r[-1]; n = 1, ..., N)$$
$$w_k^r \quad N^c(0, \Gamma_w^r[k]) \quad (3)$$

where $GMM(a_m, m_{ym}, \Gamma_{ym}; m = 1,..., M)$ denotes an M^{th}-order complex Gaussian mixture distribution with weights, mean vectors, and covariance matrices. The PDF of a GMM distributed random vector y is given by

$$f_y(y) = \sum_{m=1}^{M} a_{ym} \Phi(y; q_{ym}) \quad (4)$$

Where $\Phi(y; q_{ym})$ is a complex Gaussian PDF and q_{ym} contains the mean vector, m_{ym} and the covariance matrix, Γ_{ym}. The estimation of state vector for target r from measured data by using conditional expectation estimator, $\hat{x}_k^r = E(x_k^r \mid z_k^r)$. From the state estimated vector, the tracks of individual targets are separated and maintained by Multiple Hypothesis Tracking (MHT).

III. Non-linear Non-gaussian Kalman Filter Algorithm

In this section, the basic steps of nonlinear non-Gaussian Kalman filtering and Multiple Hypothesis Tracking (MHT) arithmetic in one cycle is described as follows.

The conditional probability that the data $X^r[k]$ by the mixture component $h_j^r[k]$ is $p(h_j^r[k] \mid X^r[k])$. The state mixture parameters at instance k-1 for target r are the mean estimation of the jth component of the mixture is $m_x^r[k-1 \mid k-1, h_j[k-1]]$, the prediction covariance matrix for the l^{th} mixture component is $M_x^r[k-1 \mid k-1, h_j[k-1]]$ and weight of the j^{th} component is $a_{xj}^r[k-1]$ [4].

The observation model in tracking systems is nonlinear because the observations are given in polar coordinates [8]. For nonlinear problems there is no general analytic expression for the posterior PDF and only approximated estimation algorithms are existed. The extended Kalman filter (EKF) is the most popular approach for recursive nonlinear estimation. The main idea of the EKF is first-order linearization of the estimation problem and the posterior PDF is assumed to be Gaussian. In nonlinear systems the PDF of the state may be multi-modal. The Gaussian approximation of this multi-modal distribution leads to poor tracking performance.

The following Steps are involved in Nonlinear non-Gaussian Kalman filtering

- Prediction of the state mixture parameters:

$$a_{xj}^r[k] = a_{vn}^r[k] \cdot p(h_j^r[k-1] \mid X^r[k])$$
$$m_x^r[k \mid k-1, h_j[k]] = F_{k-1} m_x^r[k-1 \mid k-1, h_j[k-1]] + m_{vn}^r[k]$$
$$M_x^r[k \mid k-1, h_j[k]] = F_{k-1} M_x^r[k-1 \mid k-1, h_j[k-1]] F_{k-1}^T + \Gamma_{vn}^r[k]$$
$$\forall j = 1, 2, ..., M, \forall n = 1, 2, ..., N$$

- Prediction:

$$\hat{x}_{k|k-1}^r = \sum_{j=1}^{M} a_{xj}^r[k] m_x^r[k \mid k-1, h_j[k]]$$

- Kalman gain:

$$K_j^r[k] = M_x^r[k \mid k-1, h_j[k]] H_k^{rT} \cdot (\Gamma_w^r[k] + H_k^r M_x^r[k \mid k-1, h_j[k]] H_k^{rT})$$

Estimation of mixture parameter:

$$m_x^r[k \mid k, h_j[k]] = m_x^r[k \mid k-1, h_j[k]] + K_j^r[k](z_k^r - H_k^r \hat{x}_{k|k-1}^r)$$
$$M_x^r[k \mid k, h_j[k]] = (I - K_j^r[k] H_k^r) M_x^r[k \mid k-1, h_j[k]]$$

The conditional probability that the data $X^r[k]$ by the mixture component $h_j^r[k]$ is calculated as follows

$$p(h_j^r[k] \mid X^r[k]) = \frac{a_{xj}^r[k] N^c(0, \Gamma_{xj}^r[k])}{\sum_{i=1}^{N} a_{xi}^r[k] N^c(0, \Gamma_{xi}^r[k])}$$

- Estimation:

$$\hat{x}_{k|k}^r = \sum_{j=1}^{M} p(h_j^r[k] \mid X^r[k]) m_x^r[k \mid k, h_j[k]]$$

These are the steps of nonlinear non-Gaussian Kalman filtering used to estimate the state vector of target r at instance k ($\hat{x}_{k|k}^r$). The residual and its covariance matrix are calculated as

$$\tilde{z}_k^r = z_k^r - H_k^r \hat{x}_{k|k-1}^r$$
$$S_k^r = H_k^r M_x^r[k \mid k-1, h_j[k]] H_k^{rT} + \Gamma_w^r[k]$$

IV. PATICLE FILTER ALGORITHM

Particle Filter Algorithm involves the following steps

5) Generate particles for first M random numbers.
6) Perform the weight computation and weight normalization
7) Resambling is used to avoid the problem of degeneracy of the algorithm, which is, avoiding the situation that

all but one of the importance weights close to zero.The performance of the algorithm can also be affected by proper choice of resampling method.

8) estimated values are computed. [4]

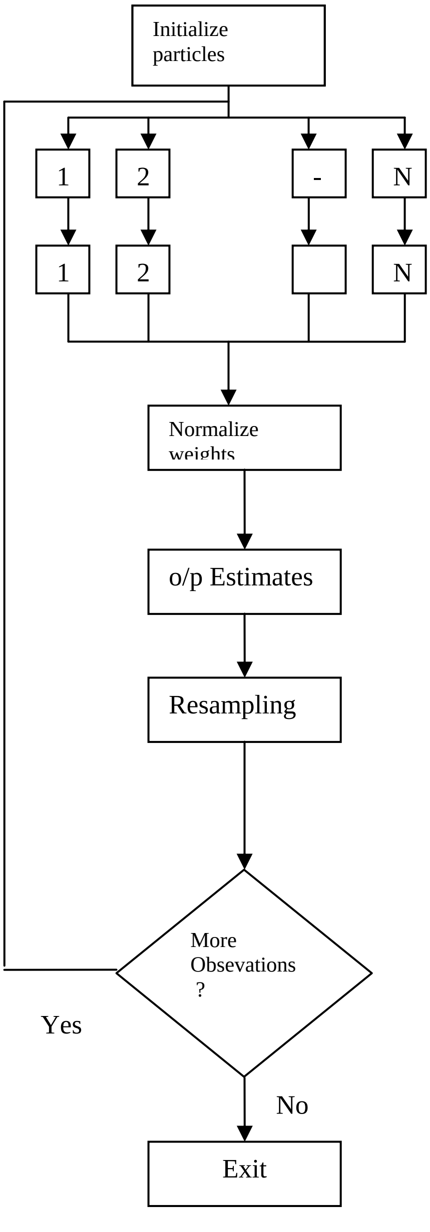

Fig 1.Flow chart of Particle filter

Particle filters represents the belief $Bel(x_t)$ by a set S_t of N_p weighted samples

$S_t = \{ (x_t^{(i)}, w_t^{(i)}) | i = 1,..N_p \}$ Where each

$x_t^{(i)}$ is a state and the $w_t^{(i)}$ are nonnegative numerical factors called importance weights, which sum up to one.

$$w_k^{(i)} \alpha \frac{p(x_{0:k}^{i} \mid z_{1:k})}{q(x_{0:k}^{i} \mid z_{1:k})}$$

Where q (.) called an importance density.

A. *Real Time Particle Filter*

The general assumption underlying particle filter is that all samples can be updated whenever new sensor information arrives. Under real time condition, it is not possible to complete the update before the next sensor measurement arrives. This can be the case for computationally complex sensor models or whenever the underlying posterior requires large sample sets. The majority of filtering approaches deals with this problem solved by skipping sensor information that arrives during the update of the filter. While this approach works reasonably well in many situations, it is prone to miss valuable sensor information.

Let n be the number of samples required by the particle filter. The time interval Δ between two observations is called observation interval. Assume that the resulting update cycle of the particle filter takes k Δ and is called the estimation interval or estimation window. Accordingly, k observations arrive during one estimation interval and this number is called the window size of the filter, ie. The number of observations obtained during a filter update.

In Real time particle filter, samples are partitioned into subsets among sensor information over estimation windows. The size for the each partitioned subset is selected such that particle filter iteration can be performed before new sensor information is acquired. At the end of estimation window, each subset is assigned with weights using the measurement. Resample the particles from each subset according to their weights. The size of the resampled particle set is n/k where n is the number of particles in the estimation window; k is the number of observation in the estimation window or window size of the filter. This resampled particle set is used in the next estimation window as prior belief[5].

B. *Mixture Representation*

Let us consider one estimation window contains k observations.The optimal belief can be represented as

$$Bel_{opt}(x_{tk}) \alpha \int ... \int \prod_{i=1}^{k} p(y_{ti} \mid x_{ti}) p(x_{ti} \mid x_{ti-1}, u_{ti-1}).Bel(x_{t0}) dx_{t0} ..... dx_{tk-1}$$

Where $Bel(x_t)$ denotes the belief generated in the previous estimation window.

RTPF generates k such beliefs, one for each observation.

C. Optimizing mixture weights

The mixture weights α is determined by minimizing Kullback-Leibler -divergence between Bel_{mix} and Bel_{opt} [8]

$$= \arg\min \int Bel_{mix}(x_{tk} \mid a).\log Bel_{mix}(x_{tk} \mid a) / Bel_{opt}(x_{tk}) d_{xtk}$$

V. 1-BACKSCAN MULTIPLE HYPOTHESES TRACKING

The 1-backscan MHT involves the following steps

1: Hypotheses Construction

It uses the structure branch algorithm for hypotheses tree construction. The main difference in 1-backscan MHT and zero backscan MHT is the hypotheses tree formed an modified in each observation.

2:Bayesian Track Scoring

A relatively simple sequential technique for track scoring can be developed by applying Bayes' rule. This technique goes beyond the method SPRT (Sequential Probability Ratio Test) because prior probabilities and track update residual information are readily included. Also, the same method will be applied to track deletion [1]. Using Bayes' rule, the probability of true track correlate with measurement data D is

$$p\left(T/D\right)=\frac{p\left(D/T\right)p_0(T)}{p(D)} \qquad (5)$$

where, $p\left(D/T\right)$ is the probability of receiving measurement data D given that a true target is present. Also, $p_0(T)$ is the priori probability if a true target appearing within the scan volume. The term p(D) is the probability of receiving the data D and is given by

$$p(D)=p\left(D/T\right)p_0(T)+p\left(D/F\right)p_0(F) \qquad (6)$$

where $p\left(D/F\right)$ and $p_0(F)$ are defined for false target in the same manner that $p\left(D/T\right)$ and $p_0(T)$ were defined for true targets. Noting that $p_0(T)$ =1 - $p_0(F)$, combining (5) and (6), and dividing numerator and denominator by $p\left(D/F\right)$ gives

$$p\left(T/D\right)=\frac{L(D)\,p_0(T)}{L(D)\,p_0(T)+1-p_0(T)} \qquad (7)$$

where L(D) is the likelihood ratio for the data as defined

$$L(D)=\frac{p\left(D/T\right)}{p\left(D/F\right)}$$

Equation (7) can be modified in convenient form for recursive computation as L_k to be the likelihood ratio for the data received at k^{th} scan to be correlated with the true track. Likelihood L_k associated with data set D_k must be determined by first defining $p\left(T/D_k\right)$ and $p\left(D_k/F\right)$. Dropping subscript k, for a true target $p\left(D/T\right)$ is taken to be the product of the probability of detection PD and the Gaussian likelihood function defined as

$$g_{ij}=\frac{\exp(-d_{ij}^{\,2}/2)}{(2p)^{M/2}\sqrt{|S|}}$$

It is likelihood function associated with the assignment of observation j to track i by assuming the Gaussian distribution for the residual. Similarly, $p\left(D/F\right)$ is taken to be the probability of a false target return times the likelihood function ($1/V_G$) associated with the assumed uniform distribution of false returns within the volume V_G of the gated region. Thus,

$$L_k=\frac{P_D e^{-d^2/2}V_G}{P_F(2p)^{M/2}\sqrt{|S|}} \qquad (8)$$

Where d^2 is the normalized distance function and $|S|$ is determinant of the residual covariance matrix. Equation (8) can be simplified by noting that $P_F=b_{FT}V_G$, where b_{FT} is the false target density. Thus (8) becomes

$$L_k=\frac{P_D e^{-d^2/2}}{b_{FT}(2p)^{M/2}\sqrt{|S|}} \qquad (9)$$

Taking log of equation (9) we get log likelihood score of hypotheses and given as

$$L_k=\ln\left\{\frac{P_D}{b_{FT}(2p)^{M/2}\sqrt{|S|}}\right\}-\frac{d^2}{2} \qquad (10)$$

The new target probability can be defined in terms of new target density and false target density as

$$p_0(T)=\frac{b_{NT}}{b_{NT}+b_{FT}} \qquad (11)$$

Equation (6) to (11) provides a convenient sequential scoring scheme that can be adjusted to the environment

3: Track and Hypothesis Scoring

Each track has a score which is essentially the log likelihood of the hypothesis that the set of observations in the track are from the same source. The track is a collection of false alarms. The score is initially set to zero at the time of the first observation. Thereafter, upon the receipt of data on scan k, the score for track i is updated according to the relationship.

$$\mathrm{L_i(k)=L_i(k-1)+\Delta L(k)}$$

Where,

$$\Delta\mathrm{L}\ (\mathrm{k})=\ln(1-\mathrm{P_D})\ ;\ \text{no track update}$$

$$=\quad \Delta\mathrm{L_G}\quad ;\ \text{track updated}$$

$$\Delta\mathrm{L_G}=\ln\left\{\frac{\mathrm{P_D}}{b_{\mathrm{F}}(2p)^{\mathrm{M}/2}\sqrt{\mathrm{S_k^r}}}\right\}-\frac{\mathrm{d^{r2}}}{2} \qquad (12)$$

$\mathrm{P_D}$ = estimated probability of detection

b_{F} = false target density

M = measurement dimensionality.

$\mathrm{S_k^r}$ = residual covariance matrix

P, R = Kalman filter predication, measurement covariance matrices

H = measurement matrix

$\mathrm{d^{r2}}$ = normalized statistical distance function

$= \tilde{\mathrm{z}}_{\mathrm{k}}^{\mathrm{r\,T}}\mathrm{S_k^{r\,-1}}\tilde{\mathrm{z}}_{\mathrm{k}}^{\mathrm{r}}$

$\tilde{\mathrm{z}}_{\mathrm{k}}^{\mathrm{r}}$ = measurement residual vector of target r at time k.

4: Track Management

Hypotheses are constructed from sets of compatible tracks. The hypothesis score is the sum of the scores of the tracks contained in the hypothesis. Given hypothesis scores,

L_{H_j}, the probability, the $P(H_j)$ of hypothesis j can be computed, using all J hypotheses, from

$$P(H_j) = \frac{\exp(L_{H_j})}{[1 + \sum_{j=1}^{J} \exp(L_{H_j})]} \quad (13)$$

Note that a given track can be contained in more than one hypothesis. Thus, the probability of a track is the sum of all hypotheses that contain the track. The number of tracks must be controlled by standard track and hypothesis pruning methods are utilized. Also, similar tracks are merged. The end result is that a number of tracks that were formed are deleted. Thus, a reduced set of tracks is maintained until the next scan of data, where the process is continued.

A. Pruning Hypotheses

The manner in which branches are eliminated (or pruned) from the hypothesis tree is, like many issues in MTT, highly dependent upon the application. One technique is to remove hypotheses with probabilities that fall below some fixed predetermined threshold. A disadvantage of this type of pruning is that it does not take into consideration the computational resources.

Another approach to pruning, called the breadth approach, is to allow only a predetermined fixed number (M) of hypotheses to be maintained. This technique involves ranking the hypotheses and choosing only the M most likely, as measured either by the probabilities or the score functions. A similar method is to rank and sum the probabilities of the most likely hypotheses. When this sum exceeds a threshold the remaining hypotheses are then deleted.

B. Combining Hypotheses

As data are accumulated certain hypotheses may tend to become similar. For example, two hypotheses might differ only with regard to correlation uncertainties that occurs several scans ago. Then, if the tracks involved in the previous correlation uncertainties have received the same recent updates, the past associations may no longer be important and the hypotheses can be combined. This is accomplished by first determining which hypotheses have the same number of tracks. Then, it must be determined if each track in one hypothesis has a corresponding track that is similar to it in the other hypothesis.

C. Hypothesis Clustering

A cluster is a group of hypotheses, and associated tracks that do not interact with any other group of hypotheses (contained within other clusters). The hypotheses within a cluster will not share observations with the hypotheses within any other cluster. The basic purpose of clustering is to divide the large tracking problem into a number of smaller ones that can be solved independently. This can greatly reduce the number of hypotheses that must be determined.

A new cluster is initiated any time an observation is received that does not fall within the gates of any track contained in an existing cluster. The cluster is initiated on the observation using the alternatives (true target or false alarm) associated with its source. A new cluster is initiated on a track that is contained in all hypotheses of a previous cluster. The track is then removed from the old cluster. In order that clusters remain distinct, the gates of the tracks within the cluster must not overlap. Thus, when an observation falls within the gates of two or more tracks from different clusters, the clusters are merged.[1][2]

VI. MHT WITH NONLINEAR NON-GAUSSIAN KALMAN FILTER ALGORITHM

In fig.2, the flow diagram of MHT with nonlinear non-Gaussian Kalman filter for multiple maneuvering targets is shown.

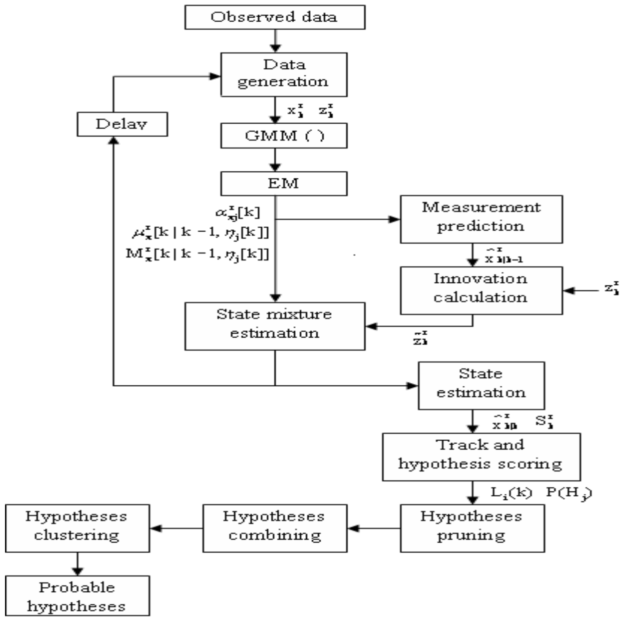

Fig. 2. Flow diagram of MHT with nonlinear non-Gaussian Kalman filter

VII. SIMULATION RESULTS

The simulation results show the tracking of two maneuvering targets. Both the targets are in coordinated turn and also have sudden steering from their coordinated turns. The state vectors of two targets are having its position, velocity and acceleration in 2D as [x y x1 y1 x2 y2]. In this algorithm the tracking system is designed based on Singer's acceleration model [6]. The transition matrix (a) and input matrix (b) of the system are considered by defining α to be the inverse of maneuver time constant and the matrices are defined as follows,

$$\Phi = \begin{bmatrix} 1 & 0 & T & 0 & a1 & 0 \\ 0 & 1 & 0 & T & 0 & a1 \\ 0 & 0 & 1 & 0 & b1 & 0 \\ 0 & 0 & 0 & 1 & 0 & b1 \\ 0 & 0 & 0 & 0 & c1 & 0 \\ 0 & 0 & 0 & 0 & 0 & c1 \end{bmatrix} \quad (14)$$

$$B = \begin{bmatrix} d1 & 0 \\ 0 & d1 \\ a1 & 0 \\ 0 & a1 \\ b1 & 0 \\ 0 & b1 \end{bmatrix} \qquad (15)$$

where $a1 = \frac{1}{a^2}[-1 + aT + e^{-aT}]$ $b1 = \frac{1}{a}[1 - e^{-aT}]$,

$c1 = e^{-aT}$, $d1 = \frac{1}{a^2}[b1 + \frac{T^2 a^2}{2} - T]$

The maneuvering excitation covariance matrix Q(k) is given as

$$Q(k) = \begin{bmatrix} q11 & q12 & q13 & 0 & 0 & 0 \\ q12 & q22 & q23 & 0 & 0 & 0 \\ q13 & q23 & q33 & 0 & 0 & 0 \\ 0 & 0 & 0 & q11 & q12 & q13 \\ 0 & 0 & 0 & q12 & q22 & q23 \\ 0 & 0 & 0 & q13 & q23 & q33 \end{bmatrix} \qquad (16)$$

Where

$$q11 = \frac{1}{2a^5}[1 - e^{-2aT} + 2aT + \frac{2a^3T^3}{3} - 2a^2T^2 - 4aTe^{-aT}$$

$$q12 = \frac{1}{2a^4}[1 + e^{-2aT} - 2aT + a^2T^2 + 2aTe^{-aT} - 2e^{-aT}]$$

$$q13 = \frac{1}{2a^3}[1 - e^{-2aT} - 2aTe^{-aT}]$$

$$q22 = \frac{1}{2a^3}[-e^{-2aT} + 2aT - 3 + 4e^{-aT}]$$

$$q23 = \frac{1}{2a^2}[1 + e^{-2aT} - 2e^{-aT}], \quad q33 = \frac{1}{2a}[1 - e^{-2aT}]$$

σ_m is the maneuver standard deviation [6].

The constants used for calculating the score of tracks in hypothesis are

P =0.9, estimated probability of detection

β =0.5, false target density

M = 2, measurement dimensionality

The Comparison of MHT with nonlinear non-Gaussian Kalman filter and MHT with Real-Time particle filter is shown in fig 1.

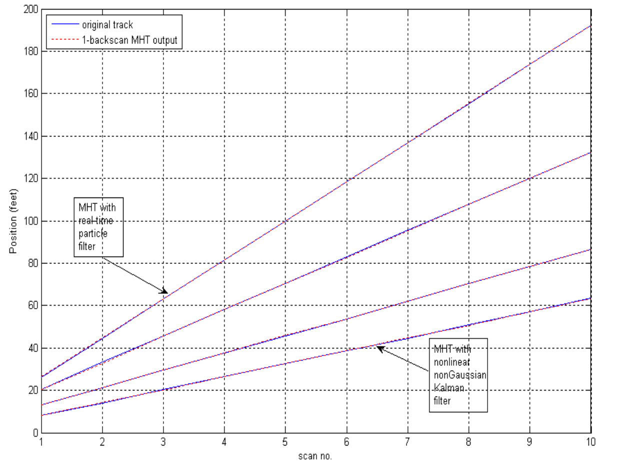

Fig. 3. Comparison of MHT with nonlinear non-Gaussian Kalman filter and MHT with Real-Time particle filter

In Fig. 4 & 5, error estimation for both methods are given. Root Mean Square Error (RSME) for MHT with Real-Time particle filter is less than 0.3,but that for nonlinear non Gaussian kalman filter is approximately equal to 1.

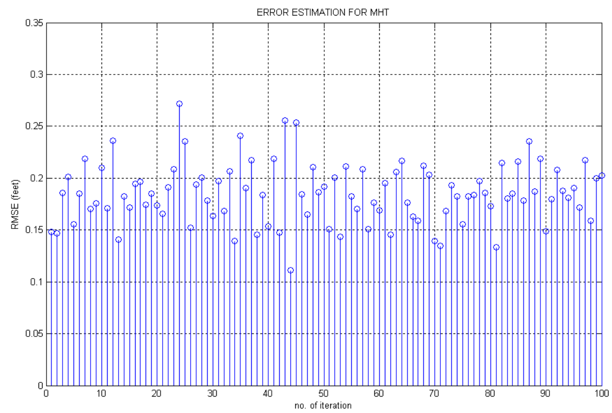

Fig 4. RMSE of MHT with Real-Time particle filter

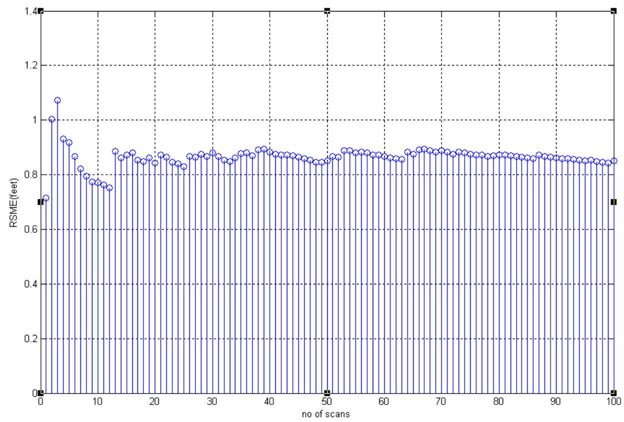

Fig 5. RMSE of MHT with nonlinear non Gaussian kalman filter

Hence for tracking multiple maneuvering targets, MHT with RTPF is an optimum choice when compared to already existing methods like JPDA, MHT with non-linear non Gaussian Kalman filter, etc., since RTPF eliminates the problem of filter divergence and very low estimation error

VIII. CONCLUSION

In this paper, a comparison of two target tracking algorithms is done. The existing technique for tracking targets is done by kalman filter. Since in real-time applications the state of the targets is nonlinear in nature, it won't give good results. The disadvantages are overcome by 1-backscan MHT with real-time particle filter. The computational burden of MHT was reduced by using 1-backscan (for calculation one step previous scan data are used). From simulation results, we concluded that the performance of 1-backscan MHT with real-time particle filter was better than MHT with nonlinear nonGaussian Kalman filter.

REFERENCES

[1] Blackman S.S. "Multiple target tracking withradar applications" Artech House 1986.
[2] I.Bilik, J.Tabrikian "Maneuvering TargetTrackingUsing the Nonlinear Non-Gaussian KalmanFilter" IEEE Proceedings 2006.
[3] X. Rong Li, Vesselin P. Jilkov "A Survey of Maneuvering Target Tracking: Dynamic Models" Proceedings of SPIE Conference on Signal and Data Processing of Small Targets April 2000.
[4] I gal Bilik, Joseph Tabrikian "Optimal Recursive Filtering Using Gaussian Mixture Model" IEE Proceedings May 2005.
[5] Blackman S., Dempster R., Broida T. "Multiple Hypothesis Track Confirmed For Infrared Surveillance Systems" IEEE Transactions on Aerospace and Electronic Systems 1986.
[6] Robert A. Singer "Estimating Optimal Tracking Filte Performance for Manned Maneuvering Targets" IEEE Transactions on Aerospace and Electronic Systems vol. AES-6, no. 4 July 1970.
[7] Cody Kwok, Dieter Fox, and Marina Meil, Real-Time Particle Filters,Proceedings of the IEEE, vol. 92, no. 3, MARCH 2004
[8] M. S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A Tutorial on Particle Filters for Online Nonlinear/non-Gaussian Bayesian Tracking. IEEE Transactions on Signal Processing, vol. 50, no. 2, pp. 174–188, Feb. 2002
[9] Gideon Ferreia,' Stereo vision based target tracking for a Gun turret utilizing low performance components', Dept: of electrical & electronics engineering, University of Johannesburg
[10] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," IEE Proceedings F, vol. 140, no. 2, April 1993.
[11] D. Fox, "KLD-sampling: Adaptive particle filters and mobile robot localization," presented at the Conf. Advances Neural Information Processing Systems (NIPS), Vancouver, BC, Canada, 2001.