Commun. Fac. Sci. Univ. Ank. Series A2-A3
V.51(1) pp 1-16 (2007)

PERFORMANCE COMPARISON OF TRACKING ALGORITHMS FOR A GROUND BASED RADAR

GÖKHAN SOYSAL AND MURAT EFE

Ankara University, Faculty of Engineering, Electronics Engineering Department, 06100, Tandogan, Ankara – TURKEY
E-mail: soysal,efe@eng.ankara.edu.tr

(Received March 22, 2007; Accepted April 09, 2007)

ABSTRACT

In this paper, performance of several tracking algorithms are compared when employed in a ground based radar. The algorithms have been compared in terms of percentage track loss and rms estimation errors where the manoeuvrability of the scenarios was varied from low to moderate.

1. INTRODUCTION

Phased array radars are used for ground, air or maritime surveillance and/or tracking. Electronic scanning feature of these radars provide fast and accurate beam direction. This feature helps the radar provide measurements with better accuracy which consequently improves the performance of the tracking algorithm employed. In radars, tracking algorithms are used to estimate the target states of interest (commonly position and velocity) from the noisy measurements that the radar has gathered.

The fact that radar is ship based, ground based or airborne calls for caution in terms designing the tracking algorithm. For instance, if the radar of interest is ship based then special caution has to be taken to deal with heavy sea clutter. Also, performance of ground based radars is hindered by tall objects in the field of view, which in return deteriorates the tracking algorithm's performance. However, electronic beam steering has the answer to this specific problem by allowing predetermined scanning patterns.

This paper presents performance comparison of several target tracking algorithms for a ground based phased array radar. The radar is assumed to rotate mechanically, completing its 360° turn in 2 s in azimuth, while scanning electronically in elevation. Each radar return goes through a detection process and any return that passes this process yileds a measurement for which the corresponding measurement errors and probability of detection (P_D) are calculated. Then, the resulting measurement is fed into the tracker.

The primary aim of the study is to determine the most appropriate tracking algorithm to be employed with a ground based short to mid range phased array radar. Selected tracking algorithms have been tested on selected scenarios where the targets perform mild to moderate manoeuvres at both short and mid ranges.

2. RADAR MODEL

Principles of phased arrays have been applied to radars since the World War II where most of the advances in theory and technology were achieved in the 50s and the 60s. However, they came into operational use in the late 60s and the 70s. The main advantage of the phased array antennas is that the beam of the antenna can be steered electronically in a certain new direction without delay.

In this study the phased array radar is modelled to scan electronically in elevation while rotating mechanically in azimuth. Also the radar is assumed to work in track and search (T&S) mode. It completes its 360° turn in 2 s during which it collects 3D (range, azimuth, elevation) measurements. The radar model is supported with realistic beam scheduling, detection and measurement units. The beam scheduling unit carries out the planning of search and track beams in accordance with T&S logic. The detection unit calculates the SNR for each hit using the predefined radar parameters, target range and target's distance to the beam centre. The calculated SNR is then employed to yield P_D which is subsequently compared with random number that is uniformly distributed between (0-1). Only radar returns whose calculated P_D passes this test is declared as a valid detection. Through this test miss detection is also modelled. The measurement unit produces 3D measurements in spherical coordinates where measurements are assumed to have been taken using amplitude comparison mono pulse technique [1]. The unit computes measurements errors in elevation and azimuth (i.e., DF errors) along with the SNR which is calculated using the beam width and target's distance to the beam centre. The DF errors are than added to the true target position in order to obtain radar measurements.

3. TRACKING ALGORITHMS

Three different tracking algorithms, namely, probabilistic data association filter (PDAF) [2], interacting multiple model probabilistic data association filter (IMMPDAF) [2] and interacting multiple model nearest neighbour (IMMNN) algorithm, in seven different configurations have been tested on the selected scenarios. The configurations belonging to the same algorithm differ either in the number/type of models they posses or the assumption of a-priori knowledge regarding the target manoeuvre. The selected algorithms (and the configurations) are the most widely used and tested tracking algorithms in the literature. All selected algorithms are Kalman based algorithms and have been used with a data association approach to deal with clutter.

3.1 Kalman Filter

Kalman filter is the traditional and most widely used state estimator [3]. The Filter is the general solution to the recursive linear minimum mean square estimation problem and it provides optimal tracking performance provided the following conditions [4] are met; *i)* the state of the target evolves according to a known linear dynamic model driven by a single known input and additive zero-mean, white Gaussian noise with known covariance and *ii)* the measurements are linear functions of the target state corrupted by additive zero-mean, white Gaussian noise with known covariance.

Assuming that the target dynamic process can be modelled in discrete Markov form, then the equation that describes the target dynamics in terms of a Markov process can be written as,

$$X(k+1) = FX(k) + \Gamma V(k) \tag{1}$$

where *X(k)* is the k dimensional target state vector, F is the known transition matrix, Γ is the known disturbance transition matrix and *V(k)* is the unknown zero-mean Gaussian process noise with assumed known covariance *Q*. Measurements are a linear combination of the system state variables and corrupted by uncorrelated noise. Thus, the *m* dimensional measurement vector is modelled as

$$Z(k) = HX(k) + w(k) \tag{2}$$

where *H* is the *mxk* measurement matrix and *w(k)* is zero mean, white Gaussian measurement noise with covariance *R*. Note that, it is assumed that *V(k)* and *w(k)* are mutually uncorrelated. A good derivation of the Kalman filter equations can be found in [5], here only the resulting standard Kalman filter equations [4] are given.

$$\tilde{X}(k+1|k)=F\hat{X}(k|k) \tag{3}$$

$$v(k+1)=Z(k+1)-H\tilde{X}(k+1|k) \tag{4}$$

$$P(k+1|k)=FP(k|k)F'+\Gamma Q\Gamma' \tag{5}$$

$$S(k+1)=HP(k+1|k)H'+R \tag{6}$$

$$W(k+1)=P(k+1|k)H'S(k+1)^{-1} \tag{7}$$

$$\hat{X}(k+1|k+1)=\tilde{X}(k+1|k)+W(k+1)v(k+1) \tag{8}$$

$$P(k+1|k+1)=P(k+1|k)-W(k+1)S(k+1)W(k+1)' \tag{9}$$

In the Eqs (3) through (9) $\tilde{X}(k+1|k)$ and *P(k+1| k)* are the predicted state and state covariance and $\hat{X}(k+1|k+1)$ and *P(k+1| k+1)* are the updated state and state covariance respectively. *S(k+1)*, *R*, *W(k+1)* and *ν(k+1)* are the covariance of the innovation, the covariance of the measurement noise, the Kalman gain and the innovation respectively.

3.2 Interacting Multiple Model (IMM) Algorithm

The Interacting Multiple Model (IMM) algorithm [4] uses a bank of filters but unlike other multiple model algorithms, it keeps the number of hypotheses fixed, which reduces the computational burden. The algorithm employs a fixed number of models that interact through state mixing to track a manoeuvring target. Every filter employed in the algorithm corresponds to a possible target motion to cover the actual modes of the target. The initial estimate at the beginning of each cycle for each filter is a mixture of all the most recent estimates from the filters. It is the mixing that enables the algorithm to effectively take the history of the modes into account. The probability of each model being true is found by using a likelihood function for the model and switching between the models is governed by a transition probability matrix. The state estimates from the sub-filters are then mixed, by means of weighting coefficients, in order to get the combined state estimate. The operation of the algorithm is completed in 4 steps;

1) Interaction of the Estimates:
Mixing the state estimates for the jth model is done by using the outputs of each model
$\hat{X}_i(k|k)$, model probability $w_i(k)$ and the transition probabilities p_{ij}.

The final interactive estimation is given by

$$\hat{X}_j^0(k\,|\,k)=\sum_{i=1}^{N}\frac{p_{ij}w_i(k)}{\sum_{t=1}^{N}p_{tj}w_t(k)}\hat{X}_i(k\,|\,k) \tag{10}$$

The corresponding error covariance for the same model is

$$P_j^0(k\,|\,k)=\sum_{i=1}^{N}\{\frac{p_{ij}w_i(k)}{\sum_{t=1}^{N}p_{tj}w_t(k)}x \tag{11}$$
$$[P_i(k|\,k)+(\hat{X}_i(k\,|\,k)-\hat{X}_i^0(k\,|\,k))(\hat{X}_i(k\,|\,k)-\hat{X}_i^0(k\,|\,k))']\}$$

where '*k*' is the scan number and $P_i(k/k)$ is the estimate covariance for the i$^{\text{th}}$ model at scan '*k*'.

2) Filtering
The standard Kalman filter equations, i.e., Eqs 3-9, are used to update the filters.

3) Mode Probability Calculation

The likelihood of the j$^{\text{th}}$ model is calculated using the innovation v_j and the corresponding covariance of innovation S_j

$$p(Z(k)\,|\,M_j(k),Z^{k-1})=\frac{1}{\sqrt{2\pi|S_j(k)|}}\exp(-\frac{1}{2}v_j'(k)S_j^{-1}v_j(k)) \tag{12}$$

Then the model probabilities (weighting coefficients) $w_j(k)$ are given by

$$w_j(k)=\frac{P(Z(k)\,|\,M_j(k),Z^{k-1})\sum_{i=1}^{N}p_{ij}w_i(k-1)}{P(Z(k)\,|\,Z^{k-1})} \tag{13}$$

where $P(Z(k)\,|\,Z^{k-1})$ is the normalisation constant which ensures that the sum of the sub-model probabilities is equal to one.

4) Estimate Combination
The final estimate is a weighted sum of the estimates from all of the filters, that is

$$\hat{X}(k \mid k) = \sum_{j=1}^{N} w_j(k)\hat{X}_j(k \mid k)$$

and the corresponding estimate of covariance is

$$P(k \mid k) = \sum_{j=1}^{N} \{w_j(k)[P_j(k) + (\hat{X}_i(k \mid k) - \hat{X}_i^0(k \mid k))(\hat{X}_i(k \mid k) - \hat{X}_i^0(k \mid k))']\} \tag{14}$$

3.3 Data Association

Kalman filter, when used in tracking, assumes that there is only one measurement with unity probability of detection in every scan and the measurement is originated from target. In real applications, these assumptions are not valid. It is possible to have more than one measurement at any time since a measurement may have originated from either target, clutter or some other source. Moreover, target detection every scan may not be possible due to sensor limitations which makes the probability of detection less than unity. Tracking a target under measurement origin uncertainty problem requires a decision to determine which measurement is originated from the target. Nearest Neighbour (NN) Data Association [2] and Probabilistic Data Association (PDA) [2,6] methods are most widely used decision rules.

3.3.1 Nearest Neighbour Data Association

This is a non-Bayesian method. In NN, decision of the target originated measurement is made in two steps. In the first step, a validation gate [2] is set up to determine possible measurements that are related to the target of interest. In the second step, the nearest measurement among the validated measurements to the predicted measurement is selected according to distance measure which is given below.

$$D(Z) = [Z - z(k+1 \mid k)]' S(k+1)^{-1} [Z - z(k+1 \mid k)] \tag{15}$$

where,. Z is the measurement under consideration, $z(k+1|k)$ and $S(k+1)$ are predicted measurement and innovation covariance respectively.

3.3.2 Probabilistic Data Association

This method takes all the measurements which lie in the validation gate into

3.3.2 Probabilistic Data Association

This method takes all the measurements which lie in the validation gate into account. It computes the association probabilities for each validated measurement at current time to the target of interest. Detailed extraction of the association probabilities is given in [2], here only resulting equations are given for the parametric and non-parametric PDA algorithms.

1) Parametric PDA

$$\beta_i(k) = \begin{cases} \dfrac{e_i}{b + \sum_{j=1}^{m(k)} e_j} & i = 1, \ldots, m(k) \\ \dfrac{b}{b + \sum_{j=1}^{m(k)} e_j} & i = 0 \end{cases} \tag{16}$$

$$e_i = e^{-\frac{1}{2} v_i(k)' S(k+1)^{-1} v_i} \tag{17}$$

$$b \cong \lambda \left| 2\pi S(k+1) \right|^{1/2} \frac{1 - P_D P_G}{P_D} \tag{18}$$

Eq 18 can be re-written as

$$b = (\frac{2\pi}{\gamma})^{n_z/2} \lambda V(k) c_{n_z}{}^{-1} \frac{1 - P_D P_G}{P_D} \tag{19}$$

In the above equations P_D and P_G are the probability of detection and gating probability respectively. *S(k)*, λ, c_{n_z}, γ, $m(k)$, $V(k)$ are innovation covariance, spatial density of the false measurements, volume of the unit hyper sphere, gating threshold, number of measurements in the validation gate and volume of the validation gate respectively.

2) Non-Parametric PDA

The non-parametric PDA is the same as above except for replacing $\lambda V(k)$ in Eq 19 by *m(k)* which requires knowledge of λ.

3.4 Nearest Neighbour Kalman Filter

The Nearest Neighbour Kalman filter [2] is simply a Kalman filter which is used in conjunction with the NN approach in order to take the measurement origin uncertainty into account. Integration of the Kalman filter and NN data association method is the simplest way of adapting Kalman filter to track a target under measurement origin uncertainty. NN algorithm needs to know the predicted measurement and innovation covariance which are computed by the Kalman filter. Thus, the data association unit using NN algorithm must be integrated after the computation of innovation covariance given by Eq 6. After the integration, the new filter computes state estimation and its covariance in three steps as follows: *i)* Compute the predicted measurement and innovation covariance in the Kalman filter manner. *ii)* Set up a validation gate and determine the measurements which are in the gate. *iii)* Select the nearest validated measurement to the predicted state using NN algorithm then update target state with this measurement, i.e., in the Kalman filter manner.

3.5.IMM-NN Filter

The IMM algorithm improves tracking performance when the target manoeuvres. Thus, it is logical to employ the NN approach together with the IMM algorithm when tracking manoeuvring targets in clutter In the IMM-NN algorithm, it is essential that each filter has to update the target state using the same validated measurements, i.e., the data association unit has to set up a validation gate which is equal to the union of the mode-conditioned gates. In practice, the largest of the validations gates is approximately equal to the union and thus can be used instead. Consequently, determining the validated measurements portion of the data association procedure is performed by the filter which has the biggest validation gate in the IMM structure. This filter will be called as centre filter of the IMM-NN algorithm. The equations given for the IMM filter and the NN approach are valid for the IMM-NN algorithm.

3.6 Probabilistic Data Association Filter (PDAF)

The (PDAF) is simply a Kalman filter which is used in conjunction with the PDA approach in order to take the measurement origin uncertainty into account. The filter assumes that the target is detected (perceived) and its track has been initialized. At each sampling interval a validation gate is set up and the validated measurements are determined. The measurement originating from the target of interest can be among the possible several validated measurements; hence, the track update is done by taking the weighted sum of all observations within the gated region. The weights are the association probabilities given by Eq 16 computed by the PDA algorithm. Since, and modified state update equations are given as follows:

$$\hat{X}(k+1\,|\,k+1) = \tilde{X}(k+1\,|\,k) + W(k+1)v(k+1) \tag{20}$$

where

$$v(k+1) = \sum_{i=1}^{m_k} \beta_i(k+1)v_i(k+1) \tag{21}$$

is the combined innovation. In Eq 21, *m_k*, *$\beta_i(k+1)$* and *$v_i(k+1)$* are the number of the validated measurements, ith association probability and the innovation term of the ith measurement respectively. The covariance of the updated state is

$$\begin{aligned} P(k+1\,|\,k+1) &= \beta_0(k+1)P(k+1\,|\,k) \\ &+[1-\beta_0(k+1)]P^c(k\,|\,k)+\tilde{P}(k+1) \end{aligned} \tag{22}$$

where the covariance of the state updated with the correct measurement is

$$P^c(k+1\,|\,k+1) = P(k+1\,|\,k) - W(k+1)S(k+1)W(k+1)' \tag{23}$$

and the spread of innovation term is

$$\begin{aligned} \tilde{P}(k+1) &= W(k+1)\{[\sum_{i=1}^{m_k} \beta_i(k+1)v_i(k+1)v_i(k+1)'] \\ &-v(k+1)v(k+1)'\}W(k+1)' \end{aligned} \tag{24}$$

3.7 IMM-PDA Filter

The Interacting Multiple Model Probabilistic Data Association Filter (IMMPDAF) [2] employs PDAF as a state estimator in IMM structure. As in IMM-NN filter, all the filters have to use same validated measurements. Thus, the filter which has the largest validation gate must be chosen as centre filter. Using PDAF in IMM structure necessitates a modification in IMM likelihood function calculation which is then utilised to compute the mode probabilities. Reason of the modification rises from the nature of the PDA algorithm's assumption that an observation is originated from either target or other sources (clutter, false alarms, etc.) with a probability which is never unity or zero. IMMPDAF takes the uncertainty about the source of the observations into account while computing the likelihood function corresponding to filter matched to mode (model) j with m the number of validated measurements as follows:

$$\Lambda_j(Z(k+1)) = \frac{1-P_D^j}{V^m} + \frac{P_D^j}{mV^{m-1}} \sum_{i=1}^{m} N[Z_i(k+1); Z^j(k+1\,|\,k), S^j(k+1)] \tag{25}$$

where P_D is the target detection probability and, $Z^i(k+1|k)$ and $S^i(k+1)$ are the predicted measurement and innovation covariance for the model j respectively.

4. SCENARIOS

Three different target scenarios are considered for this study with accelerations ranging from mild (1.5g) to moderate (4g) where all three targets are assumed to appear both at short range (15 km) and at mid range (40 km) from the radar, so that both the effects of target manoeuvrability and target range could be investigated. The simulated target motions were generated in 3 dimensions, i.e. *X-Y-Z* plane, with sampling interval of 0.1 seconds. In the first scenario, the target begins its motion from the (-15,5,5)km. point in the 3D Cartesian space with a velocity of 300m/sec. in *X* axis. The target moves on a straight line for the first 100 seconds and then proceeds with a right at 3 deg/sec for a 60 seconds rate and there is another straight line path for 100 seconds. In the last part of the motion, target turns right at 3 deg/sec turn rate for 60 seconds. The second target starts its motion from (15,10,5) km initial position with velocity of -200m/sec. in *X* axis. For 40 seconds it moves on a straight line path then turns for 15 seconds at 6 deg/sec turn rate then the target proceeds with another straight line motion for 10 seconds and before ending its motion with a turn for 20 seconds with 1 deg/sec turn rate. In the third scenario, target motion starts at (-4.6,-1.9,2.8)km. with initial velocity of (143,388,8)m/sec. in 3D Cartesian space. The motion consists of two consecutive turns at 6deg/sec and -6dec/sec turn rates. Each of the turns lasts 30 seconds.

The scenarios explained above are the short range scenarios. Mid range scenarios have been generated by moving the initial positions of the target motions generated for the short range scenarios. In mid rage scenarios starting points have been chosen to be (-25,40,7)km.,(40,25,8)km. and (-40,60,10)km. for the first, second and third scenarios respectively. Moreover, in order to create a more realistic simulation environment for comparing the algorithms' performance, targets are assumed to be moving in a cluttered environment where the number of clutter is assumed to have Poisson distribution and they are uniformly distributed in the surveillance region [2,6]. The Poisson parameter λ that describes the clutter density was chosen a 10^{-6} which corresponds to approximately 40 clutter in the chosen surveillance volume.

5. FILTERS DESIGN PARAMETERS

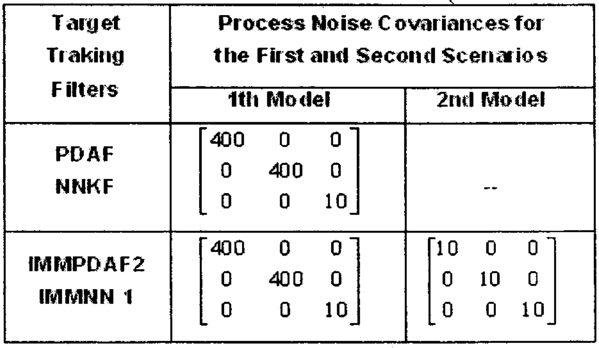
Seven different configurations of the tracking algorithms, namely PDAF, IMMPDAF, NNKF and IMMNN, have been design to be tested on the selected scenarios. The configurations belonging to the same algorithm differ either in the number/type of models they posses or the assumption of a-priori knowledge regarding the target manoeuvre. The filters using a-priori knowledge about target manoeuvre comprise second order dynamic models. These filters are PDAF, NNKF, IMMPDAF2 and IMMNN1. The IMM algorithm based filters, i.e. IMMPDAF2 and IMMNN1, have 2 models and one of the models employs large process noise covariance to track manoeuvres where the second one utilises small process noise covariance to track benign motion. Process noise covariance of the PDAF and NNKF approaches has been selected sufficient enough to track the maximum acceleration change. Tables 1 and 2 present process noise covariance values selected for the first two scenarios whereas process noise covariance values used for the third scenario where the target performs a bigger manoeuvre are shown in Table 3. Other IMM based filters, namely IMMPDAF1, IMMPDAf3, IMMNN2, have been designed to track a target without a-priori knowledge regarding the target manoeuvre. IMMPDAF1 algorithm includes two second order models (a coordinated turn model in XY plane and a linear model). Three model IMM structure has been used in IMMPDAF3 and IMMNN2 filters. Dynamic models employed in three model IMM structures are;

Model 1: Third order dynamic model with large process noise covariance for tracking onset or end of acceleration.

Model 2: Third order dynamic model with small process noise covariance for tracking nearly constant acceleration motion.

Model 3: Second order dynamic model with small process noise covariance for tracking nearly constant velocity motion.

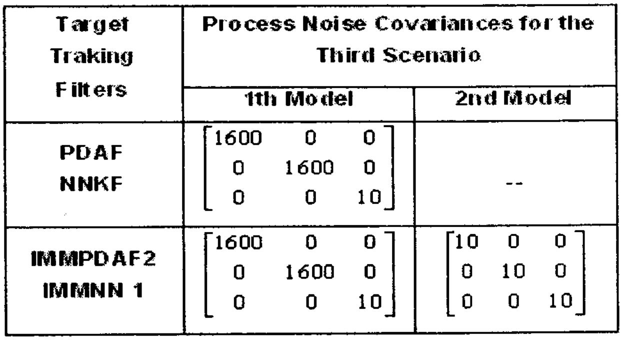
Table 1 : Process noise covariance values (mild manoeuvre)



Target Traking Filters	Process Noise Covariances for the First and Second Scenarios	
	1th Model	2nd Model
PDAF NNKF	$\begin{bmatrix} 400 & 0 & 0 \\ 0 & 400 & 0 \\ 0 & 0 & 10 \end{bmatrix}$	--
IMMPDAF2 IMMNN 1	$\begin{bmatrix} 400 & 0 & 0 \\ 0 & 400 & 0 \\ 0 & 0 & 10 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$

In Table 4, mode transition probabilities and initial model probabilities are given for the IMM based filters. The validation gate threshold has been taken as 11.4 which corresponds to 0.99 gating probability, also the probability of detection used in the simulations is 0.85.

Table 2 : Process noise covariance values (moderate manoeuvre)



Target Traking Filters	Process Noise Covariances for the Third Scenario	
	1th Model	2nd Model
PDAF NNKF	$\begin{bmatrix} 1600 & 0 & 0 \\ 0 & 1600 & 0 \\ 0 & 0 & 10 \end{bmatrix}$	--
IMMPDAF2 IMMNN 1	$\begin{bmatrix} 1600 & 0 & 0 \\ 0 & 1600 & 0 \\ 0 & 0 & 10 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$

Table 3 : Process noise covariance values (no a-priori knowledge)

Target Traking Filters	Process Noise Covariances		
	1. Model	2. Model	3. Model
IMMPDAF 1	$\begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0,1 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0,1 \end{bmatrix}$	--
IMMPDAF 3 IMMNN 2	$\begin{bmatrix} 400 & 0 & 0 \\ 0 & 400 & 0 \\ 0 & 0 & 100 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$	$\begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$

Table 4 : Mode transition and model probabilities.

Traking Filters	Mode Transition Probabilities	Initial Model Probabilities
IMMPDAF 1 IMMPDAF 2 IMMNN 1	$\begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}$	$\mu = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}$
IMMPDAF 3 IMMNN 2	$\begin{bmatrix} 0.90 & 0.05 & 0.05 \\ 0.15 & 0.85 & 0 \\ 0.15 & 0 & 0.85 \end{bmatrix}$	$\mu = \begin{bmatrix} 0.4 & 0.3 & 0.3 \end{bmatrix}$

6. SIMULATION RESULTS

Each of the filters given in previous section was tested on the generated scenarios. Performance of the filters was examined in terms of percentage track loss (Ptl) and rms estimation error in spherical coordinates (i.e., range, azimuth and elevation) through 100 Monte Carlo simulations. The simulation results for the short and mid range scenarios have been given in Table 5 and 6 respectively.

Table 5 : Results for short range scenarios.

		PDAF	IMMPDAF 1	IMMPDAF 2	IMMPDAF 3	NNKF	IMMNN 1	IMMNN2
Scenario I	rms R	22,3090m	21,9562m	21,9075m	18,7592m	22,3169m	21,5734m	20,5603m
	rms A	0,0485°	0,0528°	0,0526°	0,0487°	0,0498°	0,0632°	0,0665°
	rms E	0,0414°	0,0400°	0,0409°	0,0454°	0,0488°	0,0496°	0,0622°
	Ptl	%1	%9	%0	%0	%0	%5	%37
Scenario II	rms R	17,4602m	18,9200m	20,8751m	19,2695m	17,8514m	16,6231m	16,6032m
	rms A	0,0199°	0,0745°	0,0286°	0,0684°	0,0188°	0,0225°	0,0238°
	rms E	0,0192°	0,0480°	0,0245°	0,0414°	0,0208°	0,0206°	0,0230°
	Ptl	%0	%34	%0	%0	%0	%0	%4
Scenario III	rms R	20,6337m	23,9201m	23,1289m	29,3243m	19,4024m	21,4716m	24,0374m
	rms A	0,0377°	0,1226°	0,0612°	0,1432°	0,0085°	0,0198°	0,0682°
	rms E	0,0175°	0,0402°	0,0223°	0,0743°	0,0091°	0,0108°	0,0128°
	Ptl	%0	%84	%2	%3	%0	%0	%11

Table 6 : Results for mid range scenarios.

		PDAF	IMMPDAF1	IMMPDAF2	IMMPDAF3	NNKF	IMMNN1	IMMNN2
Scenario I	rms R	65,0882m	79,5411m	81,3030m	40,0773m	56,2954m	36,1484m	m
	rms A	0,1814°	0,1661°	0,1587°	0,1739°	0,2070°	0,1322°	°
	rms E	0,1236°	0,1195°	0,1091°	0,1574°	0,1075°	0,0869°	°
Scenario II	Ptl	%94	%94	%93	%94	%99	%99	%100
	rms R	34,4729m	77,7512m	37,1420m	36,4020m	50,3717m	57,4427m	m
	rms A	0,1597°	0,1885°	0,1752°	0,2074°	0,2582°	0,2507°	°
	rms E	0,0914°	0,1091°	0,1001°	0,1523°	0,1190°	0,1487°	°
	Ptl	%25	%60	%22	%31	%63	%67	%100
Scenario III	rms R	235,6407m	251,9296m	271,7560m	172,8245m	155,2333m	202,7313m	238,9803m
	rms A	0,3491°	0,3325°	0,3652°	0,3654°	0,2742°	0,5002°	1,1278°
	rms E	0,1273°	0,1215°	0,1550°	0,2764°	0,1588°	0,1691°	9,1667°
	Ptl	%38	%86	%34	%45	%81	%65	%93

Simulation results have revealed that:

- Regardless of the target range all algorithms, except for IMMNN2, perform well in terms of measurement errors i.e., below a specified error limit of 0.2°. However, it should be noted that a-priori knowledge of target manoeuvre is usually not possible in real life applications.

- Target range has a big effect both on the number of track loss and estimation errors, the farther the target from the radar the larger the estimation errors and % track loss. However, % track loss and estimation error criteria should be assessed separately. Despite the high number of track loss at farther ranges all algorithms have managed to keep estimation errors below the specified level as estimation errors were only calculated for un-dropped tracks. Further investigations into the number of track loss has shown that most of the losses occurred at certain times during simulations which correspond to minimum RCS, thus, low SNR, indicating that one needs a certain level of SNR in order to be able to keep the target in track.

- In order to reduce track loss declarations and increase the probability of detection at least 18dB SNR is needed. This result is consistent with approximately 20dB level given in [1].

- IMMNN approach has not produced a stable performance. The filter tends to diverge following relatively large estimation errors; due to its weighted structure IMMPDA approach is more robust in such cases. Thus it is recommended that IMM and NN approaches not be employed together.

- Three dimensional nonlinear target models (i.e., a coordinated turn model in *XY* plane and a 2nd degree model in *Z* used in IMMPDAF1) cause filter divergence, thus, degraded tracking performance and they are not recommended.

- The three model IMMPDA approach with no a-priori knowledge regarding the target motion has compared well with algorithms that, unrealistically, assumed knowledge of target manoeuvre and proved to be a successful tracking algorithm for ground based radar both at short and mid ranges.

7. CONCLUSIONS

Performance of most commonly used tracking algorithms has been compared for ground based radar. Algorithms have been compared in terms of track loss and rms estimation errors. The 3 model IMMPDAF algorithm with no a-priori knowledge of the target motion has compared favourably with all the tested algorithms even with the ones that assume knowledge of the target manoeuvre.

ÖZET

Bu makalede kara konuşlu bir radar için çeşitli hedef takibi algorimalarının performansları karşılaştırılmıştır. Algoritma performans ölçüsü olarak yüzde iz kaybı ve ortalama karekök hata esas alınmıştır. Algoritmaların performansı manevra seviyesi düşükten orta seviyeye değişen hedef senaryoları üzerinde incelenmiştir.

REFERENCES

[1] M. I. Skolnik. “Radar Handbook”, McGraw-Hill, Inc .

[2] Bar-Shalom Y., Li X. R., ”Mulltitarget–Multisensor Tracking: Principles and Techniques.”, YBS Publishing, 1995.

[3] Kalman R. E., ”A New Approach to Linear Filtering and Prediction Problems.”, Journal of Basic Engineering, Vol 82, No 1, pp 35 – 46, 1960.

[4] Y. Bar Shalom, X. R. Li, T. Kirubarajan0, “Estimation with Applications to Tracking and Navigation”, Wiley,2002.

[5] Gelb, A.,” Applied Optimal Estimation”, MIT Press, 1974.

[6] S. S. Blackman., “Design and Analysis of Modern Tracking Systems”, Artech House, Norwood, MA, 1999.