Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 9 (2): 201-213, 2009
ISSN 1812-5654

Fast Adaptive Update Rate for Tracking a Manoeuvring Target with a Phased Array Radar, Using IMM and MRIMM Algorithms

[1,2]H. Benoudnine, [2]M. Keche, [2]A. Ouamri and [3]M.S. Woolfson
[1]Faculté des Sciences et Sciences de l'Ingénieur, Abdelhamid Ibn Badis University, 227 rue Belhacel, 2700, Mostaganem, Alegria
[2]Laboratoire Signaux et Images, Département d'Electronique, Faculté de Génie Electrique, Université des Sciences et de la Technologie Mohamed Boudiaf, B.P 1505, El Mnouar, Oran, Algeria
[3]Department of Electrical and Electronic Eng., Faculty of Engineering, University of Nottingham, UK

Abstract: In this study, a new fast method for selecting the next update time in two maneuvering target tracking algorithms, namely the Interacting Multiple Models (IMM) algorithm and the Multi Rate Interacting Multiple Models (MRIMM), will be presented. Both IMM and MRIMM are used here to predict and estimate the target's possible states and to select the correct next update time. The idea is to assign to each model in the IMM and MRIMM algorithms an appropriate rate and to weight these rates by the models' probabilities to obtain the rate to use. The resulting algorithms are named, respectively, the Fast Adaptive IMM (FAIMM) algorithm and the Adaptive MRIMM (AMRIMM) algorithm. Using Monte Carlo simulations, the performances of these algorithms are compared to that of the Adaptive IMM algorithm that uses Van Keuk criterion to select the next update time and to that of the IMM algorithm and MRIMM that use a constant update time.

Key words: Variable update time, maneuver, tracking, multirate, discrete wavelet

INTRODUCTION

Both air traffic control and military applications have strict requirements on tracking algorithms. They should be accurate, survivable and fault tolerant and should require low computation and communication complexity. A lot of research work has been devoted to investigate tracking algorithms that fulfill the requirements of modern surveillance. Many works have proposed the use of variable update rate trackers to reduce the computation processing. However, most of them use single model filters and decisions oriented techniques and cannot, therefore, cope well with all possible target behaviors. For a better tracking accuracy, it is necessary to recourse to multiple models algorithms. Many of these algorithms have been developed, among which the Interacting Multiple Models (IMM) (Bar Shalom, 1995) is widely used. The main feature of this algorithm is its ability to estimate the state of a dynamic system, which can switch from one behavior mode to another.

For a real time implementation, data processing is an important issue. Many researchers have focused on reducing the computation time. A moving bank multiple models adaptive algorithm has been proposed by Mayback and Hentz (1987). Li and Bar-Shalom (1996) introduced the idea of multiple models sets and presented a variable structure multiple model algorithms. Lin and Atherton (1993) presented a selected filter IMM (SFIMM) algorithm in which models close to target acceleration are selected. Hong (1999), Hong and Dig (2000) and Hong *et al.* (1998) presented a Multirate Multiple Model (MRIMM) algorithm that uses a multi-resolution filtering approach, where each model works with a rate appropriate to a motion of the target.

In fact, Time is the most precious resource for real time application, especially when using multi function radar, such as a Phased Array Radar, whose occupancy by the tracking task should be kept as low as possible.

The Phased Array Radar has the ability to direct the antenna beam in any direction without moving mechanically the radar. It performs surveillance in a number of regions, to identify new targets; it then maintains tracks on the targets by directing the beam to the predicted target position, for each updated track. Each track may be updated at a variable rate.

Many approaches have been proposed for selecting the next update time. Some approaches (Efe and Atherton, 1998; Kirubarujan *et al*., 1995; Daeipour *et al*., 1994; Efe and Atherton, 1996) select the next update time from a set of a predefined update times and assign one

Corresponding Author: Miss Hadjira Benoudnine, Laboratoire Signaux et Images, Département d'Electronique, Faculté de Génie Electrique, Université des Sciences et de la Technologie Mohamed Boudiaf, B.P 1505, El Mnouar, Oran, 3100, Algeria Tel/Fax: 0021341560357

of these to each track, so that the track is maintained over a target trajectory. Usually, a high update rate is chosen for a manoeuvring motion and a lower update rate is chosen for a non manoeuvring motion. Another approach (Sarunic and Evan, 1997) uses an iterative selection of the next update time, whereas Van Keuk (1977, 1979), Watson (1994) and Shin (1995) an empirical form is used. In Ahmeda *et al*. (1997), an exact formula for computing the next update time in the JPDAF is derived, using on the Van Keuk criterion. This formula has been generalised to the IMMJPDAF in Benoudnine *et al*. (1999a, b).

In Benoudnine *et al*. (2006), a Fast AIMM algorithm (FAIMM algorithm), which computes the next update time by using the probabilities of action of models is presented. A considerable amount of computational time is saved, by this algorithm, in comparison with the AIMM that uses the Van Keuk criterion. In Benoudnine *et al*. (2007), the incorporation of a variable update time into the MRIMM improved the performance of this algorithm.

This study describes in more details the work presented in Benoudnine *et al*. (2006, 2007). It's also presents new simulation reseults.

TARGET AND SYSTEM MODELS

The problem addressed in this paper is the estimation of the state (position, velocity and acceleration) of a target moving in a plane. The motion of the target is assumed to obey several possible models. The discrete state equation for such a target is:

$$x^j(k+1) = F^j(k)\, x^j(k) + w^j(k),\ j = 1,...,r \tag{1}$$

where, $x^j(k)$ is the state vector of the target, $F^j(k)$ is the transition matrix and $w^j(k)$ is the process noise, assumed to be a zero mean Gaussian process with a known covariance Q^j. All these quantities are at time k and for model j.

The measurement equation is given by:

$$z(k) = H^j(k) x^j(k) + v(k) \tag{2}$$

where, z(k) is the (m×1) measurement vector at time k, due to the return from the target, H^j is the (m×n) measurement matrix for model j and v is the measurement noise vector, with zero mean and known covariance R.

DESCRIPTION OF TRACKING ALGORITHMS

IMM algorithm: The Interacting Multiple Models (IMM) algorithm has been proposed for tracking a manoeuvring target (Bar Shalom, 1995). It is a sub-optimal well documented method for solving the problem of state estimation, in the case of a manoeuvring target. In this algorithm, several filters are run in parallel, each filter being matched to an assumed model for the target's motion. The jumps between models are assumed to be governed by a Markov Chain. The overall estimated state is formed by summing the estimates from different filters, weighed by the probabilities of models. The IMM consists of the following steps:

Step 1: Mixing of state estimates from the previous time: For each target, starting with the state estimates $\hat{x}^j(k-1|k-1)$, matched to the models $M_j(k)$, their covariances $\hat{P}^j(k-1|k-1)$ and the model probabilities $\mu_{i/j}(k-1|k-1)$, the mixed state estimate $\hat{x}^{0j}(k-1|k-1)$ and its covariance $\hat{P}^{0j}(k-1|k-1)$ are computed according to:

$$\hat{x}^{0j}(k-1|k-1) = \sum_{i=1}^{r} \hat{x}^i(k-1|k-1)\mu_{i/j}(k-1|k-1),\ \ j = 1,\ldots,r \tag{3}$$

and

$$\hat{P}^{0j}(k-1|k-1) = \sum_{i=1}^{r} \mu_{i/j}(k-1|k-1)\left[\hat{P}^i(k-1|k-1) + \hat{P}_S^{ij}(k-1|k-1)\right] \tag{4}$$

$$\hat{P}_S^{ij}(k-1|k-1) = \left[\hat{x}^i(k-1|k-1) - \hat{x}^{0j}(k-1|k-1)\right]\left[\hat{x}^i(k-1|k-1) - \hat{x}^{0j}(k-1|k-1)\right]^T,\ i, j = 1,...r$$

where, r denotes the number of interacted models and $\mu_{i/j}$ (k-1|k-1) is the probability that model M_i was in effect at time (k-1) given that M_j is in effect at time k-1, conditioned on z^{k-1}, the set of measurements up to k-1:

$$\mu_{i/j}(k-1|k-1) = \frac{1}{\bar{c}_j} p_{ij}\mu_i(k-1),\ i, j = 1,2...r \tag{5}$$

In the above equation, p_{ij} is the a prior probability of transition from model i to model j, μ_i(k-1) is the probability that model i is in effect at time k-1 and $\bar{c}_j$ are normalising constants:

$$\bar{c}_j = \sum_{i=1}^{r} p_{ij}\mu_i(k-1),\ j = 1,...,r \tag{6}$$

Step 2: Kalman filtering: Based on the initial states estimates Eq. 3, their covariances Eq. 4 and the received measurements, the state estimate $\hat{x}^j(k|k)$ and its covariance $\hat{P}^j(k|k)$ are updated using the jth Kalman Filter in the IMM algorithm.

Step 3: Computation of the likelihood function: The likelihood $\Lambda^j(k)$ of model $M_j(k)$ is updated from:

$$\Lambda^j(k) = P\left[z(k)|M_j(k), Z^{k-1}\right] \tag{7}$$

Step 4: Update of the models' probabilities: The probability that model $M_j(k)$ is in effect at time k is computed from:

$$\begin{aligned}\mu_j(k) &= P\left[M_j(k)\middle|Z^k\right],\ j=1,\cdots,r \\ &= \frac{1}{c}\Lambda^j(k)\bar{c}_j\end{aligned} \tag{8}$$

where, $\bar{c}_j$ is defined in Eq. 6 and c is the normalisation constant for $\mu_j(k)$ given by:

$$c = \sum_{j=1}^{r}\Lambda^j(k)\bar{c}_j \tag{9}$$

Step 5: Combination of the model conditioned estimates: For each target, the overall state estimate $\hat{x}(k|k)$ and its corresponding error covariance $\hat{P}(k|k)$ are updated as follows:

$$\hat{x}(k|k) = \sum_{j=1}^{r}\hat{x}^j(k|k)\mu_j(k) \tag{10}$$

$$\hat{P}(k|k) = \sum_{j=1}^{r}\mu_j(k)\left(\hat{P}^j(k|k) + \hat{P}_s^j(k|k)\right) \tag{11}$$

where, $\hat{P}_s^j(k|k) = \left[\hat{x}^j(k|k) - \hat{x}(k|k)\right]\left[\hat{x}^j(k|k) - \hat{x}(k|k)\right]^T$

MRIMM algorithm: The Multirate IMM (MRIMM) algorithm is derived by Hong *et al*. (1998), at different multi-resolution levels, by using a discrete wavelet transform, a hierarchy of models and a data structure. The tracking algorithm is applied to this hierarchical structure and final tracking outputs can be obtained at specified levels (rates). In Fig. 1 a diagram of one cycle of a two models MRIMM algorithm is presented. In the algorithm considered in this work, the first model is a constant acceleration (CA) model. It operates at full rate (level 0) and is suitable for a manoeuvring motion. The second model, named half rate Multirate Constant Velocity (MRCV) model, was derived by Hong (1999) and Hong *et al*. (1998) from a geometrical interpretation of a discrete wavelet transform. It operates at half rate (level 1) and is appropriate for a non-manoeuvring motion. The superscript $()^m$ is used to denote a quantity related to the manoeuvring model (CA) and $()^n$ a quantity related to the non manoeuvring model (MRCV). The measurements and state for the non manoeuvring model are processed at level 1, whereas for the CA model the state and the measurements are processed at level 0. The results from the two models are mixed and combined at the same level (level 0 is used here). Transformation details of

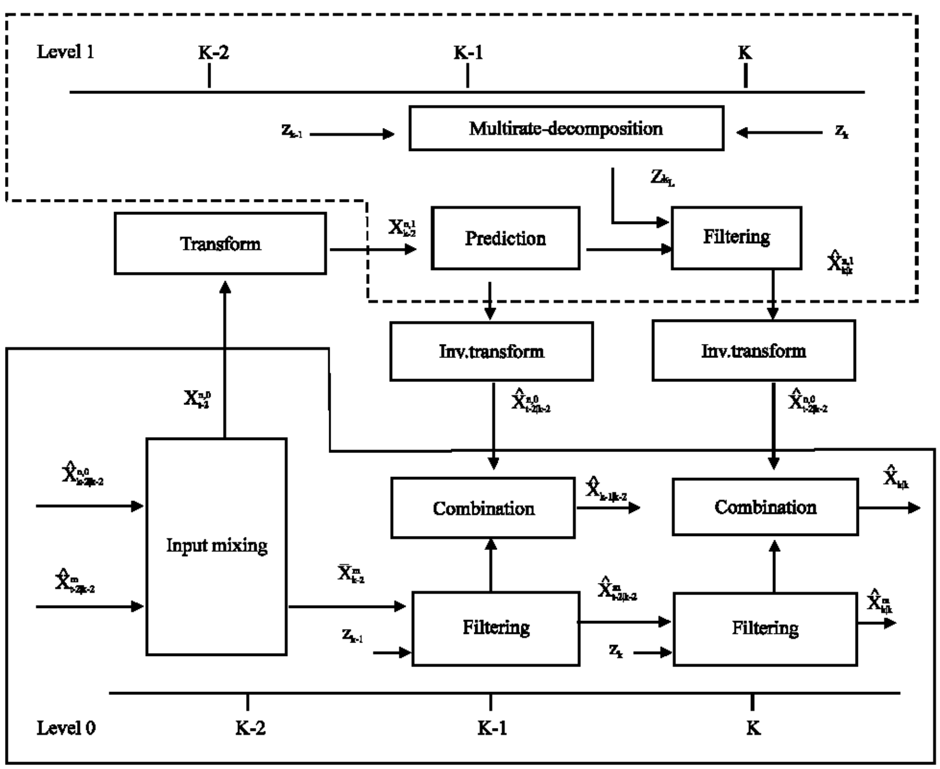

Fig. 1: A Two- model, two- level MRIMM algorithm

non manoeuvring quantities between level 0 and level 1 are given in Hong (1999), Hong *et al.* (1998) and (Hong and Ding, 2000). The main steps of the MRIMM starting at t_{k-2} and ending at t_k consist of the following.

Step 1: Mixing of state estimates from the previous time: Assuming that the state estimates and their covariances matched to each model at level 0

$$\left\{\hat{x}^{n,0}_{k-2|k-2}, \hat{P}^{n,0}_{k-2|k-2}\right\} \text{ and } \left\{\hat{x}^{m,0}_{k-2|k-2}, \hat{P}^{m,0}_{k-2|k-2}\right\}$$

are available at time t_{k-2}, the multirate models probabilities and the multirate states are derived as follow:

$$\bar{\mu}^{n}_{k-2} = P^{n,n}_{k|k-2}\mu^{n}_{k-2} + P^{n,m}_{k|k-2}\mu^{m}_{k-2} \tag{12}$$

$$\bar{\mu}^{m}_{k-2} = P^{m,n}_{k-1|k-2}\mu^{n}_{k-2} + P^{m,m}_{k-1|k-2}\mu^{m}_{k-2} \tag{13}$$

where, $\bar{\mu}^{n}_{k-2}$ and $\bar{\mu}^{m}_{k-2}$ are the non manoeuvring and the manoeuvring mixing Multirate model probability at time t^{+}_{k-2} and:

$$P^{Jump} = \begin{bmatrix} P^{n,n}_{k|k-2} & P^{n,m}_{k|k-2} \\ P^{m,n}_{k-1|k-2} & P^{m,m}_{k-1|k-2} \end{bmatrix} \tag{14}$$

is the Markovian model probabilities transitional matrix, which defines the transition between the two models.

The mixed half rate and full rate state vectors are then derived as:

$$\bar{x}^{n,0}_{k-2} = \frac{1}{\bar{\mu}^{n}_{k-2}}\left[\hat{x}^{n,0}_{k-2|k-2}P^{n,n}_{k|k-2}\mu^{n}_{k-2} + \hat{x}^{m}_{k-2|k-2}P^{n,m}_{k|k-2}\mu^{m}_{k-2}\right] \tag{15}$$

$$\bar{x}^{m}_{k-2} = \frac{1}{\bar{\mu}^{m}_{k-2}}\left[\hat{x}^{n,0}_{k-2|k-2}P^{m,n}_{k-1|k-2}\mu^{n}_{k-2} + \hat{x}^{m}_{k-2|k-2}P^{m,m}_{k-1|k-2}\mu^{m}_{k-2}\right] \tag{16}$$

The corresponding mixing multirate covariances are derived in the same way.

Step 2: Transformation of the non maneuvering quantities: The initial state estimate and its covariance for the MRCV model are transformed to level 1, to yield $\bar{x}^{n,1}_{k-2}$ and $\bar{P}^{n,1}_{k-2}$ (details transformation can be found in Hong (1998, 1999) and Hong and Ding (2000).

Step 3: Multirate propagation

- Full rate propagation from t^{+}_{k-2} to t_{k-1} for the maneuvering model is derived using the following equation:

$$\hat{x}^{m}_{k-1|k-2} = F^{m}_{k-2}\hat{x}^{m}_{k-2|k-2} \tag{17}$$

$$P^{m}_{k-1|k-2} = F^{m}_{k-2}\bar{P}^{m}_{k-2}\left(F^{m}_{k-2}\right)^{T} + Q^{m}_{k-2} \tag{18}$$

The state is composed of position, velocity and acceleration, for example in one dimensional Cartesian coordinates F^{m}_{k-2} is:

$$F^{m}_{k-2} = \begin{bmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & T \end{bmatrix}$$

- Half rate propagation from t^{+}_{k-2} to t_k for the non maneuvering model is derived using the following equation:

$$\hat{x}^{n,1}_{k|k-2} = F^{n,1}_{k-2}\bar{x}^{n,1}_{k-2|k-2} \tag{19}$$

where, $\hat{x}^{n,1}_{k|k-2}$ is the predicted state estimate at level 1 composed of low and high component of wavelet decomposition:

$$F^{n,1}_{k-2} = \begin{bmatrix} \frac{g_1^2}{g_2^2} & -2\frac{g_1}{g_2}\left(\frac{h_1}{g_2} - \frac{g_1 h_2}{g_2^2}\right) \\ 0 & \frac{g_1^2}{g_2^2} \end{bmatrix}$$

and

$$z^{n,1}_{L}(k) = h_1 z(k-1) + h_2 z(k) \tag{20}$$

$$R^{n,1}_{L}(k) = h_1^2 R(k-1) + h_2^2 R(k) \tag{21}$$

$h_1 = h_2 = g_2 = -g_1 = \sqrt{2}/2$ are, respectively, the two taps Haar low pass and high pass filters coefficients, for Wavelet decomposition.

Step 4: Propagation of the manoeuvring model: The propagation from t^{+}_{k-1} to t_k for the manoeuvring model is given by:

$$\hat{x}^{m}_{k|k-1} = F^{m}_{k-1}\hat{x}^{m}_{k-1|k-1} \tag{22}$$

$$P^{m}_{k|k-1} = F^{m}_{k-1}\bar{P}^{m}_{k-1}\left(F^{m}_{k-1}\right)^{T} + Q^{m}_{k-1} \tag{23}$$

where, $F^{m}_{k-2} = F^{m}_{k-1}$

Step 5: Updating at time t_k: A full rate update is performed on the manoeuvring model sequence, yielding $\hat{x}^m_{k|k}$ and $\hat{P}^m_{k|k}$. The half rate updates $\hat{x}^{n,1}_{k|k}$ and $\hat{P}^{n,1}_{k|k}$, are transformed to level 0 to yield the non manoeuvring states and covariance vectors $\hat{x}^{n,0}_{k|k}$ and $\hat{P}^{n,0}_{k|k}$. Also the probabilities of models are updated at time t_k yielding μ^n_k and μ^m_k . The output of MRIMM algorithm at time t_k is then:

$$\hat{x}_{k|k} = \mu^n_k \hat{x}^{n,0}_{k|k} + \mu^m_k \hat{x}^m_{k|k} \tag{24}$$

REVIEW OF SOME ADAPTIVE UPDATE TIME METHODS FOR PHASED ARRAY RADAR

Criteria for the update time choice: The next update time is calculated, according to some technical considerations:

- To maintain track in different situations (manoeuvring and non manoeuvring), the update time should be small enough so that, at the next illumination, the target will be within the predicted region, scanned by the radar beam, with a high enough probability
- The update time should be not too small, to minimise the use of the radar resources. This will allow the radar to do more within a given time

Hence, a large update time should be used for tracking a non manoeuvring target and a faster update is needed to track a manoeuvring target or fast targets, which accelerate harder or change range and which may possibly escape from the beam of the antenna.

Most of the algorithms proposed in the literature for the selection of the next update time are based on Van Keuk criterion.

Van keuk criterion: It can be observed, in any target tracking algorithm, that when the target manoeuvres, the uncertainty in the estimates increases, this is reflected by an increase in the value of the error covariance. One can exploit this observation, for selecting the sample time: Reducing it when the target manoeuvres, not to lose its track and increasing it when the target stops manoeuvring, not to waste radar resources.

This is the idea behind the Van Keuk criterion (Van Keuk, 1977, 1979), who proposed that the next update time should be selected so that the predicted error variance in position is kept under a given threshold. Based on this criterion an empirical formula for calculating the next update time T has been derived:

$$T \approx 0.4 \left[\frac{\delta_0 \sqrt{\tau_m}}{\delta_m} \right]^{0.4} \frac{v_0^{2.4}}{1 + 0.5 v_0^2} \tag{25}$$

where, v_0 is a constant, δ_0^2 is the measurement error covariance, τ_m is the manoeuvre correlation time and δ_m^2 is the covariance of target acceleration.

However, Van Keuk uses a single model to update a track. Many researchers; Watson *et al*. (1994), Shin *et al*. (1995) and Benoudnine *et al*. (1999a, b) have extended the use of Van Keuk criterion to multiple models based tracking algorithms.

Revisit time controlled using the IMM algorithm: In this algorithm (Watson, 1994), three models are used in the IMM for tracking a manoeuvring target. The first one is a Constant velocity model (CV), the second one is Exponentially Increasing Acceleration (EIA) and the third one is a three Dimensional Turning Rate (3DTR). The next update time is scheduled when the predicted error covariance in position exceeds a given threshold. The sample time, T, is computed such as:

$$\bar{P}_{k+T|k} \le \bar{P}_{th} \tag{26}$$

where, $\bar{P}_{k+T|k}$ denotes the errors covariance of the predicted position and $\bar{P}_{th}$ a threshold. Since $\bar{P}_{k+T|k}$ is a matrix and T is a scalar, the trace operator Tr is introduced and T is computed such that:

$$Tr\left[\bar{P}_{k+T|k}\right] \le Tr\left[\bar{P}_{th}\right] \tag{27}$$

$\bar{P}_{k+T|k}$ increases monotonically with T. T is chosen from the condition:

$$Tr[\bar{P}_{k+T|k}] = Tr[\bar{P}_{th}] \tag{28}$$

The sample time is determined by solving Eq. 28, using Newton's methods and choosing the next update time to be the maximum of the possible solutions.

Adaptive update rate control in the IMM algorithm: The detection probability of the target is dependent on the accuracy of the beam pointing, which depends on the accuracy in the target position prediction. Shin *et al*. (1995) integrate the Van Keuk criterion and use the empirical formula Eq. 25 to update the next update time in the IMM algorithm.

The IMM is used to estimate the manoeuvre parameter δ_m:

$$\hat{\delta}_m^2\left(k|z^k\right)=\sum_{j=1}^{r}\mu_j(k)\delta_{m_j}^2 \tag{29}$$

where, $\mu_j(k)$ is the probability of model j at time k, r is the total number of models used in the IMM and $\delta_{m_j}^2$ is the assumed acceleration covariance for model j.

Adaptive IMM algorithm (AIMM): This algorithm uses the IMM, with two models: Constant Velocity (CV) and Constant Acceleration (CA) (Benoudnine *et al*., 1999a, b). The tracking is made in 2 Cartesian co-ordinates (x, y). A variable update time is incorporated into the IMM algorithm using the Van Keuk (1977, 1979) method. For the x direction, the update time $T_x(k)$ at the kth scan is determined from:

$$[\hat{P}(k|k-1)]_{11}=v_0[R]_{11}$$

where, $[\hat{P}(k|k-1)]_{11}$ is the (1,1) element of the predicted covariance matrix, $[R]_{11}$ is the measurement variance in the x direction and v_0 is a constant.

The expression for the (1,1) element of the predicted covariance matrix is given by:

$$[\hat{P}(k|k-1)]_{11}=\sum_{j=1}^{r}\mu_j(k|k-1)\left\{\left[\hat{P}^j(k|k-1)\right]_{11}+\left[P_S^j(k|k-1)\right]_{11}\right\} \tag{30}$$

where, $\mu_j(k|k-1)$ is the predicted probability of model j:

$$[\hat{P}^j(k|k-1)]_{11}=[F^j(T_x(k))\hat{P}^{0j}(k-1|k-1)(F^j(T_x(k)))^T+Q^j(T_x(k))]_{11} \tag{31}$$

and

$$\hat{P}_S^j(k|k-1)=\left[\left\{\hat{x}^j(k|k-1)-\hat{x}(k|k-1)\right\}\left\{\hat{x}^j(k|k-1)-\hat{x}(k|k-1)\right\}^T\right]_{11}$$

In Eq. 31 F^j and Q^j denote the transition matrix and the process noise covariance matrix, both matched to model M_j.

It can be shown from Eq. 30-31 that $[\hat{P}(k|k-1)]_{11}$ is a bi-quadratic polynomial in $T_x(k)$ whose coefficients depend on the elements of the matrices $\hat{P}^{0j}(k|k-1)$, the variances of the process noises used in the models, the components of the initialisation state vectors $\hat{x}^{0j}(k|k-1)$ and the predicted model probabilities $\mu_j(k|k-1)$. The next update time $T_x(k)$ can then be determined by zeroing the bi-quadratic polynomial

$$\left[\hat{P}(k|k-1)\right]_{11}-v_0[R]_{11}$$

and taking the maximum real and positive root.

Similarly, the update time $T_y(k)$, for the y direction, can be obtained by solving the following equation:

$$[\hat{P}(k|k-1)]_{44}=v_0[R]_{22} \tag{32}$$

where, $[\hat{P}(k|k-1)]_{44}$ is the (4,4) element of the predicted covariance matrix and $[R]_{22}$ is the measurement variance in the y direction.

The update T(k) at the kth scan is taken as the time which guarantees a minimum for the position covariance error in x and y.

Fast adaptive IMM algorithm (FAIMM): In Benoudnine *et al.* (2006), a Fast Adaptive IMM algorithm (FAIMM) was proposed to select a next update time, which is appropriate to the motion of the target. In this algorithm, two models are used: (CV) and (CA). We assign to each model in the IMM a suitable rate: T_{max} to the non manoeuvring model and T_{min} to the manoeuvring model. Then, the next update time at scan k is obtained by computing the following mean:

$$T=\sum_{j=1}^{r}T_j\mu_j(k),\ j=1,\ldots,r \tag{33}$$

where, $\mu_j(k)$ is the probability of model j at time k, r is equal to 2, $T_1 = T_{max}$ and $T_2 = T_{min}$.

Adaptive MRIMM algorithm (AMRIMM): In Benoudnine *et al.* (2007), a variable update time is incorporated into the MRIMM algorithm; the resulting algorithm is called the Adaptive MRIMM algorithm (AMRIMM). One of the properties of the MRIMM algorithm is that the state estimate is obtained every second scan (k-2, k, k+2,...,etc), so we propose to calculate the next update time in the MRIMM at each second scan and keep it constant between these two scans. The selected update time is weighted by accurate models probabilities, updated at each second scan.

We assign to each model in the MRIMM a suitable rate: T_{max} to the non manoeuvring model (MRCV) and T_{min} to the manoeuvring model (CA). Then, the next update time at scan k is obtained using the Eq. 33.

SIMULATIONS RESULTS

Monte Carlo simulations are often used to assess the performance of constant update tracking algorithms. The squares of the estimation errors are computed at each time point, for each run and then the root mean square estimation errors are obtained by averaging over all runs and taking the square root. Plots of these RMS errors versus time give an indication of the tracking accuracy time's dependence. This procedure can be used to compare the performance of the algorithms that use a constant update time such as the IMM and the MRIMM.

In the case of the algorithms that use variable update, there is a difficulty in applying this procedure, since the updates occur at different times, from run to run. To overcome this difficulty, we have divided the target trajectory into segments of equal length (10 sec). The mean square error, in each segment, was obtained by averaging over all the points that lie in this segment. Thus the time step in the plots relative to the FAIMM, AIMM and AMRIMM is equal to 10 sec.

To avoid that the update time becomes too small or too large in the AIMM, FAIMM and AMRIMM, it is limited between 0.25 and 5 sec. The full rate in the IMM and MRIMM algorithms is equal to 2 sec.

The measurement noise is generated in polar coordinates with standard deviations of 185.2 m and 2.5×10^{-3} radian in range and bearing, respectively. The range and bearing are then converted to two dimensional Cartesian co-ordinates. The probability of switching between the two models is equal to 0.05 in both algorithms.

The modeling statistics were chosen so that the MRIMM work on its best performance (Hong, 1998, 1999): $Q_{k+1_{H^2}} = Q_{k+2_{H^2}} = 0.3\,I$ and $q_{CAx} = q_{CAy} = 30$for the MRIMM and $q_{CVx} = q_{CVy} = 0.1$ and $q_{CAx} = q_{CAy} = 15$ for the IMM algorithm.

The update time in the algorithms CMRIMM and CIMM is constant and equal to 2 sec.

Targets trajectories: Two target trajectories are used to evaluate the performance of the algorithms; both trajectories consist of three segments.

Trajectory 1: 90° Manoeuvre: The trajectory duration is 246 sec (Fig. 2). The first segment is a non manoeuvring segment with a constant velocity in x equal to 308.67 m sec^{-1}, it lasts 120 sec, starting at the initial position of (129650, 0 m). It is followed by a manoeuvring segment from 120 to 136.48 sec and a non manoeuvring segment from 136.5 to 246 sec.

Trajectory 2: 180° Manoeuvre: The trajectory duration is 200 sec (Fig. 3). Starting at the initial position of (0, 9166.942 m), the target travels at a constant velocity in x and y equal to 218.26 m sec^{-1}, for 80 sec. It then undertakes a manoeuvre from 80 to 113 sec, before resuming to a quiescent motion for the remaining of the trajectory.

The first results concern the IMM and MRIMM with a constant update time. The RMSE in position on x and y coordinates obtained over 1000 Monte Carlo are presented in Fig. 4-7 for trajectory one and two, respectively.

We can observe from these figures that the performance of the MRIMM algorithm is better than that of the IMM during the non manoeuvring segments and that the two algorithms have approximately the same performance during the manoeuvring segment with a significant error at the onset of the manoeuvre and lateness in detecting the end of manoeuvre, especially for the MRIMM. This lateness in the case of the MRIMM can be explained by the fact that its non manoeuvring

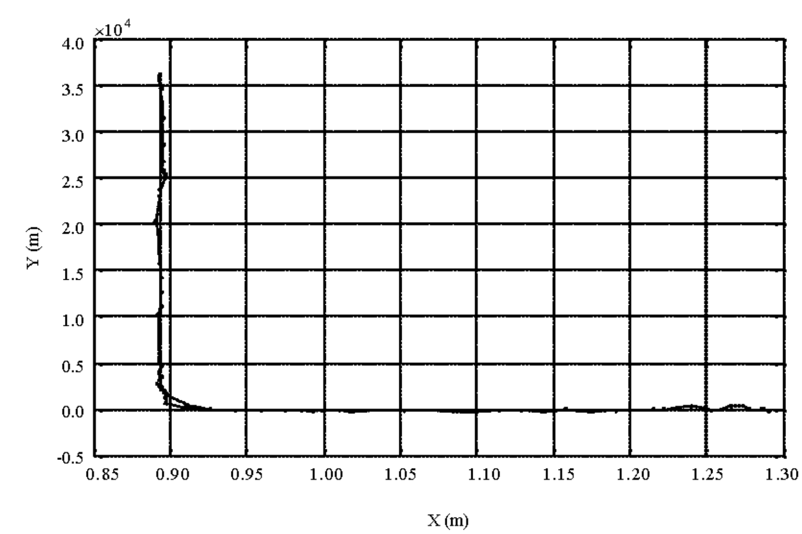

Fig. 2: True and estimated target trajectory 1 (90° manoeuvre)

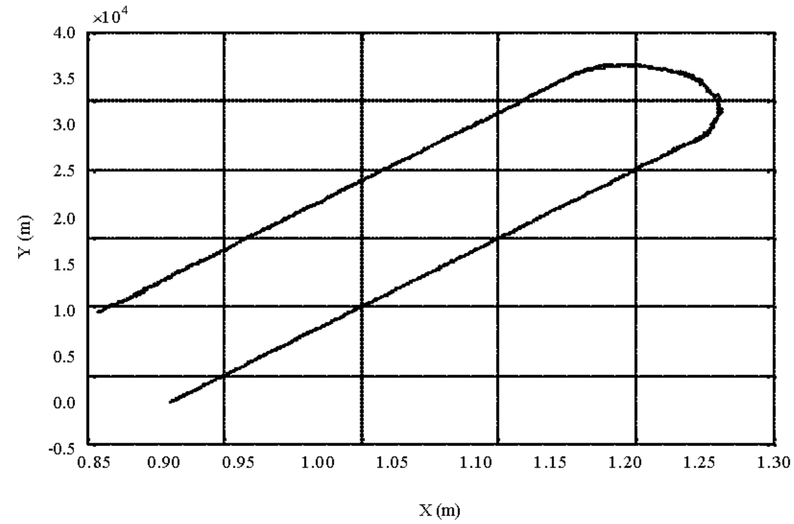

Fig. 3: True and estimated target trajectory 2 (180° manoeuvre)

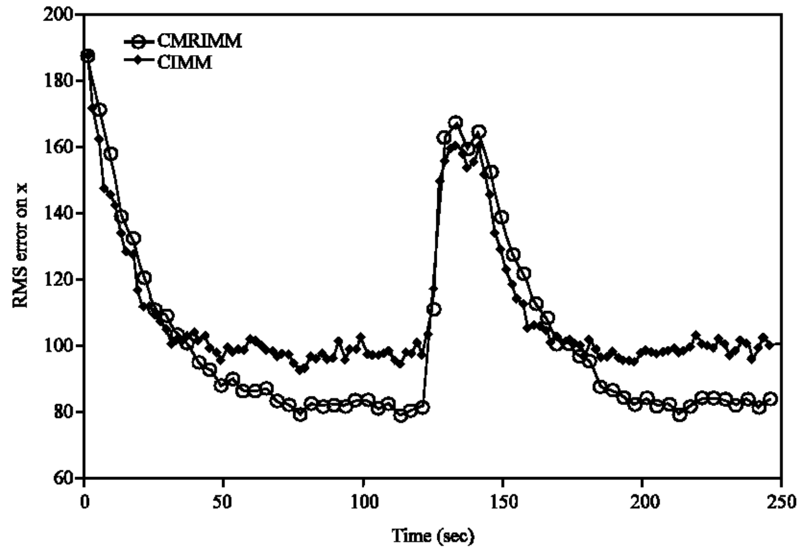

Fig. 4: RMSE in x coordinates for CIMM and CMRIMM algorithms, for trajectory 1

model is updated at half rate. However, the MRIMM is faster than the IMM. It can save between 15 and 25% of the IMM's computation time.

For a fair comparison between the algorithms that use a variable update time and those that use a constant update time, the constant update time has been chosen equal to the mean update time obtained by the FAIMM, the AIMM and AMRIMM over 1000 Monte Carlo simulations.

In Fig. 8 and 9, the RMSE in x and y co-ordinates for the FAIMM, AIMM and CIMM obtained over 1000 Monte Carlo simulations are presented. It can observed that for the same mean update time (3.5 and 2.8 sec, for trajectory 1 and 2, respectively), the performance of the FAIMM algorithm is better than those of the AIMM and the IMM during the manoeuvring segments. We also observe that the FAIMM has approximately the same performance as the AIMM during the non manoeuvring segments.

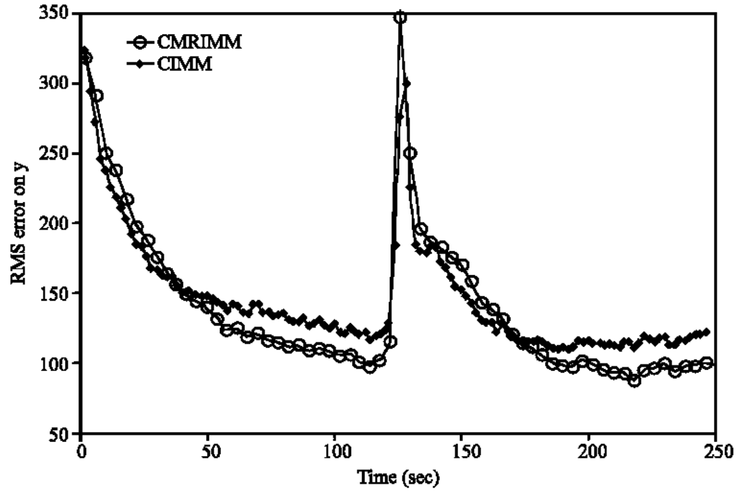

Fig. 5: RMSE in y coordinates for CIMM and CMRIMM algorithms, for trajectory 1

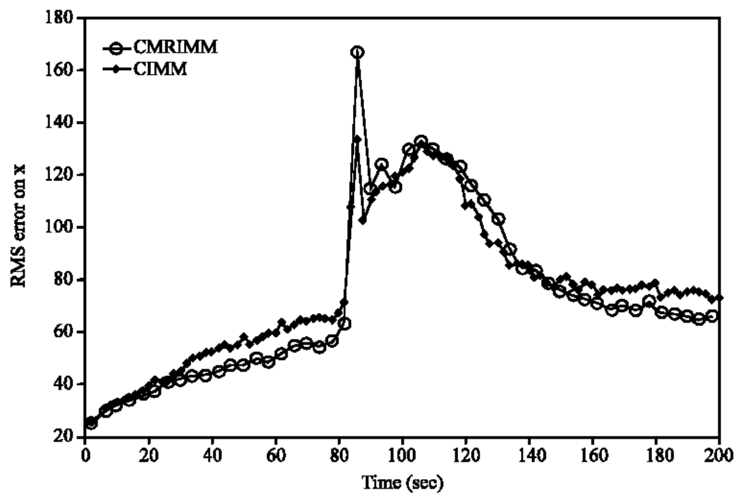

Fig. 6: RMSE in x coordinates for CIMM and CMRIMM algorithms, for trajectory 2

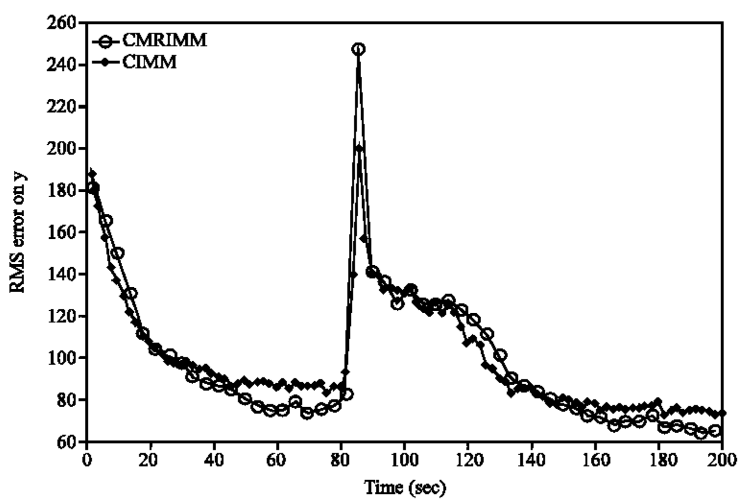

Fig. 7: RMSE in y coordinates for CIMM and CMRIMM algorithms, for trajectory 2

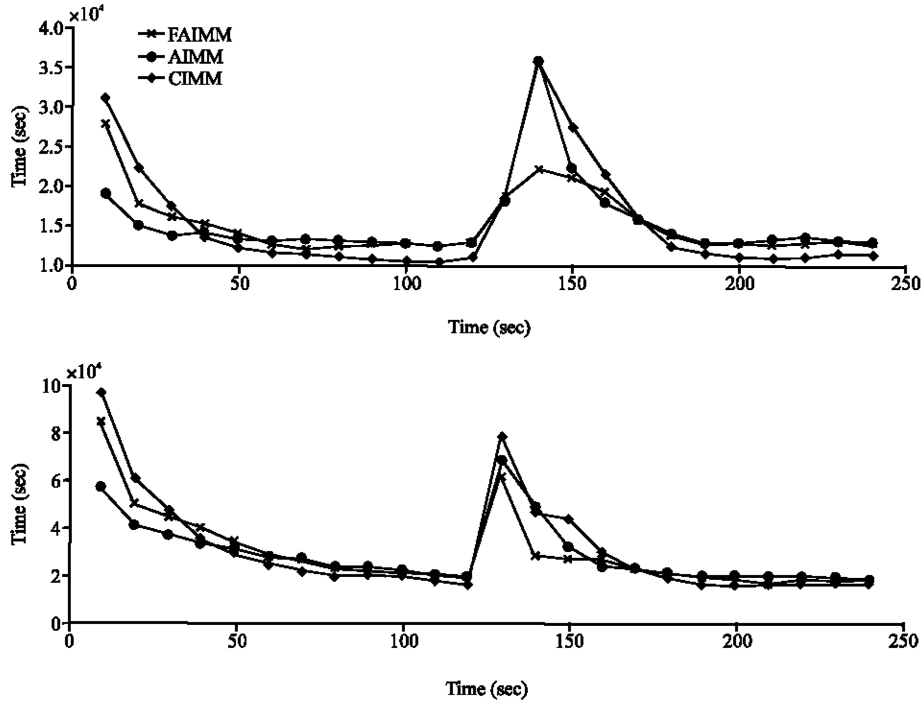

Fig. 8: RMSE in x and y coordinates for FAIMM, AIMM and CIMM algorithms over 1000 Monte Carlo simulations, for trajectory 1

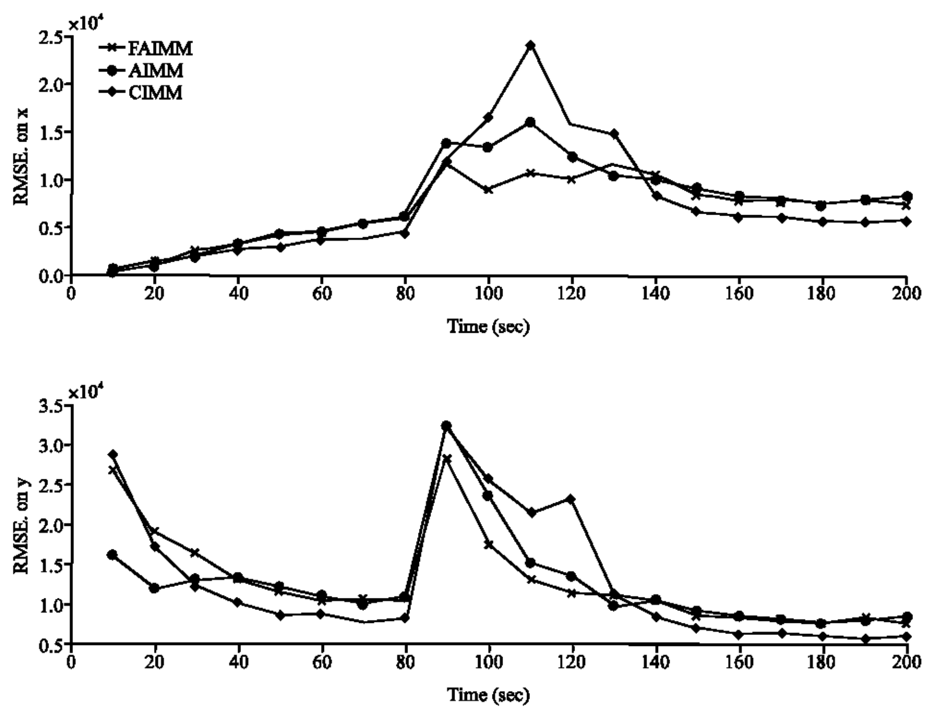

Fig. 9: RMSE in x and y coordinates for FAIMM, AIMM and CIMM algorithms over 1000 Monte Carlo simulations, for trajectory 2

However, about 25 to 40% of computation can be saved by using the FAIMM algorithm, instead of the AIMM algorithm.

In Benoudnine *et al.* (2007), it has been shown, that for the same mean update time, the performance of the AMRIMM is better than that of the CMRIMM and CIMM during the manoeuvring segments and it is approximately the same as that of the CMRIMM, during the non manoeuvring segments. However, at the onset of the manoeuvre, the CIMM is more responsive than the MRIMM algorithm.

The results of RMSE in x and y co-ordinates, obtained over 1000 Monte Carlo simulations using trajectory 2, for FAIMM and AMRIMM are presented in Fig. 10 and 11.

It can be observed that the performance of the FAIMM is better than that of the AMRIMM at the onset of the manoeuvre and has roughly the same performance during the manoeuvring and non manoeuvring segments.

In Fig. 12, the averaged update times over 1000 Monte Carlo runs, for the FAIMM and the AMRIMM algorithms are plotted versus the time for trajectory 2.

This shows that AMRIMM and the FAIMM adapt their update time to the motion of the target. It is reduced in response to a manoeuvre of the target and again increased when the manoeuvre ceases.

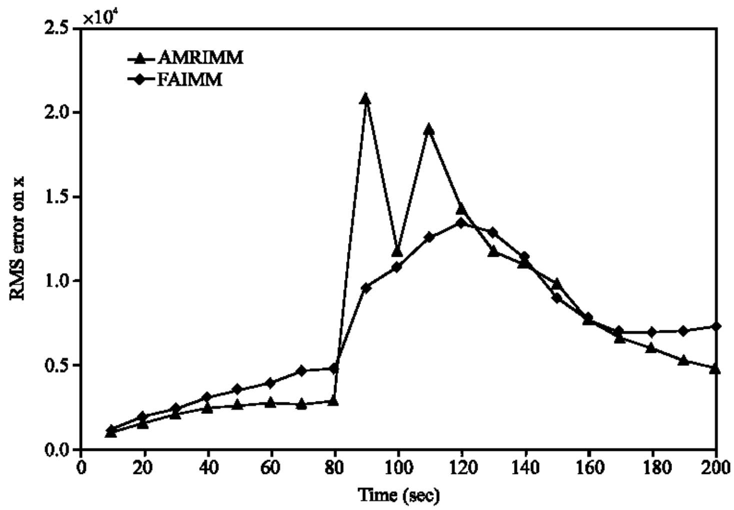

Fig. 10: RMSE in x coordinates for FAIMM and AMRIMM algorithms, for trajectory 2

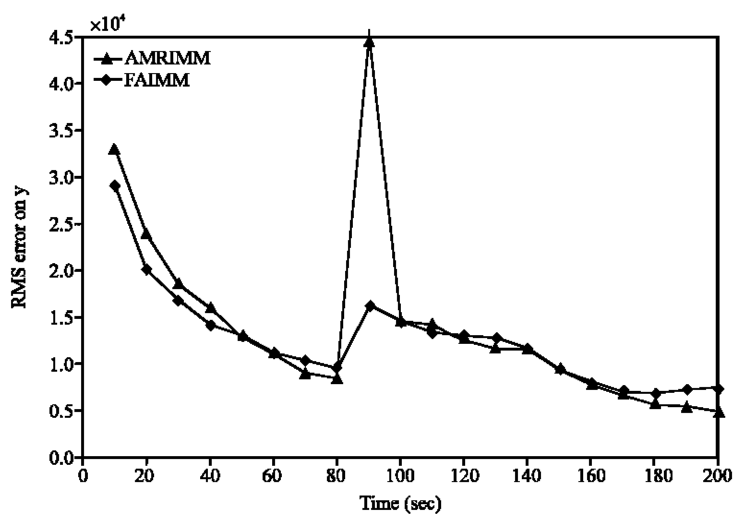

Fig. 11: RMSE in y co-ordinates for FAIMM and AMRIMM algorithms, for trajectory 2

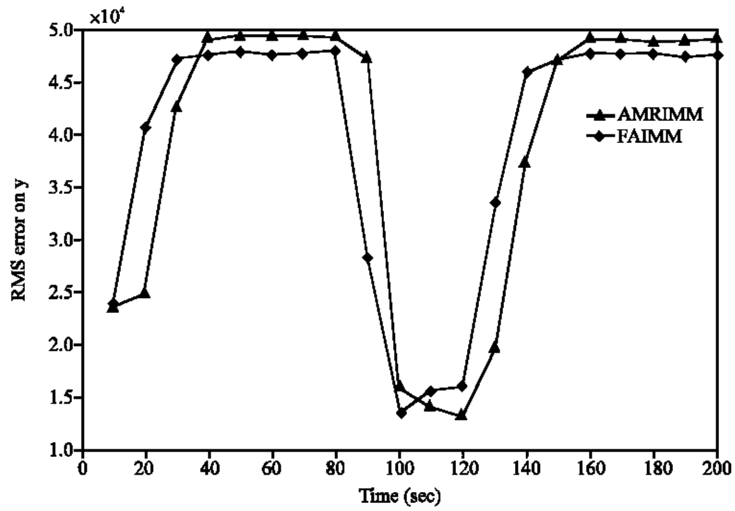

Fig. 12: Update time versus time for FAIMM and AIMM algorithms, for trajectory 2

CONCLUSION

A comparison of two adaptive manoeuvring target tracking algorithms namely the IMM and MRIMM is presented. The results show that both IMM and MRIMM have a good trade off between complexity and performance. The following conclusions can be made:

- The MRIMM improves the tracking accuracy of the IMM during the non manoeuvring segments
- The MRIMM has the same performance as the IMM algorithm, during the manoeuvring segments
- The IMM is more sensitive to any change in the motion of the target (onset and offset of manoeuvre)
- About 15 to 25% of the computation time is saved when using the MRIMM instead of the IMM algorithm

In the second part, a fast method for selecting adaptively the next update time in a Phased Array Radar is incorporated into the IMM and the MRIMM algorithms. The resulting algorithms are named, respectively, Fast Adaptive IMM (FAIMM) and Adaptive MRIMM (AMRIMM). The following conclusions can be drawn:

- Compared to the AIMM and CIMM algorithms, the FAIMM has globally a better performance in terms of tracking accuracy and complexity
- The AMRIMM algorithm improves the tracking accuracy and computation complexity of the MRIMM algorithm by using an adaptive variable Update time

Both FAIMM and AMRIMM achieve a good compromise between complexity and tracking accuracy and are therefore good candidates for tracking a manoeuvring target.

REFERENCES

Ahmeda, S.S., M. Keche, I. Harrison and M.S. Woolfson, 1997. Adaptive joint probabilistic data association algorithm for tracking multiple targets in cluttered environment. IEE Proc. Radar, Sonar, Navigigation, 144: 309-314.

Bar Shalom, Y., 1995. Multi target multisensor tracking: principles and techniques. 1st Edn., Yaakov Bar-Shalom Publishing, USA., ISBN-13: 978-0964831209, pp: 615.

Benoudnine, H., M. Keche, A. Ouamri, M.S. Woolfson and I. Harrison, 1999a. AIMMJPDAF algorithm for tracking multiple manoeuvring targets in cluttered environment. IASTED International Conference on Modeling, February 1999, Identification and Control, Austria, pp: 29-32.

Benoudnine, H., M. Keche, A. Ouamri, M.S. Woolfson and I. Harrison, 1999b. Improved AIMMJPDAF algorithm for tracking multiple manoeuvring targets in cluttered environment. International Radar Conference, RADAR 99, Brest, France.

Benoudnine, H., M. Keche, A. Ouamri and M.S. Woolfson, 2006. A fast adaptive update rate for a phased array radar using IMM target tracking algorithm. IEEE Symposium Signal Processing and Information Technology, ISSPIT 2006, August 27-30, Vancouver, Canada, pp: 277-282.

Benoudnine, H., M. Keche, A. Ouamri and M.S. Woolfson, 2007. Adaptive MRIMM for tracking manoeuvring target using phased array radar. IET International Conference on Radar Systems, Radar2007, Edinburgh, UK, 15-18 October2007.

Daeipour, E., Y. Bar-Shalom and X. Li, 1994. Adaptive beam pointing control of a phased array radar using an IMM estimator. Am. Control Conf., 2: 2093-2097.

Efe, M. and D.P. Atherton, 1996. Beam pointing control of a phased array radar using the IMM algorithm. IEE Target Tracking and Data Fusion Colloquium, November 1996.

Efe, M. and D.P. Atherton, 1998. A tracking algorithm for both highly maneuvering and non maneuvering targets. Proceeding 36th Conference Decision and Control, December 1998.

Hong, L., Z. Ding and R.A. Wood, 1998. Development of multirate model and multirate IMM algorithm for platform multisensor tracking. Optical Eng., 37: 453-467.

Hong, L., 1999. Multirate interacting multiple model filtering for target tracking using multirate models. IEEE Trans. Automatic Control,IEEE Trans. Automatic Control, 44: 1326-1340.

Hong, L. and Z. Ding, 2000. A distributed multirate IMM algorithm for multiplatform tracking. Math. Comput. Modeling, 32: 1095-1116.

Kirubarujan, T., Y. Barshalom and E. Daeipour, 1995. Adaptive beam pointing control of a phased array radar in the presence of ECM and false alarms using IMMPDAF. Proceeding American Control Conference, June 21-23, Seattle, WA, USA., pp: 2616-2620.

Li, X.R. and Y. Bar Shalom, 1996. Multiple model estimation with variable structure. IEEE Trans. Automatic Control, 42: 478-493.

Lin, H.J. and D.P. Atherton, 1993. An investigation of the SFIMM algorithm for tracking manoeuvring targets. Proceeding of 32nd IEEE Conference on Decision and Control, Dec. 15-17, San Antonio, TX, pp: 930-935.

Maybeck, P.S. and K.P. Hentz, 1987. Investigation of moving bank multiple model adaptive algorithm. AIAA, J. Guidance Control Dynam., 10: 90-96.

Sarunic, P.W. and R.J. Evan, 1997. Adaptive update rate using IMM nearest neighbour algorithm incorporating rapid re-looks. IEE Proc. Radar, Sonar, Navig., 144: 195-204.

Shin, A.J., S.M. Hong and D.H. Hong, 1995. Adaptive-update-rate target tracking for phased array radar. IEE Proc. Radar, Sonar Navigation, 142: 137-143.

Van Keuk, G., 1977. Adaptive computer controlled target tracking with phased array radar. IEEE International Radar Conference, April 1977, IEEE Xplore London, pp: 429-434.

Van Keuk, G., 1979. Software structure and sampling strategy for automatic target tracking with phased array radar. Proceedings of AGARD Conference, Monterey, CA, 1979, No. 252, pp: 11-1-11-13.

Watson, G.A. and W.D. Blair, 1994. Tracking performance of a phased array radar with revisit time controlled using the IMM algorithm. IEEE National Radar Conference, March 29-31, Atlanta, GA, USA., pp: 60-165.