Journal of Computer Science 9 (12): 1695-1709, 2013
ISSN: 1549-3636

doi:10.3844/jcssp.2013.1695.1709 Published Online 9 (12) 2013 (http://www.thescipub.com/jcs.toc)

NEW TRACK-TO-TRACK CORRELATION ALGORITHMS BASED ON BITHRESHOLD IN A DISTRIBUTED MULTISENSOR INFORMATION FUSION SYSTEM

[1]Liu Yu, [1]Wang Haipeng, [1]He You, [1]Dong Kai and [2]Xiao Chuwan

[1]Research Institute of Information Fusion,
[2]Training centre of New Equipment,
Naval Aeronautical and Astronautical University, Yantai, 264001, China

Received 2013-08-15, Revised 2013-10-19; Accepted 2013-10-29

ABSTRACT

Track-to-Track correlation (or association) is an ongoing area of interest in the field of distributed multisensory information fusion. In order to perform accurately identifying tracks with common origin and get fast convergence, this study presents independent and dependent Bi-threshold Track Correlation Algorithms (called BTCAs), which are described in detail and the track correlation mass and multivalency processing methods are discussed as well. Then, Based on BTCAs, two modified Bi-threshold Track Correlation Algorithms with average Test Statistic (called BTCA-TSs) are proposed. Finally, simulations are designed to compare the correlation performance of these algorithms with that of Singer's and Bar-Shalom's algorithms. The simulation results show that the performance of these algorithms proposed in this study is much better than that of the classical methods under the environments of dense targets, interfering, noise and track cross and so on.

Keywords: Data Fusion, Track Correlation, Radar Network, Fuzzy Set

1. INTRODUCTION

Using data from multisensor system, the multitarget tracking technology has been largely applied to military affairs and public affairs. In some applications, the data is collected by many sensors distributed over a large area. In view of the security, viability and communication bandwidth of such a multisensor system, it's unreliable to process these data with centralized method. However, the distributed structure of processing method is appreciable.

Track-to-trackassociatio n problem (You *et al*., 1996; Singer and Kanyuck, 1971; Bar-Shalom and Fortmann, 1988; Gul, 1994; Kosoka, 1983; You *et al*., 1989; Bowman, 1979; Chang and Youens, 1982; Bar-Shalom and Chen, 2004; Kaplan *et al*., 2008; Tian and Bar-Shalom, 2011; Bar-Shalom and Campo, 1986; Mori *et al*., 2011; Osbome *et al*., 2011; Wang *et al*., 2012; La Scala and Farina, 2002; Bar-Shalom, 2008) is a crux of distributed multisensor system. It's a problem of how to decide whether two tracks coming from different sensor systems represent the same target. The issue of track-to-track association was first considered in presented by Singer and Kanyuach (1971), assuming tracks with independent estimation errors (Singer's algorithm). Then, Bar-Shalom extended to the case of correlated errors in (Bar-Shalom and Fortmann, 1988) (Bar-Shalom's algorithm) and Kosaka presented the Nearest Neighbor (NN algorithm) in (Gul, 1994). A K-Nearest Neighbor algorithm was given in (Kosoka, 1983) and Bowman proposed a Maximum likelihood algorithm in (You *et al*., 1989). Chang and Youens (1982) transformed track-to-track association into Multidimensional assignment

Corresponding Author: Liu Yu, Research Institute of Information Fusion, Naval Aeronautical and Astronautical University, Yantai, 264001, China

problem and get it resolved with Hunger/Munker method (Bowman, 1979).

In these algorithms above, Singer's algorithm, Bar-Shalom's algorithm and NN algorithm are usually applied to the actual system. However, these algorithms will lead to false correlation or missing correlation when under the environments of dense targets, interfering, noise, track cross and so on. To resolve these problems of Singer's algorithm and Bar-Shalom's algorithm and in view of history information of tracks, several bi-threshold track correlation algorithms based on the double threshold detection method are proposed in this study. Besides, effective track correlation mass management and multivalency processing methods are discussed as well to get higher correlation precision and faster convergence.

1.1. System Description

The dynamics of target can be modeled in the discrete form as follows Equation (1):

$$X(k+1) = F(k)X(k) + G(k)V(k) \quad k = 1,2.....,\tag{1}$$

Where:

$X(k) \in R^n$	=	The state vector at time k
$V(k) \in R^n$	=	A sequence of zero-mean, white Gaussian process noise with covariance matrix
$Q(k)$ and $F(k) \in R^{n,n}$	=	The transition matrix of the system
$G(k) \in R^{n,h}$	=	The noise distribution matrix

The initial state vector is assumed normally distributed with mean μ and covariance P(0). Therefore, one knows that Equation (2):

$$E[V(k)] = 0,\ E[V(k)V(k)] = Q(k)\delta_{kl}\tag{2}$$

The measurement of node i at time k is given by Equation (3):

$$Z^i(k) = H^i(k)X(k) + W^i(k)\tag{3}$$

where, $W^i(k)$ is a sequence of zero-mean, white, Gaussian measurement noise vector with covariance $R^i(k)$, $H^i(k) \in R^{m,n}$ is the measurement vector of the ith node at 2 time k and I = 1, 2,…,,M. This study only discusses the situation of M = 2. Assuming that the measurement noise sequences are independent Equation (4):

$$E[W^i(k)] = 0, E[W^i(k)W^i(k)']R^i(k)\delta_{kl}\tag{4}$$

Each node processes its observations locally to produce the state estimation and prediction of a target by using Kalman filter. Assume that the tracks of targets had been initialed by using some multitarget tracking algorithms and the state estimation of targets gained in each node would be communicated to a central processor, where track fusion takes places. The state estimation of the *t*th target from the *i*th sensor can be written as follows Equation (5):

$$\begin{aligned}\hat{X}_t^i(k+1|K+1) &= \hat{X}_t^i(k+1|K) + K_t^i(k+1)\\ &\bullet[Z_t^i(k+1) - H^i(k+1)\hat{X}_t^i(k+|1|k)]\end{aligned}\tag{5}$$

One-step prediction of the state is Equation (6):

$$\hat{X}_t^i(k+1|k) = F(k)\hat{X}_t^i(k|k)\tag{6}$$

and the one-step prediction covariance is Equation (7):

$$P_t^i(k+1|k) = F(k)P_t^i(k|k)F'(k) + G(K)F(k)G'(K)\tag{7}$$

the filter gain is Equation (8):

$$\begin{aligned}K_t^i(k+1) &= P_t^i(k+1|k)H^{i'}(k+1)[H^i(k+1)P_t^i\\ &\bullet(k+1|k)H^{i'}(k+1) + R^i(k+1)]^{-1}\end{aligned}\tag{8}$$

and the update state covariance is Equation (9):

$$\begin{aligned}&P_t^i(k+1|k+1) = [I - K_t^i(k+1)H^i(k+1)P_t^i(k+1|k)]\\ &i = 1,2,....,M, t = 1,2,....,n_i\end{aligned}\tag{9}$$

1.2. BI-Threshold Track Correlation Algorithms

1.2.1. Independent and Dependent Bi-threshold Track Correlation Algorithm

There is a double threshold detection signal processing method in the automatic radar detection theory (La Scala and Farina, 2002). Based on the double threshold detection method, independent and dependent bi-threshold track correlation algorithms are proposed here.

Define the sets of track number initialed by node 1 and node 2 Equation (10):

$$U_1 = \{1,2,....,n_1\}, U_2 = \{1,2,....,n_2\}\tag{10}$$

Let $\hat{X}_i^1(l)$ denotes the state of target i estimated by node 1. Assume that for the same time one has an estimate $\hat{X}_j^2(l)$ of target j from node 2. Denote $\hat{t}_{ij}(l)$ as the estimation of $\hat{t}_{ij}(l)$ and Equation (11 and 12):

$$\hat{t}_{ij}(l) = \hat{X}_i^1(l) - \hat{X}_j^2(l) \quad (11)$$

$$t_{ij}(l) = \hat{X}_i^1(l) - \hat{X}_j^2(l) \ \ (i \in U_1, j \in U_2) \quad (12)$$

where, X_i and X_j are the corresponding true states. One wants to test for the "same target" hypothesis $H_0 : \hat{X}_i^1(l)$ and $\hat{X}_j^2(l)$ are the estimations of the same target Vs.

$H_1 : \hat{X}_i^1(l)$ and $\hat{X}_j^2(l)$ are the estimations of different targets. Then, the problem of track correlation becomes the hypothesistesting problem.

The Independent Bi-threshold Track Correlation Algorithm (IBTCA) can be described as follows.

Using the test variable of Singer's algorithm Equation (13 and 14):

$$\xi_{ij}(l) = \hat{t}_{ij}(l)'[P_i^1(l) + P_j^2(l)]^{-1}\hat{t}_{ij}(l) \quad l = 1,2,....,R \quad (13)$$

$$\begin{cases} m_{ij}(l) = m_{ij}(l-1) + 1 \\ m_{ij}(0) = 0, \text{ if } \xi_{ij}(l) < \delta \end{cases} \quad (14)$$

where, $P_i^1(l)$ is the estimation error covariance of node 1 corresponding to target i and m_{ii} (l) denotes the correlation mass that track i from node 1 correlated with track j from node 2 till time l. The first threshold is set as follow Equation (15):

$$P\{\zeta_{ij}(l) > \delta \mid H_0\} = \alpha \quad (15)$$

where, α is, say, 0.05. Then the test of H_0 vs. H_1 is as follow Equation (16):

$$\text{accept } H_1 \text{ if } m_{ij}(R) < L \quad (16)$$

However, H_0 may not be accepted if m_{ij} (l)≥L, l = L, L+1,…,R for that there may be more than one track will be correlated with track i. This problem is treated in the following part.

In the Dependent Bi-threshold Track Correlation Algorithm (DBTCA), the test variable of Bar- shalom's algorithm is used Equation (17):

$$\psi_{ij}(l) = \hat{t}_{ij}(l)'[P_i^1(l) + P_j^2(l) - P_{ij}^{12}(l) - P_i^{12'}(l)]^{-1}\hat{t}_{ij}(l) \quad l = 1,2.....R \quad (17)$$

where, $P_{ij}^{12}(l)$ denotes the cross-covariance Equation (18):

$$P_{ij}^{12}(l) = [I - K_i^1(l)H^1(l)][\Phi(l-1)P_{ij}^{12}(l)\Phi'(l-1) + G(l-1)Q_i(l-1)G'(l-1)][I - K_j^2(l)H^2(l)] \quad (18)$$

which is a linear recursion with initial condition $P_{ij}^{12}(0) = 0$. Under the Gaussian distributed assumptions, ζ_{ij} (l) and ψ_{ij} (l) is chi-square distributed with x n degrees of freedom. The x n here denotes the dimension of state estimation vector.

1.3. Track Mass Designing

Two kinds of track mass are designed here. One of them is track correlation mass and the other one is track separation mass. Similar to the association mass (La Scala and Farina, 2002; Bar-Shalom, 2008), the track correlation mass m_{ii} (l) denotes the times of track i from node 1 correlated with track j from node 2 till time l and the separation mass of track i and j is defined as follow Equation (19):

$$\begin{cases} D_{ij}(l) = D_{ij}(l-1) + 1 \\ D_{ij}(0) = 0, \ \zeta_{ij}(l) \geq \delta \text{ or } \psi_{ij}(l) \geq \delta \end{cases} \quad (19)$$

From (18) one can see that if Equation (20):

$$D_{ij}(l-1) > R - L \text{ (Where R and L have been set)} \quad (20)$$

The correlation test would not be performed between track i and j at time l. Since m_{ij} (l = R)<L (track i and j are uncorrelated) must be in existence if D_{ij} >R-L at time l ̃ 1. Similarly, the correlation between track i and j will be nearly confirmed if Equation (21):

$$m_{ij}(l-1) \geq L \quad (21)$$

The correlation test between track i and j would be cease at time l if only one track (j) can satisfy (20), then track i and j would be regarded as the correlated track and performed no correlation test any more. However, if there are more than one track (j) can suffice (20), the correlation test should be performed last l = R to give a precise correlation mass for the multivalency processing latter. On the other hand, the track I with no other track correlated till l=R will be performed test in the next cycle.

1.4. Multivalency Processing Method

There are two situations where multivalency processing method applied, one of them is l = R and the other is l<R. In case one, there are more than one track (j) suffice for m_{ij} (l = R)≥L thus will be correlated with track i. In this case, track j* which maximize the track correlation mass m_{ij}(l) will be correlated with track i Equation (22):

$$j^* = \arg\max m_{ij}(l = R)\ j \in \{j_1, j_2, ..., j_q\} \quad (22)$$

where, $\{j_1, j_2, .., j_q\}$ is the set of track (j) correlated with track i. When there are more than one track can maximize track correlation mass m_{ij} (l), the track $j^\bullet$ will be accepted if Equation (23 and 24):

$$\bar{\zeta}_{ij^\bullet}(R) = \min_{j^*} \frac{1}{R}\sum_{l=1}^{R} \zeta_{ij^*}(l) j^* \in \{j_1^*, j_2^*, ...j_q^*\} (\text{IBTCA}) \quad (23)$$

$$\bar{\psi}_{ij^\bullet}(R) = \min_{j^*} \frac{1}{R}\sum_{l=1}^{R} \psi_{ij^*}(l) j^* \in \{j_1^*, j_2^*, ...j_q^*\} (\text{DBTCA}) \quad (24)$$

In case two, the correlation test will be ceased if (19) is sufficed. Otherwise, a temp system track will be set. Corresponding to a given track i, the track j* is accepted if j* argmax m_{ij}(l) .

If there are more than one track (j*) accepted, the multivalency processing method will be applied. In this case, the track j will be correlated with track i if **argmin D_{ij} (l). However, if there are more than one track which can be correlated with track i, the track jp will be accepted if Equation (25):

$$j_P = \arg\min_{j^*} \frac{1}{l}\sum_{q=1}^{l} \left\| \tilde{x}_{ij^\bullet}(q) \right\|\ j^\bullet \in \{j_1^\bullet, j_2^\bullet, ..., j_r^\bullet\} \quad (25)$$

Where Equation (26):

$$\tilde{x}_{ij^\bullet}(q) = \hat{X}_i(q) - \hat{X}_{j^\bullet}(q) \quad (26)$$

and $\{j_1^\bullet, j_2^\bullet, ..., j_r^\bullet\}$ is the set of track (j*) can satisfy**argmin D_{ij} (l).

Once j is correlated with track i, the correlation test would not be performed to track i or j at time l. Since the data of track transformed seriatim by each sensor, set of L/R should be dynamic, such as 1/1, 2/2, 2/3, 3/4, 3/5, 4/5, 4/6, 5/7, 6/8 and so on. Also one can see that Singer's and Bar-Shalom's algorithm is a specific presentation of independent and dependent bi-threshold track correlation algorithm when L/R = 1/1.

1.5. Estimation to Sans Correlation Probability

Let P_t (A) denote the probability of statistical distance from the same target accepted by the first threshold. According to the rule of x^2 test, P_t (A) = α (α is set in (15)). Assuming that the cumulative R estimation error swatches are statistical independent, P_t(A = Y) is binomial distributed Equation (27):

$$P_t(A = R - L + 1) = \alpha^A (1-\alpha)^{R-A} = \alpha^A (1-\alpha)^{L-1} \quad (27)$$

Then, the sans probability can be estimated as follow Equation (28):

$$\hat{P}_s = P_t(A \ge y) = \sum_{A=R-L+1}^{R} C_R^A \alpha^A (1-\alpha)^{R-A} \quad (28)$$

where, L is the second threshold. From (28) one can calculate the sans probability in the case of L/R = 3/4 and L/R = 6/8: $\hat{P}_s$ (3/ 4) 0.002256 $\hat{P}_s$ (6/8) = 0.000102. Therefore, (28) can be used to set the value of L/R.

1.6. Modified Bi-Threshold Algorithms Based On Average Test Statistic

Bar-Shalom and Campo (1986), a new state statistic for correlation hypothesis is defined as follows Equation (29):

$$\begin{aligned}\lambda_{ij}(k) &= \sum_{l=1}^{k} \hat{t}_{ij}(l)' C^{-1}(l) \hat{t}_{ij}(l) \\ &= \lambda_{ij}(k-1) + \hat{t}_{ij}(k)' C_{ij}^{-1}(k) \hat{t}_{ij}(k)\end{aligned} \quad (29)$$

where, $C_{ij}^{-1}(k) = P_j^1(l) + P_j^2(l)$ and λ_{ij} (0) = 0. Under the Gaussian distributed assumptions, the individual terms Equation (30):

$$\varepsilon_{ij}(K) = \hat{t}_{ij}(k)' C_{ij}^{-1}(k) \hat{t}_{ij}(k) \quad (30)$$

Known as the normalized estimation error squared, are each chi-square distributed with x n degrees of freedom, where x denotes the dimension of state estimation vector. It should be noticed that the sum of chi-square variables (λ_{ij} (k)) as an approximately chi-square distribution with x kn degrees of freedom (and thus approximately mean x kn and variance 2 x kn) (Bar-Shalom, 2008).

Next, a modified function based on average test statistic for independent bi-threshold correlation (IBCTA-ATS) is defined as follows Equation (31):

$$\begin{aligned}\varphi_{ij}(k) &= \frac{\lambda_{ij}(k)}{k} \\ &= \frac{1}{k}\sum_{l=1}^{k} \hat{t}_{ij}(l) \left[P_i^1(l) + P_j^2(l)\right] \hat{t}_{ij}(l), \varphi_{ij}(0) = 0\end{aligned} \quad (31)$$

Approximately, φ_{ij} (k) is a chi-square distributed random variable with x n degrees of freedom, which can be used for correlation hypothesis test. Then the test of H_0 vs. H_1 is as follows Equation (32):

$$\text{accept } H_0 \text{ if } \varphi_{ij}(k) \le \delta(k), i \in U_1, j \in U_2 \quad (32)$$

The threshold is set such that Equation (33):

$$P\{\varphi_{ij}(k) > \delta(k) | H_0\} = \alpha \quad (33)$$

where, α is the significance level with $\alpha = 0.05$.

In IBCTA-ATS, track mass and multivalency processing method is as showed before.

In view of the dependence between the estimation errors from the two track files arises from the common process noise,a dependent sbi-threshold correlation algorithm based on Average Test Statistic (DBCTA-ATS) is presented here. All steps of hypothesis test for track correlation are as described in IBCTA-ATS with the following modifications. With theknown cross-covariance $P_{ij}^{12}(l)$, test statistic in (31) is modified to Equation (34):

$$\gamma_{ij}(k) = \frac{1}{k} \bullet \sum_{l=1}^{k} \hat{t}_{ij}(l)'[P_i^1(l) + P_j^2(l) - P_{ij}^{12}(l) - P_{ij}^{12'}(l)]\hat{t}_{ij}(l) \quad (34)$$

We all known that the optimal test would require using the entire database through time k and this is not easy to realize. However, the history information of track has been used in the four algorithms proposed in this study and the computation and memory requirements of these new algorithms will not grow obviously since each test statistic (as showed in (13), (17), (31) and (32)) has a recursive structure.

1.7. Simulation

One has run simulations to compare the correlative performance of four bi-threshold track correlation algorithms here with the Singer's and Bar-Shalom's algorithm.

1.8. Simulation Model and Parameter Settings

There are two nodes considered in the simulations and a 2-D radar is set in each node. A Monte Carlo simulation with 50-runs was carried out for two environments. In case 1, there are 60 targets and there are 120 targets that composed of a lots maneuvering, cross and split targets in case 2. The maneuvers of these targets are random and the initial positions of these targets are normally distributed in a region illustrated in **Fig. 1**. The initial velocity and azimuth of these targets are uniformly distributed in 4~1200 m sec^{-1} and 0~2 π, respectively. **Figure 1**, r_1, r_2 denote the observation radius, r_1', r_2' denote the radius of undetectable area and o', o'' are the coordination origin of nodes. Where r_1' = 110, r_2' =120, 1 r_1' = 2, r_2' = 2.5, a = b = 125, c = 235, d = 130, x_1 = 380, y_1 = 270 km.

The state vector in (1) is $X = (x,\dot{x},y,\dot{y})'$, the transition matrix and noise distribution matrix is Equation (35):

$$F(k) \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} G(k) = \begin{bmatrix} T/2 & 0 \\ 1 & 0 \\ 0 & T/2 \\ 0 & 1 \end{bmatrix} \quad (35)$$

where, T is the sample interval and T = 4s.

The measurement vector in (2) is Z =(x,y)', the measurement matrix is Equation (36):

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (36)$$

And Equation (37):

$$\begin{cases} Q(k) = \begin{bmatrix} q_{11}(k) & \\ & q_{22}(k) \end{bmatrix} \\ \sqrt{q_{11}(k)} = 15 \times 10^{-2} \dot{x}(k) \\ \sqrt{q_{22}(k)} = 15 \times 10^{-2} \dot{y}(k) \end{cases} \quad (37)$$

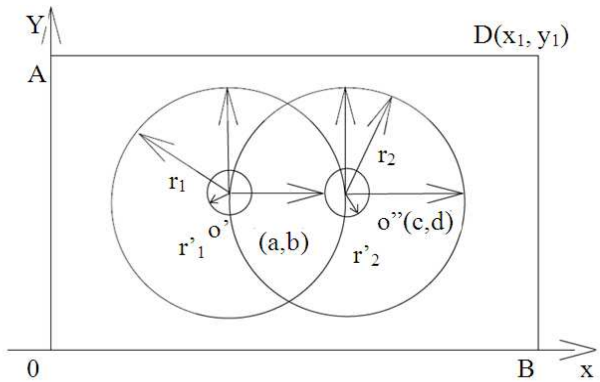

Fig. 1. The observation area of sensors

The noise process standard deviations of rang and azimuth measurements from each sensor are assumed to be 170m and 0.017rad, 180m and 0.017rad, respectively. The measurement noise covariance matrix is Equation (38 and 39):

$$R(k)=\begin{bmatrix}\sigma_x^2(k) & \sigma_{xy}(k)\\ \sigma_{yx}(k) & \sigma_y^2(k)\end{bmatrix} \quad (38)$$

$$\begin{cases}\sigma_x^2(k)=\sigma_\rho^2\cos^2\theta(k)+p^2(k)\sigma_\theta^2\sin^2\theta(k)\\ \sigma_y^2(k)=\sigma_\rho^2\sin^2\theta(k)+p^2(k)\sigma_\theta^2\cos^2\theta(k)\\ \sigma_{xy}^2(k)=[\sigma_\rho^2-p^2(k)\sigma_\theta^2]\sin^2\theta(k)\cos\theta(k)\end{cases} \quad (39)$$

where, $\sigma_{xy}(k)=\sigma_{yx}(k)$ and $\sigma_\rho^2,\sigma_\theta^2$ denote the noise process standard deviations of rang and azimuth measurements and $\rho(k)$、$\theta(k)$ denote the rang and azimuth measurements. Assuming that all of the measurements have been associated to the track correctly, the initial setting of filter is given as follows Equation (40 and 41):

$$\begin{cases}\hat{x}(1|1)=z_1(1)\\ \hat{\dot{x}}(1|1)=[z_1(1)-z_1(0)]/T\\ \hat{y}(1|1)=z_2(1)\\ \hat{\dot{y}}(1|1)=[z_2(1)-z_2(0)]/T\end{cases} \quad (40)$$

$$P(1|1)=\begin{bmatrix}\sigma_x^2(1) & \sigma_x^2(1)/T & \sigma_{xy}(1) & \sigma_{xy}(1)/T\\ \sigma_x^2(1)/T & 2\sigma_x^2(1)/T^2 & \sigma_{xy}(1)/T & 2\sigma_{xy}(1)/T^2\\ \sigma_{yx}(1) & \sigma_{yx}(1)/T & \sigma_y^2(1) & \sigma_y^2(1)/T\\ \sigma_{yx}(1)/T & 2\sigma_{yx}(1)/T^2 & \sigma_y^2(1)/T & 2\sigma_y^2(1)/T^2\end{bmatrix} \quad (41)$$

1.9. System Flow Chart of Bi-threshold Algorithm

The System Flow Chart of the Dependent Bi-threshold Algorithm is given in **Fig. 2**.

1.10. Results and Analysis

With 1-run simulation, **Table 1 and 2** show Ec and Ee of Singer's, Bar-Shalom's and bi-threshold track correlation algorithms in case 1 and case2, respectively. **Figure 3-5** show the correct correlation ratio in case 1 and case 2, respectively. **Figure 4-6** show the error correlation ratio in case 1 and case 2, respectively. From these simulation results one can see that correlative performance of Bar-Shalom's algorithm is a little better than that of Singer's algorithm. Also one can see that correlative performance of the four bi-threshold track correlation algorithms proposed in this study is much better than that of Singer's and Bar-Shalom's algorithm, especially in the case 2 where there exists a heavy target density and a lot of maneuvering targets, where the improvement ratio of Ec reaches about 30 to 45% respectively. In addition, the correlation performances of independent bi-threshold algorithms are a litter better than that of dependent bi-threshold algorithm.

However, the L/R ruler must be set before the execution of these bi-threshold algorithms. With gradual growth of targets in simulation, **Figure 7-14** show the correlation result of different algorithms proposed in this study with 3/4 rules and 6/8 rulers respectively. One can see that the correlation performance of bi-threshold algorithms with 6/8 rules is a litter better than that of bithreshold algorithms with 3/4 rules from these simulation results.

Table 1. Ec and Ee of each algorithm in case 1 (L/R = 6/8)

	Ec						Ee						
L	Singre's algorithm	Bar-Singre's algorithm	IBTCA	DBTCA	IBTAC-ATS	DBTCA-ATS	Singre's algorithm	Bar-Singre's algorithm	IBTCA	DBTCA	IBTCA-ATS	DBTCA-ATS	N = 60 Nl
1	0.6667	0.7000	0.6667	0.7000	0.8950	0.8950	0.2667	0.1833	0.2667	0.1833	0.1013	0.1013	60
2	0.6780	0.7458	0.8305	0.8475	0.9210	0.9237	0.2712	0.1864	0.1695	0.1525	0.0753	0.0727	59
3	0.6780	0.7458	0.8644	0.8644	0.9430	0.9240	0.2712	0.1864	0.1356	0.1356	0.0533	0.0723	59
4	0.6780	0.7458	0.8983	0.8656	0.9523	0.9260	0.2712	0.2034	0.1017	0.1334	0.0440	0.0703	59
5	0.6780	0.7458	0.9322	0.8814	0.9613	0.9327	0.2712	0.2034	0.0678	0.1186	0.0350	0.0637	59
6	0.6667	0.7368	0.9298	0.8896	0.9707	0.9463	0.2807	0.2105	0.0702	0.1104	0.0257	0.0500	57
7	0.6545	0.7273	0.9455	0.8909	0.9770	0.9570	0.2909	0.2182	0.0545	0.1091	0.0193	0.0393	55
8	0.6415	0.7170	0.9434	0.8868	0.9770	0.9570	0.3019	0.2264	0.0566	0.1132	0.0193	0.0393	53
9	0.6415	0.7170	0.9434	0.9057	0.9770	0.9570	0.3019	0.2264	0.0566	0.0943	0.0193	0.0393	53
10	0.6257	0.7059	0.9412	0.9216	0.9770	0.9570	0.3137	0.2353	0.0588	0.0784	0.0193	0.0393	51
11	0.6257	0.7059	0.9412	0.9216	0.9770	0.9570	0.3137	0.2353	0.0588	0.0784	0.0193	0.0393	51
12	0.6257	0.7059	0.9412	0.9216	0.9770	0.9570	0.3137	0.2353	0.0588	0.0784	0.0193	0.0393	51

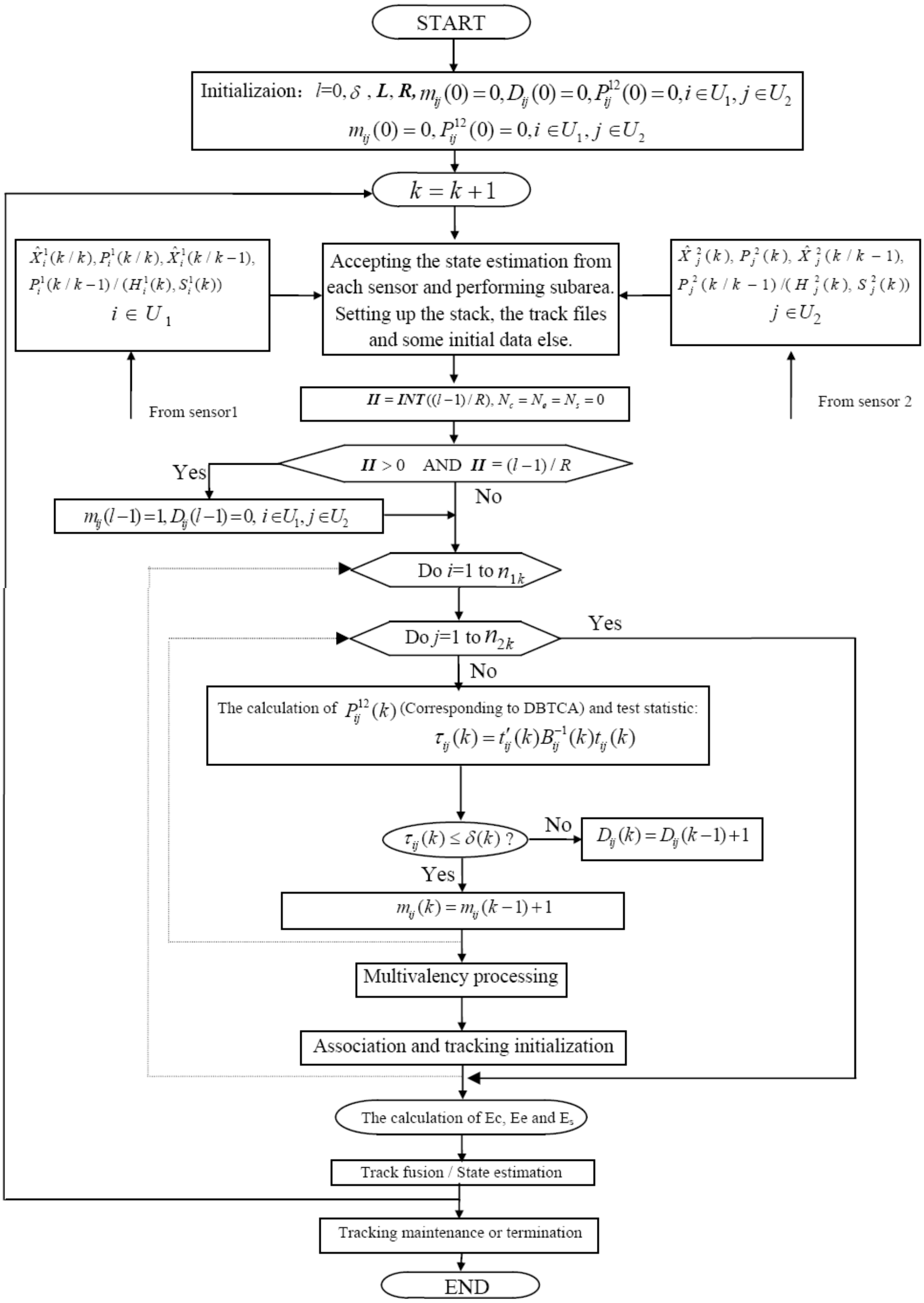

Fig. 2. The system flow chart of dependent bi-threshold track correlation algorithm (Notice: Ec, Ee and Es denote the correct, error and sans correlate ratio)

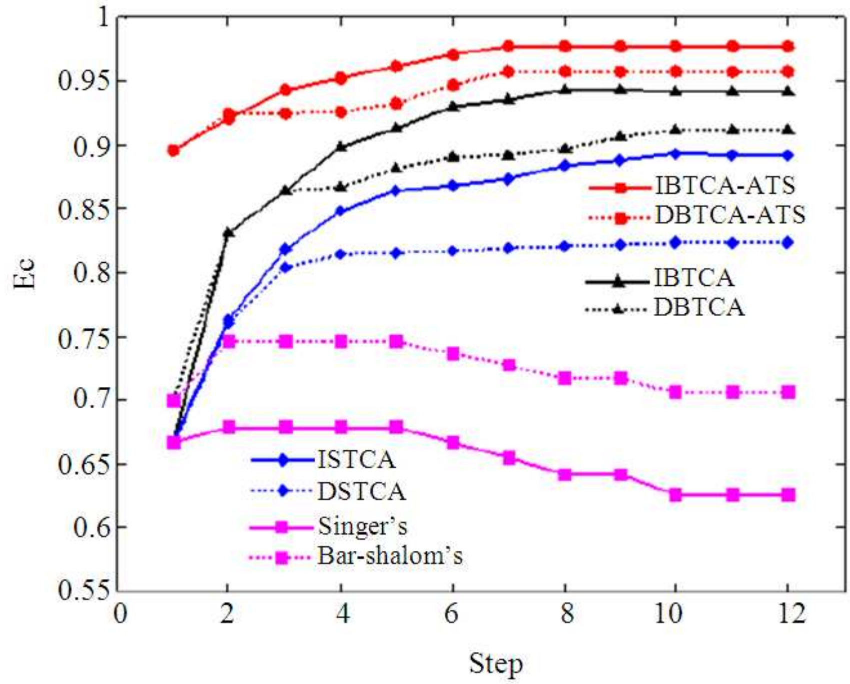

Fig. 3. Correct correlation ratio versus time (case1)

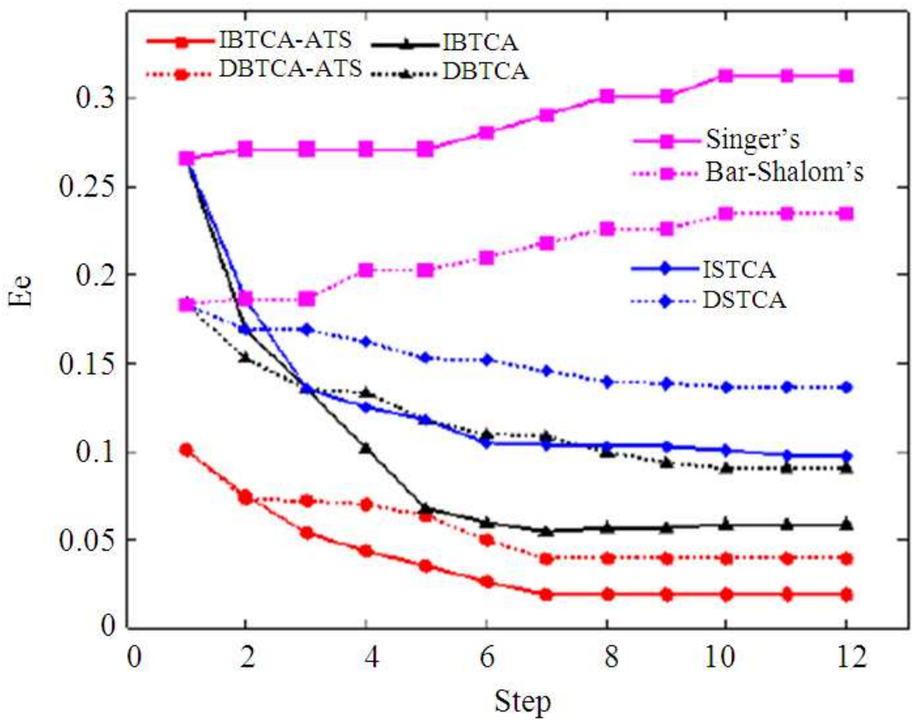

Fig. 4. Error correlation ratio versus time (case1)

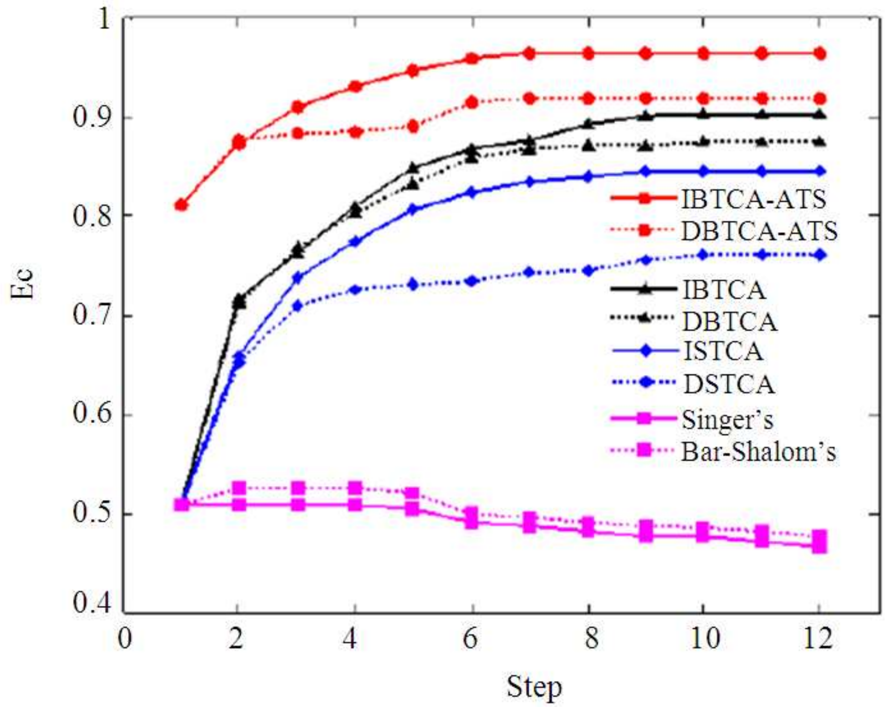

Fig. 5. Correct correlation ratioversus time (case2)

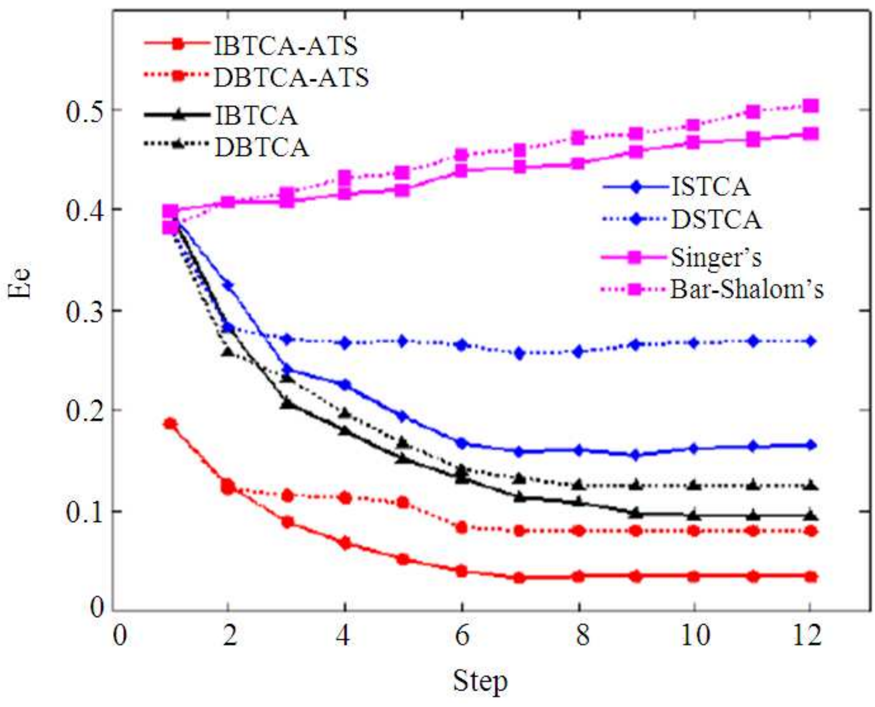

Fig. 6. Error correlation ratio versus time (case2)

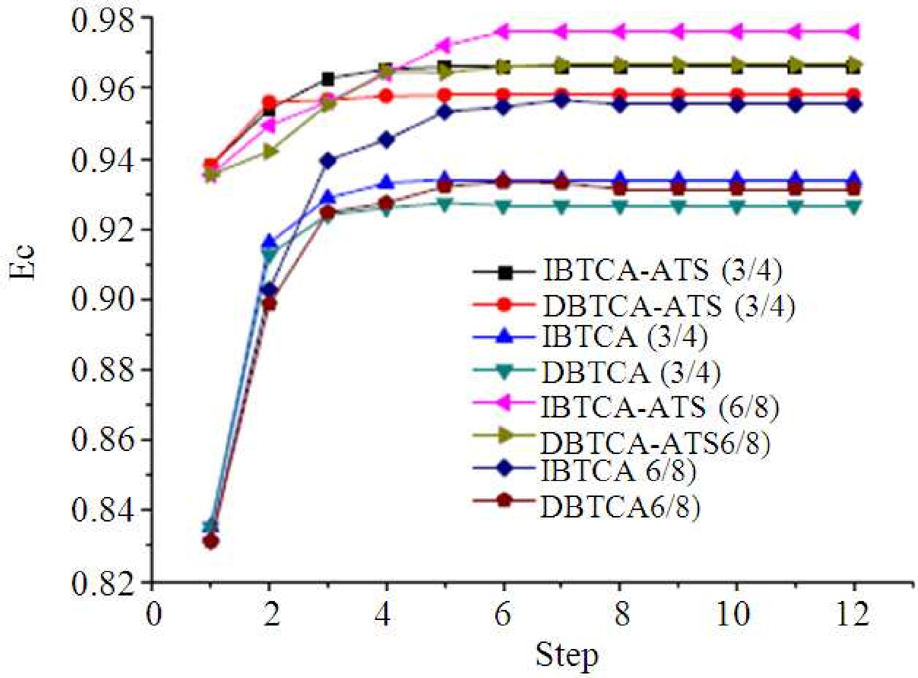

Fig. 7. Correct correlation ratio versus time (N = 30)

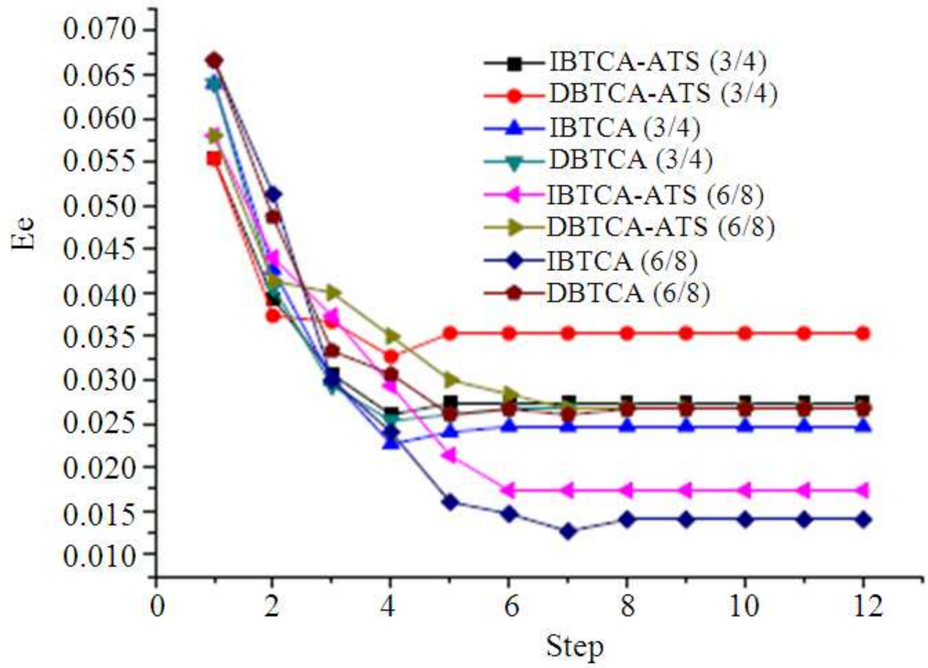

Fig. 8. Error correlation ratio versus time (N = 30)

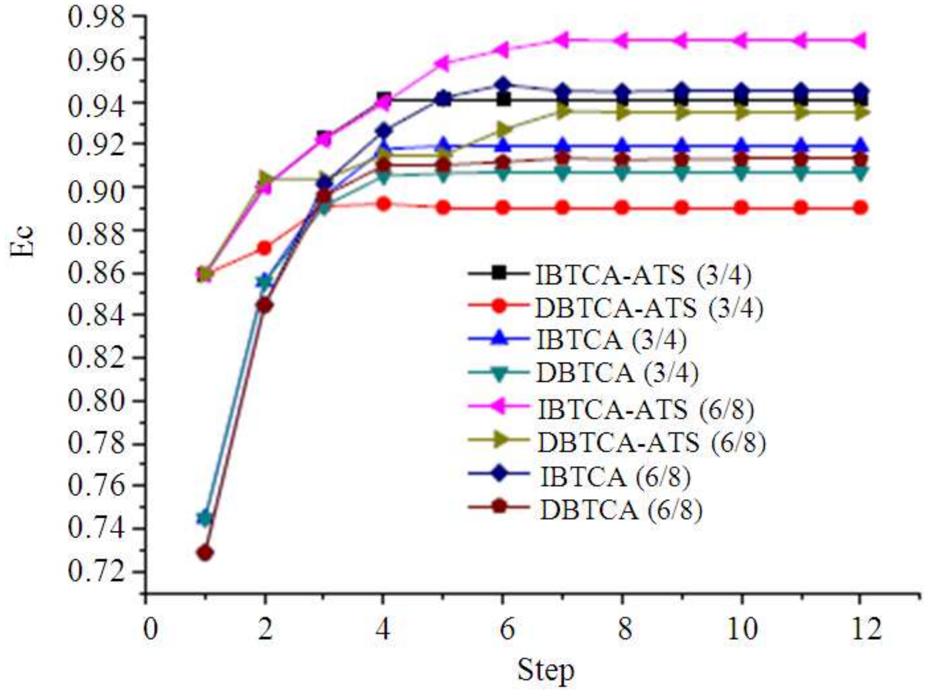

Fig. 9. Correct correlation ratio versus time (N = 90)

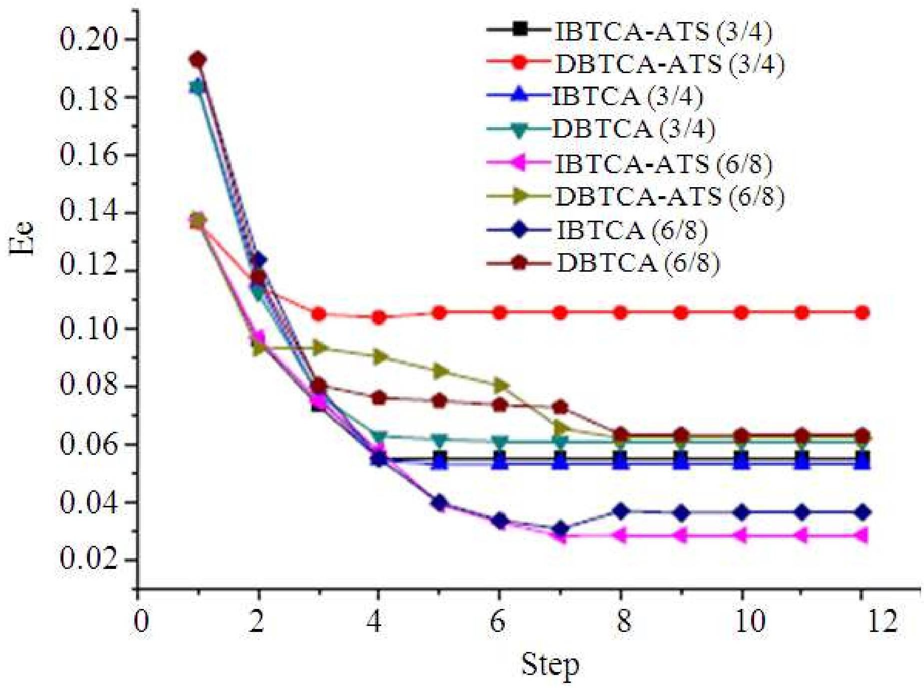

Fig. 10. Error correlation ratio versus time (N = 90)

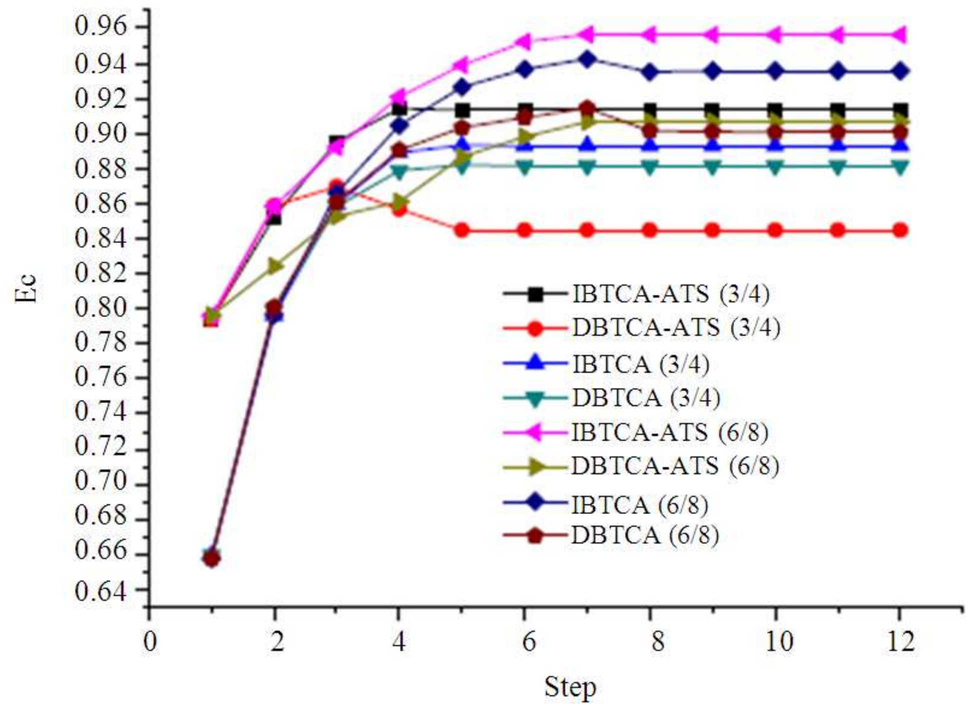

Fig. 11. Correct correlation ratio versus time (N = 150)

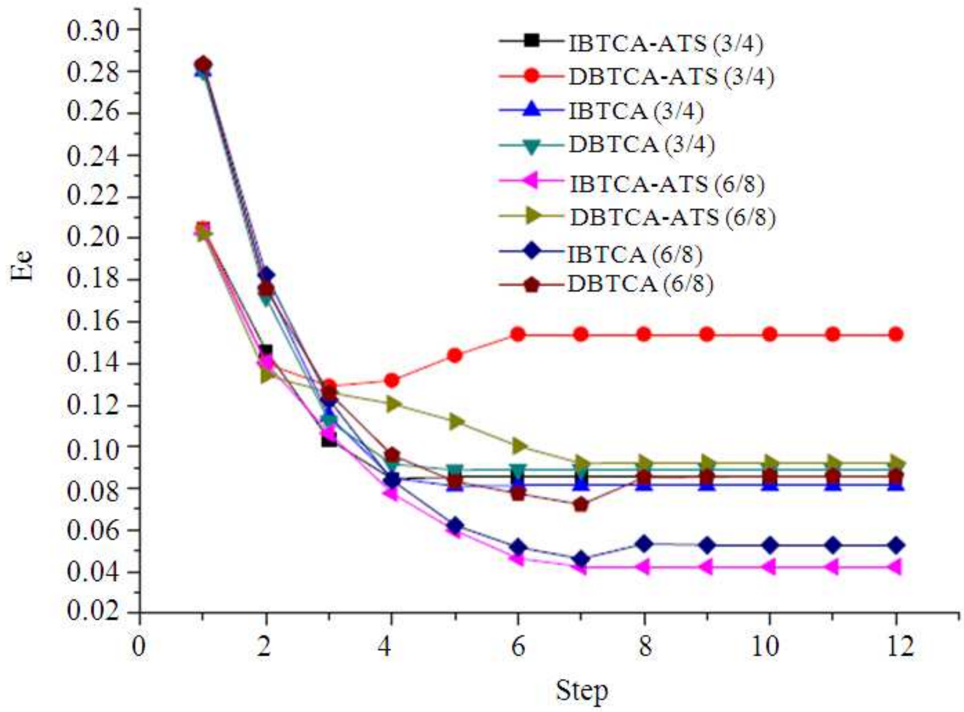

Fig. 12. Error correlation ratio versus time (N = 150)

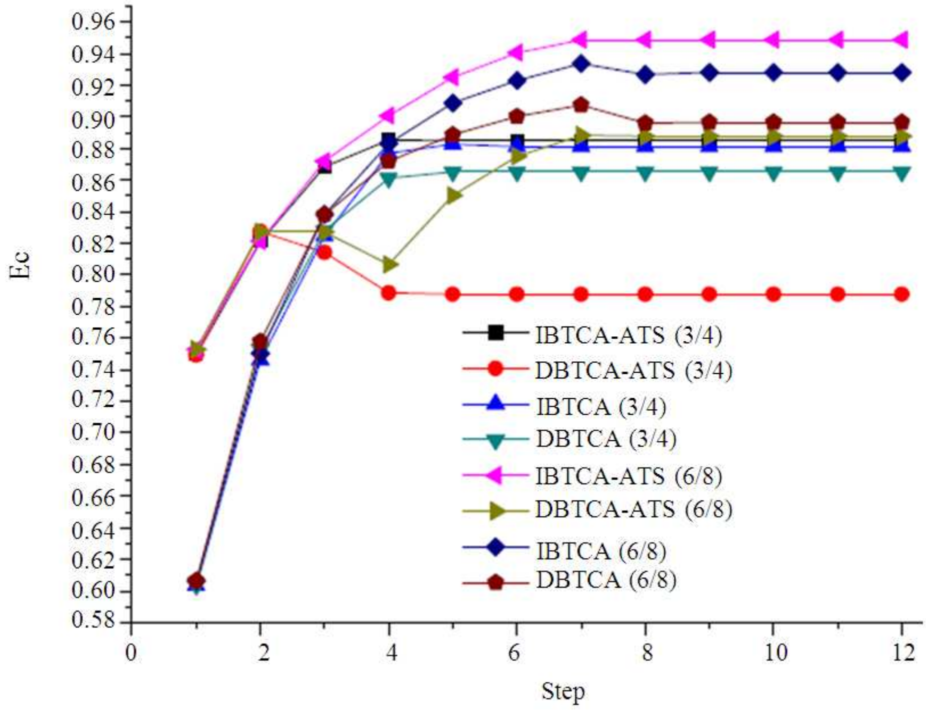

Fig. 13. Correct correlation ratio versus time (N = 210)

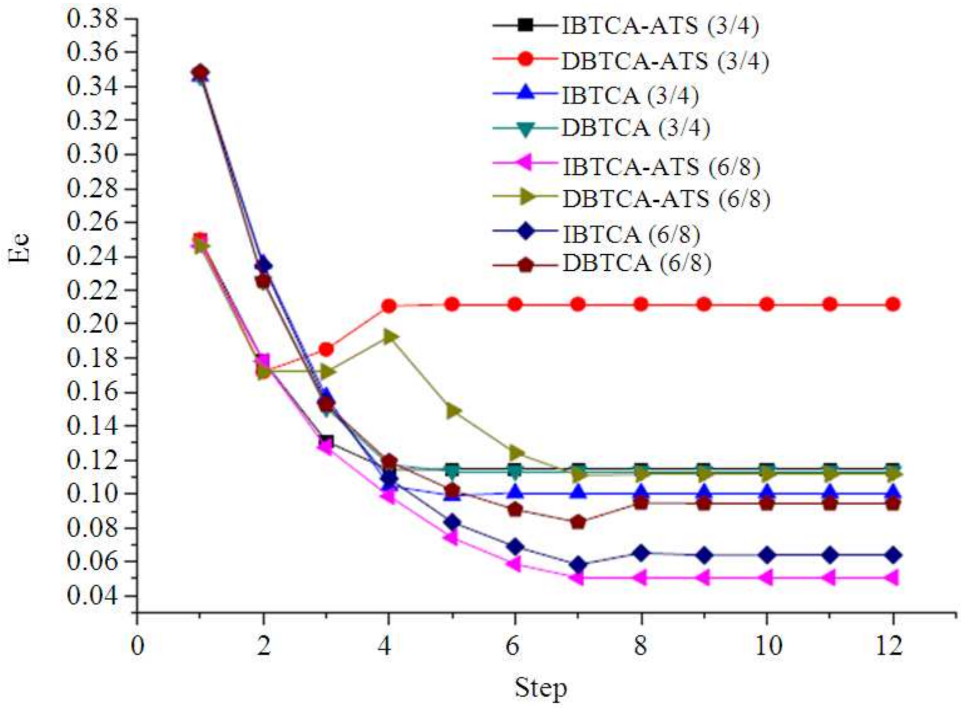

Fig. 14. Error correlation ratio versus time (N = 210)

Table 2. Ec and Ee of each algorithm in case 2(L/R = 6/8)

	Ec						Ee						
L	Singre's algorithm	Bar-Singre's algorithm	IBTCA	DBTCA	IBTAC-ATS	DBTCA-ATS	Singre's algorithm	Bar-Singre's algorithm	IBTCA	DBTCA	IBTCA-ATS	DBTCA-ATS	N = 120
1	0.5083	0.5083	0.5083	0.5083	0.8112	0.8112	0.4000	0.3833	0.4000	0.3833	0.1870	0.1870	120
2	0.5083	0.5250	0.7167	0.7417	0.8720	0.8762	0.4083	0.4083	0.2833	0.2583	0.1262	0.1220	120
3	0.5083	0.5250	0.7919	0.7667	0.9098	0.8827	0.4083	0.4167	0.2083	0.2333	0.0883	0.1155	120
4	0.5083	0.5250	0.8000	0.7833	0.9315	0.8843	0.4167	0.4333	0.2000	0.2167	0.0667	0.1138	120
5	0.5042	0.5210	0.8487	0.8319	0.9463	0.8905	0.4202	0.4370	0.1513	0.1681	0.0518	0.1077	119
6	0.4912	0.5000	0.8684	0.8596	0.9590	0.9153	0.4386	0.4561	0.1316	0.1404	0.0392	0.0828	114
7	0.4867	0.4956	0.8761	0.8584	0.9648	0.9185	0.4425	0.4602	0.1239	0.1416	0.0333	0.0797	113
8	0.4821	0.4911	0.8929	0.8750	0.9645	0.9177	0.4464	0.4732	0.1071	0.1250	0.0337	0.0805	112
9	0.4771	0.4862	0.9083	0.8716	0.9645	0.9177	0.4587	0.4771	0.0917	0.1284	0.0337	0.0805	109
10	0.4762	0.4857	0.9048	0.8857	0.9645	0.9177	0.4667	0.4857	0.0952	0.1143	0.0337	0.0805	105
11	0.4712	0.4808	0.9038	0.8750	0.9645	0.9177	0.4712	0.5000	0.0962	0.1250	0.0337	0.0805	104
12	0.4660	0.4757	0.9029	0.8738	0.9645	0.9177	0.4757	0.5049	0.0971	0.1262	0.0337	0.0805	103

Notice: Nl denotes the number of target in the common surveillance

2. CONCLUSION

Four bi-threshold track correlation algorithms are proposed and compared with the Singer's and Bar-Shalom's algorithm in this study. According to the simulation results, the difference between correlative performances of these algorithms is not so obvious when there are a few targets in surveillance and the difference between correlative performances of these algorithms will increase with the environments getting more complex. Therefore, the bithreshold algorithms present a better general correlative performance in dense multitarget environments, more cross, split and maneuvering track situations. To Singer's and Bar-Shalom's algorithm, the correlative performance of Bar-Shalom's algorithm is a little better than that of Singer's algorithm. To the bi-threshold algorithms, the correlative performance of independent algorithm is better than that of dependent algorithm and the bi-threshold algorithms with 6/8 6 rules has a better correlative performance than the bi-threshold algorithms with 3/4 rules. Though the simulations is performed in the case of correlating the data between congeneric sensors, these Four bi-threshold track correlative algorithms here can also resolve the problem of correlating tracks between heterogeneous sensors.

3. ACKNOWLEDGMENT

This study was financially supported by National Natural Science Foundation of China under grant number 61032001 and Shandong Provincial Natural Science Foundation of China under grant number ZR2011FQ002.

4. REFERENCES

Bar-Shalom, Y. and H. Chen, 2004. Multisensor track-to-track association for tracks with dependent errors. Proceedings of the 43rd IEEE Conference on Decision and Control, Dec. 14-17, IEEE Xplore Press, pp: 2674-2679. DOI: 10.1109/CDC.2004.1428864

Bar-Shalom, Y. and L. Campo, 1986. The effect of the common process noise on the two-sensor fused-track covariance. IEEE Trans. AES, 22: 803-805. DOI: 10.1109/TAES.1986.310815

Bar-Shalom, Y. and T.E. Fortmann, 1988. Tracking and Data Association. 1st Edn., Academic Press, New York, ISBN-10: 0120797607, pp: 353.

Bar-Shalom, Y., 2008. On the sequential track correlation algorithm in a multisensor data fusion system. IEEE Trans. Aerospace Electr. Syst., 44: 396-396. DOI: 10.1109/TAES.2008.4517016

Bowman, C.L., 1979. Maximum likelihood track correlation for multisensor integration. Proceedings of the 18th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes, Dec. 12-14, IEEE Xplore Press, pp: 374-376. DOI: 10.1109/CDC.1979.270200

Chang, C. and L.C. Youens, 1982. Measurement correlation for multiple sensor tracking in a dense target environment. IEEE Trans. Automatic Control, 27: 1250-1252. DOI: 10.1109/TAC.1982.1103107

Gul, E., 1994. On the track similarity test in track splitting algorithm. IEEE Trans. Aerospace Electr. Syst., 30: 604-606. DOI: 10.1109/7.272282

Kaplan, L.M., Y. Bar-Shalom and W. Blair, 2008. Assignment costs for multiple sensor track-to-track association. IEEE Trans. Aerospace Electr. Syst., 44: 655-677. DOI: 10.1109/TAES.2008.4560213

Kosoka, M., 1983. A track correlation algorithm for multisensor intergration. Proceedings of the IEEE/AIAA 5th Digital Avionics Systems Conference, (ASC' 83), pp: 1-8.

La Scala, B. and A. Farina, 2002. Effects of cross-covariance and resolution on track association. Proceedings of the 3rd International Conference on Information Fusion, Information Fusion, Jul. 10-13, IEEE Xplore Press, Paris, France, pp: 119-133. DOI: 10.1109/IFIC.2000.859854

Mori, S., C.Y. Chong and K.C. Chang, 2011. Performance prediction of feature aided track-to-track association. Proceedings of the 14th International Conference on Information Fusion, Jul. 5-8, IEEE Xplore Press, Chicago, IL., pp: 1-8.

Osbome, R.W., Y. Bar-Shalom and P. Willett, 2011. Track-to-track association with augmented state. Proceedings of the 14th International Conference on Information Fusion, Jul. 5-8, IEEE Xplore Press, Chicago, IL., pp: 1-8.

Singer, R.A. and A.T. Kanyuck, 1971. Computer control of multiple site track correlation. Automatica, 7: 455-463. DOI: 10.1016/0005-1098(71)90096-3

Tian, X. and Y. Bar-Shalom, 2011. Sliding window test Vs. single time test for track-to-track association. Proceedings of the 11th International Conference on Information Fusion, Jun. 30-Jul. 3, IEEE Xplore Press, Cologne, pp: 1-8.

Wang, G., X. Zhang and S. Tan, 2012. Effect of biased estimation on radar-to-ESM track association. J. Syst. Eng. Electr., 23: 188-194. DOI: 10.1109/JSEE.2012.00024

You, H., L. Dajin and P. Yingning, 1996. A survey of multisensor data fusion algorithms. Fire Control Command Control, 21: 12-21.

You, H., T. Qinghai and J. Rongrong, 1989. Track correlation algorithms in multisensor system. Fire Control Command Control, 14: 1-12.