

EVALUATION OF DATA ASSOCIATION AND FUSION ALGORITHMS FOR TRACKING IN THE PRESENCE OF MEASUREMENT LOSS

V. P. S. Naidu, Girija G. and J. R. Raol
Scientists, Flight Mechanics and Control Division,
National Aerospace Laboratories, Bangalore – 560 017, INDIA.

ABSTRACT

Tracking in multi sensor multi target (MSMT) scenario is a complex problem due to the uncertainties in the origin of observations. Solution to this problem requires appropriate gating and data association procedures to associate measurements with targets. A PC MATLAB program based on track-oriented approach is evaluated which uses nearest neighbor Kalman filter (NNKF) and probabilistic data association filter (PDAF) for tracking multiple targets from data of multiple sensors. For track-to-track fusion, state vector fusion philosophy is employed. The tracking performance in the presence of simulated track loss and recovery as well as in clutter is evaluated. During data loss PDAF performed better than NNKF. In the presence of mild clutter and sparse target scenarios, the NNKF and PDAF give similar performance.

1. INTRODUCTION

Tracking comprises of estimation of the current state of a target based on uncertain measurements selected according to a certain rule as sharing a common origin and calculation of the accuracy and credibility associated with the state estimate. The problem is complex even for single target tracking because of target model uncertainties and measurement uncertainties. The complexity of the tracking problem increases further when multiple targets are to be tracked from measurements of multiple sensors.

Data association i.e. to determine from which target, if any, a particular measurement originated, is the central problem in multi sensor multi target tracking¹. The problem is complex due to uncertain data and disparate data sources. The identity of the targets responsible for each individual data set is unknown, so there is uncertainty as how to associate data from one sensor which are obtained at one time and location to those of another sensor at another point in time and location. Also, false alarms and the clutter detections may be present which are not easily distinguishable from the true target measurements. In addition, one may have to

deal with measurement loss in some of the tracking sensors.

Gating and data association enable tracking in multi sensor multi target (MSMT) scenario. Gating helps in deciding if an observation (which includes clutter, false alarms and electronic counter measures) is a probable candidate for track maintenance or track update. Data association is the step to associate the measurements to the targets with certainty when several targets are in the same neighborhood. Two approaches to data association are possible: i) using the nearest neighbor (NN) approach in which a unique pairing is determined so that at most one observation can be paired with a previously established track. The method is based upon likelihood theory and the goal is to minimize an overall distance function that considers all observation-to-track pairings that satisfy a preliminary gating test, ii) decision is achieved using probabilistic data association PDA algorithm in which a track is updated by a weighted sum of innovations from multiple validated measurements.

For handling the problem of tracking in a MSMT scenario, a program based on gating and data association using both NNKF and PDAF² approach has been developed in PC MATLAB. This program is primarily an adapted version of software package of Ref¹ and is updated/modified for the present application. The main features of FUSEDAT and the upgraded MSMT packages are shown in Table-1. The steps in the MSMT program for multi-sensor multi-target tracking and data association are shown in Fig-1. In this paper, details of the algorithms, the steps in the development of the program and results of tracking for data from multiple sensors when there is measurement loss are presented. The test scenario considered for validating the program are i) data of three targets launched from different sites and nine sensors located at different locations tracking the targets. Three sensors are configured to track one target. In addition to the estimated target track position at the end of each scan, the program generates information on the target-sensor lock status. The performance has been evaluated by adding clutter to the data and simulating data loss in one or more of the tracking sensors for a short period.

Results are presented in terms of track scores, innovations of the filters with theoretical bounds and computed χ^2 distance values, ii) The situation where each of the three sensors looks at six targets and then all the three sensor-results are fused, where there could be some data loss.

2. DATA ASSOCIATION AND TRACKING ALGORITHMS

2.1 NN Kalman Filter²

In NNKF, at any instant of time, the measurement that is nearest to the track is chosen for updating the track. It is to be noted that each measurement can only be associated with one track and no two tracks could share the same measurement. If valid measurement exists, the track is updated using NN Kalman filter. The time propagation follows the standard Kalman filter equations:

$$\tilde{X}(k/k-1) = \Phi\hat{X}(k-1/k-1) \quad (1)$$

$$\tilde{P}(k/k-1) = \Phi\hat{P}(k-1/k-1)\Phi^T + GQG^T \quad (2)$$

The state estimate is updated using:

$$\hat{X}(k/k) = \tilde{X}(k/k-1) + Kv(k) \quad (3)$$

$$\hat{P}(k/k) = (I - KH)\tilde{P}(k/k-1)$$

The Kalman gain $K = \tilde{P}(k/k-1)H^T S^{-1}$

$$\text{Residual vector } v(k) = z(k) - \tilde{z}(k/k-1) \quad (4)$$

Residual covariance $S = H\tilde{P}(k/k-1)H^T + R$ where $z(k)$ is the measurement vector and $\tilde{z}(k/k-1)$ is the predicted value at scan k , the H is the measurement matrix and R is the measurement error covariance matrix given by $R = \text{diag}[\sigma_x^2 \quad \sigma_y^2 \quad \sigma_z^2]$ - for the case where three observables x, y, z are considered.

If there is no valid measurement, the track retains the extrapolated value:

$$\hat{X}(k/k) = \tilde{X}(k/k-1) \quad (5)$$

and $\hat{P}(k/k) = \tilde{P}(k/k-1)$

2.2 Probabilistic Data Association Filter²

The PDAF algorithm calculates the association probabilities for each valid measurement at the current time to the target of interest. This probabilistic information is used in a tracking filter (PDAF) that accounts for the measurement origin uncertainty. If there are m measurements falling within the gate and it is assumed that there is only one target of interest and track has been initialized², the association events

$z_i = \{ y_i \text{ is the target originated measurement} \},$
 $i=1,2,\dots,m,$

$\{ \text{none of the measurements is target originated} \}, i=0$ (6)

are mutually exclusive and exhaustive for $m \geq 1$. The conditional mean of the state can be written as

$$\hat{X}(k/k) = \sum_{i=0}^m \hat{X}_i(k/k)p_i \quad (7)$$

where $\hat{X}_i(k/k)$ is the updated state conditioned on the event that the i^{th} validated measurement is correct and p_i is the conditional probability of this event. The estimate conditioned on measurement 'i' being correct is given by

$$\hat{X}_i(k/k) = \tilde{X}(k/k-1) + Kv_i(k), \quad i=1,2,\dots,m \quad (8)$$

the conditional innovation is given by

$$v_i(k) = z_i(k) - \tilde{z}(k/k-1) \quad (9)$$

The gain K is the same as in Kalman filter eqs.(4). For $i=0$, i.e. if none of the measurements is valid ($m=0$), then

$$\hat{X}_0(k/k) = \tilde{X}(k/k-1) \quad (10)$$

Combining the equations (8), (10) & (7) yield, the state update equation of the PDAF

$$\hat{X}(k/k) = \tilde{X}(k/k-1) + Kv(k) \quad (11)$$

The combined innovation is given by

$$v(k) = \sum_{i=1}^m p_i(k)v_i(k) \quad (12)$$

The covariance associated with the updated state is

$$\hat{P}(k/k) = p_0(k)\tilde{P}(k/k-1) + (1-p_0(k))P^c(k/k) + P^s(k) \quad (13)$$

where the covariance of the state updated with correct measurement

$$P^c(k/k) = \tilde{P}(k/k-1) - KSK^T \quad (14)$$

and the spread of the innovations

$$P^s(k/k) = K \left(\sum_{i=1}^m p_i(k)v_i(k)v_i(k)^T - v(k)v(k)^T \right) K^T \quad (15)$$

The conditional probability is calculated using Poisson clutter model²

$$p_i(k) = \frac{e^{-0.5v_i^T S^{-1}v_i}}{\lambda\sqrt{|2\Pi S|} \frac{(1-P_D)}{P_D} + \sum_{j=1}^m e^{-0.5v_j^T S^{-1}v_j}}, \quad i=1,2,\dots,m$$

$$= \frac{\lambda\sqrt{|2\Pi S|} \frac{(1-P_D)}{P_D}}{\lambda\sqrt{|2\Pi S|} \frac{(1-P_D)}{P_D} + \sum_{j=1}^m e^{-0.5v_j^T S^{-1}v_j}}, \quad i=0 \quad (16)$$

where λ =false alarm probability.

P_D = Detection probability.

The information flow in PDAF algorithm is shown in Fig-3. The features of these algorithms are given in Table-2

3. PROGRAM FOR TRACKING AND DATA ASSOCIATION ALGORITHMS FOR MSMT

Two commonly used approaches for multi target tracking are ‘target oriented’ and ‘track oriented’ approaches. In the target-oriented approach, the number of targets is assumed to be known and all data association hypotheses are combined into one for each target. The track oriented approach treats each track individually while it is initiated, updated and terminated based on the associated measurement history. Track oriented approach is pursued for the application in this paper (since the other approach cannot handle track initiation and can only handle track continuation). In the track-oriented algorithm, a score is assigned to each track and is updated according to the association history. A track is initiated based on a single measurement, and will be eliminated when the score is below a predetermined threshold. A brief description of each of the steps in the program is given below.

a). Sensor attributes including sensor location, resolution, field of view (FOV), Detection probability (PD) and False alarm probability (Pfa) are provided.

Using Pfa the number of false alarms is calculated using $Nfa = Pfa * \mu FOV$ (17)

where Nfa is the expected number of false alarms and μFOV is the volume of FOV.

b). New data set: The measurements acquired from the sensors are converted to a common reference point in a Cartesian coordinate frame using:

$$\mathbf{x}_{ref} = \mathbf{x}_{traj} - \mathbf{x}_{loc},$$

$$\mathbf{y}_{ref} = \mathbf{y}_{traj} - \mathbf{y}_{loc},$$

$$\mathbf{z}_{ref} = \mathbf{z}_{traj} - \mathbf{z}_{loc}.$$

where \mathbf{x}_{ref} , \mathbf{y}_{ref} and \mathbf{z}_{ref} are \mathbf{x} , \mathbf{y} and \mathbf{z} co-ordinates of target w.r.t common reference, \mathbf{x}_{loc} , \mathbf{y}_{loc} and \mathbf{z}_{loc} are \mathbf{x} , \mathbf{y} and \mathbf{z} co-ordinates of corresponding sensor location, \mathbf{x}_{traj} , \mathbf{y}_{traj} and \mathbf{z}_{traj} are \mathbf{x} , \mathbf{y} and \mathbf{z} co-ordinates of target trajectory measured by the sensors.

c) Gating: Gating is performed to eliminate unlikely measurement-to-track pairs. Assuming that the measurement vector is of dimension m , a distance d^2 (normalized distance) representing the norm of the residual vector is computed using

$$d^2 = \mathbf{v}^T \mathbf{S}^{-1} \mathbf{v} \quad (18)$$

For example, consider two tracks ($y_i(k-1)$, $i=1,2$) at scan $(k-1)$. At scan k , as shown in Fig-4, if four measurements $z_j(k)$, $j=1,2,3,4$ are available, then the

track to measurement distance d_{ij} (from i^{th} track to j^{th} measurement) for each of the predicted tracks ($y_i(k-1)$, $i=1,2$) is computed using (18). A correlation between the measurement and track is allowed if the distance $d^2 \leq G$, where G is the χ^2 threshold. The χ^2 threshold is obtained from the tables of chi-square distribution since the validation region is chi-square distributed with number of degree of freedom equal to the dimension of the measurement². For those measurements that fall within the gate, the likelihood value computed using $\log(|2\pi\mathbf{S}|) + d^2$ is entered in the correlation matrix (called Track to Measurement Correlation matrix - TMCR) formed with the measurements along the rows and tracks along the columns. For those measurements that fall outside the gate, a high value is entered in the TMCR matrix (see Table-3).

d). Measurement to track association & track updation: When NNKF is used for tracking, the measurement that is nearest to the track is chosen for updating the track. Once the particular measurement-to-track association pair is chosen from the correlation matrix for updating track, both will be removed from the matrix and next track with the least association uncertainty will be processed. In the present example (Fig-4), measurements $z_1(k)$ and $z_3(k)$ fall within the gate region of predicted track $y_1(k)$, $z_2(k)$ falls within the gate region of predicted track $y_2(k)$ and $z_4(k)$ falls outside of both $y_1(k)$ and $y_2(k)$ gate regions as shown in Table-3. The measurement $z_1(k)$ is taken for updating the track $y_1(k)$, because it is nearer than $z_3(k)$.

In cases where PDAF is used for tracking, all measurements falling within the gate, formed around the extrapolated track and their associated probabilities, are used for track updating. In present example, the measurements $z_1(k)$ and $z_3(k)$ are taken for updating track $y_1(k)$ and $z_2(k)$ is taken for updating the track $y_2(k)$. This process continues until all tracks are considered. Measurement that has not been assigned to any track will be used to initiate a new track. A score is obtained for each track based on the association history and is used in the decision of eliminating or confirming tracks.

e). Track initiation: A new track is initiated with a measurement that is not associated with any existing track. A score is assigned to each initiated new track. A track is initiated by three position measurements ($\mathbf{x}, \mathbf{y}, \mathbf{z}$) and the velocity vector. The initial score for new track is calculated using

$$p = \frac{\beta_{NT}}{\beta_{NT} + \beta_{fa}} \quad (19)$$

where β_{NT} = expected number of true targets and β_{fa} = expected number of false alarms per unit surveillance volume per scan. In the present example, $z_4(k)$ is used for track initiation.

f). Track Extrapolation: It is possible that a track may not have any validated measurement, in which case the track will not be updated but existing tracks are just extrapolated for processing at next scan.

g). Extrapolate tracks into next sensor FOV¹: The surviving tracks in current sensor FOV are taken into next sensor FOV, because it is assumed that in MSMT scenario all sensors are tracking all targets. Also, the track score is propagated to the next sensor FOV using the Markov chain transition matrix. In computing the scoring function, two models are used one for ‘observable target’ (true track) designated as Model O and one for ‘unobservable target’ (a target outside the sensor coverage or erroneously hypothesized target) designated as model U¹. For both models, target measurements (with detection probability P_D) as well as clutter is to be considered. $P_D = 0$ for model U. The models O and U are given by a Markov chain assuming the following transition probabilities¹:

$$\begin{aligned} P(M_O|\bar{M}_O) &= 1 - \varepsilon_O, & P(M_U|\bar{M}_O) &= \varepsilon_O \\ P(M_U|\bar{M}_U) &= 1 - \varepsilon_U, & P(M_O|\bar{M}_U) &= \varepsilon_U \end{aligned} \quad (20)$$

where M_x denotes the event that model x is in effect during the current sampling interval and \bar{M}_x for the previous interval. Eq. (20) indicates that the transition between the models is assumed with low probabilities. The exact values of ε_A and ε_D are to be chosen based on the scenario under consideration.

h). Extrapolate tracks into the next scan: The surviving tracks are extrapolated for processing at next scan using target dynamic model. The target dynamic model is as follows:

$$X(k+1) = FX(k) + Gw(k) \quad (21)$$

where the target dynamic state transition matrix

$$F = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and the state vector is given by $X(k) = [x(k) \ y(k) \ z(k) \ \dot{x}(k) \ \dot{y}(k) \ \dot{z}(k)]^T$

The system noise covariance matrix

$$G = \begin{bmatrix} \frac{\Delta t^2}{2} & 0 & 0 \\ 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & 0 & \frac{\Delta t^2}{2} \\ \Delta t & 0 & 0 \\ 0 & \Delta t & 0 \\ 0 & 0 & \Delta t \end{bmatrix}$$

and $w(k)$ is assumed to be a zero-mean white Gaussian process noise with covariance $E[w(k)w(k)^T] = Q(k)$ and Δt is the sampling interval. The extrapolation is done using the Kalman filter eq(5).

i). Track management: Many tracks could be initiated in a clutter environment. Scoring threshold is used to eliminate the false tracks. The scoring threshold is one of the system design parameters and it should be adjusted based on the scenario and performance requirement. Similar tracks are fused to avoid redundant tracks. In general, the **direction** of tracks has to be considered while combining similar tracks. An N_D -scan approach is recommended in literature³ wherein tracks that have the last N_D observations in common, are combined together. Depending on the value of N_D , this approach would automatically take the velocity as well as acceleration into account for combining similar tracks, e.g. $x(2)$ - $x(1)$ can be regarded as velocity, etc. A 3-scan approach has been incorporated into the program for combining the tracks. Consider two tracks whose state vector estimates and covariance matrices are given at scan k:

$$\begin{aligned} \text{track i : } & \hat{X}_i(k/k), \hat{P}_i(k/k) \\ \text{track j : } & \hat{X}_j(k/k), \hat{P}_j(k/k) \end{aligned} \quad (22)$$

Combined state vector:

$$X_c(k) = \hat{X}_i(k/k) + \hat{P}_i(k/k) \hat{P}(k)_{ij}^{-1} [\hat{X}_j(k/k) - \hat{X}_i(k/k)] \quad (23)$$

Combined covariance matrix:

$$P_c(k) = \hat{P}_i(k/k) - \hat{P}_i(k/k) \hat{P}(k)_{ij}^{-1} \hat{P}_i(k/k) \quad (24)$$

$$\text{where } \hat{P}(k)_{ij} = \hat{P}_i(k/k) + \hat{P}_j(k/k) \quad (25)$$

The logic developed finally generates the information regarding the surviving tracks and sensors to target lock status.

j). Graphical display: This module displays the true trajectory and measurements and also performance measures such as true & false track detections, number of good and false tracks, good and false track probabilities and also the sensor and target lock status at each instant of time.

4. PERFORMANCE EVALUATION

The performance of the NNKF and PDAF is checked by computing:

i). The percentage fit error (PFE) in x, y and z positions

$$PFE = 100 * \frac{\text{norm}(x - \hat{x})}{\text{norm}(x)} \quad (26)$$

where, \mathbf{x} is the true x-position data, $\hat{\mathbf{x}}$ is the estimated x-position data

ii). The root mean square position error

$$RMSPE = \sqrt{\frac{1}{N} \sum_{i=1}^N \frac{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (z_i - \hat{z}_i)^2}{3}} \quad (27)$$

iii). The root sum square position error

$$RSSPE = \sqrt{(x - \hat{x})^2 + (y - \hat{y})^2 + (z - \hat{z})^2} \quad (28)$$

iv). Singer-Kanyuck track association metric

$$C_{ij} = \left\| \hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j \right\|_{(P_i + P_j)^{-1}}^2$$

$$C_{ij} = (\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j)^T (P_i + P_j)^{-1} (\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j) \quad (29)$$

The metric⁴ C_{ij} can be viewed as the square of the (normalized) distance between two Gaussian distributions with mean vectors $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ and a common covariance matrix $P_i + P_j$.

v). Percentage root mean square position error

$$\%RMSPE = \frac{RMSPE}{\sqrt{\frac{1}{N} \sum_{i=1}^N \frac{x_i^2 + y_i^2 + z_i^2}{3}}} * 100 \quad (30)$$

5. RESULTS AND DISCUSSIONS

The interactive program for MSMT data association and tracking is used to identify which of the sensors in the MSMT scenario are tracking same targets using the scenario of nine sensors located at different points in space and their measurements. Fig-5 shows the trajectories as seen from the 9 sensors. At each scan, the program displays the target identification (Id) and the sensors, which are tracking that particular target on the screen. It is found that initially 9 tracks survive before similar tracks are combined using a predetermined distance threshold. After this combination, it is seen that only 3 tracks survive and they have been assigned three target Id numbers (T1, T2 and T3). The sensors, which track a particular target, are shown in Table-4 from which it is clear that three sensors track one target.

Track loss is simulated in data from sensors 1-3 during 100 to 150 secs. Fig-6 shows the data with simulated clutter ($Pfa = 10^{-15}$) added to the sensor data. It is clear

from the Table-5 that the performance of the two data association algorithms in the presence of clutter for this scenario is almost identical. The comparison of true tracks and estimated tracks with NNKF is shown in Fig-7. The Fig-8 shows the track score, the innovations with bounds and the χ^2 distance measure on the X-axis data for target/track-1 (indicated as T1X in Fig-8) where there is data loss and for target/track-2 (indicated as T2X in Fig-8) where there is no data loss. The track score is zero during the measurement data loss, innovations are within the theoretical bounds and the χ^2 distance values at each scan are below the threshold values obtained from the χ^2 tables. Fig-9 and Fig-10 show the RSSPE in track-1 without and with data loss respectively. The RSSPE is very large during the data loss segment as shown in Fig-10. The PFE and %RMSPE when there is a data loss in track-1 are shown in Table-6. It is observed from the table that the PFE and %RMSPE increase as the duration of data loss increases. The Singer-Kanyuck association metric for i^{th} track and j^{th} track from the same target are almost zero, which means that the association is feasible. The association metric for i^{th} track and j^{th} track from the different targets are shown in Fig-11. The metric is large, which means that the association is infeasible. It is seen from Fig-12 and Fig-13 that the performance of PDAF is better than that of NNKF in presence of data loss. The data loss for longer time may be acceptable if PDAF is used since it gives lower PFE and RMSPE. The Fig-14 & 15 show the results of data fusion of 3-sensors and 6-targets and associated performance aspects like track probability, good tracks etc, with 20 Monte-Carlo simulation runs.

6. CONCLUDING REMARKS

A PC MATLAB program based on track-oriented approach has been evaluated NNKF and PDAF for tracking multiple targets from data of multiple sensors. The performance in the presence of simulated track loss and recovery as well as in clutter is evaluated. During data loss PDAF performed better than NNKF. In the presence of mild clutter and sparse target scenarios, the NNKF and PDAF give similar performance.

ACKNOWLEDGEMENTS: Technical discussions with Mr. Appavu Raj, Scientist, ITR and Mr. Sudesh Kashyap, Scientist, FMCD are acknowledged.

REFERENCES:

1. K.C. Chang and Y. Bar-Shalom. FUSEDAT: a software package for fusion and data association and tracking with multiple sensors. In Proc. SPIE, Vol. 2235, April 1994.

2. Y. Bar Shalom. Multi target multi sensor tracking: Principles and Techniques. Academic Press, 1990.
3. Samuel S. Blackman, Multiple-Target Tracking with Radar Applications, Artech House, INC, Norwood, 1986.
4. Shozo Mori, William H.Barker, Chee-Yee Chong and Kuo-Chu Chang, "Track Association and Track Fusion with Nondeterministic Target Dynamics", IEEE Trans. On Aero. And Elec. Sys., Vol. 38, No.2, April-2002.

Table-1 Features of two packages

S.No	FUSEDAT (obtained by NAL)	Modified MSMT
1	Simulated data w.r.t common reference point is used	(*) Simulated/Real data w.r.t sensor location is used and converted to common reference point in Cartesian coordinate frame.
2	NNKF/PDAF	NNKF and PDAF
3	Similar tracks are combined	Similar tracks are combined using distance threshold and track-to-track fusion
4	Direction of two tracks not included while tracks are combined	(*) ND-scan approach (ND =3) used to combine similar tracks while direction of tracks taken into account. Depending on the value of ND, this approach would automatically take the velocity and acceleration into account for combining similar tracks, e.q. x(2)-x(1) can be regarded as velocity, etc.
5	Performance metrics used	(*) Additional metrics: S-K, %RMSPE and RSSPE
6	-----	(*) Data loss feature included (measurements are removed from the data set for a fixed duration)
7	Target/track oriented approach	Track oriented approach is used
8	Clutter is added	Clutter is added

* Some additional features

Table-2 Features of NNKF/PDAF

Feature	NNKF	PDAF
Filter	Linear Kalman filter.	Linear Kalman filter.
State updation	Measurement nearest to the predicted measurement in validation region	Association probabilities for each measurement lying in the validation region
Possibility of Track loss	Moderately high	Less
False track adaptation	Moderately high	Less
During data loss	Degradation due to some uncertainty in estimation of previous state	Better performance due to better estimation of previous states
Computational time	Low	High (≈ 1.5 KF)
Tracking capability	Less reliable in clutter environment	Reliable in clutter environment

Table-3 TMCR table for two tracks (i=1,2) and four measurements (j=1,2,3,4) at scan k (Fig-4)

Track \ Measurement	$y_1(k)$	$y_2(k)$
$z_1(k)$	d_{11}	1000
$z_2(k)$	1000	d_{22}
$z_3(k)$	d_{13}	1000
$z_4(k)$	1000	1000

Table-4 Target and corresponding tracking sensor identification (Id) numbers

Target number	Sensor Id
T1	S1, S2, S3
T2	S4, S5, S6
T3	S7, S8, S9

Table-5 Percentage fit error in track positions (Figs-5 to 8)

Track No.	NNKF		PDAF	
	PFE in x	PFE in y	PFE in x	PFE in y
Track 1	0.0604	0.0557	0.081	0.075
Track 2	1.0398	1.0491	1.0397	1.0490
Track 3	0.0522	0.0283	0.0523	0.0284

Table-6 PFE and %RMSPE, when there is data loss in track-1 (distance in meters)

Data loss	NNKF			PDAF		
	PFE		%RMSPE	PFE		%RMSPE
	x	y		x	y	
0 sec	0.06	0.05	0.0669	0.081	0.075	0.0559
5sec	1.32	1.4	1.369	0.083	0.078	0.0593
10sec	2.04	2.18	2.13	0.11	0.102	0.1067
20sec	3.62	3.87	3.77	0.48	0.448	0.4632
30sec	5.65	6.0	5.87	1.32	1.22	1.2628

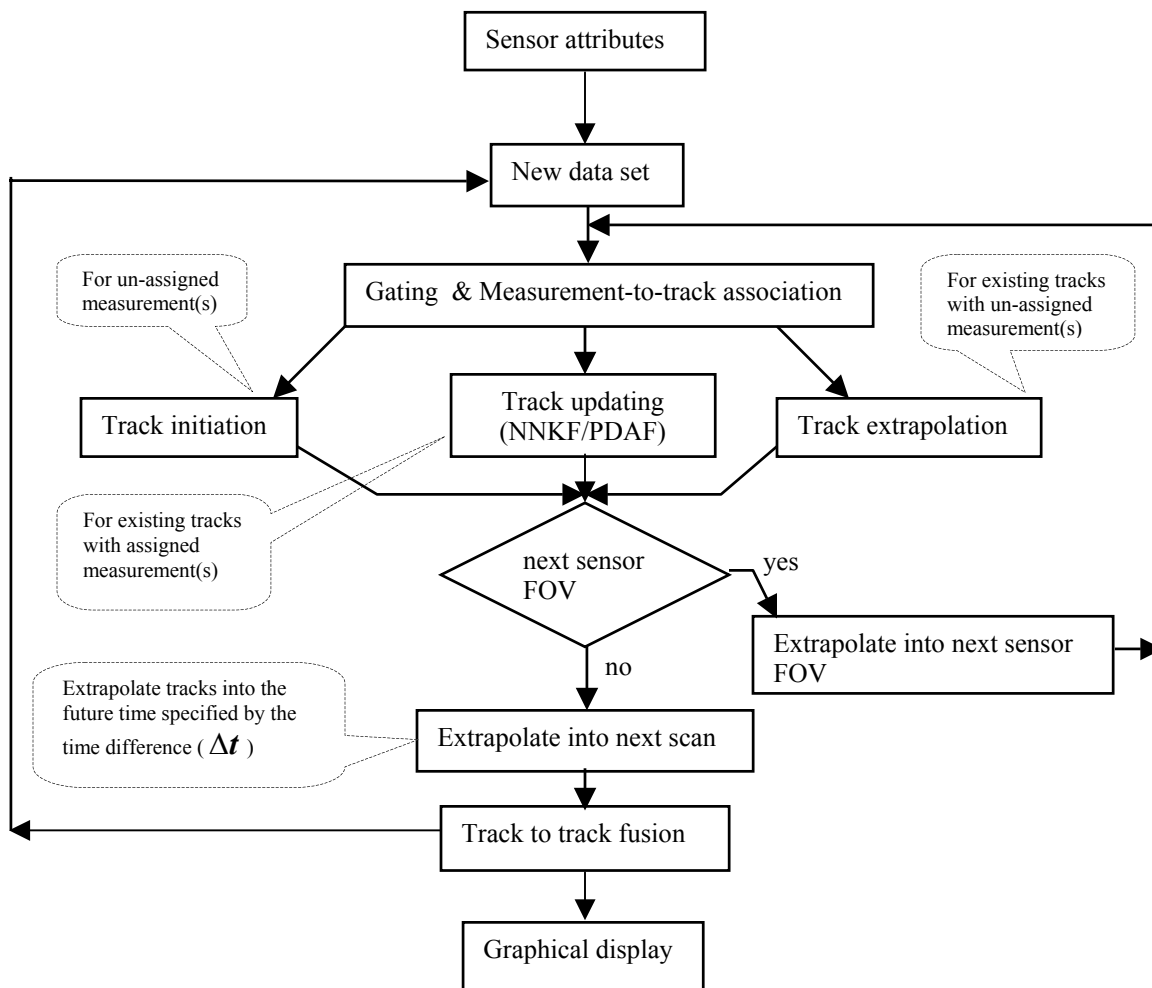


Fig-1. Flow chart of the MSMT program

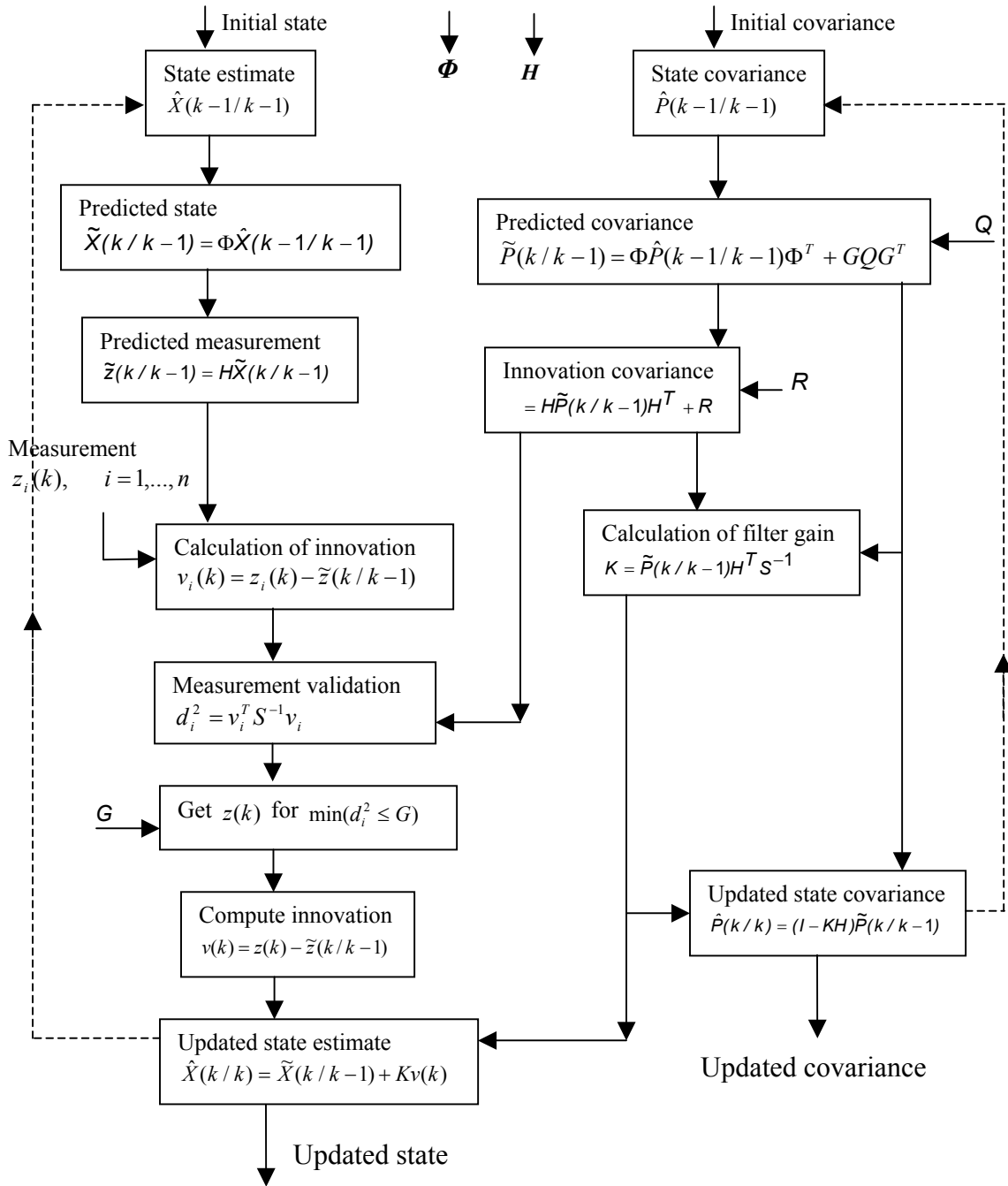


Fig-2. Information flow in NNKF

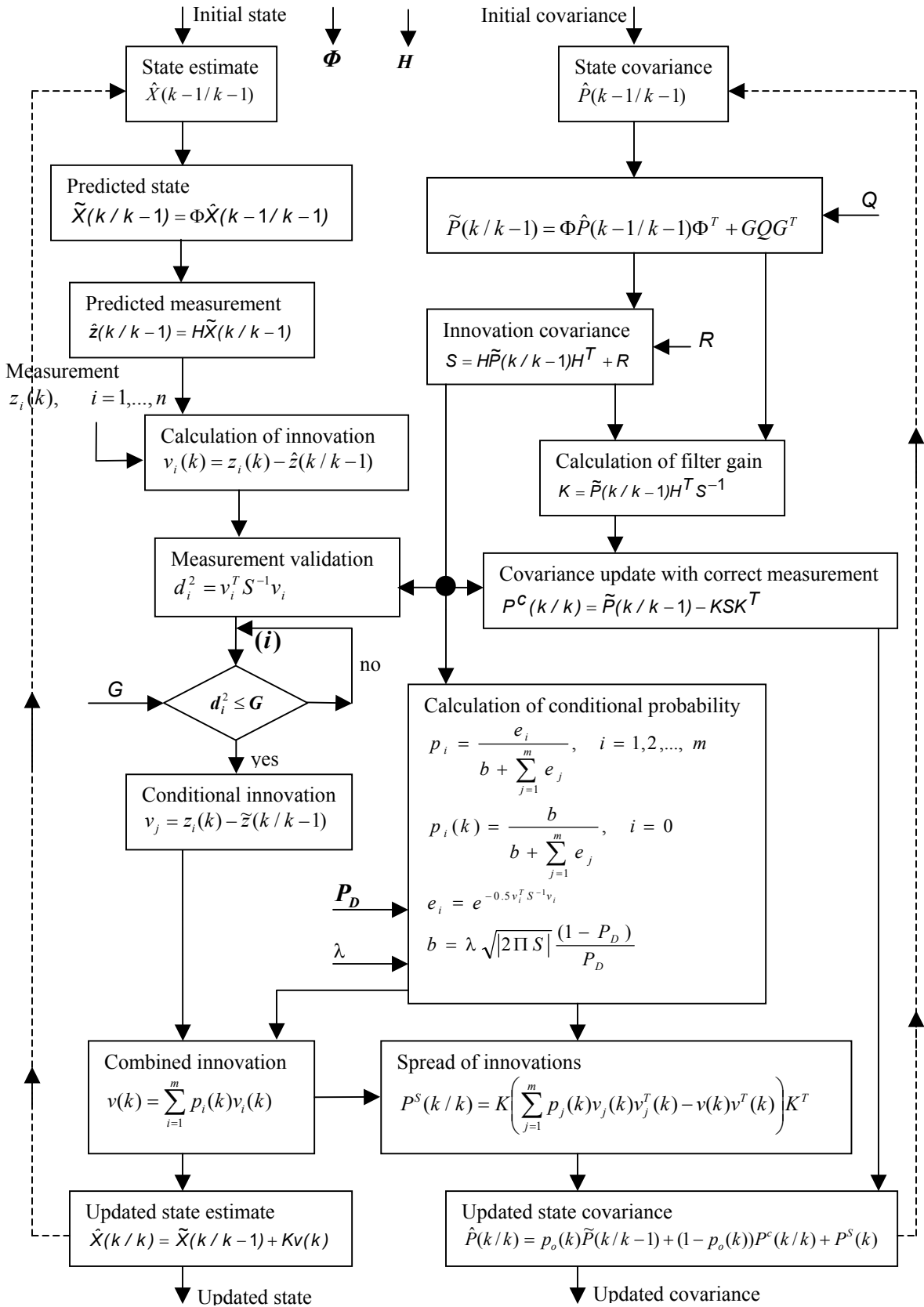


Fig-3. Information flow in PDAF algorithm

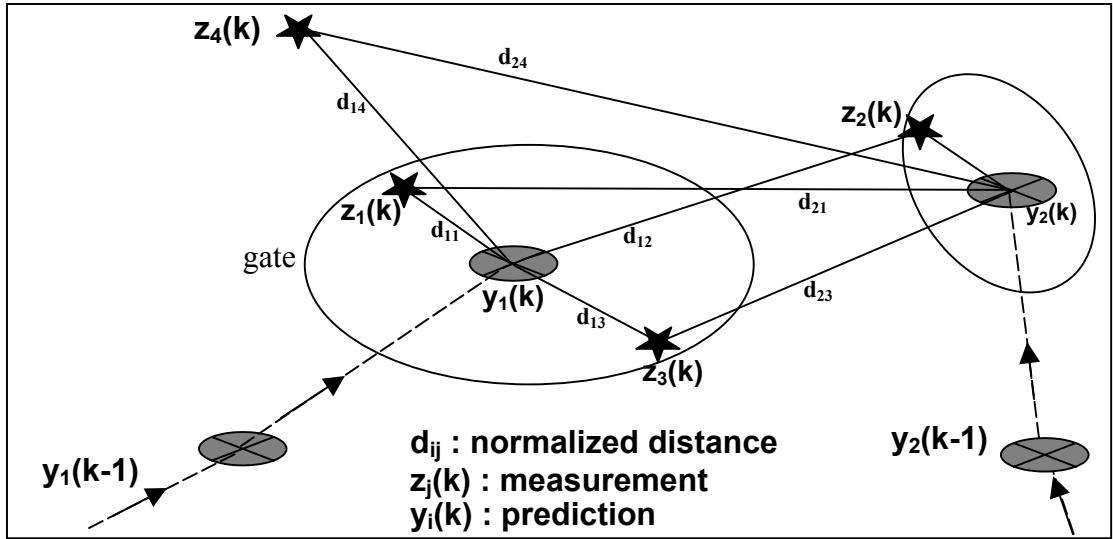


Fig-4. Illustrating gating principle.

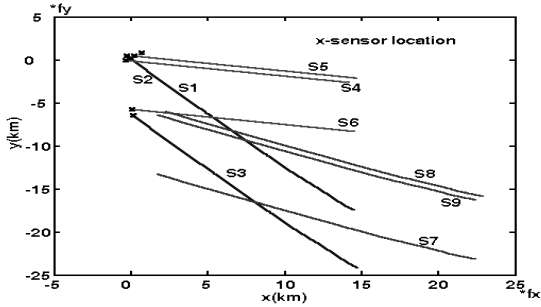


Fig-5. Trajectories as seen from respective sensor locations (f_y and f_x are factors)

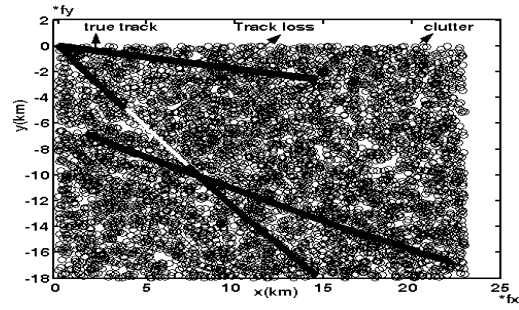


Fig-6. Measurement data with simulated clutter, (converted to a common reference location)

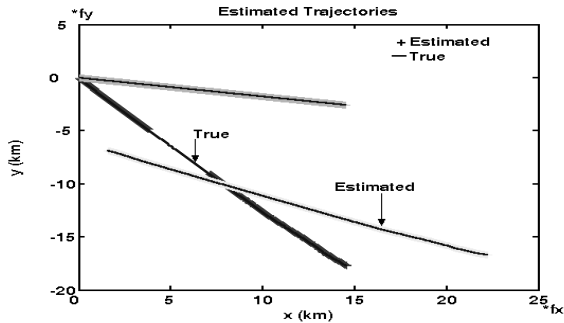


Fig-7. Comparison of estimated trajectories with true trajectories

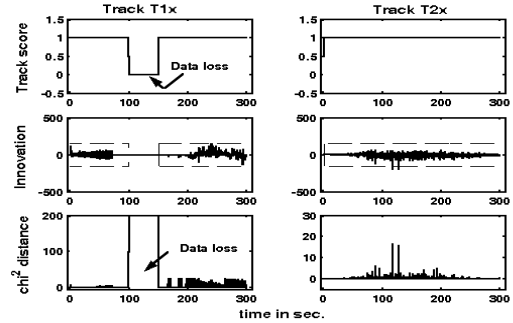


Fig-8. Performance Evaluation measures

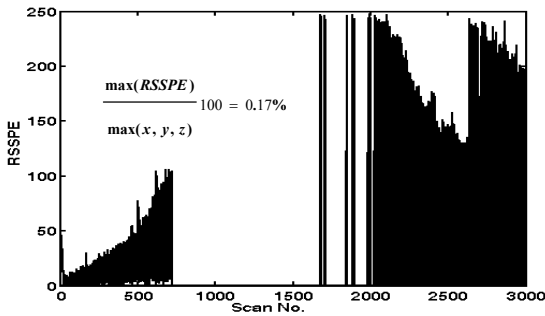


Fig-9. RSSPE in track-1 without data loss

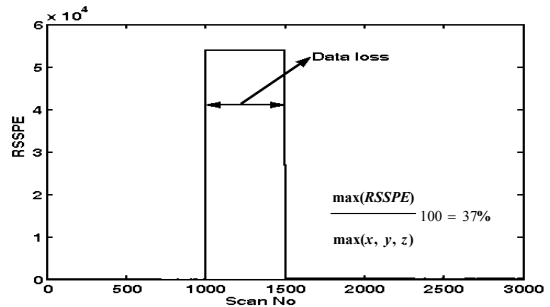


Fig-10. RSSPE in track-1 with data loss

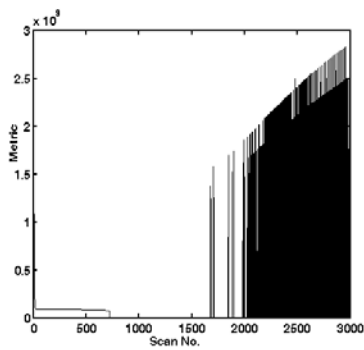


Fig-11. Singer-Kanyuck association metric for i^{th} and j^{th} tracks from different sensor (data loss)

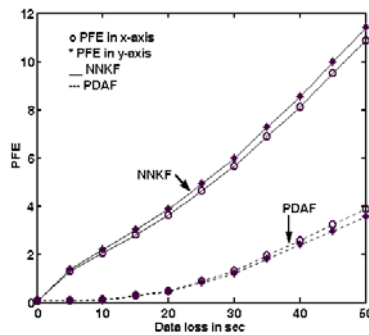


Fig-12. PFE with data loss

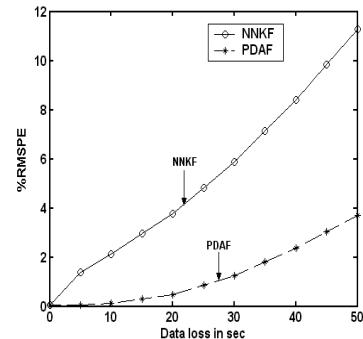


Fig-13. Percentage RMSPE with data loss

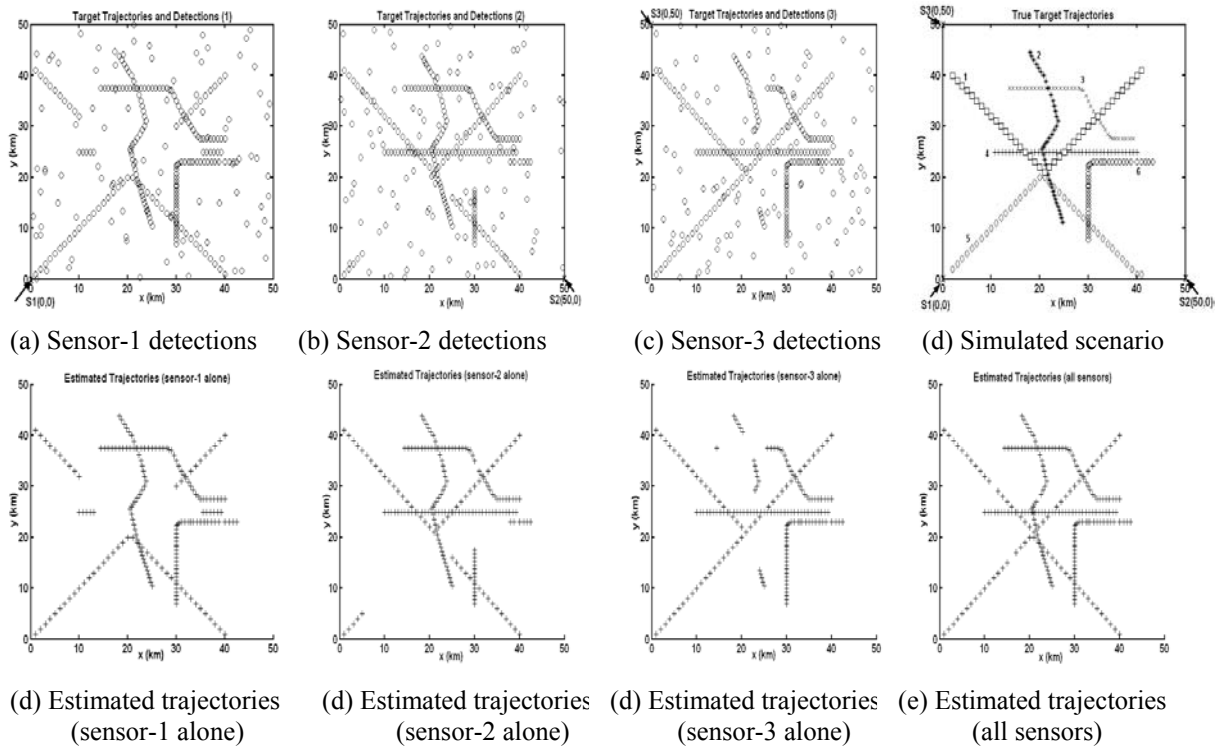


Fig-14. Simulated scenario having 6-targets tracked by 3-sensors (data loss)

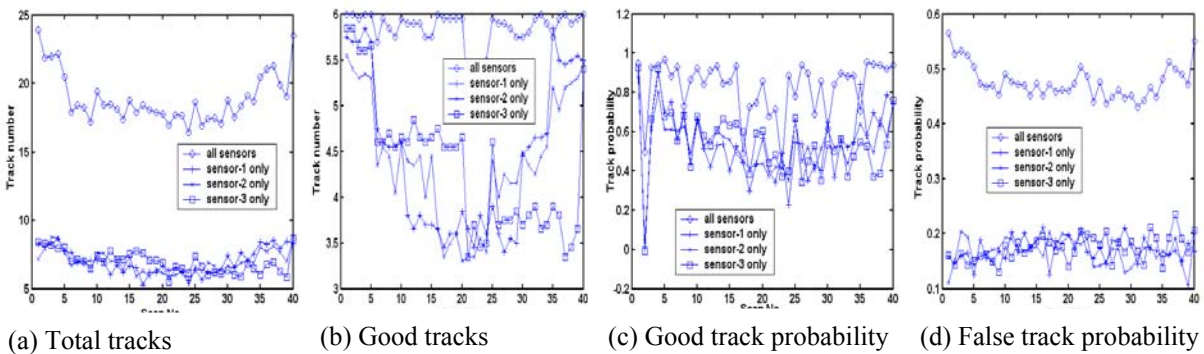


Fig-15. Performance evaluation measures