Introduction to Multiple Target Tracking

The Problem

The multiple target tracking problem is encountered in many situations, whenever sensor data is available from one or more wide angle of view sensors (such as radar, electronic support measures (ESM), infra-red (IR), video, etc.), providing information on the position (such as range, azimuth, elevation etc.) of "targets"' (aircraft, vessels, missiles, ground vehicles etc.). The sensors may also provide information useful for target identification purposes (size, velocity, luminosity, transmitter frequency, etc.). Although this information may be useful to a target tracking algorithm, the identification aspects of target tracking are beyond the scope of this report. These target tracking sensors typically scan their field of view at regular intervals. There may be many contacts from such a scan, from the different objects in the sensors field of view, as well as some which may be just noise. The sensor measurements from all contacts within a scan are generally processed together, since they can safely be assumed to correspond to different objects (something which cannot be said of returns from different scans or from different sensors).

The basic tracking problem is to estimate the position and velocity of the target(s), using the available sensor data (from a sequence of scans). It is also normally desirable to be able to predict the targets' location sometime in the near future. This is a problem because the sensor data normally has errors and/or ambiguities and the targets generally move between scans. The prediction is further complicated by unknown target maneuvers. If there is only a single target, however, then Kalman filtering techniques provide an "optimal solution. Although the nonlinearities of this problem introduce some complications in the use of Kalman filtering, this problem has been extensively studied and suitable solutions exist. The multi-target tracking problem however, is not simply a tracking problem when there is more than one target. As shall be seen below however, the possibility of more than one target significantly complicates the situation, because it introduces the problem of associating measurements with targets. When the distance between targets is comparable to the measurement error or to the error in target position prediction (from one scan period to the next), then this association problem can be very difficult. For example, fighter aircraft flying in tight formation being tracked by scanning radar. Generally, track-updating process typically begins with a procedure that is used to choose the best observation to track association. This procedure is known as data correlation and is conventionally comprised of two steps called gating and association.

Background

There currently is no satisfactory unified solution method for the Multi-target tracking problem. The single target tracking problem, however, has been studied in depth, and optimal estimation theory (Kalman filter theory etc.), does provide a satisfactory solution (see for example [1]). With this method, one can form the statistically optimal estimate of the target state vector (position and velocity) by recursively processing sensor measurements (radar, sonar, ESM etc.) taken from the target being tracked.

The multi-target tracking problem can be approached in the same way, as a number of stochastic estimation problems (one for each target). Unfortunately there is the added difficulty of deciding which measurements correspond to which targets (and which measurements don't correspond to any targets). This problem of associating measurements with targets is known as the "data association problem". Since this is not a problem of estimating the value of a random variable, it does not fall neatly into the realm of optimal estimation. However, it is amenable to stochastic treatment, and several approaches have been suggested. This report describes in brief three of the more promising approaches. All three approaches are based on the intuitive premise that the probability of a particular measurement corresponding to a specific target is a function of the proximity of that measurement to the expected target location. The simplest of the approaches described here is the "Nearest Neighbor" method, which at each time epoch (scan) makes a complete assignment decision. This complete assignment can be defined as a specific decision for each measurement:

1- that it came from one of the targets being tracked,
2- that it is noise, or
3-that it is from the new target

Furthermore this allocation of measurements is made in such a way that no target is assigned more than one measurement from any given sensor scan. the nearest neighbor approach makes this complete assignment decision at each point in time by minimizing a global distance function which represents the closeness of each measurement to its assigned target. A problem with this approach is that if an incorrect assignment is made for a single measurement, then the tracking filter that processes it will make a poor prediction for the position of it's target for the next time epoch, likely leading to subsequent incorrect assignments, and thus the process could easily break down. The two other methods described here are different approaches to solving this problem.

1.GLOABAL NEAREST NEGHBOUR APPROACH

INTRODUCTION

A main function of each radar surveillance system is the target tracking. The basic part of this problem is the process of data association. Most data association methods require a measure of probability in order to evaluate alternative hypotheses. The basic Global Nearest Neighbor (GNN) approach attempts to find and to propagate the single most likely hypothesis at each scan.

In a cluttered environment, the received measurements may not all arise from the real targets. Some of them may be from clutter or false alarm. As a result, there always exist ambiguities in the association between the previous known targets and measurements. Assigning wrong measurements to tracks often results in lost tracks and track breaks. Moreover, clutter can produce false tracks, and if the clutter density is sufficiently large, the resulting number of false tracks can overwhelm the available computational resources of the MTT systems, as well as degrade the overall picture of the environment. For these reasons techniques dealing with data association have received much attention in MTT research.

There are many data association techniques used in MTT systems ranging from the simpler nearest-neighbor approaches to the very complex multiple hypothesis tracker (MHT). The simpler techniques are commonly used in MTT systems, but their performance degrades in clutter. The more complex MHT provides improved performance, but it is difficult to implement and in clutter environments a large number of hypotheses may have to be maintained, which requires extensive computational resources. Because of these difficulties, some other algorithms having smaller computational requirements were developed .

The problem of correct data association is difficult to be resolved in dense target environment. In these cases there are clusters with multiple targets and received measurements. There often have ambiguities. Global Nearest Neighbor approach gives an optimal solution. Recently the increased computational power of the computers allows using this approach in real time implementations.

The Problem Formulation

The track updating process typically begins with a procedure that is used to choose the best observation to track association. This procedure is known as data correlation and is conventionally comprised of two steps called gating and association.

Gating

Gating is a coarse test for eliminating unlikely observation-to-track pairing. A gate is formed around the predicted position. All measurements that satisfy the gating relationship fall within the gate and are considered for track update. The manner in which the observations are actually chosen to update the track depends on the data association method but most data association methods utilize gating in order to reduce later computation. The use of Kalman filtering, with the associated covariance matrix, is assumed. At scan k −1, the filter evaluates the prediction xˆ (k | k −1) The measurement at scan k is

$$y_j(k) = H\, x_i(k) + v(k), \qquad (1.1)$$

of the state vector of the i^{-th} track.

where H is the measurement matrix and $v(k)$ is zero-mean, white Gaussian measurement noise with covariance matrix R . The vector difference between measured and predicted quantities,

$$\tilde{y}(k) = y(k) - H\hat{x}(k \mid k-1), \qquad (1.2)$$

is defined to be residual vector with residual covariance matrix $S = HPH^T + R$, where P is the state prediction covariance matrix. The time subscripts k will be dropped f or notational convenience. Assume that the measurement vector size is M . Defining d^2_{ij} be the norm of the residual (or innovation vector $d_{ij}=y_{ij}{}^T S^{-1} y_{ij}$) (1.3)

the M -dimensional Gaussian probability density for the residual is

$$g_{ij}(\tilde{y}) = \frac{e^{-\frac{d_{ij}^2}{2}}}{(2\pi)^{M/2}\sqrt{|S_i|}}\ , \qquad (1.4)$$

where $|S_i|$ is determinant of S_i .

Define a threshold constant for gate G such that correlation is allowed if the following relationship is satisfied by the norm $(d_{ij}{}^2)$ of the residual vector

$$d_{ij}^2 = \tilde{y}_{ij}^T\, S_i^{-1}\, \tilde{y}_{ij} < G \qquad (1.5)$$

The quantity d_{ij}^2 is the sum of the squares of M independent Gaussian random variables with zero means and unit standard deviations. For that reason d_{ij}^2 will have $\chi^2{}_M$ distribution for correct observation-to-track pairings with M degrees of freedom and allowable probability $P = 1- P_d$ of a valid observation falling outside the gate, where P_d is the probability for correct detection. The threshold constant G can be defined from the table of the chi-square (χ^2_M) distribution with M degrees of freedom and allowable probability of a valid observation falling outside the gate [1].

Data association:

In a dense target environment additional logic is required when an observation falls within the gates of multiple target tracks or when multiple observations fall within the gate of a target track. The optimal assignment minimizes a total distance function which is the sum of the distances for all the individual assignments.

Thus it is first necessary to define a distance measure from the predicted positions of track i to observation j . This quantity is termed a normalized or statistical distance function (actually the squared distance). There are several different ways of defining it. One convenient definition is presented in [3]. The basic goal is to choose assignments that maximize the g_{ij} terms. By taking a logarithm of (4) (for better numerical stability to avoid floating point overflow [5]) it is seen that maximization of g_{ij} is equivalent to minimization of the quantity:

$$d_{G_{ij}}^2 = d_{ij}^2 + \ln|S_i| \ . \qquad (1.6)$$

Assuming the same measurement dimension M for all observations, the quantity is a convenient distance function for use in the problem of assigning observations to tracks.
$d_{G_{ij}}^2 = d_{ij}^2 + \ln|S_i|$

Data association takes the output of the gating algorithm and makes final measurement-to-track associations [1.5]. When a single measurement is gated to a single track, an assignment can be immediately made. However for closely spaced targets, it is more likely that conflict situations will arise. Conflict situations arise when multiple measurements fall within a single gate, or when a single measurement falls within the gates of more than one track. The data association algorithm attempts to resolve these conflicts using probabilistic methods. The simplest is the so-called suboptimal nearest-neighbor (SNN) approach. The SNN assignment algorithm assigns observations to existing tracks minimizing some distance criterion. The SNN looks through the gated measurements and chooses the measurement with minimum distance d with the considered track. An example of complex conflict situation is presented on fig.1.

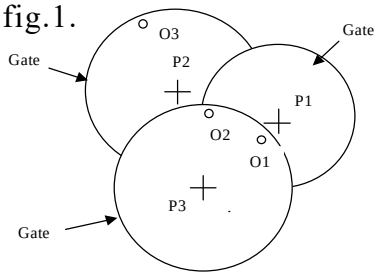


Fig.1 Example of a complex conflict situation

The predicted values define the center of the gate region. The measurement O2 falls in the gates of the three tracks. In such complex situation the SNN approach could give wrong assignment solution and to lead to missed detection for some track.

Algorithm Description

We assume the existence of a set of n tracks at the time a new observation or set of observations is received. These observations may be used for updating the existing tracks or for initiating new tracks. Suppose that m measurements are received at time index k . In a cluttered environment, m does not necessarily equal n and it may be difficult to distinguish whether a measurement originated from a target or from clutter. A validated measurement is one which is either inside or on the boundary of the validation gate of a target. Mathematically, a validation gate is defined by equations (1.5).

The choice of G has to ensure that the correct measurements will lie within the gate with the specified probability. The inequality given in (1.5) is a validation test. On the base of the validation test the cost matrix C for assignment problem solution is defined.

$$[C_{ij}] = \begin{matrix} \overset{1}{\overbrace{}} & \overset{2}{\overbrace{}} & \overset{3}{\overbrace{}}^{j} & \overset{...}{\overbrace{}} & \overset{m}{\overbrace{}} \end{matrix} \begin{matrix} \left[\begin{matrix} c_{11} & c_{12} & c_{13} & : & c_{1m} \\ c_{21} & c_{22} & c_{23} & : & c_{2m} \\ : & : & : & : & : \\ c_{n1} & c_{n2} & c_{n3} & : & c_{nm} \end{matrix} \right. & \begin{matrix} 1 \\ 2 \\ : \\ \left. \begin{matrix} n \end{matrix} \right\} i \end{matrix}$$

The elements of the cost matrix c_{ij} have the following values:

$$c_{ij} = \begin{cases} 100 & \text{if measurement } j \text{ IS NOT in the gate of track } i \\ d_{ij}^2 & \text{if measurement } j \text{ is IN the gate of the track } i \end{cases}$$

Strictly speaking if measurement j is in the gate of the track i $c_{ij} = d^2 + \ln|s_i|$ as is given in (6), but the assignment problem solution is the same if

$$c_{ij} = d_{ij}^2 \text{ [1.6].}$$

The desired solution of the assignment (cost) matrix is the one that minimizes the summed total distance. For simple cases the optimal solution can be easily found by enumeration. But the enumeration is too much time consuming in more complicated cases. We choose to solve the assignment problem by realizing the extension of MunKers algorithm, given in [1.6].

GNN algorithm description:

1. *Receiving data* for current scan.
2. *Clusterisation* – measurements to tracks allocation:

   At the beginning all tracks are clusters. In two nested cycles for all tracks and for all measurements using gating criterion it is defined if some measurement falls in the gate of the given track. When two tracks have common measurement in their gates their clusters are merged in supercluster.

3. For each cluster:

3.1. *Measurements to tracks association*.

At this stage the elements of the cost matrix for the assignment of the measurements to tracks in the current cluster is defined by equation (1.6). Solve assignment problem using Munkres algorithm.

3.2.*Track Filtering.*

Taking from the Munkres solution the associated measurement for each track state update is performed using extended Kalman filter in the frame of Interacting Multiple Model (IMM) approach.

4. *Track Initiation*.

Measurements, which are not associated with existing tracks, generate new tracks.

Step 1: Set k=l (scan index) and all control parameters.
Step 2: Initialize the system parameters; t, O, H, R.
Step 3: Simulate target trajectories.
Step 4: Simulate measurements (or receive the data from radar.) Step
5: Validate the measurements using gating test: For gate about

$$ye_{k|k-1} = H * xe_{k|k-1}$$

and select the Nk sensor reports to be used in filter updating. A measurement is valid if

$$(y_k - ye_{k|k-1})^T S^{-1} (y_k - ye_{k|k-1}) <= g$$

Step 6: Form the assignment matrix. The elements of the matrix are equal to the normalized distance function associated with the assignment of each of "Nobs" observations to each of "Ntr" tracks. If the gating relationship is not satisfied, the observation- to-track pairing can be given a very large distance to penalize this assignment.
Step 7: Solve the Assignment Matrix: Minimize the normalized distance function using the modified MunKers optimal assignment algorithm in section 2.3.
Step 8: Correlate the observations to the tracks according to the solution of optimal assignment matrix.
Step 9: Update and predict the state vectors (tracks) using (4) to (9).
Step 10: Output the predicted track positions.
Step 11: If k=kf end of tracking mission), go to the Step 11.Otherwise, set k=k+l and go to Step 4.

Modified MunKers Assignment Algorithm

The following method is based on the Munkres optimal assignment algorithm modified by Burgeois and Lassalle (12]. This method has an advantage (compared to older methods) for applications since the Reassignment matrix need not be square. For the convenience of presentation of the algorithm, the rows and colu•ns of the matrix may be marked and referred to as covered. The zeros may be marked by being starred (*) or primed (').

The Optimal Assignment Algorithm:

Step 1: Initially, no lines are covered and O's are starred or primed.
Step 2: Let v=min.(Number of rows, Number of columns}.
Step 3: If (number of rows) > (number of columns), go to Step 6.
Step 4: For each row in the matrix, subtract the smallest element of the row from each element in the row.
Step 5: If the (number of columns) > (number of rows), go co Step 7
Step 6: For each column in the matrix, subtract the smallest element of the column from each component of the column and 90 to Step 7.
Step 7: a) Find a zero, "Z", of the matrix.
b) If there is no starred zero in its row or its column,star the z ro (i.e. Z*). Repeat for all zeros of the matrix. Go to Step 8.

Step 8: a) Cover every column containing a starred zero Z*.
b) If v columns are covered, the location5 of the Z* form the row-column associations (i.e. observation-to-track pairs). The algorithm is now completed. Otherwise, continue to the next step.
Step 9: a) Choose an uncovered zero and prime it (i.e. Z').

b) If there is no starred O in the row of Z', go to Step 10.
c) If there is a starred zero Z* in the row of Z', cover this row and uncover the column of Z*.
d) Repeat until all zeros are covered and then go to Step 11.

Step 10: a1) Let Zo denote the uncovered Z'. If there is no Z* in the column of Zo, go to Step (a6}.

a2) Let Zl denote the Z* in the column of Zo.
a3) Let Z2 denote the Z' in the row cf Zl.
a4) Continue the steps a2 and a3 until a Z2 which has no Z* in its columns has been found.
a5) Un-star each starred zero of sequence
a6) Star each primed zero of the sequence.
b1) Erase all primes from primed zeros and uncover every line
c1) Go to Step-8.

Step 11: a) Find the smallest uncovered element in the matrix and call it "m". "m" will be positive.

b) Add "m" lo each covered row.
c) Subtract "rn" from each uncovered column.
d) Go to Step 9 without altering stars, primes, or uncovered lines.

2. JOINT PROBABILISTIC DATA ASSOCIATION APPROACH

Introduction

In this section, a JPDA algorithm for multiple targets in Poisson clutter is presented. This all-neighbors approach incorporates all observations within a validation region about the predicted track position into the update of that track. Also, a given observation can be used to update multiple tracks. Track update is then based on a probabilistically determined weighted sum of all observations within the validation region. In fact, the method performs an averaging over observation-to-track data association hypothesis that have roughly comparable likelihood. Further, this is a target-oriented approach, in the sense that a set of established targets is used to form gates in the measurement space and to compute posterior probabilities, in contrast to the measurement-oriented algorithm such as MHT, where each measurement is considered in turn and hypothesized to have come from some established track, a new target, or clutter (false alarm).

Problem formulation:

$$x_{k+1} = Fx_k + Gw_k \qquad (2.1)$$

$$y_k = Hx_k + v_k \qquad (2.2)$$

where xis the target state vector,y is the measurement vector, w and v are zero-mean mutually dependent white Gaussian noise vectors with covariance matrices Q and R, respectively, and k is a discrete time index. The matrices F, G, H, Q, and R are assumed known and their dependence on k is suppressed here for notational convenience. The initial state is assumed Gaussian with mean x_{010} and covariance P_{010}. A specific tar-get model is described below in Section IV.

The tracker's estimate of the target state x_k at time k, given data up to time i, is denoted $x_{k|i}$ and the corresponding estimate of the output Y_k is $Y_{k|i}$. The error in the state estimate is $y_{k|i}$ $x_{k|i}$, with error covariance matrix $P_{k|i} \triangleq E\{x_{k|} x_{k|i}\}$, where E denotes expectation. In the absence of measurement origin uncertainty, the discrete-time Kalman filter yields the state estimate and covariance via the recursions

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + W_k\tilde{y}_k = F\hat{x}_{k-1|k-1} + W_k\tilde{y}_k \tag{2.3}$$

$$P_{k|k} = P_{k|k-1} - W_k S_k W_k' \tag{2.4}$$

$$= FP_{k-1|k-1}F' + GQG' - W_k S_k W_k' \tag{2.5}$$

where the innovation vector

$$\tilde{y}_k \triangleq y_k - \hat{y}_{k|k-1} \tag{2.6}$$

has the covariance matrix

$$S_k \triangleq E\{\tilde{y}_k\tilde{y}_k'\} = HP_{k|k-1}H' + R \tag{2.7}$$

and the filter gain matrix is

$$W_k = P_{k|k-1}H'S_k^{-1}.$$

The resulting state estimate, under the above assumptions, is the *conditional mean*

$$\hat{x}_{k|k} = E\{x_k \mid Y^k\} \tag{2.8}$$

where y^k denotes the set of all data vectors Y_i for $i \leqslant k$.

In order to avoid cluttering the discussion to follow, this brief summary ignores a number of complications that arise in practice. If the system is nonlinear, for instance (as it is in the example presented below), then it can usually be linearized and the same basic equations can be applied to deviations from the nominal trajectory [7], [9]. If the target occasionally de-viates from the assumed motion model, e.g., by maneuvering, then some decision-making or other machinery must be pro-vided to deal with these instances.

A fundamental characteristic of this class of problems is that the size and composition of the measurement vector are unpredictable from one time to the next; in other words, Y_k comprises a time-varying set of independent sub vectors, as dis-cussed below in Section IV. We shall avoid the resulting nota-tional morass by restricting (2.3)-(2.6) to apply to a single measurement sub vector Y_k from a single sensor. In addition, *we will suppress the time subscript k* from all variables ex-cept ***P*** and ***Y,*** unless it is required for clarity. Without any loss of generality, the *data association problem* may now be formulated as follows.

At each time step, the sensor provides a set of candidate measurements to be associated with targets (or rejected). In most approaches, this is done by forming a "validation gate" around the predicted measurement from each target and re-taining only those detections that lie within the gate. There are many different approaches to establishing a correspondence between candidate measurements and targets; in this paper we

shall focus on the probabilistic data association (PDA) method [I], (2), (11]. Them candidate measurements at time k are

denoted yi,j = 1, ... , m, i.e.,

$$Y^k = \{y_1, \cdots y_m\} \cup Y^{k-1} \qquad (2.9)$$

and the corresponding innovations are

$$\tilde{y}_j \triangleq y_j - \hat{y}, \qquad j = 1, \cdots, m. \qquad (2.10)$$

denotes the event that the *jth* measurement belongs to that target and X_0 the event that none of the measurements belongs to it (no detection). The **PDA** approach builds upon the as-sumptions that the estimation errors x and y have Gaussian densities at each time step (this is approximate, since there is an exponentially growing tree of possible measurement se-quence hypotheses and the true densities are weighted sums of Gaussian). It is also assumed that the correct measurement is detected with probability P_0 (independently, at each time) and that all other measurements are Poisson-distributed with parameter CV, where C is the expected number (or density) of false measurements per unit volume and V is the volume of the validation gate.

In the PDA filtering approach, the conditional mean esti-mate obtained from (2.3) by using the combined (weighted) innovation

$$\tilde{y} \triangleq \sum_{j=1}^{m} \beta_j \tilde{y}_j \qquad j = 1,2,3...m \qquad (2.11)$$

where $\beta_j \quad P\{\chi_j | Y^{k}\}$, $j = 0, 1, \cdots, m$, is the posterior proba-bility that the *j*th measurement (or no measurement, for $j = 0$) is the correct one. Expressions for these probabilities are given in the Appendix. The update part of the covariance equation (2.4) becomes [2], [11]

$$P_{k|k} = P_{k|k-1} - (1 - \beta_0) W_k S_k W_k' + P_k \qquad (2.12)$$

where the positive semidefinite matrix

$$P_k = W_k \left[\sum_{j=1}^{m} \beta_j \tilde{y}_j \tilde{y}_j' - \tilde{y}\tilde{y}' \right] W_k' \qquad (2.13)$$

probability mass function p(n) = e-CV(cV)n/n! and the location of false measurements equivalent, each false number of measurements has uniformly distributed in the same gate.

This is just the Gaussian probability mass in the gate, which is often assumed to be unity in practice,since P_G>0.99 whenever g > $M^{1/2}$ + 2. accounts for the measurement origin uncertainty. Note that the data-dependent factors β_0 and P_k transform the original deterministic Riccati equation into a stochastic one.

Multiple Interfering Targets andJPDA

The equations above define the PDA filter for a single tar-get, and additional targets may be handled with multiple copies of the filter. However, with respect to any given target, meas-urements from interfering targets do not behave at all like the random (Poisson) clutter assumed above. Rather, the proba-bility density of each candidate measurement must be com-puted based upon the densities of all targets that are close enough (when projected into the measurement space) to in-terfere.

In order to account for this interdependence, consider a cluster of targets (established tracks) numbered t =i,⋯, t at a given time k. The set of m candidate measurements as-sociated with this cluster (i.e., found in the validation gates for targets 1. ⋯. t) is denoted yj, j=1, ⋯, m, as above. Each measurement belongs either to one of the t targets or to the set of false measurements (clutter), which is denoted by target number t = 0. Denoting the predicted measurement for target t by y'^t, the innovation (2.10) corresponding to measurement j becomes

$$\tilde{y}_j^t \triangleq y_j - \hat{y}^t \qquad (2.14)$$

and the combined (weighted) innovation (2.11) becomes

$$\tilde{y}^t = \sum_{j=1}^{m} \beta_j^t \tilde{y}_j^t \qquad (2.15)$$

where β_j^t is the posterior probability that measurement j originated from target t and β_0^t is the probability that none of the measurements originated from target t (i.e., it was not detected). This is used in target t's copy of (2.3) to update the state estimate $\hat{x}^t$.

In other words, the joint probabilistic data association (JPDA) and PDA approaches utilize the same estimation equa-tions; the difference is in the way the association probabilities are computed.

JPDA Algorithm

The key to the **JPDA** algorithm is evaluation of the conditional probabilities of the following joint events:

$$\chi = \bigcap_{j=1}^{m} \chi_{jt_j} \qquad (3.1)$$

A clu;ter is a set of targets whose validation gates are "connected" by measurements lying in their intersections [3 J, [4]. Note that a dif• ferent measurement subvector (e.g., from another sensor) will lead to a different target cluster.

where X;t- is the event that the measurement j originated from target $t;$, 6 '¾ $t;$ '¾ T, and $t;$ is the index of the target to which measurement j is associated. The *feasible events* are those joint events in which no more than one measurement originates from each target, i.e.,

$$j \neq l \text{ and } t_j > 0 \text{ implies } t_j \neq t_l \qquad (3.2)$$

It is also convenient to define the *measurement association indicator*

$$\tau_j(\chi) \triangleq \begin{cases} 1, & \text{if } t_j > 0 \\ 0, & \text{if } t_j = 0 \end{cases} \qquad (3.3)$$

which indicates whether measurement j is associated with any established target in event X,and the *target detection indicator*

$$\delta_t(\chi) \triangleq \begin{cases} 1, & \text{if } t_j = t \text{ for some } j \\ 0, & \text{if } t_j \neq t \text{ for all } j \end{cases} \qquad (3.4)$$

which indicates whether any measurement is associated with targett in event x (i.e. whether target t is detected).

For the purpose of deriving expressions for the joint proba-bilities, *no individual validation gates will be assumed* for the various targets in a cluster. Instead, each measurement will be assumed validated for each target, i.e., every validation gate coincides with the entire surveillance region. This approach is adopted because the resulting equations are equivalent to but simpler than those derived if different validation gates are as-sumed for each target.

The extra computational burden resulting from the con-sideration of events with negligible probability can be avoided with suitable logic limiting the probability calculations to events involving validated measurements. This logic, which has a negligible effect on the numerical results, is presented next, followed by a derivation of the joint event probabilities. Finally, the marginal event (measurement-to-target association) probabilities needed by the tracking filter are obtained from the joint event probabilities.

Validation logic

The following validation matrix is defined

$$\Omega = [\omega_{jt}], \quad j = 1, \cdots, m, \quad t = 0, 1, \cdots, T \tag{3.5}$$

with binary elements to indicate if measurement j lies in the validation gate for target t. Index $t = 0$ stands for "no target" and the corresponding column of Ω has all units–each measurement could have originated from clutter or false alarm. A typical set of gates and the corresponding validation matrix is shown in Fig. 1.

Each event χ may be represented by a matrix

$$\hat{\Omega}(\chi) = [\hat{\omega}_{jt}(\chi)] \tag{3.6}$$

consisting of the units in Ω corresponding to the associations

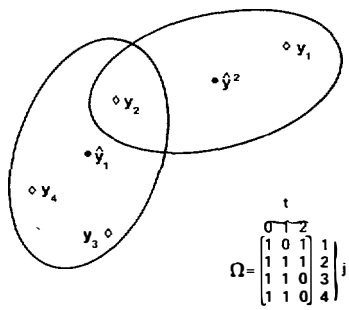


Fig. 1. Typical gates and validation matrix.

assumed in event χ. Thus

$$\hat{\omega}_{jt}(\chi) = \begin{cases} 1, & \text{if } \chi_{jt} \text{ occurs} \\ 0, & \text{otherwise.} \end{cases} \tag{3.7}$$

Given considerable patience, one can construct a procedure that scans Q and enumerates the matrices i1(x) corresponding to feasible events using the following rules:

1) Scan n by rows and pick one unit per row for Q (i.e., there can be only one origin for a measurement).
2) Only one unit from each column $t \geqslant 1$ can be taken (i.e., at most one measurement could have origi-nated from a target). The number of units from column t = 0 is not restricted In terms of the matrix D(x), the measurement association indicator (3 .3) becomes

$$\tau_j(\chi) = \sum_{t=1}^{T} \hat{\omega}_{jt}(\chi) \tag{3.8}$$

and the target detection indicator (3.4) is

$$\delta_t(\chi) = \sum_{j=1}^{m} \hat{\omega}_{jt}(\chi). \tag{3.9}$$

Joint Event Probabilities

Using Bayes' rule, the probability of a joint event con-ditioned on all measurements up to the present time is

$$P\{\chi \mid Y^k\} = P\{\chi \mid \tilde{y}_1, \cdots, \tilde{y}_m, m, Y^{k-1}\}$$
$$= p(\tilde{y}_1, \cdots, \tilde{y}_m \mid \chi, m, Y^{k-1}) P\{\chi \mid m, Y^{k-1}\}/c. \quad (3.10)$$

The normalization constant $c = p(y_1 \cdots, y_m \mid m, Y^{k-1})$ is the joint prior density of the measurements, conditioned only on m (and the past data); it is obtained by summing the numerators over all χ. The first factor in (3.10) is the joint probability density of the m candidate measurements, conditioned on the joint event χ

$$p(\tilde{y}_1, \cdots, \tilde{y}_m \mid \chi, m, Y^{k-1}) = \prod_{j=1}^{m} p(y_j \mid \chi_{jt_j}, Y^{k-1}). \quad (3.11)$$

Using the same assumptions as in the **PDA** approach, a measurement Yi associated with a target t has a Gaussian density

$$N(\tilde{y}_j^t; 0, S^t) = \exp(-\tilde{y}_j^{t\prime} S_t^{-1} \tilde{y}_j^t/2)/(2\pi)^{M/2} |S_t|^{1/2} \quad (3.12)$$

where $\tilde{y}_j^t$ is the innovation defined in (2.14) and S_t is its covariance. A measurement associated with clutter has a uniform density V^{-1}, where V is the volume of the entire surveillance region (note that in Section II and the Appendix, V was the volume of a single gate). It follows that the densities of individual measurements in (3.11) are given by

where $\tilde{y}_j^t$ is the innovation defined in (2.14) and S_t is its covariance. A measurement associated with clutter has a uniform density V^{-1}, where V is the volume of the entire surveillance region (note that in Section II and the Appendix, V was the volume of a single gate). It follows that the densities of individual measurements in (3.11) are given by

$$p(y_j \mid \chi_{jt}, Y^{k-1}) = \begin{cases} N(\tilde{y}_j^t; 0, S_t), & \text{if } \tau_j(\chi) = 1 \\ V^{-1}, & \text{if } \tau_j(\chi) = 0. \end{cases} \quad (3.13)$$

The second factor in (3.10) is the prior (to time k) probability of a joint event. To obtain it, note that the total number of false measurements in event χ is

$$\phi(\chi) = \sum_{j=1}^{m} [1 - \tau_j(\chi)]. \quad (3.14)$$

Then the number of events χ in which the same set of targets is detected is given by the number of permutations of m (number of measurements) taken as m $\phi(\chi)$ (number of detections)

Substituting (3.11), (3.13), and (3.16) into (3.10) yields

$$P\{\chi \mid Y^k\} = \frac{V^{-\phi}}{c} \prod_{j:\tau_j=1} N(\tilde{y}_j^t; 0, S_{t_j}) \cdot \frac{\phi!}{m!} \prod_{t:\delta_t=1} P_D{}^t \prod_{t:\delta_t=0} (1 - P_D{}^t) \frac{e^{-CV}(CV)^{\phi}}{\phi!}. \quad (3.17)$$

Note that $\phi!$ cancels immediately in (3.17). Furthermore, $m!$ and e^{-CV} appear in (3.17) regardless of which event is considered. Since the denominator term c from (3.17) is the sum of all the numerators, it follows that $m!$ and e^{-CV} cancel out as well. The final expression is obtained as

$$P\{\chi \mid Y^k\} = \frac{C^{\phi}}{c} \prod_{j:\tau_j=1} \frac{\exp[-\frac{1}{2}(\tilde{y}_j^t)' S_{t_j}{}^{-1} (\tilde{y}_j^t)]}{(2\pi)^{M/2} |S_{t_j}|^{1/2}}$$

Note that $\phi!$ cancels immediately in (3.17). Furthermore, $m!$ and e^{-CV} appear in (3.17) regardless of which event is considered. Since the denominator term c from (3.17) is the sum of all the numerators, it follows that $m!$ and e^{-CV} cancel out as well. The final expression is obtained as

$$P\{\chi \mid Y^k\} = \frac{C^{\phi}}{c} \prod_{j:\tau_j=1} \frac{\exp[-\frac{1}{2}(\tilde{y}_j^t)' S_{t_j}{}^{-1} (\tilde{y}_j^t)]}{(2\pi)^{M/2} |S_{t_j}|^{1/2}} \cdot \prod_{t:\delta_t=1} P_D{}^t \prod_{t:\delta_t=0} (1 - P_D{}^t) \quad (3.18)$$

where c is the (new) normalization constant.

Numerical overflows and underflows are common in the factor C^{ϕ} and the normalization constant c, because the magnitude of C (in units of 1/volume in the measurement space) is quite variable and ϕ can be 10 or more. The problem can be avoided simply by letting $1/C$ be the unit volume in calculating (3.18), so that C^{ϕ} is replaced by 1^{ϕ}. This change cancels out in the exponential factor, and it causes $|S_{t_j}|^{1/2}$ in the denominator to be multiplied by C. Alternatively, one can use logarithms to compress the numerical range.

Association probability

The probability that measurement j belongs to target t may now be obtained by summing over all feasible events x for which this condition is true

$$P^m_{m-\phi(\chi)} = \frac{m!}{\phi(\chi)!} \tag{3.15}$$

With this, the prior probability ofevent x is given by

$$P\{\chi \mid m, Y^{k-1}\} = \frac{\phi!}{m!} \prod_{t:\delta_t=1} P_D{}^t \prod_{t:\delta_t=0} (1-P_D{}^t) \frac{e^{-CV}(CV)^{\phi}}{\phi!} \tag{3.16}$$

where $P_D{}^t$ is the probability of detection of target *t,* the num-ber ϕ of false measurements is again assumed Poisson distrib-uted with parameter CV, C is the density of false measurements, and the dependence of rfJ and θ, on x has been dropped from the notation.

The Gaussian density defined in (A. 7) includes a factor $1/P_G$ to account for its restriction to the validation gate. Here we assume $P_G = 1$ (corresponding to a very large gate) for convenience.

$$\begin{aligned}\gamma_j(m) &\triangleq P\{\chi_j \mid m, Y^{k-1}\} = P\{\chi_j \mid m\} \\ &= P\{\chi_j \mid m^F = m-1, m\}P\{m^F = m-1 \mid m\} \\ &\quad + P\{\chi_j \mid m^F = m, m\}P\{m^F = m \mid m\} \\ &= \begin{cases} (1/m)P\{m^F = m-1 \mid m\} + (0)P\{m^F = m \mid m\}, \\ \qquad j = 1, \cdots, m \\ (0)P\{m^F = m-1 \mid m\} + (1)P\{m^F = m \mid m\}, \\ \qquad j = 0 \end{cases}\end{aligned} \tag{A.1}$$

because m^F, the number of false measurements, must be either m- 1 (if the target is detected) or m (if it is not). Using Bayes' rule and the assumed Poisson distribution for false measure-ments

$$\begin{aligned}P\{m^F = m-1 \mid m\} &= P\{m \mid m^F = m-1\}P\{m^F = m-1\}/P\{m\} \\ &= [P_D P_G][e^{-CV}(CV)^{m-1}/(m-1)!]/P\{m\} \\ &= P_D P_G m/[P_D P_G m + (1-P_D P_G)CV]\end{aligned} \tag{A.2}$$

$$\begin{aligned}P\{m^F = m \mid m\} &= P\{m \mid m^F = m\}P\{m^F = m\}/P\{m\} \\ &= [1-P_D P_G][e^{-CV}(CV)^m/m!]/P\{m\} \\ &= (1-P_D P_G)CV/[P_D P_G m + (1-P_D P_G)CV]\end{aligned} \tag{A.3}$$

where the denominator $P\{m\}$ is the prior probability of m and is equal to the sum of the numerators in the two equations

$$\begin{aligned}P\{m\} &= P\{m \mid Y^{k-1}\} \\ &= [P_D P_G m + (1-P_D P_G)CV]e^{-CV}(CV)^{m-1}/m!, \\ &\qquad m = 0, 1, \cdots.\end{aligned} \tag{A.4}$$

Substituting back into (A.1) yields

$$\gamma_j(m) = \begin{cases} P_D P_G/[P_D P_G m + (1-P_D P_G)CV], \\ \qquad j = 1, \cdots, m \\ (1-P_D P_G)CV/[P_D P_G m + (1-P_D P_G)CV], \\ \qquad j = 0. \end{cases} \tag{A.5}$$

Note that $\gamma_j(m)$ is independent of j for $j > 0$.

Using Bayes' rule, the posterior probabilities in (2.11) can be expressed as

$$\begin{aligned}&\cdot P\{\chi_j \mid m, Y^{k-1}\}/p(\tilde{y}_1, \cdots \tilde{y}_m \mid m, Y^{k-1}) \\ &= p(\tilde{y}_1, \cdots, \tilde{y}_m \mid \chi_j, m, Y^{k-1})\gamma_j(m)/\sum_{j=0}^{m} (\text{numerators}).\end{aligned} \tag{A.6}$$

$$\beta_j{}^t = \sum_{\chi} P\{\chi \mid Y^k\}\hat{\omega}_{jt}(\chi), \qquad j = 1, \cdots, m, \quad t = 0, 1, \cdots, T. \tag{3.19}$$

$$\beta_0{}^t = 1 - \sum_{j=1}^{m} \beta_j{}^t, \qquad t = 0, 1, \cdots, T. \tag{3.20}$$

These probabilities are used to form the combined innovation {2.15) for each target.

The first factor is the joint probability density of them candi-date measurements, conditioned on the jth one being correct. According to the PDA assumptions, the correct measurement Yi has a Gaussian density

$$N(\tilde{y}_j; 0, S)/P_G \triangleq (1/P_G)\exp(-\tilde{y}_j{}'S^{-1}\tilde{y}_j/2)/(2\pi)^{M/2}|S|^{1/2} \tag{A.7}$$

with mean O and covariance S, where the factor $1/P_G$ accounts for its restriction to the validation gate and each incorrect measurement has a uniform density v^{-1} It follows that

$$p(\tilde{y}_1, \cdots, \tilde{y}_m \mid \chi_j, m, Y^{k-1})$$
$$= \begin{cases} V^{-m+1} N(\tilde{y}_j; 0, S)/P_G, & j = 1, \cdots, m \\ V^{-m}, & j = 0. \end{cases} \quad \text{(A.8)}$$

The second factor in (A.6) is the prior probability of Xj, given by (A.5). The denominator is the *joint prior density* of the measurements, conditioned only onm (and the past data)

$$p(\tilde{y}_1, \cdots, \tilde{y}_m \mid m, Y^{k-1})$$
$$= V^{-m}\gamma_0(m) + V^{-m+1} \sum_{j=1}^{m} (1/P_G) N(\tilde{y}_j; 0, S)\gamma_j(m). \quad \text{(A.9)}$$

Note that with the above conditioning, the validated measurements are not independent, i.e., (A.9) is not equal to the product over j of the *marginal prior densities*

$$p(\tilde{y}_j \mid m, Y^{k-1}) = V^{-1}[1 - \gamma_j(m)] + (1/P_G) N(\tilde{y}_j; 0, S)\gamma_j(m). \quad \text{(A.10)}$$

Finally, combination of (A.5)-(A.9) followed by a certain amount of rearrangement yields

$$\beta_j = \frac{\exp(-\tilde{y}_j' S^{-1} \tilde{y}_j/2)}{b + \sum_{i=1}^{m} \exp(-\tilde{y}_i' S^{-1} \tilde{y}_i/2)}, \quad j = 1, \cdots, m \quad \text{(A.11)}$$

$$\beta_0 = \frac{b}{b + \sum_{i=1}^{m} \exp(-\tilde{y}_i' S^{-1} \tilde{y}_i/2)} \quad \text{(A.12)}$$

where
$$b \triangleq (2\pi)^{M/2} C |S|^{1/2} (1 - P_D P_G)/P_D$$
$$= (2\pi)^{M/2} (CV/c_M g^M)(1 - P_D P_G)/P_D. \quad \text{(A.13)}$$

A JPDA Algorithm

Step 1: Set k=l (time index) and all the control parameters. Initialize

Step 2: the system parameters; t, **Q,** H, R.

Step 3: Either simulate the target trajectories and sensor measurements (inclv:ding clutter etc.), or in a real implementation, receive and register (scale, time tag etc.) the sensor data.

Step 4: Validate the measurements using "g-o ellipsoid" gating test (eq.(10)). Form the validation matrix Q(i):

$$\Omega_{jt} = \{w(\#_{jt})\}, \quad j=1,2,\ldots,Nm; \quad t=1,2,\ldots,Nt$$

where $W(\#_{jt}) = 1$ if meas. J 1s within the gate of target t (event "#,t" occ11rs)

$= 0$ otherwise

Step 5: For each t=l,2, ... ,Nt compute the residual:

$$y_{jt} = y(j,t) - y_{ek}$$

Step 6: Compute covariance of Y_{Jt} at k:

$$s(j,t) = H_k P_{k|k-1} H_k^T + R_k$$

Step 7: Using Bayes· formula, compute the joint event prob.

$$P_{hyp}(\#) = Pr\{\# | Z_k\}$$

$$= (C^*/c) * \Pi_{j=1}^{Nt(j)} [\, -0.5 * Y^T_{jt} S^{-1} Y_{jt} \,/\, \sqrt{\{(2\pi)^{Nm} |S(j,t)|\}}\,]$$

$$* \Pi_{t=1}^{Nc(t)} [P^t]$$

$$* \Pi^{Nc'(t)} [1-P^t]$$

where S^- = inverse of S(j,t) at scan k

and where the numbers Nt(j), Nc(t) and Nc'(t) can be calculated from the validation matrix Q.

Step 8: For each target t=l,2, ... ,Nt,

compute

$$B(j,t) = \Sigma_{i=1}^{Ncr} [P_{hyp}(i) * w(j,t;i)]$$

and for t=0, compute

$$B(0,t) = 1 - \Sigma_{j=1}^{Nm} [B(j,t)]$$

Step 9: For each t=1,2,..., Nt,

compute

$$Yt = \Sigma_{j=1}^{Nm}[B(j,t)*Y_{jt}].$$

Step 10: compute the filter gain AA_k:

$$AA_k = P_{k|k-1}\ H_k^T\ S^{-1}{}_k$$

$$S^{-1}{}_k = S^{-1}$$

Step 11: Update the state and covariance:

$$xe_{k|k} = xe_{k|k-1} + AA_k*Yt$$

$$P_{k|k} = P_{k|k-1} - AA_k\ S(j,t)\ AA_k^T$$

Step 12: Predict for the next scan data:

$$xe_{k+1|k} = \Phi_k\ xe_{k|k}$$

$$P_{k+1|k} = \Phi_k P_{k|k} \Phi_k^T + G_k Q_k G_k^T$$

$$yek+1|k = Hk+1*xek+1|k$$

Step 13: If desired, calculate and output the estimated current target position, velocity etc. (at time k) from the current state estimate, $xe_{k|k}$ or the predicted target position etc., (at time k+l) using the predicted state estimate $xe_{k+l|k}$ " The associated covariance may also be desired.

Step 14: Set **k=k+l** and go to the Step 3.

Remark ;

In the above algorithm, the joint event probabilities are computed using equation (29) which is derived under the assumption that the probability mass function (PMF) of the false measurements is given by the Poisson PMF .

$$\mu(m) = \exp[-\lambda V_k](\lambda V_k)^m\ /\ m!$$

where λ is the spatial density of false measurements(i.e., the average number per unit volume) and Vk is the volume of the validation region. Thus λV_k is the expected number of false measurements in the gate and m is the number of false measurements..

3.0 THE MULTIPLE HYPOTHESIS APPROACH

3.1 Introduction

In the sequel of this section, a track history "a" at scan k is defined by selecting a single sensor report y from each scan $j \leq k$

$$\{y(Ij;j),\ j=1,2\ldots k \mid 0 \leq Ij \leq Nj\}$$

where Nj is the number of reports at scan j and Ij=0 refers to the hypothesis that none of the sensor reports within the gate originated from the target. Hence the track history "a" is just the hypothesis that the entire sequence of measurements within "a" is correct; i.e. each sensor report y(Ij;j) originated from the target when Ij 0, while no sensor report was received when $Ij=0$, $1 \leq j \leq k$. The track history "a" at scan k is obtained from the track history "a1" at scan k-1 by selecting one of 1+Nk measurements and incorporating it into the measurement set specified by "a". In notational terms, a={a1, I}. Clearly one history "a1" at scan k-1 gives rise to 1+Nk histories "a" at scan k. Therefore, the total number of hypothesis (or histories) L_k at scan k is given by $(1+Nk)*L_{k-1}$.

3.2 The Tracking Filter Equations

The equations for target state estimation (the tracking equations) that are appropriate for use with MHT data association, are listed below. Section 1.5 above provides the notation. Appendix B explains the important hypothesis probability equation (11). The other equations come largely from Kalman filtering theory, and are very similar to the standard tracking equations of section 1.4 above.

Pr(k) = Hypothesis probability at scan k

$$= (1/c)\ (P_D)^{Nc}\ (1-P_D)^{Nt-Nc}\ (B_{ft})^{Nf}\ (B_{kt})^{Nn}$$
$$* \prod_{j=1}^{Nc} N[y_k - H_k x e_{k|k-1}, S]$$
$$* Pr(k-1) \qquad (11)$$

$$Ka_{k|k-1} = Kal_{k|k-1} + Pal_{k|k-1}H_k^T Sal_{k|k-1}[y_k - ye_{k|k-1} - H_k Kal_{k|k-1}] \quad (12)$$

where

$$S^{-1}_{k|k-1} = H_k P_{k|k-1} H_k^T + R_k$$

$$Pa_{k|k-1} = Pal_{k|k-1} - Pal_{k|k-1}H_k^T S^{-1}_{k|-1} H_k Pal_{k|k-1} \quad (13)$$

or

$$Ka_{k|k-1} = Kal_{k|k-1} \quad \text{if no measurement on the target was received in the gate.} \quad (14)$$

$$Pa_{k|k-1} = Pal_{k|k-1} \quad (15)$$

$$A_k = \Sigma_{a=1}^{Lk}[Pr(k)Ka_{k|k-1}] \quad (16)$$

= optimal tracking filter correction vector at scan k

$$Ka_{k|k} = Ka_{k|k-1} - A_k \quad (17)$$

$$Pa_{k|k} = Pa_{k|k-1} \quad (18)$$

The state and covariance update:

$$xe_{k|k} = xe_{k|k-1} + A_k \quad (19)$$

$$P_{k|k} = \Sigma_{a=1}^{Lk}[Pr(k)\{Pa_{k|k-1} + Ka_{k|k-1}Kal_{k|k-1}\}] - A_k A_k^T \quad (20)$$

One scan prediction:

$$xe_{k+1|k} = \Phi_k xe_{k|k} \quad (21)$$

$$P_{k+1|k} = \Phi_k P_{k|k} \Phi_k^T + G_k Q_k G_k^T \quad (22)$$

$$Ka_{k+1|k} = \Phi_k Ka_{k|k} \quad (23)$$

$$Pa_{k+1|k} = \Phi_k Pa_{k|k} \Phi_k^T + G_k Q_k G_k^T \quad (24)$$

HYPOTHESIS TREE AND HYPOTHESIS MATRIX

Let Ω_k be the set of association hypotheses up to time k. This set is obtained from Ω_{k-1} and the latest set of measurements Y_k as follows. New hypotheses Ω_{k-1} first measurement $y1_k$, then augmenting the resulting set by associating $y2_k$, etc. The possible association for the i-th measurement yi_k are

a. It is the continuation of a previous history (or track).
b. It is a new track (or target).
c. It is a false alarm (or clutter).

Each target can be associated with at most one current measurement, which has to fall in its validation region.

Example:

For the configuration of targets and measurements shown in Fig.1, the hypothesis tree are formed using the rule mentioned above. In the hypothesis tree shown in Fig.2, each node represents the track qualities:

"0" is the false target or false alarm.
"1,2,..." are the confirmed or new targets.

The following figure 2 shows the hypothesis tree representation of the hypothesis formation technique outlined above. Each node of the tree represents an alternative hypothesis; further branches are added to each node as a new measurement point is considered.

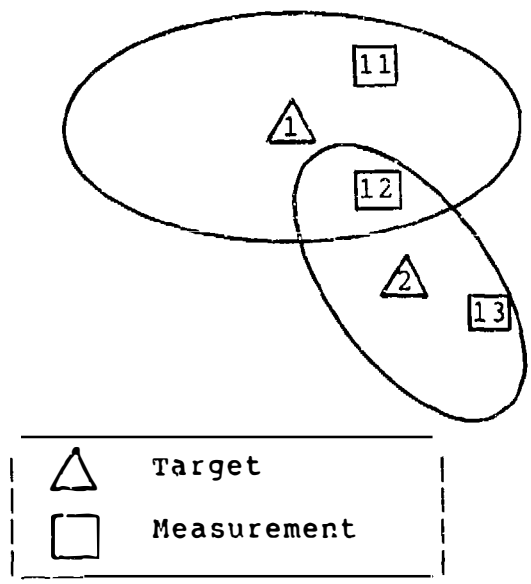


Figure A1. Configuration of Targets and Measurements.

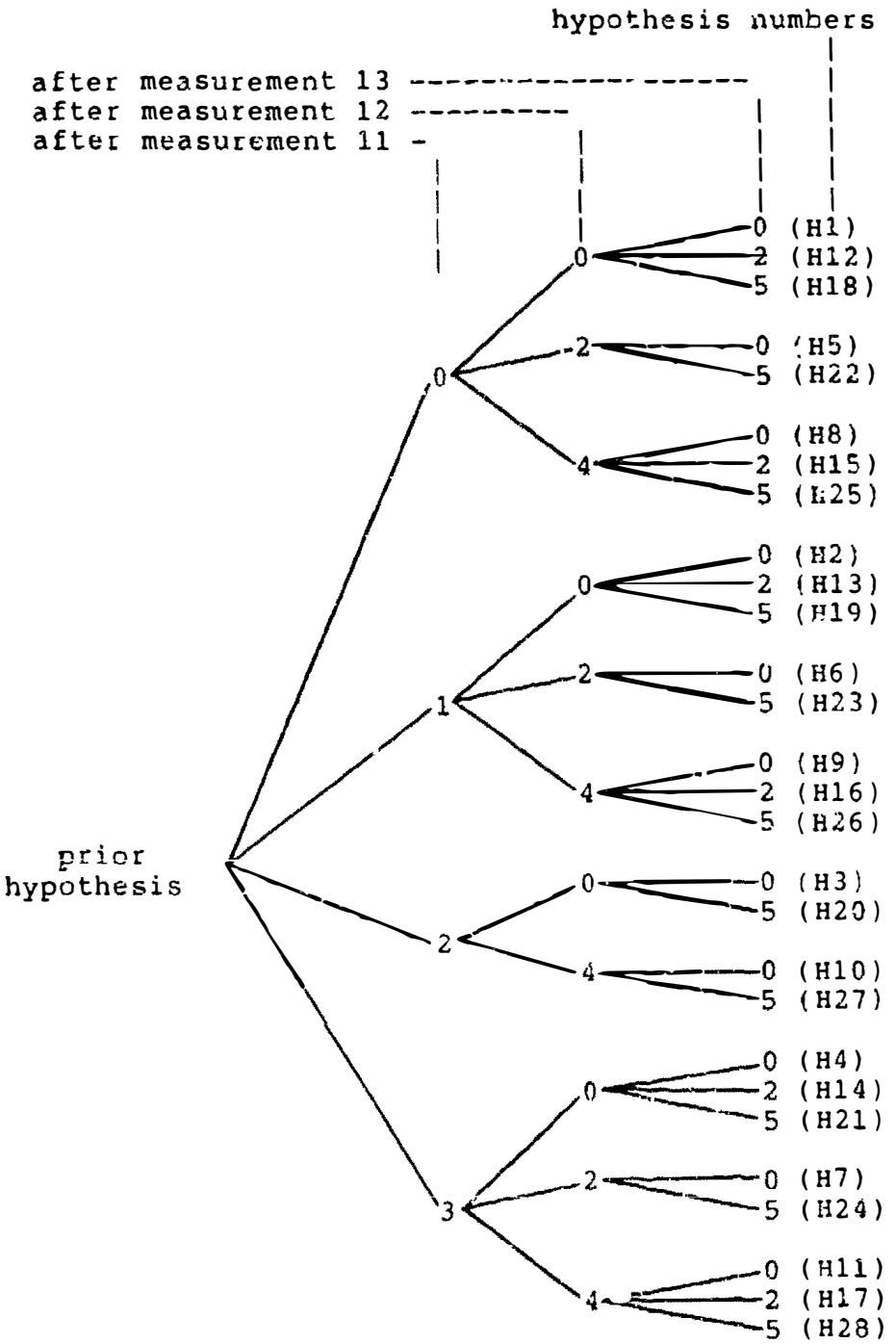


Figure A2. Hypothesis Tree for Configuration in Figure A1.

The hypothesis matrix corresponding to the tree can be formed as follows.

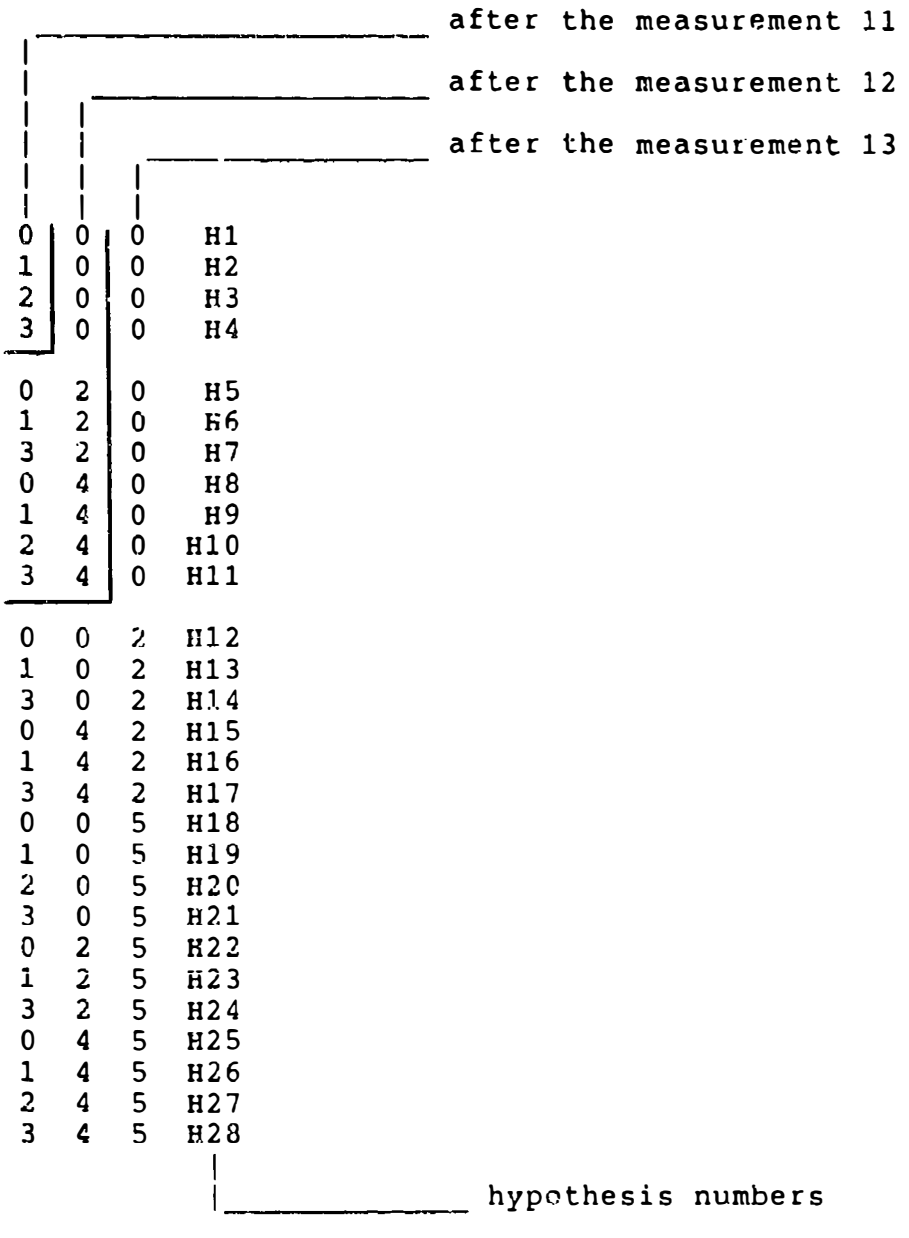


Figure A3. Hypothesis matrix corresponding to the hypothesis tree.

HYPOTHESIS PROBABILITIES

In the equation for hypothesis probability in section 3.2 above (eq. (11)), an expression suggested by Reid (eq. (16) of ref. [11]) is used. Further, the formula is valid for type 1 sensors such as a scanning radar (which provide a "complete" picture of some coverage area in each data set, and hence can provide number-of-target information). If type 2 sensors such as ESM or a tracking radar are assumed (which in each report only provide information about individual targets), some modifications have to be done as given in [11]. Similar expressions for computation of hypothesis probabilities in the non-recursive form can be found in Bar-Shalom [2].

The equation (11) can be easily implemented since it is given by a recursive form. If all the prior hypotheses are first multiplied by $(1-P_D)^{Nt}$, then as a branch is created for each measurement and its hypothesized origin, the likelihood of the branch is found by multiplying the prior probability by either B_{ft}, B_{kt}, or

$$P_D * N[y_k - H_k x e_{k|k-1}, S]/(1-P_D)$$

as appropriate. After all such branches are generated the likelihoods are then normalized.

Example

In Figure B2, the hypothesis probabilities are computed for the hypothesis tree shown in figure B1, in terms of its prior hypothesis probability and the probability of each branch.

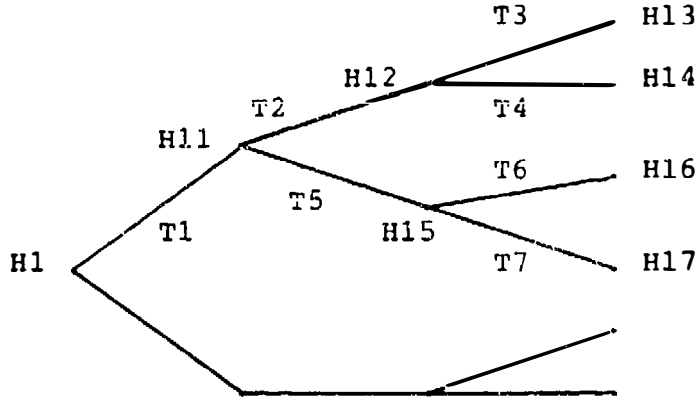


Figure B1. Hypothesis Tree.

Pr(H11) = Pr(H1)*Pr(T1)
Pr(H12) = Pr(H11)*Pr(T2)
Pr(H13) = Pr(H12)*Pr(T3)
Pr(H14) = Pr(H12)*Pr(T4)
Pr(H15) = Pr(H11)*Pr(T5)
Pr(H16) = Pr(H15)*Pr(T6)
Pr(H17) = Pr(H15)*Pr(T7)

_____ Prob. of each track.
_____ Prob. of prior hypothesis.
_____ Prob. after new measurements.

Figure B2. Hypothesis Tree Branch Probabilities.

In figure B2 the probability of each track is calculated using the simple rule in accordance with the track quality:

1/ If the measurement belongs to false alarm (or false target), the probability of the track(or branch) is given by

2/ If the measurement "j" belongs to i-th track, the corresponding probability is given by $P_D*N[0,S(i)]$, where N[a,b] is the Normal (Gaussian) density function associated with the assignment of the j-th observation to the i-th track of the prior history and is defined by

(B1)

for the residual matrix S(i) and normalized distance function d_{ij}.

3/ If an observation belongs to new target, the corresponding probability is computed by

$$B_{kt}*(1-P_D)/c.$$

Remark:

If the track deletion option is included, then the factor $e^{-1/De}$ should be properly introduced in the probability calculations (as per reference [4] pp. 255-260).

HYPOTHESIS REDUCTION

Since, in the worst case, the number of hypotheses can grow exponentially with time, there is a clear need to limit this number. Some schemes for doing so include:

a. The first opportunity for limiting hypotheses is to require an observation to satisfy a gating relationship before any of the possible track associations are to be considered to be potentially valid.

b. The JPDA algorithm, by computing all the measurements at the current time, considers only the number of known targets with a single hypothesis per target.

c. The "N-scan-back" concept, by combining all histories that have common measurements from k-N to k.

d. An alternative method is to combine those hypotheses that similar effects, i.e., same number of targets but with slightly different state estimates. The mean and covariance of the resulting estimate is a combination of the individual estimates and covariance. At the same time, hypotheses with negligible probabilities are eliminated.

The method c has been found to be most effective in practice, and it provides an efficient approximation of the method d. The process of pruning hypotheses is highly dependent on the applications. Typical ways of pruning can be summarized as:

1) To remove hypotheses with probabilities that fall below some predetermined threshold. Disadvantage of this method is that it does not take into account the computational capacity.

2) To allow only a predetermined number, say M, of hypotheses to be maintained by ranking the hypotheses and choosing only the M most likely ones, as measured either by the probabilities or the score functions.

3) To rank and sum the probabilities of the more likely hypotheses. When this sum exceeds a threshold the remaining hypotheses are then eliminated.

CLUSTERING

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

The steps in clustering are as follows. Initially, one cluster is set up for each confirmed target. Each new measurement is associated with a cluster if it falls in the validation region of any track from that cluster. A new cluster is initiated any time an observation is received that does not fall within the gates of any track contained in an existing cluster. The cluster is initiated on the observation using the alternatives (true target or false alarm) associated with its source. In order that clusters remain distinct, the gates of the tracks within the clusters must not overlap. Thus, when an observation falls within the gates of two or more tracks from different clusters, the clusters are merged. The merging must be done before the observation is processed. If an observation is associated with more than one cluster, then those clusters are combined into a super-cluster. If tracks within a cluster separate spatially and have no more common measurements, the corresponding cluster is subdivided accordingly into smaller clusters that can be handled separately.

MHT Algorithm

This section outlines the basic algorithm for implementation of the Multiple Hypothesis Testing method for multi-target tracking, either in a simulation environment or in real time. First the algorithm is described verbally, followed by a top level flowchart.

For notations and definitions, see section 1.5. In an actual implementation the simulated quantities are received from sensors.

STEP 1: Set scan counter k=1 and all the control parameters.

STEP 2: Initialize the track and system parameters;

STEP 3: Simulate target trajectories.

STEP 4 Simulate measurements.

STEP 5: For each history (or hypothesis) "a1", $1 \leq a1 \leq L_{k-1}$, receive unconditional estimation parameters (from STEP 11):

STEP 6: Form gate about

and select the Nk sensor reports to be used in filter updating. A measurement is valid if

$$(y_k - ye_{k|k-1})^T S^{-1}_k (y_k - ye_{k|k-1}) \leq g^2$$

(this is a "g-σ" elliptical gate).

STEP 7: (1) Identify new histories (or hypotheses) "a", for

$$1 \leq a \leq (1+Nk)*L_{k-1}.$$

(2) For each hypothesis, compute Pr(k) using equation (11). (see Appendix B for details.)

STEP 8: For each remaining history (or hypothesis) "a", and each track within "a"
Compute $Ka_{k|k-1}$ using equation (12).
Compute $Pa_{k|k-1}$ using equation (13).

STEP 9: For each history (or hypothesis) "a",
Calculate A_k using equation (16).
Calculate $Ka_{k|k}$ using equation (17).
Calculate $Pa_{k|k}$ using equation (18).

STEP 10: Compute new optimal updated estimates $xe_{k|k}$ using eq. (19). Compute error covariance $P_{k|k}$ using equation (20).

STEP 11: Predict for next scan data:
Compute $Ka_{k+1|k}$ using equation (23).
Compute $Pa_{k+1|k}$ using equation (24)
Compute $xe_{k+1|k}$ using equation (21).
Compute $P_{k+1|k}$ using equation (22).

STEP 12: GO TO STEP 5 for next scan data and/or output the results.

Figure 1 below provides a top level flow chart of this MHT process, showing the sequence of events. The important function of "pruning" the hypothesis tree (to reduce the number of hypothesis to consider) is described very briefly in appendix C. The clustering process, also to reduce the number of hypothesis to consider, is described in appendix D.

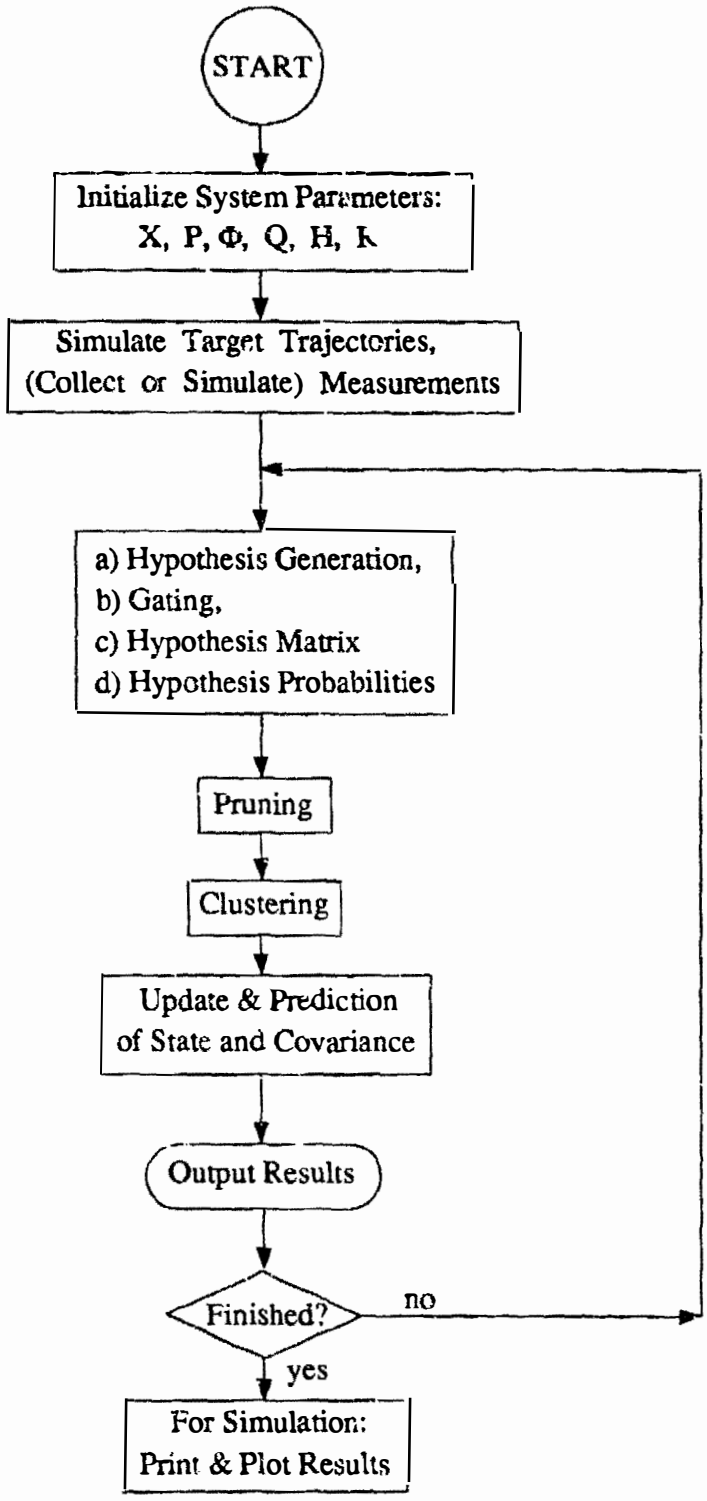


Figure 1. Multiple Hypothesis Test Flowchart.

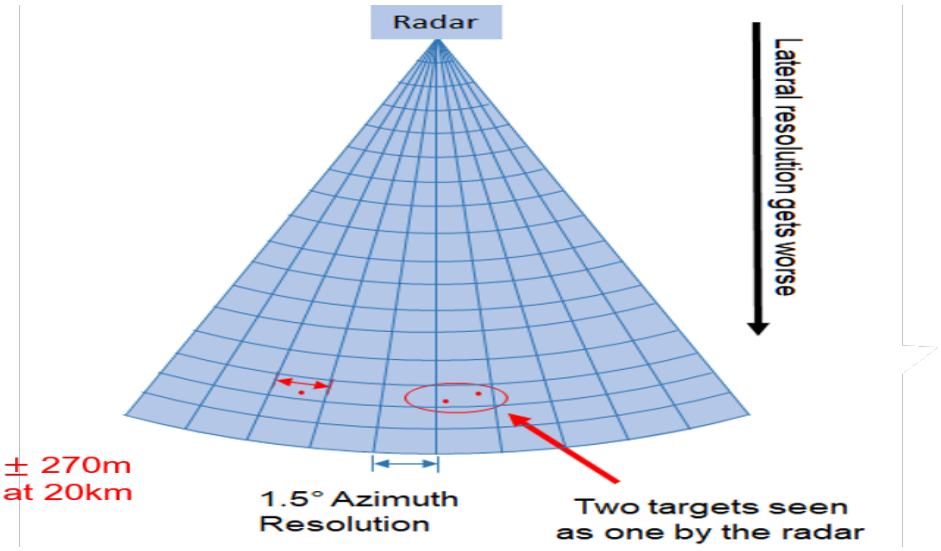
4. Performance Evaluation of Data Association Algorithm Methods

Data association. This is a comparatively complicated evaluation index. Data correlations are normally evaluated using the data under various circumstances – such as dense target environments, cross-target environments target approaching and leaving maneuvering multi-target environments, etc.

4.1. Scenario Generation and Results

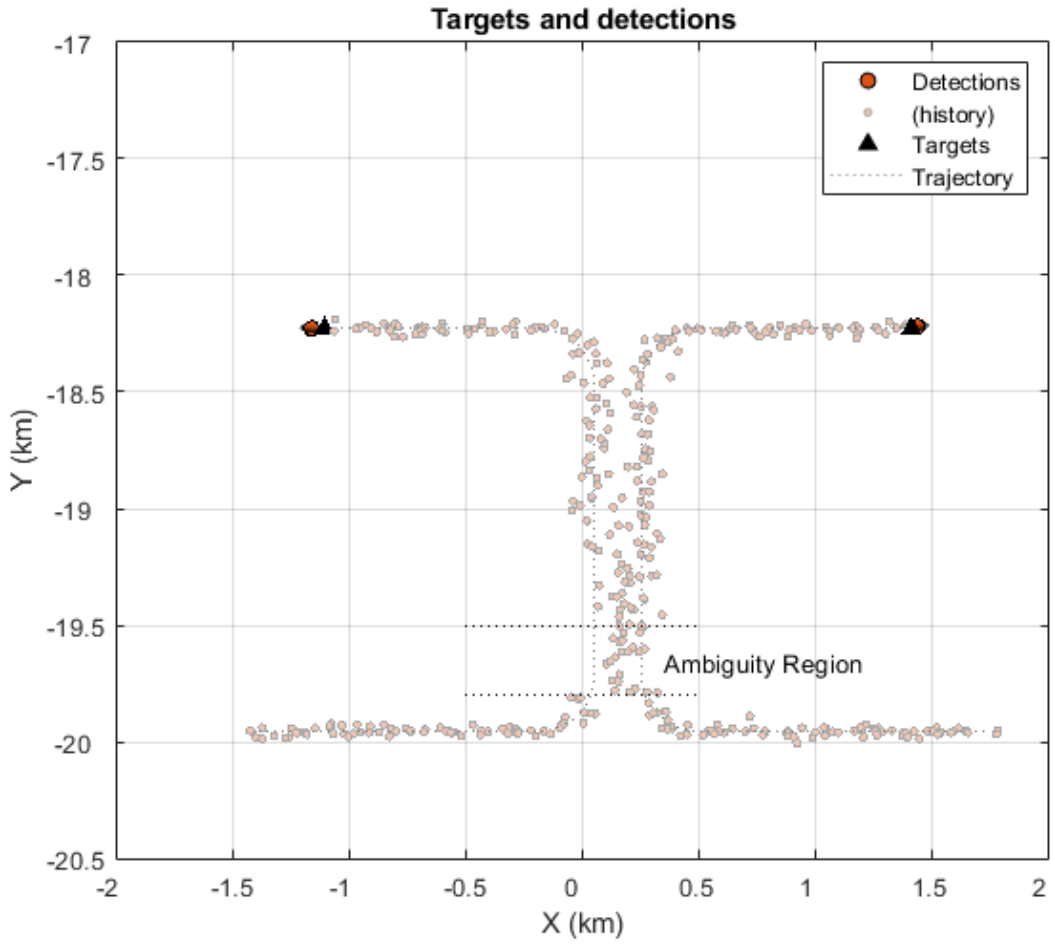
4.1.1 scenario : Tracking Closely Spaced Targets Under Ambiguity

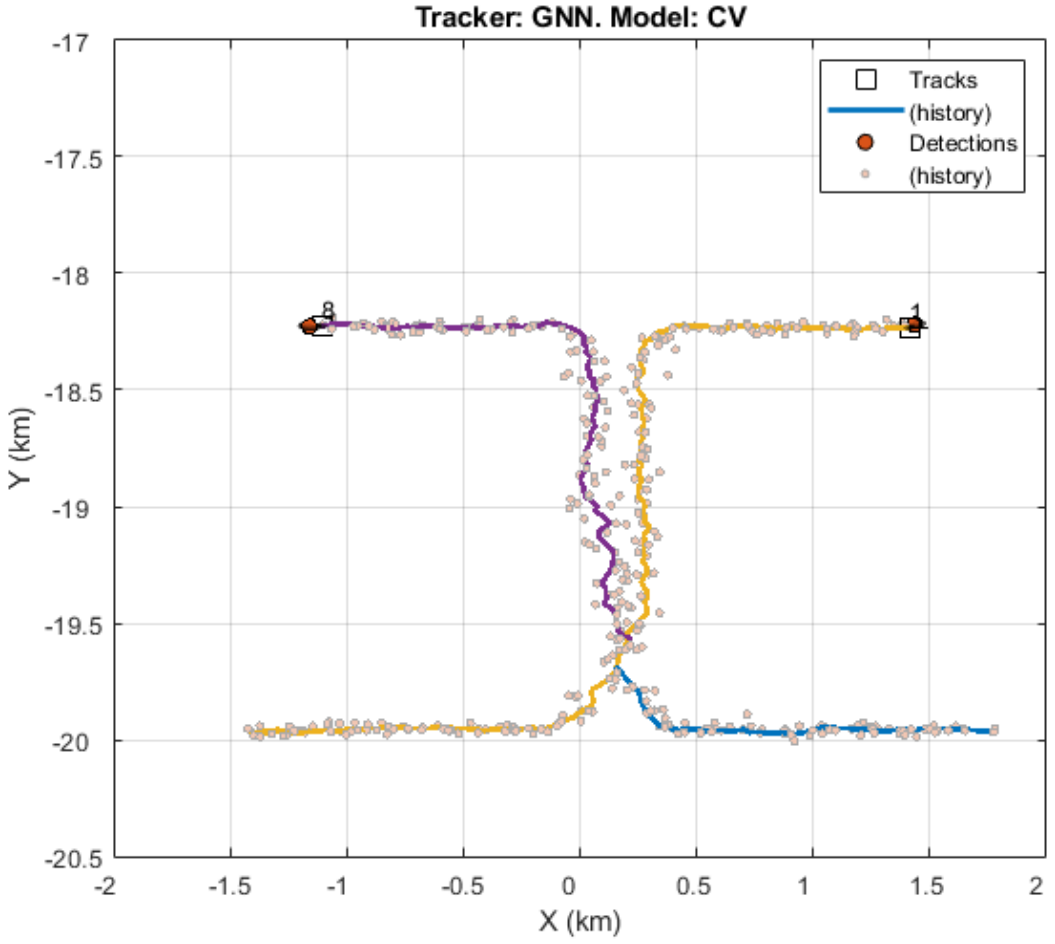
To demonstrate a case where sensor reports are ambiguously assigned to tracks, we create a simple scenario. In this scenario, a single mono-static Radar Sensor object, located at the origin (not shown), scans a small region about 20 km from the radar. Initially, the radar reports about two detections per scan. When the detections are coming from a region around the X = 0, Y = -20 km position, the radar reports a single detection per scan for a while, followed by two radar detections reported from around Y = -19.5km and toward the sensor (up).

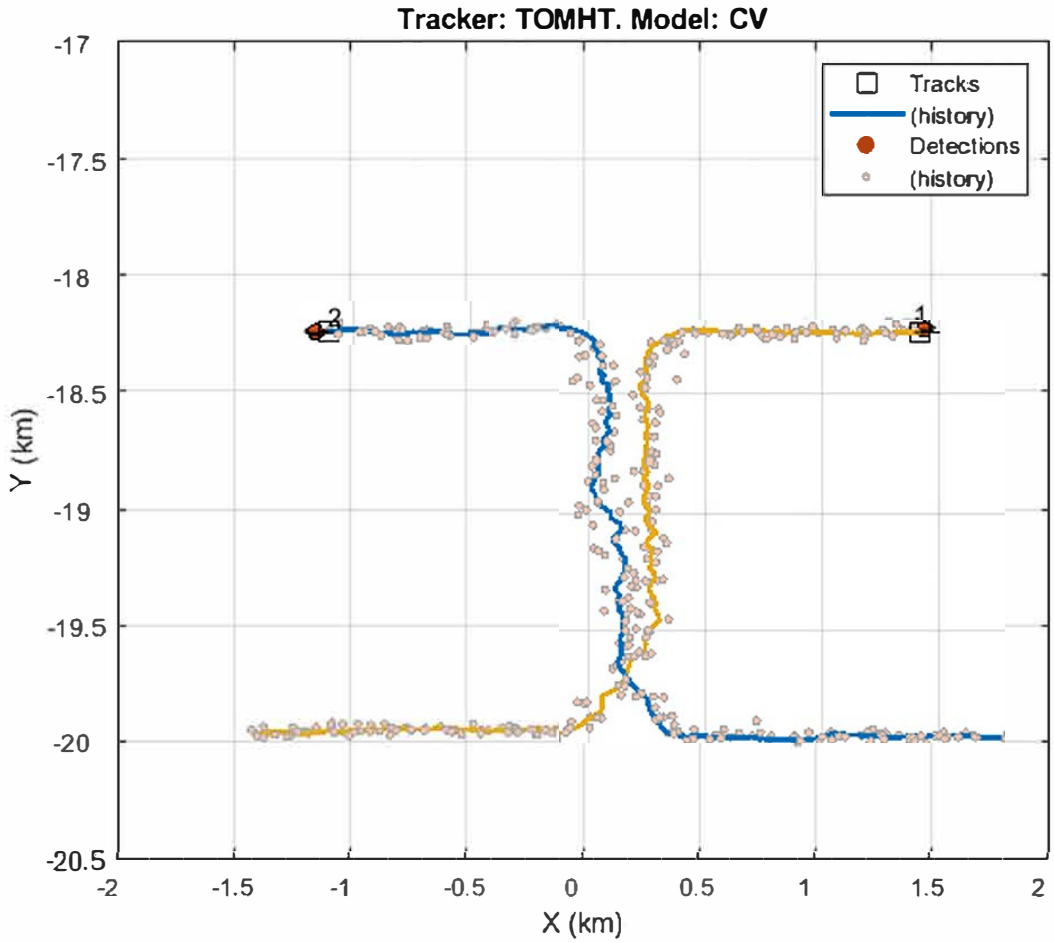


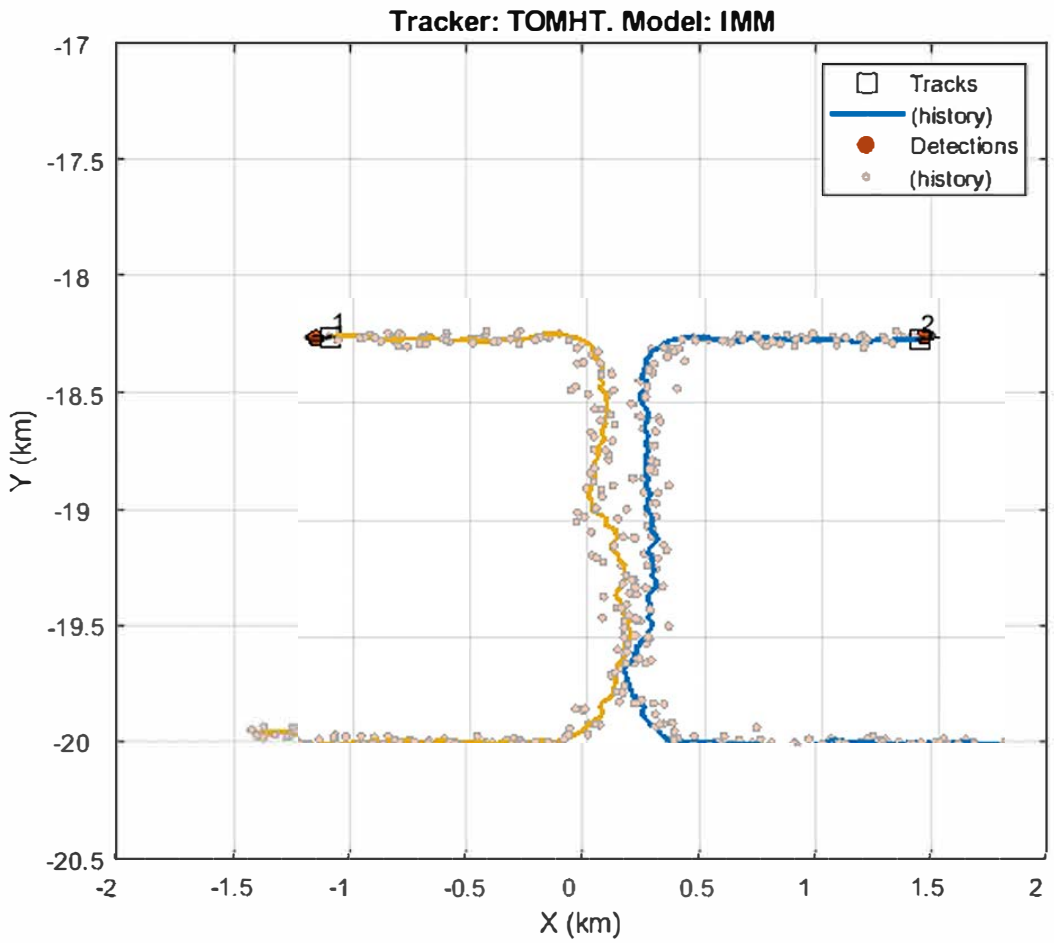
- This example shows how to track objects when the association of sensor detections to tracks is ambiguous. In this example, you use a single-hypothesis tracker, a multiple-hypothesis tracker, and a probabilistic data association tracker to compare how the trackers handle this ambiguity. To track, the manoeuvring targets better, you estimate the motion of the targets by using various models.

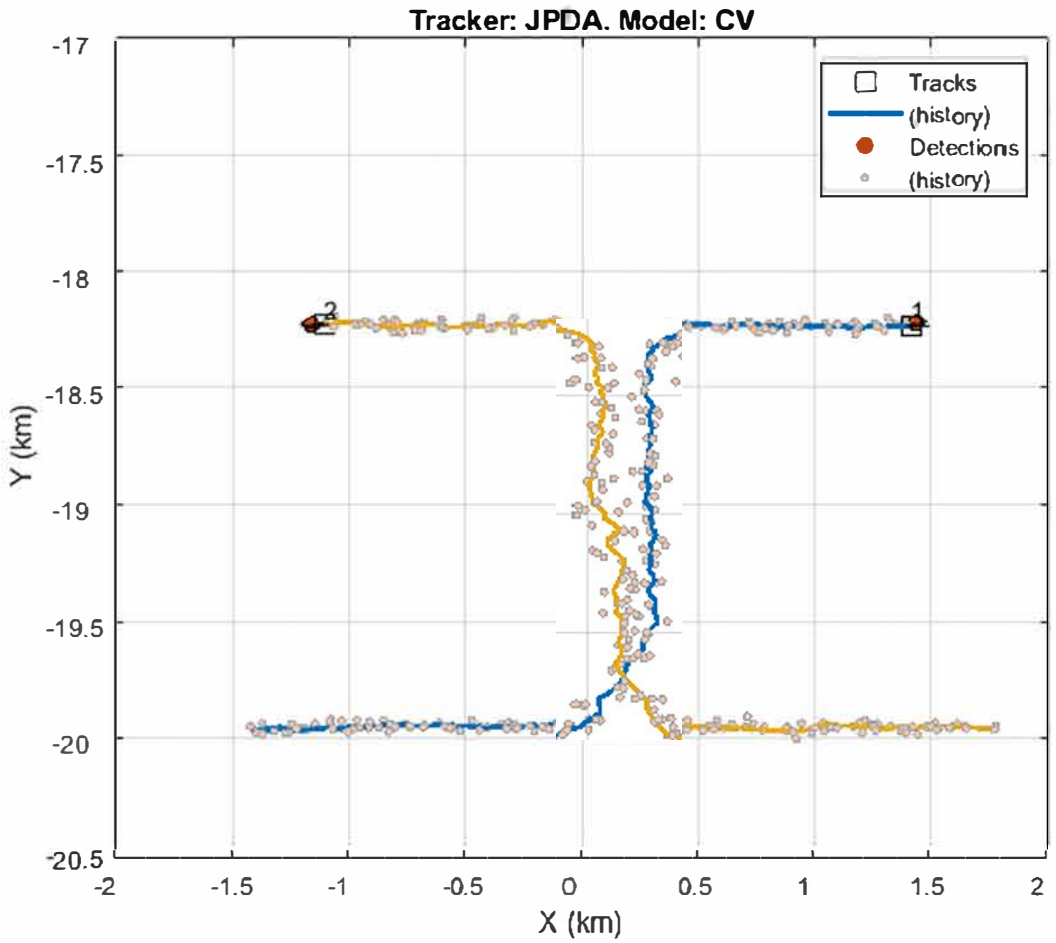
Scenario MATLAB simulation results

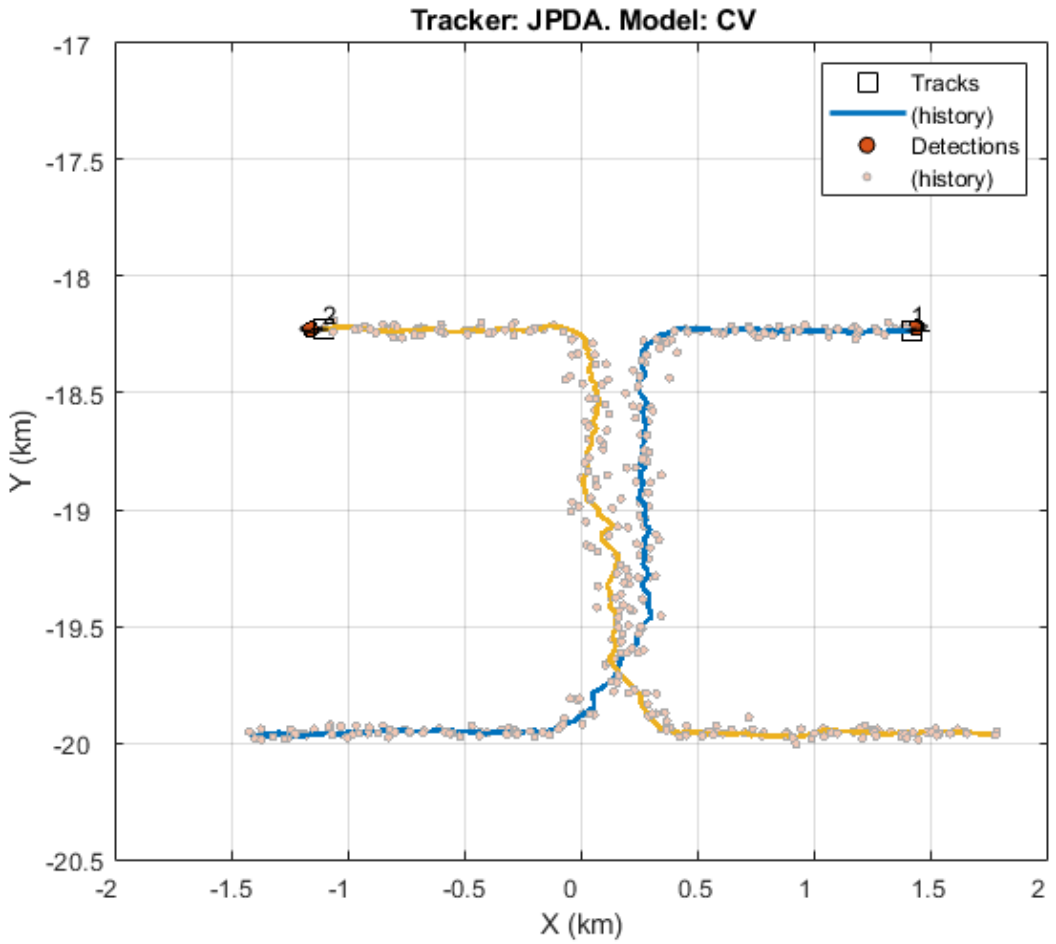


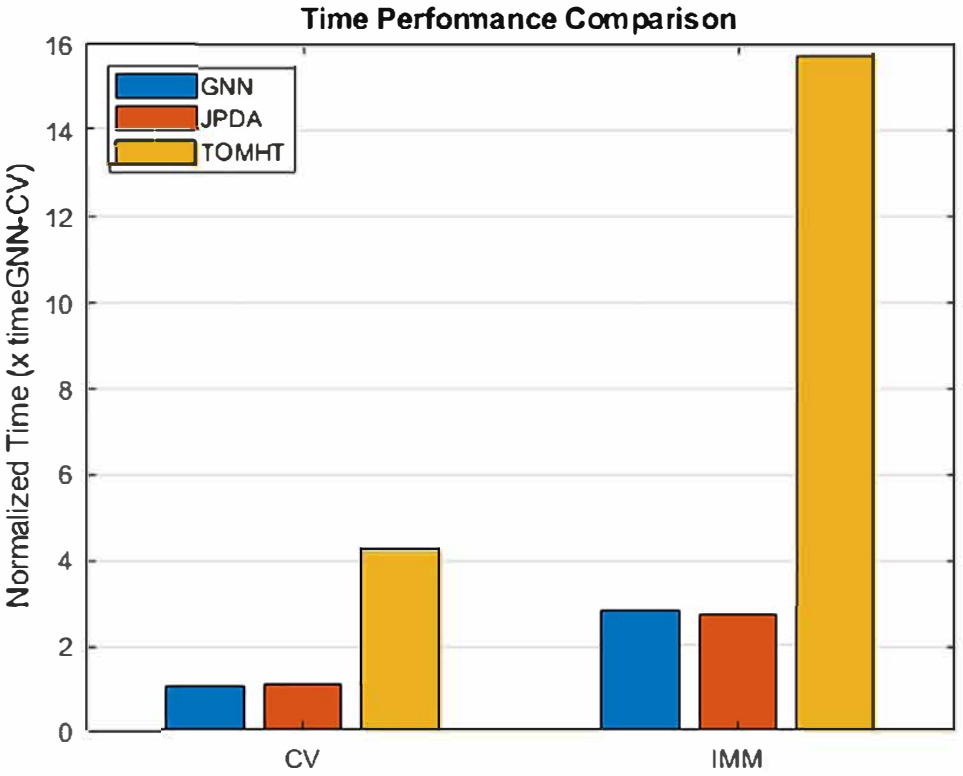



Tracker: TOMHT. Model: CV
Tracks
(history)
Detections
(history)
Y (km)
X (km)
-17
-17.5
-18
-18.5
-19
-19.5
-20
-20.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
1
2


Tracker: TOMHT. Model: IMM
Tracks
(history)
Detections
(history)
Y (km)
X (km)
-17
-17.5
-18
-18.5
-19
-19.5
-20
-20.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
1
2


Tracker: JPDA. Model: CV
Tracks
(history)
Detections
(history)
1
2
-17
-17.5
-18
-18.5
-19
-19.5
-20
-20.5
Y (km)
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
X (km)


Tracker: JPDA. Model: CV
Tracks
(history)
Detections
(history)
1
2
-17
-17.5
-18
-18.5
-19
-19.5
-20
-20.5
Y (km)
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
X (km)



The results show that GNN and JPDA can track the targets 5 to 6 times faster than MHT depending on the motion model. The IMM motion model makes all three trackers run 3 to 4 times slower. Note that each tracker processing time varies differently depending on the scenario's number of targets, density of false alarms, density of targets, etc. This example does not guarantee similar performance comparison in different use cases.

Summary

In this example, we created a scenario in which two maneuvering targets are detected by a single sensor, where some of their motion is within an area of ambiguity. we used six combinations of trackers and filters to show the contribution of each to the overall tracking. We observed that the constant velocity filter was less accurate in tracking the targets during their maneuverer, which required an interacting multiple-model filter. We also observed the ability of MHT and JPDA to handle the case of ambiguous association of detections to tracks, and how it can be used to maintain continuous tracking while a single-hypothesis tracker cannot do that. Finally, we noticed the trade-offs between tracking results and processing time when choosing a tracker. In this case JPDA proves to be the best option. In different scenarios, we may require the more complex MHT when neither GNN nor JPDA gives acceptable tracking results. we may as well prefer GNN if there are less ambiguity regions or low clutter density.

5.0 CONCLUSIONS

In this report three quite different algorithms for multiple target tracking in a clutter environment are presented: Nearest Neighbour Standard Filter (NNSF), Multiple Hypothesis Test (MHT) and Joint Probabilistic Data Association (JPDA). For further development towards implementation of the algorithm, the following comments and suggestions are offered.

Comments:

1) The main drawback of the NNSF algorithm is that the tracking performance or accuracy of the filter may become very poor in a dense target environment because of possible misassociation (choosing an incorrect measurement for processing by a target tracking filter). It's main advantage is that it is easy to implement and computationally feasible.

2) In the MHT method, the main drawback of the algorithm is that in a dense target environment the number of hypotheses can increase exponentially with each scan, leading to computational burden problems. Hence, for implementation, the development of an efficient way of pruning the hypotheses tree is necessary.

3) Another disadvantage of the MHT method is that the data association decision is often deferred, and thus a single best estimate for the target tracking solution is not always available. If target execution is desired before the MHT solution is resolved, this can be a problem. This problem can likely be overcome by providing a NNSF-type solution at all times (using the most probable current hypothesis: ie. using all measurements up to the current time).

4) One advantage of the MHT method is that it provides a systematic track initiation procedure.

5) Another advantage of the MHT method is that it is most likely to have the correct association solution as one of it's hypothesis (hopefully the hypothesis assigned highest probability).

6) The JPDA algorithm is a non-back-scan (or zero-scan) approach, meaning that all hypotheses are combined after computation of the probabilities, for each target at each time step. One problem with this method is that it implies that incorrect measurements are routinely (and purposely) used by the tracking filters, albeit hopefully with a lower weighting than the correct measurement. This method does however have the advantage of being more computationally efficient than an n-scan algorithm in a heavy

clutter environment such as sonar tracking.

7) Another difficulty with the JPDA method is in the implementation of track initiation. Since the MHT algorithm provides a systematic track initiation procedure, an effective way of combining these two methods may be of interest.

1) Comparisons of the accuracy performance of these different algorithms for various target dynamic and clutter models.

2) Comparison of the computational burdens of these algorithms for various target densities and clutter models.

3) Investigation of ways and means of overcoming the shortcommings of these methods, especially pruning for the MHT method, and providing real time output from the MHT method.