

12

AI&DS

AI&DS-TR-1015-1
December 7, 1982

AD A122618

DISTRIBUTED HYPOTHESIS FORMATION IN DISTRIBUTED SENSOR NETWORKS

TECHNICAL REPORT

A GENERAL THEORY FOR BAYESIAN MULTITARGET TRACKING AND CLASSIFICATION - GENERALIZED TRACKER/CLASSIFIER (GTC)

C.Y. Chong
S. Mori
E. Tse
R.P. Wishner

DTIC
ELECTE
DEC 20 1982
S B

Prepared for

Dr. Barry M. Leiner
DARPA
1400 Wilson Boulevard
Arlington, VA 22209

Approved for public release; distribution unlimited.

ADVANCED INFORMATION & DECISION SYSTEMS

Mountain View, CA 94040

82 12 20 042

FILE COPY

AI&DS

ADVANCED INFORMATION
& DECISION SYSTEMS

201 San Antonio Circle, Suite 286
Mountain View, CA 94040-1270
(415) 941-3912

AI&DS TR-1015-1
December 7, 1982

DISTRIBUTED HYPOTHESIS FORMATION IN DISTRIBUTED SENSOR NETWORKS

TECHNICAL REPORT

A GENERAL THEORY FOR BAYESIAN MULTITARGET TRACKING AND CLASSIFICATION - GENERALIZED TRACKER/CLASSIFIER (GTC)

C.Y. Chong
S. Mori
E. Tse
R.P. Wishner

Sponsored by

Defense Advanced Research Projects Agency (DoD)
ARPA Order No. 4272

Under Contract No. MDA903-81-C-0333 issued by
Department of Army, Defense Supply Service-Washington,
Washington, DC 20310.

The views and conclusions contained in this document
are those of the authors and should not be interpreted
as representing the official policies, either expressed
or implied, of the Defense Advanced Research Projects
Agency or the U.S. Government.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AI&DS TR-1015-1	2. GOVT ACCESSION NO. AD-A222628	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A General Theory for Bayesian Multitarget Tracking and Classification - Generalized Tracker/Classifier (GTC)		5. TYPE OF REPORT & PERIOD COVERED Technical Report July 15, 1981 - July 1, 1982
7. AUTHOR(s) C.Y. Chong, S. Mori, E. Tse, R.P. Wishner		6. PERFORMING ORG. REPORT NUMBER
3. PERFORMING ORGANIZATION NAME AND ADDRESS Advanced Information & Decision Systems 201 San Antonio Circle, Suite 286 Mountain View, CA 94040		8. CONTRACT OR GRANT NUMBER(s) MDA903-81-C-0333
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, VA 22209		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS ARPA Order No. 4272
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Defense Supply Service - Washington Room 1D 245, The Pentagon Washington, DC 20310		12. REPORT DATE December 7, 1982
		13. NUMBER OF PAGES 100
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Multitarget Tracking Surveillance Correlation Multisensor Surveillance System Tracking Track Correlation Data Association Multitarget Track Formation		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) A general theory for the tracking and classification of multiple targets using a Bayesian approach is presented, together with its specialization to independent, identically distributed target models. The implementation of the theory is through the Generalized Tracker/Classifier. Simulation results to illustrate the algorithm are also given.		

DD FORM 1473

EDITION OF 1 NOV 55 IS OBSOLETE
S/N 0102-014-66011

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
SUMMARY	iv
1. INTRODUCTION	1
2. TARGET AND SENSOR MODELS	8
2.1 Target Model.	8
2.2 Sensor Model.	10
3. TRACKS AND HYPOTHESES	14
4. RECURSIVE BAYESIAN EVALUATION OF HYPOTHESES	17
4.1 Computation of $p(\lambda Z(k,s))$	17
4.2 Computation of $p(dX N_T, Z, \lambda)$ and $p(N_T Z, \lambda)$	21
5. INDEPENDENT, IDENTICALLY DISTRIBUTED TARGET MODELS	23
5.1 Assumptions on Models	23
5.2 Hypothesis Evaluation Algorithm	24
6. RELATION TO OTHER RESULTS	31
7. HYPOTHESIS MANAGEMENT TECHNIQUES	35
7.1 Pruning	36
7.2 Combining	39
7.3 Windowing	43
7.4 Clustering	45
8. IMPLEMENTATION	49
9. AN EXAMPLE - TRACKING WITH AN MTI RADAR SYSTEM	53
9.1 Scenario Description	54
9.2 Likelihood Functions and Filtering	56
9.3 Hypothesis Management	65
9.4 A Sample Run	67
9.5 Monte Carlo Simulations	89
10. CONCLUSIONS	96
REFERENCES	98

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1-1	Distributed System	2
1-2	Generalized Tracker/Classifier	5
7-1	Pruning Strategies	38
7-2	N-scan Hypothesis Combining	42
8-1	Hypothesis Tree	51
9-1	An MTI Tracking Example.	55
9-2	Probability of Detection by MTI Radar.	55
9-3	Road Model	57
9-4(a)	Density of Undetected Targets (up to Scan 2)	60
9-4(b)	Density of Undetected Targets (after Scan 2)	61
9-5(a)	L_{00} Versus Scan	63
9-5(b)	ν Versus Scan	63
9-5(c)	New Target Likelihood (as a function of range)	64
9-6	A Sample Run - Targets and Measurements	68
9-7(a)	Cluster 1	70
9-7(b)	Clusters 2 and 3 Before Split.	71
9-7(c)	Clusters 2 and 3 After Split at Scan 4	73
9-7(d)	Clusters 4 and 8	74
9-8(a)	Hypotheses/Tracks (Scan 0 and 1)	75
9-8(b)	Hypotheses/Tracks (Scan 2)	77
9-8(c)	Hypotheses/Tracks (Scan 3)	80
9-8(d)	Hypotheses/Tracks (Scan 4)	82
9-8(e)	Hypotheses/Tracks (Scan 5)	84
9-8(f)	Hypotheses/Tracks (Scan 6)	86
9-8(g)	Hypotheses/Tracks (Scans 7 - 9).	87
9-9	Clusters (Labelled by Number).	88
9-10	A Posteriori Probability of Best Hypothesis.	90



Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By _____	
Distribution/	
Availability Codes	
Avail. Statement	
Dist	Special
A	

LIST OF TABLES

<u>Table</u>		<u>Page</u>
9-1	Cases for Monte Carlo Simulation	92
9-2	Monte Carlo Simulation Results	93

SUMMARY

The DSN project at AI&DS is concerned with the formation and evaluation of hypotheses for situation assessment in a distributed manner by the processing nodes in a DSN. In particular, the tracking and classification of multiple targets is chosen to understand the basic technical issues since the uncertain origin of the measurements for such problems requires the formation of data association hypotheses. Our approach involves the use of techniques from both estimation theory and artificial intelligence.

Little has been done on distributed multitarget tracking and classification. Even in the centralized case, the existing results are not really good enough to provide a sound foundation for developing distributed techniques. We have thus investigated the centralized multitarget tracking problem. This report contains a general theory for multitarget tracking and classification using a Bayesian estimation-theoretic approach. The models assumed are very general and are capable of handling complex situations such as targets moving in groups or sensors with a limited field of view. When specialized to independent, identically distributed target models, which are more widely studied, the theory yields algorithms which are similar to, but more general than other existing algorithms. The various terms in the evaluation algorithm have intuitive explanations which reflect the target and sensor characteristics.

For the practical implementation of this algorithm, hypothesis management techniques (pruning, combining, etc.) have to be used to control the rapid growth in the number of hypotheses. Some new techniques are proposed, but a basic theory is still lacking. This is an area where ideas from artificial intelligence can be useful.

All this has been implemented in a system called the Generalized Tracker/Classifier (GTC). It contains the following four on-line modules. The hypothesis formation module generates the data-to-data association hypothesis which identifies the data reported at different times from different sensors with the same targets. The hypothesis evaluation module evaluates the probability of each hypothesis given the information available. We have developed recursive evaluation formulae which reduce to the standard formulae in the literature under simplifying assumptions. The filtering and parameter estimation module solves the tracking and classification problem given a particular data-association hypothesis. If the targets are independent and identically distributed, then the algorithm decomposes into a parallel bank of single target algorithms. The hypothesis management module controls the (asymptotically) exponential growth in the number of hypotheses.

The GTC can be used by itself for centralized multitarget tracking and classification. It is also used to process the local sensor data at a node and, together with other modules for information distribution and fusion, is a basic component of each processing node in the DSN. This will be reported in a subsequent report.

Simulation runs for a scenario with terrain masking of targets are used to demonstrate the hypothesis management techniques and the general feature of the GTC.

1. INTRODUCTION

The distributed hypothesis formation in DSN project at AI&DS is concerned with the formation and evaluation of hypotheses for situation assessment in a distributed manner at the processing nodes in a DSN. The basic system model assumes the presence of a distributed system of sensor/processing nodes, that nodes have one or more sensor types, have overlapping sensor coverage and are connected in a packet switching network. The input information at each node consists of:

- own sensor information
- neighbor nodes hypotheses
- contextual information

While the system can also be used for distributed planning and control (see Figure 1-1), our focus in this project is on distributed situation assessment. Our main thrust is to understand the technical issues for such distributed systems through the analysis of idealized problems using techniques from both estimation theory and artificial intelligence.

One area where a DSN is useful is the tracking and classification of multiple targets. We have chosen this problem as a vehicle for understanding the basic issues. Despite the obvious importance of this problem and some applied work [1], very little has been done towards developing a theory of distributed multitarget tracking and classification. (Some attempts have been reported in [2], [3].) Even if we consider only the centralized case, the existing results are not good enough to form a basis for the development of distributed algorithms. We thus have searched for a better theory for centralized multi-target tracking. This report summarizes the results of our investigation.

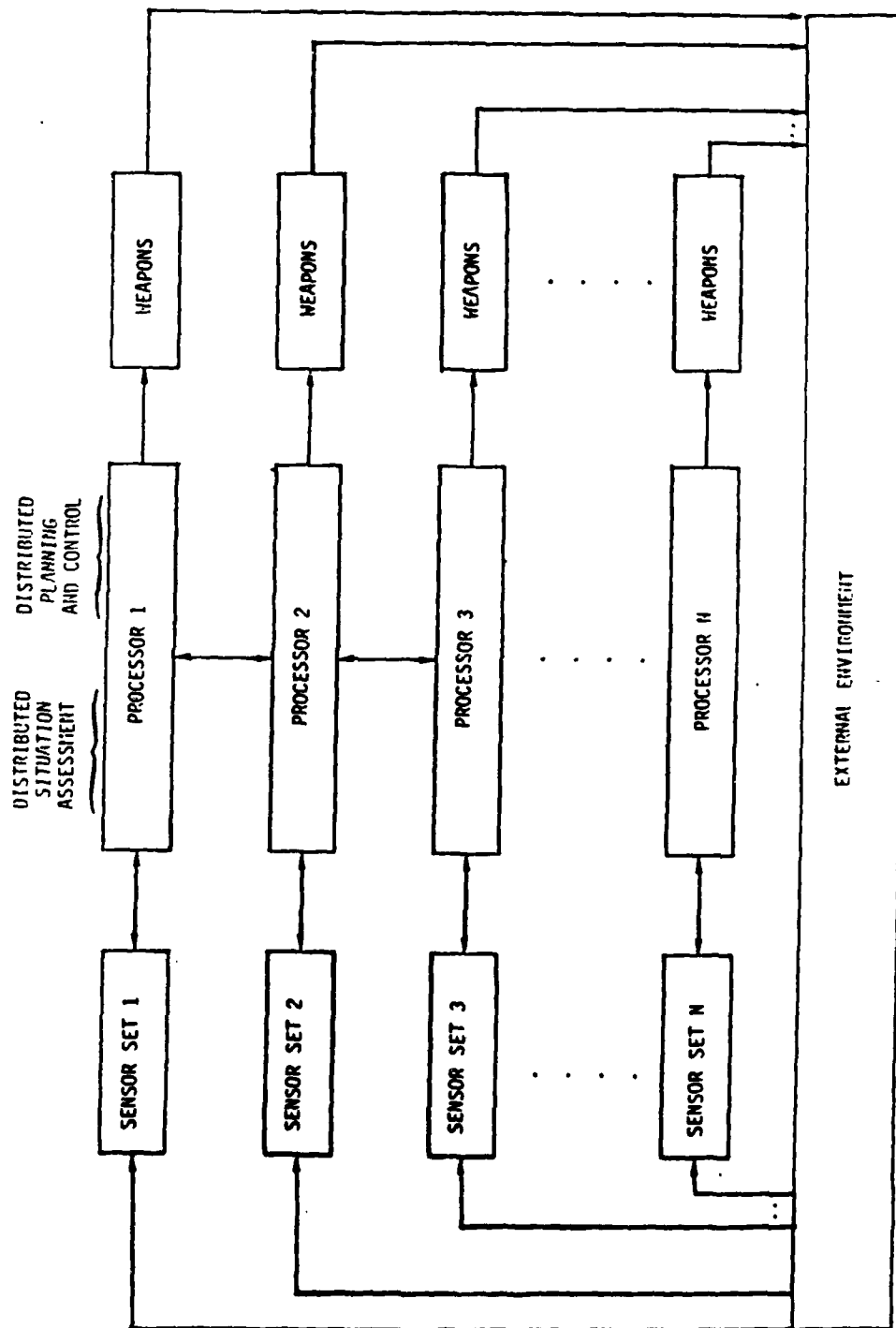


Figure 1-1 Distributed System

The multitarget tracking problem arises in many application areas such as ocean surveillance, air traffic control and air defense, etc. A key feature in such problems is that the origins of the measurements may be uncertain. As a result, it is necessary to associate the measurements with the target tracks before any estimation of the targets can be performed. Many algorithms for multitarget tracking have been suggested [4]-[21]. Surveys of the area can be found in the survey paper by Bar-Shalom [22] and the Naval Ocean Surveillance Correlation Handbooks [23],[24]. The recent paper by Reid [20] also contains a good survey of the then existing methods. Thus we will not perform a detailed survey here. Most of the existing work deals with approximate algorithms which are only applicable to specific situations. There have been some attempts to search for a unified theory [25], but the results are still not very satisfactory. Reid's paper is a step in this direction and has motivated the work of Keverian and Sandell [18].

The multitarget tracking problem is different from ordinary estimation problems in that the origins of the measurements are in general uncertain. In other words, the order in which the measurements are obtained does not contain any information about their relationship to the targets. Thus the measurements should really be considered as elements in a set rather than as elements in an ordered tuple as in the case of an ordinary estimation problem. In order to use ordinary estimation techniques for multitarget tracking, one needs first to hypothesize the origins of the measurements. These hypotheses, however, need to be generated in real time and cannot be specified a priori. The problem is thus different from ordinary hypothesis testing where the hypotheses to be tested are usually specified before hand. This introduces a new class of unconventional estimation problems which involves combinatorics (for generation of hypotheses), hypothesis testing and ordinary estimation.

Hypothesis management ideas which are important in artificial intelligence are also needed.

In this report we present a comprehensive theory on multitarget tracking and classification using a Bayesian approach. The theory is based on fairly general assumptions. In fact, tracking and classification can be considered simultaneously, and general models on target dynamics and sensors can be handled. Our results include almost all existing results as special cases when specific models are used and simplifying approximations are made. Although the results are theoretically complete, successful implementation requires good hypothesis management techniques.

Our algorithm has been implemented in a system called the Generalized Tracker/Classifier (GTC), which has the structure shown in Figure 1-2. It consists of three off-line modules and four on-line modules. The three off-line modules are:

1. generalized target dynamics model
2. environmental model
3. sensor model

These are used in the design and selections of algorithms. The on-line modules are:

1. hypothesis formation
2. hypothesis evaluation
3. filtering and parameter estimation
4. hypothesis management

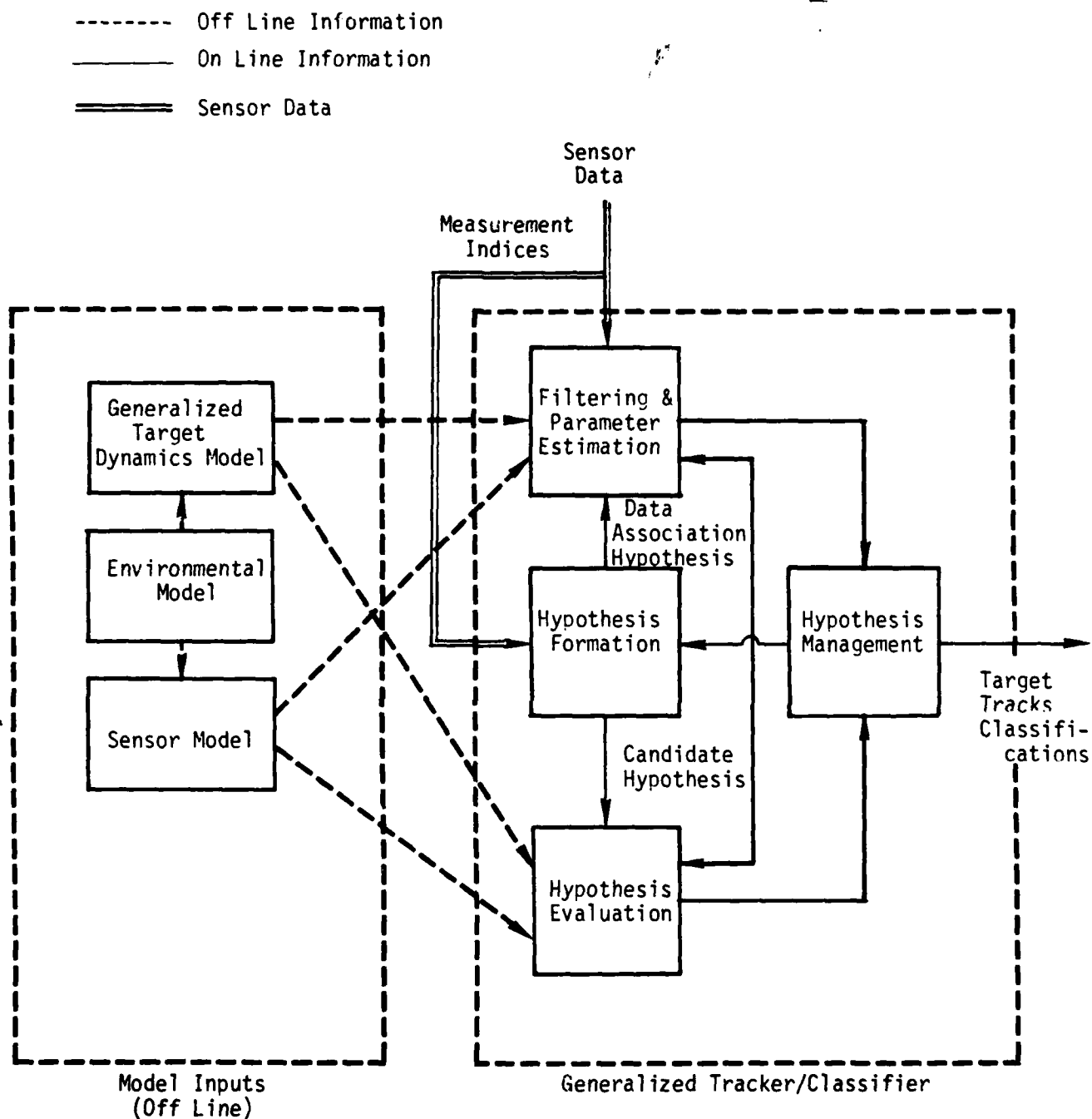


Figure 1-2 Generalized Tracker/Classifier

The hypothesis formation module generates the data-to-data association hypothesis which identifies the data reported at different times from different sensors with the same targets. The hypothesis evaluation module evaluates the probability of each hypothesis given the information available. We have developed recursive evaluation formulae which reduce to the standard formulae in the literature under simplifying assumptions. The filtering and parameter estimation module solves the tracking and classification problem given a particular data-association hypothesis. If the targets are independent and identically distributed, then the algorithm decomposes into a parallel bank of single target algorithms. The hypothesis management module controls the (asymptotically) exponential growth in the number of hypotheses. The details of these modules are described in the remaining sections and summarized in Section 8.

The GTC can be used by itself for centralized multitarget tracking and classification. It is also used to process the local sensor data at a node and, together with other modules for information distribution and fusion, is a basic component of each processing node in the DSN [26]. The details will be discussed in a subsequent report.

The structure of the report is as follows. In Section 2, we present the general target and sensor models. Section 3 introduces the definitions of tracks and hypotheses which are needed in the multiple hypotheses approach. The basic hypothesis evaluation algorithm applicable to the general target and sensor models is given in Section 4. Section 5 considers the specialization of this hypothesis evaluation algorithm to independent, identically distributed target models. The relationship of these results to the other existing algorithms is discussed in Section 6. In Section 7, we present the hypothesis

management techniques which are crucial for implementation. The implementation of the algorithm in the form of the Generalized Tracker/Classifier (GTC) is discussed in Section 8. Section 9 contains an example to illustrate the use of the GTC. A conclusion is given in Section 10.

2. TARGET AND SENSOR MODELS

In this section we describe the target and sensor models which will be assumed for the derivation of the general results. The models are quite general and include almost all other models used in the multitarget tracking literature that the authors are aware of. More specific models for a class of subproblems will appear in Section 5.

2.1 Target Model

As is customary in the multitarget tracking literature, a target is used to describe any object to be tracked, whether it is a friendly or enemy vehicle. In the general model described below, we model all targets which may be of interest as one entity rather than as a collection of individual targets. Thus the state of all targets is modeled by a continuous-time stochastic process (X, N_T) , taking values in the direct-sum space

$$\bigcup_{n=0}^{\infty} X_n \times \{n\}. \quad (2.1)$$

N_T is an integer-valued random variable representing the total number of targets. This number is assumed to be random but constant, largely to reduce unnecessary complexity. Given that the number of targets is n , X is a stochastic process taking values in X_n . In general, X_n is a hybrid set, i.e., a direct product of a subset of an Euclidean space (continuous part) and a finite set (discrete part). This is to allow for the representation of targets with both continuous and discrete states. X is also assumed to be a time-homogeneous Markov process. To complete our description, we assume there exists a time t_0 before which no sensor data is available. The target model

is then specified by the a priori probability density (mass) $p(N_T)$, distribution $p(dX_{t_0} | N_T)$, and the transition probability

$$F_{N_T}(E; X(t), \Delta t) \triangleq \text{Prob} \{X(t+\Delta t) \in E | (X(t), N_T)\} \quad (2.2)$$

defined for every $t \geq t_0, \Delta t \geq 0$ and every measurable set E in X_{N_T} . We use $p(\cdot)$ to denote the probability density (mass) or distribution of one or more random variables. What we mean should be obvious from the context.

This general target model includes the case where the targets are correlated, such as when they are moving in a group. A special case of interest which will be discussed in Section 5 consists of independent and identically distributed (i.i.d) targets. I.i.d target models constitute a very important subclass of target models since they describe many real-life situations. Furthermore, the resulting tracking algorithm is relatively easy to implement and is related to various existing tracking techniques. With the i.i.d. assumption, we can replace X_n by X^n , a direct-product space of the identical hybrid spaces X . The continuous part of X corresponds to position, velocity and other dynamics parameters, while the discrete part corresponds to the target type, maneuvering mode, communication activity, etc. Likewise, the various probability distributions and the transition probability (2.2) can also be partitioned. For example, (2.2) can now be replaced by

$$F(E; x(t), \Delta t) \triangleq \text{Prob} \{x(t+\Delta t) \in E | x(t)\} \quad (2.3)$$

which is common to all the i.i.d. targets.

2.2 Sensor Model

Let $S = \{1, \dots, N_s\}$ be the set of sensors which generate the measurements from the targets. For each $s \in S$, let the measurements take value in the set V_s . We assume V_s to be a hybrid space where the continuous part contains the position and/or velocity information and the discrete part contains feature type reports. A data set or sensor report is the set of data which a sensor generates at a given time. Formally, a data set is a random element (y, N_m, t, s) in the set

$$\bigcup_{m=0}^{\infty} \bigcup_{s \in S} V_s^m \times \{m\} \times [t_0, \infty) \times \{s\}, \quad (2.4)$$

which means that the data set from sensor s at time t consists of N_m measurements given by $y = (y_1, \dots, y_{N_m})$.

Five assumptions are made with respect to the generation of these data sets:

Assumption 1 (Known exact timing)

The time at which any data set is generated is exactly known.

Assumption 2 (Memoryless)

There is no memory in any sensor so that any target state density/distribution conditioned on the past data sets and the target state is the same as the one conditioned only on the target state.

Assumption 3

Any measurement in any data set from any one sensor
does not originate from two or more targets.

Assumption 4

No target creates two or more measurements in any data set.

Assumption 5

The order of measurements in a data set contains no
information about the targets.

Assumptions 1 and 2 are standard in filtering problems while assumptions 3 and 4 are almost standard in the multitarget tracking literature. Assumption 5 is crucial to our theoretical development and reflects the random nature about the origin of any measurement. In some situations, the sensor data should be partitioned so that these assumptions hold or at least are approximately valid. For example, the measurements from a radar with a fixed scanning pattern over an area may contain some information about the targets in the order of the measurements. In such a case, the data sets should be further divided so that the measurement order does not contain any significant information. By doing so, we do not need the notion of "type I sensors" and "type II sensors" which were loosely defined in [20].

With these assumptions, we can now describe our sensor model. A data set (y, N_m, t, s) is generated through the following four steps.

Detection:

Let δ be the detection function. Given $(X(t), N_T)$,
the target state at time t , δ is a random binary function

defined on a set $\{1, \dots, N_T\}$ as

$$\delta(i) = \begin{cases} 1 & \text{if the } i\text{-th target is detected} \\ 0 & \text{otherwise.} \end{cases} \quad (2.5)$$

This random function δ induces a set (of detected targets)

$$I_D = \{i \in \{1, \dots, N_T\} \mid \delta(i) = 1\}. \quad (6)$$

The detection model is completely specified by the conditional probability

$$p(\delta \mid X(t), N_T). \quad (2.7)$$

Generation of Number of False Alarms:

This is specified by the conditional probability

$$p(N_m \mid \delta, X(t), N_T). \quad (2.8)$$

Random Assignment:

Let α be an assignment function defined on I_D and taking values in the set $\{1, \dots, N_m\}$. $\alpha(i) = j$ means that the j -th measurement y_j in data set $((y_1, \dots, y_{N_m}), N_m, t, s)$ originates from the i -th target. Thus, with $A \setminus B$ as the difference of two sets,

$$J_{FA} = \{1, \dots, N_m\} \setminus \text{Image}(\alpha) \quad (2.9)$$

is the set of measurement indices corresponding to false alarms (or measurements not originating from a target). According to assumptions 3, 4, and 5, α should be one-to-one with probability one and we have

$$p(\alpha \mid N_m, \delta, X(t), N_T) = p(\alpha \mid N_m, \delta, N_T) = \frac{N_{FA}!}{N_m!} \quad (2.10)$$

for each α . N_{FA} , the number of false alarms, is the cardinality of the set J_{FA} . In other words, a particular ordering of the data $(y_1, \dots, y_{N_m})$ is a realization of equally likely assignments of the detected targets given they have been determined by δ .

Generation of Measurement and False Alarm Values

This is specified by the conditional probability density

$$p(y | \alpha, N_m, \delta, X(t), N_T) \quad (2.11)$$

which includes the measurement error model as well as the model of the false alarm values.

The general sensor model described above is similar to the usual measurement equation used in filtering except for the random assignment function which results from the unknown origins of the measurements and the random detection function which further complicates the process. This is the key difference between ordinary filtering and multitarget tracking.

3. TRACKS AND HYPOTHESES

Tracks and hypotheses are among the most frequently used terms in the multitarget tracking literature since there is a need to associate the measurements with the targets. However, in most cases these terms have not been precisely defined. Our definition of tracks and hypotheses follows closely Morefield's notations in [16], but differs in one crucial aspect, namely, the separation of the measurement-value information from the number-of-measurements information in each data set.

We assume that there are only a finite number of data sets. Let $\{t_1, t_2, \dots\}$ be the properly ordered set of times at which the data sets are generated by the sensors. Each data set (y, N_m, t_k, s) is then uniquely identified by the pair of integers (k, s) . We may thus refer to a data set by its index (k, s) provided that a sensor is not allowed to produce two data sets at the same time. For each data set (k, s) , a measurement index set is given by

$$I_m(k, s) = \{1, \dots, N_m\}. \quad (3.1)$$

A lexicographic order $\leq$ on the indices is defined as

- 1) $(k, s) < (k', s')$ if either $k < k'$ or $(k = k' \text{ and } s < s')$.
- 2) $(k, s) \leq (k', s')$ if either $(k, s) < (k', s')$ or $(k, s) = (k', s')$

The cumulative measurement index set is defined as

$$I_m(k, s) = \bigcup_{(k', s') \leq (k, s)} I_m(k', s') \times \{(k', s')\}. \quad (3.2)$$

The cumulative data set at (k,s) can be similarly defined as

$$Z^{(k,s)} = \bigcup_{(k',s') \leq (k,s)} \{(y(t_{k',s'}), N_m(t_{k',s'}), t_{k',s'})\}. \quad (3.3)$$

Our overall approach is Bayesian, i.e., we would like to compute $p(dX(t), N_T | Z^{(k,s)})$, the conditional probability of the target state given all the available information. Since the origins of the measurements are uncertain, our strategy is to hypothesize the origins of the measurements and use these hypotheses as intermediate variables in our computation.

We define a track τ at (k,s) to be a nonempty subset of the cumulative measurement index set $I_m^{(k,s)}$. A data-to-data association hypothesis λ , (referred to simply as a hypothesis later on) at (k,s) is a (possibly empty) set of track(s) at (k,s) . According to assumptions 3 and 4 in Section 2, any track containing two or more measurement indices from the same data set and any hypothesis with overlapping tracks are impossible in the sense that $p(\lambda | Z^{(k,s)}) = 0$.

A hypothesis $\lambda = \{\tau_1, \dots, \tau_n\}$ at (k,s) should be interpreted as a hypothesized event which states that n targets are detected in one or more data sets up to (k,s) and τ_i is the set of measurement indices (together with data set identification (k',s')) originating from the i -th detected target. Thus, for example, hypothesis $\lambda = \phi$ hypothesizes that all the measurements up to (k,s) are false alarms.

For any given (k,s) and (k',s') such that $(k,s) > (k',s')$, we call a hypothesis λ at (k',s') a predecessor of a hypothesis λ at (k,s)

if each track τ' in λ' is a restriction of a track τ in λ to $I_m^{(k',s')}$, i.e., $\tau' = \tau \cap I_m^{(k',s')}$ and each track τ in λ is either the extension of a track τ' in λ' to $I_m^{(k,s)}$ or $\tau \cap I_m^{(k',s')} = \phi$. If (k',s') is the unique immediate predecessor of (k,s) , λ' is called the parent of λ .

We should note that a parent of a hypothesis λ is uniquely determined by

$$\lambda' = \{\tau \cap I_m^{(k',s')} \mid \tau \in \lambda\}.$$

4. RECURSIVE BAYESIAN EVALUATION OF HYPOTHESES

As mentioned before our strategy is to calculate $p(dX(t), N_T | Z^{(k,s)})$ using the expansion

$$p(dX(t), N_T | Z^{(k,s)}) = \sum_{\lambda} p(dX(t), N_T | \lambda, Z^{(k,s)}) p(\lambda | Z^{(k,s)}). \quad (4.1)$$

Since the number of measurements in any data set is finite, the number of possible hypotheses given the cumulative data set is also finite. Therefore, at least in principle, we can calculate the conditional probability of the target state if $p(dX(t), N_T | \lambda, Z^{(k,s)})$ and $p(\lambda | Z^{(k,s)})$ are evaluated. In this section, we shall first discuss the recursive computation of $p(\lambda | Z^{(k,s)})$. The computation of $p(dX(t), N_T | \lambda, Z^{(k,s)})$, which is also necessary, will then follow.

4.1 Computation of $p(\lambda | Z^{(k,s)})$

Let the current data set be (y, N_m, t_k, s) and the immediate predecessor of (k, s) be (k', s') . To simplify the notation, we denote $Z^{(k,s)}$, $Z^{(k',s')}$, $(X(t_k), N_T)$, $(X(t_{k'}), N_T)$ and the parent of a hypothesis λ , by Z , $\bar{Z}$, (X, N_T) , $(\bar{X}, N_T)$ and $\bar{\lambda}$, respectively.

A straightforward application of Bayes rule gives us the following basic equation:

$$p(\lambda | Z) = \frac{p(Z, \lambda | \bar{Z}, \bar{\lambda}) p(\bar{\lambda} | \bar{Z})}{p(Z | \bar{Z})}. \quad (4.2)$$

When $\bar{Z} = \phi$, we have $\bar{\lambda} = \phi$ and $p(\bar{\lambda} | \bar{Z}) = 1$ in (4.2). Since

$p(\bar{\lambda}|\bar{Z})$ is given by recursion and the denominator is the normalizing constant, the only term to be evaluated is $p(Z, \lambda|\bar{Z}, \bar{\lambda})$ which is expanded as

$$p(Z, \lambda|\bar{Z}, \bar{\lambda}) = \sum_{N_T} p(N_T|\bar{Z}, \bar{\lambda}) \int_{X_{N_T}} p(Z, \lambda|X, N_T, \bar{Z}, \bar{\lambda}) p(dX|N_T, \bar{Z}, \bar{\lambda}) \quad (4.3)$$

where the integral is defined with respect to the hybrid probability measure $p(\cdot | N_T, \bar{Z}, \bar{\lambda})$. Both $p(N_T|\bar{Z}, \bar{\lambda})$ and $p(dX|N_T, \bar{Z}, \bar{\lambda})$ are to be generated by recursion in Section 4.2, and can be used to compute $p(dX, N_T|\bar{Z}, \bar{\lambda})$.

On the other hand, $p(Z, \lambda|X, N_T, \bar{Z}, \bar{\lambda})$ in (4.3) can be further expanded as

$$\begin{aligned} p(Z, \lambda|X, N_T, \bar{Z}, \bar{\lambda}) &= \sum_{(\alpha, \delta)} p(y|\alpha, N_m, \delta, X, N_T, \bar{Z}, \bar{\lambda}) \\ &\quad p(\lambda|\alpha, N_m, \delta, X, N_T, \bar{Z}, \bar{\lambda}) \\ &\quad p(\alpha|N_m, \delta, X, N_T, \bar{Z}, \bar{\lambda}) \\ &\quad p(N_m|\delta, X, N_T, \bar{Z}, \bar{\lambda}) \\ &\quad p(\delta|X, N_T, \bar{Z}, \bar{\lambda}) \end{aligned} \quad (4.4)$$

where α and δ are the assignment and detection functions defined in Section 2,

$$p(y|\alpha, N_m, \delta, X, N_T, \bar{Z}, \bar{\lambda}) = p(y|\alpha, N_m, \delta, X, N_T), \quad (4.5)$$

$$p(N_m|\delta, X, N_T, \bar{Z}, \bar{\lambda}) = p(N_m|\delta, X, N_T), \quad (4.6)$$

and

$$p(\delta|X, N_T, \bar{Z}, \bar{\lambda}) = p(\delta|X, N_T) \quad (4.7)$$

are provided by the measurement-error/false-alarm distribution model, the false-alarm-number model and the detection model respectively. Also, $p(\alpha | N_m, \delta, X, N_T, \bar{Z}, \bar{\lambda})$ is given by (2.10) in Section 2. The remaining term,

$$p(\lambda | \alpha, N_m, \delta, X, N_T, \bar{Z}, \bar{\lambda}) = p(\lambda | \alpha, N_m, \delta, N_T, \bar{\lambda}), \quad (4.8)$$

checks the consistency between λ and $(\alpha, N_m, \delta, N_T)$, i.e.,

$$p(\lambda | \alpha, N_m, \delta, N_T, \bar{\lambda}) = \begin{cases} 1 & \text{if } \lambda \text{ is consistent} \\ 0 & \text{otherwise} \end{cases} \quad (4.9)$$

To define consistency more precisely, we shall introduce additional index sets. Given λ , let I_D and I_{DC} be the sets of indices of targets which are hypothesized by λ to be detected in the current data set and in the cumulative data set, respectively. $\bar{I}_D$ and $\bar{I}_{DC}$ are defined similarly for $\bar{\lambda}$. Such indexing is not unique since there is no a priori indexing on the targets. Therefore, each target index set hypothesized by λ or $\bar{\lambda}$ should be interpreted as one of the equal-probability representations. In addition, let $I_N = I_D \setminus \bar{I}_{DC}$ be the index set of targets hypothesized by λ to be newly detected at (k, s) . A hypothesis, $\lambda = \{\tau_i | i \in I_{DC}\}$, with the parent, $\bar{\lambda} = \{\bar{\tau}_i | i \in \bar{I}_{DC}\}$, is consistent with $(\alpha, N_m, \delta, N_T)$ if and only if

$$1) \quad \tau_i = \bar{\tau}_i \cup \{(\alpha(i), k, s)\} \text{ for } i \in I_{DC} \cap I_D, \quad (4.10)$$

$$2) \quad \tau_i = \bar{\tau}_i \text{ for } i \in I_{DC} \setminus I_D, \quad (4.11)$$

and

$$3) \quad \tau_i = \{(\alpha(i), k, s)\} \text{ for } i \in I_N. \quad (4.12)$$

Thus, if λ is consistent with $(\alpha, N_m, \delta, N_T)$, I_D as defined here is also consistent with that given by δ in Section 2.

Consequently, (4.4) is reduced to

$$p(Z, \lambda | X, N_T, \bar{Z}, \bar{\lambda}) = \frac{N_{FA}!}{N_m!} \sum_{(\alpha, \delta)} p(y | \alpha, N_m, \delta, X, N_T) p(N_m | \delta, X, N_T) p(\delta | X, N_T) \quad (4.13)$$

where the summation is over the set of (α, δ) 's for which λ is consistent. It then follows from (4.3) and (4.13) that

$$p(Z, \lambda | \bar{Z}, \bar{\lambda}) = \frac{N_{FA}!}{N_m!} \sum_{N_T} p(N_T | \bar{Z}, \bar{\lambda}) \left\{ \sum_{\alpha, \delta} \int_{X_{N_T}} p(y | \alpha, N_m, \delta, X, N_T) p(N_m | \delta, X, N_T) p(\delta | X, N_T) p(dX | N_T, \bar{Z}, \bar{\lambda}) \right\}. \quad (4.14)$$

Equations (4.2) and (4.14) allow the recursive computation of $p(\lambda | Z)$ if $p(N_T | Z, \lambda)$ and $p(dX | N_T, Z, \lambda)$ are known. All the other conditional probability distributions or densities needed are given by the sensor model.

4.2 Computation of $p(dX|N_T, Z, \lambda)$ and $p(N_T|Z, \lambda)$

Up to this point, we have not used the Markov nature of the target model. The updating of the target state (X, N_T) is by means of

$$p(dX|N_T, Z, \lambda) = C(N_T)^{-1} \sum_{\alpha, \delta} p(y|\alpha, N_m, \delta, X, N_T) p(N_m|\delta, X, N_T) p(\delta|X, N_T) p(dX|N_T, \bar{Z}, \bar{\lambda}) \quad (4.15)$$

and

$$p(N_T|Z, \lambda) = \frac{C(N_T) (N_{FA}!) p(N_T|\bar{Z}, \bar{\lambda})}{p(Z, \lambda|\bar{Z}, \bar{\lambda}) N_m!} \quad (4.16)$$

where $C(N_T)$ is a normalization constant given by

$$C(N_T) = \sum_{\alpha, \delta} \int_{\chi_{N_T}} p(y|\alpha, N_m, \delta, X, N_T) p(N_m|\alpha, X, N_T) p(\delta|X, N_T) p(dX|N_T, \bar{Z}, \bar{\lambda}). \quad (4.17)$$

Extrapolation can be accomplished through

$$p(dX|N_T, \bar{Z}, \bar{\lambda}) = \int_{\chi_{N_T}} F_{N_T}(dX|\bar{X}, \Delta t) p(d\bar{X}|N_T, \bar{Z}, \bar{\lambda}) \quad (4.18)$$

where

$$\Delta t = t_k - t'_k.$$

To the best of our knowledge, the results presented in this section for the general target and sensor models are new. The evaluation of each hypothesis is carried out recursively using equations (4.2) and (4.14). Equation (4.14) also requires the computation of certain posterior conditional probabilities for the number of targets and the target states. These are given by equations (4.15) to (4.18). We can thus theoretically handle very complicated situations, such as targets moving as a group. However, since a nonlinear filtering problem is involved, the actual use of this algorithm is not trivial unless some special structures are exploited. An example of this for i.i.d. target models is discussed in the next section.

5. INDEPENDENT, IDENTICALLY DISTRIBUTED TARGET MODELS

As we may anticipate, the fairly general algorithm described in the previous section can be greatly simplified if we assume that given N_T , the number of targets, each individual target state is an independent and identically distributed random process. The simplification is possible because many terms in (4.13)-(4.18) can be expressed as products under these assumptions. In this section, the important subclass of i.i.d. target models is discussed and the result will be used to provide a unified view of existing tracking algorithms.

5.1 Assumptions on Models

The following additional assumptions are used for this section:

Assumption A1

The target state space X_n for $N_T = n$ is the direct product set X^n of a hybrid state X with a hybrid measure μ . Given $N_T = n$, the target state $X = (x_i)_{i=1}^n$ is a system of time-homogeneous, independent and identically distributed Markov processes with the common statistics given by the initial distribution

$$\text{Prob } \{x_i(t_0) \in dx\} = q_0(x) \mu(dx) \quad (5.1)$$

and the transition probability density

$$\text{Prob } \{x_i(t+\Delta t) \in dx | x_i(t)\} = F(x_i(t+\Delta t) | x_i(t), \Delta t) \mu(dx). \quad (5.2)$$

Assumption A2

The a priori distribution of N_T , the number of targets, is Poisson with mean ν_0 .

Assumption A3

N_{FA} , the number of false alarms for each data set (k,s) , is independent of any target state or any other data set and has a Poisson distribution with mean ν_{FA} which may be a function of (k,s) (time and sensor). Given N_{FA} , the number of false alarms at a data set (k,s) , the false alarms are i.i.d. with the common probability density $p_{FA}(y)$ on $\mathcal{Y}_s$ and are independent of any target state or any other data set.

Assumption A4

The measurement error in y which originates from a target i in any data set depends only on the target state x_i . Thus, there exists a measurement probability transition function, $p_M(\cdot|\cdot): \mathcal{Y} \times \mathcal{X} \rightarrow [0, \infty)$ such that

$$p((y_1, \dots, y_{N_m}) | \alpha, N_m, \delta, X, N_T) = \prod_{i \in I_D} p_M(y_{\alpha(i)} | x_i) \left(\prod_{j \in J_{FA}} p_{FA}(y_j) \right). \quad (5.3)$$

Assumption A5

The event pertaining to the detection of a target i depends only on its state x_i , namely, there is a detection function δ and detection probability $p_D(\cdot): \mathcal{X} \rightarrow [0, 1]$ such that

$$p(\delta | X, N_T) = \prod_{i=1}^{N_T} p_D(x_i)^{\delta(i)} (1 - p_D(x_i))^{1-\delta(i)}. \quad (5.4)$$

5.2 Hypothesis Evaluation Algorithm

Assumptions A1, A4 and A5 imply that, for any (Z, λ) at any data set (k,s) , we have

$$p(dx_1, \dots, dx_{N_T} | N_T, Z, \lambda) = \prod_{i \in I_{DC}} p(x_i | Z, \tau_i) \mu(dx_i) \prod_{i \notin I_{DC}} q(x_i) \mu(dx_i) \quad (5.5)$$

where $\lambda = \{\tau_i\}_{i \in I_{DC}}$, $p(\cdot | Z, \tau_i)$ is the probability density of $x_i(t_k)$ conditioned by the corresponding track and the measurement-value data given by Z , and q is the common probability density of a target conditioned by the fact that it has never been detected up to and including the data set (k, s) .

On the other hand, Assumptions A3 to A5 together with (5.5) can reduce (4.14) into the following form:

$$\begin{aligned} p(Z, \lambda | \bar{Z}, \bar{\lambda}) &= \frac{e^{-v_{FA}}}{N_m!} \sum_{N_T} p(N_T | \bar{Z}, \bar{\lambda}) \sum_{(\alpha, \delta)} v_{FA}^{N_{FA}} \prod_{j \in J_{FA}} p_{FA}(y_j) \\ &\quad \int_X^n \left(\prod_{i \in I_D} p_M(y_{\alpha(i)} | x_i) p_D(x_i) \right) \left(\prod_{i \notin I_D} (1 - p_D(x_i)) \right) \\ &\quad \left(\prod_{i \in I_{DC}} p(x_i | \bar{Z}, \bar{\tau}_i) \mu(dx_i) \right) \left(\prod_{i \notin I_{DC}} q(x_i) \mu(dx_i) \right) \\ &= \frac{e^{-v_{FA}}}{N_m!} \sum_{N_T} p(N_T | \bar{Z}, \bar{\lambda}) \sum_{(\alpha, \delta)} v_{FA}^{N_{FA}} \left(\prod_{j \in J_{FA}} p_{FA}(y_j) \right) \\ &\quad \left(\prod_{i \in I_D \cap \bar{I}_{DC}} \int_X p_M(y_{\alpha(i)} | x) p_D(x) p(x | \bar{Z}, \bar{\tau}_i) \mu(dx) \right) \\ &\quad \left(\prod_{i \in \bar{I}_{DC} \setminus I_D} \int_X (1 - p_D(x)) p(x | \bar{Z}, \bar{\tau}_i) \mu(dx) \right) \\ &\quad \left(\prod_{i \in I_N} \int_X p_M(y_{\alpha(i)} | x) p_D(x) q(x) \mu(dx) \right) \\ &\quad \left(1 - \int_X p_D(x) q(x) \mu(dx) \right)^{N_T - N_{DC}} \end{aligned} \quad (5.6)$$

where $N_{DC} = \#(I_{DC}) = \#(\lambda)$ is the number of detected targets being hypothesized by λ .

It is clear that $p(N_T|Z, \lambda) = 0$ if $N_T < N_{DC}$. When a triple (λ, N_m, N_T) is given, the number of δ 's such that there exists an α so that λ is consistent with (α, δ) is

$$\binom{N_T - \bar{N}_{DC}}{N_N} = \frac{(N_T - \bar{N}_{DC})!}{N_N! (N_T - N_{DC})!} \quad (5.7)$$

where $\bar{N}_{DC} = \#(\bar{I}_{DC}) = \#(\bar{\lambda})$ and $N_N = \#(I_N) = N_{DC} - \bar{N}_{DC}$. For such a δ , there are $N_N!$ permutations, or $N_N!$ α 's for which λ is consistent with (α, δ) .

Therefore, the number of (α, δ) 's with which λ is consistent is given by

$$\frac{(N_T - \bar{N}_{DC})!}{(N_T - N_{DC})!} \quad (5.8)$$

Furthermore, for a given (λ, N_m, N_T) , all (α, δ) 's with which λ is consistent should agree on $\bar{I}_{DC}$, i.e., if λ is consistent with (α, δ) and (α', δ') we must have $\alpha(i) = \alpha'(i)$ and $\delta(i) = \delta'(i)$ for all $i \in \bar{I}_{DC}$. In addition, for all (α, δ) 's with which λ is consistent, the product

$$\prod_{i \in \bar{I}_N} \int_X p_M(y_{\alpha(i)} | x) p_D(x) q(x) \mu(dx) \quad (5.9)$$

has the same value as a product even though individual multipliers and

I_N may vary. Thus, the four products behind the summation $\sum_{(\alpha, \delta)}$ in the last expression of (5.6) have the same value for all such (α, δ) 's.

Consequently, when we choose one (α, δ) with which λ is consistent and define $J_N = \{\alpha(i) \mid i \in I_N\}$, (5.6) is further reduced to

$$\begin{aligned}
 p(Z, \lambda | \bar{Z}, \bar{\lambda}) &= \frac{e^{-\nu_{FA}}}{N_m!} \prod_{j \in J_{FA}} \nu_{FA} p_{FA}(y_j) \\
 &\quad \prod_{i \in I_D \cap \bar{I}_{DC}} \int_X p_M(y_{\alpha(i)} | x) p_D(x) p(x | \bar{Z}, \bar{\tau}_i) \mu(dx) \\
 &\quad \prod_{i \in \bar{I}_{DC} \setminus I_D} \int_X (1 - p_D(x)) p(x | \bar{Z}, \bar{\tau}_i) \mu(dx) \\
 &\quad \prod_{j \in J_N} \int_X p_M(y_j | x) p_D(x) q(x) \mu(dx) \\
 &\quad \sum_{N_T = N_{DC}}^{\infty} p(N_T | \bar{Z}, \bar{\lambda}) \frac{(N_T - \bar{N}_{DC})!}{(N_T - N_{DC})!} (1 - \int_X p_D(x) q(x) \mu(dx))^{N_T - N_{DC}}
 \end{aligned} \tag{5.10}$$

Using a similar argument, equation (4.16) can be reduced to

$$p(N_T | Z, \lambda) = \begin{cases} C_N^{-1} \frac{(N_T - \bar{N}_{DC})!}{(N_T - N_{DC})!} (1 - \int_X p_D(x) q(x) \mu(dx))^{N_T - N_{DC}} p(N_T | \bar{Z}, \bar{\lambda}) & \text{if } N_T \geq N_{DC} \\ 0 & \text{otherwise} \end{cases} \tag{5.11}$$

where C_N is a normalizing constant. Then, using an inductive argument, it is easy to show that at any data set (k, s) , $p(N_T | Z, \lambda)$ is a biased Poisson distribution. In particular, we have

$$p(N_T | \bar{Z}, \bar{\lambda}) = \begin{cases} e^{-\bar{\nu}} \frac{\bar{\nu}^{(N_T - \bar{N}_{DC})}}{(N_T - \bar{N}_{DC})!} & \text{if } N_T \geq \bar{N}_{DC} \\ 0 & \text{otherwise} \end{cases} \tag{5.12}$$

where $\bar{v}$ is the expected number of undetected targets up to and including (k', s') . $\bar{v}$ is common to every λ and is given by recursion.

By substituting (5.11) into (5.10) and using the basic equation (4.2), we obtain the following recursive equation:

$$p(\lambda|Z) = \frac{p(\bar{\lambda}|\bar{Z})}{N_m! \exp(v_{FA} + (1 - L_{oo})\bar{v}) p(Z|\bar{Z})} \prod_{j \in J_{FA}} L_{FAj} \prod_{i \in I_D \cap \bar{I}_{DC}} L_{i\alpha(i)} \prod_{i \in \bar{I}_{DC} \setminus I_D} L_{io} \prod_{j \in J_N} L_{oj} \quad (5.13)$$

where

$$L_{FAj} = v_{FA} P_{FA}(y_j) \quad (5.14)$$

is the likelihood of the j th measurement being a false alarm,

$$L_{ij} = \int_{\mathcal{X}} p_M(y_j|x) p_D(x) p(x|\bar{Z}, \bar{\tau}_i) \mu(dx) \quad (5.15)$$

is the likelihood of the j th measurement originating from the i th detected target,

$$L_{io} = \int_{\mathcal{X}} (1 - p_D(x)) p(x|\bar{Z}, \bar{\tau}_i) \mu(dx) \quad (5.16)$$

is the likelihood of the i th previously detected target missing in the current data set,

$$L_{oj} = \bar{v} \int_X p_M(y_j | x) p_D(x) q(x) \mu(dx) \quad (5.17)$$

is the likelihood of the j th measurement originating from a newly detected target, and

$$L_{oo} = 1 - \int_X p_D(x) q(x) \mu(dx) \quad (5.18)$$

is the likelihood of an undetected target remaining undetected in the data set.

The updating and extrapolation equations given by (4.15), (4.16) and (4.18) can now be simplified. Updating is given by the following equations. For $i \in I_D \cap \bar{I}_{DC}$,

$$p(x|Z, \tau_i) = L_{i\alpha(i)}^{-1} p_M(y_{\alpha(i)} | x) p_D(x) p(x|\bar{Z}, \bar{\tau}_i). \quad (5.19)$$

For $i \in \bar{I}_{DC} \setminus I_D$,

$$p(x|Z, \tau_i) = L_{io}^{-1} (1 - p_D(x)) p(x|\bar{Z}, \bar{\tau}_i). \quad (5.20)$$

For $i \in I_N$,

$$p(x|Z, \tau_i) = L_{o\alpha(i)}^{-1} p_M(y_{\alpha(i)} | x) p_D(x) q(x). \quad (5.21)$$

v , the new expected number of undetected targets, is given by

$$v = \bar{v} L_{oo} \quad (5.22)$$

where v becomes $\bar{v}$ at the next data set. $q(\cdot)$ should be replaced by

$$L_{oo}^{-1} (1 - p_D(x)) q(x). \quad (5.23)$$

When $\Delta t = t_k - t_{k-1} > 0$, extrapolation is accomplished by using the following equations where after extrapolation $p(.|\bar{Z}, \bar{\tau}_i)$ and $q(.)$ are replaced by

$$\int_X F(x|\bar{x}, \Delta t) p(\bar{x}|\bar{Z}, \bar{\tau}_i) \mu(d\bar{x}) \quad (5.24)$$

and

$$\int_X F(x|\bar{x}, \Delta t) q(\bar{x}) \mu(d\bar{x}) \quad (5.25)$$

respectively.

Equations (5.13) - (5.25) provide the basic algorithms for the tracking of multiple independent and identically distributed targets. Equation (5.13) is the key equation and computes the probability of a hypothesis recursively using some basic likelihood functions. These likelihood functions are given by equations (5.14) to (5.18). All the necessary conditional probabilities on the target states are given by equations (5.19) to (5.25). All the likelihood functions have very intuitive explanations and are based on the target and sensor models. In the next section, we relate this algorithm to other existing results.

6. RELATION TO OTHER RESULTS

The results in Section 4 for general target models are new to the best of our knowledge. Even when specialized to i.i.d. target models, the algorithm given by equations (5.13)-(5.25) has been derived using more general models than usually assumed in the literature. In particular, hybrid (continuous and discrete) target models and complex p_D models can be handled theoretically, although the actual implementation may be nontrivial. In this section we compare our results on i.i.d. target models to some existing algorithms. It will be shown that these algorithms can be obtained from our basic algorithm when certain approximations are made.

Since equation (5.13) is recursive in nature, repeated application of (5.13) leads to the following batch form,

$$p(\lambda|Z) = C(Z)^{-1} L_{FA} \prod_{i \in I_{DC}} L_i, \quad (6.1)$$

where $C(Z)$ is a normalization constant given by

$$C(Z) = p(Z) \exp(\nu_0 - \nu) \prod_{(k', s') \leq (k, s)} N_m! \exp \nu_{FA}, \quad (6.2)$$

$$L_{FA} = \prod_{(k', s') \leq (k, s)} \prod_{j \in J_{FA}} L_{FAj}, \quad (6.3)$$

and L_i can be computed recursively using equations (5.15)-(5.17).

The batch form (6.1) is similar to the basic criterion function used in Morefield's algorithm[16] where 0-1 linear programming is used for the final optimization. It should be noted that in our derivation of (6.1), we do

not have to evaluate $p(\lambda)$, the a priori probability of a hypothesis in Morefield's formulation.

Almost all the known algorithms in multitarget tracking are based on assumptions introduced in Section 5, although in some cases these assumptions have not been stated explicitly. In addition, the following assumptions are usually made when classification is not of interest.

- (1) $p_M(y_j|x)$ is defined by a linear Gaussian model, i.e.,

$$y_j = Hx + v \quad (6.4)$$

where H is a matrix and v is a Gaussian noise process.

- (2) $p_{FA}(y_j)$ is a uniform density.

In many cases, it is also reasonable to assume that $q(\cdot)$, the distribution of an undetected target, has a much bigger variance than that of $p_M(\cdot|x)$. In this case, $p(\cdot|Z, \tau)$ is approximately Gaussian, and hence equations (5.19) and (5.24) reduce to the standard Kalman filter equations. The uniformity assumption of p_{FA} reduces L_{FAj} to a constant which is actually the number of false alarms per unit scan volume. Furthermore, if we assume that $p_D(\cdot)$ is constant over the sensor's field of view which is large compared with the variance of any reasonable $p(\cdot|Z, \tau)$, then (5.15) and (5.16) can be approximated by

$$L_{ij} = p_D \int_X p_M(y_j|x) p(x|\bar{Z}, \bar{\tau}_i) \mu(dx) \quad (6.5)$$

and

$$L_{io} = 1 - p_D \quad (6.6)$$

where p_D is the constant value of $p_D(.)$. It is now clear that the algorithm given by (5.13) to (5.25) coincides with Reid's algorithm [20] except for L_{oj} , the new target likelihood function, which in our case is computed from v and $q(.)$. The corresponding term in [20] is called the new target density and is said to "depend on the number of times the area has been observed by a type 1 sensor and the possible flux of undetected targets into and out of the area." However, no expression is given for its computation in [20]. Although the computation of L_{oj} as given by (5.17) is not easy, it is an important term in evaluating hypotheses and initiating new tracks, particularly when a number of returns are obtained in just a few scans.

As in [20], there is no separate track initiation process in our algorithm. This can be considered to be more general than algorithms such as those in [13],[21] where a separate initiation process is required, in the following sense. If the cumulative measurement index set of (3.2) is expanded to include $\{1, \dots, \bar{N}\} \times \{(0,0)\}$ where $\bar{N}$ is the number of a priori tracks, then the general equation (5.13) reduces to an algorithm with a separate track initiation process where the recursion (5.13) is started with a single probability one hypothesis containing the $\bar{N}$ tracks.

It is well known that the number of hypotheses grows very rapidly in an algorithm of this type. Therefore, for a multi-hypothesis type tracking algorithm to be implementable, some form of hypothesis management scheme must be used. Many hypotheses management schemes [20] are available, e.g., fixed level, fixed breadth or adaptive pruning, M-scan or similar hypothesis combining, windowing (data validation), clustering, etc. In some existing algorithms, one of the above strategies is used to its extreme. For example, a zero scan algorithm as defined in [20] is used to obtain the probabilistic

data association (PDA) filter in [10] and the joint probabilistic data association (JPDA) filter in [21]. Extensive combining is used in the Gaussian sum approach of Alspach [13]. These will be discussed in detail in Section 7.

7. HYPOTHESIS MANAGEMENT TECHNIQUES

It should be obvious that any algorithm which forms multiple hypotheses from the data suffers from the rapid growth in the number of hypotheses. The growth is actually worse than exponential although no worse than factorial. Thus adequate hypothesis management to control the number of hypotheses is essential for any successful implementation of such algorithms. Many hypothesis management techniques have been proposed in the past. For the subclass of multitarget tracking problems described in Section 5, almost all the existing techniques are needed. These techniques have usually been proposed based on intuitive reasons rather than mathematically rigorous arguments. An exception is a recent report [27] in which "optimal" pruning and windowing is discussed. It is the authors' opinion, however, that these results may only be applicable to special cases.

Thus no general theory on hypothesis management techniques exists at the present moment. The purpose of this section is to summarize some existing techniques and describe any modifications to such techniques adopted by the authors. We divide the hypothesis managements techniques into the following four classes:

1. Pruning cutting branches
2. Combining binding branches together
3. Windowing data validation
4. Clustering data partition

In the following, we shall discuss the techniques according to the above classification. Although some of the techniques described below may apply to

the general model of Section 4, for the most part we shall restrict our attention to the i.i.d. target case where all the assumptions in Section 5 are satisfied.

7.1 Pruning

Pruning techniques can be further classified into (1) thresholding, (2) breadth control, and (3) adaptive pruning techniques.

The basic philosophy behind thresholding is to cut (or remove) the "insignificant" hypotheses which in turn tend to produce more insignificant hypotheses. In [20], it is proposed to cut any hypothesis with posterior probability less than a fixed predetermined threshold. In the i.i.d. target case, the thresholding may be performed at the track level using the track likelihood functions described in Section 5. One of the disadvantages of this fixed thresholding is that it is performed without any consideration of the available computational resources or the external condition (e.g., clear versus confusing, etc.). For example, given the same computational resources, one should be able to keep more hypotheses for little available data than for a large amount of data. This, however, is not present in the fixed thresholding scheme.

This consideration leads to the second subclass of breadth control techniques in which a fixed number, say M , of the best hypotheses are chosen and propagated forward. This is proposed by Keverian in [17]. Choosing a fixed breadth M makes sense when we regard the number of hypotheses kept as a measure of the computational and memory requirements. However, fixed breadth control may lose its rationale quickly when some form of clustering is used since the resources cannot be efficiently allocated among the clusters. Also, the

breadth control or fixed breadth method requires a sorting algorithm which requires additional effort although this issue may not be important since many good sorting algorithms do exist. When breadth control is used extensively to its limit, only one (best) hypothesis is selected and propagated forward. In [20], this mode of pruning is called a zero-scan algorithm.

Although fixed-threshold pruning may be viewed as adaptive-breadth pruning, and vice versa, these techniques do not really adapt to the complexity of the situation. The third subclass of pruning techniques, called adaptive pruning, is proposed by the authors to be more adaptive. In this strategy, the hypotheses are first sorted into the descending order of their posterior probabilities. Then, when the cumulative sum of the probabilities from the best hypothesis exceeds a given threshold, the remaining low probability hypotheses are pruned. This method may be called adaptive-threshold/adaptive-breadth pruning since it adjusts both the absolute threshold and the breadth according to the complexity of the external condition, i.e., the more complex the situation is the more low probability hypotheses are retained. The adaptive pruning techniques make more sense than other pruning methods when clustering is used and may be viewed as a way of automatically allocating computational and memory resources among the clusters. However, it still suffers from the same drawback of any fixed thresholding scheme in that the actual (absolute) computational and memory resources cannot be predicted. Furthermore, some form of sorting is still needed.

From a theoretical point of view, the posterior probabilities of hypotheses may be considered as a discrete distribution. Hypothesis pruning may then be viewed as picking the approximation techniques for the distribution. Figure 7-1 displays the approximation involved in the three schemes.

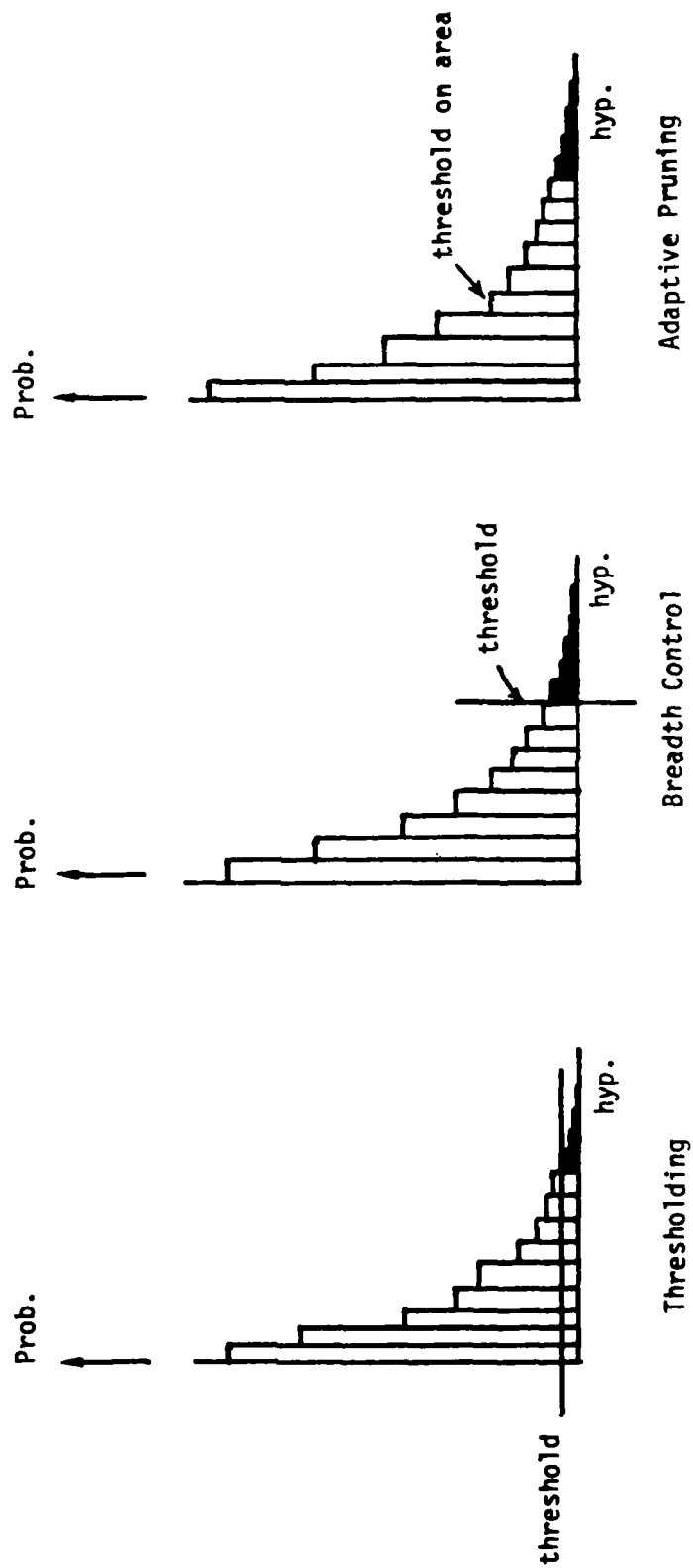


Figure 7-1 Pruning Strategies

Thus the theoretical issues in hypothesis pruning are what a good approximation should be and how it influences the future evaluation in multitarget tracking. Although some theories on the approximation of probability distributions (e.g., Sorenson and Alspach [28]) may give us some insight, we believe that hypothesis pruning is still an open research area.

7.2 Combining

The existing combining techniques can be divided into two subclasses: (1) distribution-oriented techniques and (2) measurement-index-oriented techniques.

The philosophy behind the first subclass is to combine two similar hypotheses, where similarity is interpreted in a certain way, depending on the state distribution. According to Reid [20], two hypotheses are similar if they have the same number of tracks and each track in one hypothesis has a unique companion track which is similar to it in the other hypothesis. The similarity of tracks is measured by the state estimate distributions, which accounts for the name of distribution-oriented techniques. The rationale behind this approach is that each track state distribution should reflect all the relevant information which affects any future event due to the underlying Markovian assumptions. Thus, if two state distributions of tracks are close enough, we would expect the future behavior of the two tracks to be similar.

Suppose two hypotheses λ_1 and λ_2 are similar, where $\lambda_i = \{\tau_1^i, \dots, \tau_n^i\}$ $i = 1, 2$. Then hypothesis combining leads to a new hypothesis $\lambda = \{\tau_1, \dots, \tau_n\}$ with

$$p(\lambda|Z) = p(\lambda_1|Z) + p(\lambda_2|Z) \quad (7.1)$$

and for $j = 1, \dots, n$

$$p(x|\tau_j, Z) = p(\lambda_1|Z)p(x|Z, \tau_j^1) + p(\lambda_2|Z)p(x|Z, \tau_{\pi(j)}^2) \quad (7.2)$$

where $\pi(\cdot)$ is a permutation which maps a track into a similar track.

However, there still remains the crucial question of choosing a good measure of "similarity" and a good threshold for that measure. When each track distribution is Gaussian, Reid [20] proposes inequality tests using the means and the diagonal elements of the covariance matrices.

However, no theoretical justification for the use of those particular inequalities is given. His intuitive reason is that for tracks to be similar, both their means and their variances should not be widely different. This test may work well for Gaussian distributions, but may prove to be inadequate for more general distributions.

Distribution-oriented combining is used to its extreme in [11] and [21] where all the hypotheses are combined after proper windowing (described below) and a fixed number of targets are assumed i.e., every hypothesis has the same number of tracks. When two Gaussian distributions are combined, the combined distribution becomes a Gaussian sum distribution because two different hypotheses represent two disjoint events. When a Gaussian sum distribution is approximated by a Gaussian distribution, the means and the variances are usually equated. Unlike the results in [20] or [11], the Gaussian sum form is preserved to a certain extent in [13] where each track distribution remains a Gaussian sum rather than Gaussian. In this case, the hypothesis trees are extended to include one lower level, namely, the distribution level. The hypothesis management (pruning and combining) techniques must then be extended to include this level.

In summary, unless each track distribution is assumed to be and forced to be Gaussian, the similarity criteria proposed in [20], [11], etc., may sometimes be unjustified. Theoretical results on similarity criteria are still lacking in our opinion.

On the other hand, the measurement-index-oriented combining techniques consider each track as a subset of the past cumulative measurement index set as defined in Section 3. This technique has been proposed by Singer, et. al., [9] and is a classical technique in the multitarget tracking literature. In these schemes, the tracks whose measurement indices on the past N scans are the same are regarded as "similar" and identified. Thus they are often referred to as N -scan or depth- N methods. In Figure 7-2, hypotheses 1 and 2 can be combined if $N = 3$.

The justification is that since each track distribution is driven by the measurements assigned to it, if two tracks share the same measurements in the recent scans (data sets) they should be similar. This scheme is criticized by Reid [20] on the ground that some events in the past may have a greater influence than the most recent N scans. However, since the dynamical nature of a target model removes this possibility, the N -scan approach is attractive because of its simplicity.

After identifying tracks according to the N -scan or depth- N criterion, we may have several identical hypotheses, i.e., hypotheses with the identical set of tracks. Then those hypotheses are combined in a natural way. In a sense, this approach may be actually viewed as combining tracks rather than combining hypotheses. In fact, since similarity is initially defined at the track level even in

SCANS

1

2

3

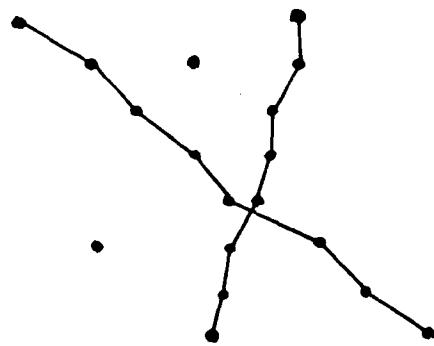
4

5

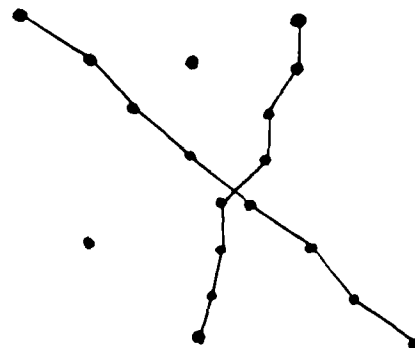
6

7

8



HYPOTHESIS 1



HYPOTHESIS 2

Figure 7-2 N-scan Hypothesis Combining

distribution-oriented methods, one may further classify the combining techniques according to when combining takes place. For example, distribution-oriented combining may be performed at the track level or at the hypothesis level. While track-level combining may seem to be more straightforward, it creates another problem, namely, how two distributions should be combined, since there is no natural weighting formula similar to that like (7.2) used in distribution-oriented combining at the hypothesis level.

To summarize, N-scan or depth-N methods have two major disadvantages, namely, the unresolved issues of (1) how to choose a right depth N, and (2) how to combine track state distributions. Just as in the case of pruning, many theoretical questions remain in combining hypotheses. Our current preference is distribution-oriented combining at the hypothesis level since there is a clear way for combining two probability distributions. The similarity condition, however, should be carefully chosen according to the physical nature of the particular problems and the chosen representation of each track distribution, etc.

7.3 Windowing

When an i.i.d. target model is used with other assumptions described in Section 5, each measurement data can be individually evaluated by likelihood functions given by (5.14), (5.15) and (5.17). When a track state distribution has a reasonable variance and the measurement errors are not exceptionally large, one can expect the track-measurement likelihood given by (5.15) to be very small for measurement data which are geometrically far from the expected position based on the state distribution associated with the track. Windowing techniques are

generally designed to set an appropriate threshold so that the track-measurement likelihood in such a case becomes zero rather than a small positive number.

Thus, one may consider such techniques to be a special kind of pruning, i.e., immediate pruning of branches based solely on one likelihood function. In other words, windowing is a method for eliminating a certain set of data to be associated with each track. For this reason, such a process is often called data validation. When track state distributions and measurement error distributions are both Gaussian, windowing can be accomplished by a classical chi-square test. As described in [16], this test may be performed in several steps. For example, the first step may consist of a χ^2 test (square test), and then a normalized square-of-innovation test (ellipse test), and finally, the likelihood function test itself.

Another view of windowing is that it is a part of the distribution representation and modeling process. According to this view, when the track state and measurement distributions are modeled as Gaussian, they really should be viewed as approximations of reality since such distributions can only have compact supports in the real world. For example, when the standard deviation of the measurement error is one mile, a data point 100 miles away from the mean of the track distribution should yield zero as its likelihood rather than a very small but positive number. The authors prefer this point of view to the pruning or approximation view. Thus any windowing process should be carefully designed so that the particular physical nature of the problem may be adequately reflected.

7.4 Clustering

The basic idea behind clustering is that two events taking place at locations far apart should be independent and can be evaluated separately. Clustering techniques have been described in an algorithmic form in [20] for general cases and more rigorously in [11] for a special case. When adequate windowing is performed, there is a natural way to avoid redundant calculations in evaluating hypotheses since the posterior probability of each hypothesis is the product of an appropriate set of likelihood functions as shown in Section 6. This constitutes another view of clustering.

Mathematically, clustering can be defined as follows.

Let Λ be the set of all non-zero-probability hypotheses at a given data set, i.e., for all $\lambda \in \Lambda$,

$$p(\lambda|Z) > 0. \quad (7.3)$$

For each possible track τ , the posterior probability of τ is given by

$$p(\tau|Z) = \sum_{\lambda \in \Lambda} p(\lambda|Z). \quad (7.4)$$

Let T be the set of all non-zero-probability tracks, i.e., for all $\tau \in T$,

$$p(\tau|Z) > 0 \quad (7.5)$$

Let C be any partition of T which satisfies the following condition:

For any pair (C', C'') of elements in C , such that

$C' \neq C''$, and any $\tau' \in C'$ and $\tau'' \in C''$,

$$\tau' \cap \tau'' = \emptyset. \quad (7.6)$$

This condition means that the non-zero-probability tracks are partitioned so that no two tracks which are in different elements (clusters) of the partition share a common measurement index. For each C in $\mathcal{C}$, let

$$\text{and } \Lambda_C = \{\lambda \cap C \mid \lambda \in \Lambda\} \quad (7.7)$$

$$I_C = \cup C. \quad (7.8)$$

Then clustering is the process of generating $\{(I_C, \Lambda_C), C \in \mathcal{C}\}$. Each Λ_C is the set of local hypotheses which consists only of tracks in C . For each local hypothesis $\lambda_C \triangleq \lambda \cap C$, the local posterior probability is given by

$$p_C(\lambda_C | Z) = \sum \{p(\lambda | Z) \mid \lambda \in \Lambda, \lambda \cap C = \lambda_C\}. \quad (7.9)$$

Then it is clear from (6.1) that

$$p(\lambda | Z) = \prod_{C \in \mathcal{C}} p_C(\lambda_C | Z) \quad (7.10)$$

Each global non-zero-probability hypothesis can thus be represented as a union of local hypotheses (one from each cluster) and its posterior probability is the product of the local probabilities of the local hypotheses. From this definition, we see that clustering involves partitions at all levels, i.e., hypothesis, track and measurement levels.

Equation (7.10) is called the orthogonality condition. According to the above definition of clustering, the orthogonality condition should hold whenever the non-intersection condition of the tracks given by (7.6) holds. However, when some approximation techniques such as pruning and combining are employed, this may not be true any more. The clustering technique described in [20] is a method in which the orthogonality condition is maintained without checking the non-intersection condition of the tracks.

This technique can be described in terms of algorithmic procedures as follows:

1. Initialization of a Cluster

Whenever there is a measurement such that the newly detected target likelihood L_{oj} is not zero but the track-measurement likelihood L_{ij} with every existing track is zero, a new cluster should be created out of the measurement.

2. Cluster Merging

Whenever there is a measurement such that each of the corresponding track-measurement likelihood functions with two or more tracks in different clusters is non-zero (in other words, there is a measurement lying in the intersection of the validation regions of two tracks in two different clusters), such clusters should be merged before the measurement can be processed. The merging of the clusters is accomplished by forming the union of the tracks in the clusters, generating the global hypotheses and evaluating the global probabilities as the products of the local probabilities.

3. Cluster Splitting

Whenever a track with probability one, i.e., one contained in every local hypothesis in a cluster is found, it is split to form a new cluster consisting of one hypothesis with the sole track. Of course, the local probability of such hypothesis is one.

It is the authors' opinion that any serious attempt on practical problems should not be made without clustering, particularly in a problem in which a large area is covered. One may even assert that clustering is probably the most powerful hypothesis management technique in controlling the number of hypotheses. Of course, how successful a clustering technique is depends on the external conditions such as target density, measurement errors and targets dynamics.

The clustering procedure described above does not necessarily guarantee the finest clustering. The finest clustering may be found according to our mathematical definition of clustering. The test of the non-intersection condition can be easily implemented if we identify two tracks with the same measurement indices in a certain number of the most recent scans just like in a measurement-index-oriented combining scheme. The orthogonality condition can be met by modifying the local probabilities using appropriate approximation techniques whenever possible. This constitutes a new cluster-splitting technique which may improve on that described above. Although this newly proposed version of clustering seems promising, we do not have any actual implementation experience yet.

8. IMPLEMENTATION

In the previous sections, we have presented a unified theory for multi-target tracking and classification. In this section we describe the Generalized Tracker/Classifier (GTC) which implements the algorithms developed in the theory. The Generalized Tracker/Classifier has a structure given by Figure 1-1. It has three off-line modules and four on-line modules.

The three off-line modules are:

1. Generalized Target Dynamics Model
2. Environmental Model
3. Sensor Model

These represent the knowledge about the physical world that needs to be considered in the design and selection of the algorithms. Mathematically, they are described in Section 2 and include almost all other models as special cases.

The four on-line modules are:

1. Hypothesis Formation
2. Hypothesis Evaluation
3. Filtering and Parameter Estimation
4. Hypothesis Management

The hypothesis formation module generates the data-to-data association hypotheses which identify the data reported at different times from different sensors with the same targets. It carries out the function described mathematically in Section 3. This module uses only the information in the measurement indices and does not require the actual values of the measurements. Since no model information is needed, this module is model

independent. Although the hypotheses are defined as collections of tracks which are in turn collections of measurement indices, an equivalent and more convenient representation for implementation is by means of a tree. In this representation, each level in the tree corresponds to a measurement index and a node corresponds to a target. Hypothesis generation given a new sensor report then reduces to the expansion of the tree and a branch of the tree represents a particular data-to-data association hypothesis. The concepts of the parent of a hypothesis, defined earlier in Section 3, is also obvious from this representation.

Figure 8-1 shows a hypothesis tree for two data sets each with two measurements in each. The tracks associated with each hypothesis are also given. For example, hypothesis 24 associates y^1_1 and y^2_1 with the same target (track 5), and y^1_2 with a different target (track 2). It thus hypothesizes y^2_2 to be a false alarm. Note that from two data sets with two measurements each, we have eight possible tracks and a total of 34 possible hypotheses.

The hypothesis evaluation module evaluates the probability of each hypothesis given the information available. For general target models, it performs the function given by equations (4.2) and (4.14). If i.i.d. target models are assumed, evaluation is through equations (5.13) to (5.17). Equation (5.13) is quite independent of the particular models and provides the structure of the evaluation. Equations (5.14) to (5.17) evaluate the likelihoods of various associations and are model dependent. Thus the implementation of the hypothesis evaluation module consists of a (model-independent) track-measurement cross reference table and the (model-dependent) computation of likelihoods. As we have seen in Section 6, this module is considerably more general and realistic than is currently available in the literature.

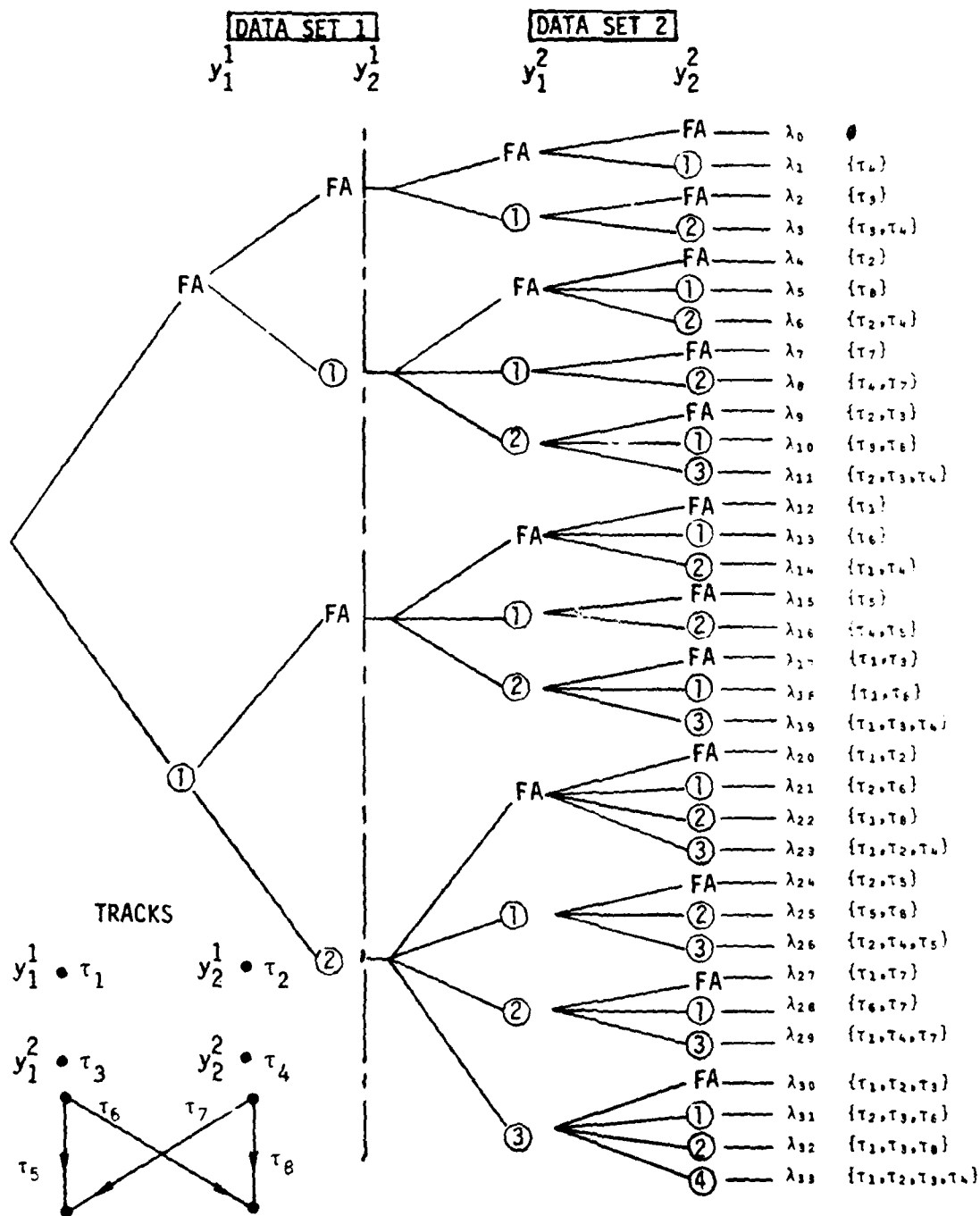


Figure 8-1 Hypothesis Tree

The filtering and parameter estimation module implements equations (4.15) and (4.18) for general target models and equations (5.19) to (5.25) for i.i.d. target models. They provide the necessary density or distribution functions for hypothesis evaluation. Conceptually, they solve the ordinary estimation problem when the origins of the measurements are known. However, even for the i.i.d. case, approximations are often needed to solve the general nonlinear estimation problem. When enough approximations are made, as discussed in Section 6, the standard Kalman filter often results. One novel feature of the GTC is that the density of undetected targets as well as the expected number of undetected targets is constantly updated at each scan. This allows for better track initiation than is currently available in other algorithms. As is expected, this module is heavily model dependent.

The hypothesis management module implements the ideas discussed in Section 7, and is crucial for the successful operation of the GTC. It is model independent in the sense that the techniques involved are applicable to a wide class of scenarios. The user, however, can select certain parameters to conform with the computational and memory requirements or to reflect his knowledge about the complexity of the situation. These parameters are the pruning thresholds in hypothesis pruning, the thresholds for testing similarity in hypothesis combining, the size of the window in windowing, and the criterion used in cluster splitting. As has been discussed in Section 7, a good theory of hypothesis management is still lacking, and thus the hypothesis management module usually requires a lot of tuning before the entire GTC will function properly. Some techniques in artificial intelligence may be useful here.

9. AN EXAMPLE - TRACKING WITH AN MTI RADAR SYSTEM

In this section we present an example to illustrate the main features of the Generalized Tracker/Classifier (GTC). Although the algorithm has been successfully tested for more complicated scenarios, a simpler case has been chosen here for ease of exposition. The results of some Monte Carlo simulation runs will be used to highlight the effects of some external and internal parameters on the performance of the tracker.

A moving target indicator (MTI) is a radar which is specially designed to detect only moving objects while avoiding ground clutter. For the purpose of this section, we assume a type of MTI which provides a three-dimensional measurement (range, bearing and radial velocity) for each detected object. The altitude difference between the sensor and the targets is assumed to be negligible. The radial velocity RV of an object at x_o in a fixed three-dimensional cartesian coordination system with respect to a fixed sensor at x_s in the same coordinate system is defined as

$$RV = \frac{(x_o - x_s \mid \dot{x}_o)}{\|x_o - x_s\|} \quad (9.1)$$

where $\dot{x}_o$ is the velocity vector of the object (the time-derivative of x_o), $(\cdot \mid \cdot)$ is the inner product of two vectors, and $\| \cdot \|$ is the norm of a vector. In general, an MTI detects objects with a nonzero probability only when their radial velocities exceed a certain fixed threshold, called the minimal discernible velocity (MDV).

9.1 Scenario Description

We are concerned with the surveillance of terrain with a road network using an MTI radar which is fixed (e.g., on a helicopter). The radar is assumed to be at a relatively low altitude, and thus the positional information is actually two-dimensional. The terrain, especially the road network, is assumed to be known. The objective is to track multiple targets moving over the terrain using a sensor which can provide both positional and velocity information.

Our specific scenario consists of a single straight road segment in the field of view of an MTI radar (Figure 9-1). The road segment is assumed to be the only part of a longer road which is observable from the radar. A target (an object to be tracked, such as a vehicle) can be identified by its position on the road as measured from the point which is closest to the MTI radar. We shall use the i.i.d. target model and the corresponding tracking algorithm described in Section 5, together with all the other assumptions made in that section. A target on the road is assumed to move at an almost constant velocity, i.e., the velocity is assumed to be a random variable plus a random walk with an intensity which is small compared with the variance of the constant part. The scan interval, Δt , is assumed to be constant. Thus, the individual target dynamics are modeled as

$$\begin{bmatrix} u(t_{k+1}) \\ v(t_{k+1}) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u(t_k) \\ v(t_k) \end{bmatrix} + \begin{bmatrix} 0 \\ w_k \end{bmatrix} \quad (9.2)$$

where $\{t_k, k = 0, 1, 2, \dots\}$ is the sequence of scan times, $u(t)$ is the position

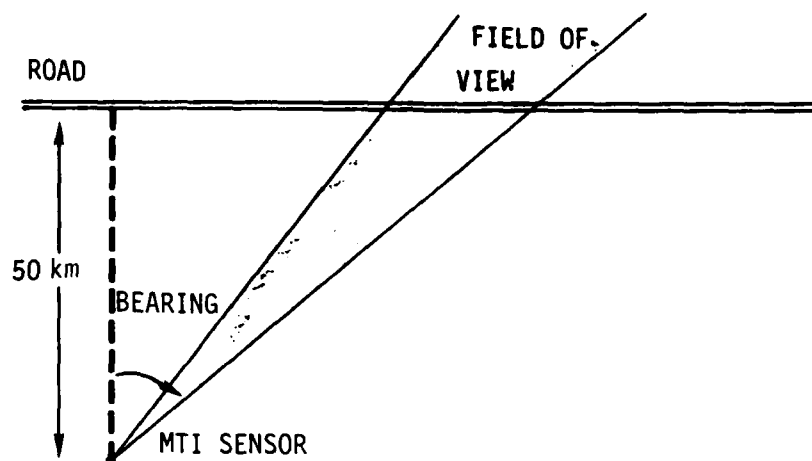


Figure 9-1 An MTI Tracking Example

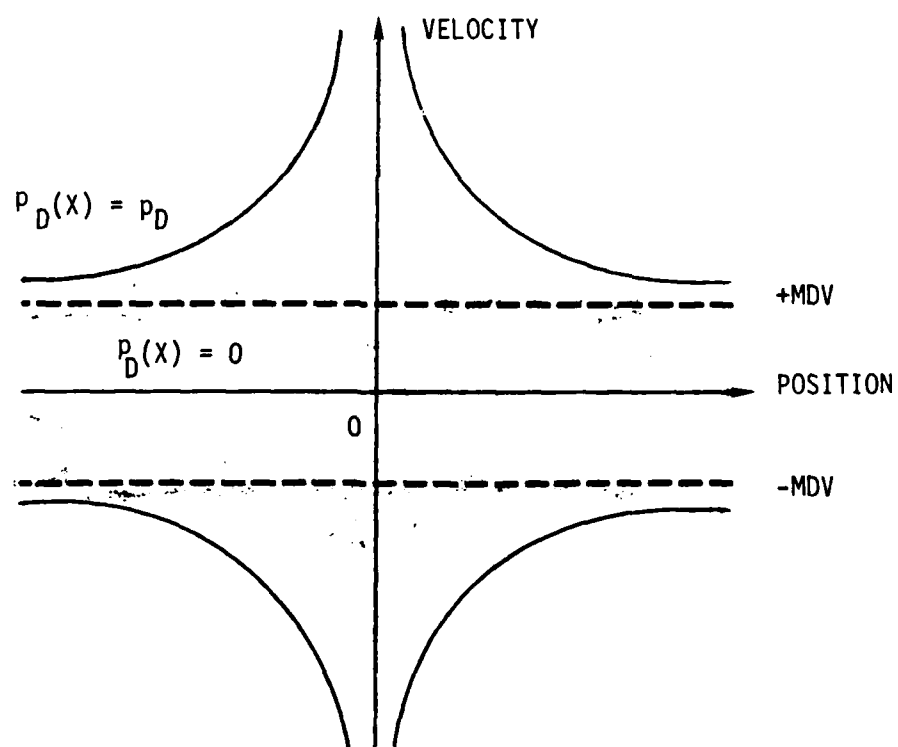


Figure 9-2 Probability of Detection by MTI Radar

of a target on the road at time t , $v(t)$ its velocity and $\{w_k, k = 0, 1, 2, \dots\}$ is a sequence of zero-mean i.i.d. Gaussian random variables. The initial position and velocity are unknown a priori. Therefore, they are assumed to be uniformly distributed over some reasonable ranges. The probability of a target being detected is a function of its state x (position and velocity), i.e.,

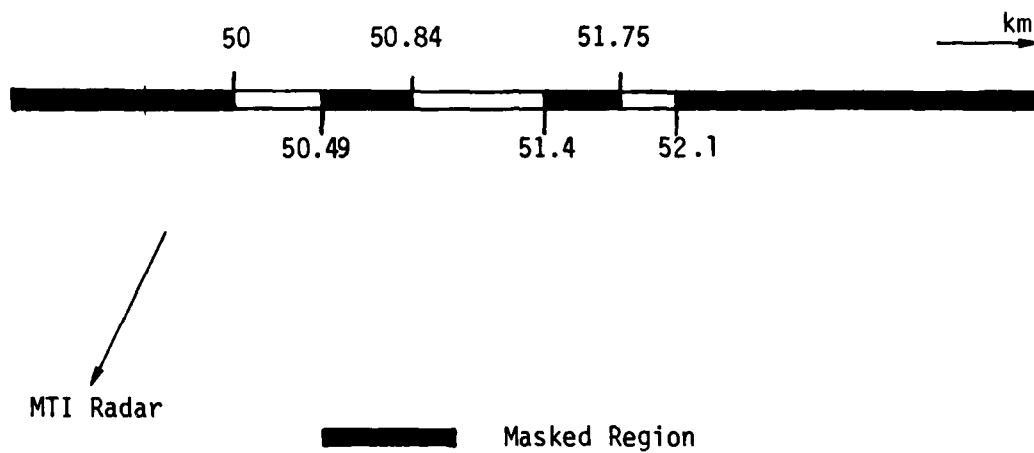
$$p_D(x) = \begin{cases} p_D & \text{if } |RV| \geq MDV \\ & \text{and } u \text{ is not masked} \\ 0 & \text{otherwise} \end{cases} \quad (9.3)$$

Figure 9-2 shows the detection probability contour when the masks are ignored. The masks on the road, created by the terrain (e.g., hills), tall trees and foliage, etc., may be known exactly or only probabilistically. In this example we assume the former case. The field of view and the masked regions used for the present example are shown in Figure 9-3.

Given that a target has been detected, the measurement (range, bearing and radial velocity) consists of the true values plus independent additive zero-mean Gaussian noises. The distribution of false alarm measurements is assumed to be uniform over the measurement space. However, the measurements which are unlikely to originate from the road are dismissed as false alarms before they are used by the tracking algorithm.

9.2 Likelihood Functions and Filtering

In our opinion, the calculation of q , the probability density of an undetected target, is a crucial part of the tracking algorithm. This is actually one part of our algorithm which is clearly different from other existing tracking algorithms. In the present example, the density q of the position



Unmasked Regions: [50, 50.49]
 [50.84, 51.4]
 [51.75, 52.1]

Figure 9-3 Road Model

and the velocity of an undetected target is approximated by a step function

$$q(u, v) = \sum_{m, n} Q_{mn} \chi(u; I_m^n) \chi(v; J_n) \quad (9.4)$$

where $\chi(\cdot; A)$ is the indicator (characteristic) function of the set A . The quantization intervals, I_m and J_n , are chosen to be

$$\begin{aligned} I_0^n &= [x_{\min}^n, x_{\ell\ell}^n), \\ I_m^n &= [m \cdot \Delta x, (m+1) \Delta x), \text{ for } m \in \{1, \dots, 30\} \\ I_{31}^n &= [x_{r\ell}^n, x_{\max}^n], \\ J_0 &= (-MDV, MDV), \\ J_1 &= [MDV, V_1), \quad J_{-1} = -J_1, \\ J_2 &= [V_1, V_2), \quad J_{-2} = -J_2, \\ J_3 &= [V_2, V_3], \quad J_{-3} = -J_3, \end{aligned} \quad (9.5)$$

where $x_{\ell\ell} = 50$ km is the left limit of the field of view, $x_{r\ell} = 52.1$ km the right limit and the other values are $\Delta x = 70$ m, $MDV = 2$ m/s, $V_1 = 10$ m/s, $V_2 = 20$ m/s, and $V_3 = 30$ m/s. $x_{\min}^n$ and $x_{\max}^n$ are shifted in each extrapolation operation. Their initial values are chosen so that there is still some inflow of density at the end of all the scans and that the position of a vehicle is uniformly distributed over the field of view.

The exact extrapolation of the density q using the dynamic equation (9.2) requires complicated numerical integration. However, since we assume that

each target moves at an almost constant velocity, we can approximate the extrapolation by a simple translation of $q(.,v)$ for each velocity v by v times Δt . In order to maintain the step-function form of (9.4), the shifted results are integrated after each extrapolation. After each scan, the density q is updated by $(1-p_D(x))$. For this purpose $p_D(.)$ is approximated by a step function with the same quantization cells as those for the density q .

An example of the calculation of the density q is shown in Figure 9-4, in which $\{Q_{mn}, m = 0, \dots, 31\}$ for $n = 1$ is shown after each operation. $n = 1$ means that the target's velocity is between 2 m/s and 10 m/s. Thus, the extrapolation operation, (2) $\rightarrow$ (3), (4) $\rightarrow$ (5), (6) $\rightarrow$ (7), (8) $\rightarrow$ (9), is a right shift with $\bar{v}\Delta t = 90$ m where $\bar{v} \approx 6$ m/s is the average velocity and $\Delta t = 15$ s is the scan interval. In Figure 9-4, this right shift operation is approximated to maintain the same quantization cells.

The probability of detection, p_D , is assumed to be .8, a relatively high value. After the first scan, the probability density drops drastically in the unmasked regions due to the relatively high p_D . The change in value in the unmasked region after one scan reflects the effect of the dependence of the detection probability on the radial velocity. Thus, after scan 0, an undetected target in the velocity sector (2 m/s to 10 m/s) is very likely to be either in the masked regions or outside the field of view. At scan 1, as shown in Figure 9-4 (a), an undetected target may appear on the right side of a masked region. However, if it does move out of a masked region, it should be detected with probability .8. Thus, the density in the unmasked regions drops again (Figure 9-4 (a)). As these steps continue, we expect almost all the targets which have been hiding in the masked regions to eventually come out and be detected.

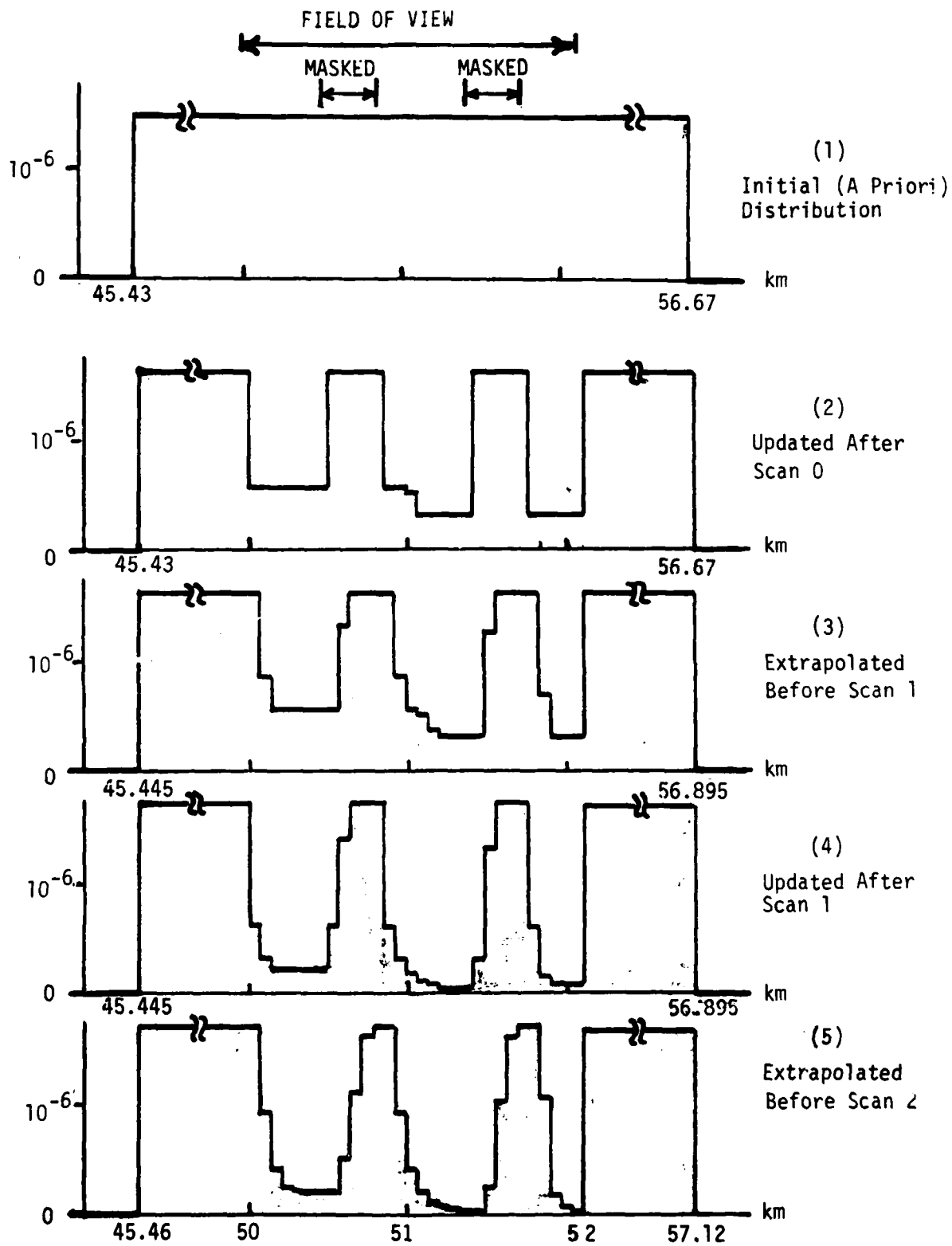


Figure 9-4(a) Density of Undetected Targets (up to Scan 2)

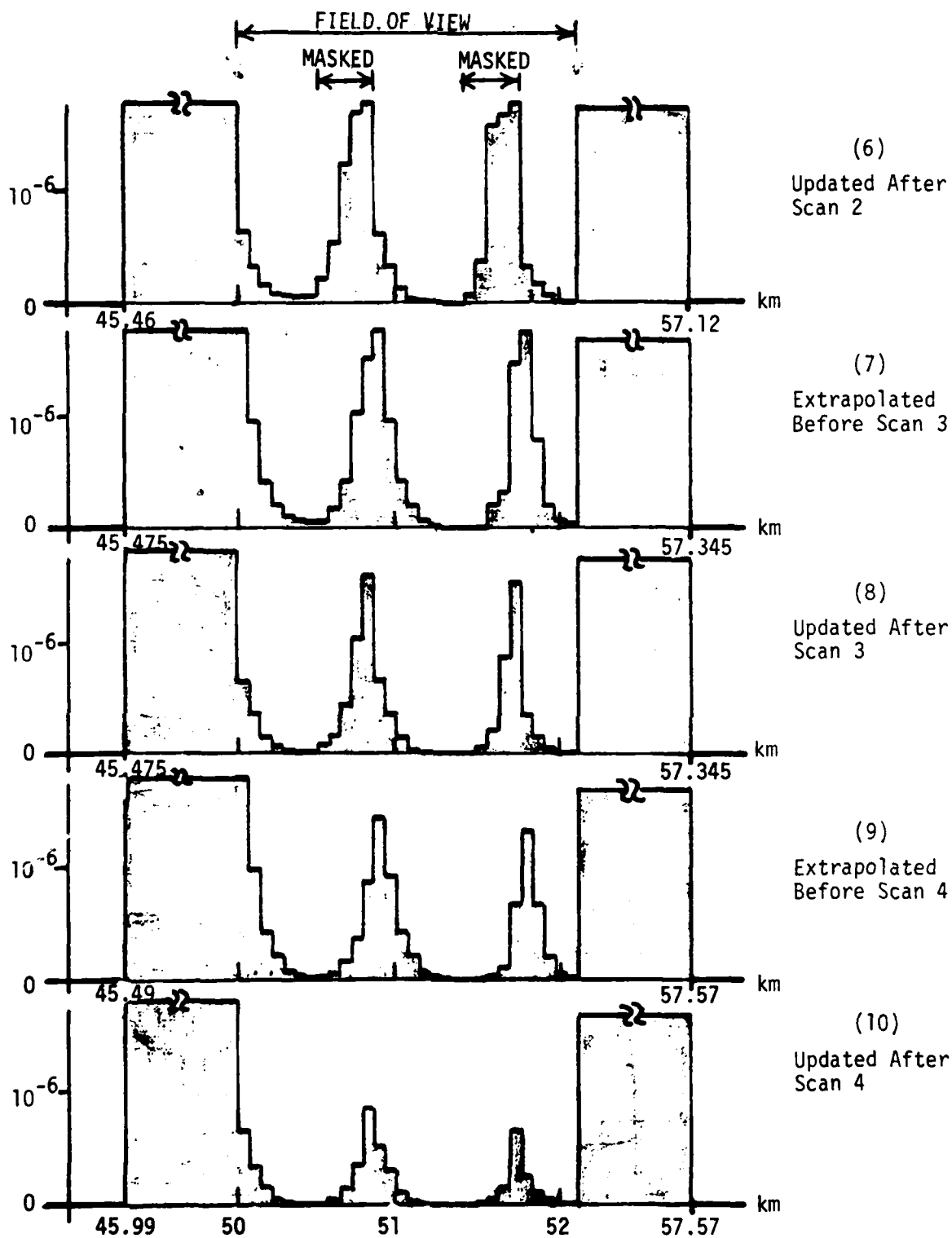


Figure 9-4(b) Density of Undetected Targets (after Scan 2)

Figure 9-5 (a) and (b) show L_{00} , the likelihood (also a probability in this case) of an undetected target remaining undetected, and v , the expected number of undetected targets versus the number of scans. The relatively high values of L_{00} reflect our assumption that the density q spreads over a wide region where a target can never be detected. The initial value for v is the a priori expected total number of targets and is set to be 27, i.e., we expect about three or four targets to appear in the field of view within the ten scans. Because of the high p_D , v drops rapidly in the first few scans. Then it decreases more gradually, due to targets moving into the unmasked regions. We should note that v is updated by $v \leftarrow v L_{00}$. If these variables continue to be calculated for further scans, L_{00} converges to 1 and v to a positive constant, reflecting the fact that eventually, all the targets, detected or not, leave the field of view and will never be detected.

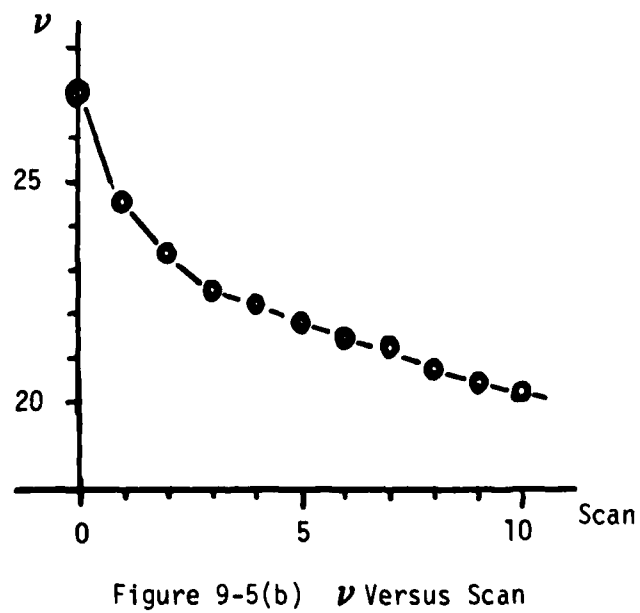
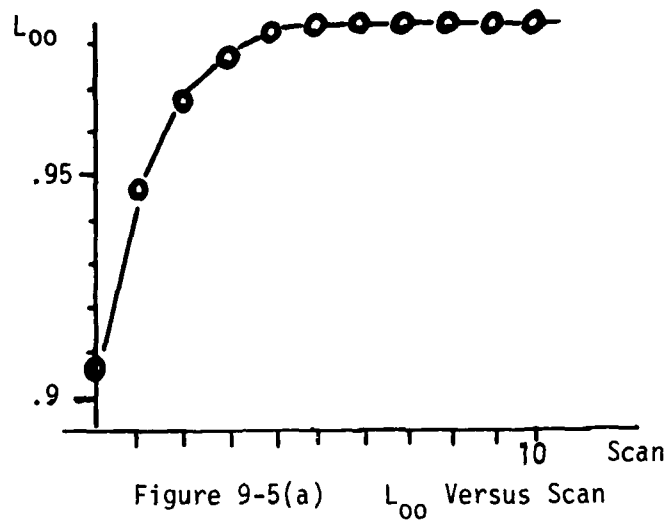
The density q is used to calculate the new target (or more precisely, the newly detected target) likelihood function. Figure 9-5(c) shows some plots for this likelihood function. In this figure, the likelihood for measurement (r, α, RV) is plotted for

$$\cos \alpha = \frac{n}{D}$$

and

$$RV = 10 \text{ m/s.}$$

where $D = 50 \text{ km}$ is the distance between the MTI and the road, r is the range measurement, α the bearing and RV the radial velocity. The exact calculation of the new target likelihood function also requires a complicated numerical integration. A relatively crude approximation is used, in which a zero-mean one-dimensional Gaussian distribution is approximated by a step function,



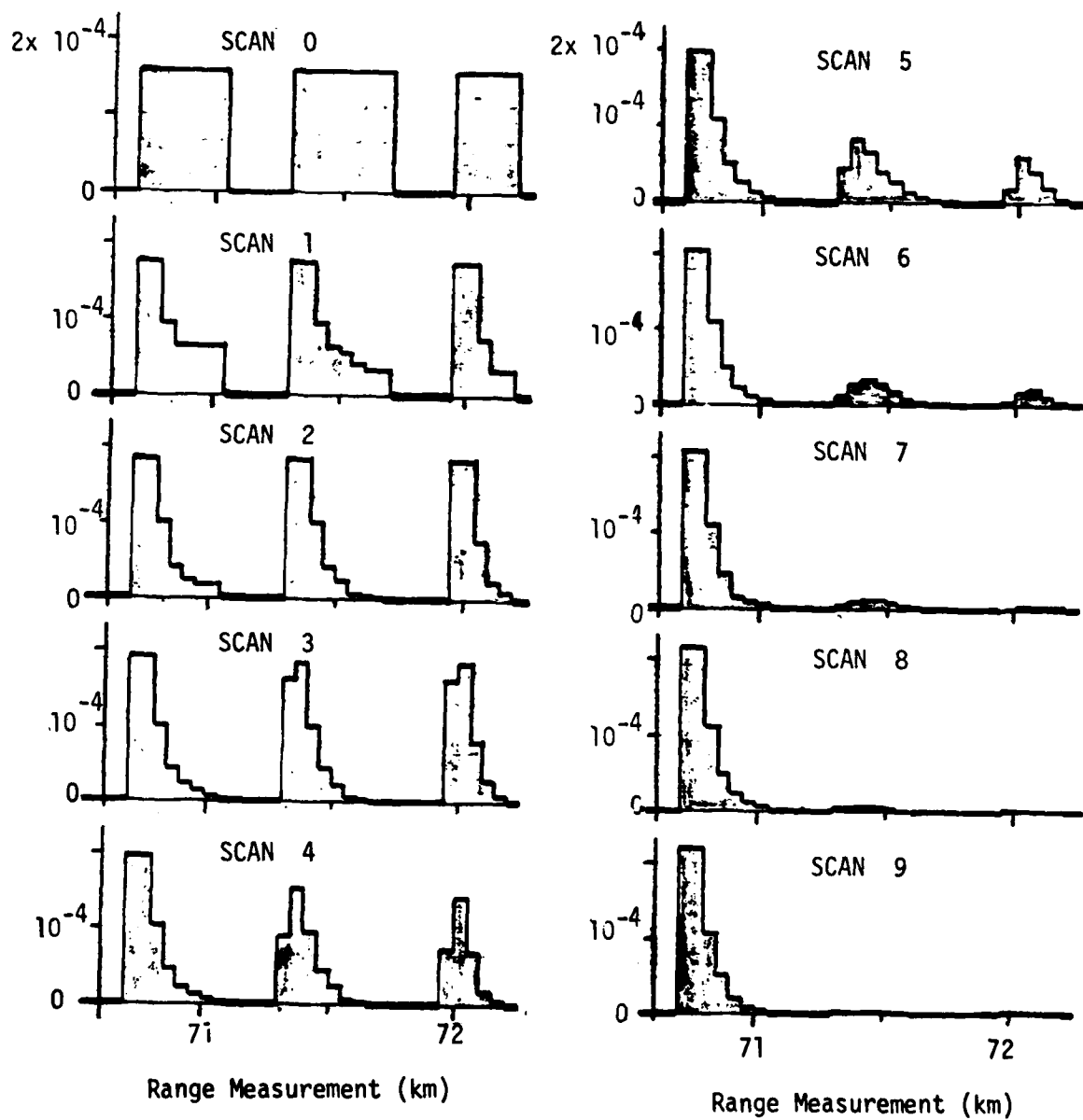


Figure 9-5(c) New Target Likelihood (as a function of range)

$$p(x) \cong \sum_{i=-1}^1 a_i \chi(x; L_i) \quad (9.6)$$

where $L_{-1} = [-2\sigma, -\sigma)$, $L_0 = [-\sigma, \sigma)$, $L_1 = [\sigma, 2\sigma)$ with $a_i = \int_{L_i} p(x)dx$ and the standard deviation σ .

As shown in Figure 9-5(c), the new target likelihood function has a positive value only in the field of view and in the unmasked regions. In each unmasked region, since RV is positive (i.e., the target is moving to the right), the left edge has a greater value. As the scans continue, targets hidden in the masks eventually come out and are detected, and hence the only likely position for a new target to appear is the left edge of the field of view.

Other likelihood functions are calculated using appropriate approximations. Once a track is formed by a single measurement, its state distribution is always represented by a two-dimensional Gaussian distribution, i.e., the mean and the variance. When a track is associated with a measurement, its distribution is updated by the extended Kalman filter with the linear target model (9.2) and the nonlinear measurement model. When a track is not associated with a measurement, the distribution is not updated although the corresponding likelihood function (missed-target likelihood) is numerically calculated for evaluating hypotheses. This approximation is justified because of the relatively accurate range measurement.

9.3 Hypothesis Management

The adaptive-threshold/adaptive-breadth pruning strategy of Section 7 is used. In this pruning method, hypotheses are sorted according to their posterior probabilities and the low probability hypotheses are removed from the tail so that the number of remaining hypotheses is minimum while the

sum of their probabilities is greater than a specified threshold, i.e.,

$$\sum_i p(\lambda_i) \geq 1 - \varepsilon_p \quad (9.7)$$

To avoid any confusion with the terminology commonly used for fixed-level pruning, we call ε_p the threshold instead of $1 - \varepsilon_p$.

Our combining is based on the track state distributions. A relatively simple method has been chosen in which two distributions with means m_1 and m_2 are judged to be combinable if $||m_1 - m_2|| \leq \varepsilon_c$. All hypotheses with the same number of tracks are examined for their pairwise combinability. Once two hypotheses are combined, combined tracks are represented by Gaussian distributions with the appropriate means and variances. Once two tracks are combined into one track in this way, all the hypotheses which refer to such tracks are modified so that the combined tracks become identical throughout these hypotheses. Windowing is based roughly on the $3-\sigma$ principle. In our case, data validation includes also the radial velocity test and the mask test. The clustering described in Section 7 is also used. However, only tracks with probability one are split from clusters.

9.4 A Sample Run

Figure 9-6 shows a data set for a single sample run described below. The parameters are summarized as follows:

D	$= 50$ km	distance between the road and the sensor
Δt	$= 15$ s	scan interval
MDV	$= 2$ m/s	minimal discernible velocity
σ_r	$= 3$ m	range measurement error standard deviation
σ_α	$= .1$ rad.	bearing measurement error standard deviation
σ_{RV}	$= .2$ m/s	radial velocity measurement error standard deviation
v_{FA}	$= 1$ per scan	expected number of false alarms
v_o	$= 27$	expected total number of targets
p_D	$= .8$	probability of detection
ϵ_p	$= 5\%$	pruning threshold
ϵ_c	$= 10$ m	combining threshold

The above set of parameters is used as the base case. The effects of varying three parameters, namely, the expected number v_{FA} of false alarms per scan, the range measurement error standard deviation σ_r and the pruning threshold ϵ_p are examined in Monte Carlo simulations. The results are shown later.

In Figure 9-6, the actual target motions are shown by lines (1) through (4). This target scenario is also used in the Monte Carlo simulations. Targets (1) through (3) move relatively slowly at 10 m/s towards the left. Target (4) moves at a higher speed, 15 m/s, to the right. Target (1) is masked at scans 0, 1 and 2, appears at scan 3 for the first time, and is not detected at scan 5 although it is in the unmasked region. Target (3) appears at scan 3 and is falsely dismissed at scan 8. The

false alarms appear almost uniformly over the field of view. All the measurements (including false alarms) are shown in Figure 8-6 by their projection on the road using the relatively accurate range measurements.

Figures 9-7 to 9-9 display the details of the hypothesis formation process for the data shown in Figure 9-6. Figures 9-7(a) to 9-7(d) show the complete histories of several clusters. Figure 9-8(a) to 9-8(g) show the clusters, their hypotheses and tracks and the probabilities for each scan, while Figure 9-9 is a summary history of all the clusters. The following is a scan-by-scan description.

SCAN 0: There are two measurements: one from target (2) and the other a false alarm. The ratio of the new target likelihood over the false alarm likelihood determines the probability of the new target hypothesis, ${}_0\lambda_1^1 = \{ {}_0\tau_1^1 \}$ or ${}_0\lambda_1^2 = \{ {}_0\tau_1^2 \}$, as shown in Figure 9-8(a). In Figures 9-8(a) through 9-8(g), hypothesis ${}_k\lambda_j^i$ and track ${}_k\tau_j^i$ are indexed by pre-subscript k, superscript i and subscript j, indicating the j-th hypothesis or track in the i-th cluster at scan k. Each measurement is identified by its measurement index (k,j), indicating the j-th measurement at scan k.

The two measurements, (0,1) and (0,2) at scan 0, are almost equally target-like. The (local) probability of measurement (0,1) originating from a target, .753, is consequently almost the same as that of measurement (0,2) in the other cluster, .781. Since there is no a priori cluster, two clusters are formed as shown in Figures 9-7(a), 9-7(b) and 9-8(a). In Figures 9-7(a) and 9-7(b), each number at a node of the tree represents the measurement association specified by a particular hypothesis. In this tree representation, a positive

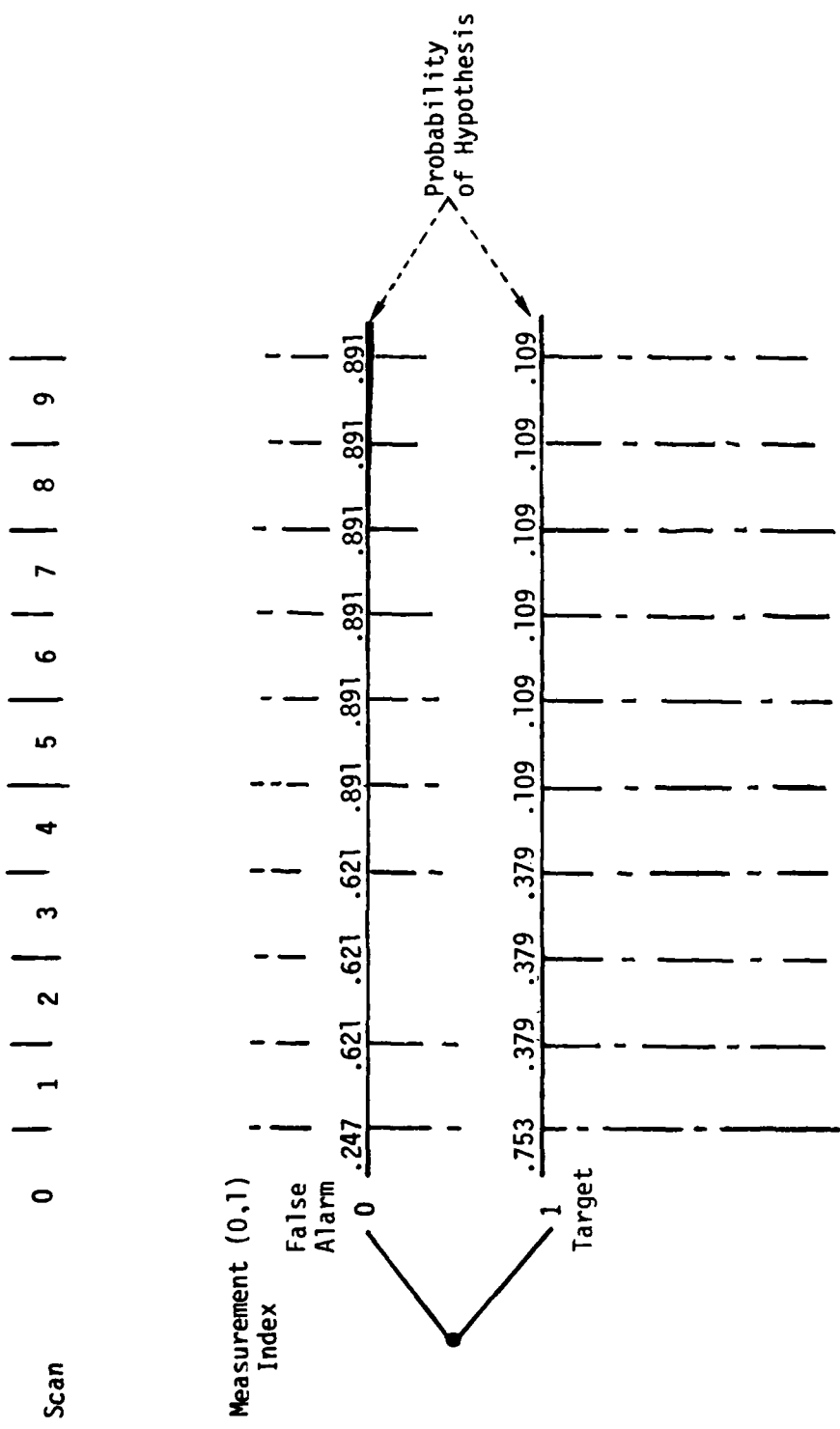


Figure 9-7(a) Cluster 1

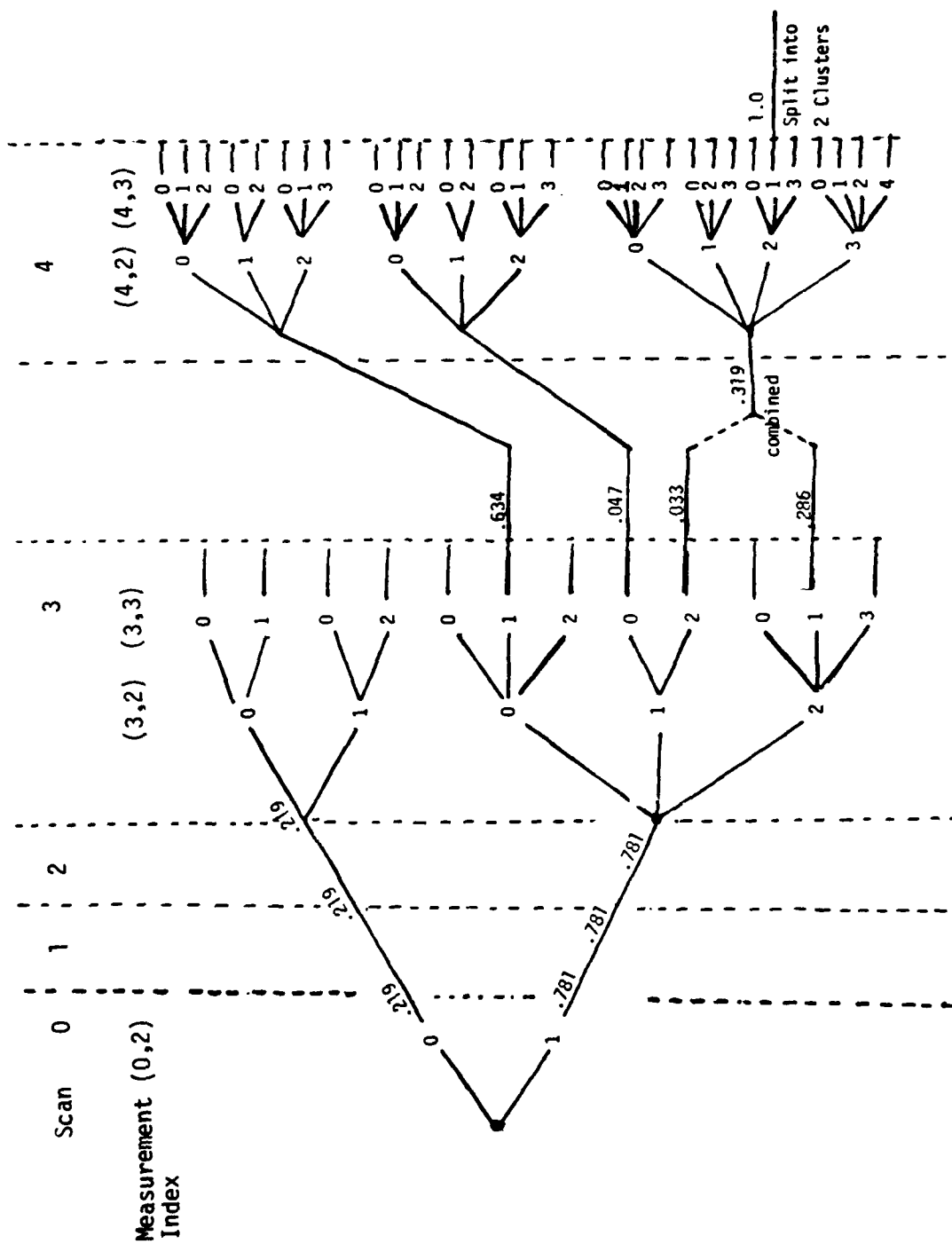


Figure 9-7(b) Clusters 2 and 3 Before Split

integer i at a node means that the i -th target in a hypothesis is hypothesized to generate the measurement indicated at the top of the column which contains this node (the node level indicator). This indexing of targets is unique only for a particular hypothesis. A "0" at a node indicates that the particular hypothesis hypothesizes the corresponding measurement to be a false alarm.

At the end of processing scan 0 measurements, there are two clusters, each of which has two hypotheses. Therefore, there are four global hypotheses. The best global hypothesis has probability .588 and hypothesizes two targets of which one is actually a false alarm.

SCAN 1: There are three measurements. The second measurement, (1,2), falls in a masked region and is therefore thrown away immediately since it is a false alarm with probability one. As seen in Figure 9-6, none of the two remaining measurements falls into the validation windows of the two existing tracks. As a matter of fact, the extrapolation of the two previous tracks indicates that both are in the masked regions if they ever exist as real targets. Therefore, the hypotheses, their probabilities and tracks remain unchanged, as shown in Figures 9-7(a), 9-7(b) and 9-8(a) for these two existing clusters.

Two additional clusters, 4 and 5, are created from the two measurements (1,1) and (1,3), as shown in Figures 9-7(d) and 9-8(a). Measurement (1,1) appears on the left edge of the field of view and its radial velocity measurement indicates a rightward movement. Therefore, the hypothesis that this measurement originates from a target just coming into the field of view is well supported. The probability of this hypothesis (actually the truth) is calculated to be .696.

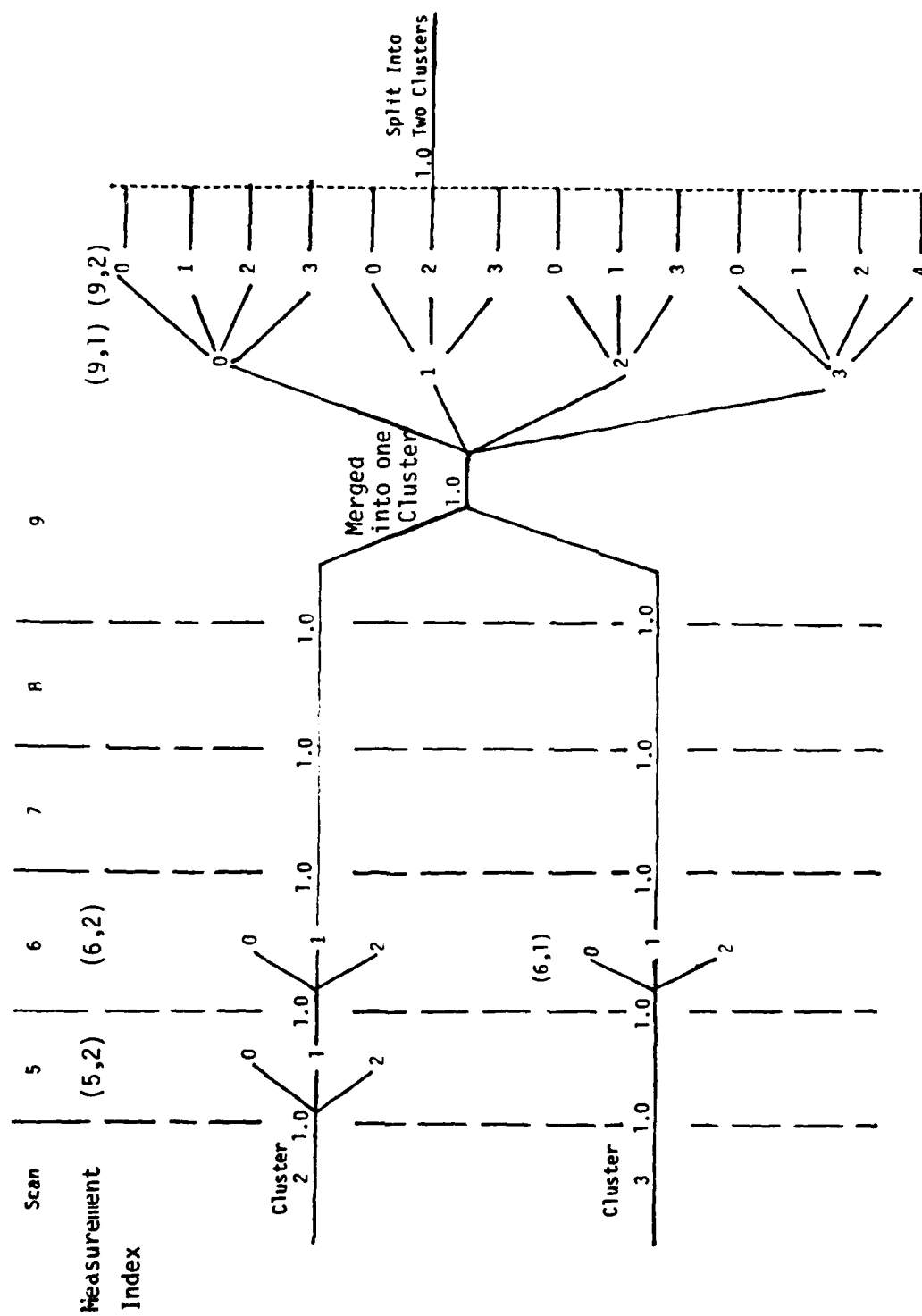


Figure 9-7(c) Clusters 2 and 3 After Split at Scan 4

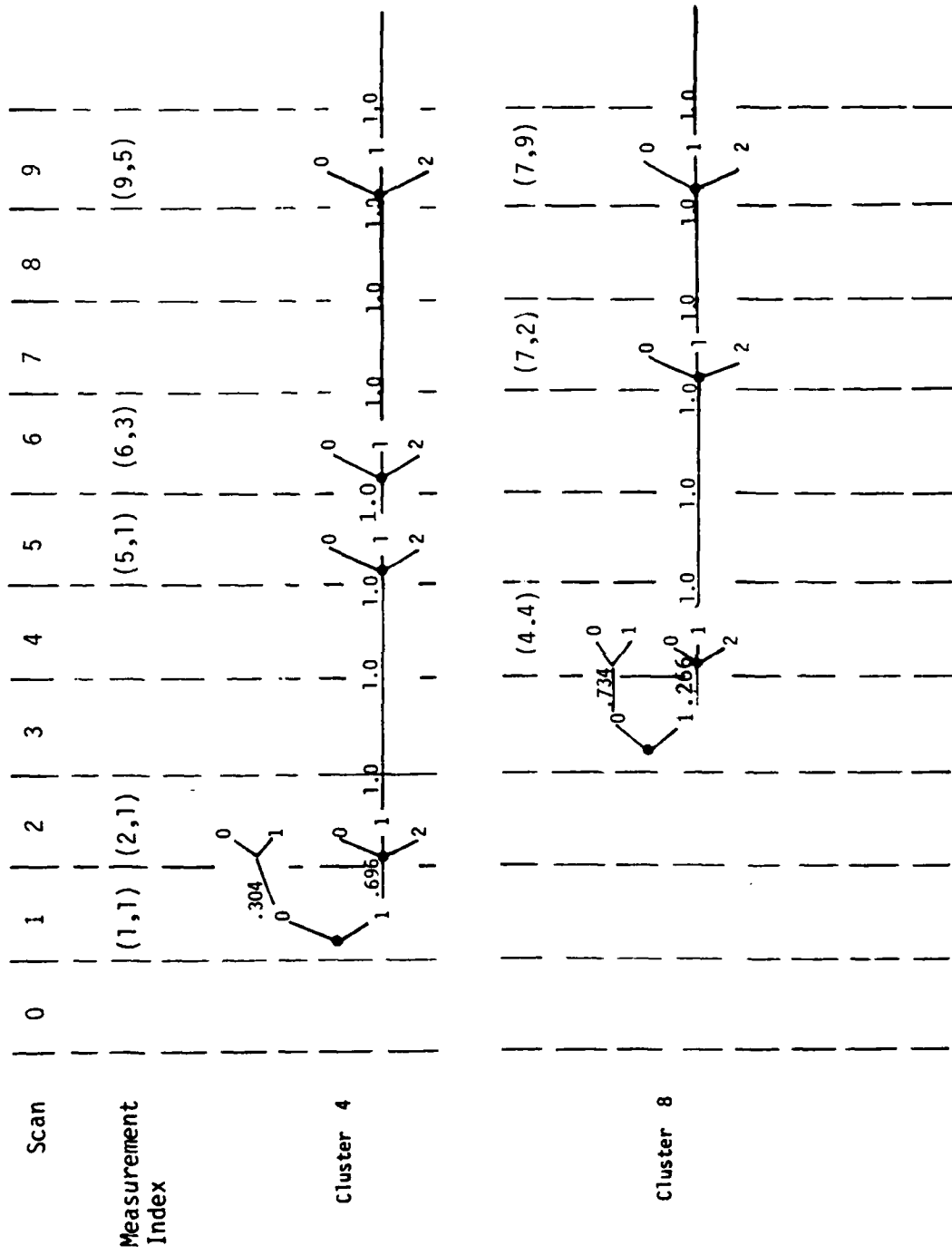
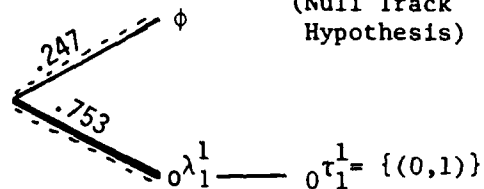


Figure 9-7(d) Clusters 4 and 8

1

Scan 0

Cluster 1



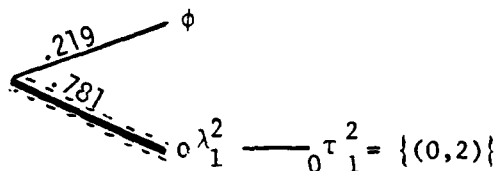
P P= Probability of Hypothesis

--- Truth

== Best Hypothesis

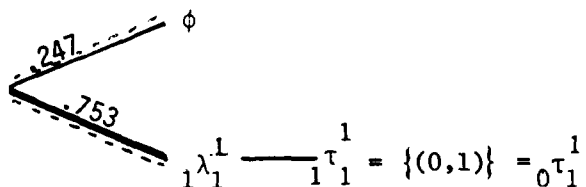
> Predecessor Indicator

Cluster 2

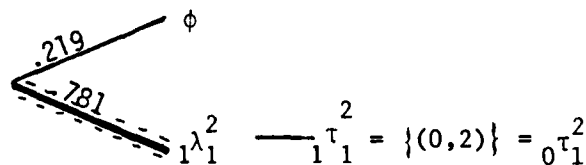


Scan 1

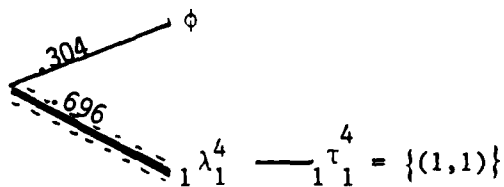
Cluster 1



Cluster 2



Cluster 4



Cluster 5

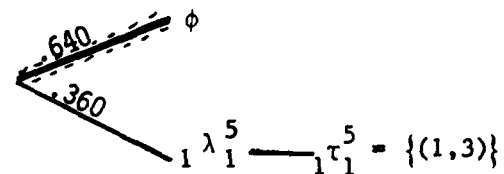


Figure 9-8(a) Hypotheses/Tracks (Scan 0 and 1)

On the other hand, the other measurement (1,3) appears on the left edge of an unmasked region and its radial velocity measurement indicates a movement towards the left. This means that, if it has originated from a target, the target must have been in the same unmasked region at scan 0 but have escaped detection. The unlikely nature of this event is reflected by the relatively low probability of .36 for the new target hypothesis $1\lambda_1^5$ in cluster 5. In reality, this measurement is a false alarm.

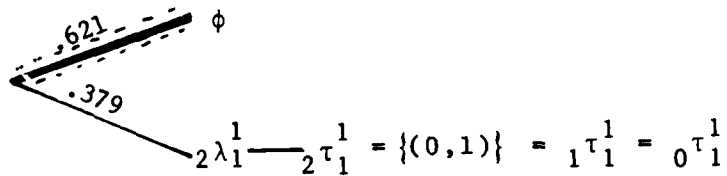
Thus, we have four clusters each of which has two hypotheses. Therefore, we have 16 (global) hypotheses. The best hypothesis has probability .262. It contains two real targets as tracks and one false track. The truth appears in the fourth best hypothesis which has only a probability of .086.

SCAN 2: There are two measurements, (2,1) and (2,2). (2,1) falls into the validation region of track $1\tau_1^4$ of cluster 4 which has been initiated in scan 1. This measurement is the sole measurement which falls into this validation region. Therefore, the hypotheses in cluster 4 are expanded to five new hypotheses as shown in Figure 9-7(d). The best hypothesis of these five hypothesizes that measurements (1,1) and (2,1) originate from the same target. Its probability is .9999449. The pruning threshold is .05, and hence the remaining four hypotheses are pruned. Consequently, there is only one probability-one hypothesis $2\lambda_1^4$ left in cluster 4. At this point, it has been confirmed that there is at least one detected target with probability one.

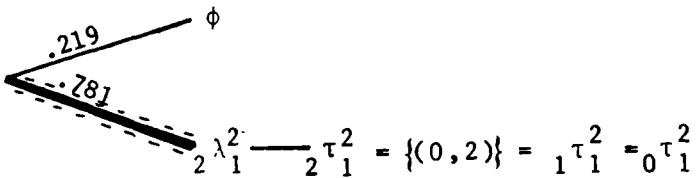
On the other hand, the remaining clusters 1, 2 and 5 have no track with a

Scan 2

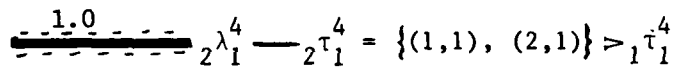
Cluster 1



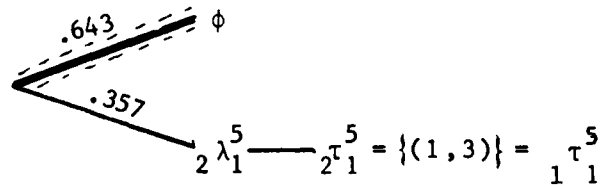
Cluster 2



Cluster 4



Cluster 5



Cluster 6

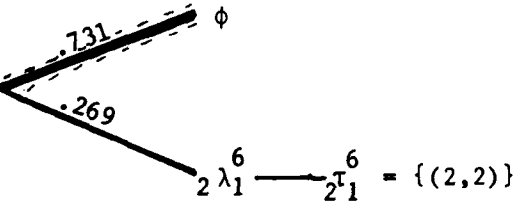


Figure 9-8(b) Hypotheses/Tracks (Scan 2)

measurement in its validation region. Track ${}_0\tau_1^1 = {}_1\tau_1^1 = \{(0,1)\}$ in cluster 1 is initiated at scan 0 and the radial velocity measurement at that time indicates that it should now appear in an unmasked region if it has ever existed as a target. Since there is no measurement which supports this hypothesis, the probability of the one-track hypothesis drops from .753 to .379. Track ${}_1\tau_1^2$ in cluster 2, however, indicates that the hypothesized target is still in the masked region. Therefore, there is no change in cluster 2 as shown in Figure 9-7(b). Track ${}_1\tau_1^5$ in cluster 5 behaves similarly with the one-track hypothesis ${}_1\lambda_1^5$ decreasing slightly in probability from .36 to .357.

Measurement (2,2) does not appear in any of the validation regions of the existing tracks, and hence, a new cluster 6 is created out of it. Since it does appear at an unlikely position for a target (left edge of an unmasked region with a negative radial velocity), the probability that this measurement originates from a newly detected target is only .269.

There are now 16 (global) hypotheses. Two detected targets are correctly incorporated into two tracks in the best hypothesis whose probability is .228. It does not contain any false tracks.

SCAN 3: There are four measurements. Target (1) and target (3) are detected for the first time. Up to this point they have been hiding in zero-detection-probability zones. Clusters 1, 4, 5 and 6 have no measurement which falls into the validation regions of their tracks.

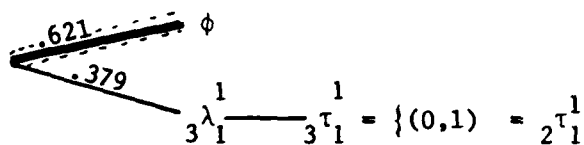
Cluster 4 has a probability-one hypothesis and remains unchanged. Tracks ${}_2\tau_1^1$ and ${}_2\tau_1^6$ in clusters 1 and 6 predict that the hypothesized

targets should be masked and hence these clusters remain unchanged also. However, track ${}_2\tau_1^5$ in cluster 5 predicts that the target should appear in an unmasked region and consequently its probability drops from .357 to .10.

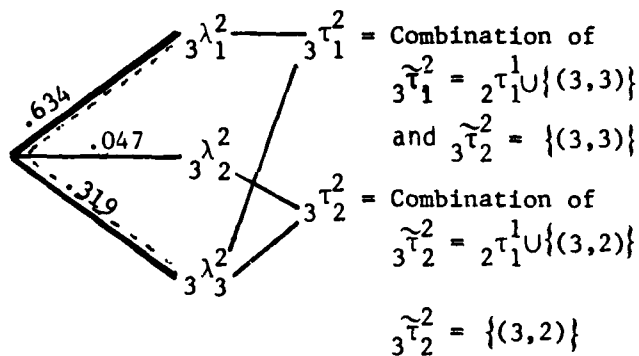
Cluster 2 has two measurements, (3,2) and (3,3), which fall into the validation region of the only track ${}_2\tau_1^1$. (3,2) originates from the newly detected target (1) while (3,3) from the previously detected target (2) which is correctly identified in the best hypothesis in cluster 2. Cluster 2 is expanded into 12 hypotheses as shown in Figure 9-7(b). Out of these twelve, eight hypotheses are pruned away using the 5% pruning threshold. The remaining four hypotheses together have probability 0.95 or more. They are then tested for their combinability. The sixth and the eighth hypotheses have one target in each while the ninth and the eleventh have two tracks in each. Therefore the sixth is compared with the eighth, and the ninth with the eleventh. By comparing the track estimates, the ninth and the eleventh hypotheses are judged to be combinable and are combined. The four tracks in these two hypotheses are combined into two tracks. Finally, the tracks in the sixth and the eighth hypotheses are modified according to this track combining. The resulting hypotheses, ${}_3\lambda_1^2$, ${}_3\lambda_2^2$, and ${}_3\lambda_3^2$ shown in Figure 9-8(c), have probabilities .634, .047 and .319, respectively. The best hypothesis ${}_3\lambda_1^2$ tracks target (2) correctly but fails to recognize the newly detected target (1). The truth is preserved in the second (locally) best hypothesis ${}_3\lambda_3^2$ with probability .319.

Scan 3

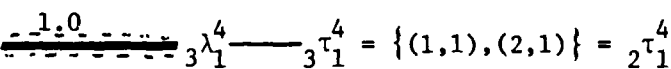
Cluster
1



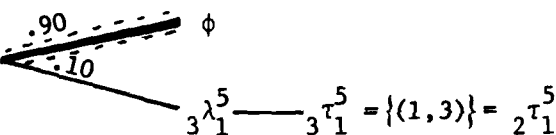
Cluster
2



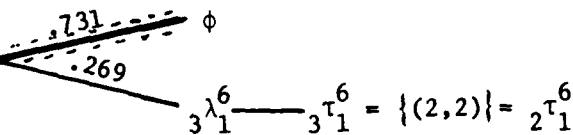
Cluster
4



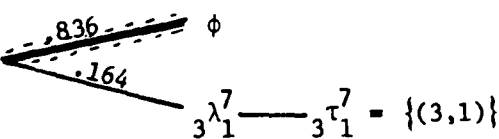
Cluster
5



Cluster
6



Cluster
7



Cluster
8

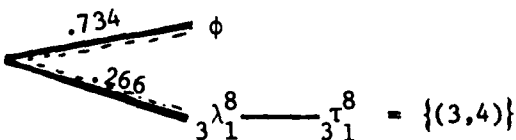


Figure 9-8(c) Hypotheses/Tracks (Scan 3)

Other measurements, (3,1) and (3,4), are used to create two new clusters 7 and 8. Although (3,4) originates actually from target (3), the corresponding hypothesis ${}_3\lambda_1^8$ has only probability .266. The best (global) hypothesis tracks two targets (2) and (4) correctly but misses two targets (1) and (3). It does not contain any false track and has a probability of only .159.

SCAN 4: Target (4) is still masked, but targets (1) and (3), which are all in unmasked regions, are detected. There is one false alarm (4,1) at a position likely to have a target.

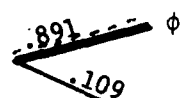
Cluster 2 has two measurements (4,2) and (4,3) which fall into the validation regions of tracks ${}_3\tau_2^2$ and ${}_3\tau_1^2$, respectively. Thus, cluster 2 is expanded as shown in Figure 9-7(b) into 30 hypotheses which are pruned down to only one hypothesis which associates the measurement correctly. This probability-one hypothesis contains two probability-one tracks, ${}_4\tau_1^2$ and ${}_4\tau_1^3$. Thus cluster 2 is now split into two clusters: cluster 3 which consists of track ${}_4\tau_1^3$, and cluster 2 which now has only track ${}_4\tau_1^2$.

Cluster 8 with hypothesis ${}_3\lambda_1^8$ indicating a target with probability .266 receives a measurement (4,4) and is expanded as shown in Figure 9-7(d). As a result of pruning, only one probability-one hypothesis ${}_4\lambda_1^8$ survives, reflecting the truth correctly. Cluster 4 which has one probability-one hypothesis is unchanged because the target is correctly predicted to be in a masked region.

On the other hand, track ${}_3\tau_1^1 = {}_0\tau_1^1$ in cluster 1 is now predicted to appear in an unmasked region but there is no measurement

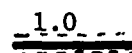
Scan 4

Cluster
1



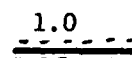
$${}_4\lambda_1^1 - {}_4\tau_1^1 = \{(0,1)\} = {}_3\tau_1^1$$

Cluster
2



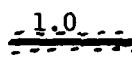
$${}_4\lambda_1^2 - {}_4\tau_1^2 = {}_3\tau_1^2 \cup \{(4,3)\} > {}_3\tau_1^2$$

Cluster
3



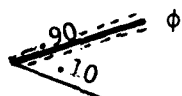
$${}_4\lambda_1^3 - {}_4\tau_1^3 = {}_3\tau_2^2 \cup \{(4,2)\} > {}_3\tau_2^2$$

Cluster
4



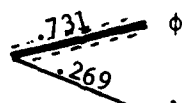
$${}_4\lambda_1^4 - {}_4\tau_1^4 = \{(1,1), (2,1)\} = {}_3\tau_1^3$$

Cluster
5



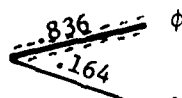
$${}_4\lambda_1^5 - {}_4\tau_1^5 = \{(1,3)\} = {}_3\tau_1^5$$

Cluster
6



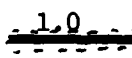
$${}_4\lambda_1^6 - {}_4\tau_1^6 = \{(2,2)\} = {}_3\tau_1^6$$

Cluster
7



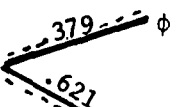
$${}_4\lambda_1^7 - {}_4\tau_1^7 = \{(3,1)\} = {}_3\tau_1^7$$

Cluster
8



$${}_4\lambda_1^8 - {}_4\tau_1^8 = \{(3,4), (4,4)\} > {}_3\tau_1^8$$

Cluster
9



$${}_4\lambda_1^9 - {}_4\tau_1^9 = \{(4,1)\}$$

Figure 9-8(d) Hypotheses/Tracks (Scan 4)

from its predicted position. Thus, the one-track hypothesis $4 \lambda_1^1$ has its probability of .379 reduced to .109. This probability assessment will remain unchanged from this scan on because the track predicts that the target escapes outside the field of view at scan 5 (see Figure 9-7(a)).

Clusters 5, 6 and 7, with their best hypotheses correctly dismissing the measurements as false alarms, remain unchanged due to the masking effect.

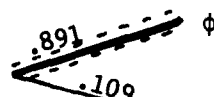
Cluster 9 is newly created from measurement (4,1) (actually a false alarm) which is not validated by any of the existing tracks. Since the position of this measurement indicates a strong possibility of coming from a target, the one-track hypothesis $9 \lambda_1^1$ has probability .621.

Thus, the best (global) hypothesis with probability .304 contains five tracks one of which is a false track. However, there is no missed target.

SCAN 5: There are three measurements. Two of them are from target 2 and target 4. Target 1 is not detected. As seen in Figure 9-6, about this scan the targets cross each other in the one-dimensional physical space on the road. There is no crossing in the two-dimensional state space and the three-dimensional measurement space. Therefore, as seen in Figures 9-7(c) and 9-7(d), there is no confusion which is usually associated with target crossing. As seen in Figure 9-8(e) and 9-9, clusters 5 and 7 disappear, i.e., the no-track hypothesis is now given a probability higher than the pruning threshold in each of these clusters, and a new cluster 10 is formed from a false alarm (5,3).

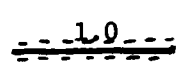
Scan 5

Cluster
1



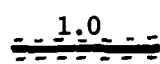
$${}_5\lambda_1^1 \text{ --- } {}_5\tau_1^1 = \{(0,1)\} = {}_4\tau_1^1$$

Cluster
2



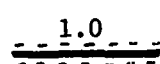
$${}_5\lambda_1^2 \text{ --- } {}_5\tau_1^2 = {}_4\tau_1^2 \cup \{(5,2)\} > {}_4\tau_1^2$$

Cluster
3



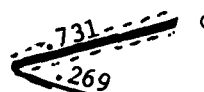
$${}_5\lambda_1^3 \text{ --- } {}_5\tau_1^3 = {}_4\tau_1^3$$

Cluster
4



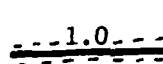
$${}_5\lambda_1^4 \text{ --- } {}_5\tau_1^4 = {}_4\tau_1^4 \cup \{(5,1)\} > {}_4\tau_1^4$$

Cluster
6



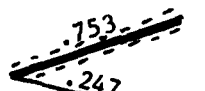
$${}_5\lambda_1^6 \text{ --- } {}_5\tau_1^6 = \{(2,2)\} = {}_3\tau_1^6$$

Cluster
8



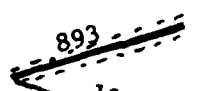
$${}_5\lambda_1^8 \text{ --- } {}_5\tau_1^8 = \{(3,4), (4,4)\} = {}_4\tau_1^8$$

Cluster
9



$${}_5\lambda_1^9 \text{ --- } {}_5\tau_1^9 = \{(4,1)\} = {}_4\tau_1^9$$

Cluster
10



$${}_5\lambda_1^{10} \text{ --- } {}_5\tau_1^{10} = \{(5,3)\}$$

Figure 9-8(e) Hypotheses/Tracks (Scan 5)

Hypothesis $4\lambda_1^9$ in cluster 9, which falsely indicates the existence of a target, now has probability .247 instead of .621. Thus, the best hypothesis with probability .438 now correctly indicates the truth for the first time. There are no false tracks and missed targets.

SCAN 6: Targets (1), (2) and (4) are detected while target (3) is masked and there is no false alarm. As seen in Figures 9-8(f) and 9-9, hypotheses declaring tracks out of false alarms have either disappeared or becoming less probable rapidly. Thus, the best hypothesis has a probability of .756 and indicates the truth correctly.

SCAN 7: Targets (1), (2) and (4) are masked but target (3) now comes out of a masked region and is detected. There is one false alarm (7,1) but the probability of the measurement originating from a target is less than the .05 pruning threshold and thus no new cluster is formed. (7,3) is from a masked region and is ignored. Clusters 6 and 9 have now disappeared. Thus the best hypothesis has probability .891 which is unchanged for the remaining scans. This is so because cluster 1 remains unchanged as we have indicated before at scan 4.

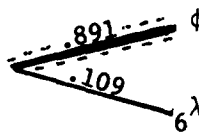
SCAN 8: There is only one measurement which comes from a masked region, and hence there is no change in any cluster.

SCAN 9: All four targets are detected and correctly associated with the four confirmed (probability-one) tracks. One false alarm appears from a masked region and is dismissed.

The measurements (9,1) and (9,2) both fall into the intersection of the validation regions of track $7\tau_1^2$ in cluster 2 and track $7\tau_1^3$ in cluster 3. Therefore, as shown in Figure 9-7(c), these two clusters are merged. Since each of them has only one probability-one hypothesis, the

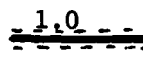
Scan 6

Cluster
1



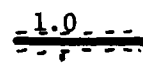
$${}_6\lambda_1^1 - {}_6\tau_1^1 = \{(0,1)\} = {}_5\tau_1^1$$

Cluster
2



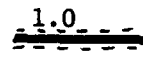
$${}_6\lambda_1^2 - {}_6\tau_1^2 = {}_5\tau_1^2 \cup \{(6,2)\} > {}_5\tau_1^2$$

Cluster
3



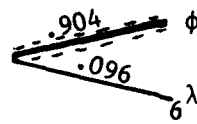
$${}_6\lambda_1^3 - {}_6\tau_1^3 = {}_5\tau_1^3 \cup \{(6,1)\} > {}_5\tau_1^3$$

Cluster
4



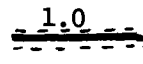
$${}_6\lambda_1^4 - {}_6\tau_1^4 = {}_5\tau_1^4 \cup \{(5,3)\} > {}_5\tau_1^4$$

Cluster
6



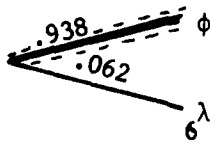
$${}_6\lambda_1^6 - {}_6\tau_1^6 = \{(2,2)\} = {}_5\tau_1^6$$

Cluster
8



$${}_6\lambda_1^8 - {}_6\tau_1^8 = \{(3,4), (4,4)\} = {}_5\tau_1^8$$

Cluster
9



$${}_6\lambda_1^9 - {}_5\tau_1^9 = \{(4,1)\} = {}_5\tau_1^9$$

Figure 9-8(f) Hypotheses/Tracks (Scan 6)

Scan 7

Cluster
1

$$7\lambda_1^1 - 7\tau_1^1 = \{(0,1)\} = 6\tau_1^1$$

Cluster
2

$$7\lambda_1^2 - 7\tau_1^2 = 6\tau_1^2$$

Cluster
3

$$7\lambda_1^3 - 7\tau_1^3 = 6\tau_1^3$$

Cluster
4

$$7\lambda_1^4 - 7\tau_1^4 = 6\tau_1^4$$

Cluster
8

$$7\lambda_1^8 - 7\tau_1^8 = 6\tau_1^8 \cup \{(7,2)\} > 6\tau_1^8$$

Scan 8

Cluster
1

$$8\lambda_1^1 - 8\tau_1^1 = \{(0,1)\} = 7\tau_1^1$$

Cluster
2

$$8\lambda_1^2 - 8\tau_1^2 = 7\tau_1^2$$

Cluster
3

$$8\lambda_1^3 - 8\tau_1^3 = 7\tau_1^3$$

Cluster
4

$$8\lambda_1^4 - 8\tau_1^4 = 7\tau_1^4$$

Cluster
8

$$8\lambda_1^8 - 8\tau_1^8 = 7\tau_1^8$$

Scan 9

Cluster
1

$$9\lambda_1^1 - 9\tau_1^1 = \{(0,1)\} = 8\tau_1^1$$

Cluster
2

$$9\lambda_1^2 - 9\tau_1^2 = 8\tau_1^2 \cup \{(9,2)\} > 8\tau_1^2$$

Cluster
3

$$9\lambda_1^3 - 9\tau_1^3 = 8\tau_1^3 \cup \{(9,1)\} > 8\tau_1^3$$

Cluster
4

$$9\lambda_1^4 - 9\tau_1^4 = 8\tau_1^4 \cup \{(9,5)\} > 8\tau_1^4$$

Cluster
8

$$9\lambda_1^8 - 9\tau_1^8 = 8\tau_1^5 \cup \{(9,3)\} > 8\tau_1^8$$

Figure 9-8(g) Hypotheses/Tracks (Scan 7 - 9)

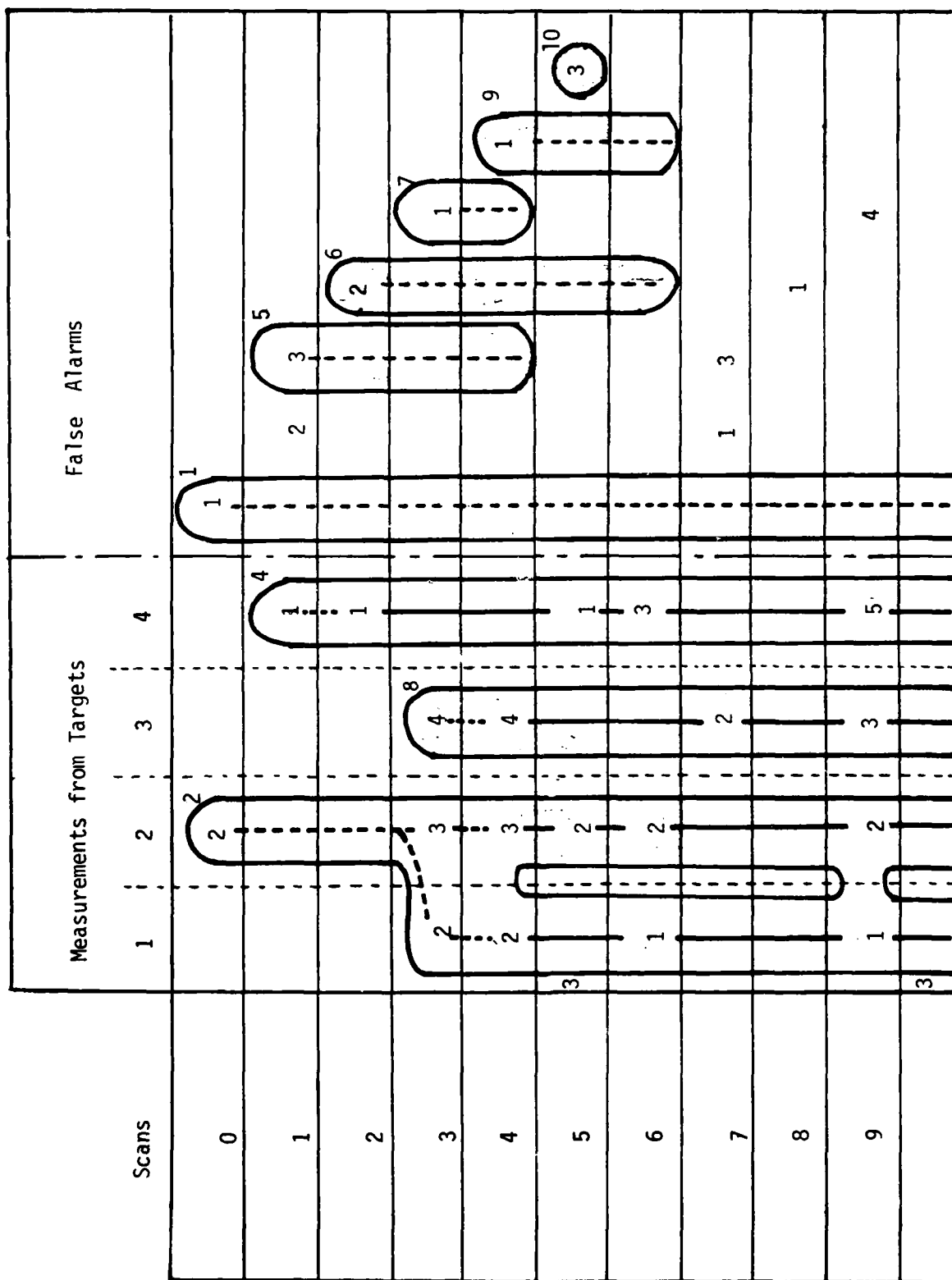


Figure 9-9 Clusters (Labelled by Number)

merged cluster has also one probability-one hypothesis which is $\{ \begin{smallmatrix} \tau^2 \\ 7 \end{smallmatrix}, \begin{smallmatrix} \tau^3 \\ 1 \end{smallmatrix} \}$. This merged cluster is expanded into 14 hypotheses as shown in 9-7(c). As a result of pruning, however, only one hypothesis survives just as in scan 4. Thus, this merged cluster is split into two clusters again.

Figure 9-10 shows how the probability of the best hypothesis changes from scan to scan. After scan 7, there are only two hypotheses. One with probability .891 correctly identifies all the targets without any false track. The other, with probability .109, hypothesizes an additional target which may have been detected at scan 0, undetected at scans 2 and 4, masked at scans 1 and 3 and then escaping out of the field of view.

9.5 Monte Carlo Simulations

The description of the sample run is intended to illustrate some distinctive features of the tracking algorithm developed in this report and how it may be implemented. It is necessary, however, to perform Monte Carlo simulations if one wants to evaluate the performance of the algorithm. This is true in general for nonlinear filtering problems for which there is no good analytic method to predict the performance. It is even more so in this case when hypothesis management techniques are used extensively. The fixed four-target scenario shown in Figure 9-6 is used for the simulation. The parameters introduced earlier are used as a base case. Six other cases are examined to see the effects of the three parameters:

- 1) ν_{FA} - expected number of false alarms per scan
 - 2) σ_r - range measurement error standard deviation,
- and
- 3) ε_p - pruning threshold

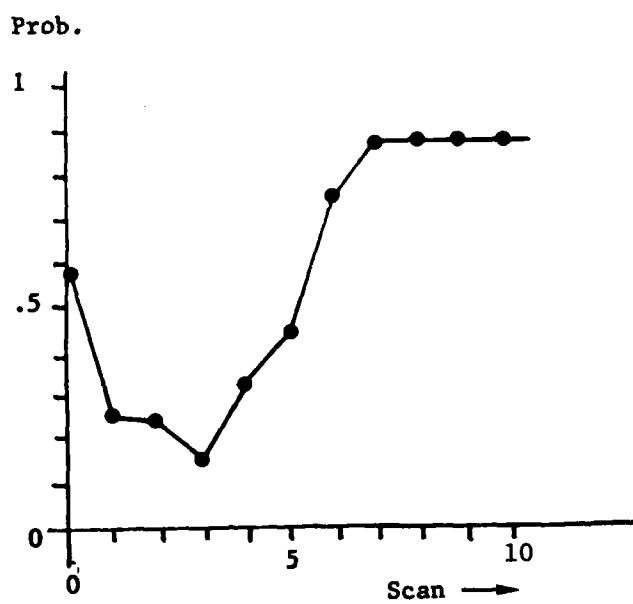


Figure 9-10 A Posteriori Probability
of Best Hypothesis

v_{FA} may be considered to represent the external condition or the noisiness of the radar-preprocessor system, σ_r , a proxy of the measurement accuracy, and ϵ_p , a proxy of the computational resource limitation in implementing the tracking algorithm. Seven cases including the base case are summarized in Table 9-1.

The chosen performance indices are: 1) average number of false tracks; 2) average number of missed targets; and 3) probability of perfect association. For each Monte Carlo run, the best hypothesis after 10 scans is compared with the truth. To do this, the distance between the expected position calculated for each track and each true target position is calculated. Then if the distance is less than a pre-specified threshold, 25 m, the pair is declared feasible. A maximum feasible assignment is sought using a modified (rectangular) zero-one Munkres algorithm. A track which is not associated with any true target is declared as a false track. Similarly, a target which is not accompanied by any feasible track is called a missed target. An event of perfect association is declared if there is no false track and no missed target. For each feasible pair in the assignment found by the Munkres algorithm, the square roots of the mean squared position and velocity errors are calculated. Each case is examined by a 50-run Monte Carlo simulation.

Table 9-2 summarizes the results. The results demonstrate a more or less expected trend, i.e., the worse (the better) the conditions are, the worse (the better) the performance becomes.

Table 9-2(a) shows the effect of false alarms. The performance as measured by the average number of false tracks, average number of missed targets and probability of perfect association seems to respond to the average number of false alarms (1/3, 1 and 3 per scan) in a direct way.

Table 9-1 Cases for Monte Carlo Simulation

		ν_{FA}	σ_r	ϵ_p
Case 0	Base Case	1.	3.m	.05
Case 1	Low ν_{FA}	1/3	3.m	.05
Case 2	High ν_{FA}	3.	3.m	.05
Case 3	Small σ_r	1.	1.m	.05
Case 4	Large σ_r	1.	9.m	.05
Case 5	Low ϵ_p	1.	3.m	.01
Case 6	High ϵ_p	1.	3.m	.1

ν_{FA} Expected Number of False Alarms Per Scan

σ_r Range Measurement Error Standard Deviation

ϵ_p Pruning Threshold

Table 9-2 Monte Carlo Simulation Results

(a) Variation of False Alarm Rates:

Expected Number of False Alarms per Scan (v_{FA})		ANFT	ANMT	PofPA	APE	AVE
Low:	1/3	.42	.26	.46	5.50 m	.259 m/s
Base Case:	1.	.48	.58	.34	5.63 m	.276 m/s
High:	3.	.66	1.02	.14	5.33 m	.258 m/s

(b) Variation of Range Measurement Error:

Range Measurement Error Standard Deviation (σ_r)		ANFT	ANMT	PofPA	APE	AVE
Small:	1. m	.46	.58	.34	4.26 m	.274 m/s
Base Case:	3. m	.48	.58	.34	5.63 m	.276 m/s
Large:	9. m	.56	.68	.30	9.33 m	.279 m/s

(c) Variation of Pruning Threshold

Pruning Threshold (ϵ_p)		ANFT	ANMT	PofPA	APE	AVE
Low:	0.01	.46	.16	.52	5.72 m	.275 m/s
Base Case:	0.05	.48	.58	.34	5.63 m	.276 m/s
High:	0.10	.46	.90	.16	5.71 m	.277 m/s

ANFT Average Number of False Tracks per Run
ANMT Average Number of Missed Targets per Run
PofPA Probability of Perfect Association (no false tracks and no missed targets)
APE Average Position Error (conditioned on correct association)
AVE Average Velocity Error (conditioned on correct association)

The mechanism of generating false tracks and missed targets can be explained as follows. Suppose that a false alarm measurement (0,1) shown in Figure 9-6 has a negative radial velocity measurement and its position measurement is slightly more to the left than the one shown in the figure. A cluster will be formed out of this measurement and the sole one-track hypothesis in this cluster very likely is given a probability greater than .5. This becomes a source of a false track no matter what is done afterwards unless this cluster is completely ignored. Can one discard such clusters? The answer is no, because the calculated probability is consistent with that of the event of a target generating one measurement and disappearing into the zero-detection-probability zones. Thus, the number of false alarms is directly related to the number of false tracks.

On the other hand, as we have seen before, in order for a sequence of measurements originating from a target to become a track with enough probability to appear in the best hypothesis, two or three scans are needed. Meanwhile, at each scan, a large number of hypotheses may be created if the number of false alarms is large. Then, as a result of pruning, a target may be lost from the best hypothesis. Therefore, it may be conjectured that, when the sensor is noisy and produces a large number of false alarms, more scans are needed before a conclusion is made and a larger number of hypotheses need to be preserved. The latter statement is actually supported by our simulation runs using a lower threshold. The effect of false alarms on the state estimate errors is not significant. This is not surprising because those errors are calculated only for the feasible pairs.

The effects of range measurement error as seen in Table 9-2(b) are less drastic. There is virtually no improvement in the basic performance indices,

measured by the average number of false tracks, average number of missed targets and the probability of perfect association when the range measurement error is smaller than that in the base case. Also, there is no drastic performance degradation when the error is larger. However, some degradation is noticed as expected, since increasing errors mean more confusion, more hypotheses, and more branch cutting, etc. We also note that the effect is more pronounced on the position estimation error while the velocity estimation error is not affected at all.

Table 9-2(c) shows the effect of the pruning threshold. The average number of missed targets is directly affected by this parameter, and a drastic change can be observed in the table. The average number of false tracks, however, is not affected. This indicates that the false-track-creating mechanism described previously is dominant and cannot be improved by keeping more hypotheses. As in the case of varying the false alarms, state estimation error is not affected by changing the pruning threshold.

It should be noted that the other performance indices ANFT (average number of false tracks), ANMT (average number of missed targets) and PofPA (probability of perfect association) all have to do with how well the data association has been performed. Of these indices, PofPA is the most stringent since perfect association (given a certain evaluation threshold) is needed. For a difficult scenario such as the one used in the example, this is hard to come by. A more reasonable set of figures to look at are ANFT and ANMT which measure the deviation from the ideal performance. From Table 9-2, we see that the GTC indeed performs quite well in most cases, with ANFT and ANMT less than 6% and 15% of the actual number of targets in the base case.

10. CONCLUSION

A general algorithm for the tracking and classification of multiple targets has been derived using the Bayesian approach. The models underlying the algorithms rely on a fairly general set of assumptions. In most of the existing work on multitarget tracking, there is a certain vagueness in the mathematical development, especially when it comes to modeling the number of targets. In some cases a certain birth-death process is implied about the model, but this fact has not been used explicitly in the development. In contrast, our basic approach to target modeling is to start with an explicit model on the number of targets. Although we have restricted our attention to the case when the number of targets is random but constant in time, a time-varying number of targets can be easily handled within our basic framework either by introducing an variable which reflects the active or inactive state of each individual target or by modifying the basic equations directly. Another distinguishing feature of our approach is the explicit modeling of the mechanisms which generate the sensor measurements, namely, detection, random assignment, false alarm generation and measurement value generation.

The result is an algorithm which is considerably more general than what has been reported in the literature. It not only can handle tracking and classification simultaneously in one unifying framework but can also take care of unconventional measurement situations such as when terrain masking is present. When restricted to i.i.d. target models, which account for most of the existing results, this algorithm provides a unifying view of comparing all the common multitarget tracking algorithms. However, we have not exploited the full potential of this approach. With some additional work, we believe this approach can handle more complicated target models, such as targets moving in a group, etc.

This work constitutes the theoretical foundation for developing estimation-based processing algorithms in a DSN. It provides the necessary algorithms to be implemented at each node for processing the local sensor data before communicating with other nodes. By looking at a distributed version of this algorithm, we can also obtain algorithms for the fusion of information from other nodes.

Implementation issues are always crucial in any tracking algorithm, especially one which relies on the formation of multiple hypotheses. In this particular case, successful implementation depends on good hypothesis management. Although we have made some progress in classifying and understanding the techniques, basic theoretical results are still lacking. It is believed that advances will probably be made through a lot of experimentation and by using techniques for tree searches from artificial intelligence.

REFERENCES

1. R. Lacoss, "Overview of the Distributed Sensor Networks Program at Lincoln Laboratory," Proc. Distributed Sensor Networks Workshop, M.I.T. Lincoln Lab. Jan 6-7, 1982.
2. R.P. Hughes and R.R. Tenney, "A Multiobject Tracking Algorithm for a Distributed Sensor Network," Proc. 3rd MIT/ONR Workshop on Distributed Information and Decision Systems, Sept. 1980.
3. R.M. Tong, E. Tse and R.P. Wishner, "Distributed Hypothesis Formation in Sensor Fusion Systems," Proc. 20th IEEE Conf. on Decision and Control, San Diego, CA, Dec. 1981.
4. R.W. Sittler, "An Optimal Data Association Problem in Surveillance Theory," IEEE Trans. Mil. Electron., vol. MIL-8, pp. 125-139, April 1964.
5. R.G. Sea, "An Efficient Suboptimal Decision Procedure for Associating Sensor Data with Stored Tracks in Real-time Surveillance Systems," Proc. 1971 IEEE Conf. on Decision and Control, Miami Beach, FL, Dec. 1971.
6. R.A. Singer and J.J. Stein, "An Optimal Tracking Filter for Processing Sensor Data of Imprecisely Determined Origin in Surveillance Systems," Proc. 1971 IEEE Conf. Decision and Control, Miami Beach, FL, Dec. 1971.
7. A.G. Jaffer and Y. Bar-Shalom, "On Optimal Tracking in Multiple-target Environments," Proc. 3rd Symp. Nonlinear Estimation Theory and Its Applications, Univ. Calif., San Diego, Sept. 1972.
8. R.A. Singer and R.G. Sea, "New Results in Optimizing Surveillance System Tracking and Data Correlation Performance in Dense Multitarget Environments," IEEE Trans. Automat. Contr., vol. AC-18, pp. 571-581, Dec. 1973.
9. R.A. Singer, R.G. Sea and K. Housewright, "Deviation and Evaluation of Improved Tracking Filters for Use in Dense Multitarget Environments," IEEE Trans. Inform. Theory, vol. IT-20, pp. 423-432, July 1974.
10. Y. Bar-Shalom and E. Tse, "Tracking in a Cluttered Environment with Probabilistic Data Association," Automatica, vol. 11, pp. 451-460, Sept. 1975.
11. Y. Bar-Shalom, "Extension of the Probabilistic Data Association Filter to Multitarget Environment," Proc. 5th Symp. Nonlinear Estimation, Univ. Calif, San Diego, Sept. 1974.
12. P. Smith and G. Buechler, "A Branching Algorithm for Discriminating and Tracking Multiple Objects," IEEE Trans. Automat. Contr., vol. AC-20, pp. 101-104, Feb. 1975.

13. D.L. Alspach, "A Gaussian Sum Approach to the Multitarget Identification-tracking Problem," Automatica, vol. 11, pp. 285-295, May 1975.
14. J.J. Stein and S.S. Blackman, "Generalized Correlation of Multitarget Track Data," IEEE Trans. Aerosp. Electron. Syst., vol AES-11, pp. 1207-1217, Nov. 1975.
15. H.L. Hurd, "A Bayesian Approach to Simultaneous Tracking of Multiple Targets," Proc. ONR Passive Tracking Conf., Monterey, CA, May 1977.
16. C.L. Morefield, "Application of 0-1 Integer Programming to Multitarget Tracking Problems," IEEE Trans. Automat. Contr., vol AC-22, pp. 302-312, June 1977.
17. K.M. Keverian, "Multi-object Tracking by Adaptive Hypothesis Testing," S.B. Thesis, M.I.T. Cambridge, MA., May 1979.
18. K.M. Keverian and N. R. Sandell, Jr., "Multi-object Tracking by Adaptive Hypothesis Testing," Report LIDS-R-959, MIT, Dec. 1979.
19. I.R. Goodman, "A General Model for the Multiple Target Correlation and Tracking Problem," Proc. 18th IEEE Conf. Decision and Contr., Fort Lauderdale, FL, Dec. 1979.
20. D.B. Reid, "An Algorithm for Tracking Multiple Targets," IEEE Trans. Automat. Contr., vol AC-24, No. 6, pp. 843-854, Dec. 1979.
21. T.E. Fortmann, Y. Bar-Shalom and M. Scheffe, "Multitarget Tracking Using Joint Probabilistic Data Association," Proc. 1980 IEEE Conf. on Decision and Control, Albuquerque, NM, Dec. 1980.
22. Y. Bar-Shalom, "Tracking Methods in a Multitarget Environment," IEEE Trans. Automat. Contr., vol. AC-23, pp. 618-626, Aug. 1978.
23. H.L. Wiener, W.W. Willman, I.R. Goodman and J.H. Kullback, "Naval Ocean-Surveillance Correlation Handbook, 1978," NRL Report 8340, Naval Research Lab., Washington, D.C., 1979.
24. I.R. Goodman, H.L. Wiener and W.W. Willman "Naval Ocean Surveillance Correlation Handbook, 1979," NRL Report 8402, Naval Research Lab., Washington, D.C., 1980.
25. K.R. Pattipati, N.R. Sandell, Jr. and L.C. Kramer, "A Unified View of Multi-object Tracking," Proc. 4th MIT/ONR Workshop on Distributed Information and Decision Systems Motivated by Command-Control-Communications Problems, vol. 1, pp. 115-135, San Diego, CA, June 1981.
26. C.Y. Chong, S. Mori, E. Tse and R.P. Wishner, "Distributed Estimation in Distributed Sensor Networks," Proc. 1982 American Control Conference, Arlington, VA, June 14-16, 1982.

27. F.E. Bruneau, "State Estimation of a Hybrid Markov Process with Application to Multitarget Tracking," Report LIDS-TH-1172, MIT, Jan. 1982.
28. H.W. Sorenson, and D.L. Alspach, "Recursive Bayesian Estimation using Gaussian Sums," Automatica, vol. 7, no. 4, pp. 465-479, July 1971.