PB2001-103973

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. ATC-298	2. Government Accession No.	3. R PB2001-103973

4. Title and Subtitle ASR-9 Processor Augmentation Card (9-PAC) Phase II Scan-Scan Correlator Algorithms	5. Report Date 26 April 2001 6. Performing Organization Code
7. Author(s) Robert D. Grappel	8. Performing Organization Report No. ATC-298
9. Performing Organization Name and Address MIT Lincoln Laboratory 244 Wood Street Lexington, MA 02420-9108	10. Work Unit No. (TRAIS) 11. Contract or Grant No. F19628-00-C-0002
12. Sponsoring Agency Name and Address Department of Transportation Federal Aviation Administration Systems Research and Development Service Washington, DC 20591	13. Type of Report and Period Covered Project Report 14. Sponsoring Agency Code

15. Supplementary Notes

This report is based on studies performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology, under Air Force Contract F19628-00-C-0002.

16. Abstract

This report documents the scan-scan correlator (tracker) algorithm developed for Phase II of the ASR-9 Processor Augmentation Card (9-PAC) project. The improved correlation and tracking algorithms in 9-PAC Phase II decrease the incidence of false-alarm tracks and increase the detection of real aircraft. The tracker processing for 9-PAC Phase II defined in this document builds upon the prototype 9-PAC Phase II tracker described in ATC-245. Tracker algorithms from Mode S (ATC-65) are also used in Phase II.

This document describes the three main processing tasks of the tracker: initialization, input/output, and the actual correlation/tracking. The tracker itself is further broken down into four main functions: report-to-track association, report-to-track correlation, track update, and track initiation. Each of these functions is described in detail and is further broken down into sub-functions. In addition to the algorithm descriptions, the 9-PAC Phase II tracker system requirements are reviewed, and main data structures used in the 9-PAC Phase II tracker are defined.

17. Key Words	18. Distribution Statement This document is available to the public through the National Technical Information Service, Springfield, VA 22161.

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 131	22. Price

FORM DOT F 1700.7 (8-72) Reproduction of completed page authorized

EXECUTIVE SUMMARY

This report documents the Scan-Scan correlator Phase II algorithms developed for the ASR-9 Processor Augmentation Card (9-PAC) project. The 9-PAC is a processor card that serves as a processing enhancement to the existing ASR-9's post-processor system. It provides increased speed and memory capabilities to the processor, which allows for the introduction of more complex scan-scan correlator algorithms. These more complex algorithms improve the ASR-9's system performance through decreased false alarms, and increased detection of aircraft.

The 9-PAC Scan-Scan Phase II correlator, also known as the Tracker, consists of three basic processing tasks: initialization, input/output, and the actual Tracker. The Tracker can be broken down further into four main processing functions: report-to-track association, report-to-track correlation, track update, and track initiation. These four 9-PAC Tracker functions are the same as in the original ASR-9 processor, but with different algorithms. Each of these functions is addressed individually in this report, and is further broken down into sub-functions for more detailed discussion.

This report is one in a series of reports that document the algorithms implemented in the 9-PAC. "The Beacon Target Detector (BTD) Algorithms Deployed in the ASR-9 Processor Augmentation Card (9-PAC)" [1] and "The Radar Correlation and Interpolation (C&I) Algorithms Deployed in the ASR-9 Processor Augmentation Card (9-PAC)" [2] are two other reports in the series. "ASR-9 Processor Augmentation Card Scan-Scan Correlator Algorithms" [7] defines an earlier prototype version of the Phase II Tracker design.

ACKNOWLEDGMENTS

The author thanks the people on the 9-PAC team who helped in this effort: Jenifer Evans who designed, coded, and documented the original Phase II prototype Tracker, Jeff Gertz who designed the Mode S and TSF algorithms that were borrowed for the Phase II Tracker, and Gabe Elkin and Claranne Bechtler for their great assistance in taking the tracker to real-time. Finally, thanks to Bill Goodchild, whose tireless testing and relentless prodding made the Phase II Tracker a far better program.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS
(CONTINUED)

LIST OF TABLES

1. INTRODUCTION

Air traffic surveillance at major airports depends on two types of radar: primary radar surveillance and secondary surveillance radar. Primary radar surveillance is the more traditional radar surveillance; a signal is transmitted by the radar, reflected off an object, and received by the radar. The aircraft do not need any special equipment to be seen by the radar. Unfortunately, neither do the birds, cars, raindrops, and countless other objects. The most frequently used primary radar system at the major U.S. airports today is the ASR-9 (Airport Surveillance Radar).

In addition to the primary radar system, there is the secondary surveillance radar system, also known as the beacon system. Secondary surveillance radar differs from primary radar surveillance: a signal (interrogation) is transmitted by the radar, the signal is received by a transponder on an aircraft, an answering signal (reply) is transmitted at a different frequency by the transponder, and the answering signal is received by the radar. This surveillance system requires the intended targets be properly equipped with transponders to be detected by the system. Two different techniques are used to measure azimuth: sliding windows in older systems and monopulse in newer systems. The ASR-9, when operating with a Mode S sensor, employs both azimuth-determination techniques – sliding window in the Interim Beacon Interrogator (IBI) backup mode, and monopulse in the Mode S mode.

The two surveillance systems produce two types of reports, typically referred to as radar and beacon reports from the primary and secondary surveillance systems, respectively. Ideally, all aircraft will have transponders and will be seen by both systems. In reality, not all aircraft have transponders, and not all transponders perform perfectly at all times. Additionally, not all aircraft can be detected with the primary system at all times, especially in high clutter regions. Hence the need for both types of surveillance. The ASR-9 is the first terminal radar system to provide both types of surveillance in a digital format to the end user.

The ASR-9 is an advanced radar system, providing significant improvements in aircraft detection in bad weather. However, as the ASR-9 has been deployed around the country, site-specific, environmentally-induced performance problems have been discovered. These problems include false beacon targets due to reflections and processing splits; radar false targets due to weather breakthrough, ground traffic, and other clutter sources; false radar tracks due to false radar targets; and missed radar tracks due to poor track initiation. These problems can all be addressed with improved, more complex processing algorithms. However, the ASR-9's post-processor, the Array Surveillance Processor (ASP), does not have spare processing capacity, and given its architecture and machine language, it is also difficult to modify and support. As a result the ASR-9 Processor Augmentation Card (9-PAC) was developed.

The 9-PAC is a processor card that replaces an ASP memory board and serves as a processing enhancement to the existing ASR-9's post processor system. With the Phase II software installed, the ASP processor boards are completely removed from the radar. The 9-PAC card provides significantly increased speed and memory capabilities that make it possible to introduce more complex algorithms for handling the aforementioned performance problems. The 9-PAC software can be divided into four basic processes: Beacon Target Detector (BTD), Correlation & Interpolation (C&I), Merge, and Scan-Scan Correlator. In addition there is the Operating System which holds it all together. This report focuses solely on the 9-PAC Phase II Scan-Scan Correlator, also known as the Tracker.

This report documents the software modifications and algorithms implemented in the 9-PAC Phase II Scan-Scan Correlator or Tracker. The 9-PAC Phase II Tracker differs from the earlier prototype Phase II Tracker (described in Reference [7]) in a number of areas, including:

1. More sophisticated radar-only track initiation algorithms reduce the incidence of false-tracks and increase Tracker performance for marginal aircraft
2. More sophisticated beacon track initiation algorithm (from Reference [7])
3. Radar tracking now in range-azimuth coordinates with extended "acceleration" smoothing provides more accurate tracking close to the sensor – better able to track maneuvers
4. Bad correlation "forgiveness-fixup" algorithm repairs the effects of erroneous radar-only correlations on track
5. More accurate track update period calculation provides more accurate tracking close to sensor
6. Calculation of track association box sizes for maneuvering aircraft improved (from Reference [7])
7. Added discrete Mode 3/A beacon association logic (from Reference [7]) to track rapidly maneuvering, beacon-equipped aircraft
8. More elaborate association scoring algorithm for conflict resolution makes better choices and handles mixed (radar-only, beacon-only, radar-reinforced beacon) cases more correctly
9. More elaborate radar-only association linearity test deals with erroneous data points
10. Feedback of uncorrelated radar-only reports to C&I helps to maintain proper clutter map
11. Special-case 2-on-2 processing for conflict resolution performance speedup
12. Extended performance analysis outputs

Section 2 of this report addresses the basics of why the Tracker is necessary and how it is used. Section 3 outlines the Tracker system requirements. The various data structures implemented in the software are addressed in Section 4, and finally, Section 5, provides descriptions of the actual algorithms.

System performance improvements provided by the new Tracker algorithms will be addressed in a separate report. The purpose of this report is to document the tracker algorithms in enough detail to support implementation by a second party. This report is one in a series of reports documenting the 9-PAC algorithms [1],[2].

2. OVERVIEW

First, a word about terminology. In the ASR-9 the Tracker function is, in reality, a scan-scan correlator. The distinction is that a true tracker generally maintains a track identification number, predicts at least a few scans ahead, and/or smoothes the input data. A scan-scan correlator determines which report most likely belongs to a single track, and then outputs that report. It does not append a track identification number, it does not do prediction beyond the time of the next antenna scan, and it does not smooth the data. A scan-scan correlator is essentially a false alarm filter. The ASR-9's current post-processor uses a scan-scan correlator, although it is often referred to as a tracker. Staying with ASR-9 terminology, this report will also refer to the Scan-Scan Correlator portion of the post-processor as the Tracker.

When the ASR-9 is operating with a Beacon Interrogator (BI), e.g., the BI-5, or with the Mode S as an Interim Beacon Interrogator (IBI), the data flow is as shown in Figure 1. Beacon replies are grouped together to form beacon targets in the Beacon Target Detector (BTD). Radar primitives are grouped together to form radar targets in the Correlation & Interpolation (C&I) process. These two streams of data are fed to the Merge process which determines which beacon targets and radar targets correspond to the same aircraft. These targets are merged together and called radar-reinforced targets. The radar-reinforced targets and the leftover beacon-only and radar-only targets are the output of Merge. They are simultaneously passed to the end user and to the Tracker. The Tracker determines which leftover radar-only reports most likely belong to real aircraft, then passes these reports to the end user as correlated radar reports. These correlated radar-only reports may actually correspond to radar-only, non-transponder equipped aircraft, or they may correspond to beacon, transponder-equipped aircraft, which did not have an interpretable beacon signal on that scan. All correlated and uncorrelated radar reports are sent back to the C&I process to aid in the development of clutter maps.

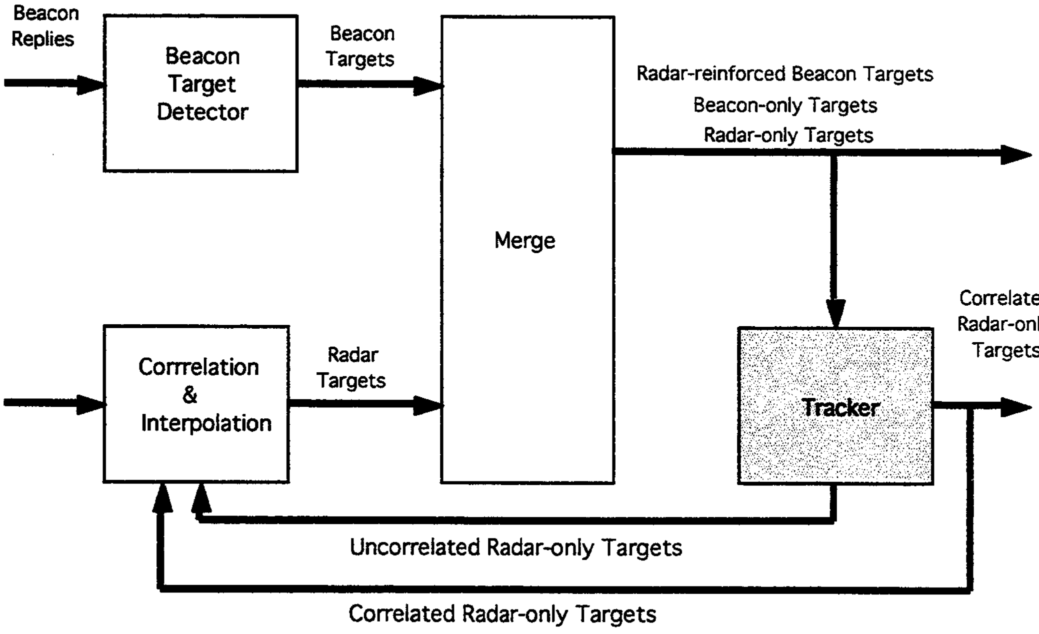


Figure 1. ASR-9 data flow.

When the ASR-9 is operating with a Mode S sensor, the data flow changes slightly. The modified data flow is depicted in Figure 2. The Mode S is responsible for the processing and forming of beacon targets. It receives the radar-only targets formed by the ASR-9's C&I. The Mode S merges these two types of data and outputs radar-reinforced targets, leftover beacon-only targets, and leftover radar-only targets to the ASR-9. The ASR-9 immediately sends the target data to the end user and to the Tracker. The Tracker behaves identically for a BI system and a Mode S system; it acts as a filter for radar-only targets and outputs correlated radar targets.

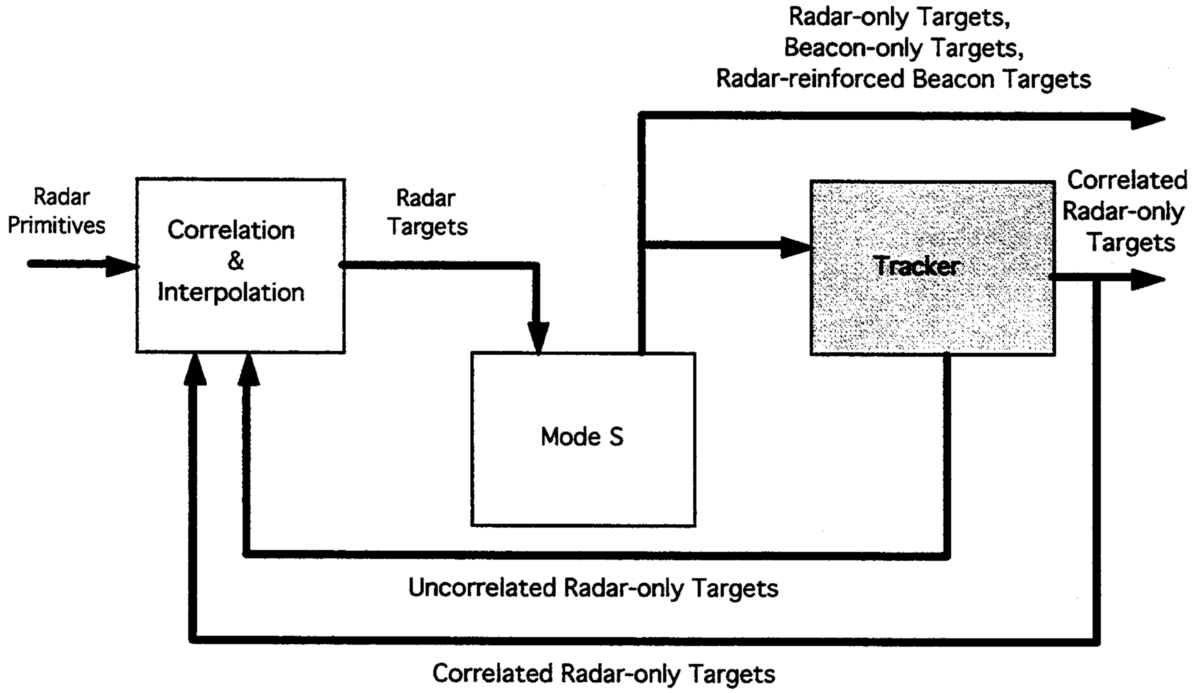


Figure 2. ASR-9 data flow when operating with Mode S.

There are a few interesting points about the data flow and system processing for both configurations. While the Tracker receives radar-reinforced targets and beacon-only targets in addition to the radar-only targets as inputs, it only outputs correlated, radar-only reports. The radar-reinforced targets and beacon-only targets are used to maintain internal tracks. This allows for substitution of a radar-only report in case of a missing beacon report.

Due to timing concerns, the ASR-9 outputs the radar-reinforced targets, the beacon-only targets, and the leftover radar-only targets as soon as possible to the end user. Simultaneously, these same data are input to the Tracker. The Tracker filters the data and subsequently outputs the correlated, radar-only data. These output data are a subset of the leftover, radar-only data that was part of the input to the Tracker and that was previously sent to the end user. Essentially, all of the output of the Tracker is a subset of previously output data, but it has a time delay due to the processing time of the Tracker and a bit set in the header indicating the report is part of a correlated track. Because of the Tracker time delay, the correlated, radar-only data is used solely for display purposes in the ATC system.

3. REQUIREMENTS

The 9-PAC Phase II Tracker is designed to meet the same specifications as the ASR-9 Tracker while addressing some performance concerns. These concerns are basically high false alarm rates, particularly in high clutter regions, and an inability to track quickly maneuvering targets. The following are system inputs, outputs, and requirements as defined in the ASR-9 specification FAA-E-2704B [3] and reiterated in the Software System/Subsystem Specification Surveillance Processor for the ASR-9 [4]. The 9-PAC Tracker must satisfy these requirements.

1. Inputs to Surveillance Processor Tracker:
    - Beacon target reports
    - Radar target reports
    - Radar-reinforced target reports
    - Azimuth word
    - Indication of transmitter failure
    - Variable Site Parameters
2. Outputs from Surveillance Processor Tracker:
    - Radar correlated target reports
    - Uncorrelated radar target reports
    - Performance monitor data
    - Tracker overflow alarms
3. Capacity requirement for Surveillance Processor Tracker (Table 1):

Table 1. Capacity Requirement for Tracker

Capacity Requirement	Number of Targets	FAA-E-2704 Paragraph No.
Peak Aircraft Targets	700	3.4.3.2
Peak Non-aircraft Targets	300	3.4.3.2
Peak targets in 90° scan	250	3.4.3.2
Peak targets across two contiguous 11.25° sectors	100	3.4.3.2
Peak 1.3° azimuth wedge targets	16	3.4.3.2
Minimum of 32 surveillance-processing steps per scan.		3.12.5.6.1

4. Target overload conditions shall be handled in an orderly manner; e.g., reduced processing range ([3], Paragraph No. 3.4.3.2).
5. The surveillance processor (tracker) shall output fewer than 1.0 false scan correlated radar target reports per scan averaged over a one hour period, during normal operating conditions. The peak rate of false scan correlated radar target reports

shall be fewer than ten per scan averaged over a one hour period, under extreme conditions of "angel" activity or ducting ([3], 3.12.5).

6. The Scan-Scan Correlator shall process the data from Mode S when operating with a Mode S ([3], 3.12.5).
7. The maximum delay of the scan-scan correlated radar reports to ATC display shall not exceed 2.1 seconds as compared to antenna boresight ([3], 3.12.5).
8. The Scan-Scan Correlator functions shall not drop tracks from the surveillance track list when the POWER-DOWN-INDICATOR = TRUE for less than 15 seconds ([3], 3.3.1).
9. The Scan-Scan Correlator shall not initiate a track by using a target report flagged as an MTI target ([3], 3.12.3.4.12).
10. The Scan-Scan Correlator shall only associate a "Real Time Quality Control" (RTQC) target report with a track that was initiated by an RTQC report during a previous scan ([3], 3.13.3.1.1).

4. DATA STRUCTURES

The data structures used in the 9-PAC Phase II Tracker are relatively simple. They are described here to help with the understanding of the following discussion of the 9-PAC algorithms. There are basically two unique data types used in the 9-PAC Phase II Tracker, reports and tracks, which are usually kept in linked lists. The specific fields that make up the two data types are listed in Appendix A with descriptions. Note that there is plenty of memory in the 9-PAC so it is not necessary to pack data into these data types; there are numerous fields that only serve as a single bit flag. (The 9-PAC processor treats all data items as 32-bit words, so there is no benefit to declaring data in smaller sizes than a processor word.)

4.1 REPORT DATA TYPE

For simplicity, the 9-PAC Phase II Tracker has only a single report data type. Each report contains a field to differentiate between radar-reinforced, beacon-only, and radar-only reports, but the overall structure is the same. Fields which are meaningless for some report types, e.g., altitude for a radar-only report, are generally set to null and ignored. The report data type has fields for the typical report elements such as range, azimuth, altitude, code, etc., which are generated by BTD and C&I. In addition, it has a number of fields which are used for associating and correlating the report to tracks. These fields count the number of associations, have pointers to associating tracks, and set flags for correlation. A number of fields in the report provide mechanisms for algorithm performance analysis. Some fields in the report are used to feedback information on correlated reports from the tracker to the C&I task. In addition, there are a few report fields which are used to reduce the overall system processing load by limiting the need to do a given calculation more than once, e.g., coordinate conversions.

4.2 TRACK DATA TYPE

The 9-PAC Phase II Tracker has only a single track data type, which has many more fields than the report data type. There are fields containing the positions of the last three reports used to update the track in addition to the predicted position for the next scan. There are also miss counters to determine how many scans occurred between each of the last three reports. There are numerous fields for the actual smoothing and predicting of the track position and velocity (maintained in both Cartesian and range-azimuth coordinate systems). There are fields for maintaining the association boxes and association and correlation information. Each track contains fields to store radar information (quality, confidence, etc.) as well as beacon information (Mode 3/A code and confidence, Mode C altitude and confidence, etc.). There are numerous fields which contain information that can be used for clutter rejection, e.g., velocity, minimum distance. A number of fields in the track provide mechanisms for algorithm performance analysis. Finally, there are fields for statistics gathering, e.g., age count and history count of the track.

4.3 REPORT AND TRACK LINKED LISTS

Both 9-PAC Phase II Tracker reports and tracks are maintained as linked list data structures. The reports are maintained in a single link-list in the order in which they are received. Due to the nature of the 9-PAC processing, the reports are received in clusters, and adjacent reports are rarely more than one sector (128 ACPs) apart. Since the order of the reports is roughly correlated with their azimuths, the processing time required to create an azimuth ordered report list would outweigh the processing benefits achieved by doing so.

Tracks are maintained in two separate link-lists: a general list and an active list. A given track entry is linked to only one list at a time. The general track list is an azimuth ordered list (by predicted azimuth) for essentially all tracks. The active track list is for tracks which are ready to be updated. When the antenna passes the predicted azimuth position for a track, the track is moved from the general track list to the active track list. After being updated by a report or coasted, the track is returned to the general list. By maintaining two separate lists, update processing is simplified and time is saved by reducing the number of tracks which need to be cross-checked against reports for possible associations, and which need to be monitored for possible correlation processing.

4.4 TRACK SORT LISTS

Each track file entry that is currently defined in the tracker is maintained on a singly-linked sort-list based on its Cartesian x-y location in the radar coverage region (-65 to +65 nautical miles centered on the radar). The sort-list mechanism is used to speed up the search processing for matching reports against "nearby" tracks. Each x-y sort bin occupies a 5 nautical mile square.

4.5 UNCORRELATED REPORT LISTS

The tracker track-initiation algorithms search lists of uncorrelated target reports from prior scans to find candidate tracks for initiation processing. In the case of beacon targets, the most-recent two scans-worth of uncorrelated beacon target reports are maintained in the beacon lists. For radar targets, the most-recent five scans-worth of uncorrelated radar reports are maintained in the radar lists. Each scan's lists (both radar and beacon) are further subdivided into range bins and azimuth sectors in order to speed the process of searching by geographic position within a given scan and report type. The range bins are 5 nautical miles in extent (0-65 nautical miles in range), and each scan is divided into 32 azimuth sectors (128 ACPs per sector).

4.6 DOPPLER TABLE

In order to speed up the calculations for the Doppler Agreement Test (see section 5.3.2.5), the tracker uses a pre-calculated square table indexed by the high and low-PRF Doppler differences found between the expected and measured Doppler values. The table contains the Doppler test pass/fail value given the Doppler differences.

4.7 SINE/COSINE TABLES

To improve processing performance, the tracker pre-calculates tables of the sine and cosine function for all azimuths measured in ACPs (0-4095). Table lookup (by ACP value) is much faster than calculating the trigonometric value each time it is needed.

4.8 BIT-DIFFERENCE TABLE

To improve processing performance, the tracker pre-calculates the number of set bits in the values 0-4095 (12 bit values). The table lookup is used in the comparison of 12-bit Mode 3/A codes and Mode C altitudes in the beacon algorithms.

4.9 GREY-CODE TABLE

To improve the tracker's processing performance in comparing beacon Mode C altitudes, a table of altitude Grey codes is pre-calculated. Table lookup is used whenever a Mode C altitude (in flight levels) must be converted to Grey code form.

4.10 LINEARITY-TEST AZIMUTH EXTENT TABLE

The azimuth threshold for the linearity tests (see sections 5.3.1.1 and 5.3.3.1) is a somewhat complex function of range. To improve tracker performance, the threshold values are pre-calculated and stored in a table indexed by range in 0.25 nautical mile increments. The small table granularity is required because the azimuth threshold changes rapidly for short ranges.

4.11 CROSS-RANGE TRACK INITIATION TABLE

The track initiation algorithms need to search for uncorrelated reports (radar or beacon) from a previous scan that could be part of a candidate track. The distance over which this search is to be performed is a function of the maximum reasonable speed assumed for an aircraft (nominally 600 knots). The azimuth threshold is a somewhat complex function of range. To improve tracker performance, the azimuth threshold is pre-calculated and stored in a table indexed by range in nautical mile increments.

4.12 TRACK HEADING HISTOGRAM TABLE

The tracker maintains a count of the number of radar-only, slow-moving (i.e. speed ≤ SLOW_VEL VSP) tracks currently maintained in the system as a function of the track's smoothed ground heading. The histogram table divides the heading range into 15-degree sectors. There are actually two sets of heading histogram tables in the tracker: one set is used to accumulate the statistics for the current scan, while the second set stores the statistics from the previous scan. The previous-scan's statistics are used in the "bird detection" tracker algorithms. Once per scan (at the northmark), the current scan histogram set is copied into the previous scan histogram and the current scan histogram set is re-initialized.

5. PROCESSING

This section discusses the 9-PAC Phase II Tracker data processing. This discussion is broken down into three general parts: the initialization process, the input/output process, and the actual Tracker process. These processes fit together as shown in Figure 3 below.

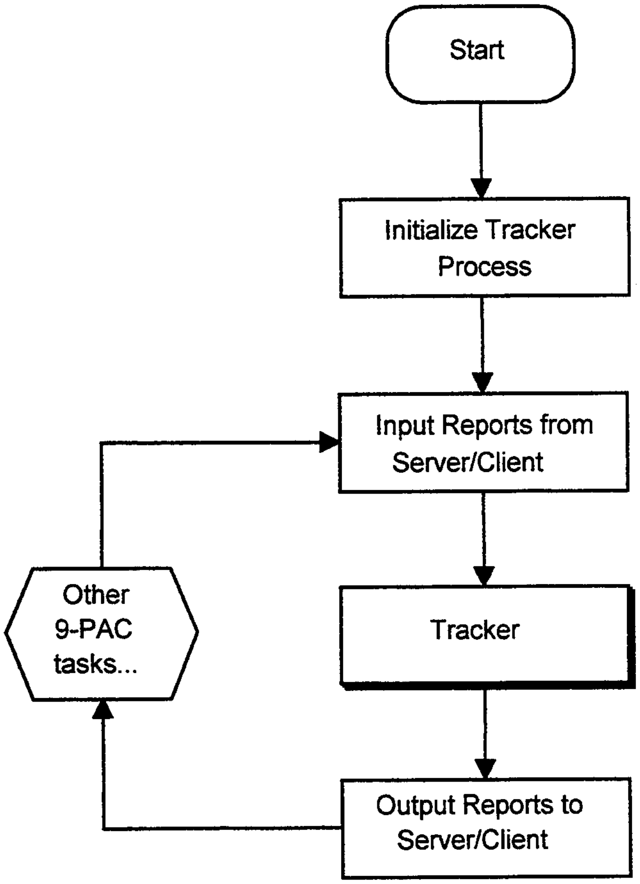


Figure 3. Overview of 9-PAC Tracker processes.

5.1 INITIALIZATION PROCESS

When the 9-PAC system is started, each of the main processes needs to be initialized. The Phase II Tracker initialization process includes allocating stack space for the maximum number of reports and maximum number of tracks. While the maximum number of tracks should never exceed 1100 (system requirement for 700 beacon targets, 300 primary radar targets, and 100 false alarms per scan), there is more than sufficient memory in the 9-PAC, so the maximum number of tracks has been set to 2000. The maximum number of reports is also oversized and limited to 1000 total reports of all types.

After allocating memory for tracks and reports, the data structures that will hold these are initialized. The 9-PAC Phase II Tracker uses three linked lists: reports, active tracks, and general tracks. Tracks move back and forth between the active list and the general list, but can only be on one list at any given time. The active tracks list contains tracks that have predictions in the vicinity of the radar antenna, and are waiting to be updated or coasted. The general tracks list contains all of the other tracks. The initialization process also performs one time functions such as generation of sine and cosine tables and other such tables employed to reduce future processing. Finally, the Variable Site Parameters (VSPs) are read for the first time, and the Performance Monitors are reset. (See Appendix B for a listing and definition of 9-PAC Phase II Tracker VSPs, and Appendix C for a listing of the 9-PAC Phase II Tracker Performance Monitor entries.)

5.2 INPUT / OUTPUT PROCESS

The 9-PAC Phase II Tracker process receives and sends data via server/client channels. The Tracker task is a client on receive from the server Merge task during stand-alone ASR-9 operations. When the 9-PAC is operating with a Mode S, the Tracker task is a client on receive from the server input task. In either mode, the 9-PAC Phase II Tracker task receives radar-only, beacon-only, and radar-reinforced reports from the serving task. The reports are mixed together in a single input stream.

Whenever the 9-PAC Phase II Tracker task is called, the first function implemented is receiving reports from the server/client channel. This occurs approximately once every 16 antenna ACPs (4096 ACPs = 360°). On receipt of the reports, the 9-PAC Phase II Tracker process copies the reports from the received format into the previously defined report data structure. Many fields are copied directly to the corresponding field in the Tracker data structure. One exception is the altitude field, which is converted from the 12-bit Gray code to an altitude in feet, if possible. In addition to copying the received report fields, a number of other report fields are initialized, e.g., Cartesian position coordinates and various flags and counters.

While the processing of incoming reports is called every time the 9-PAC Phase II Tracker task is called, the remainder of the Tracker task is only called once every 64 ACPs. System requirements (delineated in Table 1) require calling the surveillance steps at least 32 times a scan, which is once every sector, or once every 128 ACPs. While it is desirable to increase the output rate by calling the Tracker task more often, doing so increases the processing load. Sixty-four ACPs is a good compromise: calling the task even more often does not significantly improve the output rate of the tracker, or significantly change the output, but it does significantly increase the processing load. Maintenance of the radar-only and beacon-only uncorrelated report lists (for track initiation) is done on a 32 azimuth-sectors/scan basis (every 128 ACPs).

Several 9-PAC Phase II Tracker functions are performed on a once-per-scan basis at the antenna north-mark. They are performed only once each scan to increase system performance. These functions include updating the Performance Monitor entries and checking various "state" variables for VSP changes.

Each time the complete 9-PAC Phase II Tracker task is called, the Tracker task acts as a server to the 9-PAC Output task client. The 9-PAC Tracker task buffers all correlated radar-only reports corresponding to stable tracks, and at the end of the task, sends them to the 9-PAC Output task. Uncorrelated radar-only reports that do not initiate new 9-PAC tracks are output as a feedback from the Tracker to the 9-PAC C&I. (Note: this feedback is used by the C&I to indicate which reports the tracker ignored as clutter. This assists C&I in properly maintaining its clutter maps. The correlated radar reports are also fed back to the C&I, which uses them in maintaining its CFAR maps.) The 9-PAC Phase II Tracker output occurs approximately once every 64 ACPs. In addition, Performance Monitors which are updated each time the complete Tracker task is called are output in the same manner to the Output task client, but only once per scan.

5.3 TRACKER PROCESS

The discussion of the specific 9-PAC Phase II Tracker task algorithms follows and is accompanied by flowcharts depicting the main Tracker functions. The functions listed by the flowcharts and referred to in this discussion do not necessarily correspond to actual C program functions or subroutines. They are basic tracking functions necessary to achieve the desired tracker performance.

The 9-PAC Phase II Tracker architecture is similar to the original ASR-9 Tracker architecture, the earlier 9-PAC prototype Phase II Tracker [7], and most other tracker architectures. The Tracker cycles through four basic processes: report-track association (determining which reports are candidates for updating a track), report-track correlation (determining which of the associating reports is the best report for a track), track update (given the correlated report, predicting where the track will be on the next scan), and track initiation (determining which left-over reports could be new tracks). This basic flow is depicted in Figure 4.

The four basic Tracker functions, associate, correlate, update, and initiate, and their sub-functions are listed below in their respective hierarchy. Each function is accompanied by a flowchart. Some sub-functions have additional flowcharts with further detail; these sub-functions are indicated by a shadowed box in the flowchart. Not surprisingly, some sub-functions are called from more than one function. These sub-functions are only detailed and charted the first time they are called. These are the functions and sub-functions charted in the Sections 5.3.1 through 5.3.4.

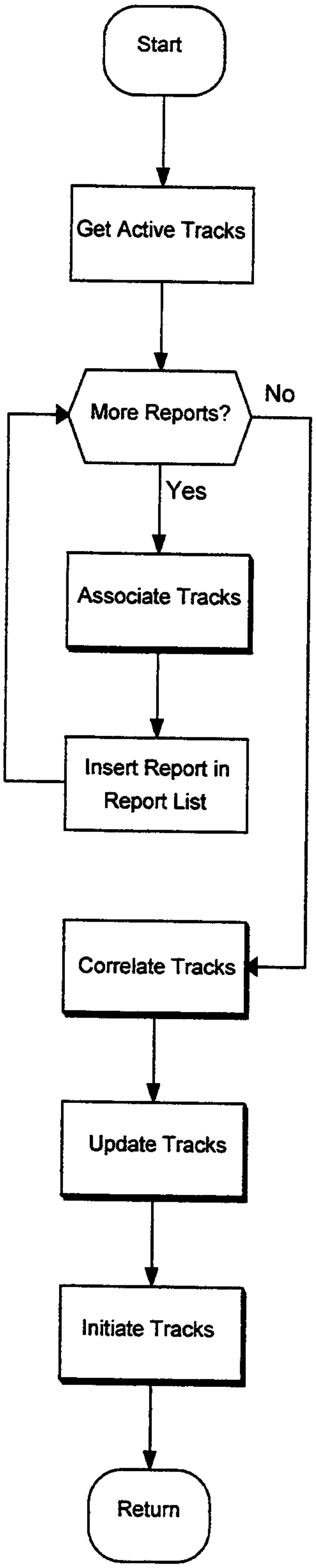


Figure 4. Basic Tracker algorithm flow.

9-PAC Tracker

1. Associate Tracks
--- Linearity Test

2. Correlate Tracks
--- Possibly Correlate
------ Beacon Split
------ Compute Score
--------- Doppler Agreement Test
------ Conflict Resolution
--------- Compute Score
------------ Doppler Agreement Test
--------- Build Association Matrix
------------ Compute Score
--------------- Doppler Agreement Test
--------- Modified Munkres Algorithm
------------ Mark Pair
--------- Mark Pair
--- Ready to Correlate
------ Beacon Split
------ Conflict Resolution
--------- Compute Score
------------ Doppler Agreement Test
--------- Build Association Matrix
------------ Compute Score
--------------- Doppler Agreement Test
--------- Modified Munkres Algorithm
--------- Mark Pair
------ Mark Pair

3. Update Tracks
--- Track Coast
------ Compute Association Boxes
--- Track Update
------ Determine Gains
------ Update Predictions
------ Compute Velocity
------ Set Track Type
------ Check Minimum Distance
------ Update Track State
------ Compute Association Boxes

4. Initiate Tracks
    --- Beacon Initiation
    --- Radar Initiation
    --- Compute Association Boxes

5.3.1 Associate Tracks

Called by:	Tracker
Calls:	Linearity Test
Purpose:	The Associate Tracks process determines which reports are candidates for updating which tracks. If a report is a candidate for updating a specific track, the pair are considered associated.

The 9-PAC report-track association process determines which reports are possible candidates for updating which tracks. If a report and a track are considered a possible match, the pair are said to be associated. The best report for a given track is subsequently chosen from the associated reports for correlation. As reports are received from the server/client channel, they are compared to each track on the active track list, and all associations are marked. (The processes for report-to-track association and discrete beacon association are shown in the flowcharts in Figures 5 and 6.)

The total number of associations for a single track or a single report is limited to six. In a typical environment, more than 90% of the reports and tracks only have one association. The limit of six is rarely hit, and even then it is due to a system anomaly and a severe clutter breakthrough. If the number of associations limit is hit, no more associations are allowed and it is assumed that with the severe clutter, the chances of a greatly superior association is small and not worth the heavy processing load that would be encumbered.

The main criteria for report-track association in the 9-PAC is location. Two association zones are defined for most tracks: zone 0 represents the predicted track position plus room for system errors; zone 1 encompasses zone 0 plus adds room for a maneuvering track. A report can associate with a track via zone 0 or zone 1. In addition to the location criteria defined by zone 0 and zone 1, there are additional criteria for report-track association. These are discussed below.

If the report is a radar-only target report, the candidate report-track match must meet each of the following conditions in order to associate:

(1) The report's "maintenance score" (derived from its radar quality, confidence, and range via a table of VSPs) must be greater than 0. Reports whose maintenance score is 0 are quite likely to be clutter and will not be allowed to associate.

(2) The number of consecutive scans of missed radar or radar-reinforced beacon correlations for the candidate track must be less than the VSP value NSC_ASSOC.

(3) If the candidate track is radar-only, the report's RTQC ("Real Time Quality Control" test target) state must match the track's RTQC state. This condition forces RTQC reports to associate only with RTQC tracks.

(4) If the track has not seen a positional "fixup" in the last 2 scans (see the description of track update for radar (Section 5.3.3.1) for the "fixup" algorithm), then the association linearity test described in Section 5.3.1.1 must be passed. The purpose of the linearity test is to assure the reports contributing to the track are moving in a manner reasonable for an aircraft.

If the report is either beacon-only or radar-reinforced beacon and does not have a discrete Mode 3/A beacon code, the candidate report-track match must meet each of the following conditions in order to associate:

The report must lie within zone 1 of the track.

(1) If the report is beacon-only, then the number of consecutive scans of missed beacon or radar-reinforced beacon correlations for the candidate track must be less than the VSP value NSC_ASSOC.

(2) If the candidate track is beacon-only or radar-reinforced beacon, the report's RTQC ("Real Time Quality Control" test target) state must match the track's RTQC state. This condition forces RTQC reports to associate only with RTQC tracks.

If the report is either beacon-only or radar-reinforced beacon and has a discrete Mode 3/A code, then additional special association testing is performed. (These tests are derived from Reference [7].) Discrete Mode 3/A codes are issued by ATC to individual aircraft in coverage, so it is unlikely that a report with such a discrete code is not the proper match for a track with a matching discrete code. The special processing here allows the 9-PAC Phase II tracker to follow more violent maneuvers that could cause the report position to fall outside of the track's zone 1 (sized for modest maneuvers). The following conditions must be met for discrete code associations:

The candidate track must be beacon-only or radar-reinforced beacon type

(1) The discrete Mode 3/A code of the candidate track must exactly match the discrete Mode 3/A code of the report.

(2) The difference in range between the report and the candidate track's predicted position must be less than or equal to 2 nautical miles. This is a consistency-check for cases where aircraft codes are assigned in error, reflections, etc.

(3) The difference in azimuth between the report and the candidate track's predicted position must be less than or equal to 4 degrees. This is a consistency-check for cases where aircraft codes are assigned in error, reflections, etc.

(4) The altitude of the report and the candidate track must pass the Altitude-Zone Scoring test described below. Altitude zone scores 0 and 1 pass, zone scores greater than 1 fail. This is a consistency-check for cases where aircraft codes are assigned in error, reflections, etc.

If any of the discrete Mode 3/A tests described above fail, then the tests for non-discrete beacon associations are applied as described previously. If all the criteria for association are met (radar-only, non-discrete beacon, or discrete beacon), the association is marked: counters in the report and track structures are incremented, and pointers are set from the report to the track and vice versa. In addition, if the report and track have matching discrete codes, a special flag is set indicating a discrete association.

The zone scoring for Mode C altitudes is based on the altitude type of the beacon report and the track. A "high validity" altitude is defined to have a validity value ≥ 3. A "low validity" altitude has a validity value < 3. Altitude types are derived from the report or track's Mode C altitude and validity values and are defined as follows:

0 - no Mode C
1 - low-validity brackets-only
2 - high-validity brackets-only
3 - low-validity flight level
4 - high-validity flight level

The zone score for Mode C altitudes is defined in the table below as a function of the altitude types for the report and track. Where a number appears in the table, this indicates the altitude zone score value. Where a capital letter appears in the table, this indicates a processing function described below the table.

	Report Altitude Type				
Track Altitude Type	0	1	2	3	4
0	0	1	1	1	1
1	1	0	0	2	2
2	1	0	0	2	3
3	1	2	2	**A**	**A**
4	1	2	3	**A**	**B**

Test **A** converts the Mode C altitudes back to 12-bit Gray codes and masks out the 3 octal 'C' bits. The number of bit differences between the masked Gray codes is computed. If the number of bit differences is ≤1 then the altitude zone score is 1. If the number of bit differences is ≤ 3 then the altitude zone score is 2. If the number of bit differences is > 3, the altitude zone score is set to 3.

Test **B** computes the absolute value of the difference between the two Mode C altitude levels. If this difference is ≤5 (500 feet), the altitude zone score is set to 0. Otherwise, continue as in test **A** above.

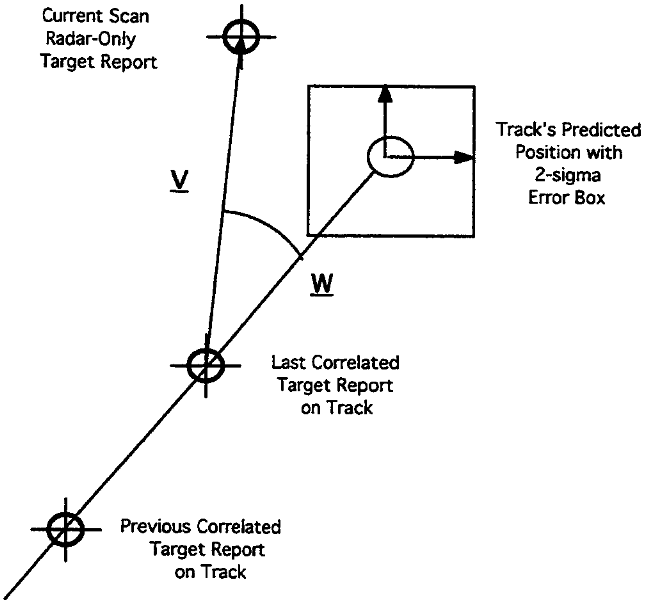
VELOCITY REASONABLENESS TEST

An additional association qualification test is currently under consideration for incorporation into the 9-PAC Phase II tracker. The velocity reasonableness test (derived from [6]) is intended to eliminate certain radar-only target-to-track associations based on their geometry. Even though the current scan radar-only report lies within the track's outer association box and passes the association linearity test (section 5.3.1.1), it is still possible that it would require an "unreasonable" velocity change for an aircraft represented by the track to have reached the target report position on the current scan. These are the association cases that the velocity reasonableness test seeks to eliminate.

The velocity reasonableness test cannot be applied when the radar-only track and target are very close to the sensor. This is due to the lack of altitude information making it too inaccurate to assume that the measured slant ranges are equivalent to the actual ground ranges. Also, the assumption that the radial and tangential components of velocity are nearly perpendicular does not hold for short ranges. Hence, if the track's predicted range for this scan is inside two nautical miles, then the velocity reasonableness test is assumed to pass.

The velocity reasonableness test does not eliminate any association that lies within the track's inner association box. Since the inner box is sized for expected measurement error for a straight track, all points within it are equally likely for an aircraft flying straight. No location within the inner box can be eliminated as "unreasonable."

The velocity reasonableness test defines several vectors, as shown in the figure below. Vector **V** goes from the last known track position (i.e., last correlated report) to the proposed radar-only associating target report position on the current scan. Vector **W** goes from the last known track position (i.e., last correlated report) to the predicted track position on the current scan. Vector **E** is the sensor's 2-sigma measurement error vector (depicted as a rectangular box around the predicted track position in the figure). All these vectors have a radial and a tangential component. The tangential component is converted into distance units so that it is commensurate with the radial component. The scalar "f" is defined to be the track's firmness – i.e., the number of scans between the last correlation to the track and the current scan.



The velocity reasonableness test is composed of two components: a magnitude test and an angle test. Both component tests must pass in order for the velocity to be declared reasonable. The magnitude of a vector gives its scalar length and is denoted "|| ||". The following equation defines the magnitude of a vector in terms of its radial and tangential components.

$$\| \underline{\mathbf{V}} \| = \text{square-root}(V_r^2 + V_t^2)$$

The magnitude component of the velocity reasonableness test is passed if the change in track speed required to get from the track to the report position is less than or equal to a parameter value (nominally 1.25 – a speed change of up to 25% is deemed reasonable). The following equation defines the magnitude test:

$$\| \underline{\mathbf{V}} \| \,/\, \| \underline{\mathbf{W}} + \underline{\mathbf{E}} \| \geq \text{param}$$

The angle component of the velocity reasonableness test seeks to compute the angle between the $\underline{\mathbf{V}}$ and $\underline{\mathbf{W}}$ vectors (i.e., a measure of the change in direction required for the track to turn to the report position). In order to simplify the math, the angle test actually computes the cosine of the angle rather than the angle itself. The following equation defines the cosine of the angle between the two vectors:

$$\text{ß} = (\underline{\mathbf{V}} \bullet \underline{\mathbf{W}}) \,/\, (\| \underline{\mathbf{V}} \| * \| \underline{\mathbf{W}} \|)$$

where the dot product of two vectors given in terms of their radial and tangential components is defined as:

$$(\underline{\mathbf{V}} \bullet \underline{\mathbf{W}}) = (V_r * W_r) + (V_t * W_t)$$

The angle test is passed if the angle between the vectors is sufficiently small (which implies that the angle cosine ß is near one), i.e.,

$$\text{ß} \geq -(f - 1) * \text{param}$$

where the "parameter" is chosen to be the reciprocal of the number of consecutive track coasts allowed before a track is dropped (i.e., 1/3). This formula allows up to a 90 degree angle between the vectors for a consistent track. (For no coasts ($f = 1$), the equation reduces to $\text{ß} \geq 0$.) A larger angle is permitted if the track is coasting ($f > 1$). This test prevents a "double back" condition from associating.

The angle test is not attempted, however, in those cases where measurement uncertainty in either the radial or tangential direction is greater than the track's velocity in that coordinate. In either case, the track's heading could be in error by more than 90 degrees, thus invalidating the angle test. Hence, if either

(1) Track's radial velocity (in per-scan units) < E_r, or
(2) Track's tangential velocity (in per-scan units) < E_t

then the angle component of the velocity reasonableness test is assumed to pass by default.

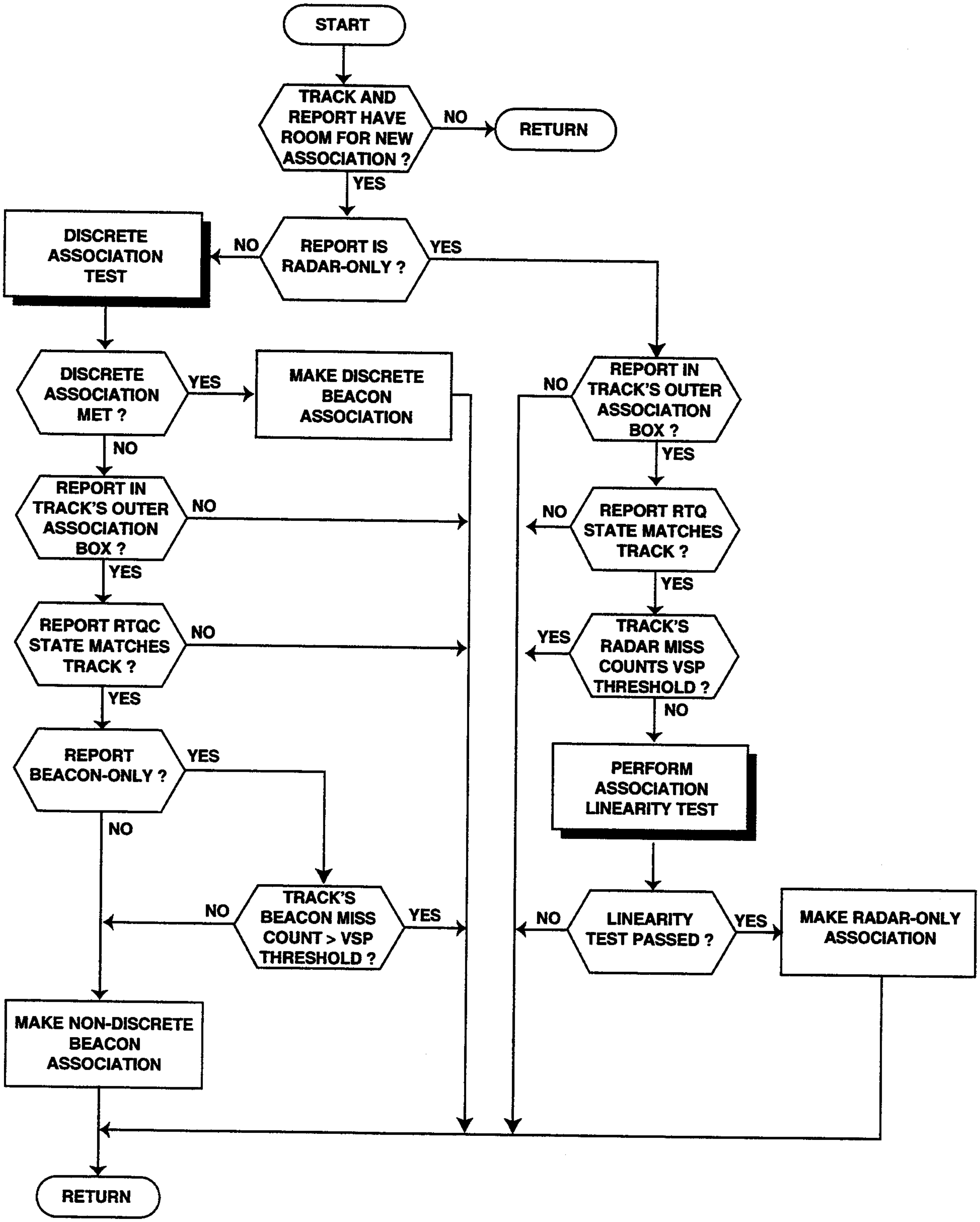


Figure 5. Report-to-Track Association.

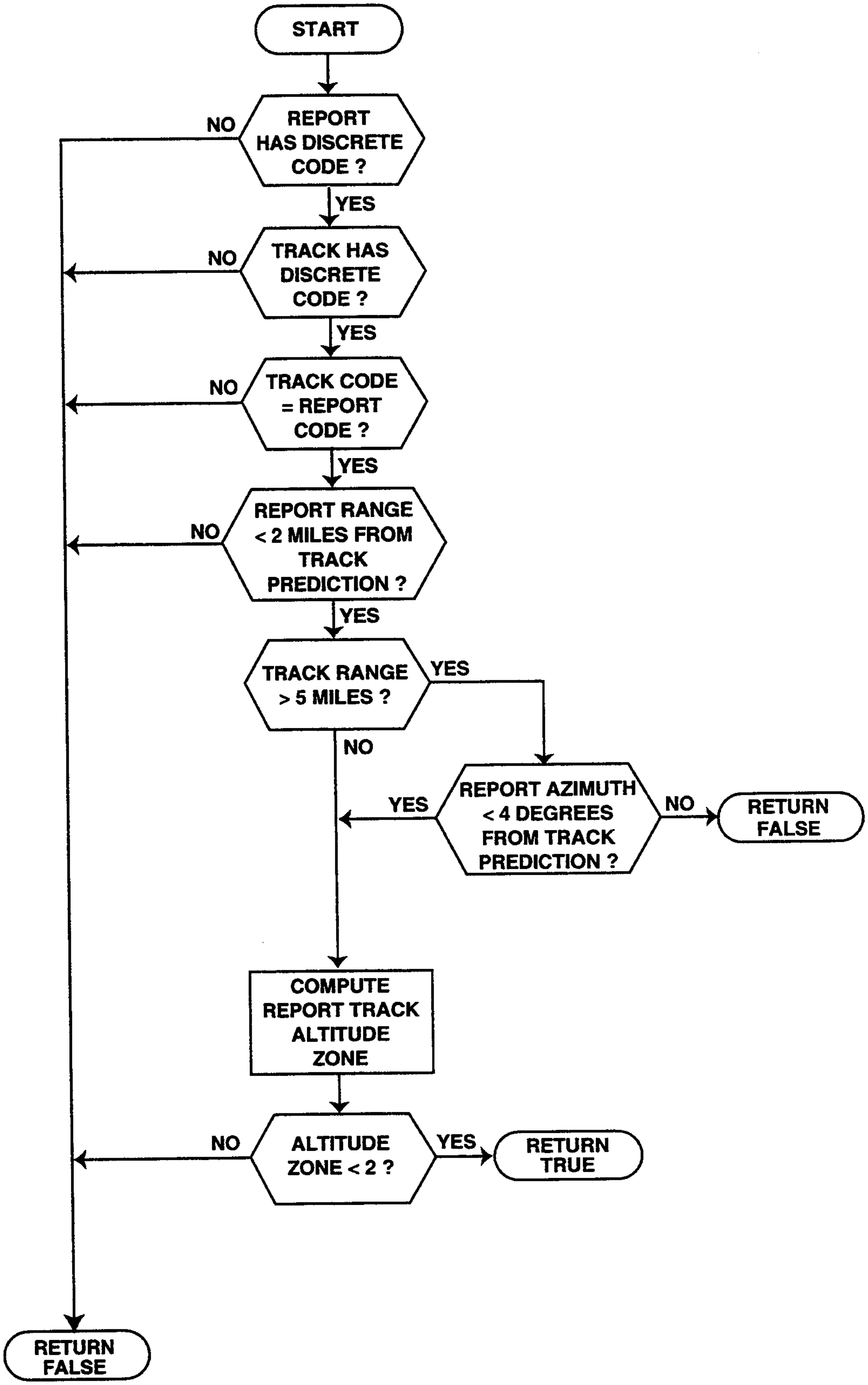


Figure 6. Discrete Beacon Association.

5.3.1.1 Linearity

Called by:	Associate Tracks
Calls:	
Purpose:	The linearity test determines whether the reports contributing to a track lie in a path consistent with a real aircraft.

The linearity test uses the track history and computes the deviation of a middle report in a series of reports on a given track from the expected position. The deviation from the expected position is compared to the expected system errors. In general, if the measured deviation differs by more than a predetermined number of standard deviations, the report and track fail the linearity test. A more detailed description of the test follows.

The track under consideration has been updated the last three scans by reports received at times *t-3*, *t-2*, and *t-1*. The range and azimuth of these reports has been saved along with the number of misses between these updates in the track. The report under consideration has been received on this scan at time *t*. The linearity test uses the three tracked reports, *t-3*, *t-2*, *t-1*, and the candidate report *t*, along with the miss history to determine if the candidate report *t*, really belongs with this track.

The association linearity test is only applied to the potential association of a radar-only target report with a radar-only track. (Beacon and radar-reinforced beacon associations can use Mode 3/A code and Mode C altitude comparisons to avoid false correlation decisions, while radar-only associations must depend primarily on positional tests, such as this one.) The basic design of the association linearity test involves the computation of the deviation in position (range and azimuth) of a middle report in a time-ordered sequence of reports from the expected position. The linearity test passes if the deviation is within a parametric number of sensor measurement error standard deviations (determined from the sensor's measurement error sigma VSPs and the linearity-test multiplier VSPs in range and azimuth coordinates).

The first step in the linearity test computes a straight line between the position of the potential association report on this scan *t* and the second previously-correlated report from the track *t-2*. The range and azimuth of the point along the line corresponding to the time of the scan of the most-recently-correlated report from the track *t-1* is linearly interpolated. If the absolute difference in range and azimuth between the interpolated point on the line at the time of the scan *t-1* and the actual position of the correlated report on the track at scan *t-1* is less than or equal to the parametric sensor measurement error tolerance, the association linearity test passes. If the differences do not both meet the measurement tolerances and the track does not have three previously-correlated reports (e.g., the report just initiated as an early initiation using only 2 reports), the linearity test fails. Otherwise, one of two additional tests is performed. Which additional test is used depends on whether the first test failed in the range coordinate. (Range measurements are assumed to be more accurate than azimuth measurements.)

If the first step in the association linearity test succeeded in the range direction (but not in the azimuth direction) and the track has had at least three previous-scan correlated reports, a second linearity testing step is performed (to account for a possibly erroneous correlation to the track on a prior scan). The second linearity test step computes a straight line between the position of the potential association report on this scan *t* and the third previously-correlated report from the track *t-3*. The range and azimuth of the point along the line corresponding to the time of the scan of the second most-recently-correlated report from the track *t-2* is linearly interpolated. If the absolute difference in range and azimuth between the computed point on the line at the time of the scan *t-2* and the actual position of the correlated report on the track at scan *t-2* is less than or equal to the parametric sensor measurement error tolerance, the association linearity test passes – otherwise, the association linearity test fails.

If the first step in the association linearity test failed in the range direction and the track has had at least three previous-scan correlated reports, a second linearity testing step is performed (to account for a possibly bad azimuth in one of the reports). The second linearity test step computes a straight line between the position of the most-recent correlated report on the track on this scan *t-1* and the third previously-correlated report from the track *t-3*. The range and azimuth of the point along the line corresponding to the time of the scan of the second-most-recently-correlated report from the track *t-2* is linearly interpolated. If the absolute difference in range and azimuth between the computed point on the line at the time of the scan *t-1* and the actual position of the correlated report on the track at scan *t-2* is less than or equal to one half of the parametric sensor measurement error tolerance, the association linearity test passes – otherwise, the association linearity test fails.

Note that a special-case linearity test is employed for radar reports with ranges less than or equal to 1 nautical mile. Problems with the geometry of range-azimuth tracking (radar reports lack altitudes and, hence, cannot be accurately converted to x-y coordinates) and aircraft accelerating during takeoff from runways near the radar make it difficult to perform the linearity tests correctly for such short range targets. If the radar report association (range ≤ 1 nautical mile) is within the inner association box of the track (sized for expected measurement errors), then the linearity test result is defaulted to "pass." This prevents problems with the linearity testing from disallowing proper radar-only associations for "close in" aircraft. It is assumed that any clutter reports that are associated by this special test will be sorted out in the correlation process.

An additional special-case is applied if the input radar-only report has confidence=4. The range and azimuth thresholds are reduced by the factor 0.85 for these reports, since radar confidence=4 reports are suspected to be clutter.

This process is shown in Figure 7.

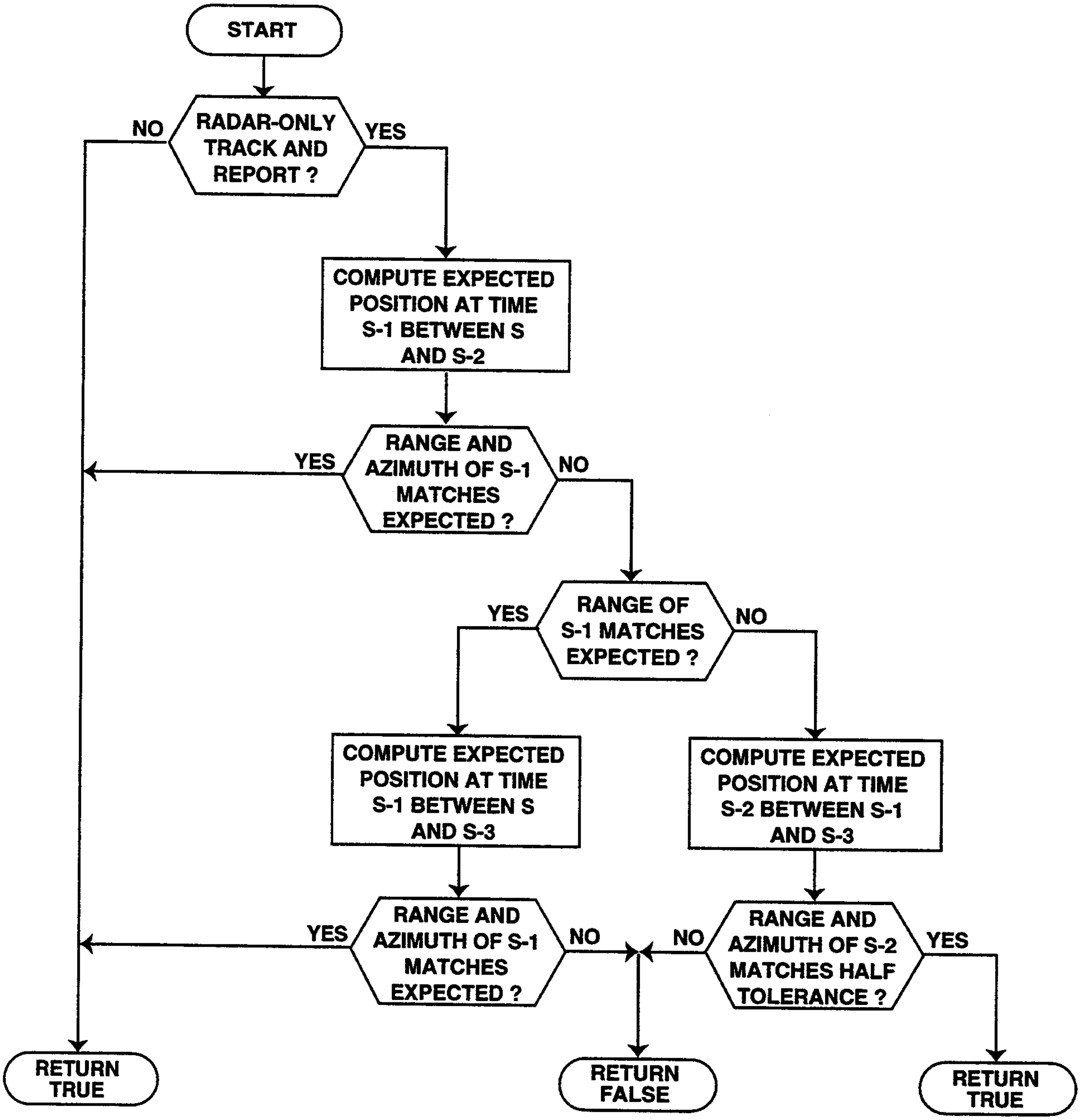


Figure 7. Association Linearity Test.

5.3.2 Correlate Tracks

Called by:	Track
Calls:	Ready to Correlate Possibly Correlate
Purpose:	Steps through the track list and determines if a track is ready for correlation, possibly ready for correlation, or not ready for correlation.

Correlation of tracks is the determination of which report-track association is the best, and marking that association as a correlated pair. Ideally, the determination would not be made until all possible associations have been made for that track, and all possible associations have been made for all of the associating reports, and all associations have been made for all of the associating reports' associating tracks, and so on. Also, ideally, the determination would be made as soon as the correct report was received. However, these two ideals are counter to each other and tradeoffs must be made. This is accomplished by considering each track as possibly ready for correlation as soon as the position of the expected report is passed. If certain conditions are met, correlation is performed early. If the conditions are not met, correlation is delayed. After all possible associations should have been received for a given track, the track is considered ready for correlation. At this point a correlation is made if possible. The process is explained in greater detail below.

Each track on the active track list is considered one at a time. First, it is confirmed the track has not already been marked for correlation. (It is possible that a track could already have been marked, e.g., if it had multiple associations of which at least one was shared with a track earlier in the list.) If the track has already been marked, it needs no further correlation processing and the next track on the list is considered.

Given that a track is not correlated, a check is made between the antenna position, the predicted azimuth position, and the maximum association box azimuth. If the antenna position has not yet reached the predicted position, no correlation attempt is made; chances are good the best association has yet to be found. If the antenna position has passed the predicted azimuth position, it is considered possibly ready for correlation; the best association may have been found, but all associations probably have not been found. If the antenna position is more than 4 sectors (4*128 ACPs) beyond the maximum association box azimuth, or more than 4 sectors (4*128 ACPs) beyond the track's predicted azimuth, the track is considered ready for correlation. The details for Possibly Correlate and Ready to Correlate are in Sections 5.3.2.1 and 5.3.2.2.

Note that if the 9-PAC Phase II Tracker is operating in conjunction with a Mode S sensor, there may be additional delays involved with the processing of reports. To account for these delays, an additional 3 sectors (3*128 ACPs) of delay is allowed before the track is considered ready for correlation.

This process is shown in Figure 8.

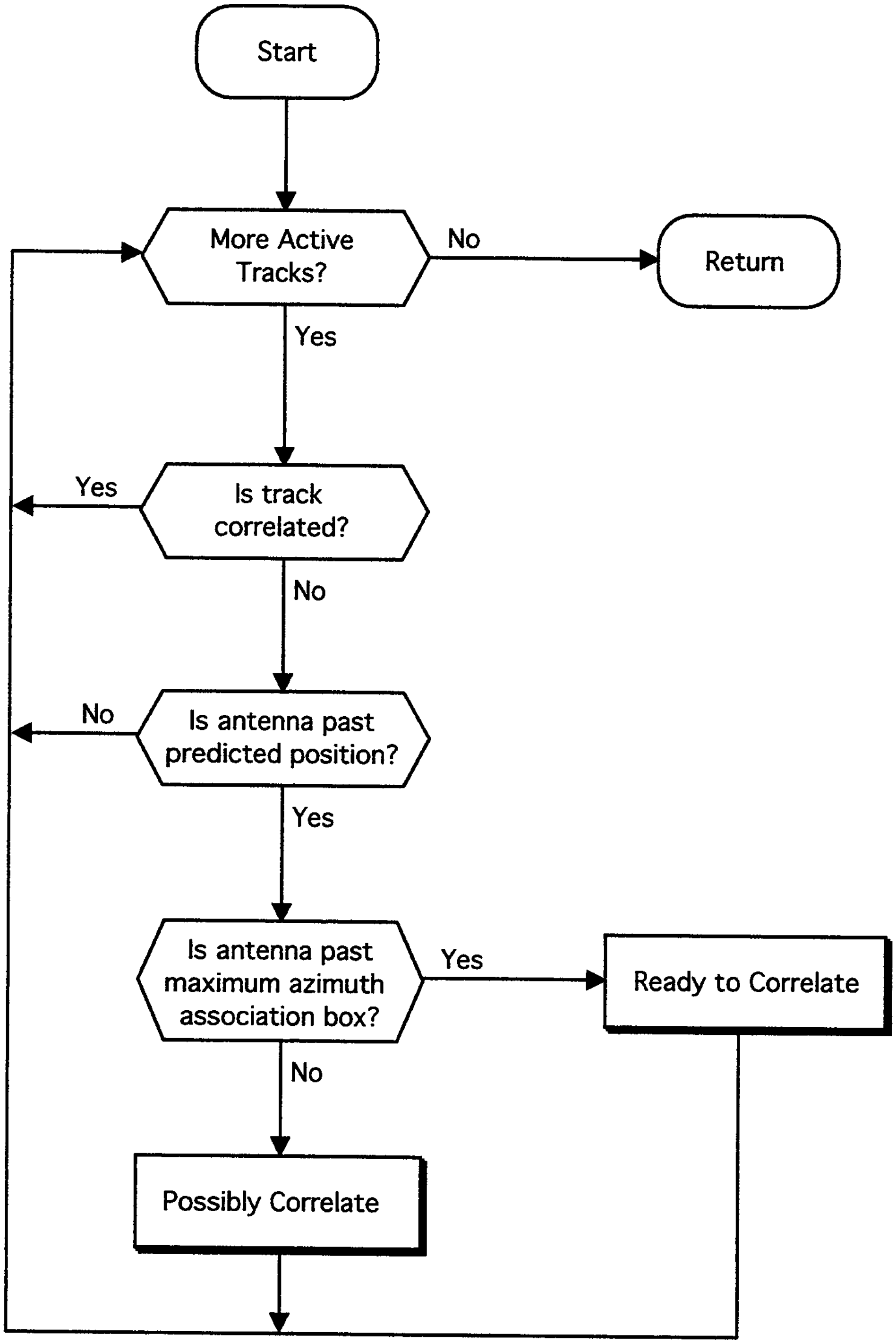


Figure 8. Correlate tracks.

5.3.2.1 Possibly Correlate

Called by:	Correlate Tracks
Calls:	Beacon Split Compute Score Conflict Resolution Mark Pair
Purpose:	Determines if a track should be correlated early. If a single very good association exists, correlation is performed early. If there are multiple associations or no outstanding associations, correlation is delayed.

A track is considered a possible candidate for correlation if the antenna position has passed the predicted azimuth position, but has not passed the maximum association box azimuth. In a typical ASR-9 environment, more than 90% of the reports and tracks have only one association. Given this information, it is likely that the best, and most likely only, association has already been made since the predicted position has been passed. If this is truly the case, then the associated pair should be considered for early correlation. This process is shown in Figure 9.

The first step before determining whether a track should be correlated early, is to assure there are not multiple discrete associations due to a beacon split. A discrete association is the association of a beacon report with a discrete Mode 3/A code to a track with an identical Mode 3/A code. If there are multiple discrete associations, a special function to handle beacon splits is called. This function is detailed in Section 5.3.2.3.

If the track of interest has a one-to-one association with a report, a measurement of the quality of the association is made. A numeric score, which is essentially a multidimensional distance, is computed. If this score falls below a predetermined threshold, the association is accepted for early report-to-track correlation. The details for computing this score are discussed in Section 5.3.2.4. The score threshold for a radar-only report association requires an inner-zone position (within measurement error) and Doppler agreement. The score threshold for a beacon-only or radar-reinforced beacon association also requires an inner-zone position.

If the track has only one associating report, but the report has more than one associating track, it is assumed the track is near a clutter area, or a crossing aircraft situation is occurring, or some other atypical situation. It is best in this situation to allow all available information to be gathered before making a correlation decision. The track of interest is returned to the list and correlation will be attempted after the maximum association box azimuth has been passed and all possible associating reports have been gathered.

It is possible that a single track has multiple associations, but only one is a discrete association: the track's discrete code matches a report's discrete code. If this is the case, it is most likely that the discrete association is the best association, and the other associations are due to surrounding clutter. To avoid unnecessarily delaying the output due to the non-discrete association(s), this track and its multiple associating reports are immediately sent to Conflict Resolution. Conflict Resolution is a method for determining the best correlations given a number of cross associating tracks and reports. This function is detailed in Section 5.3.2.6.

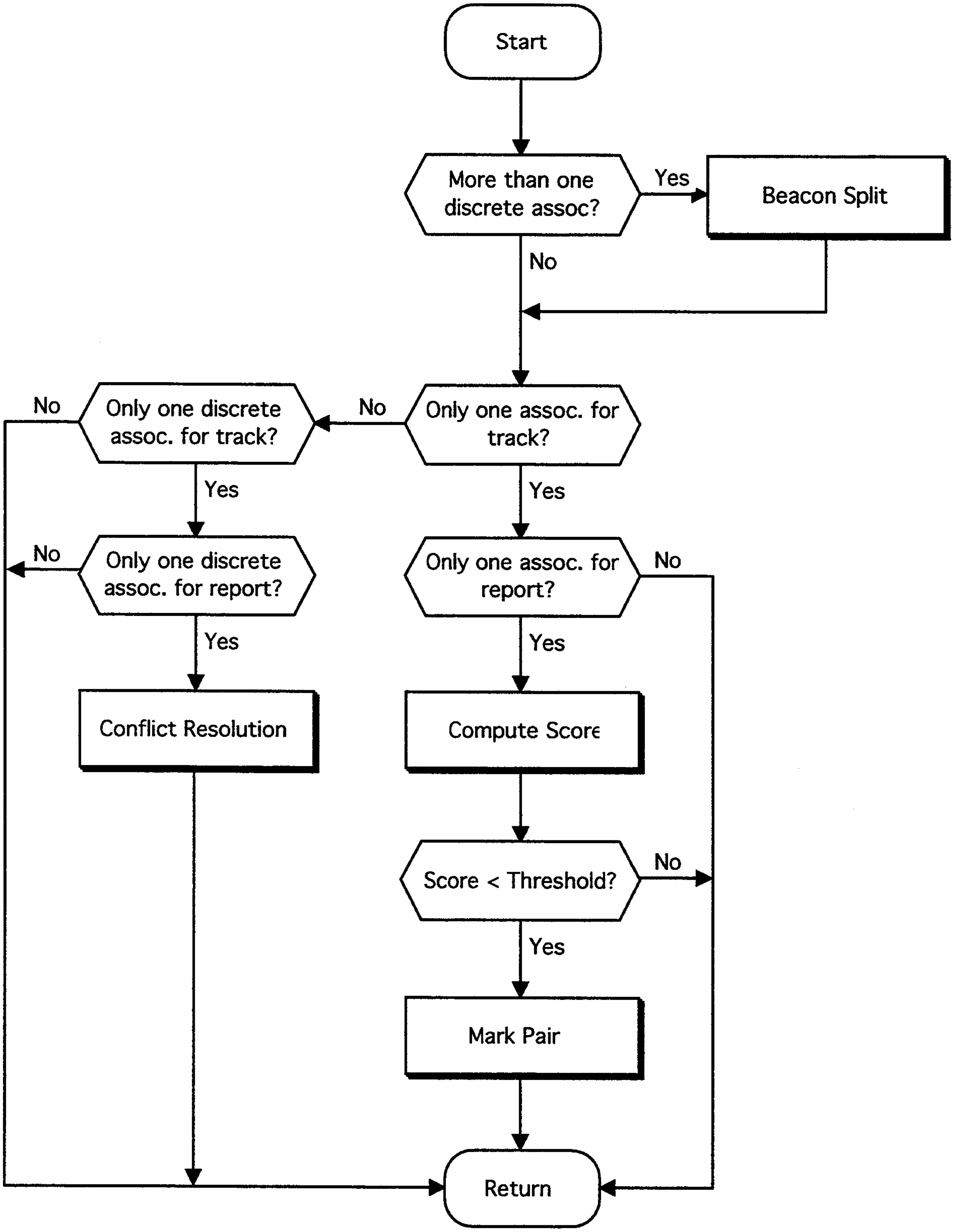


Figure 9. Possibly correlate.

5.3.2.2 Ready To Correlate

Called by:	Correlate Tracks
Calls:	Beacon Split Conflict Resolution Mark Pair
Purpose:	Attempts to correlate a report and track.

If the antenna position has passed the maximum association box azimuth, all candidate reports will have been received and associated (assuming no unusual system delay or overload problem). At this point, it is time to correlate the track to the best associating report. This process is shown in Figure 10.

As with the Possibly Correlate testing, the first step is to assure there are not multiple discrete associations for a discrete track. This would be caused by beacon splits, and a special Beacon Split function is called to handle this situation. The details of the Beacon Split function are discussed in Section 5.3.2.3.

If the track of interest has a one-to-one association with a report, the pair should be correlated. No further testing is necessary.

If the track has no associating reports, the track is marked for coasting. No further testing is necessary.

If the track has more than one associating report, or if there is just one associating report, but it has more than one associating track, the Conflict Resolution function is called. Conflict Resolution is a method for determining the best correlations given a number of cross associating reports and tracks. It is detailed in Section 5.3.2.6.

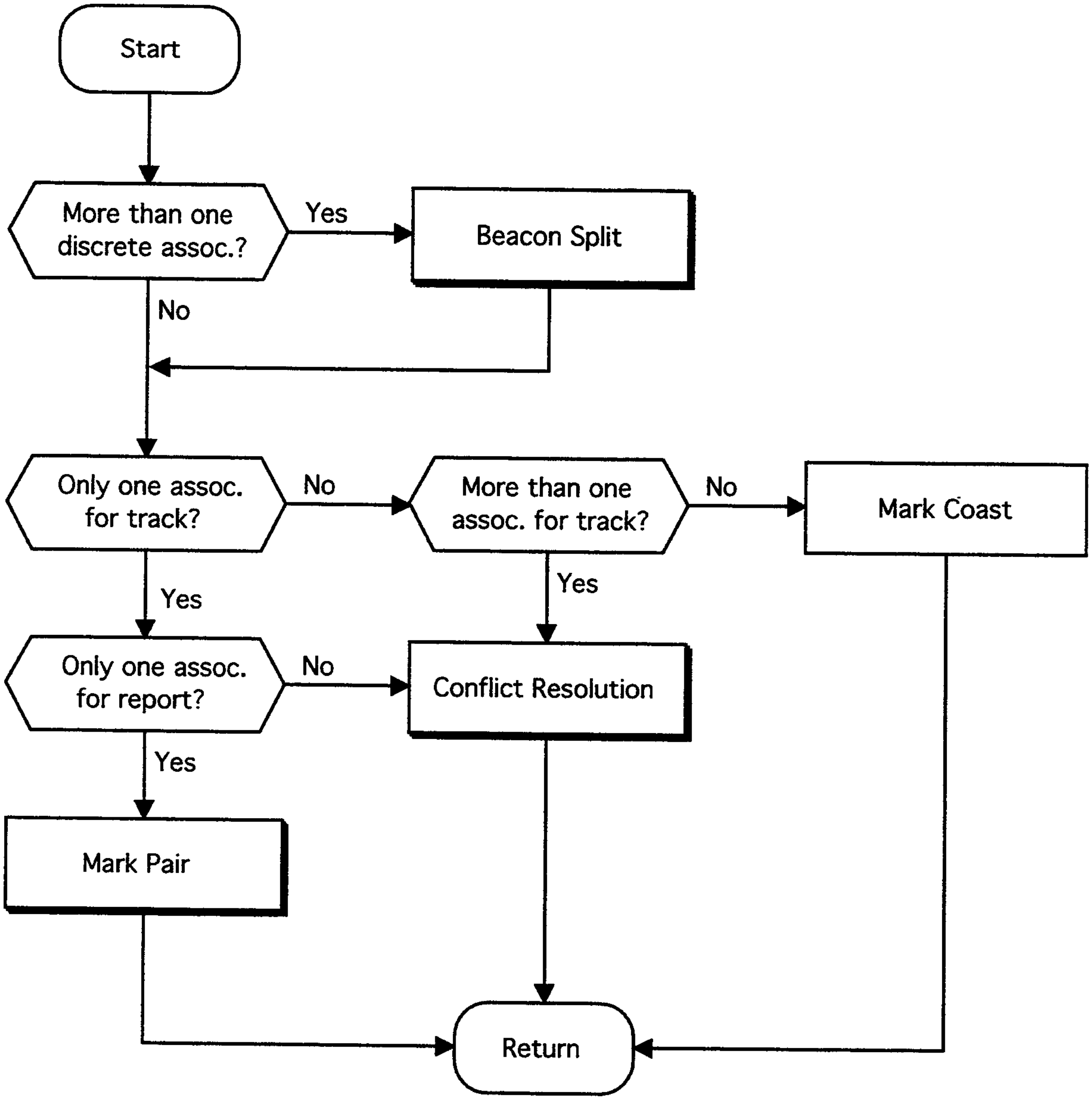


Figure 10. Ready to correlate.

5.3.2.3 Beacon Split

Called by:	Possibly Correlate Ready to Correlate
Calls:	
Purpose:	Handle discrete beacon splits.

Beacon splits (multiple beacon reports from the same aircraft on the same scan) can be due to azimuth or range splits which are typically environment induced, or due to a faulty transponder. Most beacon splits result in poor azimuth or poor range position for both reports. Regardless of the cause of the split, it is important to update the track in a reasonable manner and to avoid initializing additional discrete tracks with the extra split beacon reports.

The 9-PAC Phase II Tracker handles discrete beacon split cases (multiple discrete Mode 3/A code beacon or radar-reinforced beacon associations to a single discrete Mode 3/A code beacon or radar-reinforced beacon track) by selecting the "best" association using the beacon scoring algorithms from Section 5.3.2.4 below (in a similar manner to how other "conflict situations" are handled, see Section 5.3.2.6 below). Since the beacon scoring algorithms consider many other factors besides Mode 3/A code agreement (position, altitude, etc.), they are likely to find the proper report from the set of associations to correlate with the track. All the other associated beacon reports which failed to correlate with the track are marked so that they cannot initiate new tracks.

It is possible that one or more of the discrete Mode 3/A code beacon reports discretely associated to the given track might also be associated to other tracks. These other associations might also be discrete (somehow, multiple tracks with the same code got into the system). To resolve this situation, the 9-PAC Phase II Tracker attempts to find the most mature track from the set of associated tracks that has a unique discrete report association and has not yet correlated or coasted on this scan. If such a track is found, this track alone gets its discrete correlation. (This process makes it more likely that the other, probably spurious, discrete tracks will coast and subsequently drop.) If such a track cannot be found, the default conflict-resolution processing from Section 5.3.2.6 is used.

5.3.2.4 Compute Score

Called by:	Possibly Correlate Ready to Correlate Conflict Resolution Build Association Matrix
Calls:	Doppler Agreement Test
Purpose:	Compute a numerical association measure relating a given report to a given track.

The association measure score is a measure of the quality of the association between a track and report. It is a function of many different report and track fields in addition to the true distance between the report's position and the track's predicted position. It is essentially a multi-dimensional distance; the lower the total score, the better the association. This process is shown in Figure 11.

The score is actually computed by performing a series of nine sub-tests and assigning higher test sub-scores for less than ideal results. Each sub-score is defined as an octal "digit" of the total score – hence, each sub-score is a value between 0 and 7. The concatenation of the digit values for each of the sub-scores forms the total score. The use of digit concatenation automatically weights the contributions of the sub-tests to the overall association score (i.e. higher-order sub-score digits are more significant than lower-order digits).

There are actually two separate scoring processes: one for radar-only and a second for beacon cases. The radar-only path is used when either the track or report is radar-only (and has no beacon data available with which to perform the beacon tests). If both the track and report have beacon data available, then the beacon path is used, since the presence of Mode 3/A codes and Mode C altitudes make it easier to pick the correct report-track association. Note that the high-order digit of the radar-only scoring is set in such a way that the best radar-only score (report in the inner track association box) is commensurate with the beacon score for an outer zone/code disagree case. This is done to bias the scoring so that a beacon-supported track with a choice between a radar-only association (that might be clutter) and a reasonable beacon association (even if it is in the outer association zone) will choose the beacon association.

The association score is broken down into two component parts. The high-order seven octal "digits" of the score are always computed. The low-order two octal "digits" of the score (termed the "deviation score") are used only as tie-breakers when the high-order digits fail to resolve a conflict case. The "deviation score" (described below) requires a fair amount of computation, and is rarely needed in practice. Hence, it is normally omitted (its two digits of the total score defaulted to 0). The deviation score is only computed when necessary.

Computation of the Radar-Only Score. – Whenever the report or the track is radar-only the association score digits are computed as follows.

Digit	Decision Item	Value
8	inner association zone	3
	outer association zone	6

Note: the choice of values (3,6) is made so that an inner-zone radar-only report can compete against an outer-zone, Mode 3/A code-disagree beacon report. A zone-2 radar-only report will lose to any beacon association.

Digit	Decision Item	Value
7,6,5	scaled positional scoring (total of 9 bits) $S = 4 * \text{sqrt}[(\Delta\rho/\Delta\rho^{j1}{}_{max})^2+(\Delta\theta/\Delta\theta^{j1}{}_{max})^2]$ where the Δ quantities are the differences between the track and target.	

The value $\Delta\rho^{j1}{}_{max}$ is the radar range sigma VSP and $\Delta\theta^{j1}{}_{max}$ is the radar azimuth sigma VSP. The value of 'S' is limited to a maximum value of 511 (9 bits = 127 sigma) and rounded to the nearest integer. The scaling of the Mahalanobis-distance measure (i.e., the distance in radar sigma units) into quarter-sigma units accounts for the difference in actual distance in the range and azimuth directions. The scaling of the distance measure also accounts for the radar's measurement accuracy of the reports.

Digit	Decision Item	Value
4	report-track Doppler agreement	0
	report-track Doppler disagreement	1

The Doppler test is described in Section 5.3.2 5 below. Note: if the report's azimuth degrade flag is set, if the report's confidence is less than 3, or if the report's quality is less than 2, the Doppler agreement test fails by default.

Digit	Decision Item	Value
3	track mature (> 7 scans), not coasting	0
	track mature (> 7 scans), coasting	1
	track immature (4-7 scans), not coasting	2
	track immature (4-7 scans), coasting	3
	track new (≤ 3 scans), not coasting	4
	track new (≤ 3 scans), coasting	5

Note: a radar track with 3 or fewer update scans is considered "new." A track with 4 to 7 update scans is considered "immature." A track with more than 7 update scans is "mature." This test biases the scoring toward mature tracks that have had recent updates (not coasting).

Digit 2 is composed of the sum of two factors. The first factor involves the report and track radar confidence, while the second factor involves the report and track radar quality. The total digit sum ranges from 0 to 5.

Digit	Decision Item	Value
2	report and track confidence	0
	report confidence ≥ 3, track confidence < 3	1
	report confidence < 3, track confidence ≥ 3	2
	report confidence < 3	3
2+	report and track quality both = 3	+0
	report and track quality both < 3	+1
	report or track, but not both, quality = 3	+2

Digit	Decision Item	Value
0, 1	reserved for deviation score	

Computation of the Beacon Quality Score. – Whenever neither the report nor the track are radar-only, the association score digits are computed as follows.

Digit	Decision Item	Value
8	inner association zone, Mode 3/A codes agree	0
	outer association zone, Mode 3/A codes agree	1
	inner association zone, Mode 3/A codes disagree	2
	inner association zone, Mode 3/A codes disagree	3
	none of the above conditions	4

Note: Mode 3/A codes "agree" if there are fewer than 2 bit differences in the 12 code bits of the report and track.

Digit	Decision Item	Value
7	report & track Mode 3/A agree, both validity = 3	0
	report & track Mode 3/A agree, both validity > 0 (either validity ≠ 3)	1

report & track Mode 3/A agree, either validity = 0	2
report & track Mode 3/A, modest agreement	3
report & track Mode 3/A disagree, transition, (either validity $\neq$ 3)	4
report & track Mode 3/A disagree, transition, (both validity = 3)	5
report & track Mode 3/A disagree, no-transition, (either validity $\neq$ 3)	6
report & track Mode 3/A disagree, no-transition, (both validity = 3)	7

Note: "agree" is defined as an exact bit-match between the Mode 3/A codes of the report and track. "Modest" agreement is defined as just 1 bit difference. Disagreement is defined as more than 1 bit difference. "Transition" refers to the track code changing condition (see Section 5.3.3.5).

Digit 6: Mode C altitude agreement between report and track

Each report or track's Mode C altitude can be in one of 5 possible states as follows:

(1) No Mode C
(2) Brackets only (validity < 3)
(3) Brackets only (validity = 3)
(4) Flight level (validity < 3)
(5) Flight level (validity = 3)

The procedure for determining the value for digit 6 of the beacon association score is defined in the following table as a function of the altitude states of the report and track respectively. Wherever a number appears in the table, this is the value of the association score digit. Wherever a letter (in boldface type) appears in the table, this refers to the algorithm for computing the digit for this case (see the notes below the table). This algorithm is derived from Reference [7].

	Report Altitude State:				
	(1)	(2)	(3)	(4)	(5)
Track Alt. Type					
(1)	0	4	4	4	4
(2)	4	0	0	**A**	**A**
(3)	4	0	0	3	3
(4)	4	**A**	3	**B**	**B**
(5)	4	**A**	3	**B**	**C**

(A): Convert the report and track altitudes to 12-bit Gray code bits. Compute the number of bit differences between the altitudes, after masking out the 3 'C' bits of each altitude. If the number of bit differences is ≤ 1 then the score digit is set to 1, else it is set to 3.

(B) Convert the report and track altitudes to 12-bit Gray code bits. Compute the number of bit differences between the altitudes, after masking out the 3 'C' bits of each altitude. If the number of bit differences is ≤ 1 then the score digit is set to 1. If the number of bit differences is 2 or 3 then the score digit is set to 5, else it is set to 6.

(C): Compute the absolute value of the difference between the report and track altitudes. If the absolute difference $|\Delta h|$ is > 10 flight levels (1000 feet), perform the processing for case **(B)** above. (An altitude difference of more than 1000 feet is assumed to be due to altitude bit error(s).) If the absolute difference $|\Delta h|$ is ≤ 5 flight levels (500 feet), the score digit is set to 0. (A difference of 500 feet or less is assumed to be caused by normal aircraft maneuvering.) Otherwise (if the absolute difference $|\Delta h|$ is between 6 and 10 flight levels), the score digit is set to $(|\Delta h| - 5)$.

Digit	Decision Item	Value
5	track mature (≥ 5 scans), not coasting	0
	track mature (≥ 5 scans), coasting	1
	track immature (< 5 scans), not coasting	2
	track immature (< 5 scans), coasting	3

Note: a beacon track with fewer than 5 update scans is considered "immature." A track with 5 or more update scans is "mature." This test biases the scoring toward mature tracks that have had recent updates (not coasting).

Digit	Decision Item	Value
4	report Doppler values that agree with track range-rate	0
	report lacks Doppler values	1
	report Doppler values disagree with track range-rate	2

The Doppler test is described in Section 5.3.2.5 below. Note: if the report's azimuth degrade flag is set, if the report's confidence is less than 3, or if the report's quality is less than 2, the Doppler agreement test fails.

Digit	Decision Item	Value
2,3	scaled positional scoring	

$$S = 4 * \mathrm{sqrt}[(\Delta\rho/\Delta\rho^{j1}_{max})^2+(\Delta\theta/\Delta\theta^{j1}_{max})^2]$$

where the Δ quantities are the differences between the track and target.

The value $\Delta\rho^{j1}_{max}$ is the radar range sigma VSP and $\Delta\theta^{j1}_{max}$ is the radar azimuth sigma VSP. The value of 'S' is limited to a maximum value of 63 (6 bits = 15 sigma) and rounded to the nearest integer. The scaling of the Mahalanobis-distance measure (the distance between the report and the track in terms of radar sigma) into quarter-sigma units accounts for the difference in actual distance in the range and azimuth directions. The scaling also accounts for the measurement accuracy of the reports.

Digit	Decision Item	Value
0,1	reserved for deviation score	

Computation of Deviation Score. – The deviation score algorithm provides a measure of the likelihood that an aircraft under track could be at the position of an associated target report. Deviation scoring augments the positional scoring algorithm (digits 2-3 in beacon quality scoring or digits 5-7 in radar quality scoring) to break potential ties in the overall report-to-track scoring process. The track association zones are rectangles in range-azimuth coordinates, but all locations within an association zone are not equally likely. Aircraft speed changes from one scan to the next are uncommon, and nearly all apparent changes in aircraft velocity are due to turns. Hence, deviation scoring determines whether a reasonable aircraft turn over the last scan could deflect the aircraft track to the associated target report position. (This algorithm is derived in Reference [7].)

Deviation scoring is based on the calculation of two vectors: **D** and **T**. The vector **D** represents the deviation from the track predicted position to the associated report position and its components (in range-azimuth coordinates) are given by:

$$D_{range} = \Delta range$$

$$D_{azimuth} = \text{track predicted range} * \Delta azimuth$$

where Δrange and Δazimuth give the differences between the track's predicted position and the report's position. The **T** vector approximates the turning locus for the track (the set of current-scan positions that could be reached by an aircraft undergoing a reasonable turn from the track's position last scan). While the turning locus is actually curved, deviation scoring assumes a straight turning locus for computational ease. A rate of turn defined by the VSP value for number of aircraft 'g's (NG) is assumed, denoted by the symbol "w" (converted to radians/scan units). (Note: a 1-g standard-rate turn is defined as 3 degrees per second.) Define parameters 'A' and 'B' as:

$$A = (1 - \cos(w)) / w$$

$$B = 1 - \sin(w)/w$$

(The values of A and B are precalculated at system initialization.) Also, define the track's velocity in the range direction as "rdot" and the track's velocity in the azimuth direction as "adot" (in radians/scan). The **T** vector is divided into a unit-vector parallel to the track's direction $\mathbf{T}_l$ and a unit vector perpendicular to the track's direction $\mathbf{T}_c$. The components of $\mathbf{T}_c$ are defined:

$$\mathbf{T}_{c\ \mathrm{range}} = -A * \text{track predicted range} * \text{adot}$$

$$\mathbf{T}_{c\ \mathrm{azimuth}} = A * \text{rdot}$$

Similarly, the components of $\mathbf{T}_l$ are defined:

$$\mathbf{T}_{l\ \mathrm{range}} = B * \text{rdot}$$

$$\mathbf{T}_{l\ \mathrm{azimuth}} = B * \text{track predicted range} * \text{adot}$$

The magnitude of the project of the deviation vector **D** along (parallel to) the turning locus vector **T** is given by the absolute value of the dot product of the vectors, denoted "C_{par}". The magnitude of the projection of the deviation vector **D** across (perpendicular to) the turning locus vector **T**, denoted "C_{perp}", is calculated as follows:

$$\text{dmag}^2 = D_{\mathrm{range}}{}^2 + D_{\mathrm{azimuth}}{}^2$$

$$C_{perp} = \text{square root}(\text{dmag}^2 - C_{par}{}^2)$$

The actual deviation score is then computed as the ratio of C_{perp} to C_{par} multiplied by 25 and converted to an integer. Deviations due to turns (C_{perp} small) are penalized less than deviations due to aircraft speed changes (C_{par} small). An upper bound of 50 is imposed on the deviation score to avoid problems with numeric overflows. (Note: since the deviation score occupies the two low-order octal digits in the quality score (6 bits), values from 0 to 63 would be permissible.)

Resolution of Scoring Ties. – Since the association score value for a given track-report association includes a measure of all the available information that the 9-PAC Phase II Tracker has on the track and report, there is nothing beyond the association score to use for the extremely rare case of a tie between multiple associations. The 9-PAC Phase II Tracker utilizes a "first-come first-served" tie-break strategy.

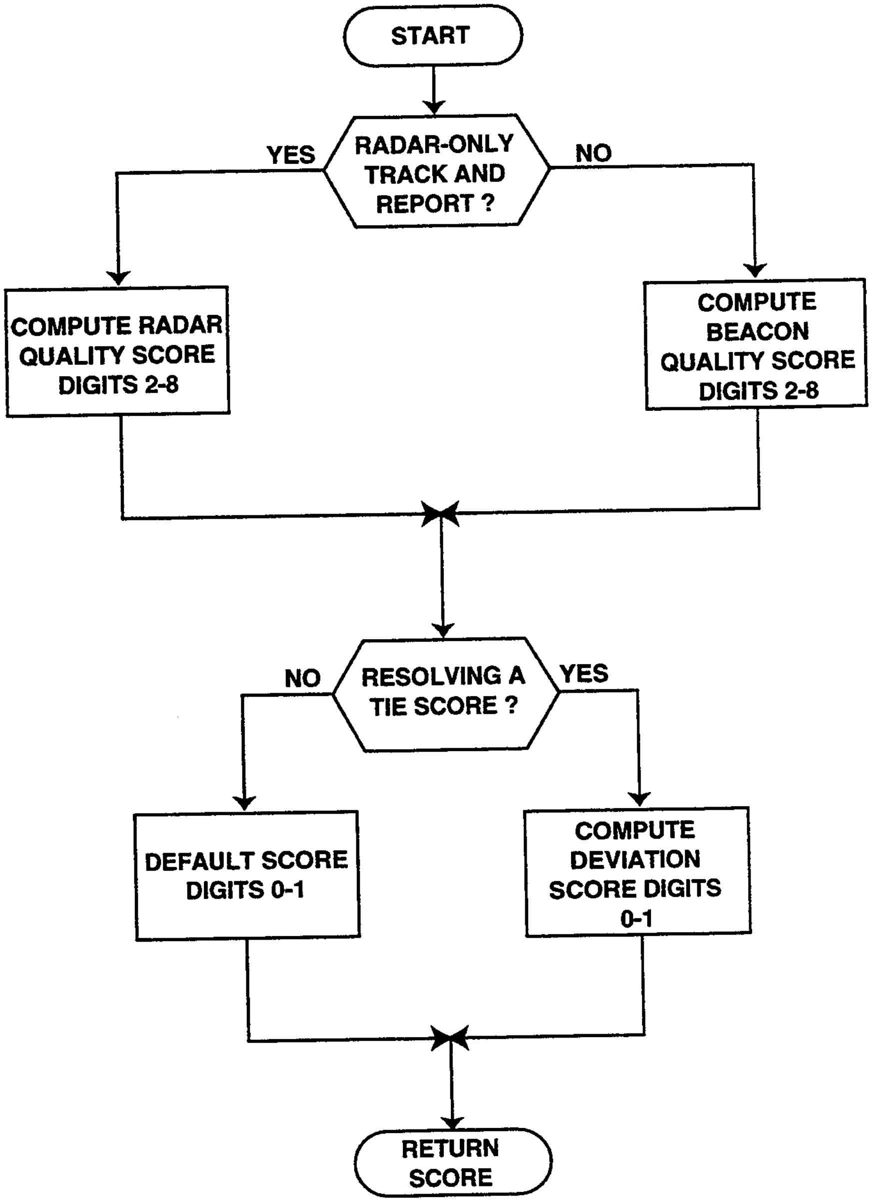


Figure 11. Compute Quality Score.

5.3.2.5 Doppler Agreement Test

Called by:	Compute Score
Calls:	
Purpose:	Determines if the interpolated Doppler numbers for a report are correct if the report were to update a certain track.

All radar reports have one or two interpolated Doppler numbers. Studies have shown that these Doppler numbers agree with the radial velocity of the target more than 85% of the time. The Doppler Agreement Test assumes that a report is used to update an existing track and computes the estimated radial velocity of the track given the new report. A comparison is then made between the new radial velocity estimate and the report's interpolated Doppler numbers. If the numbers agree, the test is passed. The high level basics of the test follows.

An estimate of the range rate of the track is made using the new report as the most recent update. This estimated range rate is used to compute the expected interpolated Doppler numbers for the report. If the actual interpolated Doppler numbers for the new report fall within a defined area (in Doppler space) around the expected interpolated Doppler numbers, the test is passed. This area is shown in Figure 12. Numerically, the test is implemented by computing the differences between the actual and expected interpolated Doppler numbers for both PRFs. Let these differences be *dl* and *dh* for the low and high PRFs, respectively. If the differences satisfy the following equations, the test is passed.

$$0.83dh + 14.0 \le dl \le 0.83dh - 3.0$$

$$2.00dh - 18.75 \le dl \le 2.00dh - 45.25$$

If the report only has one interpolated Doppler number (Quality = 0 or 2), the test must be simplified. An area can no longer be defined in Doppler space, since the test must be reduced to one dimension. The simplified test checks to see if the actual interpolated Doppler number is within 6 units of the corresponding expected interpolated Doppler number. If so, the test is passed.

When the Doppler Agreement Test is called with a report whose azimuth-degraded flag is set from the input, indicating that its position centroid is suspected of being in error, the test will return a default Doppler failure. If the range of the report is suspect, then the range-rate is not reliable, and potential false-agreements may occur. The Doppler Agreement Test also returns a default Doppler failure for reports whose confidence is less than 3 or whose quality is less than 2. Again, this is done to reduce the chances of false-agreements.

This process is shown in Figure 13.

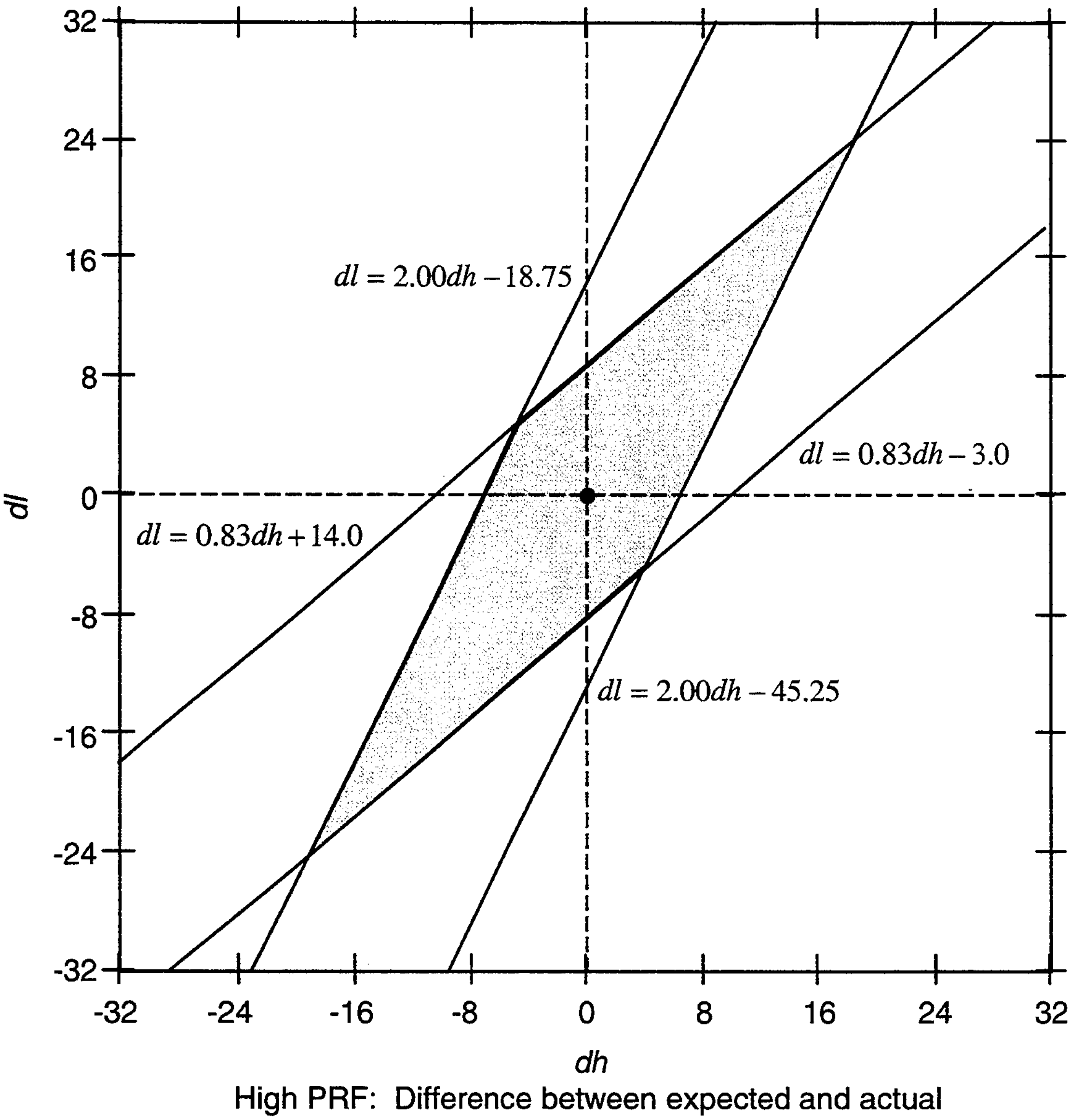


Figure 12. Graphical Doppler Agreement Test. If the differences between the high and low, expected and measured interpolated Doppler numbers falls in the shaded region, the test is passed.

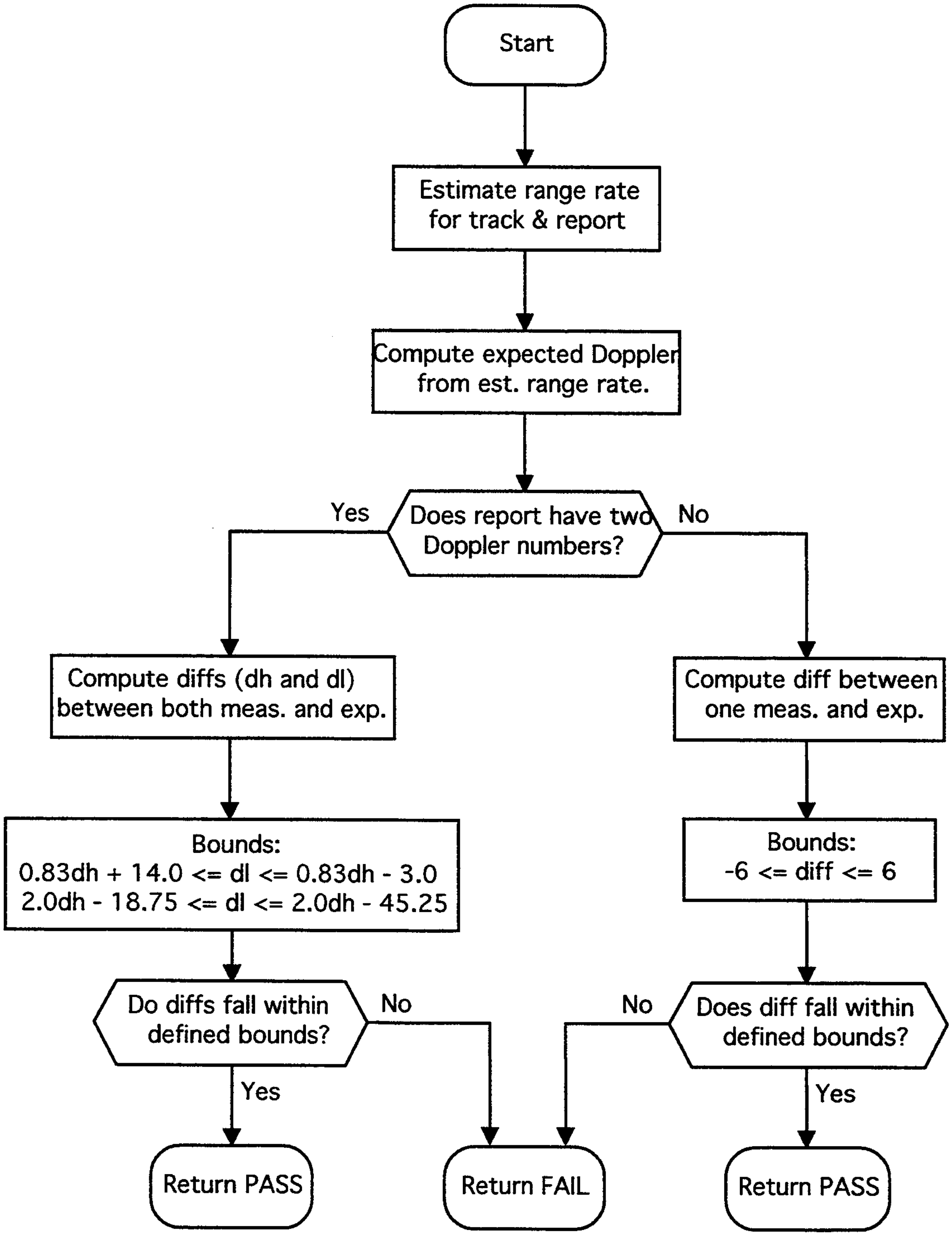


Figure 13. Doppler agreement test.

5.3.2.6 *Conflict Resolution*

Called by:	Possibly Correlate Ready to Correlate
Calls:	Build Association Matrix Modified Munkres Algorithm Compute Score Mark Pair
Purpose:	Determine the best set of correlations given a number of cross associating tracks and reports.

More than 90% of tracks and reports have a one-to-one association (a track associates to just one report, and that one report associates only to that one track). In such simple cases, the best correlation is the same as the only association. However, when any other situation occurs, Conflict Resolution is called. These other cases can be broken down into three basic groups: one track and many reports, many tracks and one report, and many tracks and many reports.

Two of the cases are quite similar and have a fairly simple resolution. For one track and many reports, the association measure is computed for each possible pair, and the report with the lowest association measure score is chosen to correlate with the track. For many reports and one track, the same basic process is used. The association measure is computed for each possible pair, and the track with the lowest association measure score is chosen to correlate with the report. (The association measure score is essentially a multi-dimensional distance; the smaller the score, the closer the association.)

The third case with many tracks and many reports is more complex. It involves building an association matrix which details the cross associations between the tracks and reports. After the matrix is formed, it needs to be reduced in such a manner as to determine which set of associations are the best and should be marked for correlation. A Modified Munkres Algorithm is used for this matrix reduction. The building and solving of the matrix is detailed in Sections 5.3.2.7 and 5.3.2.8.

Note that the association score process is divided into two components for purposes of increasing tracker performance. The simpler (faster) part of the score is computed for all conflict-resolution cases, while the more-complex (slower) "deviation" part of the score is only computed when there are ties to be resolved.

This process is shown in Figure 14.

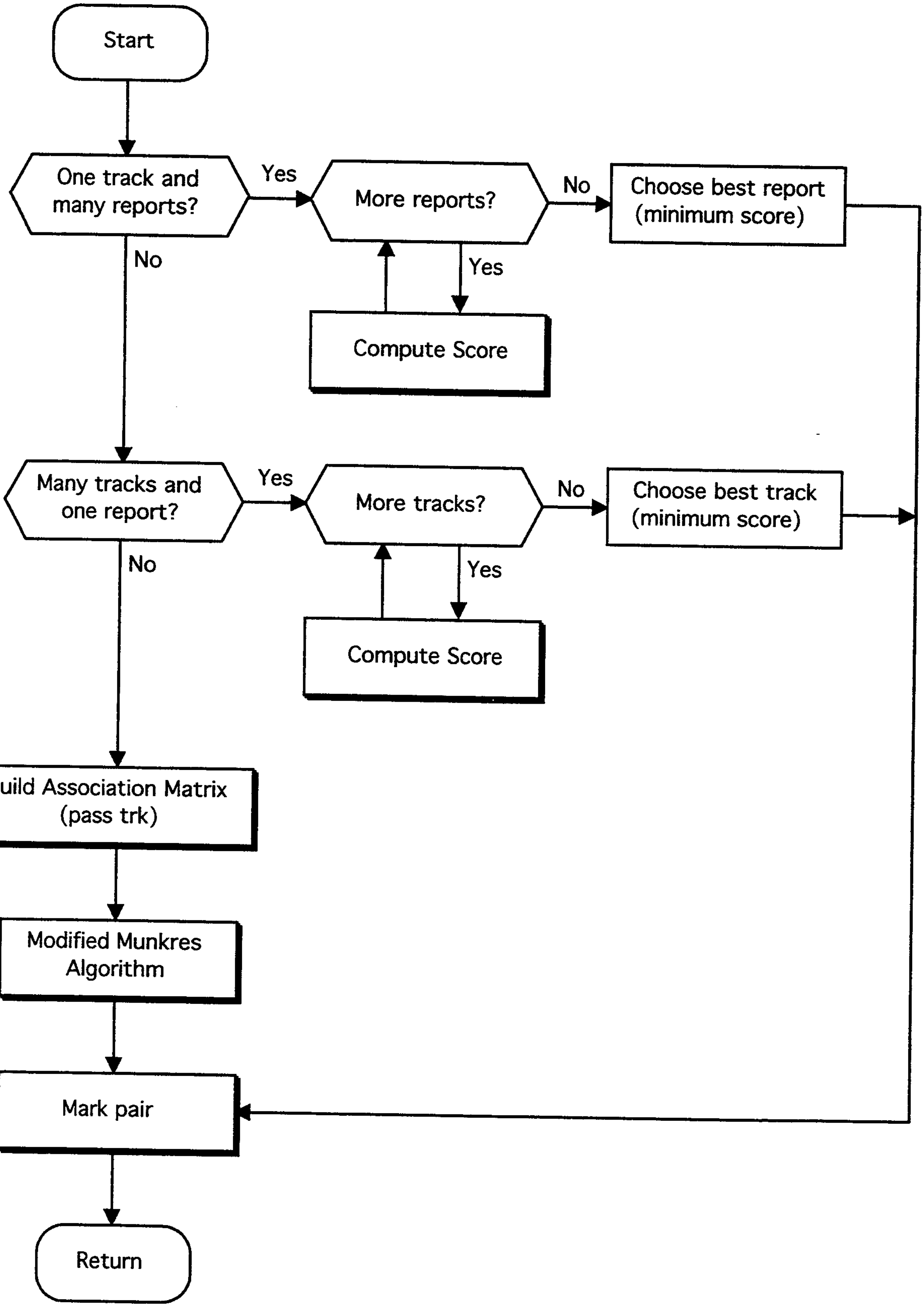


Figure 14. Conflict resolution.

5.3.2.7 Build Association Matrix

Called by:	Conflict Resolution
Calls:	Compute Score
Purpose:	Maps out cross associations between tracks and reports and determines the association measure score for each.

The association matrix is a matrix of cross associating tracks and reports. The matrix rows correspond to tracks and the matrix columns correspond to reports. The matrix elements are the association measure scores for the corresponding tracks and reports. This process is shown in Figure 15.

The first step in building the association matrix is to initialize the matrix elements. Since the association measure is essentially a multi-dimensional distance, a low association measure score corresponds to a close association. Reports and tracks that do not associate with each other but are both in the matrix, need a high number in the association matrix to assure there is no confusion between a null and a perfect association. Hence, the association matrix is initialized with a high default number, e.g., 9999.

After the association matrix is initialized, the matrix is started with a single track passed from Conflict Resolution. The track represents a row in the matrix. All reports that associate with this track are then added to the matrix and represent individual columns. The actual elements of the matrix are the association measure scores relating the specific tracks and reports. These scores are computed and entered in the matrix. An iterative process is then followed to complete the matrix. For each report that was added to the matrix, a search is made for more associating tracks (it obviously has at least one associating track, or the report would not have been added to the matrix in the first place). If there are additional associating tracks, it is confirmed that they are not already part of the matrix, and then added as new rows. After all the new reports have been checked for additional associating tracks, the new tracks are checked for additional associating reports. As new reports and tracks are added to the matrix, the corresponding association measure scores are computed and entered. This process continues until a closed set is formed, that is, there are no more associating tracks or reports to be added, or until the matrix has reached the maximum allowable size. This maximum size is currently set to 10 x 10. An example of the matrix generation process follows.

Table 2 contains a list of tracks, the number of associations for each track, and a list of the associating reports for each track. Table 3 is basically the same but for reports. It has a list of reports, the number of associating tracks for each report, and a list of the associating tracks for each report.

Table 2. Example Track and Associating Reports

Track List	# Assocs.	Assoc. 1	Assoc. 2	Assoc. 3	Assoc. 4
Track A	2	Report A	Report B	-	-
Track B	3	Report A	Report B	Report C	-
Track C	1	Report C	-	-	-

Table 3. Example Report and Associating Tracks

Report List	# Assocs.	Assoc. 1	Assoc. 2	Assoc. 3	Assoc. 4
Report A	2	Track A	Track B	-	-
Report B	2	Track A	Track B	-	-
Report C	2	Track B	Track C	-	-

The first step in building the association matrix is to initialize all of the potential matrix elements to 9999. However, to reduce clutter and increase readability, this step is not shown here.

(1) Begin with Track A, the track received from the calling routine Conflict Resolution. Track A is the first row in the matrix.

Track A			

(2) Using Table 2, add columns for each associating report: Report A and Report B. Compute the association measure score as the matrix element. (The association measure scores shown are representative scores.)

	Report A	Report B	
Track A	7	3	

(3) Using Table 3, find all tracks which associate with Report A in column 1: Track A and Track B. Since Track A is already part of the matrix, only add Track B. Compute the association measure score.

	Report A	Report B	
Track A	7	3	
Track B	6		

(4) Using Table 3, find all tracks which associate with Report B in column 2: Track A and Track B. Both Track A and Track B are part of the matrix so no new rows need to be added. The association measure score for Report B and Track B needs to be computed, however.

	Report A	Report B	
Track A	7	3	
Track B	6	3	

(5) All new columns have been checked. Start checking new rows. Using Table 2, find all reports which associate with Track B in row 2: Report A, Report B, and Report C. Since Report A and Report B are already in the matrix, only one new column for Report C needs to be added. Compute the association measure score.

	Report A	Report B	Report C
Track A	7	3	
Track B	6	3	2

(6) All new rows have been checked. Back to checking new columns (this is an iterative process). Using Table 3, find all tracks which associate with Report C: Track B and Track C. One new row needs to be added for Track C. Compute the association measure score.

	Report A	Report B	Report C
Track A	7	3	
Track B	6	3	2
Track C			4

(7) All new columns have been checked. Go back to check new rows. Using Table 2, find all reports which associate with Track C: Report C. Since this is already part of the matrix, no new columns are added. Since all new rows have been checked and no new columns were added, the iterative process is complete, a closed set has been found, and the matrix is finished.

Sometimes a large number of cross associations exist and it is not computationally practical to allow the association matrix to grow without limits. For this reason, the size of the association matrix is limited by a VSP constant, typically 10×10. If the number of cross-associations would result in a bigger matrix, there is an unusual phenomenon occurring or there is a very high clutter situation. Regardless, the chances of finding a significantly better set of correlations is not large enough to compensate for the increased processing that would result.

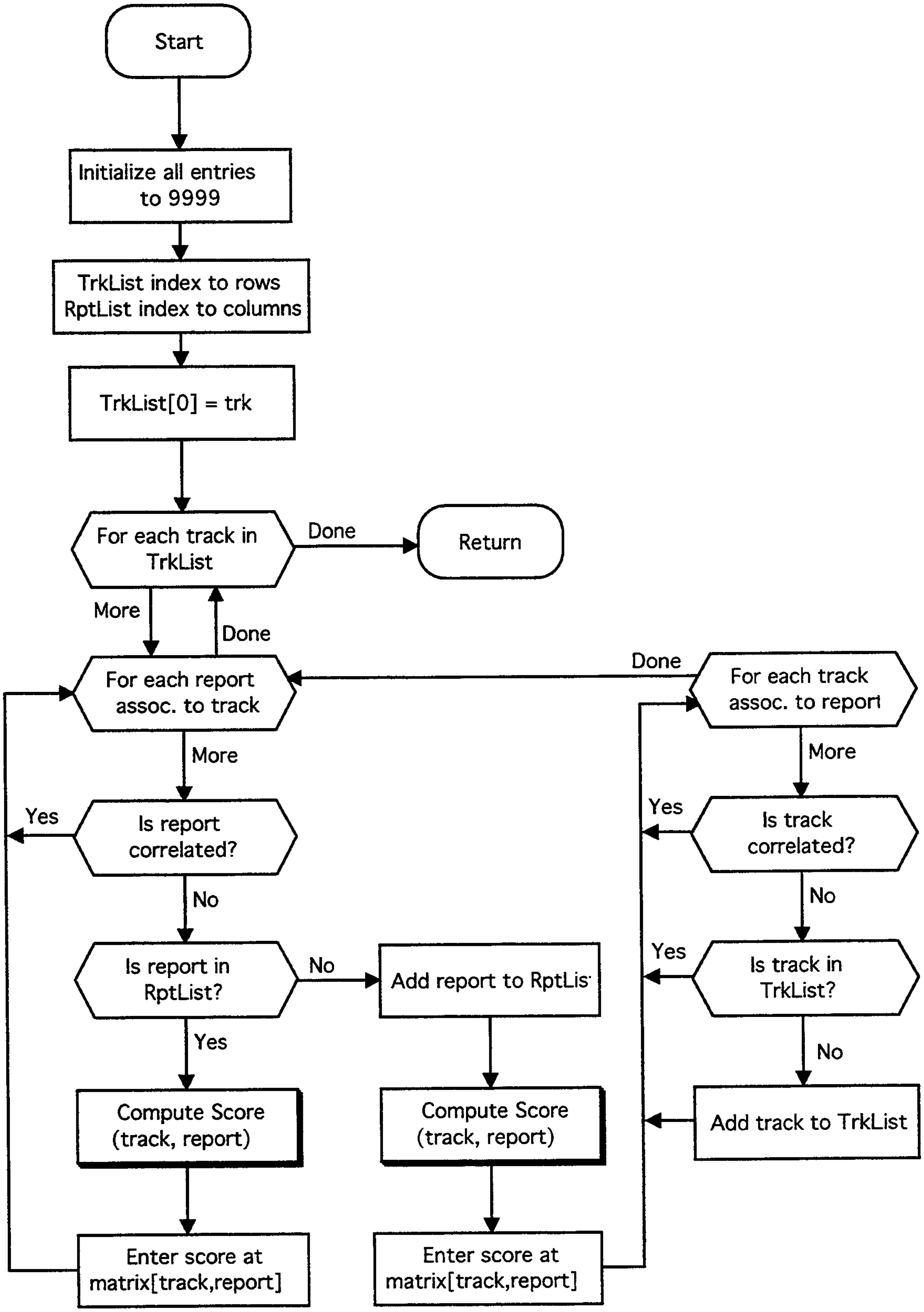


Figure 15. Build association matrix.

5.3.2.8 *Modified Munkres Algorithm*

Called by:	Conflict Resolution
Calls:	Mark Pair
Purpose:	Reduce the association matrix to a set of optimal report-track correlations.

After the association matrix is defined, it is necessary to reduce the matrix in some manner to define the best set of correlations from the closed set of cross associations. For the 9-PAC, the optimal solution has been defined to be the set of correlations with the minimum total association measure scores. If the association measure score is considered to be a multi-dimension distance, the desire is to minimize the total distance for a given set of cross associations.

The method for minimizing the total distance is the Modified Munkres Algorithm (see flowchart in Figure 16). This method gives the same result as the Munkres Algorithm [5] in almost all situations, however, it is computationally much simpler. For an *mxn* rectangular matrix, the process is started with the lesser dimension. If there are *m* rows and *m*<*n*, rows are addressed first. For an *nxn* square matrix, tracks (which are rows) are addressed first. For each row the difference is computed between the lowest and the second lowest score. Boxes with no association measure score are considered to be a very large default number like 9999.

Consider the matrix developed in Section 5.3.2.7 and shown again here as part of Table 4. The last three columns of the table are the second lowest score, the lowest score, and the difference between the two for each row of the association matrix. The computed differences for this matrix are 4, 1, and 9995. The row with the largest difference is correlated first with its best associating column. In this case row 3 has the largest difference, 9995, so Track C is correlated with Report C. Note that since Track C only has one associating report, by assigning 9999 to all of the other boxes, it assured Track C would be correlated first to its one and only associating report, assuming all other tracks have more than one association. If more than one track shared a single report as their only association, this method assures that the best track is correlated first with that report. Track C and Report C are removed from the matrix after being correlated, and the process is repeated with the remaining reports and tracks.

Table 4. Solving an Association Matrix

	Report A	Report B	Report C	2^{nd} Low	Low	Diff
Track A	7	3	9999	7	3	4
Track B	6	3	2	3	2	1
Track C	9999	9999	4	9999	4	9995

Table 5 shows the reduced matrix. Again the last three columns are the second lowest score, the lowest score, and the difference between them for each row. The differences are 4 and 3. The row with the largest difference is row 1, so Track A is correlated with its best report, Report B. Track A and Report B are removed from the matrix which leaves Track B and Report A as shown in Table 6. They are correlated with each other.

Table 5. Solving an Association Matrix

	Report A	Report B	2nd Low	Low	Diff
Track A	7	3	7	3	4
Track B	6	3	6	3	3

Table 6. Solving an Association Matrix

	Report A	2nd Low	Low	Diff
Track B	6	-	-	-

The net effect of this process leaves a total association measure score of (4 for Track C::Report C) + (3 for Track A::Report B) + (6 for Track B::Report A) = (4 + 3 + 6) = 13. Inspection shows this is the minimum possible score possible given this set of cross associations. It is the same answer obtained with the more complex Munkres Algorithm. Study has shown that for more complex cross associations, the same generally holds true.

The situation of two reports associated with two tracks is the most commonly occurring conflict-resolution case. For optimization of tracker performance, 2-on-2 conflicts are handled as a special case of the general algorithm. Let S_{ij} denote the association score for report i and track j in the 2-by-2 association matrix. Then, compute the sums of the scores for the forward and backward diagonals as indicated below:

$$S_{forward} = S_{11} + S_{22}$$

$$S_{backward} = S_{12} + S_{21}$$

If $S_{forward}$ is less than $S_{backward}$, then report 1 should correlate with track 1 and report 2 should correlate with track 2. Alternatively, if $S_{forward}$ is greater than or equal to $S_{backward}$, then report 1 should correlate with track 2 and report 2 should correlate with track 1. Note that if the association score value for the prospective correlation is the default value (indicated as 9999 here), then no correlation is performed (since the default score indicates that the report and track did not associate.

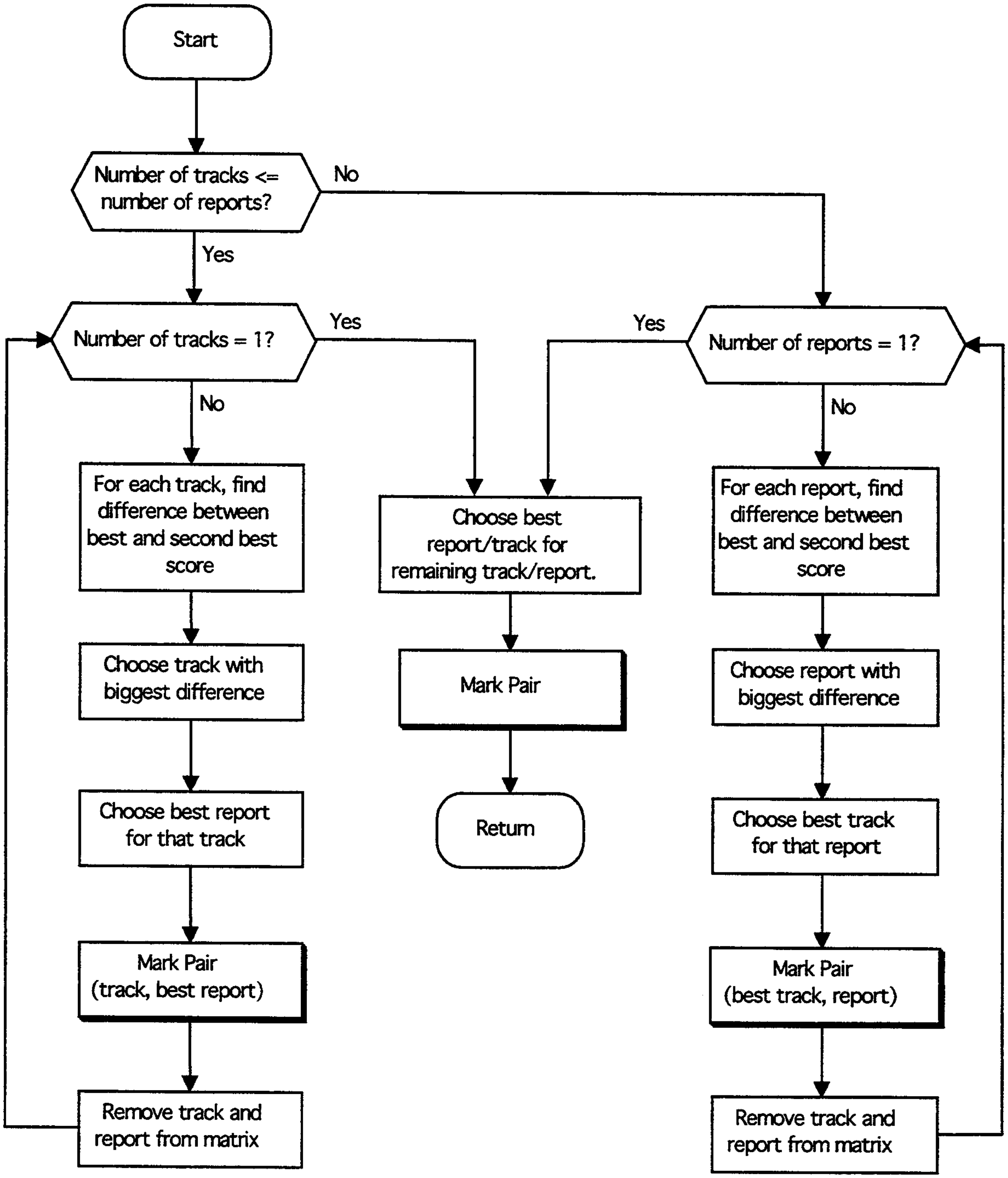


Figure 16. Modified Munkres Algorithm.

5.3.2.9 Mark Pair

Called by:	Possibly Correlate Ready to Correlate Beacon Split Conflict Resolution Modified Munkres Algorithm
Calls:	
Purpose:	Marks correlation between a report and a track, and removes all other associations.

When a report and track are correlated to each other, the correlation needs to be flagged and all associations to other unused reports and tracks should be broken. This is accomplished in the following manner. Each report, which is associated to the track of interest, has a pointer to the track. These pointers are removed. Next the pointers from the track of interest to these reports are removed. Note, the reports had to be handled first or else there would have been no way to find the reports once their pointers were removed from the track.

The process is repeated for the tracks which associated to the report of interest. After all of the pointers are removed from the tracks to the report, the pointers from the report to the track are removed.

Flags are set for the report and the track indicating they have been correlated. Finally, a special pointer is set from the track to the report to be used when updating the track predictions.

This process is shown in Figure 17.

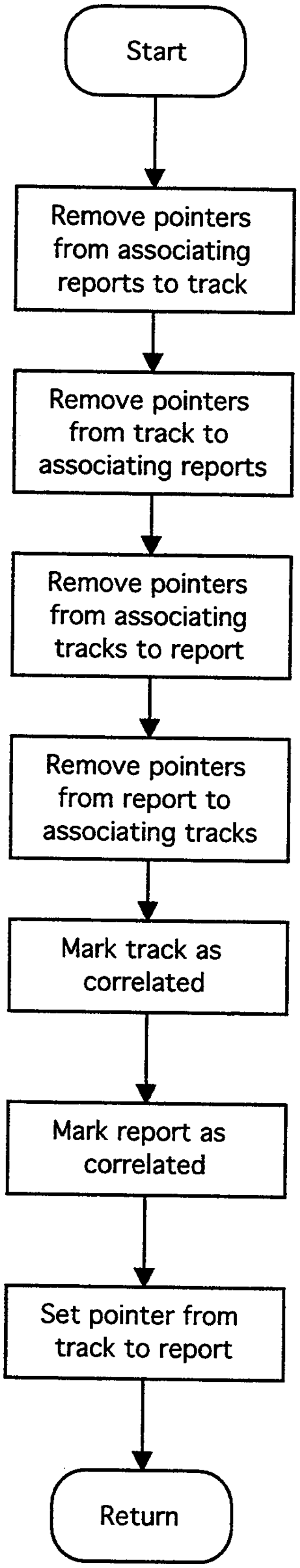


Figure 17. Mark pair.

5.3.3 Update Tracks

Called by:	Track
Calls:	Track Coast Track Update
Purpose:	Determines if a report should be updated by a correlating report or coasted.

The Update process (see Figure 18) steps through all of the Active Tracks. If a track has been marked for coasting, the Track Coast routine is called. If a track has been flagged as correlated, the Track Update routine is called. If neither has occurred, the track must not be old enough for update and the next track on the list is considered.

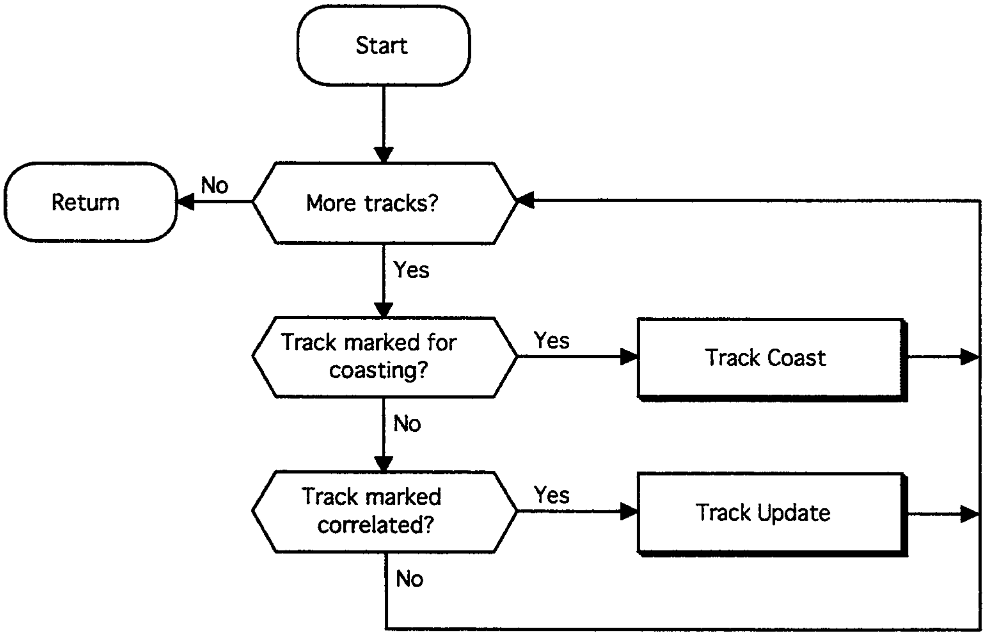


Figure 18. Update tracks.

5.3.3.1 Track Update

Called by:	Update Tracks
Calls:	Determine Gains Update Predictions Compute Velocity Set Track Type Check Minimum Distance Update Track State Compute Association Boxes
Purpose:	Step through all of the processes necessary to determine the best filter gains and predict where the track will be on the next scan. Update the various fields of the track and set feedback/analysis fields in correlating report.

Given a track and a correlating report, a series of steps are followed to determine the degree to which the report should affect the track, and where the track is likely to be on the next scan. First, the new gains are determined (Section 5.3.3.2). The new gains are used to update the track predictions (Section 5.3.3.3). The track velocities are smoothed (Section 5.3.3.4). The track state is updated according to the state diagram (Section 5.3.3.7). The track's movement flag is checked (Section 5.3.3.7). The track's type (radar-only, beacon-only, etc.), Mode 3/A code fields, and Mode C altitude fields are updated (Section 5.3.3.5). New association boxes are drawn based on the new predictions and the track state (Section 5.3.3.8).

If the correlating report has a valid altitude, then the "on the ground" check is performed. The correlating track must be mature (i.e. more than 10 scans old). If the track altitude is within 200 feet of the sensor's height VSP, then the track is declared as "landed." (See section 5.3.3.7 below for a description of how the results of this process are used.)

The final step of track update is to perform the two "track maintenance" tests. If either of the tests fail, the track is set to drop immediately (regardless of its current track state). These tests determine whether the track condition is so bad (even though it has found a correlation) that it would be better to drop the current track and start over by letting a new track initiate.

The scoring test computes the average "points/scan" value for the track averaged over the preceding 5 scans. The average confidence/quality/Doppler score per scan must be greater than the VSP "scoring" parameter (for tracks whose ground speed is greater than or equal to the VSP "speed criterion") or greater than the VSP "slow-speed scoring" parameter for slower tracks. The VSP scoring tables for correlation are used. There are "in-close" (track's range < VSP ELIG_RANGE) and "far" tables (track's range ≥ VSP ELIG_RANGE). Note: the score for a beacon or radar-reinforced beacon track is defined to be the score value for confidence 3, quality 6, plus Doppler agreement.

The current scan update report must pass linearity (similar test to that for radar track initiation) if the track is radar-only and has had at least 3 correlations since its initiation. (*Note: this test is currently disabled in the tracker implementation.*) The linearity maintenance test defaults to "pass" for beacon/reinforced beacon tracks and just-initiated radar tracks. The linearity maintenance test also defaults to "pass" if a "fixup" (see Section 5.3.3.3) has occurred within the preceding two scans. The linearity maintenance test is performed by computing the straight line between the position of the current scan uncorrelated report and the third previous report that correlated with the track. The position on this line corresponding to the scan of the second previous report that correlated with the track is linearly interpolated. The linearity test succeeds if the absolute difference of the interpolated position and the actual position of the second previous report that correlated with the track is less than a parametric number of sensor measurement standard deviations in range and azimuth. A special-case is applied if the correlating report is radar-only and has confidence=4. The range and azimuth thresholds are reduced by the factor 0.85 for these reports, since radar confidence=4 reports are suspected to be clutter.

After all of the update processing has been completed, the track is moved from the Update list back to the azimuth-ordered General Track list (assuming that it has not dropped due to failing one of the track maintenance tests).

This process is shown in Figure 19.

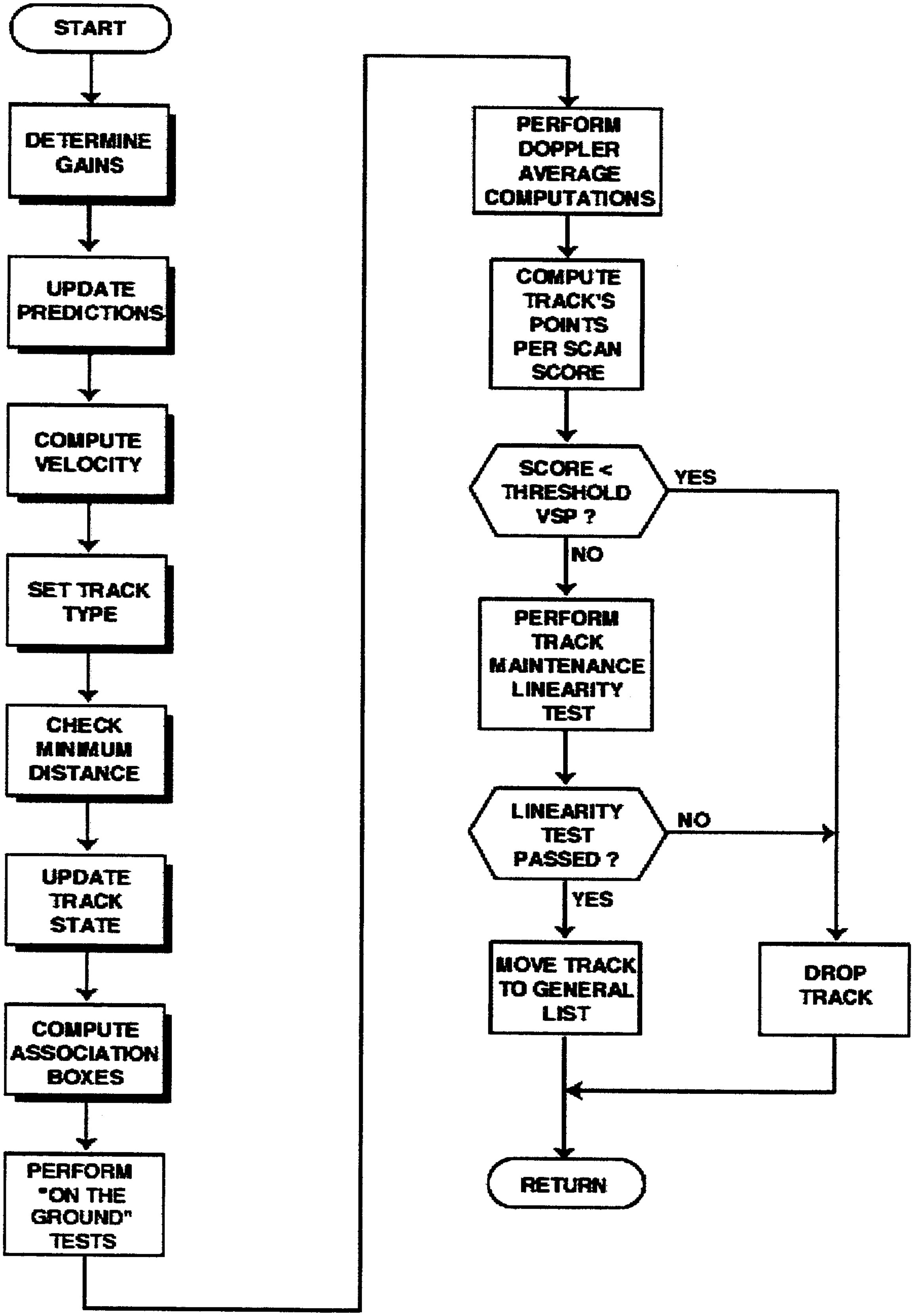


Figure 19. Track Update.

5.3.3.2 Determine Gains

Called by:	Track Update
Calls:	
Purpose:	Determine the filter gains α, β to be used for the alpha-beta track prediction.

The 9-PAC Phase II Tracker update uses an alpha-beta smoothing algorithm (see Figure 20). A comparison was made to a Kalman filter tracker, but the alpha-beta tracker with variable gains proved to be more robust and less computationally intensive than the Kalman filter. The Kalman filter was a better long-term predictor – however, the 9-PAC Phase II Tracker is a scan-scan correlator and is only looking one scan ahead. The improved long-term predictions were not a significant benefit.

The theory used to set alpha in this tracker is as follows. If the reports used to update a track are considered to be accurate, the track should become more stable and trustworthy. As the track becomes more stable and trustworthy, it is beneficial to keep alpha low and the track stiff. However, if a track is too stiff, it may lose a maneuvering or drifting target. If this happens, it is beneficial to increase alpha and follow the measured data more closely.

The original values of alpha and beta are derived from the radar confidence and quality values of the correlating radar report. (Note: the quality value for a beacon-only report is defined to be 4. The quality value for a radar-reinforced beacon may choose either the default beacon-only value '4' or the quality of the matching radar report, depending on a flag field value in the input radar-beacon report. The confidence of a beacon report defaults to 7.) The following empirically-derived table gives the original alpha values as a function of quality and confidence:

Confidence Quality	0	1	2	3	4	5	6	7
0	0.6	0.6	0.6	0.6	0.6	0.6	0.6	1.0
1	0.6	0.6	0.6	0.9	0.9	0.9	0.9	1.0
2	0.6	0.6	0.6	1.0	0.9	1.0	1.0	1.0
3	0.6	0.6	0.6	1.0	0.9	1.0	1.0	1.0
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

The following empirically-derived table gives the original beta value as a function of quality and confidence:

Confidence Quality	0	1	2	3	4	5	6	7
0	0.3	0.3	0.3	0.3	0.3	0.3	0.3	1.0
1	0.3	0.3	0.3	0.7	0.6	0.7	0.7	1.0
2	0.3	0.3	0.3	0.9	0.6	0.9	0.9	1.0
3	0.3	0.3	0.3	0.9	0.6	0.9	0.9	1.0
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

(__NOTE__: experiments conducted during the testing of the 9-PAC Phase II tracker at Salt Lake City suggested that the alpha-beta smoothing did not allow sufficient tracker flexibility to follow high-rate military aircraft maneuvers in radar-only operation. To provide maximum tracker responsiveness, all the alpha and beta values were changed to 1.0 – removing all smoothing from the tracker and reverting the algorithm to a two-point interpolator.)

If the "azimuth degrade" flag is set in the correlating radar report (indicating a suspect report position), the alpha value defaults to 0.6 and the beta value defaults to 0.3, regardless of the report's quality and confidence. The default alpha-beta values are also used if the correlation procedure was many-on-many (i.e. multiple tracks sharing multiple report associations). The beta value for a beacon report is reduced from 1.0 to 0.7 if the radar is not in Mode S mode and the runlength of the beacon report is less than 15. (These extra rules relax the tracker so that "questionable" radar or beacon reports do not cause the track to deviate too much.)

If the correlating report is beacon-only, the track counts a beacon miss -- otherwise the beacon miss count in the track is reset to zero. Similarly, if the correlating report is radar-only, the track counts a radar miss -- otherwise the radar miss count in the track is reset to zero. The track miss counts are used in association testing as follows:

(a) A track with > 'n' radar misses cannot associate with a radar-only report

(b) A track with > 'n' beacon misses cannot associate with a beacon-only report

where 'n' is a VSP parameter (NSC_SCANS). The track's updated radar quality is set to be the quality of the correlating report (defaults to 4 for a beacon-only report). Similarly, the track's updated radar confidence is set to be the confidence of the correlating report (defaults to 8 for a beacon-only report).

(__NOTE__: as a result of some 9-PAC Phase II testing, the beacon and radar miss-count VSPs have been set to 300, effectively disabling the miss -count association tests.)

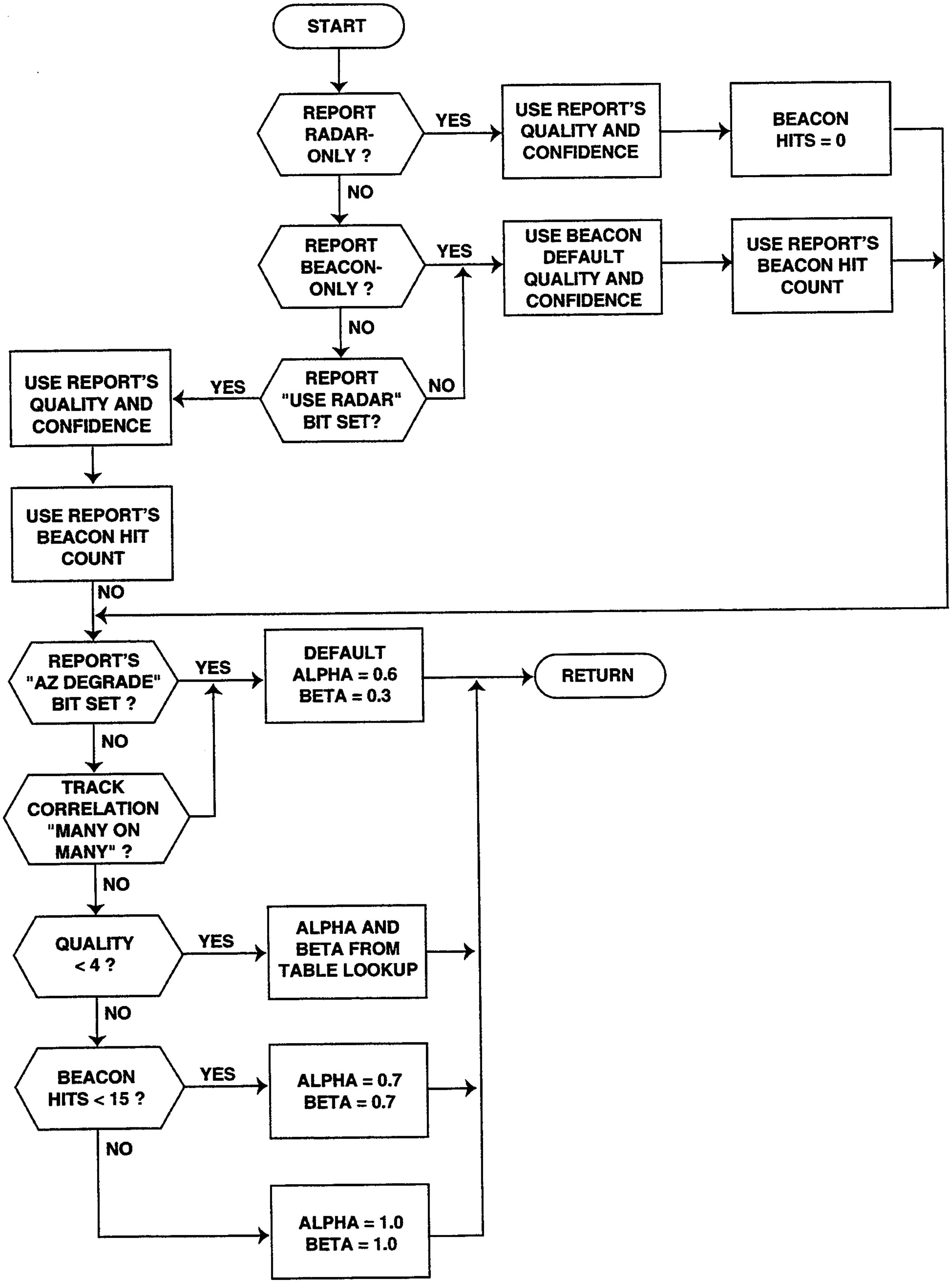


Figure 20. Determine Gains.

5.3.3.3 Update Predictions

Called by:	Track Update
Calls:	
Purpose:	Given filter gains α, β – update the track predictions.

The track prediction is updated to the next scan's update time based on previous track history. The time until the sensor antenna will next see the aircraft (denoted T) is nominally the sensor scan time VSP. However, to compute a more-accurate value for T when the aircraft is close to the sensor, the following approximation is used (derived from Reference [7]). Let 'az_{dot}' denote the track's current azimuthal velocity in radians/scan units. Assuming that azdot is constant over the update period:

$$T = 1 / (1 - az_{dot} / 2\pi)$$

Note that this approximation (assuming a constant value of az_{dot}) can break down for worst-case maneuvering aircraft very close to the sensor (within 1 nautical mile). However, computing the exact value of T requires a highly time-consuming iterative process. The 9-PAC Phase II Tracker uses the approximation in all cases. (Note: if $az_{dot}=2\pi$ radians/scan, default T=1 scan.)

The 9-PAC Phase II Tracker uses a simple alpha-beta smoother in Cartesian coordinates to smooth and predict track positions and velocities for beacon or radar-reinforced beacon tracks. A comparison was made to a Kalman filter tracker, but the alpha-beta tracker with variable gains proved to be more robust and less computationally intensive than the Kalman filter. The Kalman filter was a better long-term predictor – however, the 9-PAC Phase II Tracker is a scan-scan correlator and is only looking one scan ahead. The improved long-term predictions were not a significant benefit. Note that predicted values for beacon track position and velocity in range-azimuth coordinates are derived from the predicted Cartesian coordinates.

The alpha-beta filter gains are determined as described in Section 5.3.3.2 above. The beacon track predictions and velocities are updated as shown below. (Note: only the x-coordinate update equations are shown – the y-coordinate equations are equivalent.) The track's "firmness" (number of scans between the last track update and the current time) is denoted 'f'.

$$\Delta x = x_{predicted} - x_{report}$$

$$V_x = V_x - (\beta / fT)\Delta x$$

$$x_{smooth} = x_{predicted} - \alpha\Delta x$$

$$x_{predicted} = x_{smooth} + TV_x$$

If the correlating report is radar-only and the beacon (or radar-reinforced beacon) track has a validated altitude, the track's altitude is used to re-compute the radar report's x-y coordinates prior to prediction. The beacon (or radar-reinforced beacon) track's predicted range and azimuth values are computed from the x-y predictions and the track's altitude. Values for the track's radial and azimuthal velocity components are computed as described in Section 5.3.3.4 below.

The first step in the 9-PAC Phase II radar track update process is a "forgiveness-fixup" check that the most-recent correlation to this track was correct. This check is a form of linearity test, intended to repair a prior mistaken correlation to this track. A straight line is interpolated between the position of the report that was the second-to-the-most-recent correlation to this track and the radar report that is correlated to this track on the current scan. The position on this line at the scan of the most-recent correlation report is determined. A range-azimuth box is centered on this calculated position. The box is sized using parametric VSP multipliers of sensor measurement standard deviations (plus or minus in range and azimuth). If the actual position of the most-recent correlating report to the track lies within the box, then the correlation is considered correct. If the actual position of the most-recent correlating report lies outside the box, then the preceding correlation is declared as an error. The effects of the erroneous correlation are "fixed" by re-performing the track update as though the previous correlation did not occur and the track was actually coasted on the most-recent correlation scan. The track predicted position is recomputed in range and azimuth, as well as resetting the track's velocity components in range and azimuth. (Note: this check works in concert with the radar association linearity test (Section 5.3.1.1 above). Normally, the radar association linearity test would only permit associations to the track that would also pass the "forgiveness-fixup" check. However, there are special cases in the radar association linearity test to track maneuvering aircraft and to deal with the significant degree of azimuthal error that can occur with radar reports. Since these special cases may occasionally allow faulty correlations to occur, the "forgiveness-fixup" test here has an opportunity to undo their impact on the track.) If the track has been re-predicted, a flag in the track is set to disable the track maintenance linearity check on this track for the subsequent two scans. This allows the track to stabilize after the fixup is performed.

The 9-PAC Phase II Tracker uses an extended alpha-beta smoother in range-azimuth coordinates to smooth and predicted track positions and velocities for radar-only tracks. Since radar-only tracks have no altitude information, it is not possible to derive accurate Cartesian coordinates for them, especially when close to the sensor. The radar measurements are made directly in range and azimuth coordinates. However, range-azimuth tracking has problems with distortion of the tracks when within a few miles of the sensor tracks (where changes in range have an impact on the azimuthal velocity). To compensate for this distortion, an extended smoothing algorithm from Reference [7] employing additional velocity (denoted 'dot') and acceleration (denoted 'double-dot') terms is used. The following equations define the 9-PAC Phase II Tracker radar-only track smoothing/prediction process. The track's "firmness" (number of scans between the last track update and the current time) is denoted 'f'.

$$\Delta \text{range} = \text{range}_{\text{predicted}} - \text{range}_{\text{report}}$$

$$\Delta \text{azimuth} = \text{azimuth}_{\text{predicted}} - \text{azimuth}_{\text{report}}$$

$$\text{range}_{\text{smoothed}} = \text{range}_{\text{predicted}} - \alpha \Delta \text{range}$$

$$\text{azimuth}_{\text{smoothed}} = \text{azimuth}_{\text{predicted}} - \alpha \Delta \text{azimuth}$$

$$V_{\text{radial}} = V_{\text{radial}} - (\beta / fT) \Delta \text{range}$$

$$V_{\text{azimuthal}} = \text{az}_{\text{dot}} - (\beta / fT) \Delta \text{azimuth}$$

$\text{range}_{\text{dot}} = T\ V_{\text{radial}}$

$\text{range}_{\text{double-dot}} = \text{range}_{\text{smoothed}} * V_{\text{azimuthal}}^2 * T^2 / 2$

$\text{range}_{\text{predicted}} = \text{range}_{\text{smoothed}} + \text{range}_{\text{dot}} + \text{range}_{\text{double-dot}}$

$\text{azimuth}_{\text{dot}} = T\ V_{\text{azimuthal}}$

$\text{azimuth}_{\text{double-dot}} = - V_{\text{radial}} * V_{\text{azimuthal}} * T^2 / \text{range}_{\text{smoothed}}$

$\text{azimuth}_{\text{predicted}} = \text{azimuth}_{\text{smoothed}} + \text{azimuth}_{\text{dot}} + \text{azimuth}_{\text{double-dot}}$

(Note that if the track's smoothed range is zero, the azimuth "double-dot" term is defaulted to zero.) The radar-only track's Cartesian velocities are beta-smoothed using the same equations that were used for beacon tracks. The radar-only track's Cartesian predicted positions are derived from the predicted range and azimuth. The track entry is maintained in both coordinate systems (x-y and range-azimuth) in order to be ready for correlation to either a radar or beacon report on the next scan. The value of range-rate and azimuth-rate prior to smoothing on this scan is saved in the track where it may be used by the "forgiveness-fixup" check on the next scan.

Two heading estimates are maintained for each track. A smoothed heading is computed from the track's smoothed Cartesian velocities V_x and V_y. This heading estimate is further smoothed by the following equation (if the track has had at least 5 updates):

$\text{Heading}_{\text{smoothed}} = (0.8 * \text{Old\_Heading}_{\text{smoothed}}) + (0.2 * \text{New\_Heading}_{\text{smoothed}})$

If the track is new (less than 5 updates), then the equation below is used instead:

$\text{Heading}_{\text{smoothed}} = (\text{Old\_Heading}_{\text{smoothed}} + \text{New \_Heading}_{\text{smoothed}}) / 2$

The track's instantaneous (unsmoothed) heading is computed by converting the previous track update range and azimuth values to Cartesian x-y and then computing the difference is x-y between the correlating report's x-y position and the previous values. The track maintains the last instantaneous heading estimate as well as the current instantaneous heading estimate.

A histogram of slow (track speed ≤ the VSP "SLOW_VEL"), radar-only tracks by smoothed heading is maintained by the tracker. (Note: for this algorithm, "radar-only" tracks are those that have never been a type other than RADAR_ONLY during their lifetime.) Histogram heading bins are 15 degrees in extent. An "excessive" population of tracks in a given bin tend to indicate the presence of "bird" false tracks. If the histogram "triad" bin count (the sum of the histogram bin at the track's heading and its adjoining neighbors – a total of 45 degrees in extent) for a slow, radar-only track at a given heading exceeds the VSP HDG_TRK_CNT, then a feedback flag is set in the correlating report. This feedback flag is used by the C&I front end to modify its thresholds to reduce the likelihood of "bird" false targets getting through to the tracker. The neighboring histogram triads (plus or minus 15 degrees) are also tested – if either exceeds the threshold VSP, then the feedback flag is set.

Once each scan, the heading histogram values are checked in order to update the value of a global "birdHeadingFlag" that is used by the C&I front-end processing to alter its thresholds based on the tracker's "perception" of the extent of "bird" false tracks in the system. If any heading bin "triad" (center 15-degree bin plus its adjoining neighbor bins) has a total track count that equals or exceeds the VSP "TRK_HDG_ON_CNT", the flag is set. If no heading bin "triad" has a total track count that exceeds the VSP "TRK_HDG_OFF_CNT", then the flag is cleared. Note: the flag is initialized FALSE at system start-up or when VSPs are changed.

At each track update of a slow (track speed ≤ the VSP "SLOW_VEL"), radar-only track, the difference between the current instantaneous track heading and the previous instantaneous track heading is computed. Dramatic changes in heading tend to indicate that this track might be a false "bird" track rather than a real aircraft. A counter is maintained in each radar-only track indicating how many consecutive times the instantaneous heading change exceeded the VSP HDG_SWITCH. The value of this counter is fed back to the C&I front end where it may be used to modify processing thresholds.

This process is shown in Figure 21.

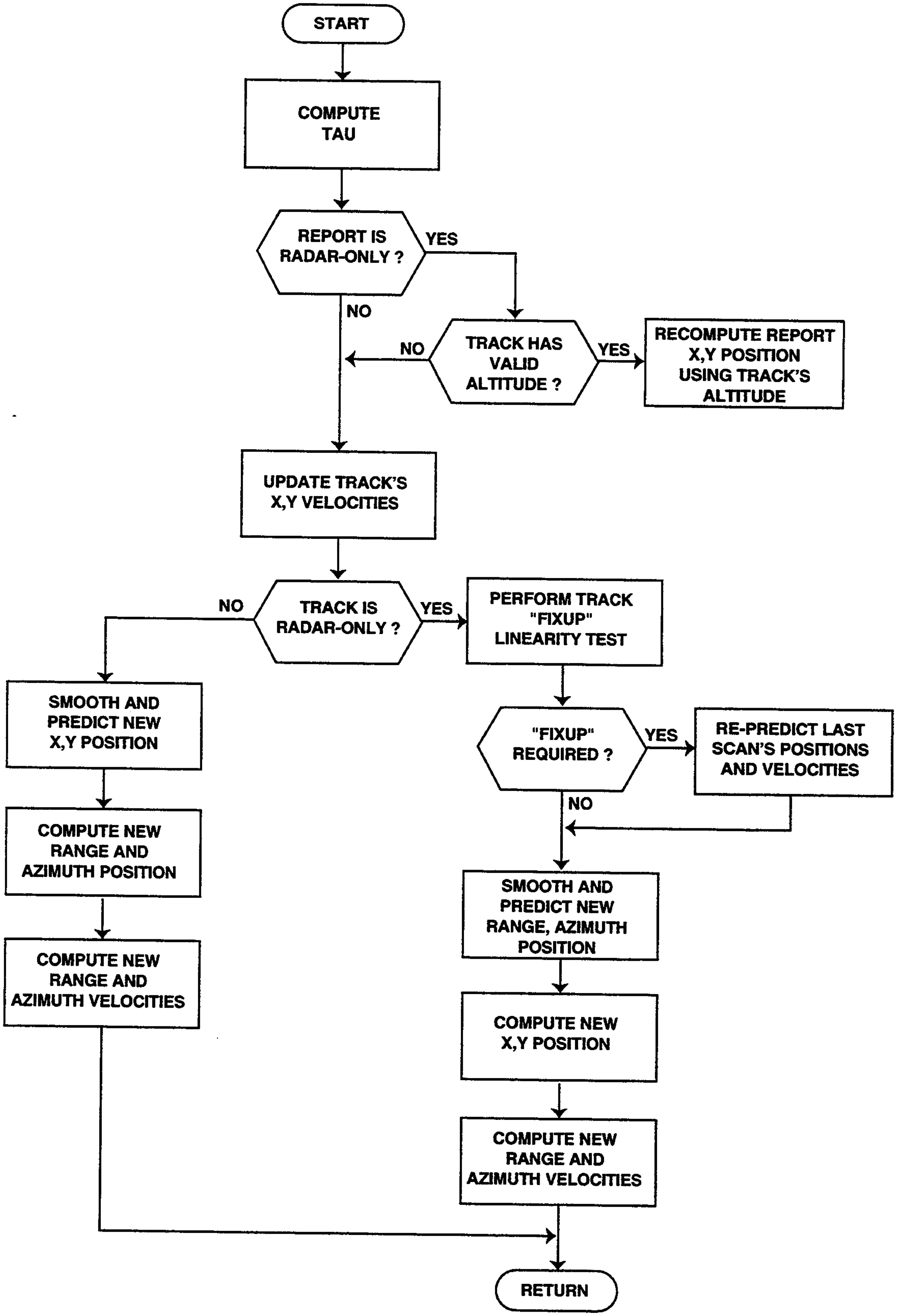


Figure 21. Update Predictions.

5.3.3.4 Compute Velocity

Called by:	Track Update
Calls:	
Purpose:	Determine various track velocities and smooth them.

This routine computes the instantaneous radial and azimuthal velocities for non-radar tracks. (Beacon tracking is done in x-y coordinates, rather than range-azimuth. The calculation of the radial and azimuthal velocities for radar-only tracks is done in the Update Predictions function described in Section 5.3.3.3 above.) The track's instantaneous ground speed is computed. The average radial velocity and average ground-speed are smoothed over time with a simple low pass filter process.

5.3.3.5 Set Track Type

Called by:	Track Update
Calls:	
Purpose:	Updates the track type, e.g., beacon-only, radar-only, radar-reinforced, given the report type used for the track update. Updates the discrete code for beacon tracks. Also updates the track's altitude fields.

The Tracker monitors the basic type of each track: beacon-only, radar-only, and radar-reinforced. It is necessary to know the track type because the track type is compared to the report type when computing the association measure score. And track types do change for various reasons: faulty transponders, blocking of the transponder antenna, and environmental issues. It is also possible for tracks to change their Mode 3/A code and to change from discrete to non-discrete and vice versa. All of these changes are monitored and recorded by this function.

To allow all tracks to be treated similarly, all radar-only tracks and reports are assigned a Mode 3/A code of 0000. If a track is updated by a report with a different Mode 3/A code, a counter is incremented. If this happens three updates in a row, the track's Mode 3/A code is set to match the last report's Mode 3/A code and the track type is set to match the report type. If the track and report code match on any update, the counter is reset to zero. This algorithm detects and handles all possible transitions: radar to beacon and vice versa, discrete to non-discrete and vice versa, and change of discrete code.

In addition to monitoring the Mode 3/A code, it is also necessary to monitor and update the track's Mode C altitude code and validity information. Again, as was the case for Mode 3/A codes, all radar-only tracks and reports are assigned an altitude of 0 with low validity. If the correlating track is a beacon-only or radar-reinforced track, it is likely to have altitude information. If the correlating report for this update has altitude data, then the report's altitude information is copied into the track and the track's "altitude change counter" is reset to zero. Otherwise, if the correlating track is updated by a report that is either radar-only or lacking in altitude information, the "altitude change counter" is incremented. If this happens three times in a row, the track's altitude information is defaulted to altitude 0 with low validity and the counter is reset. This algorithm preserves the track altitude data through short periods when beacon data is missing and the track is being maintained with radar reports.

This process is shown in Figure 22.

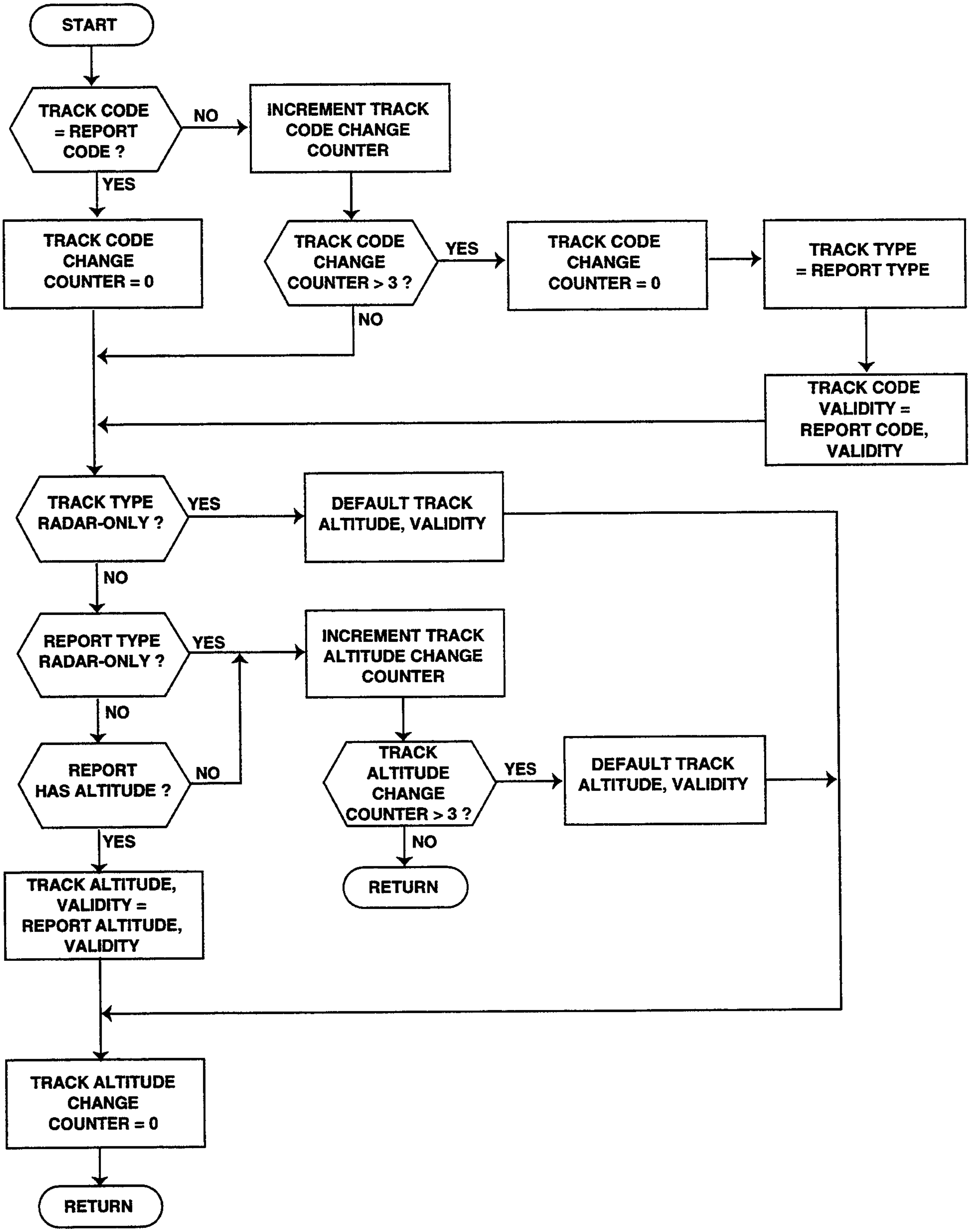


Figure 22. Set Track Type.

5.3.3.6 Minimum Distance Test

Called by:	Track Update
Calls:	
Purpose:	Determines if a track has passed a minimum distance criteria. This test effects the track's path through the state diagram.

The purpose of the minimum distance test is to reduce false tracks due to radar-only clutter. The most serious breakthrough clutter tends to occur close to the sensor. For this reason, the minimum distance test is not applied to tracks beyond a certain distance (VSP dependent, typically 20 nmi) or to beacon tracks.

For processing purposes, the first check made is to see if the test has already been passed which is indicated by a flag. If the flag is set, the test is complete. Tracks which are not subject to the test, automatically have this flag set.

The test is applied in Cartesian coordinates and independently for the x and y dimensions. If the minimum distance is moved in either dimension, the test is passed. The size of the distance is determined by a VSP (typically 0.25 nmi). If the test is not passed, the flag is not set. The result of the test does not affect the report-track correlation. Its only effect is on the path through the state diagram.

This process is shown in Figure 23.

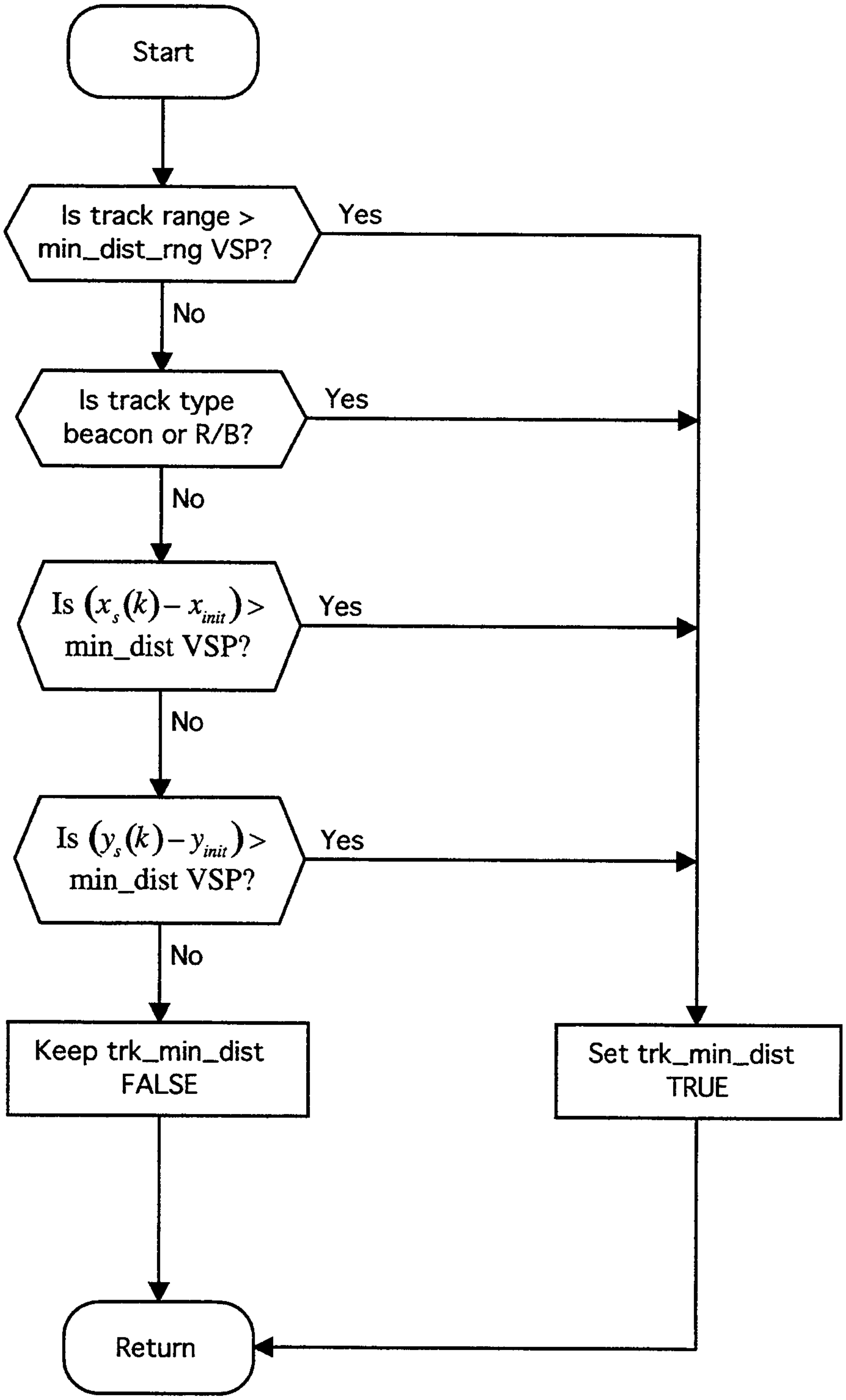
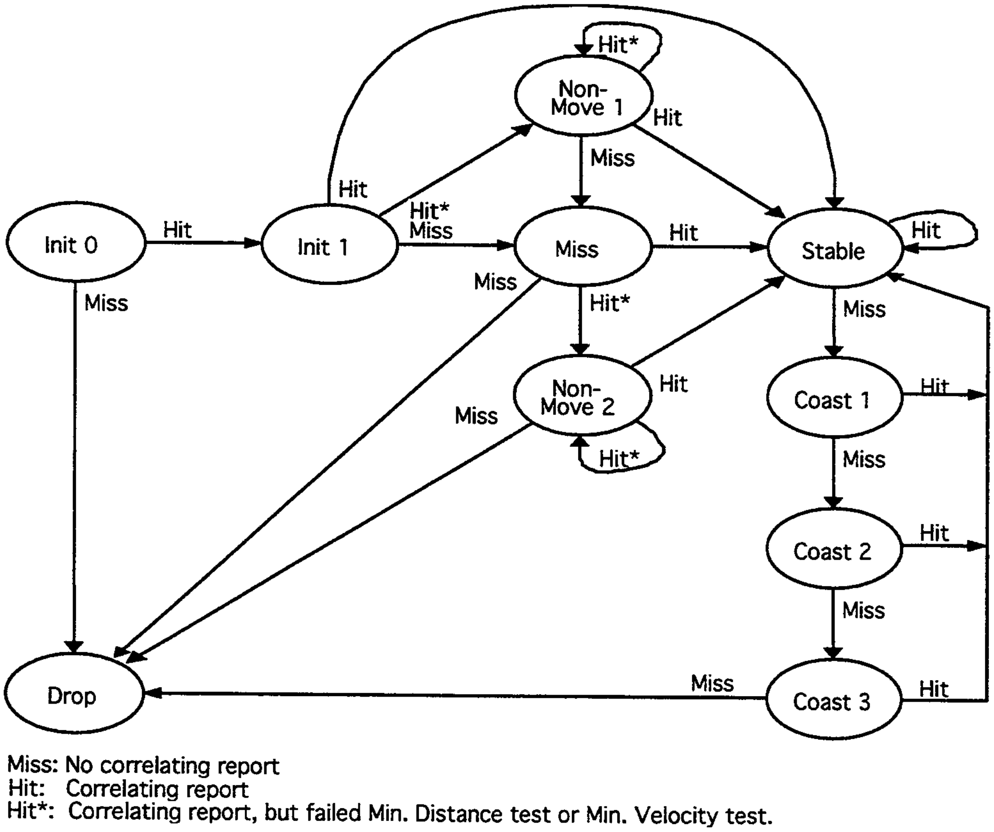


Figure 23. Minimum distance test.

5.3.3.7 Update Track State

Called by:	Track Update
Calls:	
Purpose:	Guides the Tracker state through the state diagram. The Tracker state affects association and update decisions.

The Tracker state diagram is shown in Figure 24. The track state is updated according to the diagram. The decision branches in the diagram are based on the following categories: hit or miss, minimum distance pass or fail, and minimum velocity pass or fail.



Miss: No correlating report
Hit: Correlating report
Hit*: Correlating report, but failed Min. Distance test or Min. Velocity test.

Figure 24. State diagram.

There is one exception to the Tracker state diagram, called into play when the most-recent altitude update on the track had a high-validity altitude level within 200 feet (2 flight levels) of the ground (as determined by the sensor HEIGHT VSP). Tracks with such low altitudes are candidates for the "on the airport" tests. The sensor VSP table may define up to 8 "airport" regions. If the track is predicted to lie within any one of these regions (as defined by minimum and maximum range-azimuth region VSPs), then only one track coast is permitted before dropping the track. (In this case, states COAST2 and COAST3 are bypassed.) This algorithm was put in place to deal with the case that a beacon-equipped aircraft lands and its transponder is disabled. The aircraft track may subsequently be maintained with radar reports as it taxis across the airport. There may be sufficient clutter reports in the area to keep the track "alive" long after the aircraft has landed. By limiting the track to a single coast in this case, the track will drop more quickly once the aircraft has landed, rather than wandering about the airport surface. This process is shown in Figure 25.

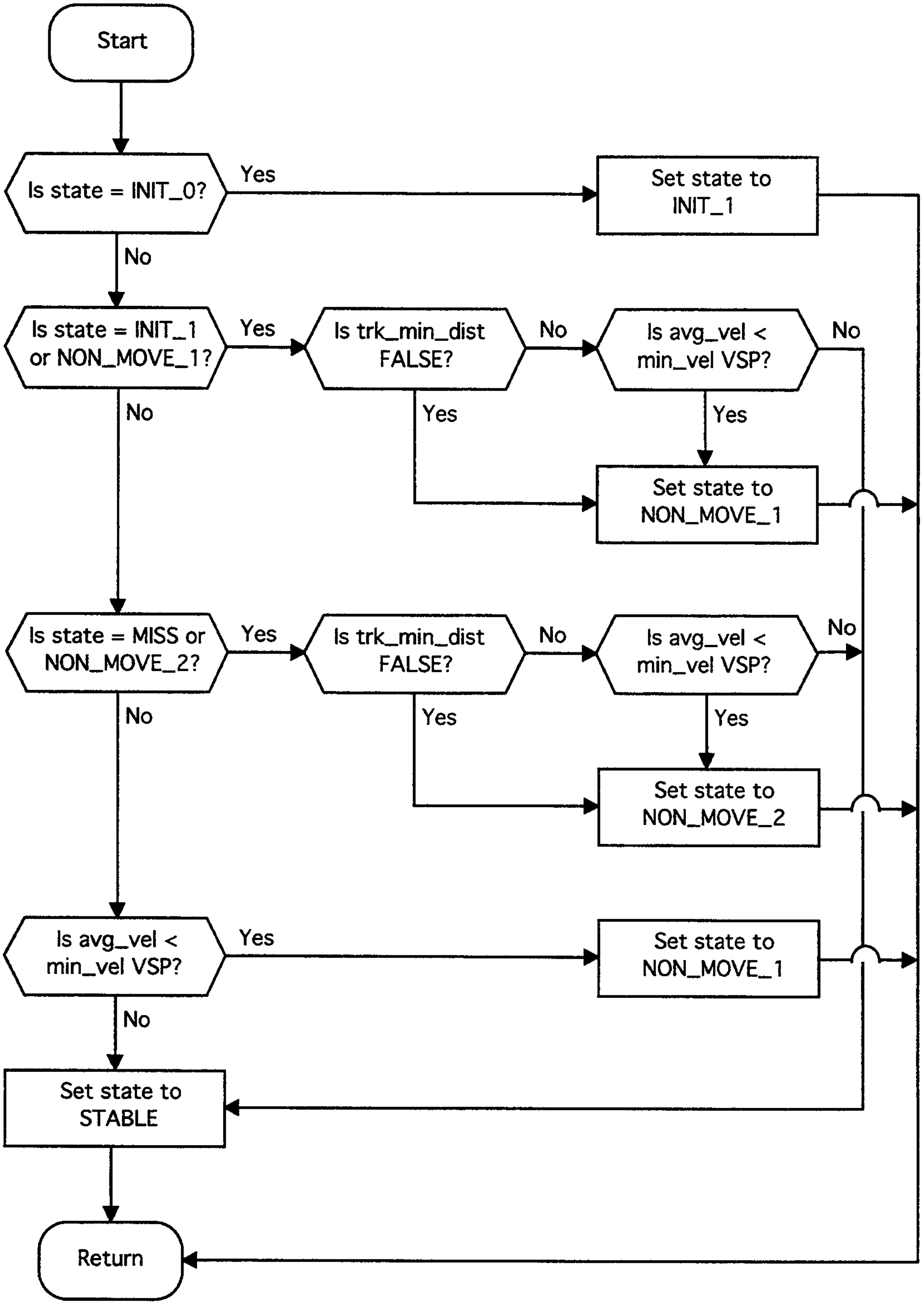


Figure 25. Update track state.

5.3.3.8 Compute Association Boxes

Called by:	Track Update
Calls:	
Purpose:	Set the limits for the association boxes.

The association boxes define the geographic region where candidate reports for correlation will be found. The size of the association boxes is dependent on the track state. The association boxes are drawn in the radar system coordinates since the radar system errors occur in the system coordinates. There are two association boxes for each track: a zone 0 (inner) box corresponding to the system errors, and a zone 1 (outer) box corresponding to the system errors of zone 0 plus anticipated target maneuvering. For association purposes as discussed in Associate Tracks, a report need only to be in zone 1 to associate with the track. When correlation is performed with otherwise identical reports in zone 0 and zone 1, the report in zone 0 is given preference.

Two association boxes in range-azimuth coordinates are built around each track's predicted position as part of track update (or track coast). The first (inner) box is sized assuming that the track is actually going straight and only measurement errors need be considered. The second (outer) box is sized assuming that not only must measurement error be considered but also that the track has begun a turn. (Note that the derivation for these association box sizes is taken from Reference [7].)

The number of scans elapsed between the latest update on the track and the predicted scan (called the track's "firmness" and denoted 'f') is one factor in the calculation of 9-PAC Phase II Tracker association box sizes. If the track has not coasted this scan, f=1. The number of scans elapsed between the latest update on the track and the next previous update scan (called the track's "history firmness" and denoted 'g') is another factor in the calculation of 9-PAC Phase II Tracker association box sizes.

<u>Zone 0 (inner) Association Box</u>

The inner association box size for range is calculated by multiplying the VSP range-sigma value by the box-one multiplier given by the equation:

$$m0 = 1 + (2*f)/g$$

This multiplier accounts for the increase in track positional uncertainty due to coasts. The box edges (in range) are given by the predicted range + and - the box size (except that the inner edge cannot be negative).

The inner association box size computation for azimuth is a bit more complex. The VSP range-sigma value is converted to an angle using the track's predicted range and assuming that the range error is completely tangential (a worst-case assumption). The larger of this angle and the VSP azimuth-sigma value is multiplied by the box-one multiplier 'm0' to yield the inner association box azimuth size. The box edges (in azimuth) are given by the predicted azimuth + and - the box size. Since this size can get very large for small track ranges, the box size is limited to at most one-half of a scan.

Zone 1 (outer) Association Box

The outer association box size (for maneuvering tracks) multiplier is given by the equation:

$$m1 = (f^2 + f*g)/2$$

The assumption is that the track actually began a turn at the time of the last track update. The multiplier accounts for the effects of coasting on the track positional uncertainty. An acceleration (a turn is an acceleration) term is derived from the VSP number of g-factors parameter. An aircraft performing a standard-rate turn (3 degrees/second) is pulling 1 'g'. Small aircraft (especially military) can pull several 'g's in maneuvers. The acceleration factor is converted into miles/scan^2 units. The "turn-term" is computed as the product of 'm1' and the acceleration term. Note that this turning factor is added to the measurement-error factor derived as "zone 0" above.

The size of the outer association box (in range) is given by the sum of the inner box size (in range) and the maximum of the "turn-term" or one-half of the inner box size (in range). The box edges (in range) are given by the predicted range + and - the box size (except that the inner edge cannot be negative).

The outer box size (in azimuth) is computed in a manner similar to the inner box size (in azimuth). The "turn-term" is converted to an angle using the track's predicted range and assuming that the range error is completely tangential. Set "amax" to the larger of the "turn-term" angle and one-half of the track's inner box size (in azimuth). The track's outer box size (in azimuth) is the sum of the track's inner box size (in azimuth) plus "amax." The box edges (in azimuth) are given by the predicted azimuth + and - the box size. Since this size can get very large for small ranges, the box size is limited to at most one-half of a scan.

5.3.3.9 *Track Coast*

Called by:	Update Tracks
Calls:	Compute Association Boxes
Purpose:	Update the track prediction and perform bookkeeping when a track is coasted due to a lack of a satisfactory report for updating.

If no satisfactory report is found to update a track, the track is coasted. The track state is updated according to the state diagram shown in Section 5.3.3.7 above. If the track has coasted for three scans (state = COAST3), or is just initiating (state = INIT_0), or has missed for a second time without becoming stable (state = MISS or state = NON_MOVE_2), the track is dropped. The track is removed from the Active and General track lists and the allocated memory is returned to the stack. If the track is initiating, has received at least two points, but has not become stable whether due to a lack of points or a lack of distance moved, the track state is set to MISS. Finally, if the track has been stable at anytime, the track state is incremented through the coasting states allowing for three misses before the track is dropped.

The track prediction is updated to the next scan's update time based on previous track history. The time until the sensor antenna will next see the aircraft (denoted T) is nominally the sensor scan time VSP. However, to compute a more-accurate value for T when the aircraft is close to the sensor, the following approximation is used (derived from Reference [7]). Let 'az_{dot}' denote the track's current azimuthal velocity in radians/scan units. Assuming that azdot is constant over the update period:

$$T = 1 / (1 - az_{dot} / 2\pi)$$

Note that this approximation (assuming a constant value of az_{dot}) can break down for worst-case maneuvering aircraft very close to the sensor (within 1 nautical mile). However, computing the exact value of T requires a highly time-consuming iterative process. The 9-PAC Phase II Tracker uses the approximation in all cases. (Note: if $az_{dot}=2\pi$ radians/scan, default T=1 scan.)

The track prediction is the previous prediction plus the estimated velocity. If the track is beacon or radar-reinforced beacon, the prediction ahead to time 'T' is done linearly in Cartesian x-y coordinates. The track's ground-range and azimuth for the new time is computed from the predicted x-y values. The track's predicted range is then computed using the ground-range and the track's last known altitude. If the track is radar-only, the prediction ahead to time 'T' is done linearly in range-azimuth coordinates (since radar-only tracks lack altitude and cannot compute an accurate x-y position). The track's predicted x-y position for the new time is computed from the predicted range-azimuth values.

A histogram of slow (track speed $\leq$ the VSP "SLOW_VEL"), radar-only tracks by smoothed heading is maintained by the tracker. (Note: for this algorithm, "radar-only" tracks are those that have never been a type other than RADAR_ONLY during their lifetime.) Histogram heading bins are 15 degrees in extent. An "excessive" population of tracks in a given bin tend to indicate the presence of "bird" false tracks. If the histogram "triad" bin count (the sum of the histogram bin at the track's heading and its adjoining neighbors – a total of 45 degrees in extent) for a slow, radar-only track at a given heading exceeds the VSP HDG_TRK_CNT, then a feedback flag is set. This feedback flag is used by the C&I front end to modify its thresholds to reduce the likelihood of "bird" false targets getting through to the tracker. The neighboring histogram triads (plus or minus 15 degrees) are also tested – if either exceeds the threshold VSP, then the feedback flag is set.

The track's association boxes for the new scan are computed from its predicted position as described in Section 5.3.3.8 above. Various track flags and counters are reset for the next scan's processing. The track's "points/scan" 5-scan average is updated (a coast has 0 point value). If the "points/scan" value for the track now drops below the minimum track maintenance VSP (either the normal or slow-speed VSP, depending on the track's ground-speed), the track is dropped – the track is removed from the Active and General track lists and the allocated memory is returned to the stack. The final step in the coast process removes the track from the Active Track list and places it on the azimuth-ordered General Track list. This process is shown in Figure 26.

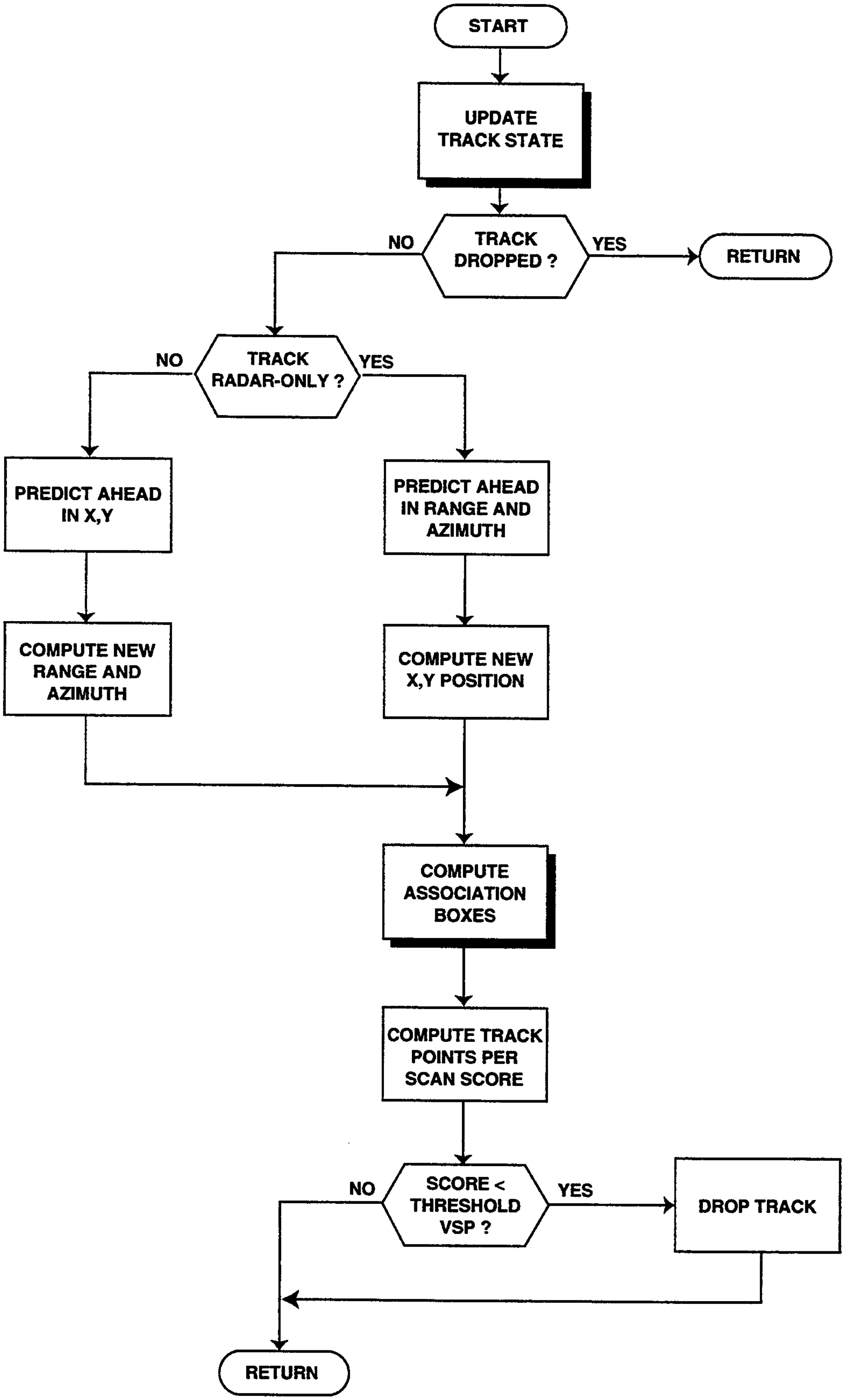


Figure 26. Track Coast.

5.3.4 Initiate Tracks

Called by:	Track
Calls:	Beacon Initiation, Radar Initiation, Compute Association Boxes
Purpose:	Create new tracks from previous uncorrelated reports.

The processing of reports that fail to correlate with any existing tracks is handled in this function. There are separate track initiation algorithms for radar-only reports (Section 5.3.4.1) and beacon reports (Section 5.3.4.2). Separate lists of uncorrelated reports from previous scans are maintained for radar-only and beacon. Two scan lists of uncorrelated beacon data (current scan and previous scan) are maintained for beacon track initiation. Five scan lists of uncorrelated radar-only data (current scan and four previous scans) are maintained for radar track initiation. These lists are linked into lists by range cells (5 nautical miles in extent) and azimuth sectors to facilitate rapid searching by report position.

If the uncorrelated report is radar-only, the first step in track initiation processing is to check for the radar test-target MTI indication in the report. The first 10 track numbers in the 9-PAC Phase II Tracker are reserved for potential MTI tracks. (Note: MTI "tracks" are actually fixed radar reflectors that do not move.) If the reported range and azimuth of an uncorrelated radar-only MTI report fails to match (within the sensor's measurement error VSPs) any of the current MTI track locations, a new MTI track is assigned to the report position, and the report is marked as "correlated" with the new MTI track number assigned. Otherwise, if the reported range and azimuth of an uncorrelated radar-only MTI report matches one of the current MTI track locations, the report is marked as "correlated" with the existing MTI track number.

Track initiation processing of an uncorrelated radar-only report that is not flagged as MTI is done as described in Section 5.3.4.2 below. A new track number is assigned for each radar-only track initiated, and numerous other bookkeeping fields are initialized at this time, using the input uncorrelated radar-only report and selected uncorrelated radar-only report(s) from previous scan(s). After all the initiating radar-only track data fields have been initialized, Compute Association Boxes (Section 5.3.3.8) is called to define where the next report to update the track is most likely to be found. After this is complete, the new track is entered in the azimuth-ordered General Track list.

Uncorrelated radar-only reports that fail to initiate a track as described in Section 5.3.4.2 below are fed back from the 9-PAC Phase II Tracker to the 9-PAC C&I task. The C&I task utilizes this feedback to assist in maintaining its radar clutter maps (i.e. if the Tracker ignored the report as clutter, the C&I task can have more assurance that the report truly was clutter).

Track initiation processing of an uncorrelated beacon-only report is done as described in section 5.3.4.1 below. A new track number is assigned for each beacon track initiated, and numerous other bookkeeping fields are initialized at this time, using the input uncorrelated beacon report and a selected previous-scan uncorrelated beacon report. After all the initiating beacon track fields have been initialized, Compute Association Boxes (Section 5.3.3.8) is called to define where the next report to update the track is most likely to be found. After this is complete, the new track is entered in the azimuth-ordered General Track list.

Track initiation processing of an uncorrelated radar-reinforced beacon report is done as a sort of hybrid case of both radar-only and beacon-only. The radar-reinforced report will be treated for the purposed of track initiation as though it were both a radar-only and a beacon-only.

A cross-referencing linkage ties both "instances" of the report together. The report is first processed as though it were beacon-only, using the algorithm in Section 5.3.4.1 below. If this beacon-only processing results in the initiation of one or more new beacon-only tracks, no further track initiation processing is done with this report. If the beacon-only processing does not result in the initiation of new tracks, then radar-only track initiation processing is attempted as described in Section 5.3.4.2 below. Note that if this report is used on this or a later scan for track initiation (either beacon or radar), then it is disqualified from further consideration as its "alternative" form. (Note: this hybrid algorithm is employed to get earlier track initiation for aircraft on take off. This mechanism allows for earlier track initiation when aircraft take off (the beacon transponder is shut off when the aircraft is on the ground) and there may not be sufficient radar coverage to allow track initiation by radar-only processing. The track initiation process is shown in Figure 27.

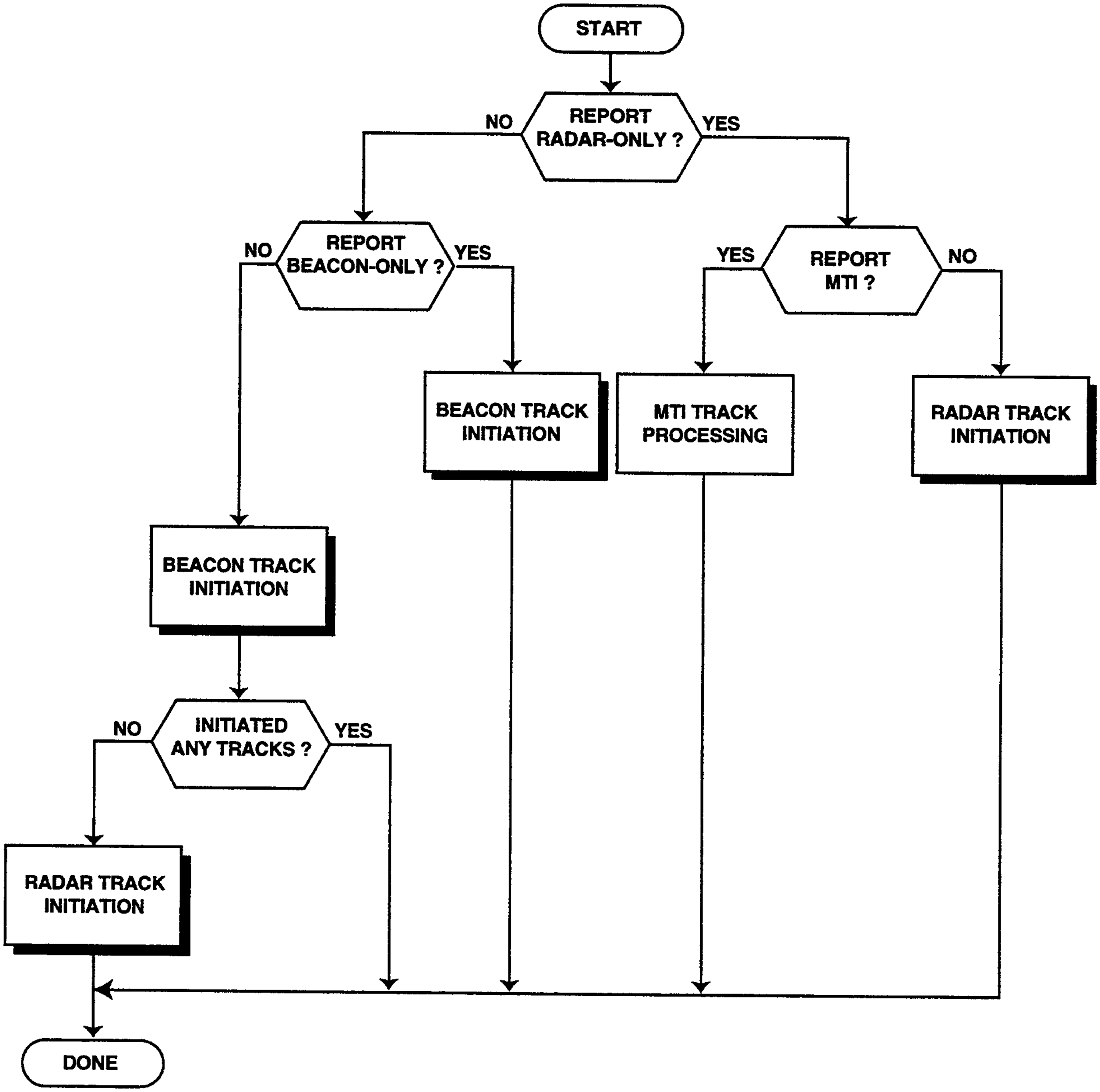


Figure 27. Track Initiation.

5.3.4.1 Beacon Initiation

Called by:	Track
Calls:	
Purpose:	Create new tracks from previous uncorrelated beacon reports.

Each uncorrelated beacon report from the current scan is compared against uncorrelated beacon reports from the previous scan. In order to be a candidate prior-scan report for beacon track initiation with the current-scan report, the uncorrelated beacon report must not have been already used and there must be a satisfactory positional match with the current-scan report. Each of the following tests must be passed:

a) The prior-scan uncorrelated beacon report has not been used to initiate a previous beacon track with another current-scan uncorrelated report.

b) The absolute value of the difference in range between the current-scan and previous-scan reports is less than or equal to the distance that an aircraft flying at the VSP maximum aircraft speed (nominally 600 knots) would travel in one sensor scan. (Note: linking the previous-scan uncorrelated reports into range-ordered lists is used to reduce the amount of report searching required to perform this test. Each range list defines a range band 5 nautical miles in extent. The report search includes the range list corresponding to the range of the current-scan uncorrelated beacon report and the adjacent two range lists.)

c) The absolute difference in azimuth between the current-scan and previous-scan reports is less than or equal to the angular distance that an aircraft flying at the VSP maximum aircraft speed (nominally 600 knots) would travel in one scan. This is a function of the aircraft range. (Note: linking the previous-scan uncorrelated reports into azimuth-sector-ordered lists is used to reduce the amount of report searching required to perform this test. There are 32 azimuth-sector lists. The report search includes the azimuth-sector list corresponding to the azimuth of the current-scan uncorrelated beacon report and the adjacent two or more azimuth-sector lists.)

d) The current-scan beacon report must not have a Mode 3/A code of zero with validity equal to 0.

e) The current-scan beacon report must have either Mode 3/A code validity greater than zero or Mode C altitude validity greater than 0.

The set of previous-scan uncorrelated beacon reports selected by the above tests is further subdivided into a set of "zones" based on Mode 3/A code and Mode C altitude tests. Zone 0 is defined to be the best code/altitude match, while zone 4 is defined to be a worst-case mismatch. A "high validity" code is defined to have a validity value equal to 3. A "low validity" code has a validity value less than 3. The zone scoring for Mode 3/A codes is defined by the following sequence of tests:

a) If neither the current-scan report nor the previous-scan report Mode 3/A code has high validity, then the Mode 3/A zone is set to 1.

b) If test (a) failed (i.e. either of the report Mode 3/A codes has high validity) and the number of bit differences between the Mode 3/A codes is less than or equal to 1, then the Mode 3/A zone is set to 0.

c) If either the current-scan report or the previous-scan report Mode 3/A code has high validity (but not both) and the number of bit differences between the Mode 3/A codes is greater than 1, then the Mode 3/A zone is set to 2.

d) If both the current-scan report and the previous scan report Mode 3/A codes have high validity and the number of bit differences between the Mode 3/A codes is greater than 2, then the Mode 3/A zone is set to 4 and no further testing is to be done with this previous-scan report (the mismatch is too great for this pair of reports to initiate a track).

The zone scoring for Mode C altitudes is based on the altitude types of each of the uncorrelated beacon reports. A "high validity" altitude is defined to have a validity value equal to 3. A "low validity" altitude has a validity value less than 3. Altitude types are derived from the report's Mode C altitude and validity values and are defined as follows:

0 - no Mode C
1 - low-validity brackets-only
2 - high-validity brackets-only
3 - low-validity flight level
4 - high-validity flight level

The zone score for Mode C altitudes is defined in the table below as a function of the altitude types for each report. Wherever a number appears in the table, this indicates the altitude zone score value. Wherever a capital letter in boldface type appears in the table, this indicates a processing function described below the table.

	Current-scan Report Altitude Type				
Previous-scan Altitude Type	0	1	2	3	4
0	0	1	1	1	1
1	1	0	0	2	2
2	1	2	2	**A**	**A**
(4)	1	2	3	**A**	**B**

Test **A** converts each report's Mode C altitude back to 12-bit Gray codes and masks out the 3 octal 'C' bits. The number of bit differences between the masked Gray codes is computed. If the number of bit differences is less than or equal to 1 then the altitude zone score is 1. If the number of bit differences is less than or equal to 3 then the altitude zone score is 2. If the number of bit differences is greater than 3, the altitude zone score is set to 3.

Test **B** computes the absolute value of the difference between the two Mode C altitude levels. If this difference is less than or equal to 5 (500 feet), the altitude zone score is set to 0. Otherwise, continue as in test **A** above.

The total zone score for a pair of reports is the sum of the Mode A code and Mode C altitude zone scores defined above. If the total zone score is greater than 3, then these reports are a mismatch and not candidates for beacon track initiation.

From the set of previous-scan reports found in the range/azimuth search (if any were found), select those which have the best (lowest) code/altitude zone scores. Each of these previous-scan uncorrelated reports initiates a new beacon track with the current-scan uncorrelated report. Each of these previous-scan uncorrelated reports is marked so that they will not be candidates for any further track initiations, and the current-scan uncorrelated report is not stored for potential track initiation on the next scan. Alternatively, if no previous-scan reports were found in the range/azimuth search, store the current-scan uncorrelated report for possible beacon track initiation on the next scan.

Note: it is possible for a single uncorrelated report on the current scan to initiate more than one track. Only one of these tracks is likely to be the real aircraft; the other tracks are probably spurious and are unlikely to find subsequent associations. (Measurements made with high aircraft-density recordings from Los Angeles show that less than 3.5% of beacon track initiations involve more than one track, and the worst-case was 3 beacon tracks initiated on a given report (only one such case was found in more than one hour of data).) It was decided that the extra storage and processing complexity required to link together all the beacon tracks initiated from a given uncorrelated report in order to immediately drop the spurious tracks when the actual track correlates on the next scan is not worthwhile – the spurious tracks will drop due to coasting a few scans later anyway (and multiple initiations are a low-probability event).

The beacon track initiation process is shown in Figure 28.

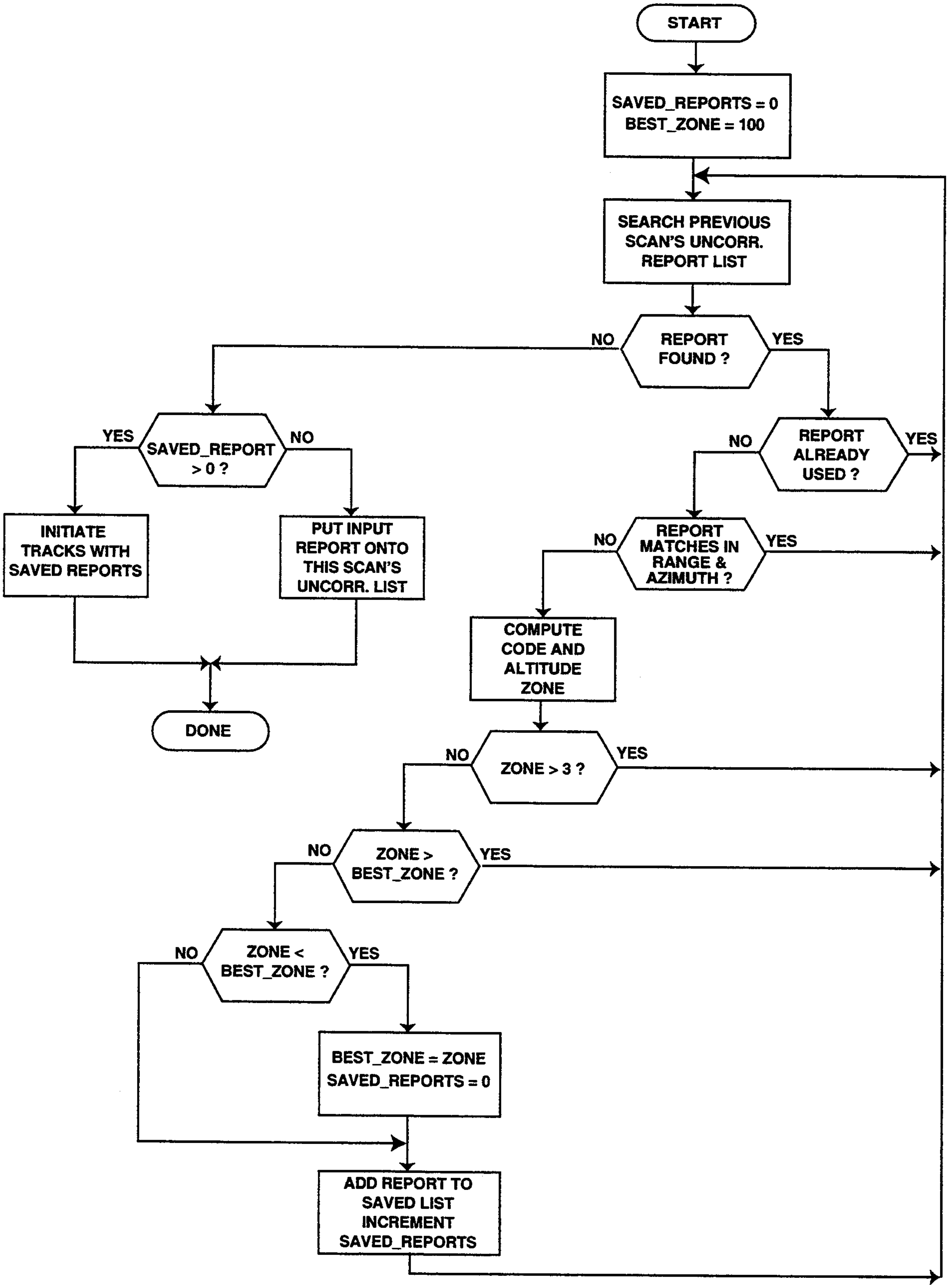


Figure 28. Beacon Track Initiation.

5.3.4.2 Radar Initiation

Called by:	Track
Calls:	
Purpose:	Create new tracks from previous uncorrelated radar-only reports.

A. Definitions

A.1 Report Score

The "report score" (RS) of a radar report is the sum of its confidence-quality score (CQS) and its Doppler agreement score (DAS). (Note: Doppler agreement is determined by using the apparent radial velocity computed from the range difference of a pair of radar reports to derive expected Doppler values for the radar's high and low PRFs. The 9-PAC Doppler agreement algorithm (Section 5.3.2.5) is used to compare the actual Doppler values with the expected values.) The CQS is determined from the VSP "eligibility" table using either the "in close" entries (if the report range is less than the VSP "range," or the "far" entries from the initiation table. The DAS is zero if the report's high and low PRF Doppler values do not agree with the computed radial velocity of the candidate track (in track initiation), or the VSP "Doppler agreement" parameter if the Doppler values pass the agreement test.

A.2 Report "Triple"

A "report triple" refers to a set of three or more uncorrelated radar-only reports from separate, previous scans that have passed the tests for candidate track initiation, but their total score (sum of report scores (RS) for the reports comprising the triple) is less than the minimum score required for track initiation VSP SINIT or there has been insufficient movement for track initiation. The tests required for adding another uncorrelated radar report to a "triple" are:

a) each report added to a triple must pass two linearity tests
   1) linear with respect to last 3 reports in triple, and
   2) linear with respect to triple origin report or curvilinear (see below) with respect to last 3 reports in triple

b) each report in a triple cannot have started a track (either early or normal)

c) no more than VSP NMISSES missed scans allowed in a triple

d) no more than VSP TRIPLE_SCANS total scans allowed in a triple (else delete the triple)

e) the "points/scan" (total score divided by the number of scans that the triple has existed) must be greater than or equal to the VSP TRIPLE_PPS_EXT

f) no more than one of the uncorrelated radar reports incorporated into the "triple" has confidence equal to 1 (this test seeks to minimize the generation of false tracks from geocensor reports over roads)

The linearity test in (a1) is performed by computing the straight line between the position of the potential new report and the next-to-latest report added to the triple. The position on this line corresponding to the scan of the most-recent report added to the triple is linearly interpolated. The linearity test succeeds if the absolute difference of the interpolated position and the actual position of the most-recent report added to the triple is less than a parametric number of sensor measurement standard deviations in range and azimuth, otherwise the (a1) linearity test fails. The range and azimuth thresholds for this linearity test are reduced by a factor of 0.85 if the new report's confidence is four.

The linearity test in (a2) is performed by computing the straight line between the position of the potential new report and the triple origin report. The position on this line corresponding to the scan of the next-to-last report added to the triple is linearly interpolated. The linearity test succeeds if the absolute difference of the interpolated position and the actual position of the next-to-last report added to the triple is less than a parametric number (NSIGMAS) of sensor measurement standard deviations in range and azimuth. If the (a2) linearity test fails, the "curvilinear" test is performed as described in sub-section C below. The range and azimuth thresholds for this linearity test are reduced by a factor of 0.85 if the new report's confidence is four.

B. Radar Track Initiation

There are three processes that may initiate a radar-only track. "Early Initiation" processing involves only two uncorrelated radar reports on subsequent scans and of sufficient quality, Doppler agreement, etc. to justify immediate track initiation. "Normal Initiation" processing involves three uncorrelated radar reports over the last 4-5 scans depending on VSP "3OF5". "Delayed Initiation" processing involves an uncorrelated radar report on the current scan and a "triple" (see definition A.2 above) formed on previous scans. "Delayed" initiation processing allows marginal tracks to be initiated over extended time while requiring additional testing to minimize false track generation. The radar track initiation processes are shown in Figures 29-34.

A current-scan uncorrelated radar report will be eliminated from consideration in the track initiation process if either of the following conditions is met:

a) the CQS for the report (see definition A.1 above) is zero

b) the report quality=0, the report confidence=3, the "Q0C3_disable" flag is set (for the azimuth sector of the report), and either of the following conditions is met

    1) the report amplitude is less than or equal to 34 dB

    2) the "high-amplitude Q0C3_disable" flag (for the azimuth sector) is set

The Q0C3_disable flag (for the report's azimuth sector) is normally cleared to FALSE. The Q0C3_disable flag is set TRUE when the number of quality=0, confidence=3 radar reports in the system (averaged over the preceding 5 scans) exceeds the Q0C3MAX VSP. The flag is cleared when the number of q=0,c=3 reports in the system (averaged over the preceding 5 scans) drops below 80 percent of the VSP setting. Similarly, the high-amplitude HQ0C3_disable flag (for the report's azimuth sector) is normally cleared to FALSE. It is set TRUE when the number of quality=0, confidence=3 radar reports whose amplitude is greater than 34 dB in the system (averaged over the preceding 5 scans) exceeds the Q0C3MAX VSP. The flag is cleared when the number of q=0,c=3 radar reports with amplitude greater than 34 dB in the system (averaged over the preceding 5 scans) drops below 80 percent of the VSP setting.

B.1 Early Initiation

Early initiation attempts to form new tracks using only two uncorrelated radar-only reports that must meet a set of quite-stringent conditions. To initiate an "early" track, the following conditions must hold:

a) current-scan uncorrelated radar report's confidence ≥ 3, and

b) unique positional match (within the distance that an aircraft could fly at VSP MAXSPEED in one scan) with uncorrelated radar report from last scan (no missed scans allowed), and

c) sum of CQS for both reports greater than or equal to VSP SINIT, and

d) have Doppler agreement, and

e) the distance between the first and second report must pass either

   1) range difference greater than VSP MIN_DIST, or

   2) azimuth difference greater than VSP MIN_ACPS pro-rated at VSP setting "in-close"

The reports that were used to form an "early" track may not participate in any further track initiation testing.

B.2 Normal Initiation

Normal initiation uses three uncorrelated radar-only reports to form a new track. To initiate a "normal" track the following conditions must hold for the current scan uncorrelated report and two uncorrelated reports from different prior scans:

a) positional match (within the distance that an aircraft could fly at VSP MAXSPEED (nominally 600 knots) in the number of scans between the reports) with current scan uncorrelated radar report and 2 additional uncorrelated radar reports from the previous 4 scans (3 out of 5) or previous 3 scans (3 out of 4) depending on VSP 3OF5, and

b) 3 reports pass linearity test, and

c) sum of CQS for each of three reports plus DAS for current scan and first previous scan reports is greater than or equal to VSP SINIT, and

d) the distance between the first and third report must pass either

   1) range difference greater than VSP MIN_DIST, or

   2) azimuth difference greater than VSP MIN_ACPS pro-rated for reports within the VSP "in-close" range parameter

e) no more than one of the uncorrelated radar reports may have confidence equal to 1

The linearity test in (b) is performed by computing the straight line between the position of the current scan uncorrelated report and the position of the oldest previous scan uncorrelated report. The position on this line corresponding to the scan of the more-recent prior scan uncorrelated report is linearly interpolated. The linearity test succeeds if the absolute difference of the interpolated position and the actual position of the more-recent prior scan uncorrelated report is less than a parametric number of sensor measurement standard deviations in range and azimuth, otherwise the (b) linearity test fails. The range and azimuth thresholds for this linearity test are reduced by a factor of 0.85 if the new report's confidence is four.

The two previous-scan reports used to initiate a "normal" track may not participate in any further track initiation testing. If the confidence of the current scan radar uncorrelated report is greater than or equal to 3, the current scan report may participate in further testing -- if its confidence is less than 3, it may not participate. If normal track initiation fails test (d) above, the three reports are made into a "triple report" and are candidates for delayed initiation.

B.3 Delayed Initiation

Delayed initiation attempts to form on a track with a sequence of 4 our more uncorrelated radar tracks that have failed to meet the conditions for early or normal track initiation. Delayed initiation typically handles cases of "marginal" data. Delayed initiation is performed with an uncorrelated radar report on the current scan and a triple report formed 1-4 scans previously. The combination of the current scan report and the triple must pass the following tests:

a) current scan report does not initiate an early or normal track, and

b) less than VSP NMISSES missed scans in the triple over the last 5 scans, and

c) less than VSP TRIPLE_SCANS scans in triple, and

d) linearity test passed with current scan radar report and triple, and

e) sum of CQS for the current scan report plus DAS for current scan radar report plus RS of triple report greater than or equal to VSP SINIT, and

f) RS score sum computed in (e) above is greater than or equal to any other RS score utilizing the current scan radar report, and

g) using the origin point of the triple report and the current scan uncorrelated report

    1) computed speed of candidate track formed from triple must be greater than or equal to the VSP MINVEL parameter, and

    2) candidate track must move greater than or equal to VSP MIN_DIST parameter or greater than or equal to VSP MIN_ACPS in azimuth, and

h) the total score sum of the triple divided by the total number of scans that the triple has existed (the "points-per-scan" value) must be greater than or equal to the VSP TRIP_PPS_SCORE, and

i) either

    1) linearity test passed with current scan radar report and triple origin point (the triple origin point is the oldest report used in formation of the triple), or

2) curvilinear test (subsection C below) passed with current scan report and triple point

j) no more than one uncorrelated radar report in the "triple" sequence may have confidence equal to 1 (this test reduces the formation of false tracks using geocensor reports over roads)

The linearity test in (d) is performed by computing the straight line between the position of the current scan uncorrelated report and the oldest previous scan uncorrelated report. The position on this line corresponding to the scan of the more-recent report added to the triple is linearly interpolated. The linearity test succeeds if the absolute difference of the interpolated position and the actual position of the more-recent report added to the triple is less than a parametric number of sensor measurement standard deviations in range and azimuth. The range and azimuth thresholds for this linearity test are reduced by a factor of 0.85 if the new report's confidence is four.

The linearity test in (i) is performed by computing the straight line between the position of the current scan uncorrelated report and the triple origin report. The position on this line corresponding to the scan of the next-to-last report added to the triple is linearly interpolated. The linearity test succeeds if the absolute difference of the interpolated position and the actual position of the next-to-last report added to the triple is less than a parametric number of sensor measurement standard deviations in range and azimuth. The range and azimuth thresholds for this linearity test are reduced by a factor of 0.85 if the new report's confidence is four. If the (a2) linearity test fails, the "curvilinear" test is performed as described in subsection C below.

If a delayed initiation is performed, the given triple report may no longer participate in further testing. If the confidence of the current scan radar uncorrelated report is greater than or equal to 3, the current scan report may participate in further testing -- if its confidence is less than 3, it may not participate in any further testing.

C. Curvilinear Test

The 9-PAC Phase II Tracker "curvilinear" test is used to determine whether an uncorrelated radar report on the current scan is a candidate for addition to an existing radar "triple." The curvilinear test attempts to determine if the set of uncorrelated radar reports consisting of those in the triple and the candidate current-scan report determine an arc with a constant turn rate reasonable for a normally-maneuvering aircraft. This allows delayed track initiation on aircraft turning before they begin straight flight (e.g. on climb-out after takeoff). Inputs to the curvilinear test are 4 uncorrelated radar reports as follows:

1. current scan uncorrelated report (range, azimuth, scan)
2. most-recent uncorrelated report in triple (range, azimuth, scan)
3. previous uncorrelated report in triple (range, azimuth, scan)
4. origin uncorrelated report in triple (range, azimuth, scan)

The first step of the curvilinear test is to convert each of the 4 input report positions from range-azimuth to x-y coordinates, denoted (x_n, y_n) for each of the 4 input reports. The number of scans between report 1 and 2 is denoted "sc1", the number of scans between report 2 and 3 is denoted "sc2", and the number of scans between report 3 and 4 is denoted "sc3".

The second step of the curvilinear test is to compute the orientation angles for the 3 lines joining the 4 report points as follows:

1. $a1 = \text{atan2}(x_2 - x_1, y_2 - y_1)$ *line between 1 and 2*
2. $a2 = \text{atan2}(x_3 - x_2, y_3 - y_2)$ *line between 2 and 3*
3. $a3 = \text{atan2}(x_4 - x_3, y_4 - y_3)$ *line between 3 and 4*

The third step of the curvilinear test is to compute the turn rate between lines 1 and 2 (denoted "da1"), and the turn rate between lines 2 and 3 (denoted "da2"). Note that the turn rate is assumed to apply at the midpoint of each line.

$da1 = (a2 - a1) / (0.5 * (sc1 + sc2))$ *angle between line 1 and 2*
$da2 = (a3 - a2) / (0.5 * (sc2 + sc3))$ *angle between line 2 and 3*

To pass the curvilinear test, both da1 and da2 must have the same sign and the absolute difference between them must be less than 6 degrees per second. (Note: a normally-maneuvering aircraft performs standard-rate turns at a 3 degrees-per-second rate.)

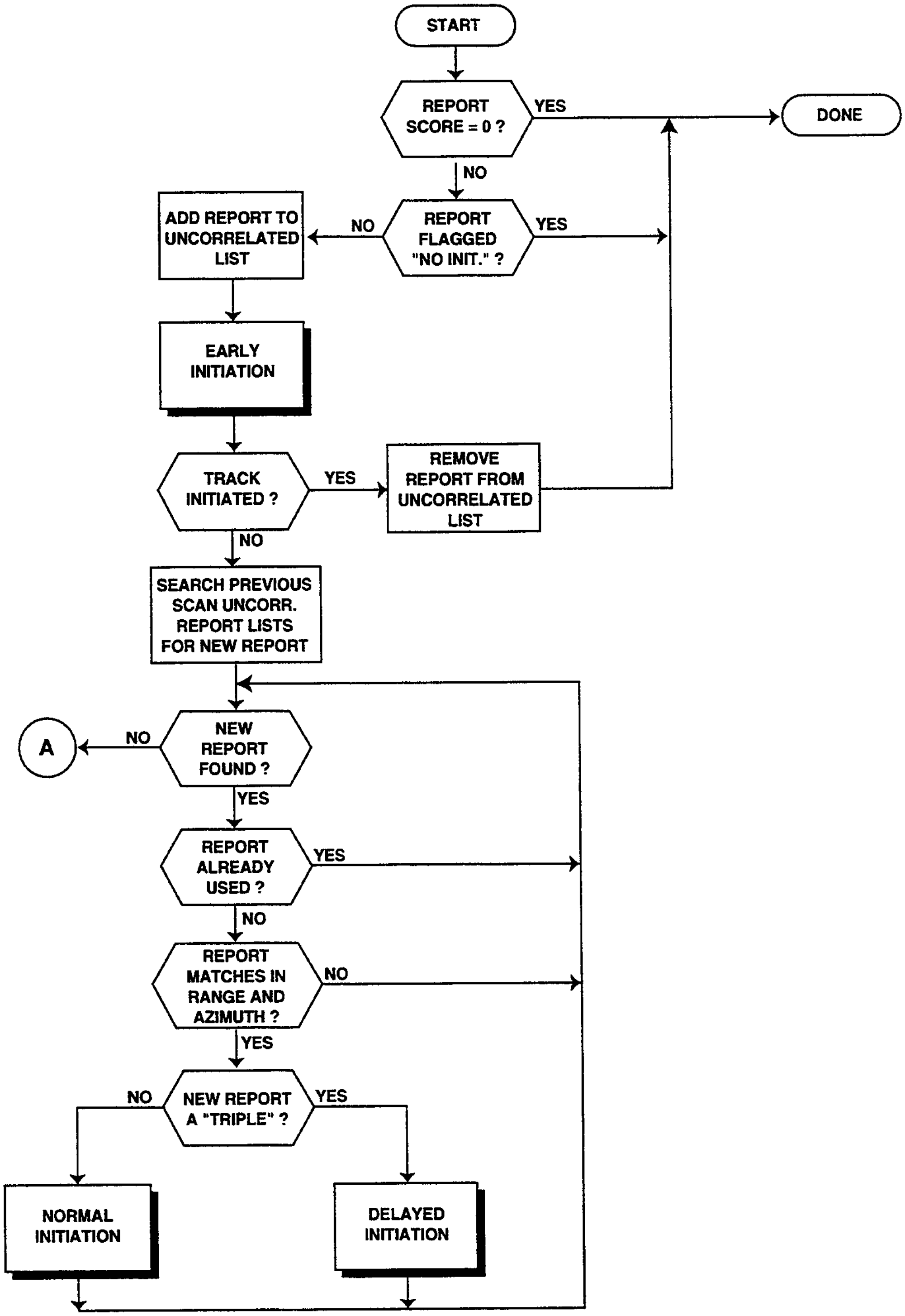


Figure 29. Radar Track Initiation (1 of 2).

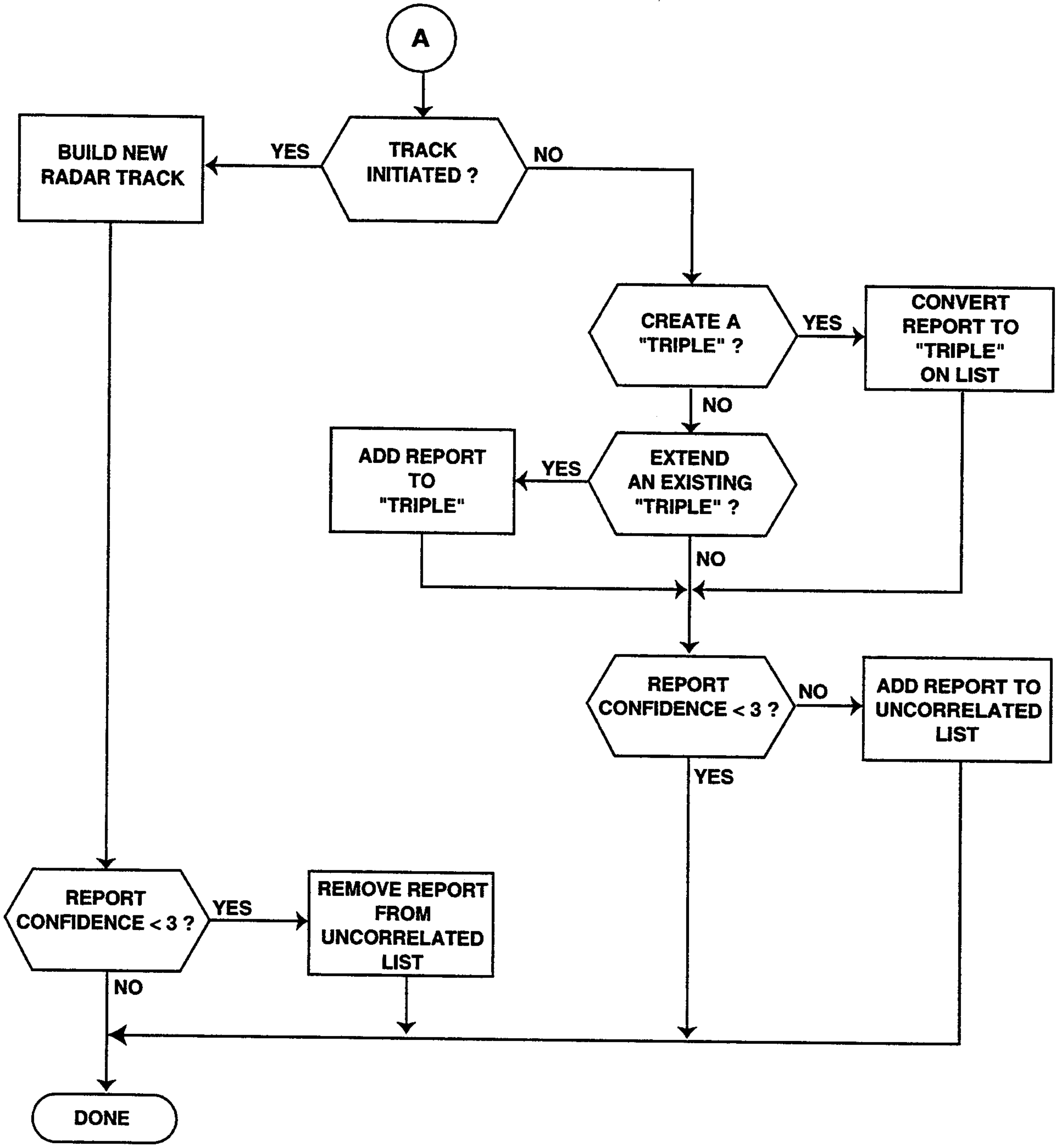


Figure 30. Radar Track Initiation (2 of 2).

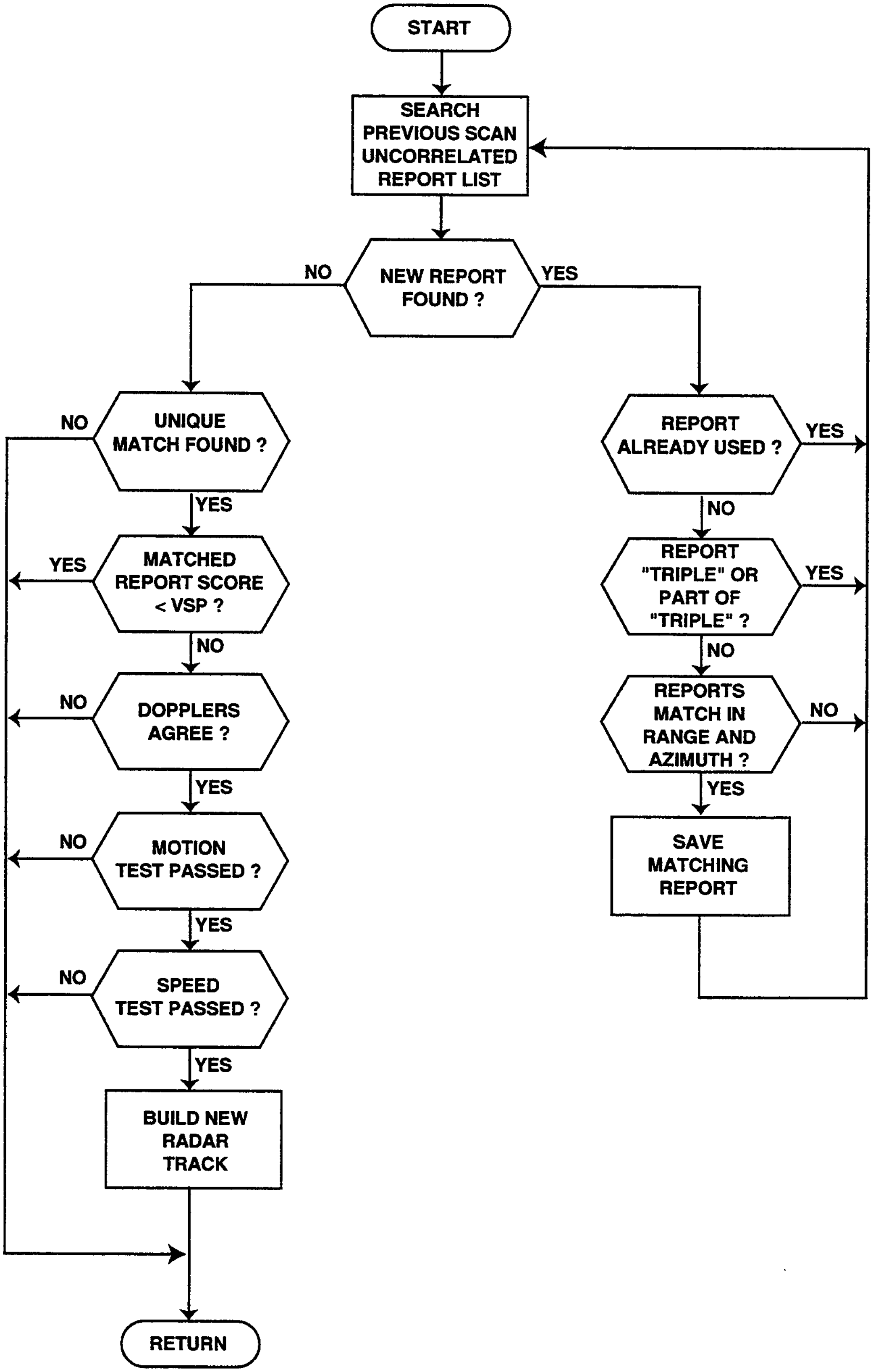


Figure 31. Early Initiation.

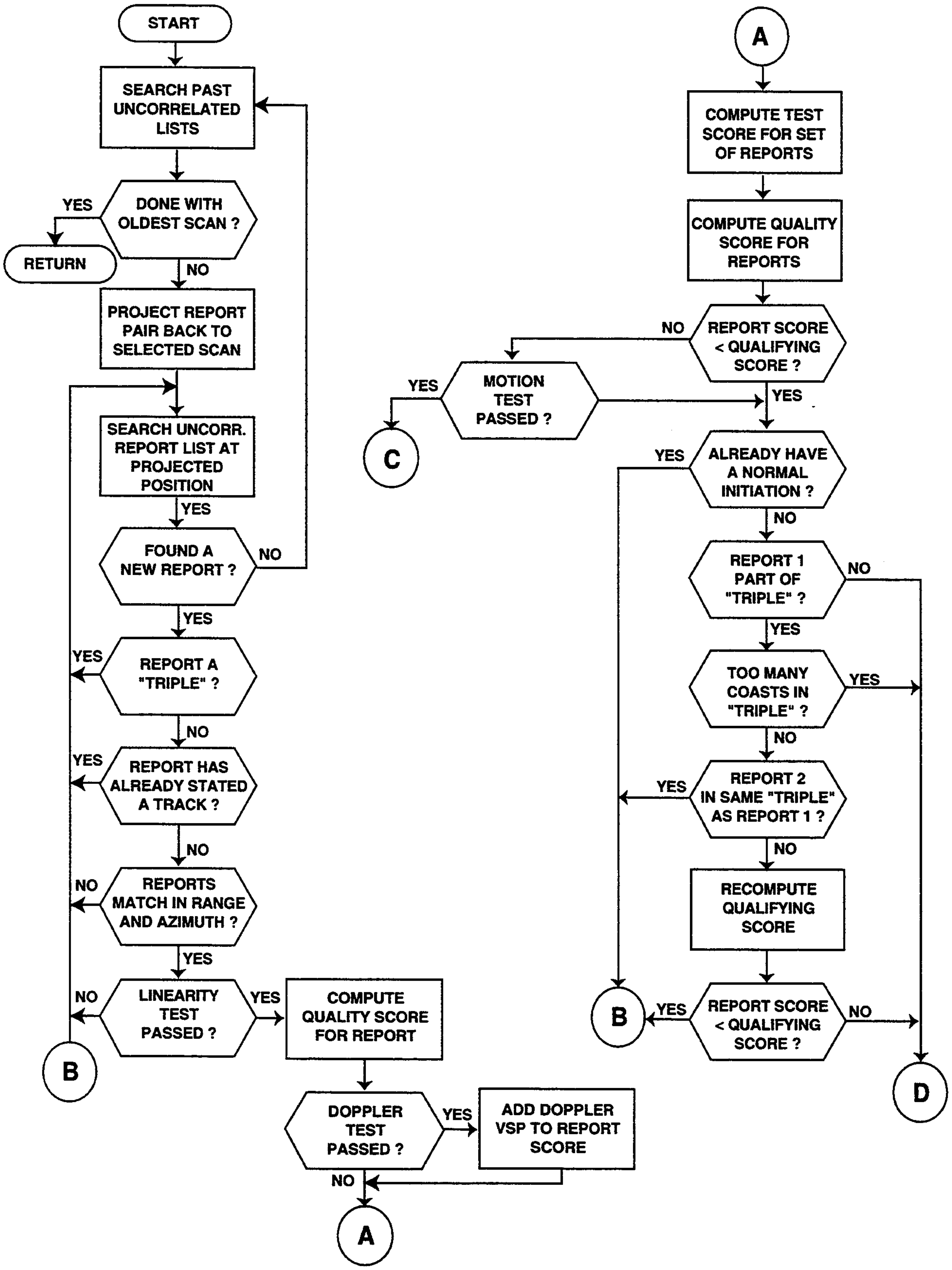


Figure 32. Normal Initiation (1 of 2).

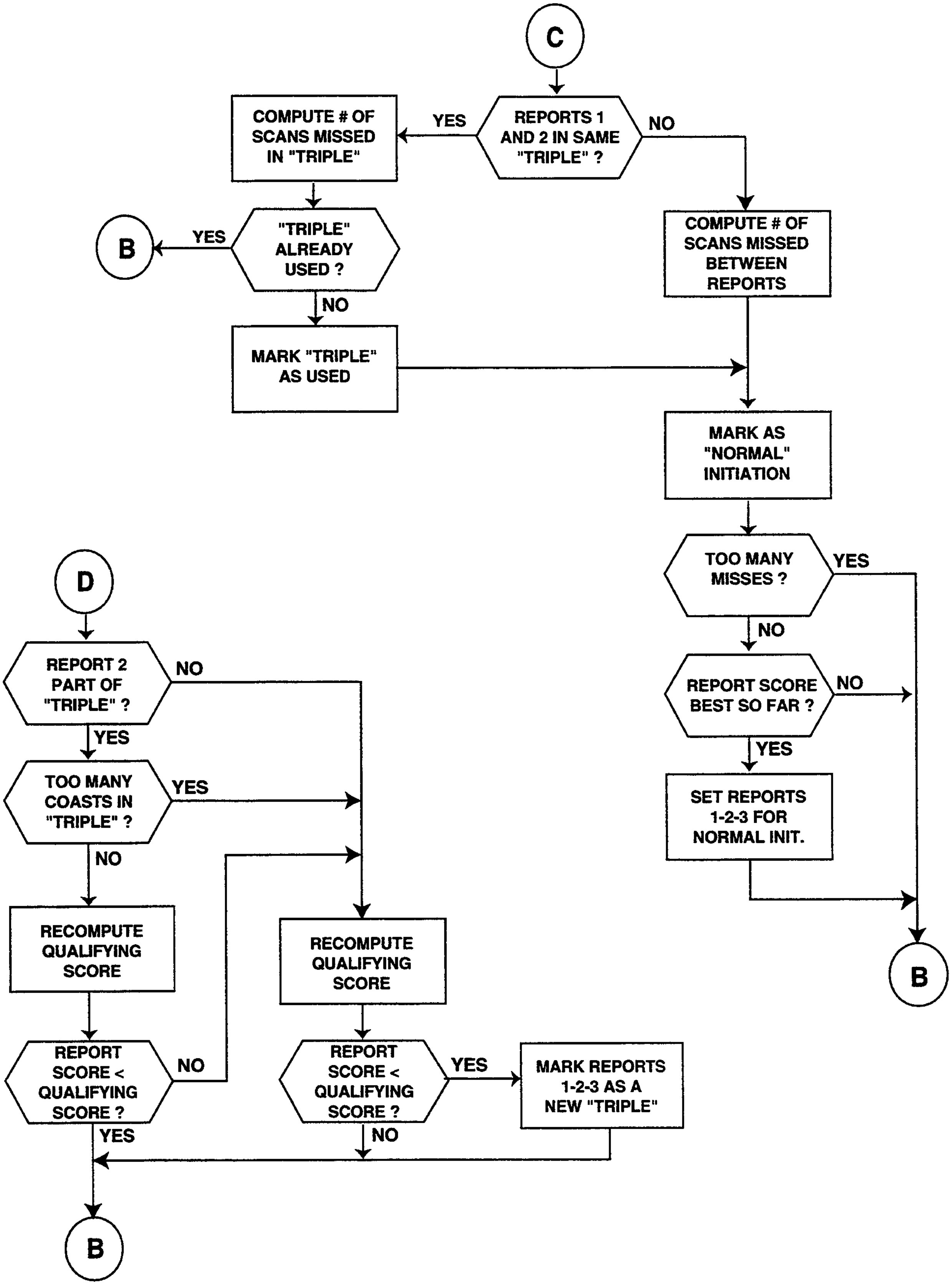


Figure 33. Normal Initiation (2 of 2).

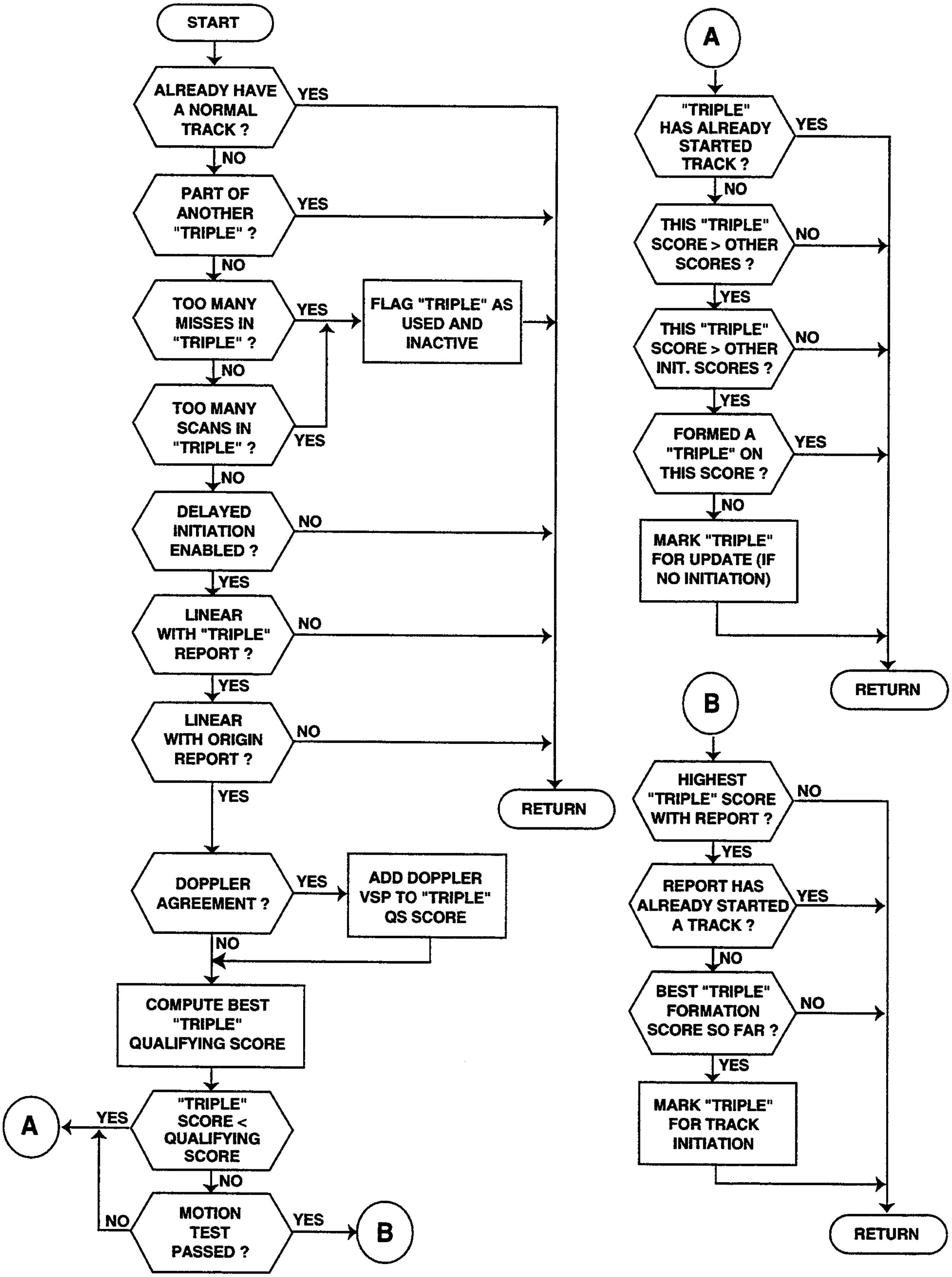


Figure 34. Delayed Initiation.

AFTERWORD

This report has covered in detail the algorithms that are in the 9-PAC Phase II Scan-Scan Correlator or Tracker. The 9-PAC Phase II Tracker builds upon the earlier prototype version of the Phase II Tracker design described in Reference [7]. This report is one in a series of reports that document the 9-PAC algorithms in enough detail to support implementation by a second party.

The Tracker algorithms described herein have been tested with recorded field data in a real-time test facility, are now undergoing field evaluation in ASR-9s at Lincoln Laboratory, Albuquerque, and other sites. As these tests proceed algorithm refinements may occur. Refinements will be documented in a future report or issued as an addendum to this report.

APPENDIX A. DATA STRUCTURES

Tables 7 and 8 are the two main data structures used in the 9-PAC Tracker: reports (trkrptype) and tracks (trktype). Note that the structures are not dependent on the type of report or track. Beacon and radar reports use the same structure definition, and beacon and radar tracks use the same structure definition.

Table 7. Report Data Structure

Field Name	Description
Node_header node_hdr;	Pointer;
/*common fields*/	
int rpt_type;	Radar, Beacon, or RB
int scan;	Current scan counter
int rtqc;	Set by BTD or C&I; Flag: 0- no, 1 - yes
int acp;	Set by BTD or C&I; Azimuth, 4096 ACPs = 360°
int run;	Set by BTD or C&I; Runlength
float range;	Set by BTD or C&I; range in nmi
float x	Cartesian coordinate, nmi; uses alt. if available
float y;	Cartesian coordinate, nmi; uses alt. if available
/*beacon fields*/	
int mode_a;	Set by BTD; Mode 3A code; set to 0 to radar
int mode_2;	Set by BTD; Mode 2 code, typically 0
int alt;	Mode C code converted to feet; set to 0 for radar
int val_flags;	Set by BTD: bits 0-4 are beacon hits count, bit 8 glags discrete Mode 3/A report, bit 10 flags Mode 3/A X validity, bits 11-12 are Mode C validity, bit 13 flags SPI, bits 14-15 emergency codes (7600, etc.)
int arts_qual;	Set by BTD
int false;	Set by BTD
/*radar data fields*/	
int quality;	Set by C&I; [0..3]; set to 4 for beacon
int conf;	Set by C*I; [0..5]; set to 7 for beacon
int elig;	Radar eligibility score [qual] [conf] from VSP tables
int maint;	Radar maintenance score [qual] [conf] from VSP tables
float amp;	Set by C&I; 3/32 of a dB units
int lo_filt;	Set by C&I; [0..8] maps to [=3..+3]
int hi-filt;	Set by C&I; [0..10] maps to [-4..+4]
int lo_dopp;	Set by C&I; [0..63]
int hi_dopp;	Set by C&I; [0..63]

Table 7. Report Data Structure (continued)

Field Name	Description
int max filt;	Set by C&I: contains highest amplitude Doppler filter
int flag1;	Set by C&I: contains the MTI flag
int flag2;	Set by C&I:
int adapt_thresh_info;	Set by C&I; Two words: flag plus threshold, amplitude information; 0 - unflagged, 1 - flagged
int az_degrade_flag;	Bit 0 flags bad C&IO centroid, bit 15 flags "use radar quality"
int suspect;	Set by C&I flags slow-moving "suspect" report
/*working fields*/	
int no init;	Flag: 0 - OK for initiation. 1 - not for initiation
int num_assoc;	Counter
int num disc assoc;	Counter
int corr;	Flag: 0 - not correlated. 1 - correlated
int output;	Flag: 0 - not output. 1 - output
int num maxassoc;	Counter. used for analysis
/*report status analysis fields*/	
int seq_number;	Report index number
int assoc count;	Number of valid entries in "assoc data[]"
Assoc type assoc data[7];	Report association analysis fields
int init count;	Number of valid entries in "init data[]"
int type init data[3];	Report track initiation analysis fields
/*C&I Feedback fields*/	
int age;	# of scans in correlated track
int avg speed;	Correlating track's average ground-speed
int bird ice flag;	Count of "bird/ice" suspect track updates (-1 if disabled)
int hdg_flag;	=1 if corr. track heading histogram count > VSP
int hdg_switch-cnt;	# of scans with erratic heading
/*Tracker analysis fields*/	
int trk num;	Track number of correlated track
int state;	Track state (INIT 1.STABLE, etc.)
TRK TRK T*assoc trk[11];	Array of pointers to associated tracks

Table 8. Track Data Structure

Field Name	Description
Node_header node_dhr;	Pointer
int trk_type;	Beacon, radar, or radar-reinforced; Set by initial report type
int old_scores[5];	Radar "elig" scores for last 5 scans
int sum_score;	Total radar qual/conf score over last 5 scans
unsigned int flags;	Analysis flags; 0=radar-only linearity test, 1-socring test, 2-=Doppler test, 3-10=track initiation type, 11-tracker "fixup," 12-15=reserved, 16-31=correlating report sequence number.
unsigned int flags2;	More analysis data: 0-15=triple index for radar initiation
int old_acp_1;	Last azimuth update (ACP)
float old_range_1;	Last range update (nmi)
int old_acp_2;	2nd to last azimuth update (ACP)
float old_range_2;	2nd to last range update (nmi)
int old_acp_3;	3rd to last azimuth update (ACP)
float old_range_3;	3rd to last range update (nmi)
int az_pred;	Predicted azimuth (ACP)
float range_pred;	Predicted range (nmi)
int old_az_pred;	Previous value of "az_pred" - for "fixup" algorithm
float old_range_pred;	Previous value of "range_pred" - for "fixup" algorithm
float x_pred;	Tracker x-position output (nmi)
float x_smooth;	Tracker filter x-position output (nmi)
float x_sm_dot;	Tracker x-velocity output
float y_pred;	Tracker x-position output (nmi)
float y_smooth;	Tracker filter y-position output (nmi)
float y_sm_dot;	Tracker y-velocity output
float x_init;	Position of first report; for min_dist test (nmi)
float y_init;	Position of first report; for min_dist test (nmi)
float range_box_max[2];	Association box upper limit; zone 0 and 1 (nmi)
float range_box_min[2];	Saved association box upper limit; zone 0 and 1 (nmi)
float old_range_box_nmi[2];	Saved association box lower limit; zone 0 and 1 (nmi)
int az_box_max[2];	Association box upper limit; zone 0 and 1 (ACPs)
int az_box_min[2];	Association box lower limit; zone 0 and 1 (ACPs)
int old az_box_max[2];	Saved association box upper limit; zone 0 and 1 (ACPs)
int old_az_box_min[2];	Saved association box lower limit; zone 0 and 1 (ACPs)
float vel;	instantaneous velocity (knots)
float avg_vel;	filtered velocity (knots)
float rad_vel;	instantaneous radial velocity (knots)
float avg_rad_vel;	filtered radial velocity
float az_vel;	instantaneous azimuthal velocity; (ACPs/hour)
float old_rad_vel;	Saved "rad_vel" - used in "fixup" algorithm
float old_az_vel;	Saved "az_vel" - used in "fixup" algorithm
int inst_hdg;	track's instantaneous heading (ACPs)
int old_inst_hdg;	track's previous value of inst_hdg (ACPs)
int smooth_hdg;	track's smoothed heading (ACPs)

Table 8. Track Data Structure (continued)

Field Name	Description
/*beacon specific fields*/	
int code;	Mode 3/A code + 2-bit validity; set to 0 for radar
int discrete;	Flag: 0 - non-discrete, 1 - discrete
int code_change;	Counter for changing Mode 3/A code
int alt;	Mode C converted to feet; set to 0 for radar
int alt_validity	Mode C validity (0-3)
int split;	Counter for discrete beacon split logic
/*track status fields*/	
int trk_num;	Track file number
int state;	Track state - see state diagram
int rtqc;	Flag "Real Time Quality Control" track
int qual_old;	Last updates Quality
int conf_old;	Last updates Confidence
int avg_scans;	# of scans to compute new Doppler average
int dopps[];	High-PRF Doppler values (0-avg_scans)
int avg_hi_dopp;	Averaged high-PRF Doppler value
int zone;	Association Zone of latest correlation
int fixup_flag;	-2 when last scan corr. removed. Counts down to zero
int had_beacon;	Flag that the track has ever had a non-radar-only state
int radar_miss;	Count of scans without beacon data in correlation
int beacon_miss;	Count of scans without beacon data in correlation
int miss_1;	# of misses since last update
int miss_2;	# of misses between last and 2nd to last update
int miss_3;	# of misses between 2nd to last and 3rd to last update
int hit;	Total # of hits on track; for statistics
in bird_ice_count;	Total # of "bird/ice" consistent corrs. (-1 is disabled)
int hdg_switch_cnt;	# of scans of erratic heading
int age;	Total # of possible hits on track; for statistics
int min_dist;	Flag: 0 - not passed test, 1 - passed test;
int force_coast;	Flag: 0 - not forced, 1 - forced; for beacon split
int coast	Flag: 0 - not coasted, 1 - coasted
int corr;	Flag: 0 - not correlated, 1 - correlated
int num_maxassoc;	Count of associations for track (analysis)
int available	Flag: 0 - not avail, 1 - avail; for Gen. to Act. list
int num_assoc;	Count number of associated reports
int num_disc_assoc;	Counter of associated discrete Mode 3/A reports
int sect_pred;	Azimuth sector that track is predicted to be
int sect_max;	Azimuth sector that track processing must complete
int row;	Track x-y sort-bin linkage
int col;	Track x-y sort-bin linkage
int corr_type	Type of correlation (analysis; 1-on-1, Many-Many, etc.)
int landing;	Flag if mature track below VSP HEIGHT
TRK_RPT_T *assoc_rpt[11];	Array of pointers to associating reports
TRK_RPT_T *corr_rpt;	Pointer to correlating report

APPENDIX B. VARIABLE SITE PARAMETERS

GENERAL RADAR SYSTEM PARAMETERS

FREQ_A

FREQ_B

These VSPs set the operating frequency of the radar's A and B channels respectively. The appropriate VSP (determined by the active channel) is used together with the PRF values to compute the expected interpolated Doppler value for a given track range-rate. The comparison between the expected Doppler and the measured Doppler value in the input radar report is used in the conflict resolution algorithm when multiple reports/tracks associate. These VSPs are input in units of MHz. A typical value is 2730.

HEIGHT

This VSP sets the elevation of the radar antenna above sea level. It is given in flight levels (100 feet = 1 flight level). A typical value is 6.

PRF_H

PRF_L

These VSPs set the pulse-repetition frequency for the high and low intervals of the radar. These VSPs are used along with the radar frequency value (from the active channel) to compute the expected interpolated Doppler value for a given track range-rate. The comparison between the expected Doppler and the measured Doppler value in the input radar report is used in the conflict resolution algorithm when multiple reports/tracks associate. These VSPs are input in units of Hz. A typical value is 940.

SIGMA_RANGE, INIT_SIGMA_RANGE

SIGMA_AZ, INIT_SIGMA_AZ

These VSPs set the sensor typical range and azimuth standard deviation error value. They are used in sizing track association boxes, etc. The range sigma is input in units of 64ths of a nautical mile. The azimuth sigma is input in units of 16ths of an ACP. Separate VSPs are provided for radar-only track initiation and for radar track association/correlation use. (This allows for the separate tuning of the algorithms. The sigmas are typically set to the same value anyway.) A typical value for the range sigmas is 3 (3/64ths of a mile). A typical value for the azimuth sigmas is 32 (2 ACPs).

TEST-MODE SWITCHES

RADAR_ONLY

This VSP is used to force the 9-PAC tracker to treat input beacon targets as though they were radar-only data. This mode is used for testing the radar tracking algorithms. The nominal value is 0 (off – normal operation of the beacon tracking algorithms). Set the VSP to a non-zero value to enable the test mode.

OVERALL 9-PAC RADAR TRACKER PARAMETERS

ELIG_RANGE

This VSP sets the range threshold separating the usage of the "close-in" scoring tables (see CLOSE_INIT_TAB and CLOSE_MAINT_TAB) from the usage of the "far-out" scoring tables (see FAR_INIT_TAB and FAR_MAIN_TAB). Ranges exceeding this VSP use the "far" tables. This VSP is input in units of nautical miles. A nominal value for this VSP is 40 nautical miles.

NG

This VSP sets the maximum expected turn-rate for a real radar-only or non-discrete beacon aircraft. It is used to enlarge the outer track association box to a sufficient size to encompass reasonably turning aircraft. This VSP is input in "g" units (an aircraft performing a "standard-rate" turn of 3 degrees/second is pulling 1 "g" of acceleration). The VSP is input in 10ths of a "g" units. The nominal value for NG is 28 (i.e. 2.8 "g").

NSC_ASSOC

This VSP sets the maximum number of scans of radar-only correlation required for the tracker to convert a track that was formerly beacon-supported to radar-only (and, vice versa, the number of scans of beacon-only correlation required to convert a track that was formerly radar-only to beacon-supported). The nominal value for this VSP is 300.

QUAL_SCORE

This VSP sets the minimum radar quality-confidence "points/scan" value for the track maintenance test performed on tracks moving at normal speeds. Each track computes its "score" (based on report quality, confidence, and Doppler agreement – see the ELIG VSP table below) every scan. The average "points/scan" value is maintained as a measure of track "goodness." If the track is moving at normal speed (see the SLOW_VEL VSP), then the track's "points/scan" value must be greater than the VSP in order to be maintained. If the score drops below the VSP, then the track is dropped. This VSP is input in 10ths of a score "point." A nominal value for this VSP is 7 (0.7 points/scan).

QUAL_SCORE_SLOW

This VSP sets the minimum radar quality-confidence "points/scan" value for the track maintenance test performed on tracks moving at below-normal speeds (i.e. below the SLOW_VEL VSP). Each track computes its "score" (based on report quality, confidence, and Doppler agreement – see the scoring VSP table(s) below) every scan. The average "points/scan" value is maintained as a measure of track "goodness." The track's "points/scan" value must be greater than the special "slow" VSP in order to be maintained. If the score drops below the VSP, then the track is dropped. A separate quality-scoring VSP is used for slow tracks so that these more "suspect" tracks (often false-tracks caused by birds) may be subjected to a more-stringent scoring test. This VSP is input in 10ths of a score "point." A nominal value for this VSP is 13 (1.3 points/scan).

SLOW_VEL

This VSP is used to determine the track speed below which the special QUAL_SCORE_SLOW VSP is to be used as the scoring threshold in track maintenance testing instead of the normal QUAL_SCORE VSP. This VSP is input in knots. A nominal value for this VSP is 140 knots.

HDG_TRK_CNT

This VSP is used to determine whether slow (track speed ≤ the VSP "SLOW_VEL") radar-only tracks in a particular heading bin are to set their heading feedback flag in the output report. (Note: a track's heading bin is determined from its smoothed heading.) If the number of radar-only tracks in the system in a given heading bin (each bin occupies 45 degrees) exceeds this VSP, then the feedback flag will be set, indicating that this heading has an excessive number of radar-only tracks --- a symptom of "birds" producing false tracks. The C&I processing will use the feedback to modify its thresholds to reduce the incidence of these false reports. A nominal value for this VSP is 6 tracks.

HDG_SWITCH

This VSP is used to determine whether a slow (track speed ≤ the VSP "SLOW_VEL") radar-only track's instantaneous heading has changed too much from its previous instantaneous heading – indicating the chance that this track is a "bird" rather than an aircraft. The track's heading switch counter is incremented when this occurs, and the counter value is fed back to the C&I processing. The C&I processing uses the feedback to alter its thresholds to reduce the incidence of false "bird" reports. A nominal value for this VSP is 700 ACPs.

HDG_TRK_ON_CNT

This VSP is used in the setting of the "birdHeadingFlag" global variable that indicates to the C&I processing when the tracker has found a number of "bird" tracks. If any heading bin "triad" (any histogram bin and its adjoining neighbor bins) has more than this VSP number of tracks, then the global flag is set. (This search is performed once per scan.) A nominal value for this VSP is 5 tracks.

HDG_TRK_OFF_CNT

This VSP is used in the clearing of the "birdHeadingFlag" global variable that indicates to the C&I processing when the tracks has found no incidence of "bird" tracks. If all the heading bin "triads" (histogram bin and its adjoining neighbor bins) have fewer than this VSP number of tracks, then the global flag is cleared. (This search is performed once per scan.) A nominal value for this VSP is 4 tracks.

QUALITY-CONFIDENCE SCORING TABLES

CLOSE_INIT_TAB
FAR_INIT_TAB
CLOSE_MAINT_TAB
FAR_MAINT_TAB

There are 4 quality-scoring tables input as VSPs to the 9-PAC Phase-II tracker. Each table is two-dimensional. There are 4 rows in each table, corresponding to radar quality values 0 through 3. There are 8 columns in each table, corresponding to radar confidence values 0 through 7. (Note: the table entry for quality 3, confidence 7 is reserved for beacon reports. Beacon-only reports are given a score equal to the sum of the equivalent table entry for quality=3, confidence=6 plus the DOPP_SCORE VSP.) Each table entry is a "score" value in "points," weighting how sure the radar is that a report with this quality-confidence combination is a return from a real aircraft and not generated from clutter.

Two of these tables (CLOSE_MAINT_TAB and FAR_MAINT_TAB) are used to compute the scoring function used in the track maintenance testing function. (See the QUAL_SCORE and QUAL_SCORE_SLOW VSPs above.) The other two tables (CLOSE_INIT_TAB and FAR_INIT_TAB) are used to compute the scoring used in the track initiation function. The selection of the CLOSE or FAR table in each case is performed by comparing the track (or report) range against the ELIG_RANGE VSP. The separation into "close" and "far" tables allows separate tailoring of the scoring parameters for the close-in region where clutter is more severe versus the far-out region where aircraft returns may be weaker. The division into "initiation" and "maintenance" tables allows separate tailoring of the scoring for track initiation (usually more stringent) and track maintenance (usually less stringent).

Typical values for CLOSE_INIT_TAB are:

0	0	0	1	0	0	0	0
0	0	0	2	2	0	4	0
0	0	0	2	2	0	4	0
0	0	0	3	3	0	5	0

Typical values for FAR_INIT_TAB are:

0	0	0	0	1	1	0	0
0	0	0	0	2	2	4	0
0	0	0	0	2	2	4	0
0	0	0	0	3	3	5	0

Typical values for CLOSE_MAINT_TAB are:

1	1	0	1	1	0	0	0
2	2	0	2	2	0	4	0
2	2	0	2	2	0	4	0
2	2	0	3	3	0	5	0

Typical values for FAR_MAINT_TAB are:

0	0	0	0	1	1	0	0
0	0	0	0	2	2	4	0
0	0	0	0	2	2	4	0
0	0	0	0	3	3	5	0

TRACK LINEARITY PARAMETERS

NSIGMAS_RANGE, INIT_NSIGMAS_RANGE

NSIGMAS_AZIMUTH, INIT_NSIGMAS_AZIMUTH

These VSPs are used in conjunction with the radar tracker SIGMA_RANGE and SIGMA_AZIMUTH VSPs to size the range and azimuth thresholds for the track linearity test used in track association, track initiation and maintenance testing. These VSPs are multipliers to select how many sigmas to allow a report to deviate from the linearly-predicted position. They are input in 10ths of a sigma units. (Note: separate VSPs are provided for radar track initiation and for track association/correlation. This allows for the separate tuning of the tracker radar-only track initiation and the radar track association/correlation linearity tests. Normally, the tests for radar-only track initiation would be more-stringent than those for radar track association.) The nominal value for the track initiation range VSP is 52 (5.2 sigmas) and the nominal value for the track initiation azimuth VSP is 48 (4.8 sigmas). The nominal value for the association/correlation range multiplier VSP is 48 (4.8 sigmas) and the nominal value for the association/correlation azimuth multiplier VSP is 48 (4.8 sigmas).

RADAR TRACK INITIATION PARAMETERS

Control Switches

THREE_OF_5

This switch determines whether valid radar track initiation sequences may have 2 scan misses (3 reports out of 5 scans) or whether only one scan miss (3 out of 4) is valid for radar track initiation. The nominal value for this VSP is 0 (only 1 miss permitted – 3 out of 4). Set this VSP to a non-zero value to enable the more-lenient 3-out-of-5 radar track initiation algorithm. (Note: 3-out-of-5 radar track initiation, while it may initiate marginal aircraft tracks earlier, may also cause more false-tracks to be initiated.)

USE_TRIPLES

This switch determines whether the radar "triple" algorithms are enabled for radar track initiation or not. The "triples" mechanism allows radar-only track initiation on report sequences that do not meet the usual stringent requirements for normal radar-only track initiation – they might take more than 3 scans of uncorrelated radar reports to achieve a sufficient score or movement distance as used for normal initiation. The "triples" algorithms require extra system storage and processing, and they may sometimes cause false tracks to be initiated – however, they can also initiate real tracks earlier (in fewer scans) and initiate tracks on marginal aircraft or aircraft in regions of heavy radar clutter. The nominal value for this VSP is 1 ("triples" processing is enabled). Set this VSP to zero in order to disable "triple" processing.

Movement Test Parameters

MIN_ACPS

In order for a radar-only track to be initiated, it must have moved sufficiently far in distance (range and/or azimuth) to pass the movement test. The MIN_ACPS VSP sets the movement threshold in the azimuth direction. If the prospective track has moved farther than this VSP in azimuth, then the movement test is passed. This VSP is input in units of ACPs. Since a change in ACPs is equivalent to a larger linear movement the farther out the prospective track is in range, the azimuth threshold is reduced linearly for ranges beyond the VSP RANGE value. The nominal value for the MIN_ACPS VSP is 28 ACPs.

MIN_DIST

In order for a radar-only track to be initiated, it must have moved sufficiently far in distance (range and/or azimuth) to pass the movement test. The MIN_DIST VSP sets the movement threshold in the range direction. If the prospective track has moved farther than this VSP in range, then the movement test is passed. This VSP is input in 64ths of a nautical mile units. The nominal value for the MIN_DIST VSP is 16 (0.25 miles).

MAX_VEL

This VSP is used to compute the size of the search window required for track initiation. The computation uses this maximum expected aircraft speed VSP to determine how far a real aircraft could move in a certain number of radar scans. This VSP is input in units of integer knots. The nominal value for this VSP is 600 knots.

MIN_VEL

Prospective radar-only tracks must have a ground speed greater than this VSP in order to be initiated. Tracks whose speed is too slow to be an aircraft in flight are more- likely to be false. (Note: the tracker's determination of initial track speed is subject to radar measurement errors that can cause speed errors of 10-20 knots typically.) This VSP is input in integer knots. A nominal value for this VSP is 30 knots.

RANGE

This VSP sets the range threshold inside which the MIN_ACPS value is used unmodified in the movement test. For prospective tracks outside RANGE, the movement test azimuth threshold is linearly reduced to make the required movement distance the same as it was at RANGE. This VSP is input in nautical mile units. The nominal value for this VSP is 4 nautical miles.

Track Initiation Scoring Parameters

DOPP_SCORE

This VSP sets the number of "scoring points" to be added to a radar quality-confidence score when a radar report's interpolated Doppler agrees with the measured range-rate during track initiation or track maintenance processing. This VSP is input in units of score "points." The nominal value is 1 point.

SINIT

This VSP sets the minimum quality-confidence-Doppler score value required for radar track initiation. The set of uncorrelated radar reports which are candidates for track initiation must have a total score greater than or equal to the sum of SINIT plus the Doppler agreement score parameter DOPP_SCORE. (Note: if the set of reports has a score less than this threshold but greater than TRIPLE_MIN_SCORE, then the reports may create a "triple" instead of performing track initiation.) This VSP is input in units of score "points." The nominal value for this VSP is 10 points.

“Triple Algorithm” Parameters

NMISSES

This VSP sets the maximum number of missed scans that can occur in a “triple” before it will be dropped from the system. The nominal value is 2 scans.

TRIPLE_MIN_SCORE

This VSP sets the minimum radar quality-confidence-Doppler score value required to create a radar “triple” from three uncorrelated radar reports that meet most tests for track initiation but have insufficient score (i.e. less than the normal radar track-initiation SINIT VSP) or movement. This VSP is normally set lower than SINIT in order to allow weaker (lower-scoring) reports to form “triples” when they would not have been good enough to initiate tracks. This VSP is input in units of “score points.” The nominal value is 6 points.

TRIPLE_PPS_EXT

This VSP sets the minimum threshold for a radar “triple” “points/scan” value if the current uncorrelated radar report under consideration were to be added to the “triple” thus extending it to the next scan. Each “triple” computes its “score” (based on report quality, confidence, and Doppler agreement – see the ELIG VSP table below) every scan that the “triple” is extended. The average “points/scan” value is maintained as a measure of prospective track “goodness.” If the score drops below the VSP, then the current radar report is not used to extend the “triple.” This VSP is input in 10ths of a score “point.” A nominal value for this VSP is 15 (1.5 points per scan).

TRIPLE_PPS_SCORE

This VSP sets the minimum threshold for a radar “triple” “points/scan” value if the current uncorrelated radar report under consideration were to join two other radar reports from previous scans to create a “triple” on this scan. Each “triple” computes its “score” (based on report quality, confidence, and Doppler agreement – see the ELIG VSP table below) every scan that the “triple” is extended. The average “points/scan” value is maintained as a measure of prospective track “goodness.” If the score drops below the VSP, then the current radar report is not used to extend the “triple.” This VSP is input in 10ths of a score “point.” A nominal value for this VSP is 16 (1.6 points per scan). (Note: this test is in addition to the TRIPLE_MIN_SCORE test above.)

TRIPLE_SCANS

This VSP sets the maximum number of scans that a radar “triple” may be extended without initiating a track. “Triples” whose life exceeds the TRIPLE_SCANS VSP value are removed from the system. This mechanism keeps triples formed from radar clutter from "cluttering up" the "triple" algorithm processing and potentially causing false track initiations. The nominal value for this VSP is 8 scans.

Quality=0, Confidence=3 Radar Track Initiation Controls

These VSPs control radar track initiation logic that enables and disables the use of uncorrelated radar reports with quality=0 and confidence=3 (these reports generally indicate clutter) in radar track initiation processing. The tracker maintains a continuous count of the total number of radar reports with quality=0 and confidence=3 in the system as a per-scan value (averaged over the preceding 5 scans).

Q0C3MAX

If the per-scan average count of radar reports with quality=0 and confidence=3 in a given 30-degree azimuth sector exceeds the C3MAX VSP, then radar track initiation using such reports is disabled (for that sector). This will reduce the incidence of false tracks that would initiate on clutter. This VSP would be set to "infinity" (100000) to turn off the special logic. The nominal value for this VSP is 3 reports.

Q0C3PCT

This VSP is used to control when radar track initiation for uncorrelated radar reports with quality=0 and confidence=3 in a given 30-degree azimuth sector is re-enabled after it has been disabled (see Q0C3MAX above). Q0C3PCT is given as a percentage. If the per-scan average count of radar reports with quality=0 and confidence=3 (in the given azimuth sector) drops below Q0C3PCT of Q0C3MAX, then radar track initiation on such reports is re-enabled. A nominal value for this VSP would be 100 percent. (Note: this VSP provides for a controllable amount of hysteresis in the enable/disable logic.)

"ON-THE-GROUND" REGIONS

These VSPs allow the provision of up to 8 special regions (defined in terms of range-azimuth "wedges") where the number of coasts allowed on a track is reduced from 3 to 1 if it is determined that the tracked aircraft is "on the ground." These regions generally define airport locations. The "on the ground" state is determined for tracks that are mature (age ≥ 10 scans) and their most-recent valid beacon altitude update was below 2 flight levels above the HEIGHT VSP.

GND_WEDGE_CNT

This VSP defines how many special regions are currently defined. Up to 8 regions may be defined. The nominal value for this VSP is 0 (no special regions).

GNDWEDGES

The special "on the ground" regions are defined as range-azimuth "wedges." Each region is defined with four values, as described below.

Min_range: The innermost range for the special region. The range is given in 64ths of a nautical mile units. The minimum range can range from 0 to the value for the maximum range.

Max_range: The outermost range for the special region. The range is given in 64ths of a nautical mile units. The maximum range must be larger than the minimum range and less than the radar's maximum range (60 nautical miles).

Min_azimuth: The lowest azimuth for the special region. The azimuth is given in ACPs.

Max_azimuth: The highest azimuth for the special region. The azimuth is given in ACPs.

APPENDIX C. PERFORMANCE MONITORS

Table 9 below details the various performance monitor values which are derived by the 9-PAC tracker each scan. The performance monitor values provide statistical measurements on the performance of various processes within the tracker. They provide a means to evaluate the impact of changes in VSP settings or unusual environmental conditions.

Table 9. Performance Monitors

Name	Description
Total_tracks	Total # of tracks this scan.
RO_tracks	# of radar-only tracks this scan.
BO_tracks	# of beacon-only tracks this scan.
RB_tracks	# of radar-reinforced tracks this scan.
Init_tracks	# of initiating tracks (Track State=Init, Miss, Non-Move, Stable) this scan.
NonMoving_tracks	# of non-move tracks (Track State = Non-Move 1, Non-Move 2) this scan.
Coasting_tracks	# of coasting tracks (Track State = Coast 1, Coast 2, Coast 3) this scan.
Stable_tracks	# of stable tracks (Track State = Stable) this scan.
Total_Corr_tracks	# of correlating tracks this scan.
OneToOne_corr	# of one-to-one correlations this scan.
ManyToOne_corr	# of many-to-one correlations this scan.
ManyToMany_corr	# of many-to-many correlations this scan.
Radar_only_inputs	# of radar-only input reports this scan.
Beacon_only_inputs	# of beacon-only input reports this scan.
Radar_Beacon_inputs	# of radar-reinforced beacon input reports this scan.
Total_hit_ratio	Hit ratio (%) for all tracks this scan.
Radar_hit_ratio	Hit ratio (%) for radar-only tracks this scan.
RB_hit_ratio	Hit ratio (%) for radar-reinforced beacon tracks this scan.
Beacon_hit_ratio	Hit ratio (%) for beacon-only tracks this scan.
Rcorr_report_count	# of correlated radar-only reports this scan.
Total_triples	# of radar "triple" reports in initiation processing this scan.
Drop_linearity_count	# of tracks dropped due to maintenance linearity test failure this scan.
Drop_scoring_count	# of tracks dropped due to points/scan test failure this scan.
Q0C3_init_disable	Flag indicating that the number of radar-only quality-0, confidence=3 reports has exceeded the allowable initiation threshold VSP (if TRUE).
Fast_Tracks	# of radar-only tracks with speed > SLOW_VEL VSP this scan.
Fast_Avg_Age	Average age (scans) for radar-only tracks with speed > SLOW_VEL VSP
Fast_PD	Probability of detection (%) for fast radar-only tracks this scan.
Slow_Tracks	# of radar-only tracks with speed ≤ SLOW_VEL VSP this scan.
Slow_Avg_Age	Average age (scans) for radar-only tracks with speed ≤ SLOW_VEL VSP
Slow_PD	Probability of detection (%) for slow radar-only tracks this scan.
Hdg_Trk_Count	# of heading "triads" that exceed TRK_HDG_CNT VSP this scan.
Hdg_On_Count	# of heading "triads" that exceed TRK_ON_CNT VSP
Hdg_Max_Count	Maximum heading bin "triad" count this scan.
Bird_Flag	Flag indicating that the "bird track" feedback is in effect this scan.
Trk_Overload	Flag indicating that a system resource (i.e. track file entries, uncorrelated reports, buffer reports) has been exceeded (if TRUE).

REFERENCES

[1] G. R. Elkin and J. L. Gertz, "The Beacon Target Detector (BTD) Algorithms Deployed in the ASR-9 Processor Augmentation Card (9-PAC)," Lexington, MA, MIT Lincoln Laboratory Project Report ATC-288, 5 September 2000.

[2] G. R. Elkin, "The Radar Correlation and Interpolation (C&I) Algorithms Deployed in the ASR-9 Processor Augmentation Card (9-PAC)," Lexington, MA, MIT Lincoln Laboratory Project Report ATC-297, (under review).

[3] Department of Transportation Federal Aviation Administration "Specification for the Airport Surveillance Radar (ASR-9)," FAA-E-2704B, 1 October 1986.

[4] "Software System/Subsystem Specification Surveillance Processor for the ASR-9 Airport Surveillance Radar," Westinghouse Electric Corporation, Baltimore, Maryland, 1989.

[5] Burgiois, F., and J.-C. Lassalle, "An Extension of the Munkres Algorithm for the Assignment Problem to Rectangular Matrices," *Communication of the ACM*, Vol. 14, Dec. 1971, pp. 802-806.

[6] J. L. Gertz, "The ATCRBS Mode of DABS," Lexington, MA, MIT Lincoln Laboratory Project Report ATC-65, 31 January 1977.

[7] J. B. Evans, "ASR-9 Processor Augmentation Card Scan-Scan Correlator Algorithms," MIT Lincoln Laboratory Project Report ATC-245, 2 April 1996.