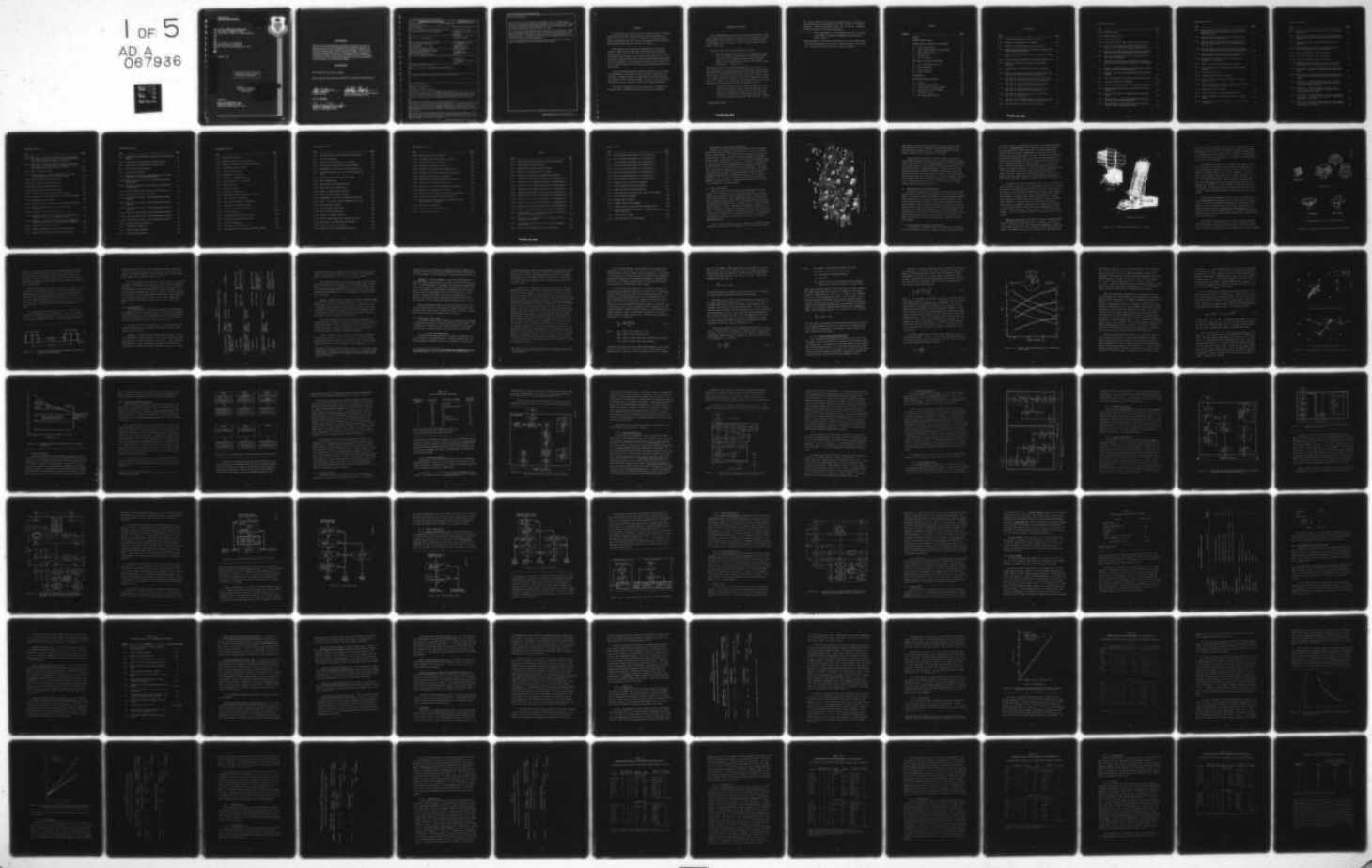
AD-A067 936 GENERAL RESEARCH CORP SANTA BARBARA CALIF F/G 17/9
TACTICAL FORWARD AREA SURVEILLANCE AND CONTROL INTERFITTING STU--ECT.(U)
NOV 78 G W DELEY, J H BALLANTINE F19628-78-C-0020
UNCLASSIFIED CR-1-814-VOL-2 ESD-TR-78-180-VOL-2 N/L

1 OF 5
AD A
067936



ESD-TR-78-180

AD A067936

TACTICAL FORWARD AREA SURVEILLANCE AND CONTROL INTERNETTING STUDY
Final Report, Volume II, Technical Details

G. W. Deley, J. H. Ballantine
General Research Corporation
P.O. Box 6770, Santa Barbara, CA 93111

November 1978

Approved for Public Release;
Distribution Unlimited.

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U.S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA. 22161

Prepared for

DEPUTY FOR DEVELOPMENT PLANS
ELECTRONIC SYSTEMS DIVISION
HANSCOM AIR FORCE BASE, MA 01731

388

LEGAL NOTICE

When U. S. Government drawings, specifications or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

OTHER NOTICES

Do not return this copy. Retain or destroy.

This technical report has been reviewed and is approved for publication.

ALFRED F. ANDERSON
Project Engineer

ANTHONY NAPOLI, Colonel, USAF
Director, Tactical C^3I Systems Planning
Deputy for Development Plans

FOR THE COMMANDER

MICHAEL H. ALEXANDER, Colonel, USAF
Deputy for Development Plans

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ESD-TR-78-180 Vol II	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Tactical Forward Area Surveillance and Control Internetting Study Technical Details		5. TYPE OF REPORT & PERIOD COVERED Final Contract Report 6. PERFORMING ORG. REPORT NUMBER CR-1-814
7. AUTHOR(s) G.W. Deley, J.H. Ballantine		8. CONTRACT OR GRANT NUMBER(s) F19628-78-C-0020
9. PERFORMING ORGANIZATION NAME AND ADDRESS General Research Corporation P.O. Box 6770 Santa Barbara, Calif. 93111		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS P.E. 63101F Project No. E251 Task Area No. 0 Work Unit No. 0
11. CONTROLLING OFFICE NAME AND ADDRESS Electronic Systems Division, AFSC Hanscom AFB, MA 01731 Attn: A.F. Anderson, ESD/XRT		12. REPORT DATE November 1978 13. NUMBER OF PAGES 389
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Published in two volumes
Volume I: Summary
Volume II: Technical Details

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Tactical Air Surveillance, Track-While-Scan Radar, Phased-Array Radar, Netted Systems, Distributed Data Processing, Communications, Real-Time Tracking Algorithms, Track Initiation, Track Association (correlation), Track Filtering

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A tactical air surveillance and control system concept for tracking large numbers of aircraft, many of which are highly maneuverable and capable of flying at low altitudes, is analyzed. The survivable, automated, non-hierarchically netted system investigated provides continuous, non-redundant System Tracks for tactical air control.

Nine alternative netted system configurations are defined and analyzed. These differ by radar mode (track-while-scan or computer-directed-track), (Continued)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20 (Continued)

track file structure and location, and whether or not all radars or only selected ones are used to track each aircraft. Flow charts, and communications and data processing requirements are developed for each configuration. Some versions are simulated in the TACRAN (Tactical Air Control Radar Net) distributed system simulation developed during the study.

The real-time data processing functions required by such a system are investigated, with emphasis on track initiation, association, and filtering. Parametric analyses of algorithms for these functions provide measures of their performance under a variety of conditions and also determine the accuracy and rates required of the radar measurements.

The results demonstrate that the system concept is feasible with reasonable radar, communications, and data processing requirements.

ABSTRACT

A tactical air surveillance and control system concept for tracking large numbers of aircraft, many of which are highly maneuverable and capable of flying at low altitudes, is analyzed. The survivable, automated, non-hierarchically netted system investigated provides continuous, non-redundant System Tracks for tactical air control.

Nine alternative netted system configurations are defined and analyzed. These differ by radar mode (track-while-scan or computer-directed-track), track file structure and location, and whether or not all radars or only selected ones are used to track each aircraft. Flow charts, and communications and data processing requirements are developed for each configuration. Some versions are simulated in the TACRAN (Tactical Air Control Radar Net) distributed system simulation developed during the study.

The real-time data processing functions required by such a system are investigated, with emphasis on track initiation, association, and filtering. Parametric analyses of algorithms for these functions provide measures of their performance under a variety of conditions and also determine the accuracy and rates required of the radar measurements.

The results demonstrate that the system concept is feasible with reasonable radar, communications, and data processing requirements.

INTRODUCTION TO VOLUME 2

This volume describes in detail the analyses and results of a study of an automated, netted tactical forward area air surveillance and control system concept that has potential for operating in the complex tactical air environment of the near future. Volume 1 contains background and a summary of the study.

The three study tasks can be summarized as follows:

1) Define and analyze alternative system configurations to determine the tradeoffs, with emphasis on the different possible locations for performing the various tracking functions. Define the data processing and communications loads for each *configuration*. [Tasks 3, 4][1]

The detailed results of this task are presented in Secs. 1 and 3. In Sec. 1, the basic netted-system concept is described and various alternative radar and system configurations based on this concept are described and analyzed. Three system variations were simulated on a digital computer to further understand their performance and to demonstrate the viability of the concepts. The simulation, called TACRAN (Tactical Air Control Radar Net), and the simulation results are presented in Sec. 3.

2. Describe the real-time data processing functions to be performed, and recommend solutions to any problem areas. Functions to be considered in detail include track initiation, association (correlation) of measurements with tracks, association of tracks with tracks, and track filtering. [Task 1]

---

[1] Original task numbers.

Preceding page blank

The detailed results of this task are presented in Sec. 2. Several of the algorithms were incorporated into the three versions of the TACRAN simulation. These are described in Secs. 3.3, 3.4, and 3.5. Simulation results are given for the tracking algorithms.

3. Define requirements on the information that must be provided by the radars to support an automated netted surveillance system. [Task 2]

Results pertinent to this task are presented in Sec. 1.2 on Radar Characteristics and Requirements, and in Sec. 2, where studies on track initiation, association (correlation) and filtering are presented.

CONTENTS

SECTION		PAGE
	ABSTRACT	3
	INTRODUCTION TO VOLUME 2	5
1	SYSTEM CONCEPT, ALTERNATIVES, AND ANALYSES	21
	1.1 Basic System Concept	21
	1.2 Radar Characteristics and Requirements	23
	1.3 System Configurations	42
	1.4 Communications	63
	1.5 Data Processing	99
2	REAL-TIME DATA PROCESSING FUNCTIONS	121
	2.1 List of Real-Time Functions	121
	2.2 Track Initiation	128
	2.3 Track Association	152
	2.4 Track Filtering	206
	BIBLIOGRAPHY	245
3	DISTRIBUTED NETWORK SIMULATION	247
	3.1 Overview	247
	3.2 TACRAN Simulation Models	249
	3.3 Simple-Algorithm System (TACRAN1)	265
	3.4 Kalman-Filter System (TACRAN2)	277
	3.5 Pooled-Data System	291
	3.6 Simulation System Details	375

ILLUSTRATIONS

NO.		PAGE
1.1.1	Internetted Tactical Air Surveillance System	22
1.2.1	Examples of Track-While-Scan 3-D Radars	25
1.2.2	Examples of Computer-Directed Track 3-D Radars	27
1.2.3	Possible Tracking-Measurement Timing With a Mechanically Scanned Planar Phased Array	28
1.2.4	Relative Range and Coverage Area of a Single-Face Phased Array	38
1.2.5	Tracking Geometries Idealized Deployment--Hexagonal Grid	41
1.2.6	Maximum Low-Altitude Tracking Range for Idealized Deployment	42
1.3.1	Six Variations of Track File Structure at the Radar Sites	44
1.3.2	Flow Chart for TWS Operation With a LTF and STF at Each Radar and All Radars Tracking (Configuration 1)	47
1.3.3	Flow Chart for TWS Operation With DLTF and STF at Each Radar and Selected Radars Tracking (Configuration 2)	49
1.3.4	Flow Chart for TWS Operation With LTF Only at Each Radar and All Radars Tracking (Configuration 3)	52
1.3.5	Flow Chart for TWS Operation With STF Only at Each Radar and All Radars Tracking (Configuration 4)	54
1.3.6	Flow Chart for TWS Operation With No Track Files at the Radars (Configuration 5)	55
1.3.7	Flow Chart for CDT Operation With an LTF and STF at Each Radar and Selected Radars Tracking (Configuration 6)	56
1.3.8	Association-With-STF Logic (STF at Each Radar)	58

Preceding page blank

ILLUSTRATIONS (Cont.)

NO.		PAGE
1.3.9	Add-Tracker Logic	59
1.3.10	Primary-Tracker Logic	60
1.3.11	Other-Tracker Logic	61
1.3.12	Association With STF Logic (Partial STF at Each Radar)	62
1.3.13	Flow Chart for CDT Operation With STF Only at Each Radar and Selected Radars Tracking (Configuration 9)	64
1.4.1	Data Rate on Incoming Link Versus Fraction of Tracking Radars Sending STF Update Messages in Configuration 1	80
1.4.2	Data Rate on Incoming Link Versus Time Between STF Updates for Configuration 2	83
1.4.3	Data Rate on Incoming Link Versus Fraction of Measurements for Which a Distance is Requested in Configuration 2	84
1.4.4	Example of Communication Rate Versus Number of Aircraft One-Way Data Rate Into Isolated Node	98
1.5.1	Example of Data Processing Requirements Versus Number of Aircraft	119
2.1.1	Track Loop Diagram in a Computer-Directed-Track System	124
2.2.1	Assumed Association Cell Geometry	133
2.2.2	Plan View of Association Cell With Doppler Measurements	134
2.2.3	Geometry Showing Bias Terms in Association-Cell Size Equations	137
2.2.4	Vertical Cross Section of Assumed Radar Surveillance Volume	139
2.2.5	Expected Number of False-Alarm Track Initiations per Scan With Two Position Measurements	141
2.2.6	Expected Number of False-Alarm Track Initiations per Scan With Three Equally Spaced Pulses	141

ILLUSTRATIONS (Cont.)

NO.		PAGE
2.2.7	Expected Number of False-Alarm Track Initiations per Scan With Three-Pulse Algorithms as a Function of the Measurement Error	144
2.2.8	Expected Number of False-Alarm Track Initiations per Scan With Two Measurement Pairs	145
2.2.9	Expected Number of False-Alarm Track Initiations per Scan With a Measurement Pair Plus One Measurement on the Next Scan	145
2.2.10	Expected Number of False-Alarm Track Initiations per Scan With Two Position and Radial-Velocity Measurements	146
2.2.11	Exepcted Number of False-Alarm Track Initiations per Scan--Comparison of the Algorithms Considered	148
2.2.12	Probability of Successful Track Initiation Versus Probability of Detection	150
2.3.1	Simple Closest-Pair Actual-Distance Algorithms	154
2.3.2	Two-Dimensional Association Example for Uncorrelated Components	156
2.3.3	Chi-Square Distribution	158
2.3.4	Multiple-Target Association Geometry	160
2.3.5	Matrix of Association Distances-Squared for Geometry of Fig. 2.3.4	161
2.3.6	Growth of Association Volume for Two Different Timings of the Measurements by Multiple Radars	163
2.3.7	Maneuvering Aircraft Geometry	165
2.3.8	Flight Envelope of a High-Performance Fighter	168
2.3.9	Effect of Maximum Sustained Accelerations for Human Pilots	168
2.3.10	Projection of Association Volume Due to Possible Target Acceleration	170

ILLUSTRATIONS (Cont.)

NO.		PAGE
2.3.11	Projection of Association Volume Due to Possible Target Acceleration	171
2.3.12	Relative Acceleration Capability of a High Performance Fighter/Bomber Along and Perpendicular to the Direction of Flight	172
2.3.13	Least-Square Filtering and Prediction Geometry for Association	173
2.3.14	Error in Predicted Position for Linear Extrapolation From a Quadratic Fit to N Measurements	175
2.3.15	Association Volume for Track Association	177
2.3.16	Association Volume for Track Association	178
2.3.17	Association Volume for Track Association	180
2.3.18	Quadratic Fit to Three Points on a Circular Path and the Extrapolation of These Curves and Their Tangents	181
2.3.19	Model Error: Difference Between Constant 3g Acceleration Circle and Quadratic Prediction, One Sample Time Ahead	182
2.3.20	Model Error: Difference Between Constant 10g Acceleration Circle and Quadratic Prediction, One Sample Time Ahead	182
2.3.21	Model Error: Difference Between Tangent to Constant 3g Acceleration Circle and Tangent to Quadratic Prediction One Sample Time Ahead	183
2.3.22	Model Error: Difference Between Tangent to 10g Acceleration Circle and Tangent to Quadratic Prediction One Sample Time Ahead	184
2.3.23	Model Error: Difference Between Constant 3g Acceleration Circle and Quadratic Prediction for a Time Between Measurements of 2 Seconds	184
2.3.24	Model Error: Difference Between Constant 10g Acceleration Circle and Quadratic Prediction for a Time Between Measurements of 2 Seconds	185

ILLUSTRATIONS (Cont.)

NO.		PAGE
2.3.25	Model Error: Difference Between Tangent to Constant 3g Acceleration Circle and Tangent to Quadratic Prediction for a Time Between Measurements of 2 Seconds	186
2.3.26	Model Error: Difference Between Tangent to Constant 10g Acceleration Circle and Tangent to Quadratic Prediction for a Time Between Measurements of 2 Seconds	186
2.3.27	Error Distribution Model	188
2.3.28	Probability That a Position Measurement is Less Than a Given Distance From its Predicted Location	189
2.3.29	False Association With Random Noise	192
2.3.30	False Association With Random Noise	192
2.3.31	False Association With Random Noise	193
2.3.32	False Association With Other Targets	194
2.3.33	Track Association With Closely Spaced Targets	196
2.3.34	Probability of False Association--Closely Spaced Targets	198
2.3.35	Maneuvering Targets Geometry	199
2.3.36	Target Geometry for One-Dimensional Association Analysis	200
2.3.37	Assumed Probability Density Function of Target Acceleration	201
2.3.38	Probability Density Function of $y_d = y_2 - y_1$	202
2.3.39	Probability of Incorrect Association of Two Targets and Tracks as a Function of the Target Separation	203
2.3.40	Probability of Incorrect Association of Two Targets and Tracks as a Function of the Time Between Measurements	204
2.4.1	Kalman Filtering (Discrete Form)	213
2.4.2	Example of Combining Two Tracks Using Covariances	219
2.4.3	Kalman Filter Tracking With 4g of Maneuver Noise	220

ILLUSTRATIONS (Cont.)

NO.		PAGE
2.4.4	Enlargement of Kalman Filter Tracking With 4g of Maneuver Noise	220
2.4.5	Kalman Filter Tracking With 8g of Maneuver Noise	221
2.4.6	Kalman Filter Tracking With No Maneuver Noise	221
2.4.7	Kalman Filter Tracking Example	223
2.4.8	α-β Filter Illustration	229
2.4.9	Extrapolation Error Due to Measurement Error for Least-Squares Fit to Second-Degree Polynomial	236
2.4.10	Model and Maneuver Error in Second-Degree Polynomial Extrapolation	237
2.4.11	Extrapolation Methods for Second-Degree Polynomial Fit to Three Points	238
2.4.12	Distributed Local Track Performance for Aircraft No. 3	238
2.4.13	Trajectory for High-g Tracking Runs	240
2.4.14	Track With Five Points, 2 Second Measurement Interval (Case 1)	241
2.4.15	Track With Five Points, 2 Second Measurement Interval (Case 2)	241
2.4.16	Track With Five Points, 2 Second Measurement Interval (Case 3)	242
2.4.17	Track With Three Points, 2 Second Measurement Interval	242
2.4.18	Track With Seven Points, 2 Second Measurement Interval	243
2.4.19	Track With Five Points, 1 Second Measurement Interval	244
3.1.1	Distributed Network Simulation	248
3.2.1	TACRAN Radar Simulation	250
3.2.2	Aircraft Flight Simulation	252
3.2.3	Communications Simulation	254

ILLUSTRATIONS (Cont.)

NO.		PAGE
3.2.4	Route Table Structure	256
3.2.5	Example of Addressed Message Routing	257
3.2.6	Sending of Unaddressed System Route Messages	259
3.2.7	Main Communications Routine	260
3.2.8	Start a Not-Busy Link	262
3.2.9	Management of a Busy Link	263
3.3.1	Flow Chart of TACRAN1 Simulation	266
3.3.2	Association With Local Track File	268
3.3.3	Track Initiation	269
3.3.4	Local Track Update	270
3.3.5	System Track Update Test	272
3.3.6	Update System Track File From Local Track	273
3.3.7	"Son" and "Dad" Relationship	275
3.3.8	Update System Track File From Message	276
3.4.1	Flow Chart of TACRAN2 Simulation	279
3.4.2	Association With Local Track File	280
3.4.3	Local Track Update	284
3.4.4	Update System Track From Local Track	288
3.4.5	Association With System Track File	289
3.5.1	Track File Structure	294
3.5.2	Flow Chart of TACRAN3 Simulation	295
3.5.3	Main Logic Flow for Processing a New Radar Return	299

ILLUSTRATIONS (Cont.)

NO.		PAGE
3.5.4	Association With the Distributed Local Track File	301
3.5.5	DLT Track Processing	302
3.5.6	Should This Node Continue to be a Tracker?	304
3.5.7	Next Expected Reception Times and Time Window	307
3.5.8	Determination of Best Set of Reception Times (I)	308
3.5.9	A "Goodness" Measure for a Set of Tracker Reception Times	309
3.5.10	Distributed Local and System Track Update	311
3.5.11	"Help" Message Logic	314
3.5.12	Association With the Non-Track File	315
3.5.13	Association With a System Track	316
3.5.14	Detail: Get Distances as Required	318
3.5.15	Should This Node Become a Tracker?	320
3.5.16	Determination of Best Set of Reception Times (II)	321
3.5.17	Detail: Get a Copy of the Distributed Local Track	322
3.5.18	Track Initiation Sequence	324
3.5.19	Track Initiation (Three-Scan)	325
3.5.20	Track Initiation (Two-Scan)	326
3.5.21	Geometry for TACRAN3 Test Run 1	327
3.5.22	Aircraft Ranges Versus Time, TACRAN3, Test Run 1	329
3.5.23	Distributed Local Track History, Aircraft 1	330
3.5.24	System Track History, Aircraft 1	331
3.5.25	Distributed Local Track History, Aircraft 2	332

ILLUSTRATIONS (Concl.)

NO.		PAGE
3.5.26	System Track History, Aircraft 2	333
3.5.27	Distributed Local Track History, Aircraft 3	334
3.5.28	System Track History, Aircraft 3	335
3.5.29	Detailed Track History, Aircraft 3	337
3.5.30	Geometry for Weighted-Filter TACRAN3 Runs	338
3.5.31	Distributed Local Track Performance on Aircraft 1	339
3.5.32	System Track Performance on Aircraft 1	340
3.5.33	Distributed Local Track Performance on Aircraft 2	341
3.5.34	System Track Performance on Aircraft 2	342
3.5.35	Distributed Local Track Performance on Aircraft 3	343
3.5.36	System Track Performance on Aircraft 3	344
3.5.37	Distributed Local Track Performance for Aircraft No. 3	345
3.5.38	System Track Performance for Aircraft No. 3	346
3.5.39	Three-Measurement (Scan) Track Initiations Per Scan Versus Association Distance	350
3.5.40	Geometry for 15-Node Run	351
3.5.41	Track Initiation of Aircraft 1 and 3	352
3.6.1	List Internal Linkages	380

TABLES

NO.		PAGE
1.2.1	Radar Antenna Configurations and Operational Concepts	30
1.3.1	System Configurations Considered	46
1.4.1	Message Header Components and Bit Lengths	67
1.4.2	Message Text Components and Bit Lengths	68
1.4.3	Parameters Involved in the Communications Analysis	71
1.4.4	Communications Data-Rate Requirements for Configuration 1	77
1.4.5	Communications Data-Rate Requirements for Configuration 2	81
1.4.6	Communications Data-Rate Requirements for Configuration 3	85
1.4.7	Communications Data-Rate Requirements for Configuration 4	87
1.4.8	Communications Data-Rate Requirements for Configuration 5	89
1.4.9	Communications Data-Rate Requirements for Configuration 6	90
1.4.10	Communications Data-Rate Requirements for Configuration 7	92
1.4.11	Communications Data-Rate Requirements for Configuration 8	94
1.4.12	Communications Data-Rate Requirements for Configuration 9	96
1.4.13	Summary of Average Communications Data-Rate Requirements	97
1.5.1	Estimated Numbers of Instructions Required to Perform the Major Data-Processing Operations	102
1.5.2	Parameters in the Expressions for the Number of Data-Processing Instructions	103
1.5.3	New Parameters Involved in the Data Processing Requirements Analysis	105
1.5.4	Data Processing Requirements for Configuration 1	108

Preceding page blank

TABLES (Concl.)

NO.		PAGE
1.5.5	Data Processing Requirements for Configuration 2	109
1.5.6	Data Processing Requirements for Configuration 3	110
1.5.7	Data Processing Requirements for Configuration 4	111
1.5.8	Data Processing Requirements for Configuration 5	112
1.5.9	Data Processing Requirements for Configuration 6	113
1.5.10	Data Processing Requirements for Configuration 7	114
1.5.11	Data Processing Requirements for Configuration 8	115
1.5.12	Data Processing Requirements for Configuration 9	116
1.5.13	Summary of Data Processing Requirements	118
2.2.1	Track Initiation Algorithms Considered	129
2.4.1	Kalman Filtering Covariance Equations	218
2.4.2	Measurements Used in Kalman Filter Runs (from TACRAN3)	222
3.2.1	Radar Simulation Parameters	251
3.5.1	Message Types in TACRAN3	298
3.5.2	TACRAN3 False-Alarm Run Summary	348
3.5.3	Sustained Tracks From False Alarms (TACRAN3 Run 9)	349
3.5.4	Data Structure for Third Version of TACRAN--Tactical Air Control Radar Net	355
3.5.5	Input Data for TACRAN3	362
3.5.6	Post Processor Output From TACRAN3	371

1 SYSTEM CONCEPT, ALTERNATIVES, AND ANALYSES

This section contains descriptions and analyses of the alternative system configurations considered during the study. Section 1.1 defines the basic system concept. The types of radars suitable for tactical air surveillance and control applications and the performance requirements of these radars are discussed in Sec. 1.2. The antenna configuration and its operational implications are emphasized. For each of two basic modes of radar operation, several alternative system concepts are defined in Sec. 1.3 in terms of the track file structure and locations. The operation of each of these configurations is described by means of flow charts or top-level logic diagrams. The communications and data processing requirements for each of the systems are analyzed in Secs. 1.4 and 1.5, respectively, and the results are presented in the form of parametric equations. Numerical values of bandwidths and execution rates are given for an example set of parameter values.

1.1 BASIC SYSTEM CONCEPT

The basic concept for the tactical air surveillance and control system under consideration in this study, depicted in Fig. 1.1.1, was conceived at the Air Force Systems Command's Electronic Systems Division and the MITRE Corporation. Many highly mobile radars are internetted with an average spacing of about 30 km to provide low-altitude coverage and line-of-sight communications. The maximum range of the radars (80-90 km) is sufficient to provide considerable overlapping coverage; up to 25 radars can see each aircraft (other than those that are at very low altitudes). To the extent possible, each radar site is connected to its three or four nearest neighbors in a non-hierarchical network. Air operations are planned and executed at a second type of site, which is called an Operations Facility.

Such a system is designed to operate against large numbers of high-performance aircraft, many of which may be flying at very low altitudes.



Figure 1.1.1. Internetted Tactical Air Surveillance System

Other threats (not directly considered in this study) that the system must effectively counter include ground and airborne jammers, anti-radiation missiles (ARMs), and chaff. Advanced targets will include remotely piloted vehicles (RPVs) and cruise missiles.

The detection and tracking is entirely automated to handle the large numbers of aircraft that must be simultaneously tracked. The basic output of the system is a file of System Tracks, which are representations of the flight paths of all the aircraft which have been detected in the surveillance volume. A System Track File is maintained at each of the Operations Facilities, where it is suitably displayed and used in planning and carrying out air operations, and in most of the system alternatives at each radar site in the system. To the extent possible, all copies of the System Track File are maintained identically. The System Track File, or information extracted from it, may also be relayed to higher echelon planning and operations centers.

1.2 RADAR CHARACTERISTICS AND REQUIREMENTS

The design and operation of a tactical air surveillance and control system depends in part on the characteristics of the radars used in the system. To provide a basis for analyzing the data requirements and assessing the feasibility of the various possible system configurations, different radar types were considered. The differences between the radars involve primarily their antenna configuration, particularly the means by which the antenna beams are scanned, and their mode of operation for performing the surveillance, track initiation, and tracking functions. A categorization of radar types based on these differences and their operational implications is described in this section. Some aspects of the radar power requirements and the surveillance and tracking coverage are also discussed.

1.2.1 Operating Modes and Antenna Configurations

In performing the search and tracking functions that are essential in an air surveillance and control system, the radars can operate in one

of two basic modes: (1) track-while-scan (TWS), or (2) computer-directed-track (CDT). Track-while-scan (TWS) radars use search measurements for both search and track. These radars, which are usually mechanically rotated in azimuth, provide a measurement interval equal to the scan period for all tracking and search functions. Some of these radars have no resolution in elevation (these are called "2-D" radars because they measure only the dimensions of range and azimuth); 2-D radars were given little consideration during the study as required by the Statement of Work. Others have resolution in elevation provided by several fixed beams or by electronic steering. These latter radars are sometimes called "3-D" radars because they measure range, azimuth, and elevation (or height). Generally the fourth radar dimension, Doppler velocity, is not measured with such radars because it is highly ambiguous and provides little useful additional tracking information (more on this in Sec. 1.2.4).

Two examples of TWS radars are shown in Fig. 1.2.1. In the AN/TPS-43(E) a cluster of feed horns is used to provide a number of beams stacked in elevation to provide altitude information. Both search and track are performed at the search rate, and measurements used for track initiation are separated by the scan period. With a planar phased array, multiple beams stacked in elevation can be formed electronically, or a single pencil beam can be scanned rapidly in elevation as is done in the Marine Corp's AN/TPS-59. In addition, the set of beams could be scanned a limited amount in azimuth--or additional sets of beams could be formed at other azimuth angles--to make closely spaced measurements for track initiation and to provide some timing flexibility for the tracking measurements. The basic tracking rate would still be equal to the scan rate, however.

Computer-directed-track (CDT) radars use special track measurements that are scheduled by the data processor for tracking. The usual CDT radar is a stationary phased array with complete flexibility in surveillance, track initiation, and track scheduling, subject only to power limitations. The average scan rate can be relatively low with higher rates



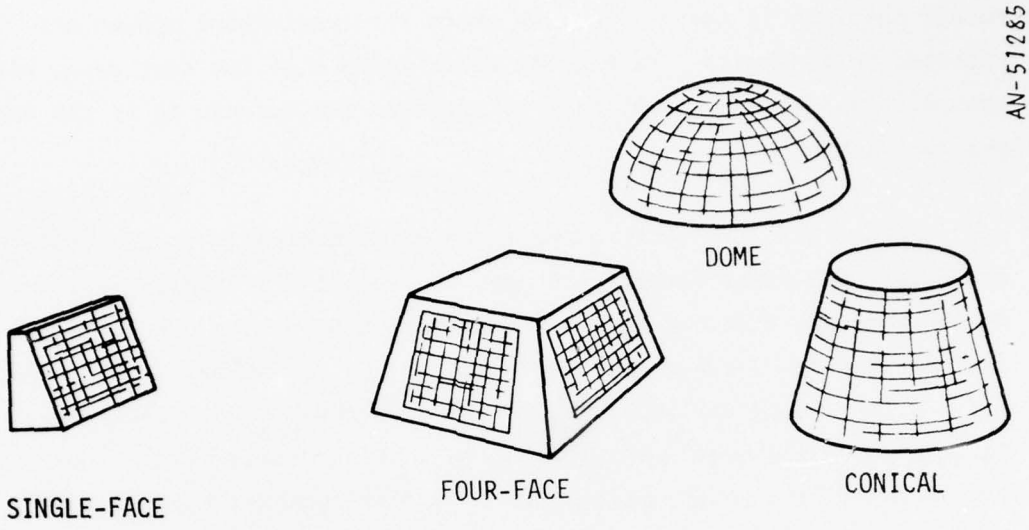
Figure 1.2.1. Examples of Track-While-Scan 3-D Radars

in critical regions as needed, track initiation can be performed with a pulse pair or series of pulses with appropriate spacings, and tracking measurements can be made at any rate needed to maintain a high association probability and using a waveform that depends upon the characteristics of the target and its environment. Such a radar is characterized by the need for energy management.

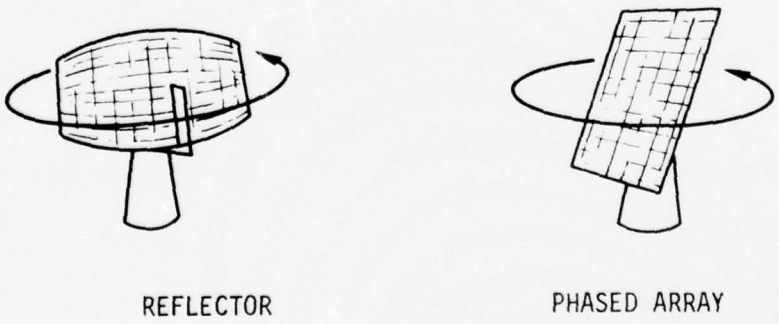
The coverage of a planar array is limited by its maximum off-boresight scan angle, but a network of properly deployed single-face arrays can provide complete coverage of a specified volume; the coverage of such radars is discussed further in Sec. 1.2.3.2. Examples of stationary phased array antennas are depicted in Fig. 1.2.2(a). Military radars which employ single-face arrays include the Army's SAM-D (Patriot) radar and its AN/TQP-36 and AN/TPQ-37 mortar and artillery location radars. Multi-face phased arrays and arrays of special designs can provide full hemispherical coverage, or at least 360-degree coverage in azimuth, by a single radar. The Army ballistic missile defense Site-Defense Radar and the Navy's AN/SPY-1 (Aegis) and AN/SPS-33 radars use four faces for this purpose.

A dome antenna configuration, using a single, active phased array plus a dome-shaped lens as implemented in several recent Sperry prototypes, could provide hemispherical coverage with acceptable beam conformations. A conical phased array, such as the one designed recently at Lincoln Laboratory for use in a multistatic ballistic-missile defense system, could also provide complete coverage. The multi-face and specially designed phased arrays are generally more complex and more expensive than single-face phased arrays.

An alternative computer-directed-track radar type has an antenna that is mechanically scanned in azimuth as is the track-while-scan radar, but at a higher rate (scan period of 1-3-seconds). The high scan rate makes it possible to perform search and tracking as more or less independent



a. Phased Arrays



b. Fast Mechanical Scan (1-3 seconds) in Azimuth

Figure 1.2.2. Examples of Computer-Directed Track 3-D Radars

functions. The average search rate can be considerably lower than the scan rate to conserve power since a high rate is not needed for search except possibly in certain regions where the acquisition ranges are expected to be short. Tracking is normally done at the scan rate, with transmissions scheduled for each target when the antenna is at the appropriate azimuth angle.

Figure 1.2.2(b) depicts two types of mechanically scanned CDT radars, the first with fixed elevation beams, the second electronically scanned. Existing radars that can be scanned at such a high rate are generally used for 2-D short-range applications. To design and build such radars for 3-D long range applications at high scan rates imposes structural and drive-team design requirements which must be carefully analyzed to determine feasibility. Wind effects are a serious consideration in such a design.

The rotating phased array depicted in Fig. 1.2.2(b) offers some flexibility in scheduling track measurements. In fact, it is possible to make two or three tracking measurements per scan with a phased array as indicated in the sketch of Fig. 1.2.3, thereby increasing the average

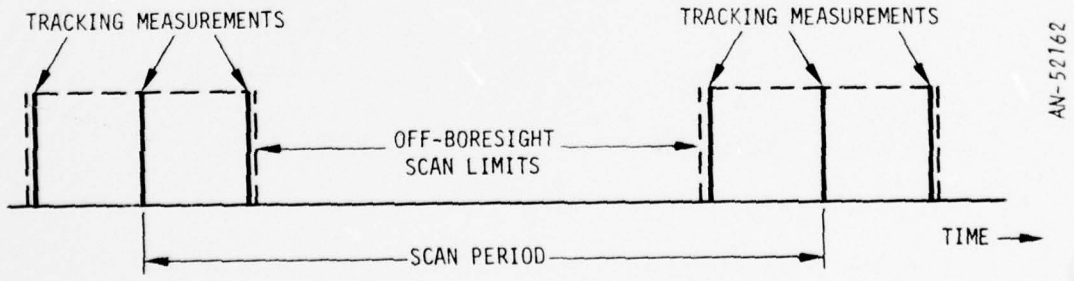


Figure 1.2.3. Possible Tracking-Measurement Timing With a Mechanically Scanned Planar Phased Array

tracking rate but with unequal times between measurements. With a scan time in the 2-to-3-second range, such an increase in the average tracking rate is probably necessary to enable a single radar to track a maneuvering target, as is discussed further in Sec. 2.3.

The characteristics of the three basic types of radars as categorized for use in a tactical air surveillance and control system are summarized in Table 1.2.1. For each radar type, major features of the surveillance, track initiation, and tracking operations are briefly described. The first two types are mechanically scanned in azimuth, the difference being the scan period. The slow scanned radar, defined as having scan periods of more than 4 seconds, must be operated in a TWS mode. Radars with faster scans and phased arrays can be operated in the preferred CDT mode.

1.2.2 Data Requirements

This section presents a qualitative discussion of what information is required from the radar signal processor to support a netted automated air surveillance system. More quantitative data on requirements such as measurement accuracies and rates are given in Sec. 2, where analyses of track initiation, association, and filtering are described.

The dimensions in which a single radar can provide information are position (range, azimuth, elevation), radial velocity (Doppler), and amplitude (cross section).

Position. The simplest pulsed radars provide only two of the positional dimensions: range and azimuth. Thus these 2-D radars do not provide elevation, from which altitude could be derived. For an automated system designed to track large numbers of targets, two dimensions are insufficient to provide reliable track initiation (see Sec. 2.2), unless

TABLE 1.2.1

RADAR ANTENNA CONFIGURATIONS AND OPERATIONAL CONCEPTS

Antenna Configuration	Operational Mode	Surveillance	Track Initiation	Tracking
1. Mechanically Scanned in Azimuth at Low Rate (4-12 s)	Track-While-Scan (TWS)	At scan rate		
A. Reflector			Successive scans	Fixed spacing at scan rate
B. Phased Array			Pairs on same scan	Adjustable timing within window
2. Mechanically Scanned in Azimuth at High Rate (1-3 s)	Computer-Directed-Track (CDT)	Slower than scan rate		
A. Reflector			Successive scans	Fixed spacing at scan rate
B. Phased Array			Pairs if needed	Somewhat higher rate possible with non-uniform spacing
3. Stationary Phased Array	Computer-Directed-Track (CDT)	Adaptable		
A. Single Face			Measurement pairs or trains	Adaptable; independent of scan rate
B. 360-Degree Coverage			Measurement pairs or trains	Adaptable; independent of scan rate

the measurement rate is very high.[1] The Tactical Air Forces Tactical Air Control System 2-D radars (AN/TPS-44) are presently all being replaced with the 3-D AN/TPS-43(E) radar, which also provides altitude.

Three dimensional radars--those which also measure elevation angle from which altitude can be derived--provide the minimum amount of information that satisfies the requirements of the netted automated tactical air surveillance system considered in this study, and all of the system configurations described in Sec. 1.3 and simulated as described in Sec. 3 are of this type.

Doppler. Radars are capable of measuring a fourth dimension, radial velocity, at the same time as making the position measurements by measuring the Doppler shift of the return radar signal. As discussed in Sec. 2.2, adding a fourth dimension can help track initiation for slow scan, track-while-scan radars with simple track-initiation algorithms, but is not particularly useful in computer-directed-track systems, which have more powerful means of improving track initiation.

This fourth dimension may help unravel multiple associations of returns with tracks. However, it does not appear that there is any point in providing Doppler for the tracking filter, since track accuracy is mostly determined by the maneuver capability of the aircraft; that is, how far the aircraft can maneuver between measurements.

Doppler information can, of course, be used to reject stationary or slow moving clutter. In fact, such a clutter-rejection or moving-target-indication (MTI) capability is essential to the successful operation of the radars in the system concept under consideration. It is

[1] One way of providing a high measurement rate in a 2-D surveillance radar, as suggested by Ed Muehe of MIT Lincoln Laboratory, is to use a circular linear array of elements. Such an antenna should be relatively inexpensive to build, and provides the flexibility in azimuth available with electronic steering.

assumed in all of the discussions and analyses that all or most of the ground and precipitation clutter is eliminated by Doppler processing such as that developed by Lincoln Laboratory for the Moving Target Detector.[1]

Amplitude. The other parameter a radar can directly measure is amplitude, which is proportional to (the square root of) radar cross section. A single measurement of amplitude tells essentially nothing about the aircraft because of the complex radar cross section patterns of aircraft. For example, a low measurement of amplitude could mean either a small aircraft or a measurement near a null in the pattern of a large aircraft. However, the average of several measurements made over a period of time could be used in selecting the radars which are in the best position to track a particular target as described in Sec. 1.3.2.6.

Several measurements either from a single radar spaced over time or from several radars with different aspect angles might provide some statistics that could be used to identify the type of aircraft. However, this possibility was not investigated during the study.

1.2.3 Comparison of Radar Modes

This appendix section first compares the relative power requirements of track-while-scan (TWS) and computer-directed-track (CDT) radars. This is followed by a comparison of the coverage capabilities of TWS radars and single-face, stationary phased arrays operating in the CDT mode.

1.2.3.1 Relative Power Requirements

The required radar output power for tactical air surveillance and control depends on the operating range, the target characteristics, and the environmental conditions as well as on the design of the radar and

---

[1] L. Cartledge and R.M. O'Donnell, Description and Performance of the Moving Target Detector, Lincoln Laboratory Report No. FAA-RD-76-190.

its mode of operation. While no attempt was made in this study to determine the required radar power under a variety of circumstances,[1] some calculations were made of the relative power requirements for computer-directed-track (CDT) operation as compared with those for track-while-scan (TWS) operation assuming that other conditions and parameter values are held constant.

If surveillance is performed at the same average rate, the power requirements for this function are independent of the mode of operation. With CDT operation, however, it may be possible to reduce the scan rate, at least in some portions of the surveillance volume; such a reduction would not be possible in TWS system without adversely affecting the tracking and especially the association performance. Since the average power required for surveillance is inversely proportional to the scan time, the power for this function could be reduced by the same factor as the increase in scan time. For tactical deployments and operating conditions, however, there will undoubtedly be regions in which long scan times will be unacceptable--for example, at low elevation angles in some azimuth sectors where terrain masking prevents detection at longer ranges. In these regions, surveillance coverage at appropriately high rates can be easily scheduled in a CDT system. Furthermore, this rate can be varied in response to changes in the tactical situation and the radar load. The average scan rate in a CDT system thus depends on the details of the deployment, the operational environment, and the target characteristics. However, it seems likely that this rate can be less than, or at most comparable to, reasonable TWS rates, so that the average power requirements for CDT surveillance will be less than or comparable to those for TWS surveillance, even after taking into account the off-boresight scan losses in phased-array radars.

---

[1] The design of the AN/TPS-43, which operates at a nominal average power of 4.7 kW, provides a good indication of the power levels required for this application.

In a CDT system, however, track initiation and tracking require additional power over and above that used for surveillance since they generally involve the use of special radar transmissions. An indication of the magnitude of the additional power requirements can be obtained by estimating the number of these special transmissions as compared with the number of surveillance transmissions in a specified time interval and accounting for any additional losses involved.

In a CDT system, track initiation is performed by making one or more position measurements at appropriate intervals after a surveillance detection which does not associate with an existing target track. Thus an extra set of track-initiation pulses must be transmitted for each false alarm during surveillance-pulse reception as well as for each new target that enters the surveillance coverage volume during the scan period (and for any surveillance returns from targets in track that fail to associate); in fact, for the parameter values of interest, the false alarms predominate. Assuming that the energy in a track-initiation transmission is equal to that in a surveillance transmission, the average power required for track initiation P_{TI} relative to that required for surveillance P_S is given by

$$\frac{P_{TI}}{P_S} = \frac{(N_{NT} + N_{FA})N_{IM}}{N_B} \tag{1.2-1}$$

where N_{NT} = number of new targets per scan

N_{FA} = number of false alarms per scan

N_{IM} = number of track-initiation measurements per detection

N_B = number of surveillance beam positions

Assume that new targets enter the radar coverage volume at a rate of one per second (a rate typical of those used in the sizing estimates presented in Secs. 1.4 and 1.5), or, for a nominal scan time of 10 seconds,

$N_{NT} = 10$; the number of false alarms per scan is specified in the Statement of Work to be from 50 to 200, with $N_{FA} = 100$ being a reasonable nominal value; and for a beamwidth of 1 deg the number of beam positions in the surveillance volume is of the order of $N_B = 10^4$. With these parameter values and $N_{IM} = 2$,

$$\frac{P_{TI}}{P_S} = 0.022 = 2.2\%$$

Thus the additional power required for track initiation in a CDT system is a small fraction of the power required for surveillance.

Additional power for track initiation would also be required in a TWS using a mechanically scanned phased-array antenna if a track-initiation measurement were made while the target is still within the off-boresight scan coverage of the rotating phased array. In fact, in this case there would be another term in the numerator of Eq. 1.2-1 representing the off-boresight scan loss. If the track initiation measurement were made at an off-boresight scan angle of 60°, the magnitude of this loss factor would be 2. With $N_{IM} = 1$ (one extra transmission per target for track initiation) and the same values as used above for the other parameters, the value of P_{TI}/P_S would again be 2.2%.

The relative magnitude of the additional power required for tracking in a CDT system can be estimated in a similar fashion. Neglecting any differences between the radar losses for surveillance and tracking, the average power required for target tracking P_T relative to that required for surveillance P_S is given by

$$\frac{P_T}{P_S} = K_M \frac{N_T/T_T}{N_B/T_S} \qquad (1.2\text{-}2)$$

where N_T = number of targets being tracked by the radar

N_B = number of surveillance beam positions

T_T = time between tracking measurements

T_S = scan time

K_M = factor by which the tracking power can be reduced by matching the transmitted energy to the target range

Again using quantities typical of those used later for sizing estimates, for 1,000 targets within the coverage of 70 radars with each target tracked by 3 radars (see Sec. 1.3), $N_T \simeq 43$ targets. For a beamwidth of 1°, $N_B \simeq 10^4$ beam positions as before. For the targets in track uniformly distributed throughout the flat-topped coverage volume of interest, the factor K_M is approximately 1/6 assuming the transmitted energy for each tracking measurement is perfectly matched to the target range so as to maintain a constant signal-to-noise ratio independent of range. With the nominal values of $T_T = 1$ second and $T_S = 10$ seconds ,

$$\frac{P_T}{P_S} = 0.0072 = 0.72\%$$

Even with imperfect matching of the transmitted energy and some variation in the assumed parameter values, it is clear that the additional radar power required for tracking in a CDT system is a small fraction of the power required for surveillance.

1.2.3.2 Planar-Array Coverage Capabilities

All of the antenna configurations described in Sec. 1.2.1 provide 360-degree coverage in azimuth with one exception--the stationary single-face planar array. This antenna is of interest because of its relatively low cost and enhanced mobility. As a first step in assessing the feasibility of using radars with such an antenna in a tactical air surveillance and control system, measures of attainable coverage were calculated and compared with those for antennas providing 360-degree coverage.

Since with a stationary single-face array the radar power available for surveillance is concentrated in a sector limited by the maximum scan angle, the surveillance range is greater than for 360-degree coverage. As compared to the range R_0 of a radar using a mechanically scanned reflector antenna with all of the same parameter values except scan times and scan sector, the surveillance range R of the stationary phased-array radar is given approximately by

$$\frac{R}{R_0} = \left(\frac{\pi}{\theta_s} \frac{\sin \theta_S}{\theta_s} \frac{T_S}{T_{SO}} \right)^{1/4} \tag{1.2-3}$$

where θ_S is the maximum off-boresight scan angle of the array (half the total angle coverage) and T_S and T_{SO} are the scan times of the phased-array and mechanically scanned antennas, respectively. The $\sin \theta_S/\theta_S$ term represents the average off-boresight scan loss (reduction of the effective receiving aperture); this term would not be present, or would have a different form, if the comparison were with a multi-face or other type of phased-array radar rather than with a radar using a mechanically scanned reflector. The relative-scan-time term is included because, as pointed out in Sec. 1.2.2, it may be desirable to use a lower scan rate (longer scan time) for surveillance, as is possible with CDT operation. The relative detection range as given by Eq. 1.2-3 is plotted in Fig. 1.2.4 as a function of the maximum scan angle for three values of relative scan time.

Of perhaps greater interest than the radar range is the area of the surveillance coverage. The area A of the sector coverage provided by the single-face array relative to the area A_0 of the circular coverage provided the mechanically scanned antenna is

$$\frac{A}{A_0} = \frac{\theta_S}{\pi} \left(\frac{R}{R_0} \right)^2 \tag{1.2-4}$$

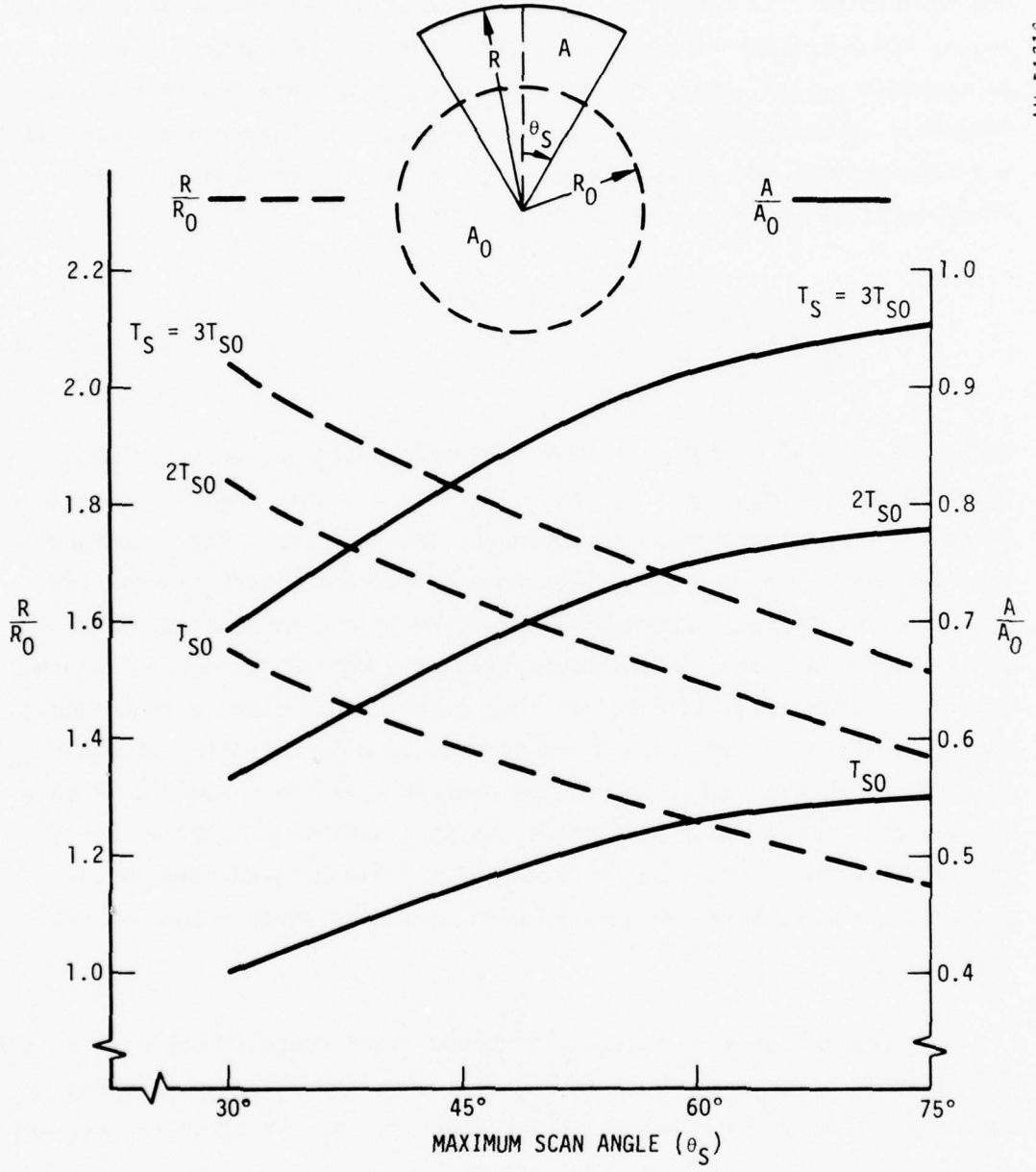


Figure 1.2.4. Relative Range and Coverage Area of a Single-Face Phased Array

with (R/R_0) as given by Eq. 1.3. This relative area is also plotted in Fig. 1.2.4 as a function of the maximum scan angle for three values of relative scan time. The use of a maximum scan angle of about 60° is reasonable, both because of the rather rapid beam degradation beyond that point and because the (A/A_0) curves start to level off in that region. At $\theta_S = 60°$, (A/A_0) ranges from 0.53 for $T_S = T_{SO}$ to 0.92 for $T_S = 3T_{SO}$. Thus if the longer average scan times are reasonable, as they appear to be, the surveillance area coverage with a stationary phased array is nearly as great as with a mechanically scanned antenna.

While the surveillance coverage that can be provided by the two types of antennas may be comparable in area, the shapes of the coverage patterns differ considerably. The single-face phased-array radars must be deployed so as to provide adequate coverage over the entire area of interest. Fortunately there will be considerable overlap of the coverage patterns of nearby radars, so the exact placement and orientation of each phased-array is not critical for high-altitude coverage. It may be more critical for low-altitude coverage, however, since some of the phased-array coverage is at longer ranges where the effects of terrain masking are greater. On the other hand, since the planar phased-array coverage is restricted to a limited azimuthal sector, it may be possible to orient the phased arrays so that this sector lies in a direction where there is little terrain masking--along a valley, for example. The effects of terrain masking on the surveillance coverage can be determined only by considering in detail particular terrain samples and radar deployments.

The low-altitude coverage for tracking is related to the range at which tracking is performed. In a system using mechanically scanned antennas and operating in the TWS mode, several radars must track each target cooperatively in order to obtain a data rate that is sufficiently high for reliable association, as is shown in Sec. 2.3. Assume that three TWS radars are required for adequate tracking and that the three radars closest to the target are used. The distance from the furthest of the

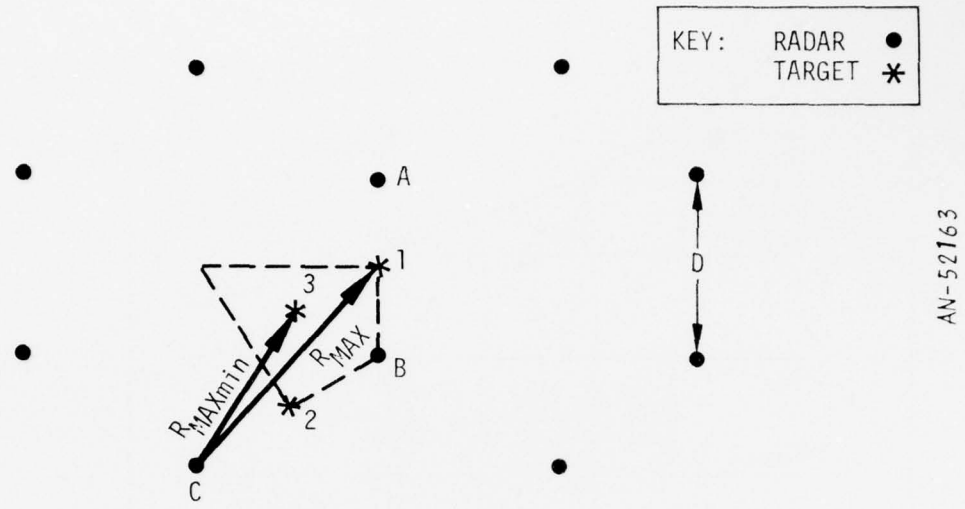
three radars to the target is then the most critical range in determining the low-altitude coverage. For the idealized hexagonal-grid deployment depicted in Fig. 1.2.5(a), Radars A, B, and C will track targets that are within the dashed lines. The maximum range R_{MAX} occurs when a target is at either Position 1 or 2; this range is $R_{MAX} = \sqrt{7}/2 = 1.32D$, where D is the spacing of the radars. The minimum maximum range R_{MAXmin} of the radar that is furthest from the target occurs when the target is at Position 3; in this case $R_{MAXmin} = \sqrt{3}/2 = 0.87D$.

For CDT operation with a single-face phased-array antenna, a single radar can track a target at the required rate. Assume that the closest radar whose coverage sector includes the target is used for this purpose. For the hexagonal-grid deployment with the coverage sectors oriented as shown in Fig. 1.2.5b, a target is at the maximum tracking range when it is just outside the coverage of Radars A and B at the position indicated, and therefore must be tracked by Radar C. This range is given by

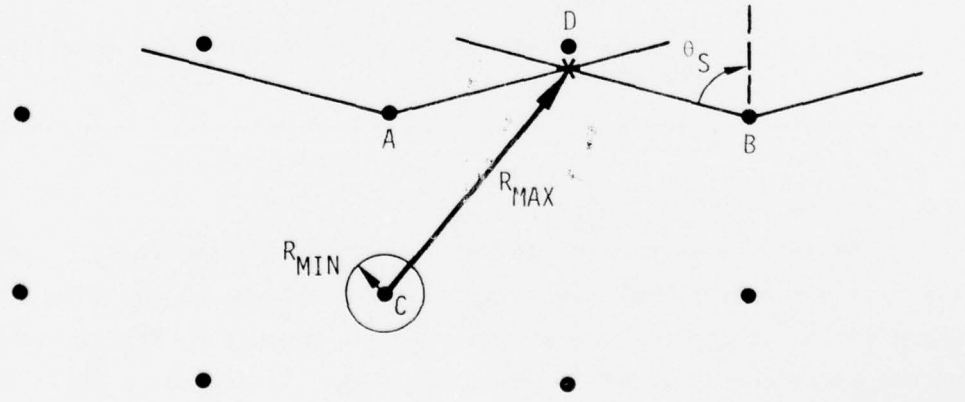
$$R_{MAX} = D\left[0.75 + (1 + 0.866\cot\theta_S)^2\right]^{1/2} \qquad (1.2\text{-}5)$$

for $\theta_S \geq 60°$. For $\theta_S < 60°$, the maximum tracking range is the lesser of the value of R_{MAX} from Eq. 1.2-5 and the minimum operating range of the radars since the target would fall in the sector covered by Radar D. The minimum SWT tracking range is simply the minimum radar range.

These maximum tracking ranges, expressed as multiples of the radar spacing D, are plotted in Fig. 1.2.6 as functions of the maximum scan angle. While the maximum tracking range for the single-face CDT radars (Case b) is somewhat greater than the maximum range to the furthest of three TWS radars (Case a) (1.73D versus 1.32D for a maximum scan angle of 60°), the minimum tracking range is considerably less (R_{MIN}, which is usually close to zero, versus 0.87D). Thus it is reasonable to expect that the low-altitude tracking coverage of the two types of radars will be comparable, although a detailed analysis of specific deployments is needed to determine this coverage.



a. TWS Operation With 360-Degree Coverage



b. CDT Operation With Single-Face Arrays

Figure 1.2.5. Tracking Geometries Idealized Deployment--Hexagonal Grid

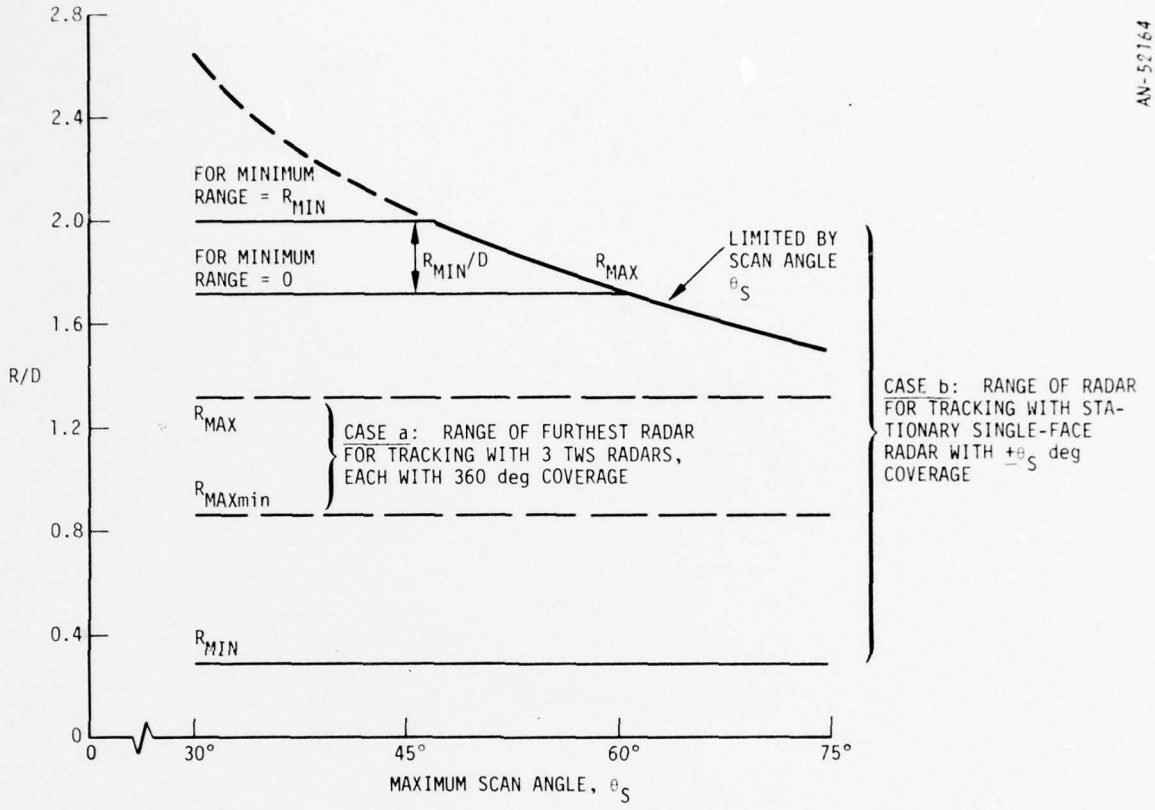


Figure 1.2.6. Maximum Low-Altitude Tracking Range for Idealized Deployment

1.3 SYSTEM CONFIGURATIONS

The data requirements and the performance characteristics of the tactical air surveillance and control system depend on the structure and organization of the communications and data processing systems as well as on the radar characteristics and deployment. To provide a basis for analyzing and comparing the requirements and feasibility of a variety of concepts, a number of representative system configurations were defined. These alternative configurations are described in this section. Following a general discussion of the track-file structure, the operation of

each of the systems selected for further analysis is described and a flow chart is presented to show the major logic decisions and computations which determine the communications and data processing requirements.

1.3.1 Track-File Structure and Location

The basic output of the system, the System Track File (STF), can be maintained at the Operations Facilities using various track-file structures at the radar sites. For example, a copy of the STF might be maintained at every radar site as well as at every Operations Facility. It is possible that not all of the System Tracks in the STF are maintained at every radar node; a Partial System Track File (PSTF) can exist at some radar nodes.

The System Track File may be used by each radar for tracking using its own measurements, but since the STF by definition is the file that is replicated at each node,[1] it must be updated at all nodes whenever any single radar updates its own copy for tracking purposes. This imposes a large load on the communications network. Therefore each radar may have its own track file, called the Local Track File (LTF), in which it maintains Local Tracks that are updated solely by using its own measurements. An LTF may be maintained at each radar in addition to or instead of an STF. Another possibility is that a Local Track is maintained by pooling measurements from several radars; thus this Local Track, which is distributed among the participating radars, is kept in a file called the Distributed Local Track File (DLTF).

The combinations of these files at the radar sites considered during this study and described in this section are shown in Fig. 1.3.1, which also shows that each Operations Facility has a copy of the System Track File.

---

[1] Radar sites and Operations Facilities will be collectively called the "nodes" of the network.

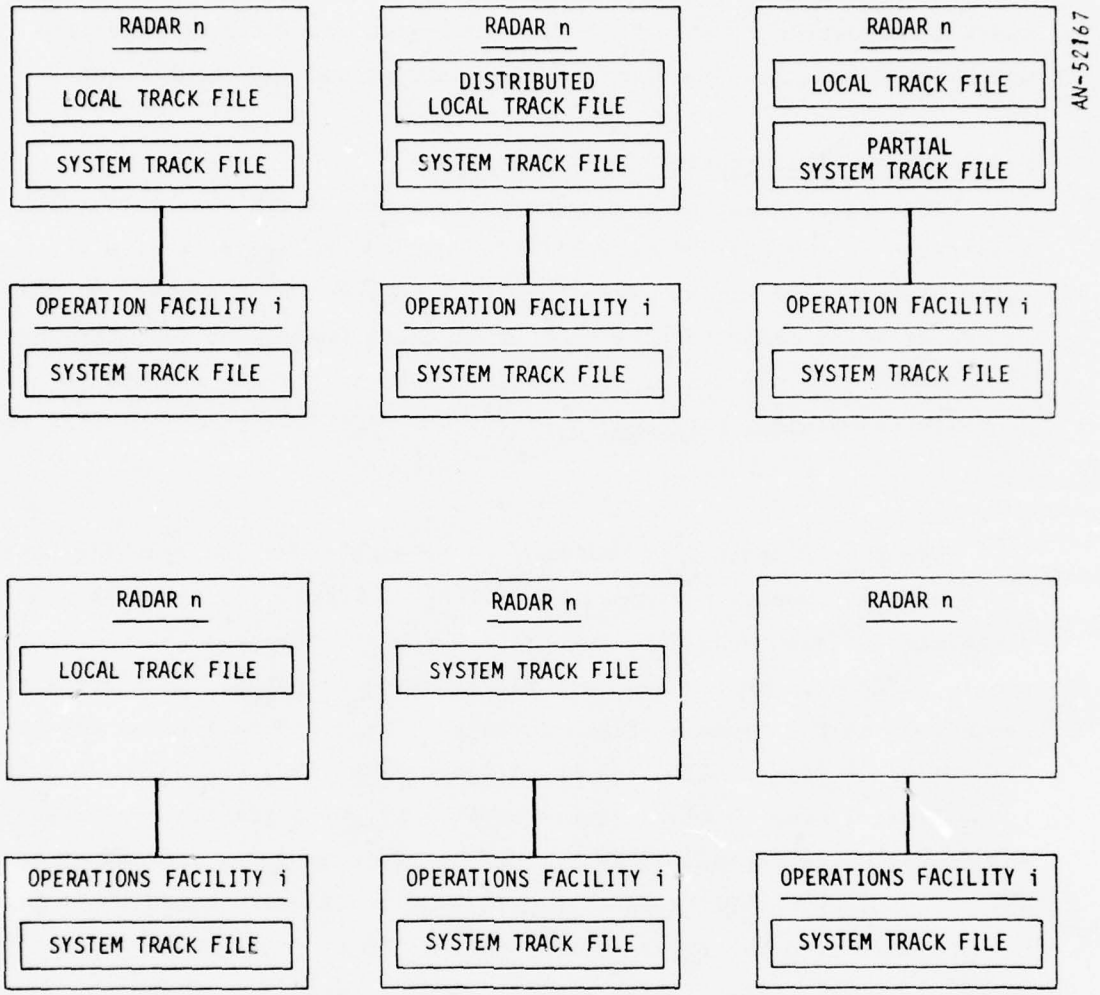


Figure 1.3.1. Six Variations of Track File Structure at the Radar Sites

---

In addition to the radar sites and the Operations Facilities, the system could include intermediate nodes where the processing to maintain the System Track File is performed, with STF updates being forwarded to the Operations Facilities. However, the operation of such systems, as well as their communications and data processing requirements, are

essentially the same as for comparable systems without the intermediate nodes, so the intermediate-node systems were not analyzed separately.

The system operation and data requirements depend not only on the track-file structure and location but also on the radar mode of operation. The major functional difference between the track-while-scan (TWS) and the computer-directed-track (CDT) modes of operation is that a CDT capability enables each individual radar to track independently at a rate high enough for reliable association under most conditions. This capability is exploited in the CDT systems by selecting a few radars out of the many radars (perhaps as many as 25) within tracking range of each target to track each target independently. One of these is designated as the primary tracker and is normally responsible for updating the System Track File for its primary targets and for supplying any additional information about these targets needed by other radars. At least one other radar, and possibly two or three, are also assigned to track the target as backups in case the primary tracker fails or is destroyed.

All combinations of the track file organizations of Fig. 1.3.1, radar modes of operation, and the use of selected or all radars to track targets lead to conceivable system configurations, but some are clearly more reasonable or interesting than others. The nine system configurations indicated in Table 1.3.1 were selected for further analysis. Except for Configuration 2, the five track-while-scan systems use all radars that can see a target to track that target; this has a considerable effect on the communications requirements for these configurations. Any of these TWS configurations (except Configuration 5) could also be operated with selected trackers, which would reduce their required communications. These TWS configurations were designed without the selected-tracker concept to provide cases using both concepts.

1.3.2 System Descriptions

The operation of each of the system configurations designated in Table 1.3.1 is described in the following subsections. The purposes of

TABLE 1.3.1

SYSTEM CONFIGURATIONS CONSIDERED

Configuration Number	Radar Mode*	Track Files at Radar†	Selected Trackers?
1	TWS	LTF and STF	No
2	TWS	Distributed LTF and STF	Yes
3	TWS	LTF	No
4	TWS	STF	No
5	TWS	None	No
6	CDT	LTF and STF	Yes
7	CDT	LTF and Partial STF	Yes
8	CDT	LTF	Yes
9	CDT	STF	Yes

*TWS--Track-While-Scan; CDT--Computer-Directed-Track.

†LTF--Local Track File; STF--System Track File.

these high-level operational descriptions are (1) to demonstrate that the concepts are logically feasible by giving an example of a possible implementation, and (2) to provide a basis for determining the communications and data processing requirements associated with each configuration by identifying the messages that must be transmitted and the computations that must be performed.

1.3.2.1 System Configuration 1

Configuration 1 was also historically the first to be considered. Considerable time was spent on it and two versions were modeled in the TACRAN simulation. Configuration 1 has a number of conceptual flaws that were uncovered and corrected in Configuration 2.

Configuration 1 is the most basic and simplest configuration considered. The radars operate in a track-while-scan mode with a relatively

long scan period of from 4 to 12 seconds. All radars that can see a target track it, maintaining a Local Track in the Local Track File (LTF). Each radar also has a copy of the System Track File (STF).

A flow chart or top-level logic diagram for a system of this configuration is shown in Fig. 1.3.2. Each radar return (measurement) is

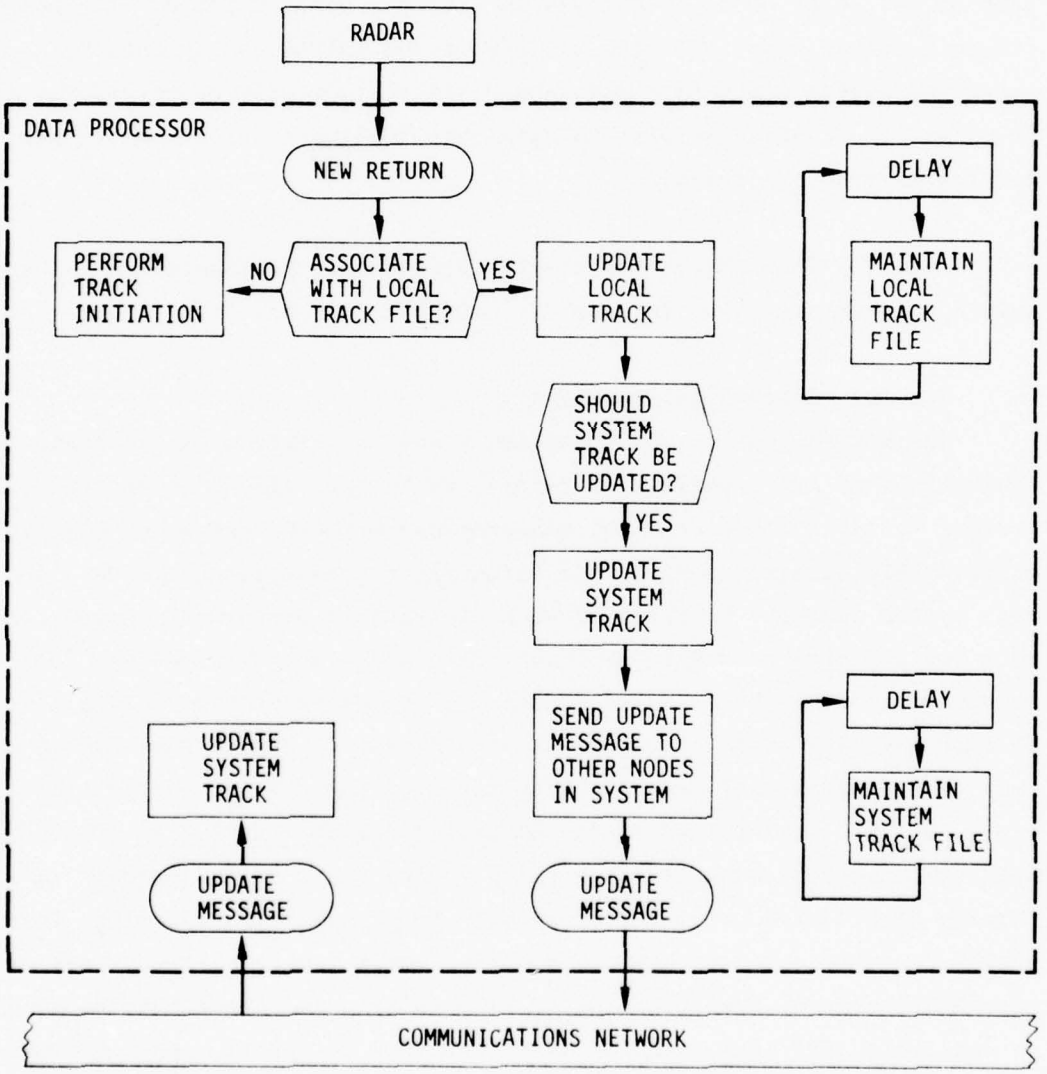


Figure 1.3.2. Flow Chart for TWS Operation With a LTF and STF at Each Radar and All Radars Tracking (Configuration 1)

first associated with the LTF (i.e., a decision is made as to which Local Track in the LTF, if any, belongs to this return). When an association is made, the associated Local Track is updated using the return. The System Track on the same target is updated when (1) a maneuver is detected by observing that the distance of the measured target position from the target track in the LTF exceeds a threshold, or (2) a specified time interval has elapsed. Whenever a System Track is updated, a message is sent to all other nodes in the system with the updated track. If the return does not associate with the LTF, track initiation is entered. Local and System Track File maintenance is periodically performed (as it is in all the configurations) to eliminate duplicate tracks and to purge old tracks from the files.

Detailed flow charts for the two versions of Configuration 1 that were simulated are given in Secs. 3.3 and 3.4.

1.3.2.2 System Configuration 2

The design goal of Configuration 2 was to overcome the inherent limitations of track-while-scan radars, which typically scan at rates too low for reliable association of measurements with the tracks of highly maneuverable aircraft while still using the track-while-scan mode. In this system concept, tracking of each aircraft is performed cooperatively by a few selected radars to minimize communication requirements. The measurements from the selected radars are pooled to provide a single track on each aircraft at an effective data rate that is higher than the scan rate of the individual radars. Each tracking radar has a copy of this track, which is maintained in its Distributed Local Track File (DLTF). The determination and control of which radars track which aircraft is totally distributed among the nodes with no centralized controller whatsoever, an attribute which increases the system survivability. Each radar also has a copy of the System Track File (STF). The algorithms required to implement this configuration were developed in detail, and Configuration 2 was modeled in the TACRAN3 simulation described in Sec. 3.5.

An aspect of this concept which somewhat complicates the association process is the fact that there are a number of radars which are in range of each target but are not tracking the target. Detections by these radars must therefore be associated with the System Track File to determine whether the target is being tracked or is in fact a new detection.

A flow chart for this system concept is shown in Fig. 1.3.3. The sequence of association decisions that are made is described by the blocks

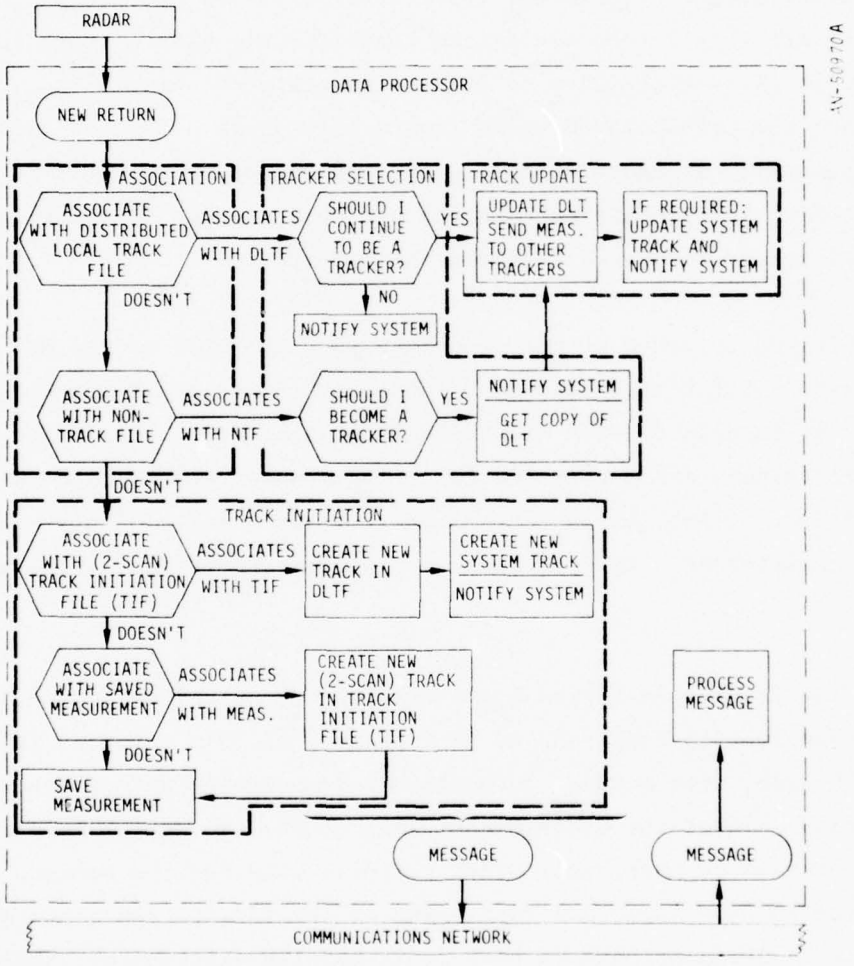


Figure 1.3.3. Flow Chart for TWS Operation With DLTF and STF at Each Radar and Selected Radars Tracking (Configuration 2)

along the left side of the flow chart. If a new radar return (measurement) associates with the Distributed Local Track File (DLTF), the return is processed through the Tracker Selection and Track Update Logics (described below). If it does not associate with the DLTF, then this radar is not presently a tracker of this target. Next, if the return associates with the Non-Track File (the NTF is the portion of the System Track File which contains all of the System Tracks except for those on targets being tracked by the local radar), then the Tracker Selection Logic is also entered. Because the System Tracks are not maintained as accurately as the Distributed Local Tracks, in the process of determining whether or not a return associates unambiguously with a track in the NTF (STF), it may be necessary to obtain the distance between the measurement and the extrapolated track from a more up-to-date, and presumably more accurate, estimate of the track as provided by the DLTF at another radar which is maintaining this track. This process is accomplished by communications (messages) between the two radars.

If the return does not associate with the DLTF or the NTF (STF), the track initiation procedure is entered. First it is determined whether or not it associates with the track initiation file, where data from the two preceding scans are stored for a three-measurement track-initiation algorithm. If not, it is determined if it associates with a single previous measurement. In all cases, appropriate track-initiation steps are taken.

The Tracker Selection Logic involves computations and decisions to determine if this radar should be a tracker of this target. It is entered every time a radar associates a measurement with a track. A major criterion is that the measurement times of the set of trackers be as evenly spaced as possible. Other criteria consider the present number of trackers and the range and range rate of the target. An important aspect of this decision process is that it is totally distributed; there is no centralized controller which selects the tracking radars. The process generates several types of communication messages between nodes.

1.3.2.3 System Configuration 3

This system configuration is similar to the first system described above except that there is no System Track File at the radars, only a Local Track File. All of the track-while-scan radars track all targets that are within range. Since the STF exists only at the Operations Facilities, all STF update messages are sent only to these nodes.

A flow chart for this system configuration appears in Fig. 1.3.4. It consists of two major blocks--one for the data processing performed at the radars and one for the processing at the Operations Facilities--which are connected through the communications network. The logic steps are similar to those for the Configuration 1 described above except for the track labeling and association of the LTF with the STF. When a STF update message is received at the Operations Facilities, its label is checked to see if it corresponds to that of a track in the STF. If it does, that track is updated; if it does not, a determination is made as to whether or not the track associates with one in the STF. If it does, that track is updated and the System Track label is sent to the radar for assignment to the Local Track; if it does not, the track is added to the STF and its Local Track label is retained. For purposes of assigning a System Track label to a track which associates with a System Track, one of the Operations Facilities is designated as the primary processor for that track to avoid the possibility of assigning different labels to the same Local Track.

As indicated in the flow chart, the only types of communications messages in this system are the STF update and STF label messages.

1.3.2.4 System Configuration 4

In this track-while-scan configuration, each radar has a copy of the System Track File, but no Local Track File. With no LTF at the radars, the returns must be associated with the STF. Each time a radar updates its copy of the STF, it must send the update to all other nodes (since by

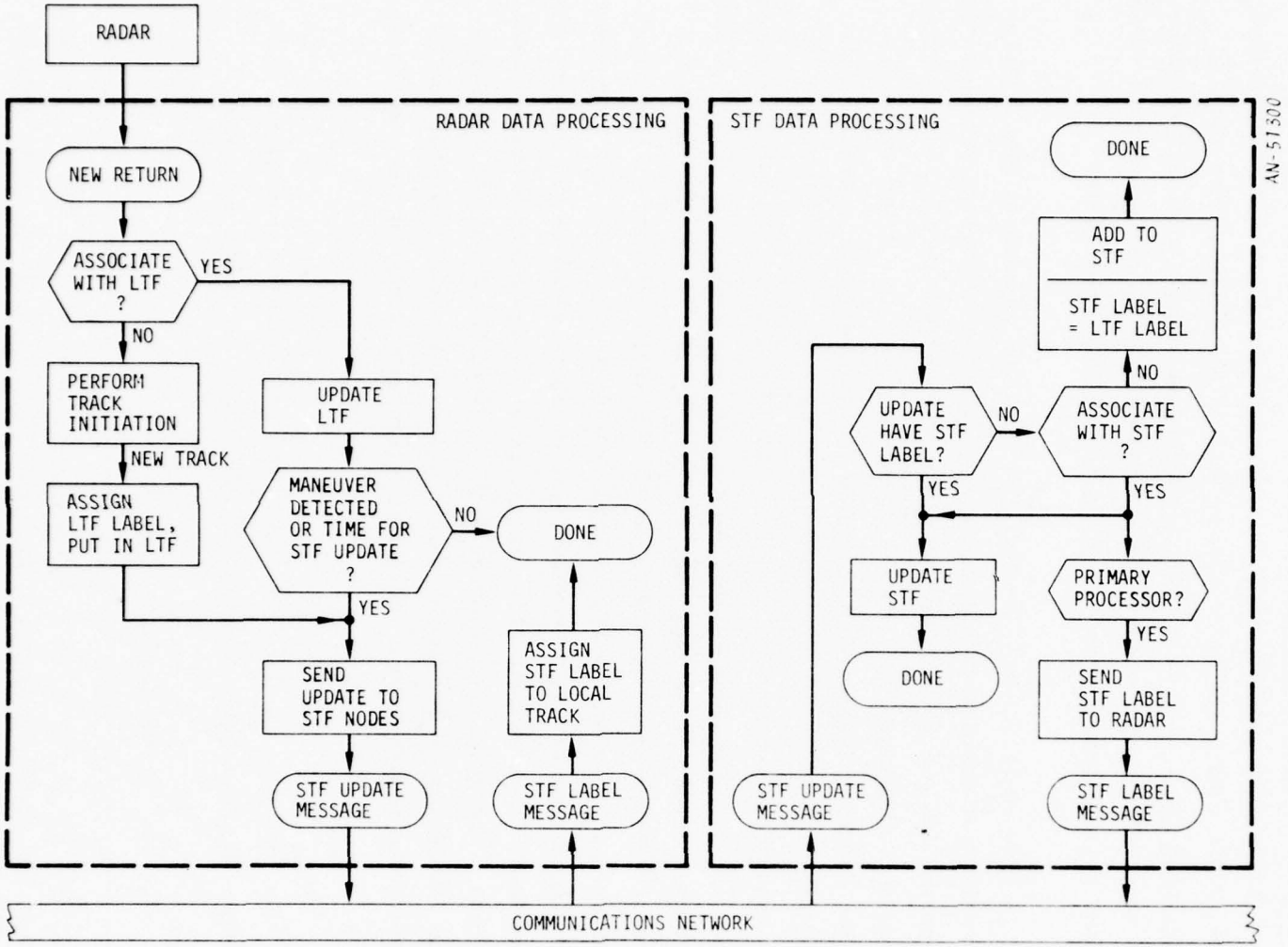


Figure 1.3.4. Flow Chart for TWS Operation With LTF Only at Each Radar and all Radars Tracking (Configuration 3)

definition the STF is the system file that is identically maintained at every node). It is also assumed in this configuration that all radars that can see a target track it. The flow chart for this relatively simple system is given in Fig. 1.3.5. As it indicates, the only type of message that must be communicated is the STF update message.

1.3.2.5 System Configuration 5

This interesting track-while-scan configuration has no track files at the radars. The only track files in the system are copies of the System Track File at each of the Operations Facilities. With no track files at the radars, the parameters describing each detection (i.e., the measurement itself) must be sent to the Operations Facilities for further processing. There the measurements are associated with the STF and, for targets that are in track, the STF is updated with each measurement. As indicated in the flow chart of Fig. 1.3.6 which shows the steps in this process, the only type of message transmitted from the radars to the Operations Facilities is the measurement message.

1.3.2.6 System Configuration 6

Configuration 6 is the first of the systems using the computer-directed-track mode. A major difference between TWS and CDT operation is that CDT transmissions are scheduled for specific purposes such as surveillance, track initiation, and tracking, and the returns are processed according to the function being performed. In Configuration 6, which includes both a Local Track File and a System Track File at each radar, this operational concept is reflected in the flow chart of Fig. 1.3.7, which is one possible implementation of a CDT system. At the left side of the flow chart the three types of returns of primary interest are routed to different portions of the logic diagram by the Returns Distribution logic. At the right side, after the return-signal data have been appropriately processed, requests for subsequent transmissions are sent to the Radar Scheduler. Thus the operation of the radar and data processor constitutes a closed-loop process.

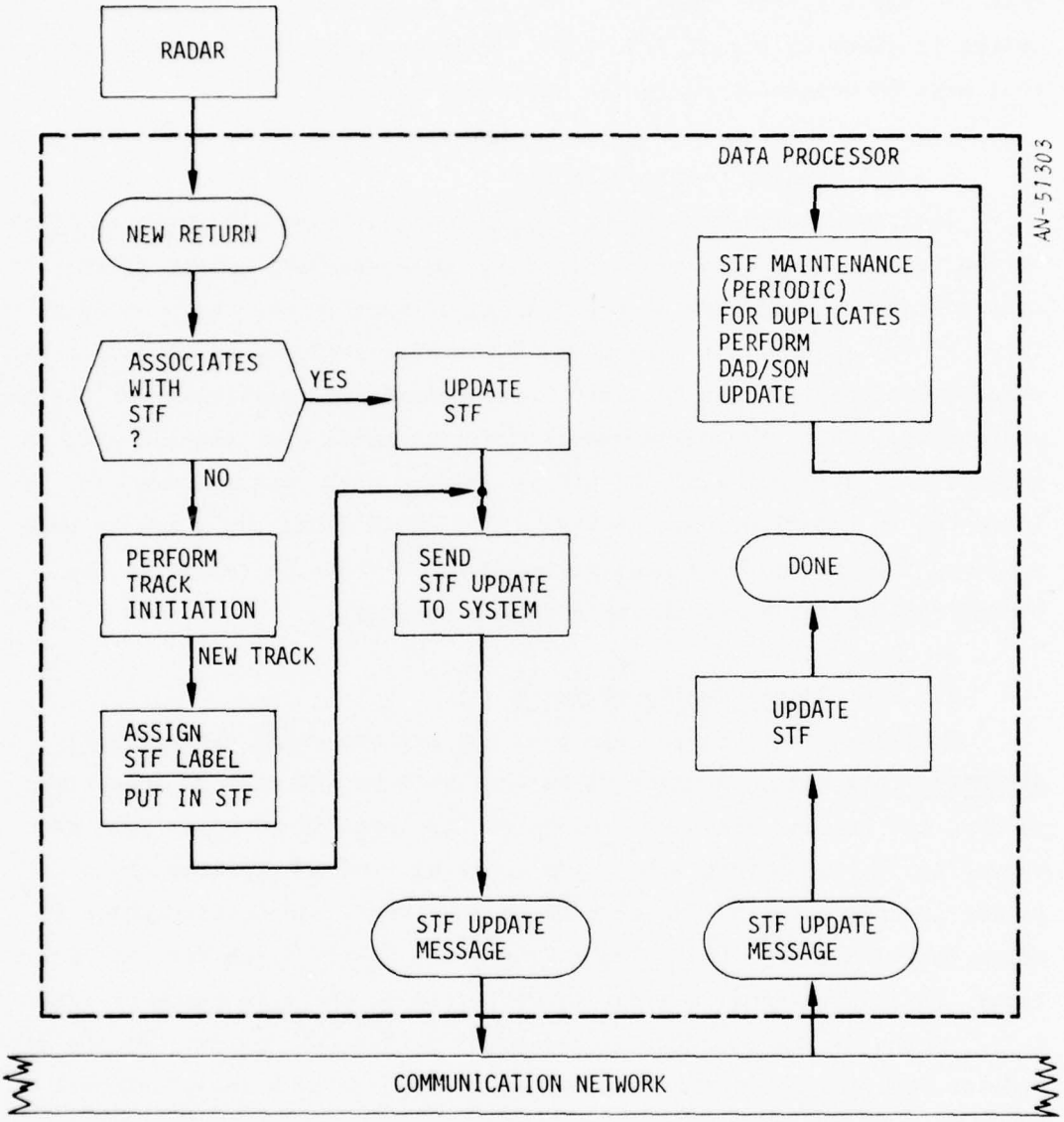


Figure 1.3.5. Flow Chart for TWS Operation With STF Only at Each Radar and All Radars Tracking (Configuration 4)

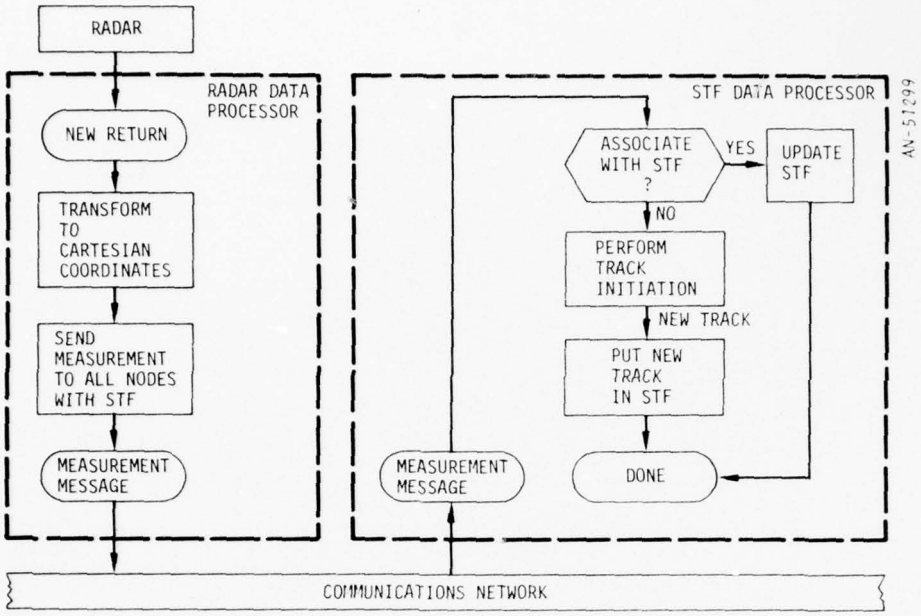


Figure 1.3.6. Flow Chart for TWS Operation With No Track Files at the Radars (Configuration 5)

Since this is the first of the CDT configurations, and because this type of system is not covered in the simulation section, it will be described here in more detail than the other configurations. Surveillance returns are associated first with the Local Track File to see if the target is being tracked by this radar. If not, these returns are then associated with the System Track File to see if the target is being tracked by any radar. If it is being tracked by other radars, logic to determine whether or not this radar should become a tracker is entered. This logic is presented in greater detail below, as is the logic for determining whether or not the surveillance return associates with the System Track File. If the target is not in track by any radar, a Track Initiation File (TIF) is created and the track-initiation process is started.

Track-initiation returns are checked to see if they associate with the appropriate entry in the Track Initiation File (the only possible

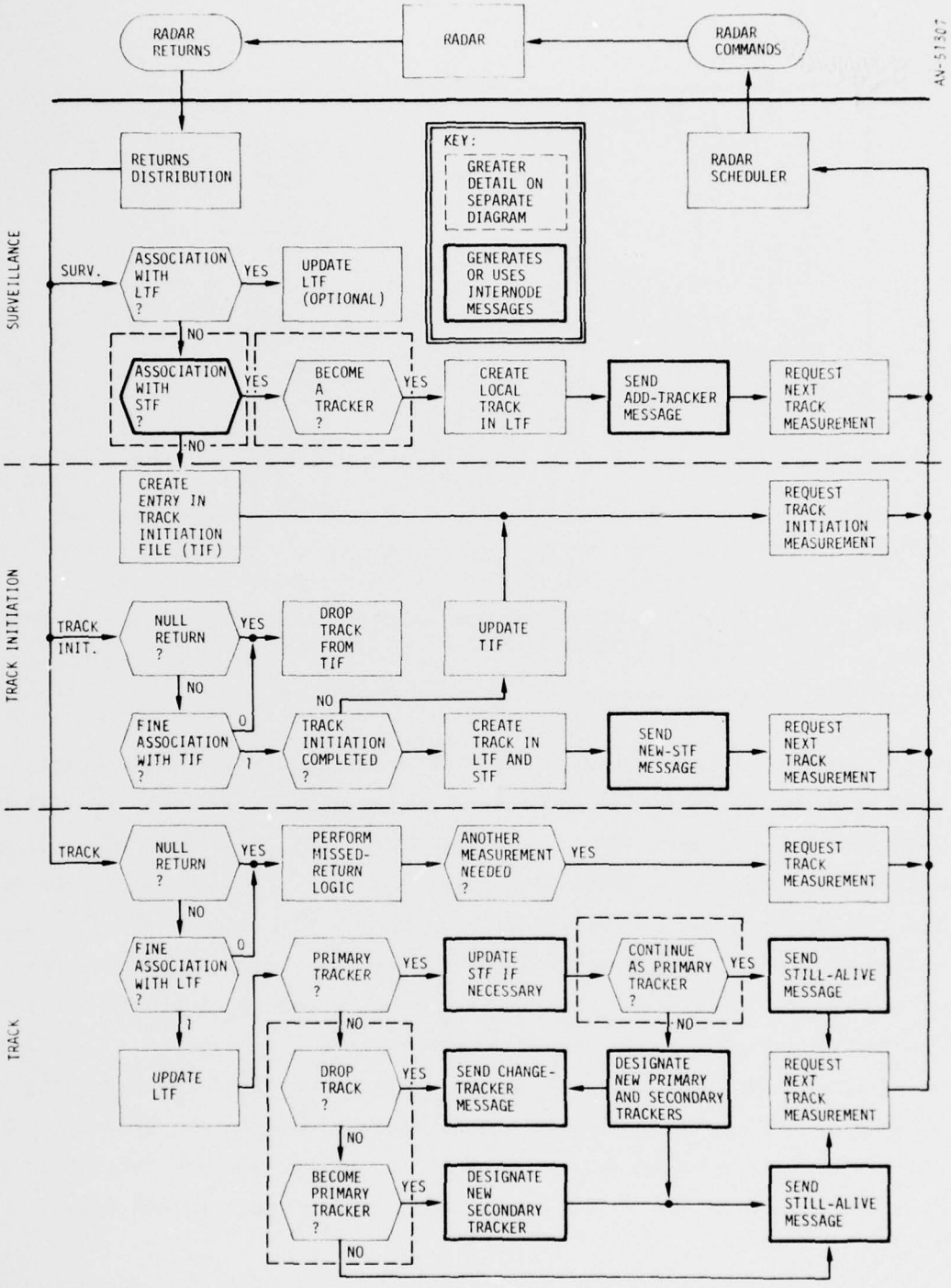


Figure 1.3.7. Flow Chart for CDT Operation With an LTF and STF at Each Radar and Selected Radars Tracking (Configuration 6)

outcomes of this fine association process are a positive association with this entry or no association). If they do associate, a track is created for the new target or the track-initiation process is continued as appropriate.

The returns from track transmissions are already tagged with the label of the track. (Information about each transmission is provided by the Radar Scheduler to the Returns Distribution logic to simplify the association process, but the flow of such association data is not shown in Fig. 1.3.7.) However, a fine association with particular entries in the Local Track File is made to check that the association is correct. If the outcome is positive as expected, the Local Track is updated. If this radar is the primary tracker for this target, the System Track is updated if the deviation of the Local Track from the System Track exceeds a specified threshold. Logical operations are performed to determine whether this radar should drop track, become a primary tracker, or continue as the primary tracker, as shown in greater detail in subsequent logic diagrams. These operations generate several types of messages as indicated.

As indicated in the key of Fig. 1.3.7, several of the blocks are shown in greater detail in separate diagrams. The logic for the Association With STF block is shown in Fig. 1.3.8. In cases where the tracks are so close together that unambiguous association with one of the System Tracks is not possible, it is necessary to obtain more accurate values of the distances between the measurements and the tracks within the association volume from the radars primarily responsible for maintaining these tracks in their Local Track Files. This requires communications messages between the nodes.

An example of a selection logic for adding new trackers based on one possible set of criteria is presented in Fig. 1.3.9; other criteria could be used also. The check on the average power being used for tracking is a simple means of distributing the tracking load among the radars;

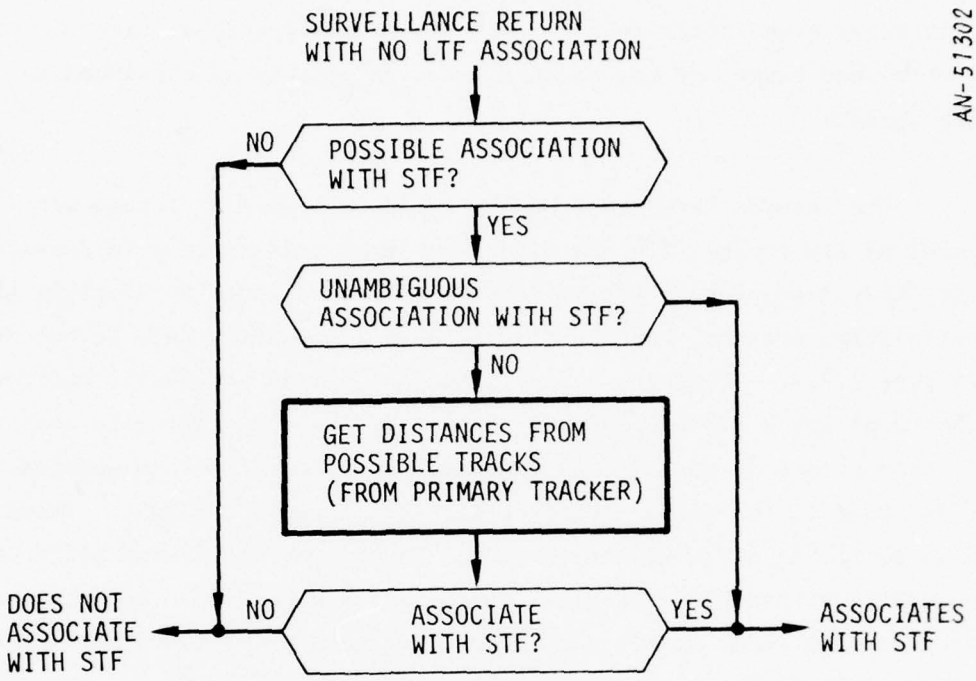


Figure 1.3.8. Association-With-STF Logic (STF at Each Radar)

more complex load-balancing procedures would probably be necessary in a practical system. If the desired number of radars N_R is already tracking the target, the decision as to whether this radar should replace one of the trackers, as shown in Fig. 1.3.9, is based on the relative signal-to-noise ratios and geometries, where the latter criterion involves such considerations as proximity to the radar coverage limits or to regions of high clutter or jamming.

Similar considerations are involved in determining whether a radar that is already tracking the target should continue as a primary tracker, become a primary tracker, continue as a secondary tracker, become a secondary tracker, or continue as the third (or other tracker). Again various selection criteria might be used; examples of tracker-designation logic based on the average signal-to-noise ratio (averaged over the last

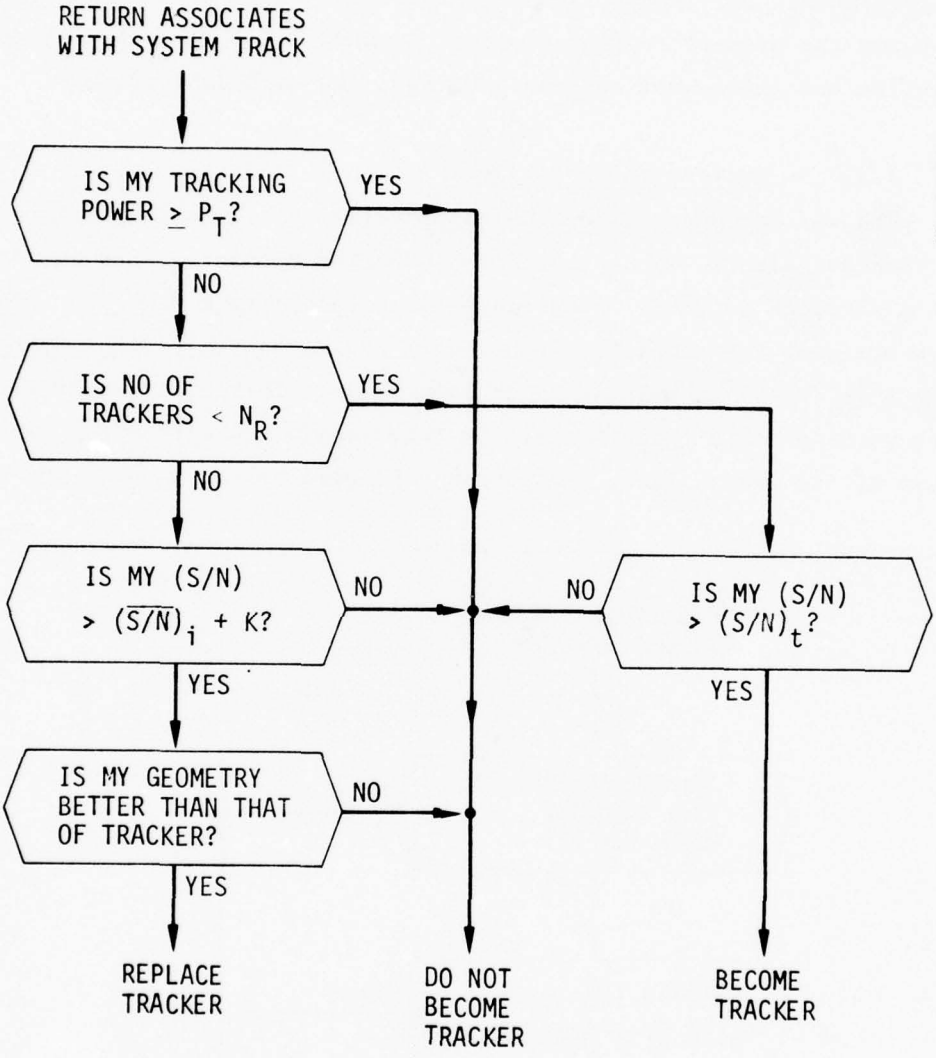


Figure 1.3.9. Add-Tracker Logic

few tracking measurements) are shown in Figs. 1.3.10 and 1.3.11. Any such selection and designation logic must ensure that there is always at least one (preferably only one) primary tracker, one secondary tracker ready to become the primary tracker whenever needed, and a clear procedure for effecting the changeover and designating a new secondary tracker.

1.3.2.7 System Configuration 7

The next computer-directed-track system configuration designated for consideration in Table 1.3.1 is a modified version of the one described in the previous section. Each radar has a Local Track File, but instead of maintaining the complete STF at every radar, only a Partial System Track File (PSTF) is maintained at each radar. This PSTF contains the track data for only those targets in track which are within the coverage volume of the radar, i.e., can possibly be seen by the radar. The reason

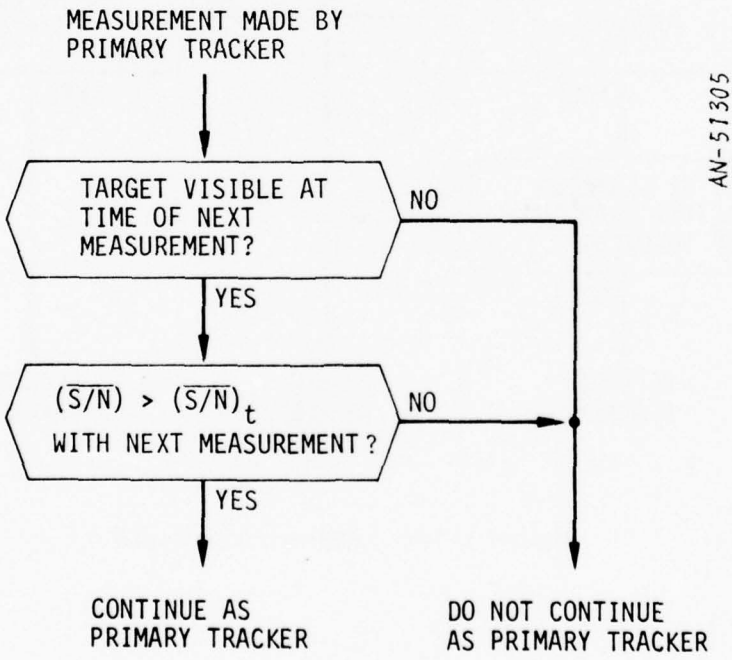


Figure 1.3.10. Primary-Tracker Logic

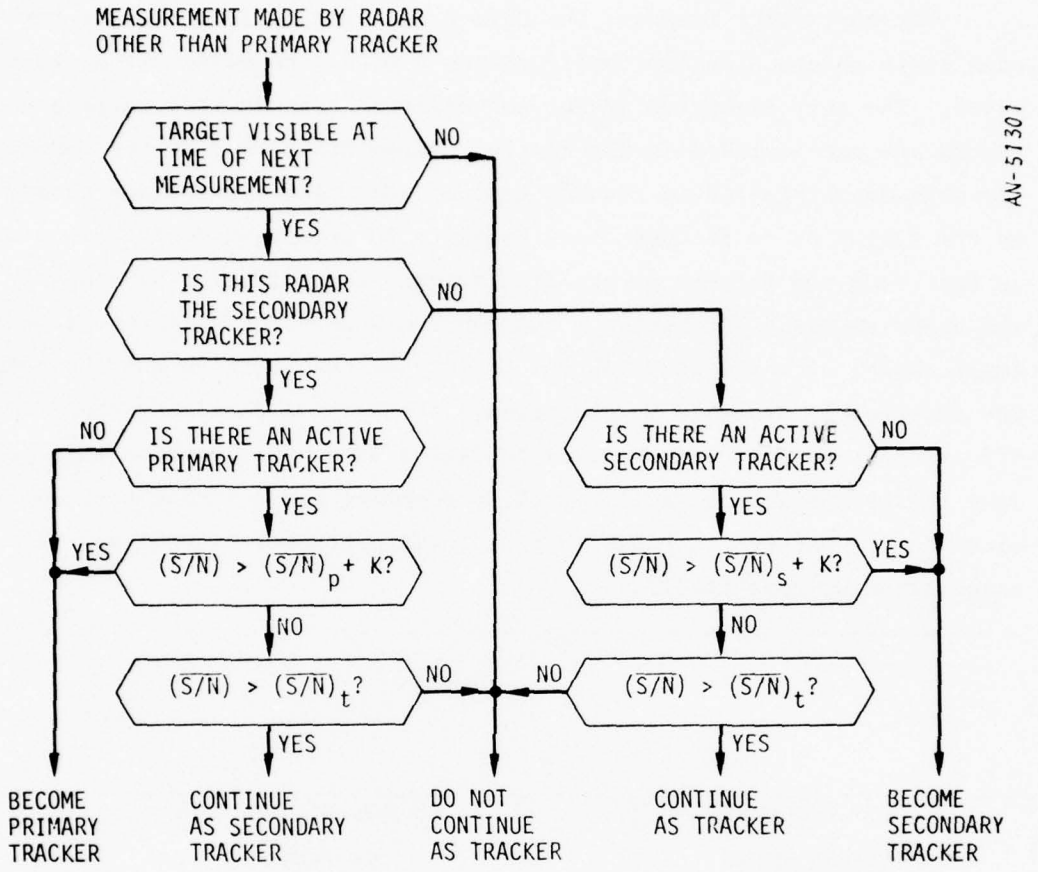


Figure 1.3.11. Other-Tracker Logic

for considering the use of such a Partial System Track File is to reduce the communications requirements by limiting the dissemination of the STF updates to those radars which need these data for association with the surveillance returns. Once a target is in track and its position is known, the radars which are likely to be able to see the target can be determined by the primary tracker, and the STF updates can then be transmitted by the primary tracker to only those radars. Of course, a complete STF must be maintained at the Operations Facilities for use in the planning and control operations.

For most radar returns, the data processing follows exactly the same logic sequence as for Configuration 6 with a complete STF at each radar. The only exception is for surveillance returns from targets whose tracks are not included in the Partial System Track File at the radar. The data describing these returns must be sent to another radar closer to the target or to an Operations Facility to determine whether they are in fact from new targets or are from targets being tracked by some of the other radars. The latter situation could occur only in the presumably small number of cases in which the ability of the radar to see the target was incorrectly assessed by the primary tracker. A flow chart for the STF association logic is provided in Fig. 1.3.12. For returns that are sent to the Operations Facility, the processing there includes the determination of whether the radar should become a tracker as well as the association process itself.

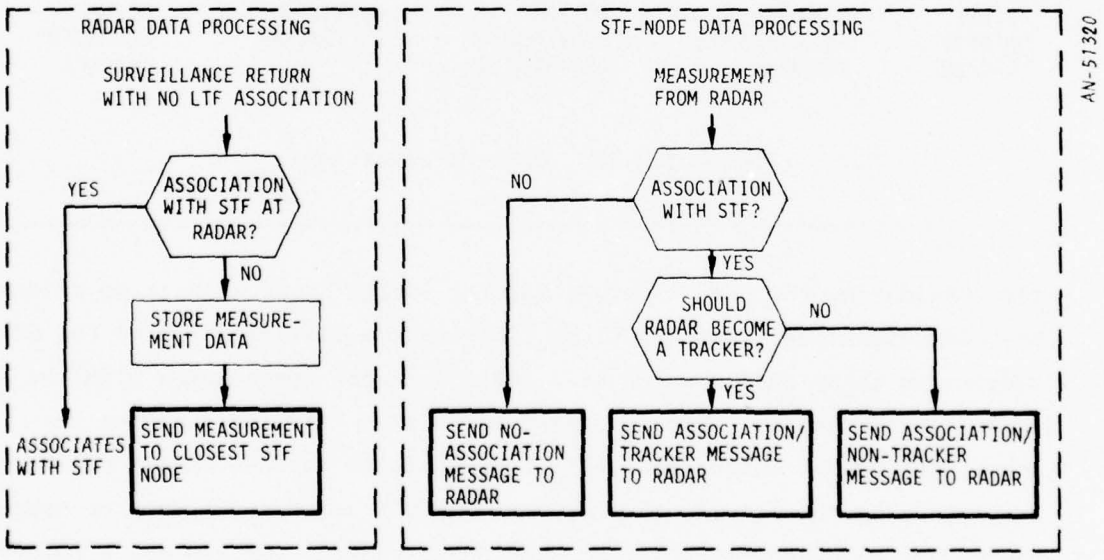


Figure 1.3.12. Association-With-STF Logic (Partial STF at Each Radar)

1.3.2.8 System Configuration 8

In this computer-directed-track configuration, each radar has only a Local Track File. With no System Track File at the radars, all of the surveillance returns which do not associate with the LTF must be sent to an Operations Facility for association with the STF. With this modification--the elimination of the check for association with the STF at the radars--the logic for this system configuration is the same as that for Configuration 6. As in the other CDT systems, several radars are selected to track each target and one of these is designated the primary tracker and another the secondary tracker. The LTF is updated with each tracking measurement, and the STF is updated when a maneuver is detected by the primary tracker based on the deviation of a measurement from the predicted target track (or when a specified time has elapsed). The STF updates are of course sent only to the Operations Facilities, not to any of the radars.

1.3.2.9 System Configuration 9

The last computer-directed-track configuration has only the System Track File at each radar. With no Local Track Files at the radars, both the surveillance and tracking returns must be associated with the System Track File at each radar. The flow chart of Fig. 1.3.13 is a modified version of Fig. 1.3.7 which reflects the absence of the Local Track Files. Other than that, the data processing logic for the two systems is identical except that the STF must be updated with each tracking measurement made by the primary tracker since no other track file is maintained. As with the other CDT systems considered, each target is tracked by a few selected radars, including one designated as the primary tracker and another as the secondary tracker.

1.4 COMMUNICATIONS

In a tactical air surveillance and control system whose operation depends on the interchange of information among a number of radar/data processor nodes and with the Operations Facilities, the communications system is a vital element. To provide the redundant non-hierarchical

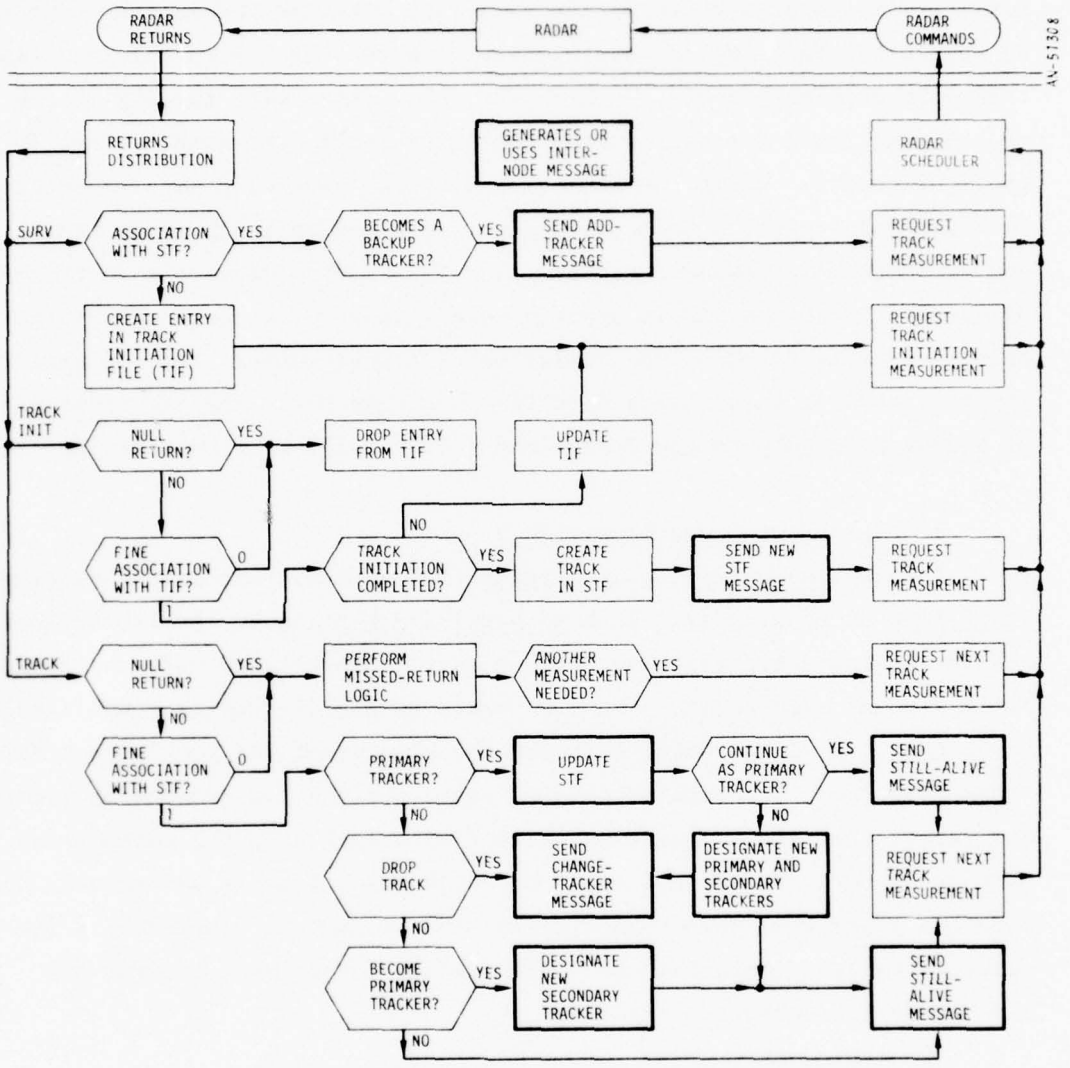


Figure 1.3.13. Flow Chart for CDT Operation With STF Only at Each Radar and Selected Radars Tracking (Configuration 9)

netting that is being considered in this study, the communications system links each node directly with several of its nearest neighbors. Such netting provides the redundancy and flexibility needed to support gradual or graceful degradation of system performance with the loss of radar/data processor nodes and communication links. Although the radars are deployed so that the links between neighboring nodes can generally be over line-of-sight paths for which a wide-bandwidth capability can be provided, it is desirable to minimize the bandwidth in order to keep the link cost and complexity down and/or to provide ECM resilience through the use of spread-spectrum techniques. To determine the effect of the radar mode of operation and the track-file structure on the bandwidth, the communications data-rate requirements of the nine system configurations described in the preceding section were analyzed. This analysis and its results, which include both parametric expressions for the communications data rates and numerical examples based on a particular set of parameter values, are presented in this section following a discussion of the factors which affect the bandwidth requirements.

It is important to note that the communication data rates presented in this section were calculated primarily for comparative purposes and to gain insight into the communication requirements. They are not to be considered as definitive statements of bandwidth requirements. In particular, the data rates presented are averages over the temporal variations and the spatial distributions of the targets although the worst-case netting geometry is considered as described in Sec. 1.4.4. Also, transmissions that involve only protocols have been ignored. Nevertheless, the examples presented should provide a useful comparison of configurations and an indication of the quantitative bandwidth requirements.

1.4.1 Message Types

There are two types of messages used in the system configurations described in Sec. 1.3: system messages and directed messages. A system message is one which is sent to all other nodes in the system. It contains information required by all other nodes, such as a message to update

a particular System Track. A directed message is one which is specifically addressed to one or a few nodes. The algorithm used for routing system messages affects the average data rates. The one used in the communication-data-rate calculations is due to Otto Wech of ESD and is sometimes called the flooding algorithm. With this algorithm, the message traverses every link exactly once (except in instances where the same message is transmitted from opposite ends of the link at nearly the same time). (A discussion of routing algorithms for system messages is given beginning on p. 253 under the heading of Communication Simulation.)

Directed messages, which are sent to one or more specified nodes, must include the addresses of their destinations in their header. In the cases of interest here, directed messages are normally sent to one or two other nodes. The variation of header length with number of receivers in this range has only a minor effect on the communications data-rate requirements.

1.4.2 Message Lengths

The required communication bandwidth is directly related to the number of bits contained in the message to be transmitted. Each message consists of a header, text, and a cycle redundancy check (CRC).

The header contains data required by the communications system in transmitting the message. Table 1.4.1 lists the header components and the CRC, which is used to ensure accurate transmission. Although the CRC follows the message text in the transmission sequence, it is included as part of the header in accounting for the contributions of the various components to the total message length. The bit lengths used in the data rate calculations are also listed in Table 1.4.1. The names of the components are self-explanatory, except perhaps for the number of parcels and the parcel number. If some messages turn out to be exceptionally long, it might be advantageous to divide them into two or more parcels for transmission, and it would be necessary to identify the parcels; otherwise

TABLE 1.4.1

MESSAGE HEADER COMPONENTS AND BIT LENGTHS

Component	Number of Bits
Sender Label	8
Number of Receivers (N_R)	8
Receiver Addresses	$8N_R$
Message Type	4
Priority	4
Number of Parcels and Parcel Number	4
Cyclic Redundancy Check * (CRC)	16
Total Number of Bits	$44 + 8N_R$

*Follows message text.

there would be no need to include these components in the header. As shown on the bottom line, the total number of header bits is $44 + 8N_R$, where for directed messages N_R is the number of nodes to which the message is addressed and for system messages $N_R = 0$.

The contents of the message text depends upon the type of message. Elements of information which must be conveyed between nodes are listed in Table 1.4.2 along with the number of bits required for each element and the basis for determining this number. (The "special components" are used in only certain configurations; the "general components" are used in all.) The various messages of interest consist of one or more of these elements in appropriate combinations. For example, a System Track File update message in a track-while-scan system with a Local Track File and System Track File at each node (Configuration 1) consists of the following components with the specified bit lengths:

TABLE 1.4.2

MESSAGE TEXT COMPONENTS AND BIT LENGTHS

Component	Information Content	Number of Bits
General Components:		
Target Label	2^8 nodes and 2^8 targets/node	16
Time	8 min extent with 1/64 sec accuracy	15
X,Y Coordinates	1,000 km extent with 10 m accuracy	2 @ 17
Z Coordinate	30.5 km extent with 10 m accuracy	12
$\dot{X},\dot{Y},\dot{Z}$	1,300 m/s extent with 4 m/s accuracy + sign bit	3 @ 10
$\ddot{X},\ddot{Y},\ddot{Z}$	100 m/s^2 extent with 1 m/s^2 accuracy + sign bit	3 @ 8
Special Components:		
Number of Distances Requested	Maximum of 8 per message	3
Distance	2.5 km extent with 10 m accuracy	8
Number of Trackers	Maximum of 8	3
Signal-to-Noise Ratio	1 part in 250	8
Node Label	2^8 nodes	8
Help Request	Binary	1

Target Label		16 bits
Time		15
Target State:		
X,Y,Z	45	
$\dot{X},\dot{Y},\dot{Z}$	30	100
$\ddot{X},\ddot{Y},\ddot{Z}$	24	
Total		131 bits

The numbers of bits in each of the types of messages needed in the system configurations under consideration are given in Sec. 1.4.4 as part of the analysis of the communications data rates. (The detailed breakdowns of each message for each configuration have been omitted.)

1.4.3 System Parameters and Example Values

In addition to the message length, the required bandwidth of the communications links depends on the values of a number of system parameters. In Sec. 1.4.4, the communication data rates for the system configurations under consideration are expressed in terms of these parameters. These expressions are also evaluated as examples for a particular set of parameter values.

The values used in the examples are derived from an example deployment of 70 netted radars on a hexagonal grid covering approximately a square 270 km on a side. The radars are spaced 30 km apart and have a range of 50 from 80 to 90 km. With such a deployment up to 25 radars can see each aircraft that is not at low altitude (i.e., not masked by terrain).

A total of 1,000 aircraft is assumed to be in the surveillance volume for purposes of the example calculations. Nearly all of the data-rate components scale linearly with the number of aircraft, and plots of communications data rates as functions of the number of aircraft are provided in Sec. 1.4.4.10.

The above parameters and their example values are listed in Table 1.4.3. A number of other parameters needed in the communication bandwidth calculations are also listed. The bases for selecting each of these other parameter values for the example calculations are as follows:

Average Search Period (T_S). As a nominal value within the range specified in the Statement of Work, a value of 6 seconds was used as the search period of the track-while-scan (TWS) radars. As discussed in Sec. 1.2.2, it is likely that computer-directed-track (CDT) radars will have a longer search period for this application; a value of 12 seconds was used in the calculations.

Number of Radars Tracking a Target (N_T). In the TWS system in which selected radars cooperatively track each target, the number of trackers should be sufficient to provide a measurement of target position about every two seconds for reliable association (see Sec. 2.3). In TWS systems with a nominal radar scan period of 6 seconds, at least three radars must be assigned to track each target, pooling their data as described under Configuration 2. In the CDT systems in which each radar can track independently, it is assumed there is a primary tracker, a secondary tracker, and at least one other radar prepared to become the secondary tracker immediately if the primary or secondary trackers should become inoperative; the nominal number of trackers in these systems was taken to be three as well.

Number of STF Nodes (Operations Facilities) (N_O). For redundancy and operational convenience, there should be several Operations Facilities where the System Track File is maintained and used. In some system configurations, the System Track File is maintained only at a few locations which could be either the Operations Facilities or selected radar/data processor nodes (which could be changed from time to time). In any case, in these latter configurations the existence and use of three STF nodes was assumed where needed in the example calculations.

TABLE 1.4.3

PARAMETERS INVOLVED IN THE COMMUNICATIONS ANALYSIS

Symbol	Parameter	Value for Example
F	Fraction of Returns for Which a Distance Must be Obtained	0.5
N	Number of Nodes (Radars)	70
N_A	Number of Targets (Aircraft)	1,000
N_F	Number of False Alarms Per Scan Per Radar	100
N_O	Number of STF Nodes (Operations Facilities)	3
N_S	Number of Radars Within Tracking Range of a Target	25
N_T	Number of Radars Tracking a Target	3
T_A	Time Between Still-Alive Messages From Each Radar	1 s
T_C	Average Time Between Change of Tracking Radars for Each Target	50 s
T_H	Average Time Between Help Messages for Each Target	100 s
T_L	Average Time Between STF Label Messages for Each Target	20 s
T_P	Average Time Between Change of Primary (and Secondary) Trackers for Each Target	50 s
T_R	Average Time Between S/N Updates for Each Target	3 s
T_S	Average Radar Search Period	6 s for TWS 12 s for CDT
T_T	Time Between Tracking Measurements on Each Target (CDT Configurations)	1 s
T_U	Average Time Between STF Updates for Each Track	8 s

Average Time Between Tracking Measurements (T_T). As shown in Sec. 2.3, the time between tracking measurements should be approximately 2 seconds or less for reliable association of highly maneuvering targets. In a computer-directed-track system, each radar tracks independently and must do so at a data rate which is high enough to ensure a high probability of correct association. A rate of one measurement per second, or 1 second between measurements, is easily obtained and was used in the calculations of the example bandwidth requirements for CDT configurations.

Average Time Between STF Updates (T_U). In the system configurations which include a Local Track File, the length of time between updates of the System Track Files depends on the maneuvers performed by the target and the ability of the tracking filter to predict the future target position. With the polynomial filter implemented in TACRAN3 (see Sec. 3.5) and a 1-km limit on the deviation between the LTF and the STF, the average time between STF updates over a large number of simulation runs was found to be approximately 8 seconds, and this value was used in the communication examples. Other filters and other methods of determining when to update the STF as well as a different set of aircraft trajectories might lead to a different value. However, the same value (8 seconds) was used for all configurations. It should be noted that the results are very sensitive to this parameter.

The communications bandwidths are relatively insensitive to the following parameters:

Average Time Between Change of Tracking Radars (T_C). In all the system configurations in which each target is tracked by selected radars operating in either the TWS or CDT mode, the assignment of radars to be a part of the set that tracks a particular target will change at a rate which depends on the target velocity and to some extent on the radar deployment and selection criteria. If it is assumed that a switch from one radar to another is made each time a target travels half the distance

between radars, an average of 15 km, and that the average target velocity is 300 m/s, then the average time between changes of tracking radars is 50 seconds, which is the value used in the example calculations.

Average Time Between Change of Primary Trackers (T_P). A change in the radar designated as the primary tracker of a set of radars tracking a particular target in a CDT system, and in the radar designated as the secondary tracker, would be expected to occur on the average at the same rate as the change in tracking radars. For the values of radar spacing and target velocity assumed above, then, the average time between changes of the primary and secondary trackers of a particular target is 50 seconds.

Average Time Between $\overline{S/N}$ Updates (T_R). The average signal-to-noise ratio used in designating the tracking radars must be updated often enough to follow the variations in $\overline{S/N}$ but the update rate cannot usefully exceed the tracking rate. Three seconds was selected rather arbitrarily as the time between $\overline{S/N}$ updates to be used in the data-rate evaluations.

Average Time Between Help Messages (T_H). Since Help messages are sent only under unusual circumstances, they should be infrequent in occurrence; an average time of 100 seconds between such messages for each target is assumed here.

Time Between Still-Alive Messages (T_A). Since up-to-date knowledge of the status of other trackers in a CDT system is vital for timely and unambiguous designation of the primary, secondary, and other trackers on a decentralized basis, the Still-Alive message should be transmitted as frequently as practical. To prevent the time between measurements by a primary tracker from ever exceeding one second by a substantial amount, the time between the Still-Alive messages transmitted by each radar was specified to be one second.

Average Time Between STF Label Messages (T_L). In a TWS system with Local Track Files at the radar and the System Track Files at the Operations Facilities, a STF Label message is sent from the STF node to a radar when the radar first detects each target. If the targets move across the radar field at an average velocity of 300 m/s, then each target enters the coverage of a new radar about every 20 seconds on the average, and this value of the time between STF Label messages was used in the example calculations.

Number of False Alarms Per Scan (N_F). The number of random false alarms per scan was specified in the Statement of Work to be between 50 and 200 depending on the radar range. A value of 100 was used in the example data-rate calculations.

Fraction of Returns for Which a Distance Must be Obtained (F). In the system configurations in which the distance of a measurement from a track must be obtained from a tracker in some cases as part of the association process, the need for this distance depends on whether there are other tracks near the one in question. It was assumed rather arbitrarily that half of the returns will be from targets which are close enough to other targets to make it necessary to request a distance from another radar for unambiguous association with a track in the STF.

In addition to these parameters, there are a few special parameters peculiar to one or a few system configurations. These parameters and the values ascribed to them are explained as part of the analysis of these system configurations in the next section.

1.4.4 Data Rates

Estimates of the communications data rates involved in the operation of each of the previously described system configurations are presented in this section. The system configurations are defined in Table 1.3.1 on page 46 of Sec. 1.3. Considering the length of each type of message, the number of targets for which such a message must be sent, and

the frequency with which it must be transmitted, parametric expressions are derived for the data rates in kilobits per second over a particularly situated link. As examples, these expressions are then evaluated for the parameter values specified in the preceding section. In some cases, plots of the required data rate versus a particular parameter are presented to show the sensitivity to that parameter. The example values of the required data rates for the nine system configurations under consideration are summarized in the final subsection.

The data rates that are presented are those over the most heavily used link in the system. This worst-case link is one involved in communicating with an isolated node--one which is connected to the rest of the network by only one (two-way) link because of equipment failures or enemy actions which have rendered the other links inoperative. This situation is a worst case in that all messages to or from the node must traverse a single link, leading to the maximum required data rate over the link. The incoming and outgoing data rates are calculated and presented separately. If separate one-way links are used, each should have the capacity to handle the higher of the two data rates (i.e., the incoming rate) in the event that either of the nodes at the two ends of the link becomes an isolated node. If a single two-way link is used with time sharing between transmission directions, the link should have the capacity to handle the sum of the data rates for the two directions. This sum, while calculated for an isolated node, is in fact approximately equal to the sum of the data rates in both directions over any link in the network because the system messages are the predominant contributors to the total communications load in most cases as will be seen, and each system message must be transmitted between two nodes once in one direction or the other.

In system configurations in which the System Track File is maintained at only a few nodes, it is similarly assumed that the STF nodes are connected to the rest of the network by only a single link, and the data rates are calculated for the messages into and out of the STF node. It

is further assumed that the STF nodes are also radar/data processors nodes rather than separate Operations Facilities, although their location has only a minor effect on the communications data rates.

It should be pointed out that while the incoming single-link data rates that are presented here are peak values in the network-geometry sense described above, they are averages in terms of the spatial distribution of the targets in the surveillance volume and the temporal variations of the communications traffic. That is, it is assumed in the calculations that each radar is observing and tracking an equal fraction of the total number of targets in the surveillance volume. The effects of non-uniform target distributions can be minimized by the use of appropriate load-balancing logic and algorithms, and appropriate buffering of communications messages can smooth the more rapid fluctuations of the communications load, but the residual variations must be taken into account in determining the communications bandwidth requirements. The evaluations of the magnitude of these variations, which may be different for each of the system configurations, requires a more detailed analysis than was possible as part of this study.

1.4.4.1 Configuration 1

In the TWS system with a Local Track File (LTF) and System Track File (STF) at each node, there is only one type of inter-node message--the STF update message. The factors involved in calculating the data rate for transmitting this message are listed in Table 1.4.4. Although the table contains only two rows it serves to introduce the format of the tabulations used for all the system configurations, and it is a good vehicle for explaining the entries in the columns of the table.

The total message length in bits is simply the sum of the lengths of the header and the text. For the component lengths listed in Table 1.4.1, the header of a system message is 44 bits in length. For a tracking filter implementation which does not utilize the measurement covariances, the text of the STF update message must contain the target label,

TABLE 1.4.4

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 1

(TWS Operation--LTF and STF at Each Radar--All Radars Tracking)

Message	Message Length, bits			Message/s per Target	Number of Targets	Data Rate per Link, kbits/s	
	Header	Text	Total			Equation	Example
Incoming Messages							
STF Update	44	131	175	$F_U N_S / T_U$	$N_A - N_A / N$	$0.175 F_U N_A N_S (N-1)/NT_U$	89.8
						Total	89.8 kbits/s
Outgoing Messages							
STF Update	44	131	175	$F_U N_S / T_U$	N_A / N	$0.175 F_U N_A N_S / NT_U$	1.3
						Total	1.3 kbits/s

F_U = Fraction of tracking radars that update the STF = 1/6 in example.

the time, and the target state. Summing the lengths of these components as listed in Table 1.4.2 gives a text length of 131 bits. The total message length is then 44 + 131 = 175 bits.

If each STF update message for a particular target were transmitted by all of the tracking radars, then the number of messages per second for that target would be simply the number of trackers (N_S) divided by the time between updates (T_U). Actually, there would be no need for all the trackers to transmit an update message each time an update is required; in fact, a message from one radar--presumably the one that first detects a maneuver--would be sufficient. However, with transmission delays, some of the trackers may have sent out their update messages before they receive the update from another tracker that was triggered by the same maneuver. The factor F_U, representing the fraction of radars that send STF update messages in response to each maneuver, was introduced to account for this effect. For $N_S = 25$, the value of N_S could range from 1/25 to 1; a value of 1/6, corresponding to an average of about four trackers sending update messages for each target each time an update is required, was selected somewhat arbitrarily for use in the example data rate calculation. This value corresponds to data processing and communication-link delays of a little more than a second (for T_U = 8 seconds). (The sensitivity of the results to F_U is shown below.)

The number of targets for which a STF Update message is received by each node is the total number of targets in track (N_A) minus the portion of these targets for which the radar at that node is responsible (an average of N_A/N). The expression for the data rate in bits per second is simply the product of the message length in bits times the number of messages per second for each target times the number of targets. This product for the incoming STF Update messages (after dividing by 1,000 to convert to kilobits per second per link) is given in the next-to-last column of Table 1.4.4. For $F_U = 1/6$ and the other parameter values given in Table 1.4.3, the value of this expression for the incoming data rate is 89.8 kbits per second per link as given in the last column of Table 1.4.4.

The calculation of the data rate for the outgoing STF Update messages is similar except that the average number of targets for which the message must be sent is N_A/N, the effective number of targets being tracked by the radar. For the specified parameter values, the outgoing data rate is then 1.3 kbits per second per link as given in Table 1.4.4.

Variations in the value of F_U, the fraction of tracking radars that send STF Update messages, have a proportionate effect on the data rate. For example, if a scheme were devised whereby only one radar updates the STF (i.e., there is no redundancy in the update) then incoming data would drop from 90 kbits per second to only 22 kbits per second. If, on the other hand, all the trackers update the STF, the data rate increases to 540 kbits per second. The outgoing data rate would also change by the same proportions. The relationship between the incoming data rate and F_U is plotted in Fig. 1.4.1.

If a Kalman filter were used for tracking, the covariance matrix would have to be transmitted along with the target state as part of the STF Update message. In this case, the length of the message text would be 422 bits[1] and the total message length would be 466 bits. The incoming and outgoing rates would be increased proportionately to 239 and 3.5 kilobits per second, respectively, for $F_U = 1/6$ and the specified values of the other parameters.

1.4.4.2 Configuration 2

With only selected radars tracking each target and a Distributed Local Track File (DLTF) and STF at each of these radars, a number of types of messages must be transmitted from node to node. These message types are listed in Table 1.4.5; their purpose and content are described in detail in Sec. 3.5.

---

[1] Assumes 31 bits for label and time, 76 bits for position and velocity and 315 bits for 21 elements (prescaled) of a symmetrical 6×6 matrix.

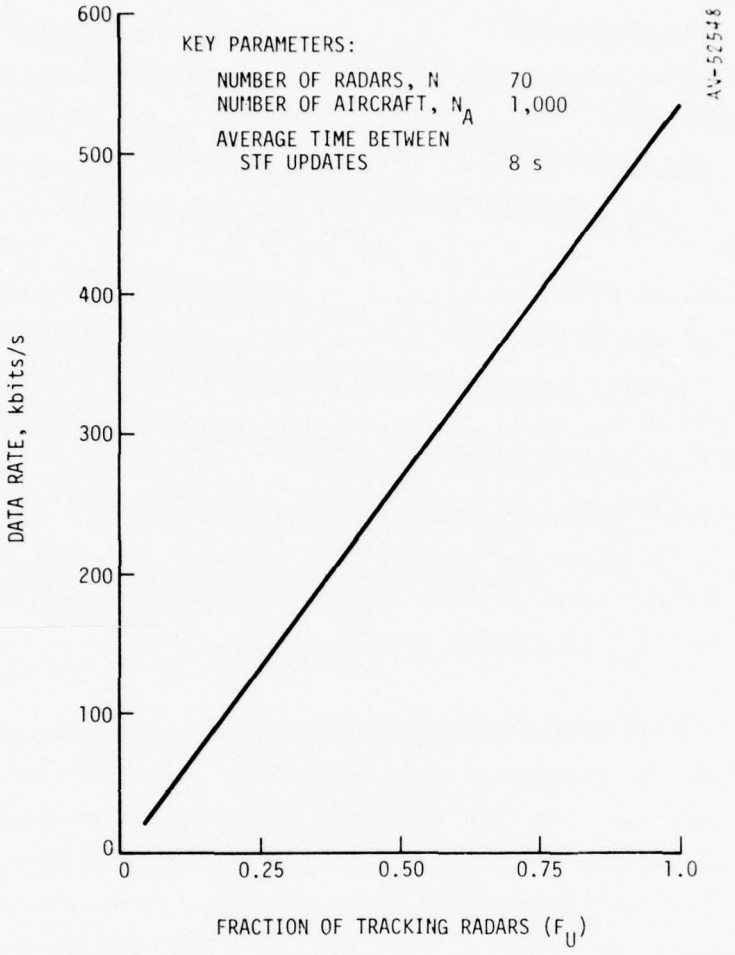


Figure 1.4.1. Data Rate on Incoming Link Versus Fraction of Tracking Radars Sending STF Update Message in Configuration 1

The first five messages listed in each part of Table 1.4.5 are directed messages sent to one, or in the case of the measurement message, two particular nodes. A directed message over the link to an isolated node has only one address, however, and the length of the header was determined on that basis (i.e., with $N_R = 1$). In the case of the Distance Request and Distance Reply messages, the length so obtained was divided by five under the rather arbitrary assumption that an average of five requests or replies to the same radar are combined in each message. The

TABLE 1.4.5

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 2

(TWS Operation--DLTF and STF at Each Radar--Selected Radars Tracking)

Incoming Messages

Message	Message Length, bits			Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s	
	Header	Text	Total			Equation	Example
Distance Request	10*	85	95	$F(N_S-N_T)/T_S$	N_A/N	$0.095FN_A(N_S-N_T)/NT_S$	2.5
Distance Reply	10*	32	42	F/T_S	$N_A(N_S-N_T)/N$	$0.042FN_A(N_S-N_T)/NT_S$	1.1
DLT Request	52	24	76	$1/T_C$	N_A/N	$0.076N_A/NT_C$	0
DLT Reply	52	439	491	$1/T_C$	N_A/N	$0.491N_A/NT_C$	0.1
Measurement	52	77	129	$(N_T-1)/T_S$	N_AN_T/N	$0.129N_AN_T(N_T-1)/NT_S$	1.8
STF Update	44	206	250	$1/T_U$	N_A-N_A/N	$0.250N_A(N-1)/NT_U$	30.8
Drop Tracker	44	24	68	$1/T_C$	N_A-N_A/N	$0.068N_A(N-1)/NT_C$	1.3
Add Tracker	44	47	91	$1/T_C$	N_A-N_A/N	$0.091N_A(N-1)/NT_C$	1.8
Help	44	40	84	$1/T_H$	N_A-N_A/N	$0.084N_A(N-1)/NT_H$	0.8
						Total	40.2 kbits/s

Outgoing Messages

Distance Request	10*	85	95	F/T_S	$N_A(N_S-N_T)/N$	$0.095FN_A(N_S-N_T)/NT_S$	2.5
Distance Reply	10*	32	42	$F(N_S-N_T)/T_S$	N_A/N	$0.042FN_A(N_S-N_T)/NT_S$	1.1
DLT Request	52	24	76	$1/T_C$	N_A/N	$0.076N_A/NT_C$	0.0
DLT Reply	52	439	491	$1/T_C$	N_A/N	$0.491N_A/NT_C$	0.1
Measurement	60	77	137	$1/T_S$	N_AN_T/N	$0.137N_AN_T/NT_S$	1.0
STF Update	44	206	250	$1/T_U$	N_A/N	$0.250N_A/NT_U$	0.4
Drop Tracker	44	24	68	$1/T_C$	N_A/N	$0.068N_A/NT_C$	0.0
Add Tracker	44	47	91	$1/T_C$	N_A/N	$0.091N_A/NT_C$	0.0
Help	44	40	84	$1/T_H$	N_A/N	$0.084N_A/NT_H$	0.0
						Total	5.1 kbits/s

*Assumes that an average of five requests or replies are combined in each message.

lengths of the texts are the sums of the numbers of bits in the appropriate components as listed in Table 1.4.2.

The last four types of messages in Table 1.4.5 are system messages sent to all the other nodes in the network. The text lengths were again obtained using the component lengths in Table 1.4.2. The text of the STF update message is longer than that for the previous system because data on the tracking radars must be included.

The numbers of messages per second per target depend upon the number of nodes which transmit each type of incoming message or receive each type of outgoing message, and on the time between transmissions or receptions of each message for each target. For example, a Distance Request message could be sent to the isolated node under consideration by each radar which can see the target but is not tracking it--that is by $(N_S - N_T)$ nodes--and such a message would be sent each scan period. A Distance Reply message would then be sent in response to each of these nodes. Actually, an accurate target-to-track distance would be required only for some fraction (here represented by F) of the returns. The average number of target tracks for which such a request would be received by the node under consideration is $1/N_T$ times the number of targets it is tracking, or N_A/N. Similarly, a Distance Request message would be sent by the node, and a Distance message would be received, for a fraction F of the $N_A(N_S - N_T)/N$ targets which are seen but are not being tracked by the radar.

The number of messages per second per target and the number of targets for the other types of messages are similarly determined. The data rate for each message type is then just the product of these two factors times the message length. The expressions for the data rates that are so obtained are given in the next-to-last column of Table 1.4.5. In all cases, the data rate is directly proportional to the number of targets N_A and inversely proportional to the number of nodes N. These expressions were evaluated for the parameter values in Table 1.4.3 to give the

example data rates in the last column. As shown, the sum of these data rates is 40.2 kbits per second for the incoming messages and 5.1 kbits per second for the outgoing messages. The STF Update message is by far the largest contributor to the incoming data rate (and to the total of the two components). The dependence of the incoming data rate on the STF Update time T_U is shown in Fig. 1.4.2.

Since the number of requests combined in one Distance Request message (and the number of responses combined in one Distance Reply message) and the fraction of the returns for which a distance must be obtained from another node were both selected rather arbitrarily, the sensitivity to these parameters was examined and the results obtained for the incoming data rate are plotted in Fig. 1.4.3. As might be expected since the contribution of the Distance Request and Distance Reply messages to the incoming data rate is small compared with that of the STF Update message,

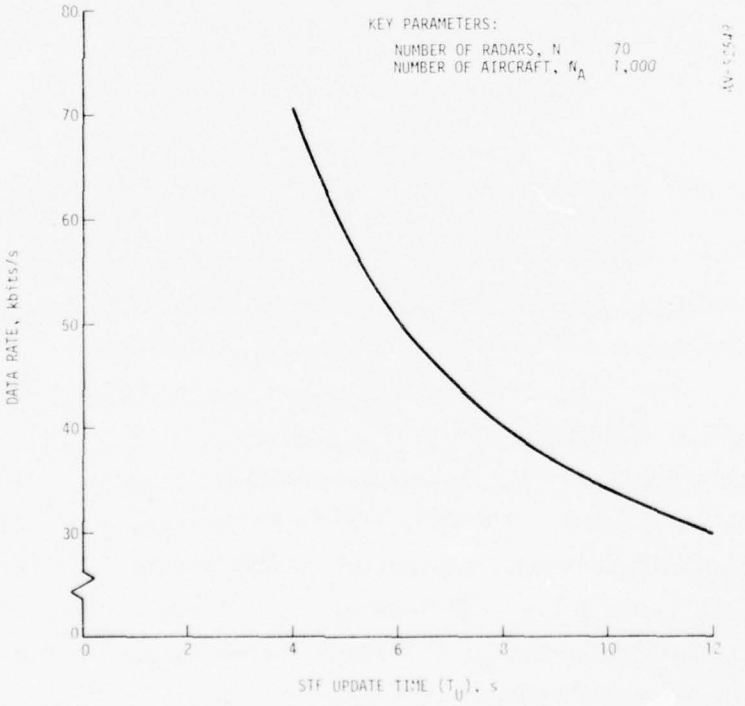


Figure 1.4.2. Data Rate on Incoming Link Versus Time Between STF Updates for Configuration 2

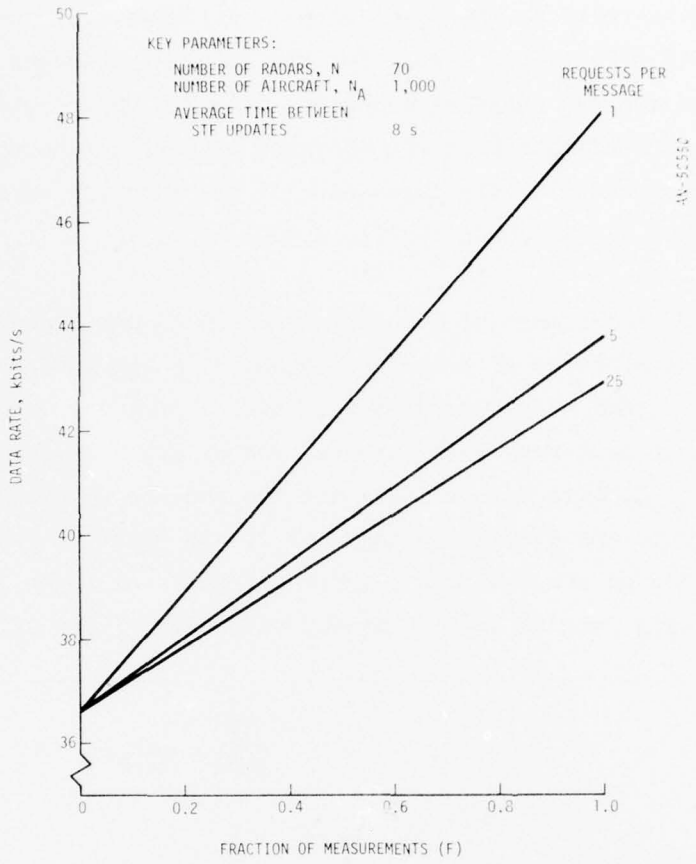


Figure 1.4.3. Data Rate on Incoming Link Versus Fraction of Measurements for Which a Distance is Requested in Configuration 2

the data rate per link is not particularly sensitive to variations in these parameters.

1.4.4.3 Configuration 3

With only an LTF at the radars and the STF only at the Operations Facilities or special STF nodes, there are just two types of messages that must be communicated--the STF Update message and the STF Label message--as listed in Table 1.4.6. The calculation of the data-rate capacity required for the transmission of these messages is somewhat different that for the previous two system configurations, however, because the messages must all be transmitted to or from the Operations Facilities. Consequently

TABLE 1.4.6

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 3

(TWS Operation--LTF Only at Each Radar--All Radars Tracking)

Message	Message Length, bits			Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s	
	Header	Text	Total			Equation	Example
Incoming Messages to STF Node							
STF Update	52	131	183	N_S/T_U	N_A-N_A/N	$0.183N_AN_S(N-1)/NT_U$	564
STF Label	52	24	76	$1/T_L$	N_A/N	$0.076N_A/NT_L$	0
						Total	564 kbits/s
Outgoing Messages from STF Node							
STF Update	60	131	191	N_S/T_U	N_A/N	$0.191N_AN_S/NT_U$	8.5
STF Label	52	24	76	$1/T_L$	N_A-N_A/N	$0.076N_A(N-1)/NT_L$	3.7
						Total	12.2 kbits/s

the data rate is high over network links close to the Operations Facilities and becomes lower with increasing distance from the Operations Facilities. If any node in the network must be able to serve as an Operations Facility or STF node, then every link must have the high capacity required of the direct links to such facilities.

It is the data rate over a high-traffic link directly connected to the Operations Facility or STF node that is of primary interest here and is presented in Table 1.4.6. In preparing this table it was assumed that an Operations Facility is located at a radar/data processor node, but the results are not very sensitive to this assumption. Each radar that is within range of the target updates the System Track File when it detects a maneuver, so the number of STF Update messages per target is just N_S/T_U . An STF Label message is sent whenever a target is newly acquired by a radar. Evaluating the expressions obtained for the data rates using the parameter values in Table 1.4.3 gives the data-rate values in the last column of Table 1.4.6. The incoming data rate (and the total two-way rate) is dominated by the contribution of the STF Update message.

1.4.4.4 Configuration 4

With a System Track File, but no Local Track File, at each radar, the STF must be updated with each measurement and the update must be sent to all of the other nodes. The STF Update message must be sent by each radar that sees the target. These requirements are reflected in Table 1.4.7 which gives the data rate for the single type of message transmitted in this system configuration.

1.4.4.5 Configuration 5

With no track files at the radars, each measurement must be sent to the Operations Facilities where the System Track File is maintained. Since the false alarms cannot be distinguished from target returns at the radar, their data must be transmitted to the Operations Facilities along with that of the actual returns.

TABLE 1.4.7

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR TWS OPERATION WITH CONFIGURATION 4

(TWS Operation--STF Only at Each Radar--All Radars Tracking)

Message	Message Length, bits			Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s	
	Header	Text	Total			Equation	Example
Incoming Messages							
STF Update	44	131	175	N_S/T_S	N_A-N_A/N	$0.175N_AN_S(N-1)/NT_S$	719
						Total	719 kbits/s
Outgoing Messages							
STF Update	44	131	175	$1/T_S$	N_AN_S/N	$0.175N_AN_S/NT_S$	10.4
						Total	10.4 kbits/s

As for other configurations in which all the messages must be sent to a few STF nodes, the traffic on the links leading directly to the Operations Facilities will be relatively high and that on links in parts of the network far from the Operations Facilities will be considerably lower. As discussed previously, the data rates on these high-traffic links are of primary interest for communications sizing purposes, and it is those data rates which are given in Table 1.4.8. The calculated data rates include the contribution of the false alarms, which amounts to about 22% of the total. Interestingly, the incoming data rate for this configuration, in which the radar measurements are transmitted directly to the Operations Facilities, is somewhat less than for the previous configuration in which the processing of the measurements is done at the radars and only the System Track File updates are transmitted to other nodes. This is because raw measurements contain less data than processed STF updates.

1.4.4.6 Configuration 6

In the computer-directed-track systems in which selected radars track each target, messages of various types must be transmitted as part of the decentralized selection process and to obtain needed track data from other nodes. The types of messages needed in the CDT system with a Local Track File and System Track File at each radar are listed in Table 1.4.9. Some of these message types are the same as those in the TWS system with Distributed Local Track Files discussed in Sec. 1.4.4.2. New message types are the Average Signal-to-Noise Ratio Update used in designating the tracking radars, the Still-Alive message sent to the other trackers to insure continuous tracking if a tracker should become inoperative, and the Tracker Status message designating a new primary or secondary tracker. The message lengths were determined as discussed previously using the data in Tables 1.4.1 and 1.4.2. Expressions for the data-rate contributions of each message type are given in the next-to-last column as in previous tables in this section, and the values of these data rates for the example parameter values in Table 1.4.3 are given in the last column. As indicated in the footnote to the table, it was assumed in

TABLE 1.4.8

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 5

(TWS Operation--No Track Files at the Radars--All Radars Tracking)

Message	Message Length, bits			Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s	
	Header	Text	Total			Equation	Example
Incoming Messages to STF Node							
Measurement	52	77	129	$N_S/T_S+NN_F/N_AT_S$	N_A-N_A/N	$0.129(N-1)(N_AN_S/N+N_F)/T_S$	678
						Total	678 kbits/s
Outgoing Messages From STF Node							
Measurement	60	77	137	$1/T_S+NN_F/N_AN_ST_S$	N_AN_S/N	$0.137(N_AN_S/N+N_F)/T_S$	10.4
						Total	10.4 kbits/s

TABLE 1.4.9

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 6

(CDT Operation--LTF and STF at Each Radar--Selected Radars Tracking)

Message	Message Length, bits: Header	Text	Total	Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s: Equation	Example
Incoming Messages							
Distance Request	10*	85	95	$F(N_S-N_T)/T_S$	N_A/N	$0.095FN_A(N_S-N_T)/NT_S$	1.2
Distance Reply	10*	32	42	F/T_S	$N_A(N_S-N_T)/N$	$0.042FN_A(N_S-N_T)/NT_S$	0.6
$\overline{S/N}$ Update	52	24	76	$(N_T-1)/T_R$	N_AN_T/N	$0.076N_AN_T(N_T-1)/NT_R$	2.2
Still Alive	52	15	67			$0.067N_S/T_A$+	1.7
STF Update	44	166	210	$1/T_U$	N_A-N_A/N	$0.210N_A(N-1)/NT_U$	25.9
Drop Tracker	44	24	68	$1/T_C$	N_A-N_A/N	$0.068N_A(N-1)/NT_C$	1.3
ADD Tracker	44	47	91	$1/T_C$	N_A-N_A/N	$0.091N_A(N-1)/NT_C$	1.8
Tracker Status	44	39	83	$1/T_P$	N_A-N_A/N	$0.083N_A(N-1)/NT_P$	1.6
Help	44	40	84	$1/T_H$	N_A-N_A/N	$0.084N_A(N-1)/NT_H$	0.8
						Total	37.1 kbits/s
Outgoing Messages							
Distance Request	10*	85	95	F/T_S	$N_A(N_S-N_T)/N$	$0.095FN_A(N_S-N_T)/NT_S$	1.2
Distance Reply	10*	32	42	$F(N_S-N_T)/T_S$	N_A/N	$0.042FN_A(N_S-N_T)/NT_S$	0.6
$\overline{S/N}$ Update	60	24	84	$1/T_R$	N_AN_T/N	$0.084N_AN_T/NT_R$	1.2
Still Alive	60	15	75			$0.075/T_A$	0.1
STF Update	44	166	210	$1/T_U$	N_A/N	$0.210N_A/NT_U$	0.4
Drop Tracker	44	24	68	$1/T_C$	N_A/N	$0.068N_A/NT_C$	0.0
Add Tracker	44	47	91	$1/T_C$	N_A/N	$0.091N_A/NT_C$	0.0
Tracker Status	44	39	83	$1/T_P$	N_A/N	$0.083N_A/NT_P$	0.0
Help	44	40	84	$1/T_H$	N_A/N	$0.084N_A/NT_H$	0.0
						Total	3.5 kbits/s

*Assumes that an average of five requests or replies are combined in each message.

+Assumes that N_S radars are involved in tracking all of the targets tracked by a particular radar.

determining the data rate for the Still-Alive message that the radar at the node under consideration is paired as a tracker with a total of N_S other radars; this assumption is rather arbitrary but conservative. Note that all of the data rate components are proportional to the number of targets except for that for the Still-Alive message, which is transmitted periodically by each radar independent of the number of targets. However, since the contribution of the Still-Alive message is negligibly small, the total incoming and outgoing data rates are essentially proportional to the number of targets. Note too that again the major contributor to the incoming data rate is the STF Update message.

1.4.4.7 Configuration 7

The set of message types for this CDT system configuration includes those for the preceding configuration plus a Measurement message sent to an Operations Facility for some returns to determine whether or not they associate with the full System Track File, and an Association message for reporting the results of the association processing. Actually there are three possible types of Association messages depending on the association outcome as discussed in Sec. 1.3.2.7; as indicated in a footnote to Table 1.4.10, the text length for the Association message is a weighted average of the three possible text lengths, assuming 80% "no association" outcomes (includes virtually all the false alarms) with a text length of 16 bits (target label), 10% "association/become-a-tracker" outcomes with a text length of 166 bits, and 10% "association/do-not-become-a-tracker" outcomes with a text length of 16 bits. The Measurement and Association messages are transmitted for only a small fraction of the returns--those for which the target track is not contained in the Partial System Track File at the radar. This fraction is represented by the symbol F_M in the data-rate expressions, and its value is taken to be 0.1 in the example calculations. In addition, these messages must be transmitted for each of the false alarms. The Measurement messages could all be sent to the Operations Facility to obtain Association-messsage responses, but it is assumed here that each Measurement message is sent to a node in the vicinity of the target which will have the track in its Partial System Track File if the target has in fact been acquired by any radar.

TABLE 1.4.10

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 7

(CDT Operation--LTF and PSTF at Each Radar--Selected Radars Tracking)

Message	Message Length, bits: Header	Text	Total	Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s: Equation	Example
Incoming Messages							
Measurement	52	77	129	$F_M N_S/T_S + NN_F/N_A T_S$	N_A/N	$0.129(F_M N_A N_S/N+N_F)/T_S$	1.5
Association	52	30[§]	80	F_M/T_S	$N_A N_S/N+N_F/F_M$	$0.082(F_M N_A N_S/N+N_F)/T_S$	0.9
Distance Request	10[*]	85	95	$F(N_S-N_T)/T_S$	N_A/N	$0.095FN_A(N_S-N_T)/NT_S$	1.2
Distance Reply	10[*]	32	42	F/T_S	$N_A(N_S-N_T)/N$	$0.042FN_A(N_S-N_T)/NT_S$	0.6
$\overline{S/N}$ Update	52	24	76	$(N_T-1)/T_R$	$N_A N_T/N$	$0.076N_A N_T(N_T-1)/NT_R$	2.2
Still Alive	52	15	67			$0.067N_S/T_A$[+]	1.7
STF Update	52	166	218	$1/T_U$	N_A-N_A/N	$0.218N_A(N-1)/NT_U$[¶]	26.9
Drop Tracker	52	24	76	$1/T_C$	$N_A N_S/N-N_A/N$	$0.076N_A(N_S-1)/NT_C$	0.5
Add Tracker	52	47	99	$1/T_C$	$N_A N_S/N-N_A/N$	$0.099N_A(N_S-1)/NT_C$	0.7
Tracker Status	52	39	91	$1/T_P$	$N_A N_S/N-N_A/N$	$0.091N_A(N_S-1)/NT_P$	0.6
Help	52	40	92	$1/T_H$	$N_A N_S/N-N_A/N$	$0.092N_A(N_S-1)/NT_H$	0.3
						Total	37.1 kbits/s
Outgoing Messages							
Measurement	52	77	129	F_M/T_S	$N_A N_S/N+N_F/F_M$	$0.129(F_M N_A N_S/N+N_F)/T_S$	1.5
Association	52	30[§]	82	$F_M N_S/T_S + NN_F/NAT_S$	N_A/N	$0.082(F_M N_A N_S/N+N_F)/T_S$	0.9
Distance Request	10[*]	85	95	F/T_S	$N_A(N_S-N_T)/N$	$0.095FN_A(N_S-N_T)/NT_S$	1.2
Distance Reply	10[*]	32	42	$F(N_S-N_T)/T_S$	N_A/N	$0.042FN_A(N_S-N_T)/NT_S$	0.6
$\overline{S/N}$ Update	60	24	84	$1/T_R$	$N_A N_T/N$	$0.084N_A N_T/NT_R$	1.2
Still Alive	60	15	75			$0.075/T_A$	0.1
STF Update	60	166	226	$1/T_U$	N_A/N	$0.226N_A/NT_U$	0.4
Drop Tracker	60	24	84	$1/T_C$	N_A/N	$0.084N_A/NT_C$	0.0
Add Tracker	60	47	107	$1/T_C$	N_A/N	$0.107N_A/NT_C$	0.0
Tracker Status	60	39	99	$1/T_P$	N_A/N	$0.099N_A/NT_P$	0.0
Help	60	40	00	$1/T_H$	N_A/N	$0.100N_A/NT_H$	0.0
						Total	5.9 kbits/s

[*] Assumes that an average of five requests or replies are combined in each message.

[+] Assumes that N_S radars are involved in tracking all of the targets tracked by a particular radar.

[§] Weighted average over the three possible types of "Association" messages, assuming 80% "No Association," 10% "Association/Tracker," and 10% "Association/Non-Tracker" messages.

[¶] Over links into STF nodes only.

The data-rate expressions for the other directed messages are the same as for the previously considered configuration, but the expressions for the "system" messages are different because they must now be sent to only the radars that have been determined to be within range of the target, a difference which increases the header length (the "system" messages must now be addressed) but in most cases reduces the number of targets for which the messages must be received. However, all the STF Update messages must be sent to the Operations Facilities, and the incoming data rate for this message in Table 1.4.10 is for the high-traffic link into the node at which the Operations Facility is located (the STF nodes). The data rate into the other nodes would be lower by 9.3 kbits per second for the parameter values used in the example--i.e., the total incoming data rate would be 27.8 kbits per second at the non-STF nodes. At any node, the STF Update message is still the dominant contributor to the incoming data rate. As in the previous configurations, the outgoing data rate is small.

1.4.4.8 Configuration 8

The set of message types for this CDT system configuration is the same as for Configuration 7. However, with no STF at the radars, a Measurement message must be sent to an Operations Facility for each return that does not associate with the LTF and for each false alarm, and an Association message must be sent back in response to each of the Measurement messages. Furthermore, these messages must all be transmitted over the links leading to and from the Operations Facilities, as must all the Distance Request, Distance Reply, and STF Update messages. The data-rate expressions and example values are given in Table 1.4.11 for all the messages based on the assumption that the Operations Facilities are located at radar/data processor nodes. The major contributors to the total incoming data rate are the Measurement, Association, and STF Update messages. In contrast to the situation for the other configurations, the outgoing data rate is fairly high--about half of the incoming rate--because of the high data rates required to transmit all of the Association and Distance messages.

TABLE 1.4.11

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 8

(CDT Operations--LTF Only at Each Radar--Selected Radars Tracking)

Message	Message Length, bits: Header	Text	Total	Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s: Equation	Example
Incoming Messages to STF Node							
Measurement	52	77	129	$(N_S-N_T)/T_S+NN_F/N_AT_S$	$N_A/N_O-N_A/N$	$0.129(N/N_O-1)[N_A(N_S-N_T)/N+N_E]/N_T$	99.5
Association	52	30[§]	82	$1/T_S$	$N_A(N_S-N_T)/N+N_F$	$0.082[N_A(N_S-N_T)/N+N_E]/T_S$	2.8
Distance Request	10[★]	85	95	$F(N_S-N_T)/T_S$	$N_A/N_O-N_A/N$	$0.095FN_A(N_S-N_T)(N/N_O-1)/NT_S$	27.8
Distance Reply	10[★]	32	42	F/T_S	$N_A(N_S-N_T)/N$	$0.042FN_A(N_S-N_T)/NT_S$	0.6
$\overline{S/N}$ Update	52	24	76	$(N_T-1)/T_R$	N_AN_T/N	$0.076N_AN_T(N_T-1)/NT_R$	2.2
Still-Alive	52	15	67			$0.067N_S/T_A$[†]	1.7
STF Update	52	166	218	$1/T_U$	N_A-N_A/N	$0.218N_A(N-1)/NT_U$	26.9
Drop Tracker	52	24	76	$1/T_C$	N_A-N_A/N	$0.076N_A(N-1)/NT_C$	1.5
Add Tracker	52	47	99	$1/T_C$	N_A-N_A/N	$0.099N_A(N-1)/NT_C$	2.0
Tracker Status	52	39	91	$1/T_P$	N_A-N_A/N	$0.091N_A(N-1)/NT_P$	1.8
Help	52	40	92	$1/T_H$	N_A-N_A/N	$0.092N_A(N-1)/NT_H$	0.9
						Total	167.7 kbits/s
Outgoing Messages from STF Node							
Measurement	52	77	129	$1/T_S$	$N_A(N_S-N_T)/N+N_F$	$0.129[N_A(N_S-N_T)/N+N_E]/T_S$	4.5
Association	52	30[§]	82	$(N_S-N_T)/T_S+NN_F/N_AT_S$	$N_A/N_O-N_A/N$	$0.082(N/N_O-1)[N_A(N_S-N_T)/N+N_E]/T_S$	63.2
Distance Request	10[★]	85	95	F/T_S	$N_A(N_S-N_T)/N$	$0.095FN_A(N_S-N_T)/NT_S$	1.2
Distance Reply	10[★]	32	42	$F(N_S-N_T)/T_S$	$N_A/N_O-N_A/N$	$0.042FN_A(N_S-N_T)(N/N_O-1)/NT_S$	12.3
$\overline{S/N}$ Update	60	24	84	$1/T_R$	N_AN_T/N	$0.084N_AN_T/NT_R$	1.2
Still-Alive	60	15	75			$0.075/T_A$	0.1
STF Update	60	166	226	$1/T_U$	N_A/N	$0.226N_A/NT_U$	0.4
Drop Tracker	60	24	84	$1/T_C$	N_A/N	$0.084N_A/NT_C$	0.0
Add Tracker	60	47	107	$1/T_C$	N_A/N	$0.107N_A/NT_C$	0.0
Tracker Status	60	39	99	$1/T_P$	N_A/N	$0.099N_A/NT_P$	0.0
Help	60	40	100	$1/T_H$	N_A/N	$0.100N_A/NT_H$	0.0
						Total	82.9 kbits/s

[★] Assumes that an average of five requests or replies are combined in each message.

[†] Assumes that N_S radars are involved in tracking all of the targets tracked by a particular radar.

[§] Weighted average over the three possible types of "Association" messages.

1.4.4.9 Configuration 9

With an STF but no LTF at each radar, all of the association processing is done at the radar which makes the measurement, and there is no need for the Measurement, Association, Distance Request, and Distance Reply messages. However, the STF must be updated with each tracking measurement made by the primary tracker, and for the tracking rate of one per second used in the example, the STF Update messages constitute virtually all the incoming communications traffic and most of the relatively small outgoing traffic as indicated in Table 1.4.12.

1.4.4.10 Summary

The communications data rates calculated as examples using the parameter values in Table 1.4.3 for each of the nine system configurations considered are summarized in Table 1.4.13. The sum of the incoming and outgoing data rates over a single link to an isolated node is given along with the separate values of these two rates. The incoming data rates, which are considerably larger than the outgoing rates in all cases, are plotted in Fig. 1.4.4 as functions of the total number of aircraft in track. As indicated in the table, the data rates for some configurations are for any node in the network which through attrition becomes connected by only one link to the rest of the network (and in fact the sum of the incoming and outgoing rates for this node is approximately equal to the total two-way data rate on any link in the network), while for other configurations the data rates are those on the high-traffic links leading to and from the Operations Facilities located at certain radar/data processor nodes. In these latter configurations, all the links would have to have sufficient capacity to accommodate the indicated data rates if the Operations Facility could be located at any node, but many of the links would not be operating at their full capacity at any given time and the excess bandwidth might be used for ECCM purposes.

For the example results summarized in Table 1.4.13 and Fig. 1.4.4, the communications data rates are lower for configurations in which

TABLE 1.4.12

COMMUNICATIONS DATA-RATE REQUIREMENTS FOR CONFIGURATION 9

(CDT Operation--STF Only at Each Radar--Selected Radars Tracking)

Message	Message Length, bits			Messages/s per Target	Number of Targets	Data Rate per Link, kbits/s	
	Header	Text	Total			Equation	Example
Incoming Messages							
$\overline{S/N}$ Update	52	24	76	$(N_T-1)/T_R$	$N_A N_T/N$	$0.076N_A N_T(N_T-1)/NT_R$	2.2
Still Alive	52	15	67			$0.067N_S/T_A$*	1.7
STF Update	44	166	210	$1/T_T$	N_A-N_A/N	$0.210N_A(N-1)/NT_T$	207.0
Drop Tracker	44	24	68	$1/T_C$	N_A-N_A/N	$0.068N_A(N-1)/NT_C$	1.3
Add Tracker	44	47	91	$1/T_C$	N_A-N_A/N	$0.091N_A(N-1)/NT_C$	1.8
Tracker Status	44	39	83	$1/T_P$	N_A-N_A/N	$0.083N_A(N-1)/NT_P$	1.6
Help	44	40	84	$1/T_H$	N_A-N_A/N	$0.084N_A(N-1)/NT_H$	0.8
						Total	216.4 kbits/s
Outgoing Messages							
$\overline{S/N}$ Update	60	24	84	$1/T_R$	$N_A N_T/N$	$0.084N_A N_T/NT_R$	1.2
Still Alive	60	15	75			$0.075/T_A$	0.1
STF Update	44	166	210	$1/T_T$	N_A/N	$0.210N_A/NT_T$	3.0
Drop Tracker	44	24	68	$1/T_C$	N_A/N	$0.068N_A/NT_C$	0.0
Add Tracker	44	47	91	$1/T_C$	N_A/N	$0.091N_A/NT_C$	0.0
Tracker Status	44	39	83	$1/T_P$	N_A/N	$0.083N_A/NT_P$	0.0
Help	44	40	84	$1/T_H$	N_A/N	$0.084N_A/NT_H$	0.0
						Total	4.3 kbits/s

*Assumes that N_S radars are involved in tracking all of the targets tracked by a particular radar.

TABLE 1.4.13

SUMMARY OF COMMUNICATIONS DATA-RATE REQUIREMENTS

Configuration Number	Data Rate in kbits/s Over Link to Isolated Node		
	Incoming	Outgoing	Total
1	90	1	91
2	40	5	45
3	564*	12*	576*
4	719	10	729
5	678*	10*	688*
6	37	4	41
7	37*	6	43*
8	168*	83*	251*
9	216	4	220

*Over link to Operations Facility or STF node only.

there is both a Local Track File and System Track File of some sort at each node than for configurations in which there is only one or none of these types of files at each node. The data rates are generally lower for the computer-directed-track systems with selected radars tracking than for the track-while-scan systems with all radars within view tracking each target. However, the data rate for the only track-while-scan (TWS) system in which selected radars are used for tracking (Configuration 2) is comparable to the lowest data rate for the computer-directed-track (CDT) systems. These lower data rates (of the order of 40 kilobits per second) lead to communications bandwidth requirements which are certainly reasonable for line-of-sight communication links.

AD-A067 936 GENERAL RESEARCH CORP SANTA BARBARA CALIF F/G 17/9
TACTICAL FORWARD AREA SURVEILLANCE AND CONTROL INTERNETTING STU--ETC(U)
NOV 78 G W DELEY, J H BALLANTINE F19628-78-C-0020
UNCLASSIFIED CR-1-814-VOL-2 ESD-TR-78-180-VOL-2 NL

2 OF 5
ADA
067936

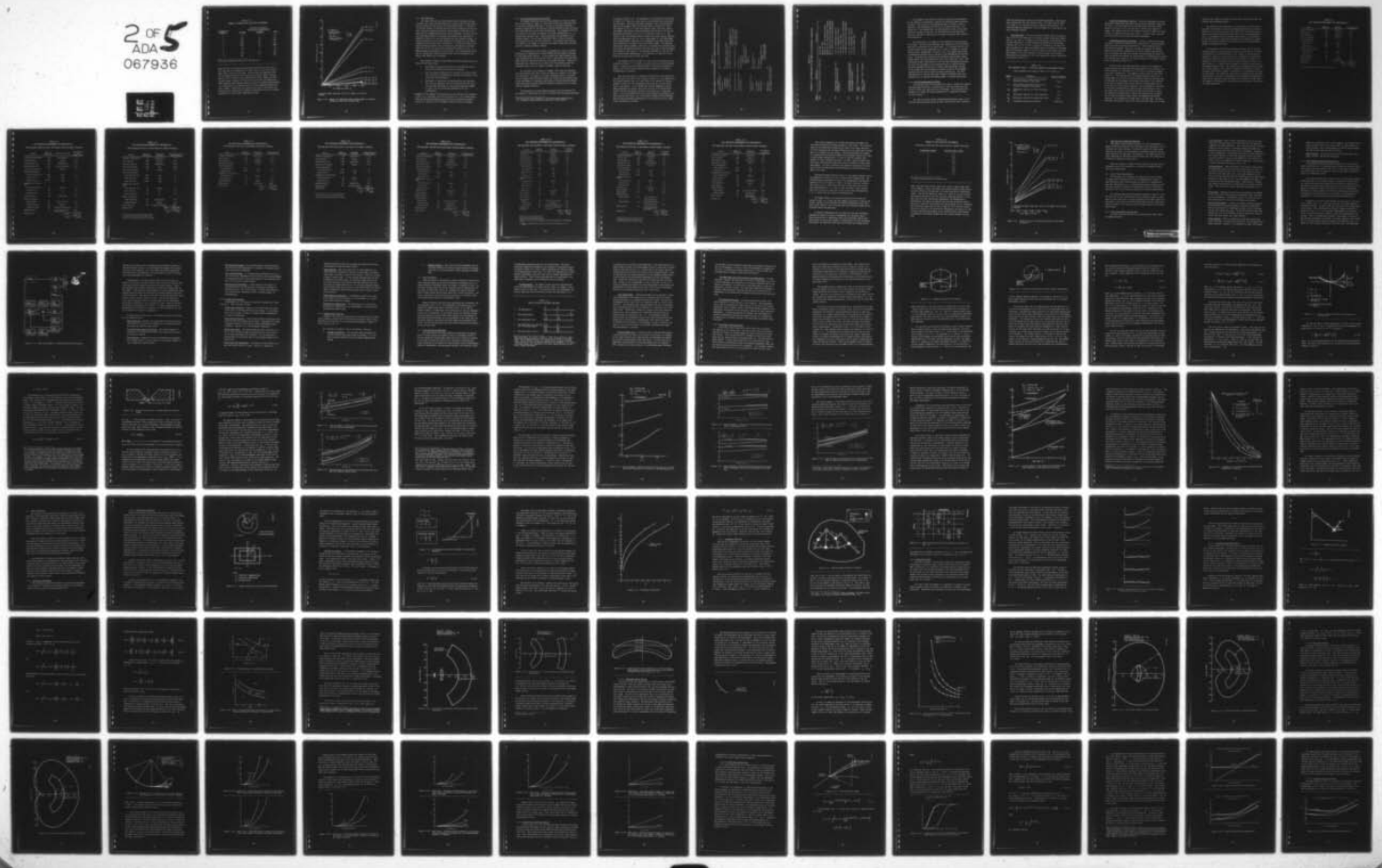


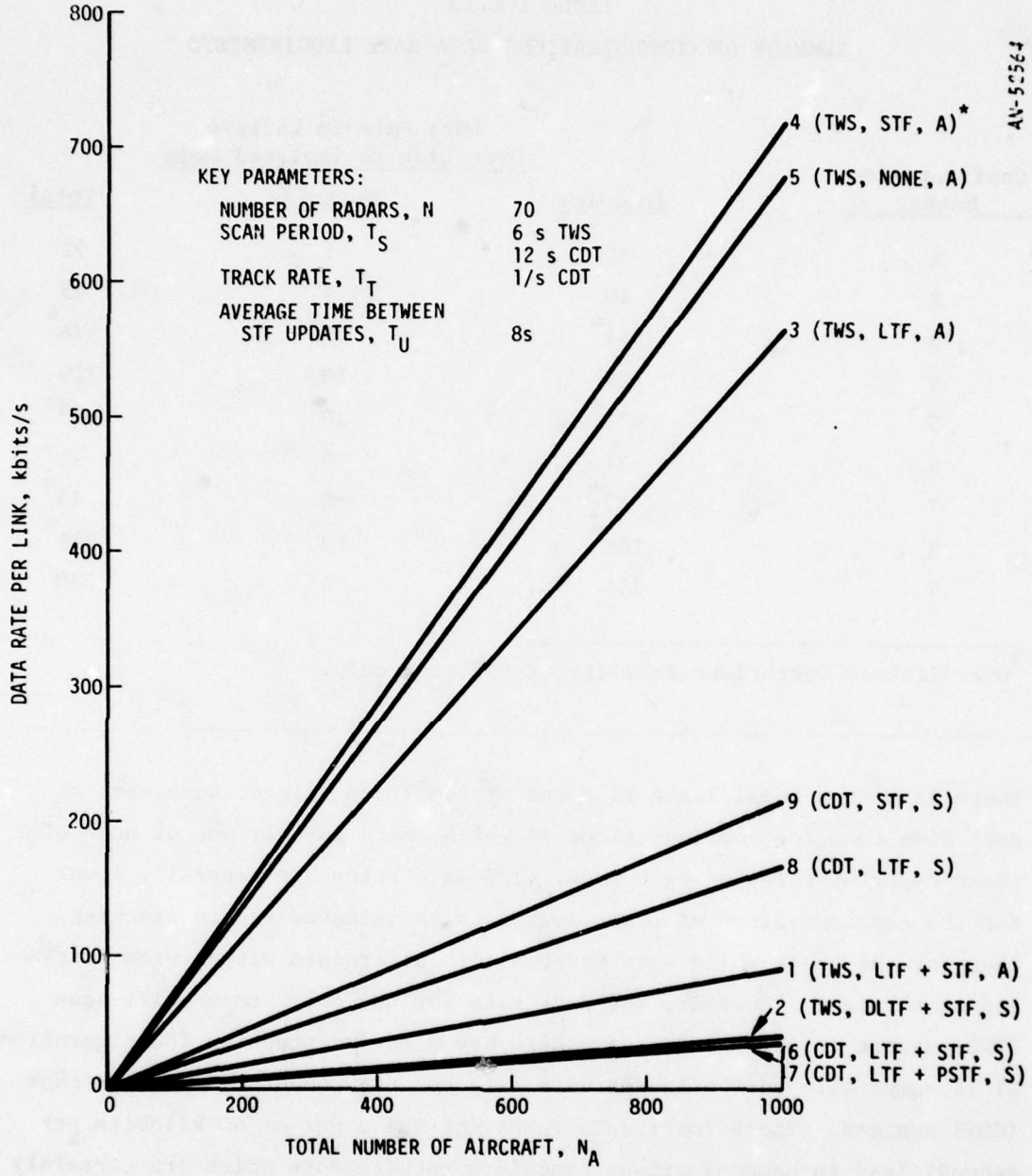
TABLE 1.4.13

SUMMARY OF COMMUNICATIONS DATA-RATE REQUIREMENTS

Configuration Number	Data Rate in kbits/s Over Link to Isolated Node		
	Incoming	Outgoing	Total
1	90	1	91
2	40	5	45
3	564*	12*	576*
4	719	10	729
5	678*	10*	688*
6	37	4	41
7	37*	6	43*
8	168*	83*	251*
9	216	4	220

*Over link to Operations Facility or STF node only.

there is both a Local Track File and System Track File of some sort at each node than for configurations in which there is only one or none of these types of files at each node. The data rates are generally lower for the computer-directed-track systems with selected radars tracking than for the track-while-scan systems with all radars within view tracking each target. However, the data rate for the only track-while-scan (TWS) system in which selected radars are used for tracking (Configuration 2) is comparable to the lowest data rate for the computer-directed-track (CDT) systems. These lower data rates (of the order of 40 kilobits per second) lead to communications bandwidth requirements which are certainly reasonable for line-of-sight communication links.



*CONFIGURATION NUMBER (RADAR MODE, TRACK FILES AT RADAR, ALL OR SELECTED TRACKERS).

Figure 1.4.4. Example of Communication Rate Versus Number of Aircraft One-Way Data Rate Into Isolated Node

1.5 DATA PROCESSING

The output data from the radar sensors must be suitably processed to produce the track files and displays of aircraft position that constitute the outputs of the tactical air surveillance system. In addition to the basic sequence of data-processing operations required for this purpose, the data processing involved in controlling the operation of the radars and the communications system must be performed. The data processing requirements depend on the system configuration and radar mode of operation as well as on the environmental and radar parameter values. To obtain an indication of the data processing requirements for the highly redundant netted systems under consideration, a estimate was made of the data-processing execution rates in millions of instructions per second (MIPS) for each of the nine system configurations defined in Sec. 1.3. These estimates are presented in this section. The resulting MIPS counts are for "traditional" types of instructions, such as performed by the Control Data 6000 series of machines.

The procedure followed in determining the data-processing rates consists of three basic steps:

1. The number of instructions required to perform each of the major data-processing operations was estimated.
2. For each system configuration of interest, the rates at which each operation is performed were specified as functions of the threat and system parameters.
3. The number of instructions was multiplied by the repetition rate for each operation, and the resulting values for all of the operations performed by a particular system were summed to obtain the data processing requirements for that configuration.

Estimates of the numbers of instructions for each operation are presented in Sec. 1.5.1, and Steps 2 and 3 are covered in Sec. 1.5.2, where expressions for the data-processing execution rates are presented and their values for a particular set of parameter values are given as examples.

1.5.1 Data Processing Instruction Counts[1]

The TACRAN3 simulation was used where possible as a point of departure in estimating the number of instructions involved in performing each of the data processing operations. For most of the major operations, parametric equations for the execution time were derived in terms of primitive operations and low-level data manipulations. The parameters involved are related to the threat environment, the system configuration, and the computational approach used. Timing experiments were performed on one computer system, a Control Data Corporation 7600, to obtain execution time estimates for the low-level operations. The results were converted to estimates of the number of instructions executed using known execution rates for standard benchmark problems.

The equations were then refined to compensate for some of the major differences in implementation between TACRAN3 and what might be required for deployable-grade software. It was assumed that computationally efficient algorithms and data structures would be employed for functions which might otherwise require excessive data processing resources. Finally the expressions were simplified by deleting terms and parameters which had negligible effects over the range of parameter values of interest.

For operations not performed in TACRAN3, other data and experience with other systems were used as bases for making an estimate of the number of instructions. For example, for radar scheduling, which is performed only in computer-directed-track systems, the results of some previous work on the development of scheduling algorithms for ballistic-missile-defense radars were used here even though the design and operations of the radars would be somewhat different.

The expressions or single-number estimates that were developed for the numbers of instructions for each of the major data-processing operations

[1] The instruction-count estimates in this section were prepared by Mr. P.T. Alexander, General Research Corporation, Santa Barbara.

are listed in Table 1.5.1. The parameters in the expressions are defined in Table 1.5.2, where the values used in the execution-rate calculations are specified. The threat and system parameters used in specifying the value of the parameter FLEN (the average number of entries in a file) were defined in Sec. 1.4.3 and their nominal values, which were used in the example calculations of data-processing rates as well as of the communications data rates in Sec. 1.4.4, are given in Table 1.4.3. The values of NM (the average number of tracks that could possibly associate with each measurement) and NP (the average number of track-measurement pairs simultaneously considered by the assignment algorithm) depend upon the distribution of aircraft in the surveillance volume. The specified values were selected somewhat arbitrarily to represent the situation where the aircraft are generally widely distributed but some of them are grouped more closely in some regions. The indicated dependence of NM and NP on the time between measurements or STF updates is a reflection of the growth of the association volume with time.

The parameter values in Table 1.5.2 were used in evaluating the expressions in Table 1.5.1 to obtain the instruction-count estimates listed there. In some cases, the estimated number of instructions required to perform a particular data-processing operation depends on the system configuration as indicated.

Note that the process of updating the System Track File using data from the Local Track File is not included in Table 1.5.1 (the System Track File Update operation that is listed refers to the process of fitting a quadratic function to five measurements when there are no Local Track Files in the system). It is assumed that this STF updating is done by simply inserting the current Local Track in place of the System Track, and that the data processing involved in this operation is negligibly small. If the System Track File were instead updated by smoothing in the data from the Local Track, e.g., by the use of Kalman-filter processing, then the data-processing requirements for this operation would be significant and shculd be included.

TABLE 1.5.1

ESTIMATED NUMBERS OF INSTRUCTIONS REQUIRED TO PERFORM THE MAJOR DATA-PROCESSING OPERATIONS

Operation	Number of Instructions
Coordinate Transformations	100
Track Initiation	
First Measurement	100
Second Measurement	120 + 130 min(FLEN,20)* + 500 NM = 3,700 | 6,900
Third Measurement	10 + 120 min(FLEN,20) + 1,900 NM = 3,100 |
Association With Local Track File	170 min(FLEN,20) + 300 NM + 500 (NP - 1) = 3,700 for CDT Systems and TWS System With DLTF = 4,500 for All Other TWS Systems
Association With System Track File	170 min(FLEN,20) + 300 NM + 500 (NP - 1) = 3,700 for Systems With No LTF = 4,500 for All Other Systems
Local or System Track File Update	240 + 130 NPTS = 900
Track File Maintenance	20 FLEN = 20,000 for STF = 7,000 for PSTF = 7,000 for LTF, All Radars Tracking = 800 for LTF, Selected Radars Tracking = 2,000 for TIF
Bookkeeping for New File Entry	150
Tracker Selection	900 for TWS System With DLTF 750 for All Others
Tracker Designation	750
Radar Scheduling	100 per Transmission
Message Transmission	150 + 120 NRCS = 150 for Unaddressed System Messages = 400 for Two-Address Messages = 300 for One-Address Messages
Message Reception	200

*Note: min(FLEN,20) ≡ minimum of FLEN and 20.

TABLE 1.5.2

PARAMETERS IN THE EXPRESSIONS FOR THE NUMBER OF DATA-PROCESSING INSTRUCTIONS

Symbol	Definition	Value Used
FLEN	Average Number of Entries in Track File	N_A for STF
		$N_A N_S/N$ for PSTF
		$N_A N_S/N$ for LTF, All Radars Tracking
		$N_A N_T/N$ for LTF, Selected Radars Tracking
		N_F for Second Track-Initiation Measurement
		10 for Third Track-Initiation Measurement
NM	Average Number of Tracks That Could Possibly Associate With Each Measurement	1 for 1-3 s Measurement Interval
		2 for 5-6 s Measurement Interval
		2 for 8-10 s Update Time
		2 for Second Track-Initiation Measurement
		1 for Third Track-Initiation Measurement
NP	Average Number of Track-Measurement Pairs Considered Simultaneously by the Assignment Algorithm	1 for 1-3 s Measurement Interval
		2 for 5-6 s Measurement Interval
		2 for 8-10 s Update Time
NPTS	Number of Measurements Smoothed by Filter	5
NRCS	Number of Nodes to Which a Message is Addressed	0 for System Messages
		2 or 3 for Directed Messages

As an example of the process involved in deriving the expressions for the numbers of instructions, consider the Association-With-Track-File operation. This operation is performed whenever an attempt is made to associate a new measurement with the Local or System Track File. For simplicity, it was assumed that the same algorithm is used in both cases across all systems; in practice, the algorithms would probably embody different criteria for association, and might thus exhibit some execution time differences.

The expression for the number of data-processing instructions has three terms. The first term represents the comparison of the new measurement with the existing file of known tracks. It is assumed that the file is kept ordered so that comparison with, at most, 20 tracks is required [thus the term min(FLEN,20) in Table 1.5.1]. The computational burden of maintaining ordered lists is included in the Bookkeeping for New File Entry. The second term represents the processing required when a measurement passes the coarse association tests for a particular track. This processing includes transferring of data and some bookkeeping operations. The third term represents the processing required to resolve multiple potential associations and select a single track measurement pairing. It was assumed that the mathematical procedure known as the assignment algorithm would be used to resolve multiple associations. The exact implementation of the algorithm would probably be dependent upon such parameters as the expected and maximum number of multiple associations.

1.5.2 Data Processing Execution Rates

The execution rate in instructions per second to perform the required data-processing operations is proportional to the rate (repetitions per second) at which each operation must be performed. This repetition rate depends on the threat and system parameters and is generally different for each function.

For each of the nine system configurations described in Sec. 1.3.2, the major data-processing operations were identified and their repetition

rates were specified in terms of the pertinent parameters. These parameters include some that were defined in Sec. 1.4.3 and listed in Table 1.4.3 (page 71) in connection with the communications analysis. The following additional parameters, listed in Table 1.5.3, were used in the data-processing analysis.

New-Target Rate. The rate at which new targets enter the surveillance volume of each radar and of the entire system, represented respectively by S_R and S_S, depends on the distribution of the aircraft and on their velocities. If 1,000 aircraft are assumed to be distributed uniformly over a 270×270 km area and to be moving in the same direction at a speed of 0.5 km/s, new targets enter the coverage of each radar with an 80 km-search radius at a rate of about 1.2 per second, and enter the coverage of the entire system at a rate of about 1.8 per second. These values were rounded to 1 and 2 new targets per second, respectively, as indicated in Table 1.5.3.

TABLE 1.5.3

NEW PARAMETERS INVOLVED IN THE DATA PROCESSING REQUIREMENTS ANALYSIS

(Other Parameters are Listed in Table 1.4.3 on page 71)

Symbol	Parameter	Value for Example
S_R	Rate at Which New Targets Enter the Surveillance Coverage of Each Radar	1 per s
S_S	Rate at Which New Targets Enter the Surveillance Coverage of the System	2 per s
T_{MI}	Maintenance Interval for Track Initiation File	3 s
T_{ML}	Maintenance Interval for Local Track File	10 s
T_{MS}	Maintenance Interval for System Track File	30 s
R_T	Transmission Repetition Frequency	1,000 per s

Track-File Maintenance Intervals. Track-file maintenance should be performed at appropriate intervals to purge old and redundant tracks from the files. The time between track-file-maintenance executions should depend on the rate at which the contents of the file are expected to change. The example values of this interval specified in Table 1.5.3 were selected as reasonable but arbitrary values. As will be seen, they have very little effect on the data processing requirements.

Transmission Repetition Frequency. The rate at which radar pulses or groups of pulses are transmitted for surveillance or tracking purposes affects the data processing requirements for radar scheduling in the directed-track systems. Typically, it might be necessary to schedule about 830 surveillance transmissions per second (10,000 beam positions with a 12-second scan time) plus another 50 transmissions per second for track initiation and tracking; the total number of transmissions per second was conservatively rounded up to 1,000 per second for the example calculations as indicated in Table 1.5.3.

The major data processing operations, along with the numbers of instructions and the repetition-rate expressions for each operation, are listed in Tables 1.5.4 through 1.5.12 for each of the system configurations under consideration. The numbers of instructions are taken from Table 1.5.1. The repetition-rate expressions generally involve the average number of aircraft and false alarms for which the operation must be performed and the average time between repetitions of the operation. In the system configurations in which there is a System Track File at every node, all of the listed operations must be performed at every node. In the system configurations in which the STF is maintained at only a few nodes, which could be the Operations Facilities located at some of the radar/data processing nodes, some of the operations are performed only at these STF nodes and some, namely the communications-related operations, are performed at a higher rate at the STF nodes. The data-processing load would be lower at the other (non-STF) nodes, but if every

node is to be capable of serving as the STF node, they must all have the required data-processing capacity.

The repetition-rate expressions are independent of the location of the node in the network, except for the Message Transmission and Message Reception rates which are specified for nodes that are interconnected to several other nodes to which they relay some of the incoming messages; in fact, it is assumed that each incoming system message is retransmitted by each node and that each directed message with one address is retransmitted once before it reaches its destination and that each two-address message is retransmitted twice.

The expression for the data-processing execution rate in instructions per second for each operation is just the expression for the number of repetitions per second multiplied by the number of instructions per repetition. Values of the execution rates in thousands of instructions per second (TIPS) are given for each of the operations in Tables 1.5.4 through 1.5.12 for the example parameter values specified in Table 1.4.3 and in Table 1.5.3. These values are summed to obtain the average data-processing execution rates. The subtotals are increased by 50% to account for control overhead. The totals are expressed in millions of instructions per second (MIPS) required to perform all of the operations in each of the systems. The instruction counts and the expressions for the repetition rates can be used to determine the data processing requirements for other sets of parameter values that might be of interest (in some cases the numbers of instructions for some operations also depend to some extent on certain parameters.)

TABLE 1.5.4

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 1

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	7.6
Track Initiation	6,900	S_R+N_F/T_S	121.9
Association With LTF	4,500	$(N_A N_S/N+N_F)/T_S$	342.9
Local Track File Update	900	$N_A N_S/NT_S$	53.6
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
Local Track File	7,000	$1/T_{ML}$	0.7
System Track File	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	2.7
Local Track File	150	S_R	0.2
System Track File	150	S_S	0.3
Message Transmission	150	$F_U N_A N_S/T_U$	78.1
Message Reception	200	$F_U N_A N_S(N-1)/NT_U$	102.7
Overhead (50%)		Subtotal	712.1 TIPS
			356.0
		Total	1.1 MIPS

F_U = Fraction of Radars that Update the STF = 1/6 in Example.

TABLE 1.5.5

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 2

(TWS Operation--DLTF and STF at Each Radar--Selected Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	7.6
Track Initiation	6,900	S_R+N_F/T_S	121.9
Association With LTF	3,700	$(N_A N_S/N+N_F)/T_S$	281.9
Association With STF	4,500	$N_A(N_S-N_T)/NT_S+N_F/T_S$	310.7
Local Track File Update	900	$N_A N_T/NT_S$	6.4
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
Local Track File	800	$1/T_{ML}$	0.1
System Track File	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	2.7
Local Track File	150	S_R	0.2
System Track File	150	S_S	0.3
Tracker Selection	900	$N_A(N_S-N_T)/NT_S$	47.1
Message Transmission			
System Messages	150	$N_A(1/T_U+2/T_C+1/T_H)$	26.3
2-Address Messages	400	$3N_A N_T/NT_S$	8.6
1-Address Messages	300	$4N_A/NT_C+4FN_A(N_S-N_T)/NT_S$	31.8
Message Reception	200	Sum of Message Transmission Rates	60.6
Overhead (50%)		Subtotal	907.6 TIPS
			453.8
		Total	1.4 MIPS

TABLE 1.5.6

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 3

(TWS Operation--LTF Only at Each Radar--All Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	7.6
Track Initiation	6,900	S_R+N_F/T_S	121.9
Association With LTF	4,500	$(N_A N_S/N+N_F)/T_S$	342.9
Association With STF*	4,500	NS_R	315.0
Local Track File Update	900	$N_A N_S/NT_S$	53.6
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
Local Track File	7,000	$1/T_{ML}$	0.7
System Track File*	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	2.7
Local Track File	150	S_R	0.2
System Track File*	150	S_S	0.3
Message Transmission†			
1-Address Messages	300	$N_A(N-N_O)/N_O NT_L$	14.8
2-Address Messages	400	$N_A N_S/NT_U$	17.9
Message Reception†	200	$N_A N_S(N-1)/NT_U+N_A/NT_L$	616.8
		Subtotal	1,495.8 TIPS
Overhead (50%)			747.9
		Total	2.2 MIPS

*At Operations Facilities (STF nodes) only.

†Values for Operations Facilities (STF nodes).

TABLE 1.5.7

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 4

(TWS Operation--STF Only at Each Radar--All Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	7.6
Track Initiation	6,900	S_R+N_F/T_S	121.9
Association With STF	3,700	$(N_A N_S/N+N_F)/T_S$	281.9
System Track File Update	900	$N_A N_S/NT_S$	53.6
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
System Track File	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	2.7
System Track File	150	S_S	0.3
Message Transmission	150	$N_A N_S/T_S$	625.0
Message Reception	200	$N_A N_S(N-1)/NT_S$	821.4
Overhead (50%)		Subtotal	1,915.8 TIPS
		or	957.9
		Total	2.9 MIPS

TABLE 1.5.8

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 5

(TWS Operation--No Track Files at the Radars--All Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	7.6
Track Initiation*	6,900	S_R+N_F/T_S	121.9
Association With STF*	3,700	$(N_A N_S/N+N_F)/T_S$	281.9
System Track File Update*	900	$N_A N_S/NT_S$	53.6
Track File Maintenance*			
Track Initiation File	2,000	$1/T_{MI}$	0.7
System Track File	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry*			
Track Initiation File	150	S_R+N_F/T_S	2.7
System Track File	150	S_S	0.3
Message Transmission†	400	$(N_A N_S/N+N_F)/T_S$	38.1
Message Reception†	200	$(N-1)(N_A N_S/N+N_F)/T_S$	1,051.4
Overhead (50%)		Subtotal	1,558.9 TIPS
			779.5
		Total	2.3 MIPS

*At Operations Facilities (STF nodes) only.

†Values for Operations Facilities (STF nodes).

TABLE 1.5.9

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 6

(CDT Operation--LTF and STF at Each Radar--Selected Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	3.8
Track Initiation	6,900	S_R+N_F/T_S	64.4
Association With LTF	3,700	$(N_A N_S/N+N_F)/T_S$	141.0
Association With STF	4,500	$N_A(N_S-N_T)/NT_S+N_F/T_S$	155.4
Local Track File Update	900	$N_A N_T/NT_T$	38.6
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
Local Track File	800	$1/T_{ML}$	0.1
System Track File	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	1.4
Local Track File	150	S_R	0.2
System Track File	150	S_S	0.3
Tracker Selection	750	$N_A(N_S-N_T)/NT_S$	19.6
Tracker Designation	750	$N_A N_T/NT_T$	32.1
Radar Scheduling	100	R_T	100.0
Message Transmission			
System Messages	150	$N_A(1/T_U+2/T_C+1/T_P+1/T_H)$	29.3
2-Address Messages	400	$3(N_A N_T/NT_R+1/T_A)$	18.3
1-Address Messages	300	$4FN_A(N_S-N_T)/NT_S$	15.7
Message Reception	200	Sum of Message Transmission Rates	58.6
Overhead (50%)			
		Subtotal	680.2 TIPS
			340.1
		Total	1.0 MIPS

TABLE 1.5.10

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 7

(CDT Operation--LTF and PSTF at Each Radar--Selected Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	3.8
Track Initiation	6,900	S_R+N_F/T_S	64.4
Association With LTF	3,700	$(N_A N_S/N+N_F)/T_S$	141.0
Association With PSTF	4,500	$N_A(N_S-N_T)/NT_S+N_F/T_S$	155.4
Association With STF*	4,500	$(F_M N_A N_S/N+N_F)/T_S$	50.9
Local Track File Update	900	$N_A N_T/NT_T$	38.6
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
Local Track File	800	$1/T_{ML}$	0.1
Partial System Track File	7,000	$1/T_{MS}$	0.2
System Track File*	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	1.4
Local Track File	150	S_R	0.2
Partial System Track File	150	S_S	0.3
System Track File*	150	S_S	0.3
Tracker Selection	750	$N_A(N_S-N_T)/NT_S$	19.6
Tracker Designation	750	$N_A N_T/NT_T$	32.1
Radar Scheduling	100	R_T	100.0
Message Transmission†			
Partial System Messages§	600	$3N_A N_S/N(1/T_U+2/T_C+1/T_P+1/T_H)$	125.4
2-Address Messages	400	$3(N_A N_T/NT_R+1/T_A)$	18.3
1-Address Messages	300	$4(F_M N_A N_S/N+N_F)/T_S+4FN_A(N_S-N_T)/NT_S$	29.3
Message Reception†	200	N_A/T_U + Sum of Message Transmission Rates	95.5
Overhead (50%)			
		Subtotal	878.2 TIPS
			439.1
		Total	1.3 MIPS

*At Operations Facilities (STF nodes) only.

†Values for Operations Facilities (STF nodes).

§Assumes that each Partial System message has an average of four addresses and is retransmitted twice.

F_M = Fraction of measurements that must be sent to another node for association = 0.1 in example

TABLE 1.5.11

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 8

(CDT Operation--LTF Only at Each Radar--Selected Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	3.8
Track Initiation	6,900	S_R+N_F/T_S	64.4
Association With LTF	3,700	$(N_A N_S/N+N_F)/T_S$	141.0
Association With STF*	4,500	$N_A(N_S-N_T)/NT_S+N_F/T_S$	155.4
Local Track File Update	900	$N_A N_T/NT_T$	38.6
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
Local Track File	800	$1/T_{ML}$	0.1
System Track File*	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	1.4
Local Track File	150	S_R	0.2
System Track File*	150	S_S	0.3
Tracker Selection	750	$N_A(N_S-N_T)/NT_S$	19.6
Tracker Designation	750	$N_A N_T/NT_T$	32.1
Radar Scheduling	100	R_T	100.0
Message Transmission†			
2-Address Messages	400	$(N_A/N)(1/T_U+2/T_C+1/T_P$ $+1/T_H)+3(N_A N_T/NT_R+1/T_A)$	19.5
1-Address Messages	300	$(N/N_O)[N_A(N_S-N_T)/N+N_F)]/T_S$ $+FN_A(N/N_O)(N_S-N_T)/NT_S$	333.3
Message Reception†	200	$N_A(1/T_U+2/T_C+1/T_P+1/T_H)$ + Sum of Message Transmission Rates	271.0
Overhead (50%)		Subtotal	1,182.1 TIPS 591.0
		Total	1.8 MIPS

*At Operations Facilities (STF nodes) only.

†Values for Operations Facilities (STF nodes).

TABLE 1.5.12

DATA PROCESSING REQUIREMENTS FOR CONFIGURATION 9

(CDT Operation--STF Only at Each Radar--Selected Radars Tracking)

Operation	Number of Instructions	Repetitions per Second	Execution Rate, TIPS Example Values
Coordinate Transformation	100	$(N_A N_S/N+N_F)/T_S$	3.8
Track Initiation	6,900	S_R+N_F/T_S	64.4
Association With STF	3,700	$(N_A N_S/N+N_F)/T_S$	141.0
System Track File Update	900	N_A/NT_T	12.9
Track File Maintenance			
Track Initiation File	2,000	$1/T_{MI}$	0.7
System Track File	20,000	$1/T_{MS}$	0.7
Bookkeeping for New File Entry			
Track Initiation File	150	S_R+N_F/T_S	1.4
System Track File	150	S_S	0.3
Tracker Selection	750	$N_A(N_S-N_T)/NT_S$	19.6
Tracker Designation	750	$N_A N_T/NT_T$	32.1
Radar Scheduling	100	R_T	100.0
Message Transmission			
System Messages	150	$N_A(1/T_T+2/T_C+1/T_P+1/T_H)$	160.5
2-Address Messages	400	$3(N_A N_T/NT_R+1/T_A)$	18.3
Message Reception	200	Sum of Message Transmission Rates	223.2
Overhead (50%)		Subtotal	778.9 TIPS 389.5
		Total	1.2 MIPS

The relative magnitudes of the data processing requirements for the various operations are readily apparent in the tables. The execution rates for the Association operations are large in all cases, and those for Track Initiation and for Radar Scheduling (in the Direct-Track Systems where it is needed) are also sizable. The data-processing requirements for transmitting and receiving communications messages become quite large in some systems configurations, at least as they are estimated here: possibly if these configurations were to be built, special-purpose hardware and/or more efficient algorithms could be developed to reduce these requirements. The data processing loads imposed by background functions such as Track File Maintenance and Bookkeeping for New File Entries are small in all cases.

The data-processing execution rates for the example parameter values are summarized in Table 1.5.13 for the nine system configurations. The data-processing requirements for all of the configurations and modes of operation are of the order of 1-3 MIPS. The lower values are obtained when there is both a Local Track File and a System Track File at each node (Configurations 1, 2, 6, and 7), with the higher values required when there is only one track file or none at each node (Configurations 3, 4, 5, and 8) with one exception (Configuration 9).

The example values of the data-processing execution rates are plotted in Fig. 1.5.1 versus the total number of aircraft in track by the system. Because of the technique for handling association, these plots are straight lines or nearly straight lines with different slopes for the different system configurations.

It should be emphasized that the results of this data processing analysis are based on a series of assumptions and estimates, many of which are difficult to substantiate until the algorithms involved have been defined in detail and a method of implementation has been developed. Furthermore, the execution rates that were obtained are average values;

TABLE 1.5.13

SUMMARY OF DATA PROCESSING REQUIREMENTS

(Including an Additional 50% Instructions for Overhead Functions)

Configuration Number	Execution Rate in MIPS
1	1.1
2	1.4
3	2.2*
4	2.9
5	2.3*
6	1.0
7	1.3*
8	1.8*
9	1.2

*At Operations Facilities or STF nodes only.

under conditions where certain radars must track an unduly large share of the targets, or where the targets are bunched together making association more difficult, the peak data processing loads may be considerably higher. Also the 50% additional instructions made for the data processing overhead association with the real-time operating system is an estimate that could vary considerably depending on the implementation. Nevertheless, the results do indicate that the data processing requirements for a netted air surveillance and control system are in a reasonable and interesting range--they are not so small as to be trivially attainable nor so large as to be practically unattainable, particularly in view of recent and projected advances in the state-of-the-art which provide increasing data processing capabilities with reduced hardware size and cost.

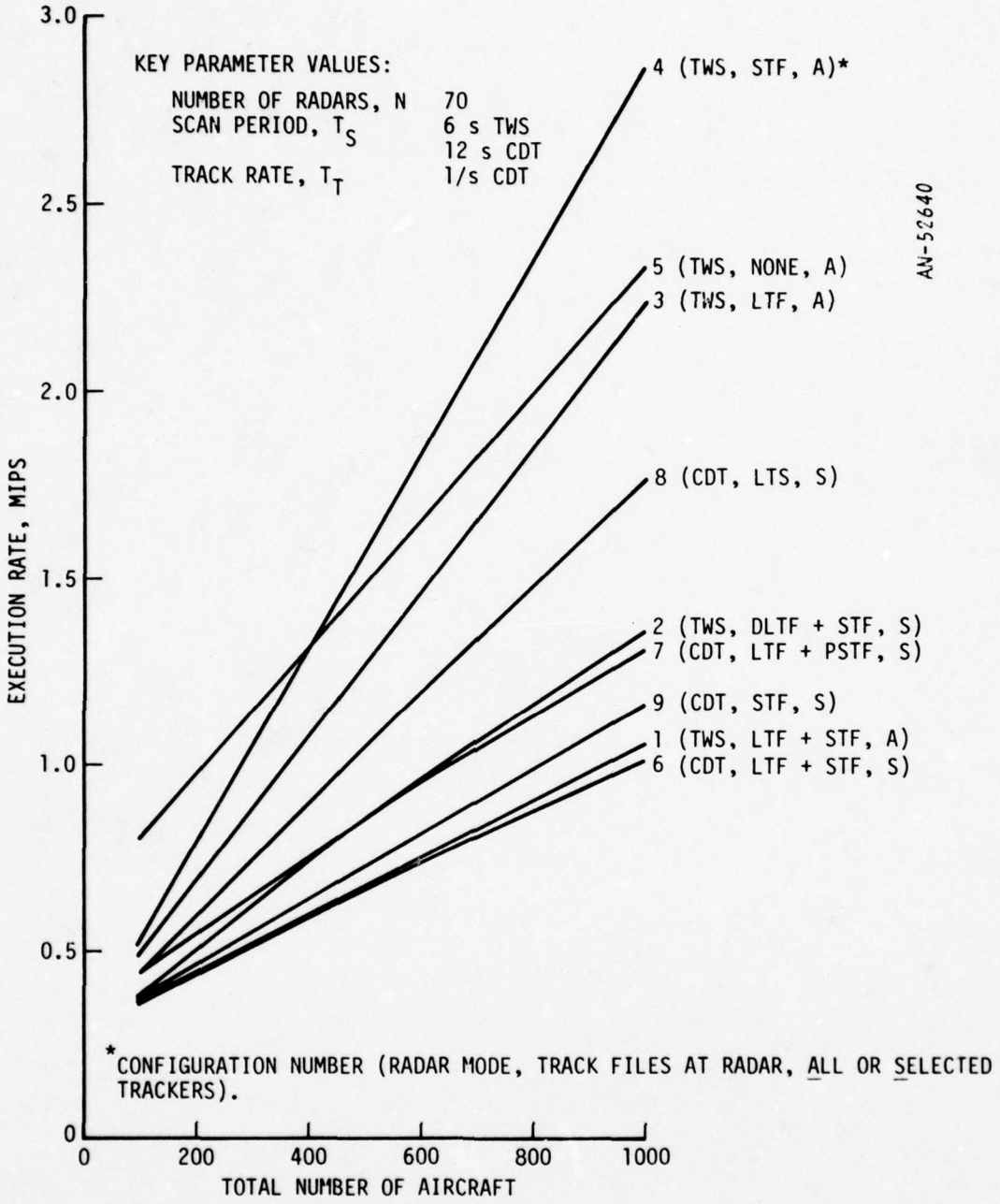


Figure 1.5.1. Example of Data Processing Requirements Versus Number of Aircraft

2 REAL-TIME DATA PROCESSING FUNCTIONS

The primary data-processing functions that were investigated in detail during this study are track initiation, track association (or correlation), and track filtering. Successful execution of these primary functions is critical to the operation of automated netted air surveillance systems of the type under consideration. The capabilities of specific algorithms for performing these functions under the conditions of interest are analyzed in this section.

There are, however, a number of other functions that must also be performed as part of the operation of a network of automated radars. These are described next.

2.1 LIST OF REAL-TIME FUNCTIONS

The functions to be performed fall generally into four classes: (1) those local functions required by a track-while-scan system, (2) those additional local functions that are required by a phased array with full beam agility, i.e., a computer-directed-track system, (3) the functions required to maintain the system-level tracks in the System Track File, and (4) the functions required for handling communications.

The following descriptions distinguish between two types of track: The first is the Local Track; these are the individual tracks (if they exist) at each radar that are directly updated by the radar returns from that radar. The second is the System Track; these are the tracks replicated at several or all nodes and are the primary output of the surveillance system.

2.1.1 Local Track-While-Scan Functions

The local functions required by a track-while-scan (TWS) system are summarized below:

PRECEDING PAGE BLANK-NOT FILMED

Track Initiation. Radar returns (measurements) that do not associate with existing tracks in the Local Track File are potential candidates for track initiation. (Of course they may also be noise pulses or clutter returns.) Successive returns are saved, and association with the previous return(s) is attempted. Depending on the circumstances, a successful track may be initiated with only two returns; however, in some cases additional returns may sometimes be required to ensure a high probability of successful initiation with a low probability of false initiations. This function is further described and analyzed in Sec. 2.2.

Track Association. Track association (correlation) is the process of associating a new radar return with an existing track file. It involves extrapolating each track in the file forward in time to the time of the radar return (or a time nearby depending on the track accuracies involved), and then determining if the radar return occupies the same space (in up to as many dimensions as exists in the return) as the extrapolated target. Association is considerably aided by the presence of an identification tag with the radar return. This function is further described and analyzed in Sec. 2.3.

Track Update. Updating is the process of computing a new estimate of the track of an aircraft based on information from a new radar return (measurement). As described in Sec. 2.4, there are recursive and nonrecursive filters for performing track update.

Track File Maintenance. A number of important tracking functions are listed under the general heading of track file maintenance. These include deleting (or combining) multiple tracks on the same aircraft and purging old tracks. Old tracks may arise because the aircraft has left the surveillance volume or because the track resulted from false alarms, not a real aircraft.

Clutter Mapping. Some air surveillance radars may require additional processing of returns from clutter over that provided by the signal processor. However, it is reasonable to expect that future

radars will eliminate clutter to the maximum extent possible in the signal processor before such returns reach the data processor by using techniques such as those employed in the MIT Lincoln Laboratory MTD (moving target detector) processor.

Other Functions. These include manual intervention modes, local jammer response, and IFF data processing.

2.1.2 Local Computer-Directed-Track Functions

In a computer-directed-track (CDT) system track is not performed using search pulses as in a track-while-scan system, but rather by directing a pulse toward the expected position of a target being tracked whenever another data point is required. Search is performed in addition to track as a background function.

A CDT system requires all of the functions of a TWS system and more. Track initiation in a CDT system, however, is much more versatile than in a TWS system because the second return can follow the initiation of an initial detection closely enough in time such that association of the second (or additional returns, if needed) is relatively unquestioned. Association of track returns is not required, because each track return is tagged with the identification of the target, as explained below. Association is still required for returns obtained during search.

A diagram of the flow of data through the track loop of a CDT system is shown in Fig. 2.1.1. Only the most basic of the many required functions are shown. The basic track loop consists of the unshaded boxes in the figure. Track operation is as follows: Assume an aircraft is already in track; i.e., a track has been established on the aircraft. It is up to the data processor to schedule a new track measurement on the aircraft. This is accomplished by having a track-request routine note when the next measurement is required. The Radar Scheduler routine takes all of the track requests (and scan requests) and schedules them. The result is a set of Radar Orders which drive the radar. The approximate

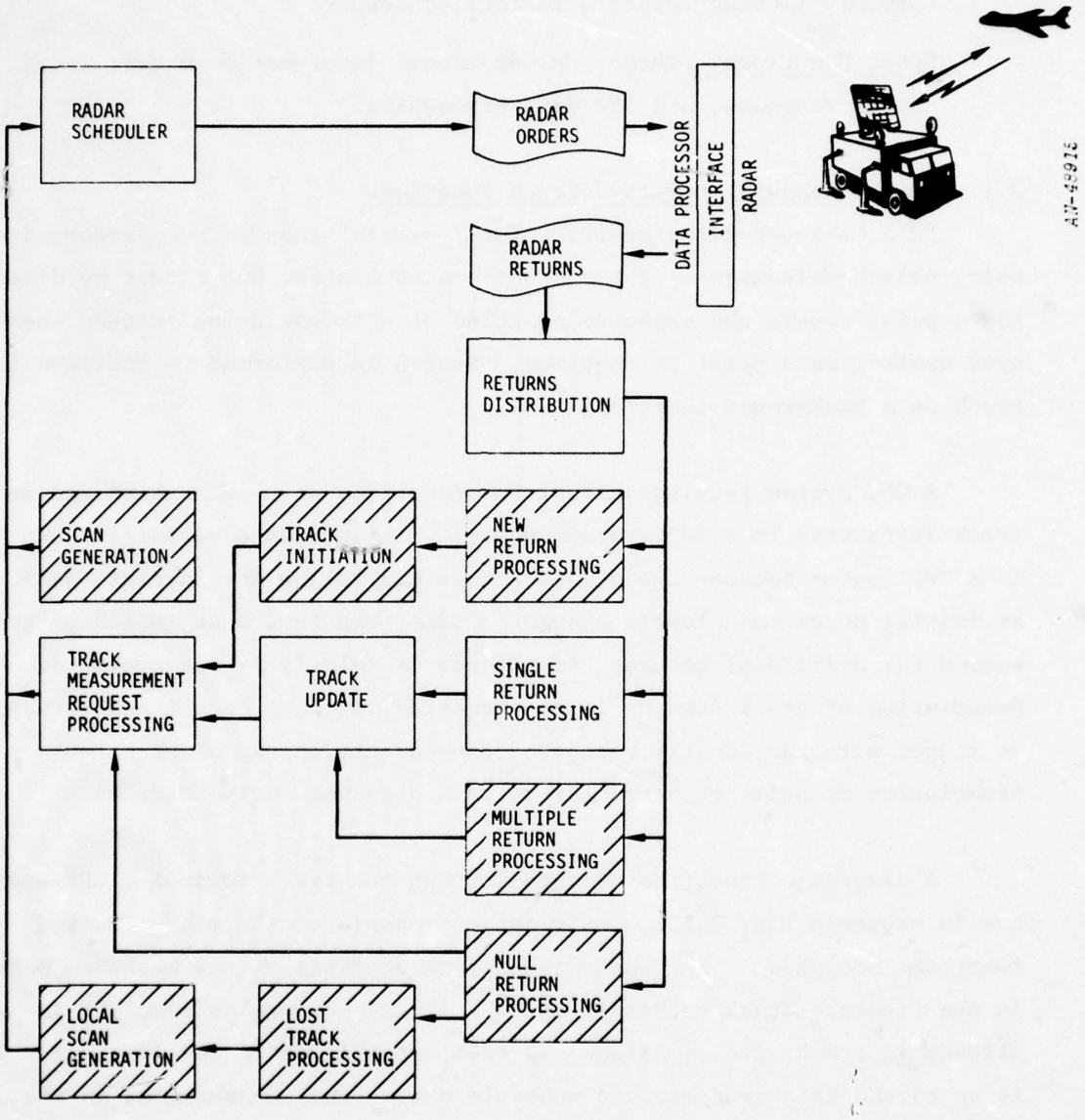


Figure 2.1.1. Track Loop Diagram in a Computer-Directed-Track System

location of the return from a requested track measurement is known, so that when the echo returns, it can immediately be tagged with an identification of the requesting track file, making association unnecessary. The return data is distributed to the track update routine, after which a new track measurement is requested.

Complications of the above simple track loop occur because (1) search is occurring at the same time as track, and (2) the expected track return may not be received or may consist of several returns. Some of the principal other required functions are shown in the shaded boxes in the figure. Scan Generation makes requests for search transmission; these requests enter the queue for servicing by the Radar Scheduler along with the track requests. The search returns will include new detections, noise, clutter, and all aircraft already in track. The aircraft already in track can be associated with the existing tracks, and the measurement can either be discarded or used to update the track. Other returns must be processed as potentially new aircraft. A second measurement can be requested shortly thereafter to complete the track initiation (this loop is not shown in the figure for simplicity).

The additional functions (over a TWS system) required by a CDT system are summarized below:

Radar Scheduling. Requests for radar service from several sources (including scan and track) are scheduled onto the radar time line and turned into specific radar orders.

Track Measurement Request Processing. This function generates requests for track measurements based on the time that each track file should be updated.

Scan Generation. Requests for search transmissions are generated here, perhaps based on a fixed template of scan beams, perhaps in a more dynamically adaptive manner.

Null Return Processing. This function handles missed detections. Either another track measurement is requested, or additional Lost Track Processing is performed.

Lost Track Processing. If too many missed detections occur in a row, the track is considered lost and either a new track established by reacquisition with a local search about the last predicted position, or the track is simply dropped.

Multiple Return Processing. If more than one track return is received when only one was expected, processing is required to determine which is the correct return. An extra return might be from a noise pulse, unexpected clutter, crossing targets, or countermeasures.

2.1.3 System Track Functions

An additional set of functions is required to maintain the System Track File. These functions include:

System Track Initiation. Whenever a new track enters the system at any point, a new System Track must be created. At the system level this is either a simple or a complex process, depending on the system configuration.

System Track Association. This function involves associating local track data with the existing System Tracks. Techniques for accomplishing this depend on the specific system configuration being considered. (See Sec. 2.3.1 for additional discussion.)

System Track Update. The System Tracks may need to be kept up to date at least at some minimum rate defined by the users, even if no new radar measurements are available. This is performed by extrapolating the last known state of the aircraft forward to the present time.

System Track File Maintenance. The purpose of this function is to keep the System Track File "clean." That is, it deletes (or

combines) multiple tracks, and it purges old tracks that have not been updated for some specified duration.

Target Handover. When an aircraft that is being handled by one radar enters the surveillance volume of another radar, the second radar needs to initiate a track on the aircraft. This can be accomplished in several ways: (1) the original tracking radar can send a message to the second radar, (2) the second radar can wait for its first detection on the aircraft and then associate the detection with the System Track File, or (3) a routine constantly checking the System Track File can start a Local Track File at the second radar, automatically permitting the second radar to begin a track on the aircraft.

Jammer Response Coordination. An additional function that could be performed at the system level is the development of a coordinated system response to jamming.

Automated Site Registration. The registration of a site can be determined by comparing its local measurements or tracks with the corresponding System Tracks.

2.1.4 Communications Functions

A number of functions are associated with the preparation and sending of messages, and with their receiving and processing. Many of these functions are specialized to a particular system design; an example of this is given in Sec. 3.5.

Two functions are common to any of the systems. These are:

1. Message Transmission. This function accepts messages from various routines and prepares them for sending. It is the data processing interface with the sending communications system.

2. Message Reception. This function receives messages from the receiving communication system, perhaps modifies the message, and gives it to the appropriate routine depending on message type.

2.2 TRACK INITIATION

Track initiation is the process in which a sequence of returns from newly observed targets are stored and associated (correlated) so as to obtain sufficient information about the target to establish a track. All new radar returns--those that do not associate with an existing track in the Local or System Track Files--are candidates for track initiation. Some of these returns will be false alarms due to noise and clutter. One of the functions of the track-initiation process is to eliminate these false alarms before tracks are established on them.

Several different track-initiation algorithms were considered; they are described in Sec. 2.2.1. The probability of initiating track on a sequence of random false alarms is analyzed in Sec. 2.2.2, and the expected numbers of false-alarm track initiations per scan are determined for each of the algorithms. False tracks can also be initiated through incorrect pairings of returns from nearby targets, but the frequency of such false-track initiations, and the persistence of the tracks that are initiated, depend on the particular target geometries and maneuver histories, and the multiple-target track-initiation problem was not analyzed as part of this study.

2.2.1 Track-Initiation Algorithms

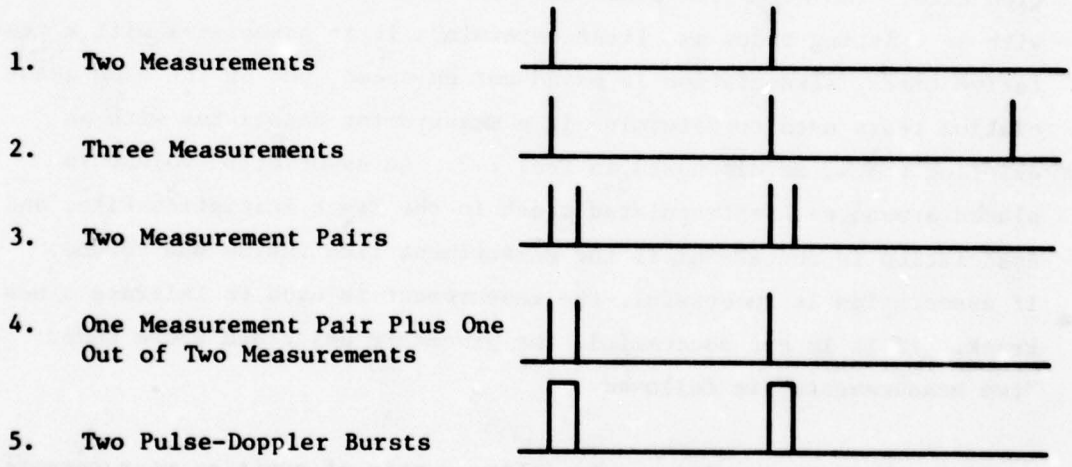
The track-initiation process involves the association of measurements of target position and possibly velocity to determine whether or not the returns are from the same target and if so to obtain an estimate of the target position and velocity for track-filter initialization. Track initiation can be carried out using different types of measurements, numbers of measurements, and measurement spacings, with different levels

of performance associated with each of the variations. The track-initiation algorithms considered in the analysis of false-alarm effects are listed in Table 2.2.1 with a sketch indicating the relative timing of the measurements in each case. The first four algorithms all utilize position measurements only; the fifth utilizes doppler measurement of radial velocity. Each of these algorithms and its relationship to the radar mode of operation can be described briefly as follows.

Two Measurements. The simplest track-initiation algorithm associates two measurements[1] of target position, made, for example, on successive scans of a TWS radar. The algorithm takes each measurement that does not associate with an existing track and first determines if it

TABLE 2.2.1

TRACK INITIATION ALGORITHMS CONSIDERED



[1] Each measurement may involve a number of radar pulses which are transmitted and received during the time the target is within the antenna beam. In the AN/TPS-43 radar, for example, about 8 pulses are transmitted during the time the antenna beam moves one beamwidth. The returns from these pulses are integrated by the radar signal processor to produce a single measurement.

associates with a previously saved measurement. The association test is simply based on speed; if two measurements produce a track with a credible speed (or velocity vector)--i.e., a speed that is not greater than the maximum speed of any aircraft of interest at the target altitude--a track is initiated with the second measurement as the estimated position and a velocity based on the two position measurements and the time between them. Tracks are initiated on all pairs of pulses that pass the speed test. Whether or not a measurement is used to initiate a track, it is saved for possible association with later measurements; this ensures that measurements on valid targets are not lost during the track initiation process by an accidental association with a false alarm.

Three Measurements. Improved performance can be obtained by using three measurements on successive scans or possibly three more closely spaced measurements in a computer-directed-track system. The first two measurements are not used to set up a tentative track in a Track Initiation File. The algorithm takes each measurement that does not associate with an existing track and first determines if it associates with a tentative track. Association is based not on speed, but on the same association tests used to determine if a measurement associates with an existing track, as discussed in Sec. 2.3. An association volume is placed around each extrapolated track in the Track Initiation File, and association is successful if the measurement lies inside the volume. If association is successful, the measurement is used to initiate a new track. If it is not successful, the procedure described above under "two measurements" is followed.

Two Measurement Pairs. By making a pair of position measurements a relatively short time apart (of the order of one second), an estimate of the target velocity can be made and used in determining the association volume for measurements a scan time later. (A crude track can be immediately initiated with the first measurement pair, if required.) With a mechanically scanned reflector, a second measurement (following

the one made by the surveillance beam) might be obtained by using an offset feed arrangement to produce a second beam displaced in azimuth, but the use of a mechanically scanned phased array to form a second, off-boresight beam is probably a more practical approach.

One Measurement Pair Plus One-Out-of-Two Measurements. If track initiation requires the detection of four sets of measurements as in the previous algorithm, the probability of successful initiation will be low unless the probability of detection for each measurement is high as is discussed in the next section. This algorithm constitutes an attempt to alleviate this problem by requiring that only one of the two sets of measurements in the second pair be detected (and associate with the first pair).

Two Pulse-Doppler Bursts. By using coherent pulse bursts, the radial component of target velocity can be measured. This measurement can be used in predicting the target position at the time of the second measurement, and the radial velocity constitutes another dimension of the association volume which can be utilized to improve the track initiation performance. It should be pointed out, however, that the design of a suitable pulse-doppler waveform and signal processor is not simple for the application of interest because of the wide range of possible target velocities.

2.2.2 False-Alarm Track Initiations

In any radar, false alarms due to noise will occur on occasion. For purposes of the analysis here, the rate at which they occur will be specified in terms of an average number per scan, as given in the Statement of Work. Track initiation will be attempted on each false alarm which does not associate with an existing track (i.e., virtually all the false alarms). In most cases, there will be no false alarm or radar return within the association volume at subsequent measurement times, and the track-initiation process will be terminated. It is possible, however, that false alarms will occur in the association volume and that tracks

will be initiated on a sequence of false alarms. The frequency with which such false-alarm initiations could be expected to occur is of interest--if it is high it could overload the data processing and communications systems and lead to constantly changing and confusing presentations of the tactical air situation--so the dependence of the expected number of false-alarm track initiations on the pertinent radar and target parameters was analyzed for the algorithms defined in the preceding section.

The probability that a random false alarm occurs within an association cell depends on the size of the cell relative to the total volume within which the false alarms can occur (generally the surveillance volume). The size of the association cell depends on the target velocity and acceleration capabilities, the time between measurements (i.e., the time during which target displacements can build up), and the radar measurement accuracy.

The horizontal and vertical dimensions of the association cell may differ in magnitude because of different target velocity and acceleration capabilities and different radar measurement accuracies in these two dimensions. As a simple representation of the association cell geometry which easily accommodates this difference in dimensions (but assumes a symmetrical distribution of target locations in the horizontal plane), a right circular cylinder was used in the false-initiation calculations for the first four algorithms under consideration (those not involving Doppler measurements). As depicted in Fig. 2.2.1, the center of this cylinder is at the predicted target position, its radius is R, and its height is H. The dimensions of the association cell must be large enough to encompass all possible locations of the target, taking into account its possible displacement due to unknown velocity and acceleration components as well as uncertainties due to errors in measuring the target position and velocity. Since the measurement errors are random in nature (any bias errors will presumably cancel out in associating two

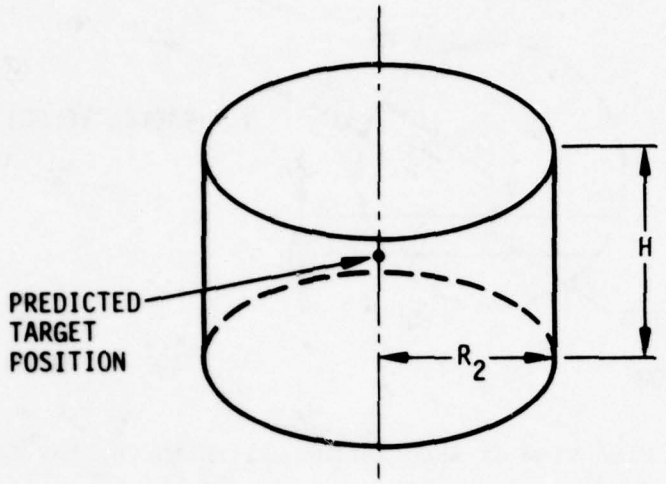


Figure 2.2.1. Assumed Association Cell Geometry

nearby measurements), the association-cell dimensions can only be large enough to ensure that the anticipated measurement will lie within the cell with high probability. The $3\text{-}\sigma$ value of a random variable, where σ is the standard deviation, is often used as a high-level-of-confidence specification; this value was used in the false-initiation analysis in specifying the contribution of the measurement error to the association-cell dimensions.

For the fifth track-initiation algorithm in which a Doppler measurement of radial velocity is used, the association volume is a vertical slice through the cylinder in Fig. 2.2.1. A plan view of this association cell is shown in Fig. 2.2.2. The magnitude of R (and of H in the vertical plane) depends on the assumed target velocity and acceleration capabilities and on the radar position-measurement accuracy; the magnitude of D depends on the Doppler measurement accuracy. In addition to its three spatial dimensions, the association cell in this case has a fourth dimension--radial velocity V. The size of the association cell

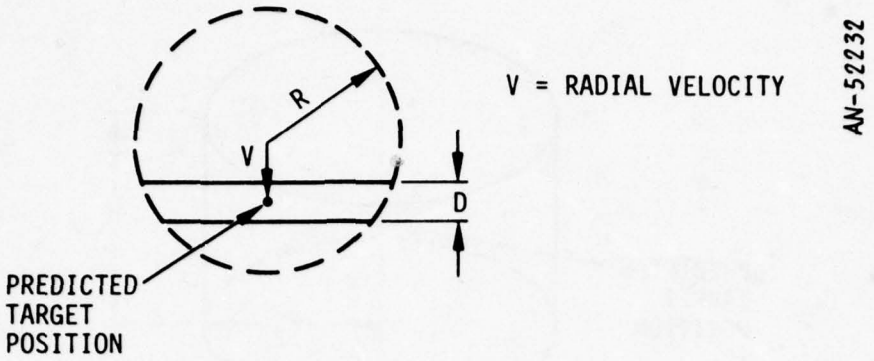


Figure 2.2.2. Plan View of Association Cell With Doppler Measurements

in that dimension depends primarily on the change in velocity due to unknown target acceleration since the last measurement and on the Doppler measurement accuracy.

For the case where only the target position has been measured (i.e., for the second measurement in the first four of the algorithms under consideration), the predicted target position is its previously measured position, and the maximum target displacement in any direction is the maximum target velocity in that direction times the elapsed time since the previous measurement. The maximum velocity in the horizontal plane must be taken to be the maximum level-flight velocity of any aircraft that could be flying in the airspace under surveillance unless additional velocity information is available that would allow the assumption of a lower velocity (such as an IFF signal which identifies the types of aircraft or a priori knowledge that certain types of aircraft operate only in certain portions of the airspace). The maximum velocity is a function of altitude (as shown in the example aircraft flight envelope of Fig. 2.3.8 on p. 169); however in this analysis a single maximum velocity is used. The maximum target velocity in the vertical direction is generally less than its maximum horizontal velocity; after some discussion with aerodynamicists

and an experienced USAF pilot, it was concluded that the maximum practical vertical velocity for high-performance aircraft is roughly one-quarter of their maximum horizontal velocity, and that figure was used in the analysis. Thus, dimensions of the correlation cell in this case are

$$R = v_m T + 3\sqrt{2}\sigma_p \tag{2.2-1}$$

$$H = 2\left(\frac{1}{4} v_m T + 3\sqrt{2}\sigma_h\right) \tag{2.2-2}$$

where v_m is the maximum aircraft velocity, T is the time since the last measurement, σ_p is the standard deviation of the position measurement error in the horizontal plane (assumed to be equal in the range and azimuth directions), and σ_h is the standard deviation of the position measurement error in the vertical plane (i.e., in target height). The two σ's are multiplied by $\sqrt{2}$ because the errors in both the first and the second position measurements contribute to the uncertainty in the relative positions of the target at the two times. As discussed above, the factor of 3 is introduced to insure that the second measurement lies within the association volume with high probability. The factor of 2 is present in Eq. 2.2-2 simply because H is defined as the total height of the association cell whereas the target displacement and measurement error extend in both directions (presumably symmetrically) from the predicted target position.

For the case where the target velocity has been determined from two previous position measurements (i.e., for the third measurement in algorithms 2, 3, and 4), the target position can be predicted using the measured velocity, and it is the error in this measurement rather than the maximum target velocity that affects the dimensions of the association cell. Unknown target accelerations can displace the target from the predicted position, however, and must be taken into account in determining

the size of the cell. In this case the dimensions of the association cell are given by

$$R = \frac{1}{2} a_m T^2 + \frac{1}{2} a_m T_1 T + 3\sqrt{\sigma_{\dot{p}}^2 T^2 + 2\sigma_p^2} \qquad (2.2\text{-}3)$$

$$H = 2\left(\frac{1}{2} a_m T^2 + \frac{1}{2} a_m T_1 T + 3\sqrt{\sigma_{\dot{h}}^2 T^2 + 2\sigma_h^2}\right) \qquad (2.2\text{-}4)$$

where a_m is the maximum aircraft acceleration, σ_p and $\sigma_{\dot{p}}$ are the standard deviations of the measured position and velocity errors in the horizontal plane, σ_h and $\sigma_{\dot{h}}$ are the position and velocity standard deviations in the vertical plane, and T_1 and T are the times between the first and second and the second and third measurements, respectively.

The first two terms in Eqs. 2.2-3 and 2.2-4 are bias terms caused by possible aircraft acceleration, as illustrated in Fig. 2.2.3. The first term, $\frac{1}{2} a_m T^2$, is caused by the deviation of the path from that which would have occurred had there been no acceleration. The second term, $\frac{1}{2} a_m T_1 T$, is caused by the error in the direction of the velocity estimate at the time of the second measurement as determined from the first two measurements, $\frac{1}{2} a_m T_1$, projected ahead at time T. (The form of these two terms is only approximately correct; more accurate representations are derived in Sec. 2.3.2.1.)

The third term is caused by measurement errors. Since these are random they are combined by root-sum-squaring. The position measurement term, $2\sigma_p^2$, includes the errors from the second and third measurements. The velocity term, $\sigma_{\dot{p}}^2 T^2$, is the velocity error at the second measurement extrapolated T seconds to the third measurement. If the velocity is determined by $V = (p_1 - p_2)/T_1$, where p_1 and p_2 are the first and second position measurements respectively, then $\sigma_{\dot{p}}^2 = 2\sigma_p^2/T_1^2$. For the case where $T_1 = T$, the third term of Eq. 2.2-3 (and similarly for Eq. 2.2-4) becomes simply $6\sigma_p$.

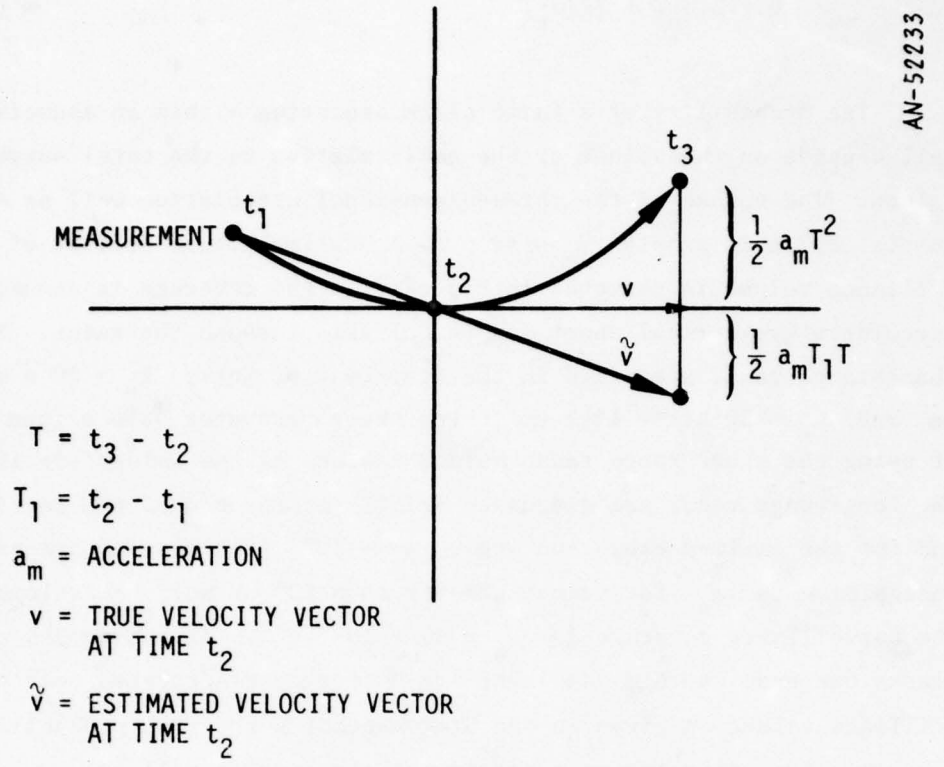


Figure 2.2.3. Geometry Showing Bias Terms in Association-Cell Size Equations

---

For the case where a Doppler measurement of radial velocity is used (Algorithm 5), the R and H dimensions for the second measurement are given by Eqs. 2.2-1 and 2.2-2 and D is given by

$$D = 2\left(\frac{1}{2} a_m T^2 + 3\sqrt{\sigma_{\dot{r}}^2 T^2 + 2\sigma_p^2}\right) \qquad (2.2\text{-}5)$$

where $\sigma_{\dot{r}}$ is the standard deviation of the radial-velocity measurement error. The size of the association cell in the fourth (radial-velocity) dimension is given by

$$V = 2(a_m T + 3\sqrt{2}\sigma_r) \qquad (2.2\text{-}6)$$

The probability of a false alarm occurring within an association cell depends on the volume of the cell relative to the total surveillance volume. The volume of the three-dimensional association cell as defined in Fig. 2.2.1 is simply $V_A = \pi R^2 H$. A vertical cross section of the surveillance volume is sketched in Fig. 2.2.4; the coverage is assumed to be circularly symmetrical about a vertical axis through the radar. For the short-range radar specified in the Statement of Work, R_s = 50 n mi = 92.6 km and H_s = 50 kft = 15.2 km . For these parameter values (the effect of using the short range radar values instead of the values specified for the long-range radar are discussed briefly at the end of the section), and for the maximum elevation angle $\phi_s = 20°$ (the results are relatively insensitive to ϕ_s for values greater than 20° or so), the volume of the surveillance coverage is $V_s = 39 \times 10^{13}$ m^3 . If the number of false alarms per scan is N_{FA} and they are uniformly distributed over the surveillance volume as given in the Statement of Work,[1] the probability of a false alarm occurring in a particular association cell is

$$P_{FA} = N_{FA} \frac{V_A}{V_s} = 0.8 N_{FA} R^2 H \times 10^{-4} \qquad (2.2\text{-}7)$$

---

[1] The actual distribution of false alarms in space depends on the design of the radar and the fact that the beam diameter increases with range R . The type of thresholding and the technique for reducing the wide dynamic-range requirements of the radar (the R^4th effect) are the primary contributors to determining the actual distribution of false alarms in space. For example, in a radar which utilizes Sensitivity Time Control (STC) to maintain a constant detection probability for a given target cross section from the minimum range of interest to the maximum surveillance range, the false alarms due to noise will tend to be concentrated near the maximum range. The assumption of uniform density simplifies the analysis, and since it is the local density of false alarms that is important, if a different density is specified on the basis of a different distribution, the results of the analysis can be scaled accordingly.

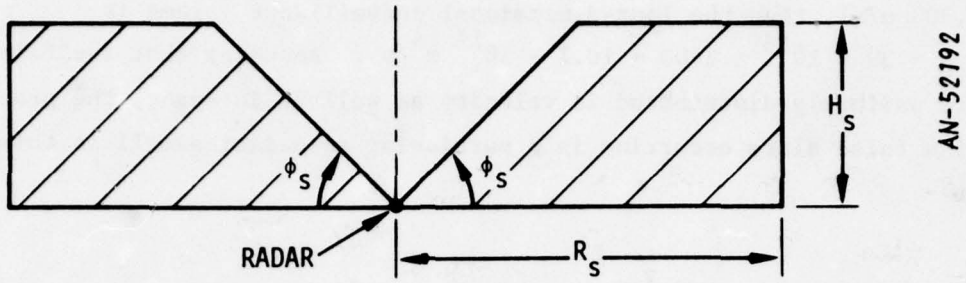


Figure 2.2.4. Vertical Cross Section of Assumed Radar Surveillance Volume

for $P_{FA} << 1$ (the situation of primary interest). Since a track will be initiated each time a false alarm occurs within the sequence of association cells (one, two, or three depending on the algorithm) following each of the N_{FA} false alarms per scan, the expected number of track initiations on false alarms per scan is then

$$\bar{N}_{FI} = N_{FA} \prod_i P_{FA_i} \qquad (2.2\text{-}8)$$

where $\prod_i P_{FA_i}$ (i = 1, 2, or 3) is the product of the probabilities given by Eq. 2.2-7 for each of the measurements in the track-initiation sequence.

For the two-pulse burst algorithm where the association cell is a vertical slice through the cylindrical volume as depicted in Fig. 2.2.2 and, in addition, includes the radial velocity as a fourth dimension, the volume of the four-dimensional correlation cell is $V_{AD} = 6RDHV$. It is assumed here as an approximation that the average width of the radial-velocity slice in Fig. 2.2.2, which could be located anywhere within the circle, is equal to R, the radius of the circle. If the total range of possible radial velocities is taken to be twice the maximum aircraft velocity as specified in the Statement of Work (2,500 knots or approximately

1,300 m/s), then the four-dimensional surveillance volume is $V_{SD} = 39 \times 10^{13} \times 2600 = 10.1 \times 10^{17}$ m^4/s . Assuming that the false alarms are uniformly distributed in velocity as well as in space, the probability of a false alarm occurring in a particular association cell in this case is

$$P_{FA} = N_{FA} \frac{V_{AD}}{V_{SD}} \simeq N_{FA} RDHV \times 10^{-18} \qquad (2.2\text{-}9)$$

The expected number of false alarms per scan in this case is calculated using this value of P_{FA} in Eq. 2.2-8.

The expected numbers of false-alarm track initiations per scan ($\bar{N}_{FI}$) were calculated for several sets of parameter values for each of the algorithms under consideration. The resulting values of $\bar{N}_{FI}$ for the two-measurement and three-measurement algorithms are plotted in Figs. 2.2.5 and 2.2.6 as functions of the scan time or the time between measurements. As indicated, measurement-error standard deviations of $\sigma_p = 100$ m and $\sigma_h = 200$ m (for the three-dimensional measurements) were used in these calculations; these values are consistent with the capabilities of the AN/TPS-43 radar at typical ranges and at least at the lower elevation angles, and are used as representative values for other radars that might be used in this application. A false-alarm rate N_{FA} of 100 per scan, the maximum value specified in the Statement of Work for the short-range radar, was used in the calculations, but the dependence of $\bar{N}_{FI}$ on N_{FA} is specified (i.e., proportional to N_{FA}^2 for the two-measurement algorithm and to N_{FA}^3 for the three-measurement algorithm) so that the results can be easily scaled to other values of the false-alarm rate. The plots of $\bar{N}_{FI}$ are drawn for two values of the assumed maximum target velocity, the maximum high-performance aircraft velocity of 2,500 knots ≈ 1,300 m/s specified in the Statement of Work and half that value to show the sensitivity of the results to this parameter, and for two values of the assumed maximum target acceleration, 6g and 3g, for

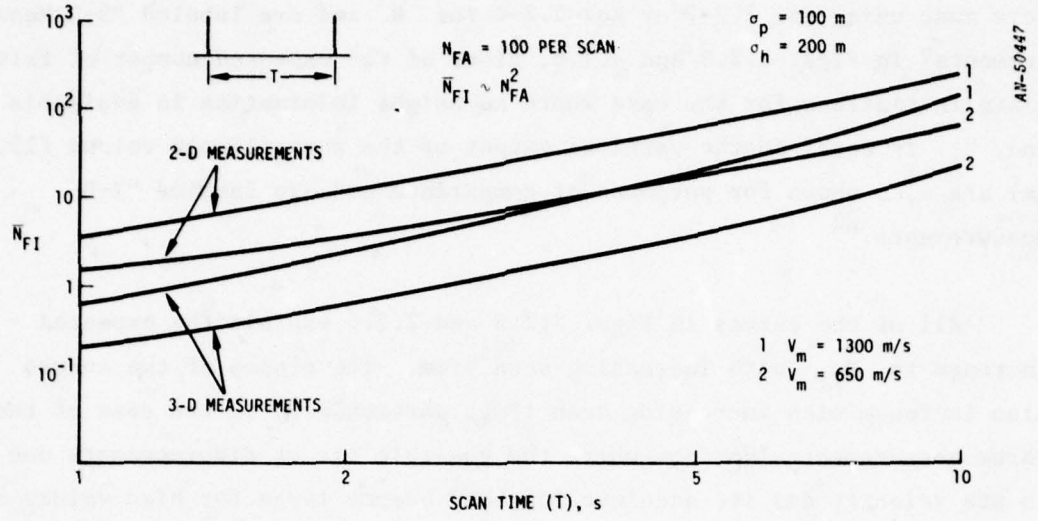


Figure 2.2.5. Expected Number of False-Alarm Track Initiations per Scan With Two Position Measurements

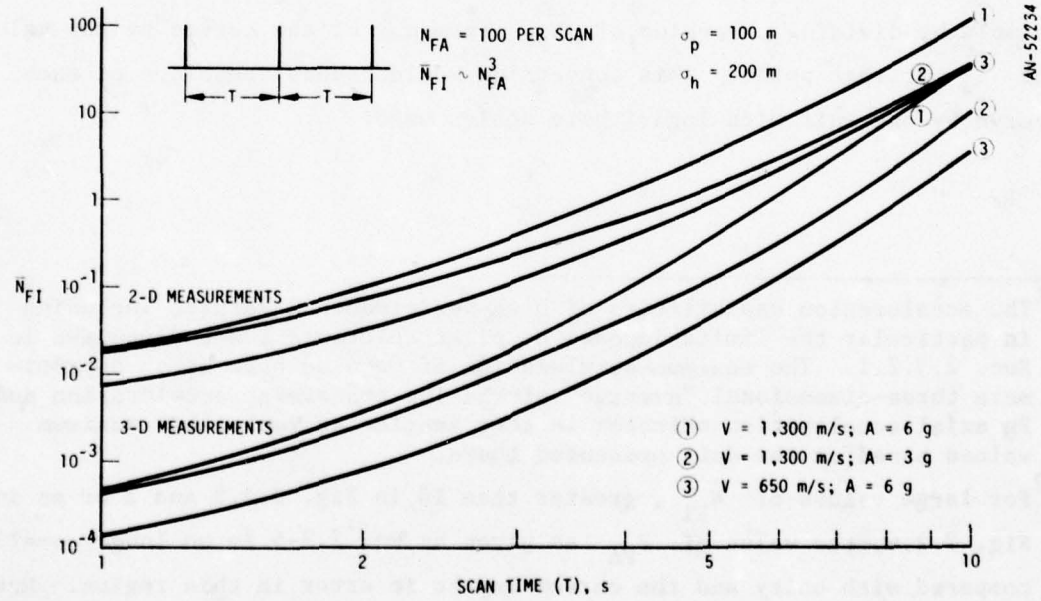


Figure 2.2.6. Expected Number of False-Alarm Track Initiations per Scan With Three Equally Spaced Pulses

the three-measurement algorithm.[1] In addition to the plots of $\bar{N}_{FI}$ which were made using Eq. 2.2-2 or Eq. 2.2-4 for H and are labeled "3-D Measurements" in Figs. 2.2.5 and 2.2.6, plots of the expected number of false-alarm initiations for the case where no height information is available and H_c is equal to the vertical extent of the surveillance volume (15.2 km) are also shown for purposes of comparison and are labeled "2-D Measurements."[2]

All of the curves in Figs. 2.2.5 and 2.2.6 exhibit the expected increase in $\bar{N}_{FI}$ with increasing scan time. The slopes of the curves also increase with increasing scan time, particularly in the case of the three-measurement algorithm where the possible target displacements due to its velocity and its acceleration both become large for high values of the scan time. For some purposes, such as determining the data processing and communications loads imposed by the false-alarm track initiations, the expected number of false-alarm initiations per second may be of greater interest than the number per scan. If so, the expected number of false-alarm initiations per second can be obtained from Figs. 2.2.5 and 2.2.6 simply by dividing the value of $\bar{N}_{FI}$ from any of the curves by the value of T_s at that point. This conversion would reduce the slope of each curve by one unit with logarithmic scales used.

---

[1] The acceleration capabilities of high-performance aircraft, including in particular the limits imposed by pilot tolerances, are discussed in Sec. 2.3.2.1. The maximum acceleration of 6g used here is an approximate three-dimensional "average" of the 10g transverse acceleration and 2g axial acceleration selected in that section as reasonable maximum values based on the data presented there.

[2] For large values of $\bar{N}_{FI}$, greater than 10 in Fig. 2.2.5 and 3 or so in Fig. 2.2.6, the value of P_{FA} as given by Eq. 2.2-5 is no longer small compared with unity and the curves may be in error in this region. However, sets of parameter values that produce values of $\bar{N}_{FI}$ of this magnitude are probably of little interest as is discussed further below.

The dependence of $\bar{N}_{FI}$ on the assumed maximum target velocity varies slightly with scan time depending upon the relative magnitude of the measurement errors. The expected number of false-alarm initiations is rather insensitive to the assumed target acceleration, particularly for the shorter scan times. An indication of the dependence on the magnitude of the measurement errors is provided in Fig. 2.2.7 where $\bar{N}_{FI}$ for the three-measurement algorithm is plotted as a function of the standard deviation σ_p of the horizontal-plane measurements for three values of scan time. As indicated, the vertical-measurements standard deviation is taken to be equal to $2\sigma_p$ in all cases, and the results plotted are for an assumed velocity of 1,300 m/s and acceleration of 6g's. As expected, $\bar{N}_{FI}$ is more sensitive to variations in σ_p for the shorter scan times where the velocity and acceleration displacements are smaller. At the shorter and intermediate scan times the dependence can be significant--for example, for a scan time of 5 seconds, reducing σ_p from 100 m to 50 m reduces $\bar{N}_{FI}$ by a factor of 2--but the effects of the choice of algorithm and the scan time are far greater than the effects of small variations in σ_p.

The performance of the third and fourth track-initiation algorithms, which utilize a measurement pair followed by a single measurement or another pair, is affected by another parameter--the measurement-pair spacing--in addition to the parameters considered so far. The expected numbers of false-alarm initiations are plotted as functions of this spacing in Fig. 2.2.8 for the two-measurement-pair algorithm and in Fig. 2.2.9 for the measurement-pair-plus-single-measurement algorithm. As indicated, these plots are for a scan time of 5 seconds and for the same set of parameter values used with the first two algorithms. These results show that $\bar{N}_{FI}$ depends very little on the measurement-pair spacing, at least over the range of values considered. It turns out that as the spacing T_1 increases, the velocity measurement accuracy improves and the size of the association cell for the third measurement decreases, but that the

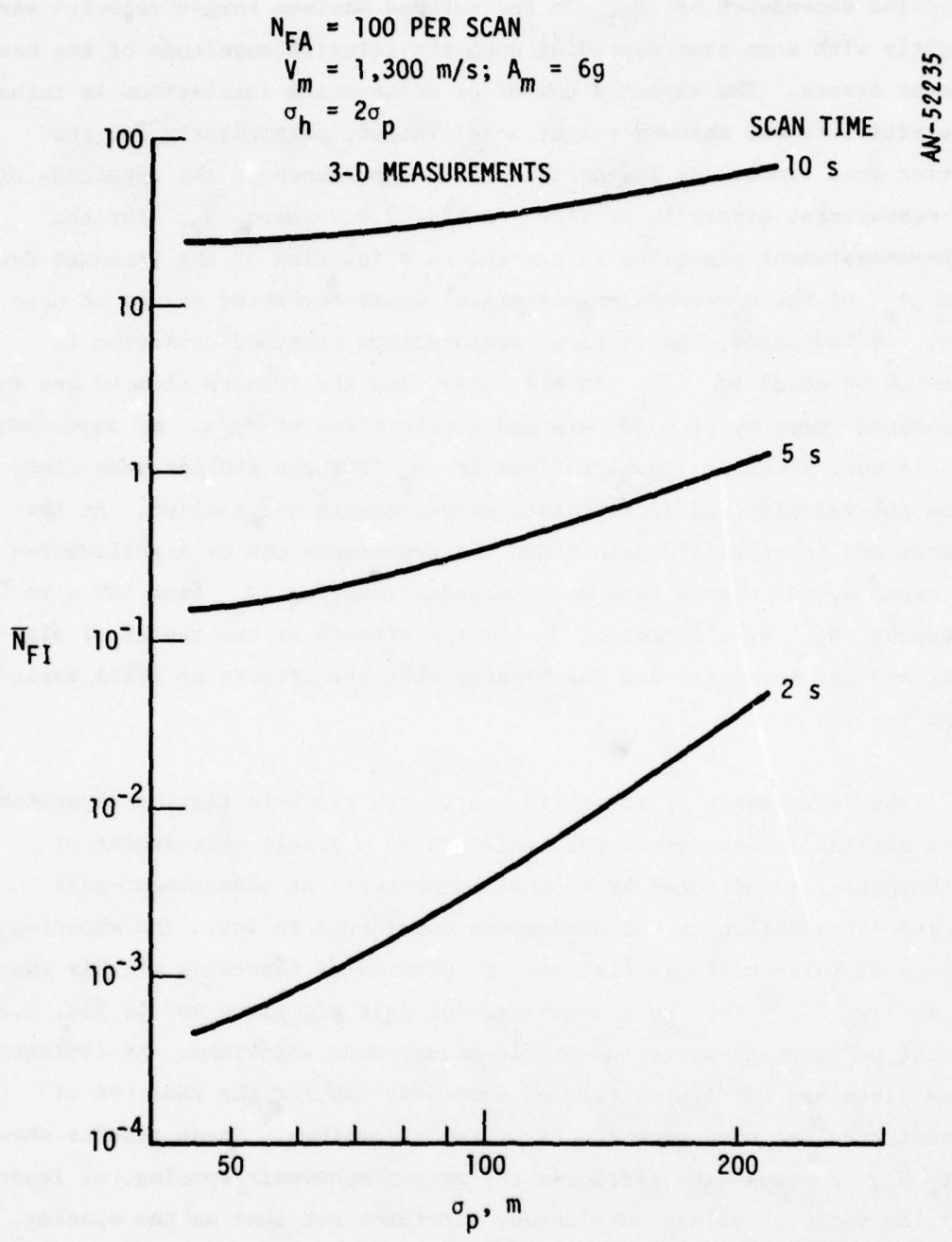


Figure 2.2.7. Expected Number of False-Alarm Track Initiations per Scan With Three-Pulse Algorithms as a Function of the Measurement Error

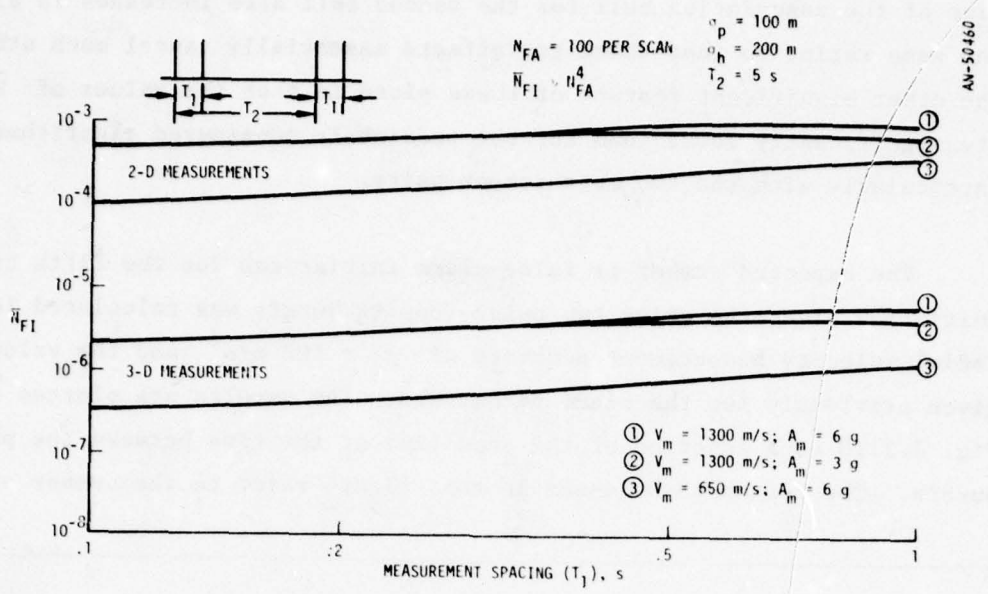


Figure 2.2.8. Expected Number of False-Alarm Track Initiations per Scan With Two Measurement Pairs

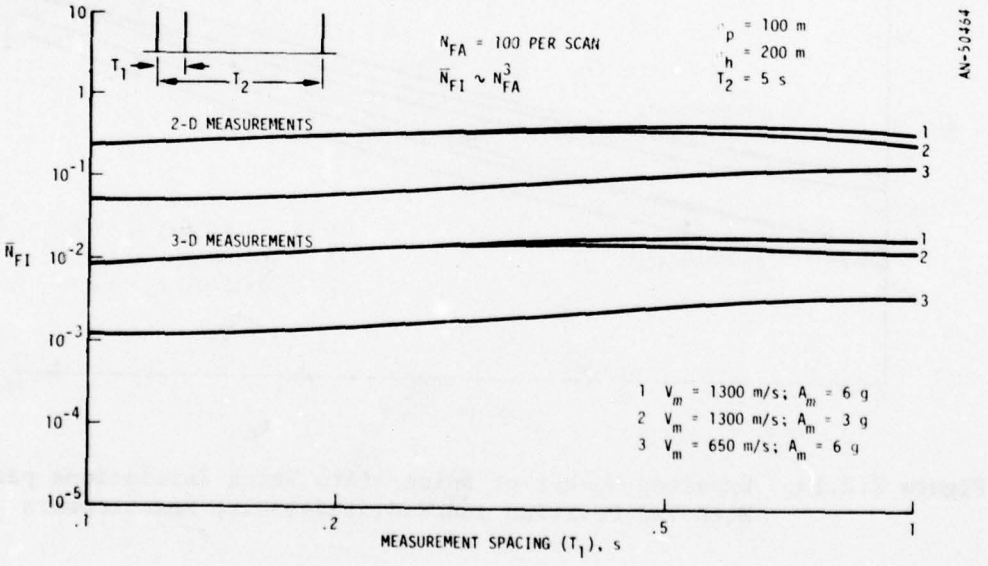


Figure 2.2.9. Expected Number of False-Alarm Track Initiations per Scan With a Measurement Pair Plus One Measurement on the Next Scan

size of the association cell for the second cell also increases in almost the same ratio, so that these two effects essentially cancel each other. The other significant feature of these plots is that the values of $\bar{N}_{FI}$ are significantly lower than for the previously considered algorithms, particularly with the two measurement pairs.

The expected number of false-alarm initiations for the fifth track-initiation algorithm using two pulse-Doppler bursts was calculated for a radial velocity measurement accuracy of $\sigma_{\dot{r}}$ = 100 m/s[1] and the values given previously for the other parameters. The results are plotted in Fig. 2.2.10 as a function of the scan time or the time between the pulse bursts. The 2-D and 3-D labels in this figure refer to the number of

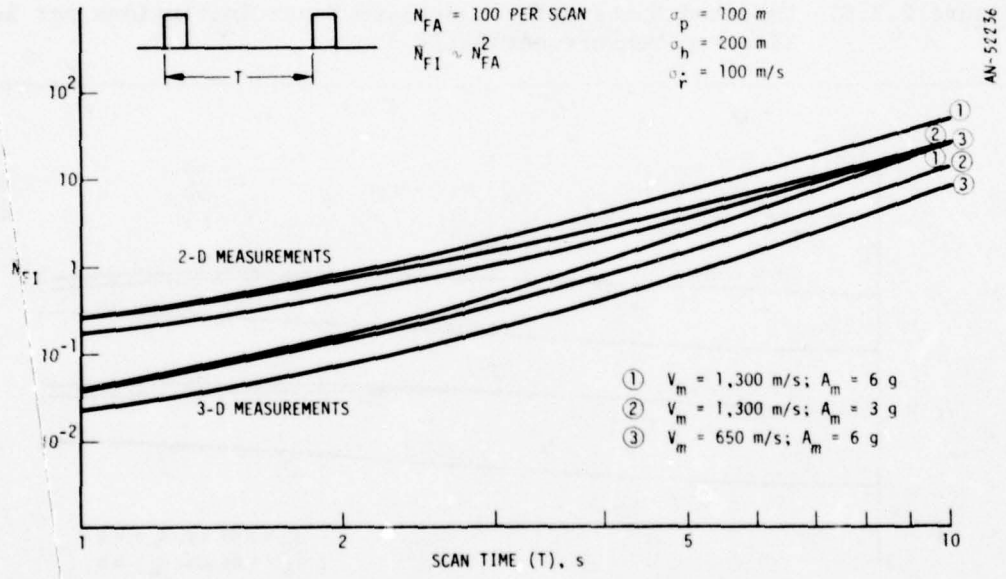


Figure 2.2.10. Expected Number of False-Alarm Track Initiations per Scan With Two Position and Radial-Velocity Measurements

[1] At S-Band, a pulse burst duration of about 50 μs with a high signal-to-noise ratio can provide a velocity measurement accuracy of 100 m/s.

spatial dimensions as described previously; the Doppler measurement of radial velocity provides an additional dimension but it is not included in this labeling. The curves in Fig. 2.2.10 are similar in shape to those for the two-measurement algorithm in Fig. 2.2.5, but the values of $\bar{N}_{FI}$ are lower as expected with the additional information provided by the Doppler measurements.

A comparison of the false-alarm track-initiation rates for all five of the algorithms considered is provided in Fig. 2.2.11. The plots of the expected number of false-alarm initiations versus scan time are for three-dimensional measurements with the parameter values discussed previously and listed in the figure. The improvement in performance with increasing algorithm complexity and amount of information is obvious and in some cases quite marked. In comparing the shapes of the curves, it should be noted that the measurements for the three-measurement algorithm are made over a time interval equal to twice the scan time, whereas the measurements for the other algorithms are made over an interval approximately equal to the scan time.

To use these results in selecting a track-initiation algorithm and specifying suitable values of the radar parameter values (particularly the scan time), a tolerable level of false-alarm track initiations per scan (or per second) must be established. A detailed analysis of the consequences of track initiations on false alarms is required to establish such a specification, but the application of a simple criterion might provide a useful basis for drawing some preliminary conclusions. It might be reasonable to assume, for example, that the false-alarm initiation rate should not exceed 10% of target track-initiation rate. For 1,000 aircraft uniformly distributed over the coverage volume of the system and moving through that volume at an average velocity of 300 m/s, an aircraft will enter the surveillance volume of each radar at an average rate of approximately one per second. Then based on the criterion suggested above, the false-alarm initiation rate--the expected number of

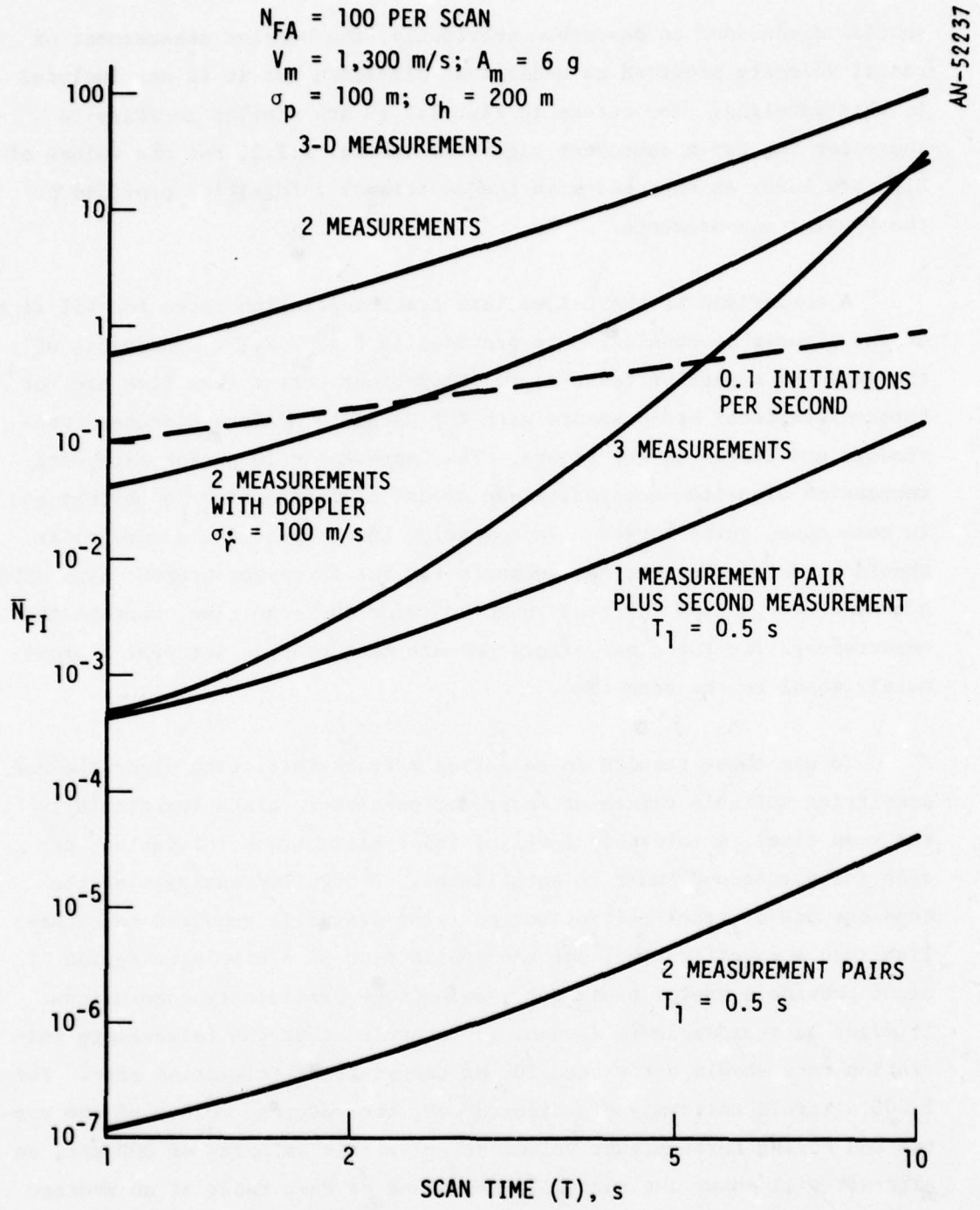


Figure 2.2.11. Expected Number of False-Alarm Track Initiations per Scan--Comparison of the Algorithms Considered

track initiations on false alarms per second--should not exceed 0.1. This threshold is plotted as the broken line in Fig. 2.2.11. The expected number of false-alarm initiations is seen to be above this threshold for all values of the scan time for the two-measurement threshold, and for all but the shorter scan times with two Doppler measurements. The false-alarm-initiation rate is below this threshold for the three-measurement algorithm except for scan times approaching ten seconds and for the measurement-pair-plus-one-measurement algorithm, and is well below this level with two pulse pairs.

In addition to the false-alarm-initiation performance, another factor that should be considered in selecting a track-initiation algorithm and radar parameter values is the probability of successful initiation as affected by the radar probability of detection. For successful track initiation, each set of returns in the track-initiation sequence must be detected, i.e., must produce a signal which exceeds the detection threshold. Ideally, the probability of detection is close to unity, but in the environment of a tactical engagement the detection probability may be degraded significantly--a probability of detection as low as 0.25 is mentioned in the Statement of Work. The probability of successful track initiation falls off rapidly as the probability of detection is reduced from unity as shown in Fig. 2.2.12. The expressions used in calculating the probability of initiation for the four types of algorithms[1] under consideration are listed in the figure. In these expressions the probability of detection P_D is assumed to be the same for all of the measurements in a track-initiation sequence. There is a significant difference between the probabilities of initiation for the various algorithms, particularly for intermediate values of P_D. Note that Algorithm 4, which produces the second lowest number of false-alarm initiations among the algorithms considered, provides a probability of successful initiation

[1] Algorithms 1 and 5 are identical as far as calculation of the probability of successful initiation are concerned.

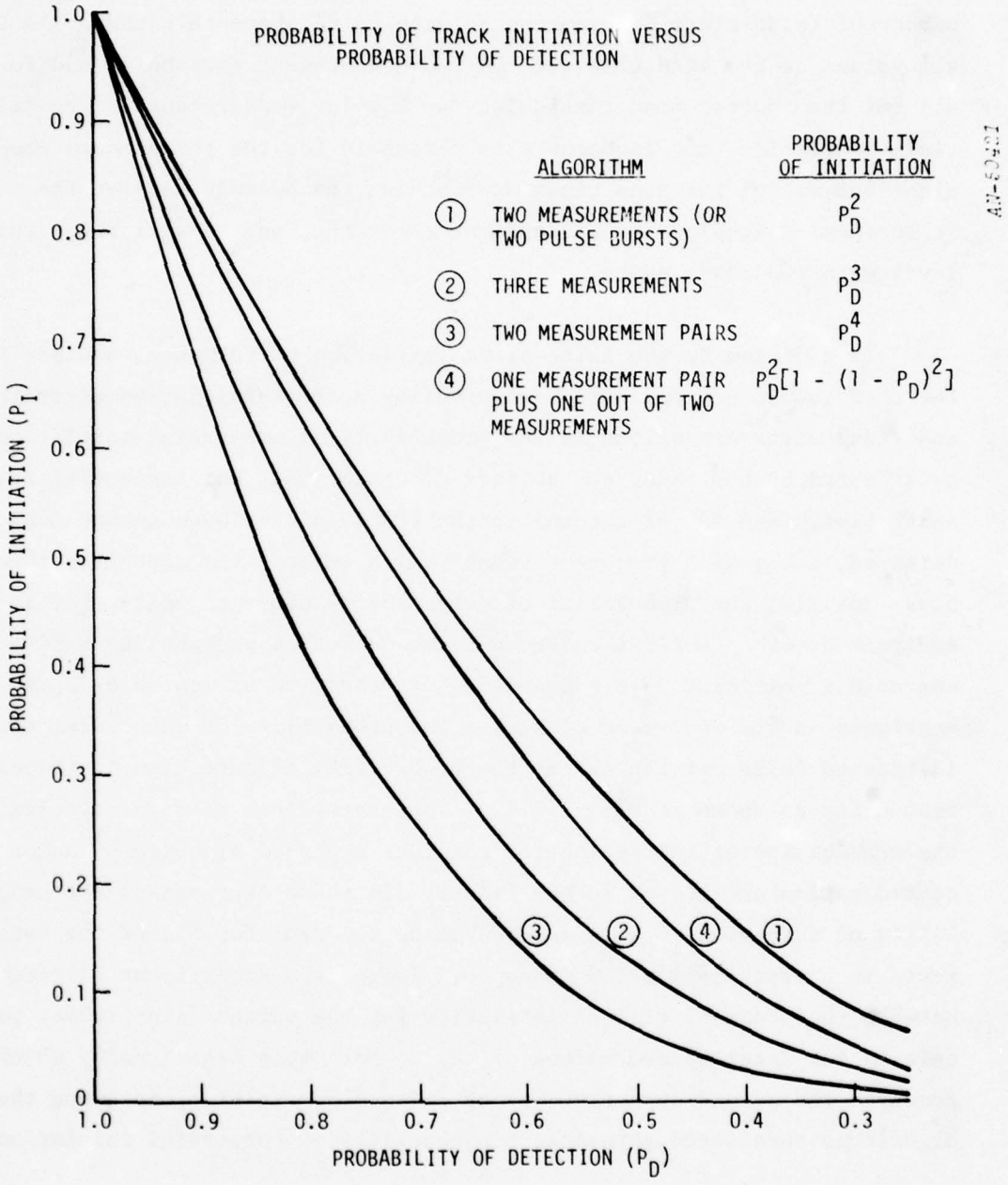


Figure 2.2.12. Probability of Successful Track Initiation Versus Probability of Detection

that is close to the best attainable. The probabilities of initiation plotted in Fig. 2.2.12 are of course for a single attempt (i.e., a single sequence of measurements); as a target moves through the surveillance volume, a number of track-initiation attempts would presumably be made by each radar, and the combined probability of initiation would be much higher. Nevertheless, reliable early track initiation is undoubtedly desirable to provide timely information to the system users and to conserve radar and data processing resources.

As pointed out earlier in this section, the expected numbers of false-alarm track initiations were calculated for parameter values consistent with those specified in the Statement of Work for a short-range radar. The results can be scaled fairly easily to reflect changes in these parameter values. For example, consider the use of a long-range radar specified in the Statement of Work to have a range of 200 n mi (compared with 50 n mi for the short-range radar) and coverage to an altitude of 100 kft (instead of 50 kft), with up to 200 false alarms per scan (instead of 100). The surveillance volume of the long-range is greater than that of the short-range radar by a factor of 32. On the other hand, the N_{FA}^{i} term is greater by 4, 8, or 16, depending on the number of measurements used (2, 3, or 4). Thus, ignoring the effect of changes in measurement accuracy which tend to be dominated by the velocity and acceleration-induced displacements, the expected numbers of false-alarm initiations with the long-range radar would be reduced by factors of 8, 4, or 2 from the values for the short-range radar, depending on the type of algorithm used.

The expected numbers of false-alarm initiations per scan calculated using the formulation in this section were compared with the results of TACRAN3 simulation runs as described in Sec. 3.5.3. These comparisons were made for the two-measurement and three-measurement track-initiation algorithms, with the calculated values scaled to the parameter values used in the simulation. Good agreement between the calculated values and simulation results was obtained for both algorithms as indicated in Sec. 3.5.3.

2.3 TRACK ASSOCIATION

The process of track association (correlation) is central to the operation of any system which must detect and track large numbers of targets. Failure to associate a radar return with the proper target track--or to determine correctly that it does not associate with any track in the file--can cause duplicate tracks to be established, introduce large errors in the track estimate, and possibly cause a loss of track. The latter consequence leads at best to increased radar and data processor loads for repeated track initiations, and at worst to the loss of needed information about the target and its flight path.

Track association is similar to track initiation in that a current measurement of target position (and perhaps Doppler velocity) is compared with the expected target position based on previous measurements. The principal difference is that in the association process the expected target position is predicted from a number of previous radar measurements which have been appropriately combined and smoothed, rather than from only one or two previous track-initiation measurements.

The probability of correctly associating a radar return with the proper track depends on the magnitudes of the measurement error, the track prediction error, and the deviation of the target position from the predicted track due to maneuvers initiated since the last measurement, as well as on the association algorithm used and on the density of false alarms and other targets in the vicinity of the target of interest. Analyses of the effects of these factors on the association performance are presented in this section.

2.3.1 Association Algorithms

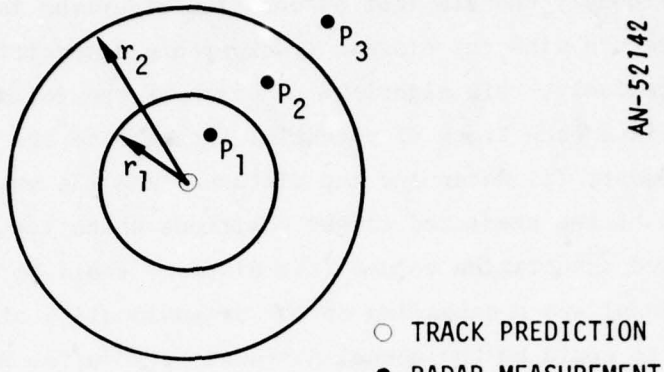
The association algorithms considered in the study are described in this section. Few papers on the subject appear in the literature; some of these are given in the Bibliography (1-3).

2.3.1.1 Closest-Pair Algorithm

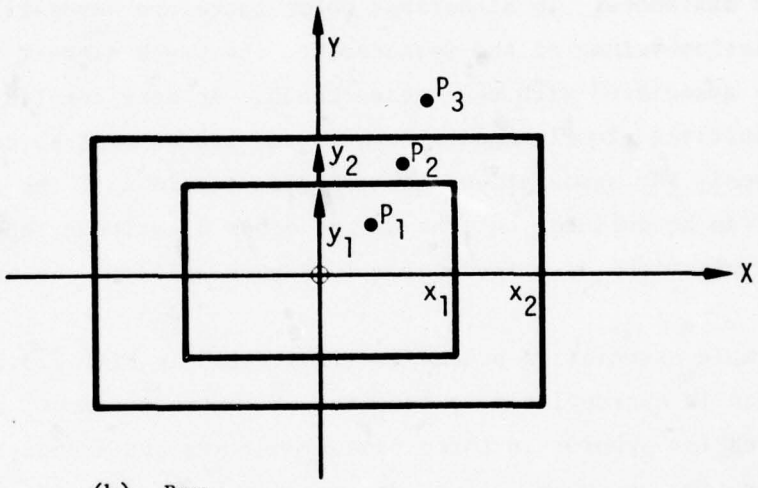
Probably the simplest association algorithm is to associate each radar return with the closest track within a specified association volume. More precisely, this algorithm consists of the following steps: (1) extrapolate each track of potential interest to the time of the radar measurement; (2) determine the distance from the measured target position to each of the predicted target positions which lie within a previously specified association volume (the distance could be in two, three, or four dimensional space depending on the dimensionality of the radar measurements; it could be the actual distance between the measured and predicted target positions or it could be a statistical distance as described below); and (3) associate the radar return with one of the tracks based on the calculated distances. In situations where there are several tracks within the association-volume of the measurement, the track closest to the measurement is associated with that measurement. In more complex situations in which multiple closely spaced returns and tracks must be considered simultaneously for association, more complex versions of the closest-pair algorithm can be defined, but the use of other algorithms such as the assignment algorithm described below in Sec. 2.3.1.2 is preferable.

A simple association scheme is illustrated in Fig. 2.3.1(a). A target position is extrapolated to the time of the measurement, and one or more circles (or spheres in three dimensions) are put around the extrapolated position as shown. If the measured point is outside the outer circle (as in P_3), the measurement does not associate. The radius of the outer circle is chosen so that it encompasses any possible maneuver plus errors due to noise.

In some tracking filters (Sec. 2.4) an estimate of whether or not the target is maneuvering is required. The inner circle is used for this purpose. If the measurement point is between r_1 and r_2 (as in P_2), the target is maneuvering. If the distance is less than r_1 (as in P_1),



(a) Sphere



(b) Box

CASES:

P_1 ASSOCIATES, NONMANEUVERING

P_2 ASSOCIATES, MANEUVERING

P_3 NO ASSOCIATION

Figure 2.3.1. Simple Closest-Pair Actual-Distance Algorithms

the target is not maneuvering. The threshold r_1 is chosen so that it encompasses only a very modest maneuver (such as 1g) plus measurement noise.

The above computations of course use distance-squared rather than distance for computational simplicity. Another computationally simple algorithm is to put a box around the extrapolated target rather than a sphere. The sides of the box are lined up with the axes (in either radar or Cartesian coordinates), as shown in Fig. 2.3.1(b), and thus the association algorithm can be performed one coordinate at a time without computing the distance-squared. The disadvantage is that it is unknown which of two tracks is closest to the measurement without computing the distance. An advantage, however, is that if the association is done in radar coordinates, the box size can be different for each dimension depending on the different measurement errors in each dimension.

Statistical Distance. A considerable refinement of the concept of having the box size different in each dimension is to compute the so-called "statistical distance" between the measurement and the extrapolated state. A two-dimensional example illustrates this concept. Figure 2.3.2 shows a measurement (x_m, y_m) and an extrapolated target position $(\hat{x}_s, \hat{y}_s)$. The difference between the measurement and extrapolated position (called the measurement "residual") is $(\tilde{x}, \tilde{y})$. The true distance D between the two is computed from

$$D^2 = \tilde{x}^2 + \tilde{y}^2$$

It may be, however, that the variance of $\tilde{x}$ is considerably larger than the variance in $\tilde{y}$. Thus if the measurement is far off in x it still perhaps should associate, but if it is equally far off in y it should not associate. A measure is needed which normalizes the errors by their variances; this measure, d, is called the statistical distance:

$$\tilde{x} = x_m - \hat{x}_s$$

$$\tilde{y} = y_m - \hat{y}_s$$

TRUE DISTANCE:

$$D^2 = \tilde{x}^2 + \tilde{y}^2$$

STATISTICAL DISTANCE:

$$d^2 = \frac{\tilde{x}^2}{\sigma^2_{\tilde{x}}} + \frac{\tilde{y}^2}{\sigma^2_{\tilde{y}}}$$

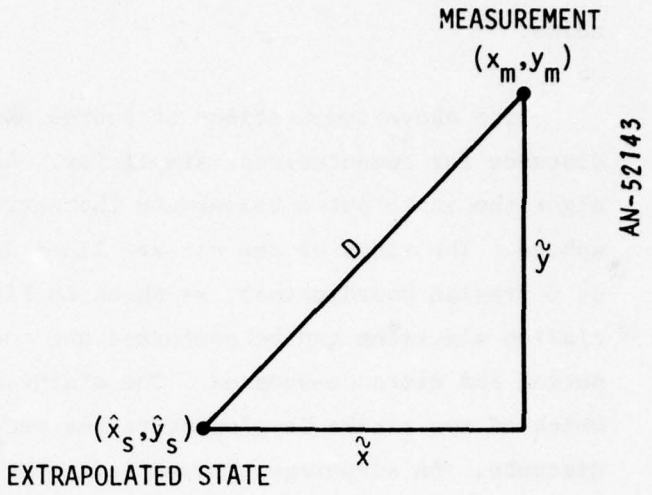


Figure 2.3.2. Two-Dimensional Association Example for Uncorrelated Components

---

$$d^2 = \frac{\tilde{x}^2}{\sigma^2_{\tilde{x}}} + \frac{\tilde{y}^2}{\sigma^2_{\tilde{y}}}$$

In general, the statistical distance should be calculated using the entire covariance matrix, not just the variances. Thus the statistical distance is correctly calculated as

$$d^2 = \underset{\sim}{\tilde{z}}^T V^{-1} \underset{\sim}{\tilde{z}} \qquad (2.3\text{-}1)$$

where $\underset{\sim}{\tilde{z}}$ is the measurement residual vector (the difference between the measurement vector and the extrapolated state in measurement coordinates--see Eq. 2.4-4, pp. 211, 212) and V is the covariance matrix of $\underset{\sim}{\tilde{z}}$ (see Eq. 2.4-7, p. 215).

One reason that the statistical distance is sometimes considered for association is because d^2 has a known probability distribution and therefore a meaningful threshold can be set. Since d^2 is a sum of the squares of Gaussian random variables, all with unity variances (because the variances have all been normalized), d^2 is chi-square distributed. Therefore, given any threshold d_t^2, the probability that $d^2 < d_t^2$ can be computed.

Figure 2.3.3 shows this probability as a function of the dimension n of the association space. (This is called a chi-square distribution with n degrees of freedom.) For association in position only, $n = 3$. Thus, for example, if it is desired that probability of association of a measurement with its correct track is to be 90%, then the threshold is $d_t^2 = 6.25$. If the threshold is made too large, then the probability of an incorrect association being made increases.

If, as is often the case in tactical air surveillance systems, the primary association errors are caused by aircraft maneuvers rather than statistical measurement errors (see Sec. 2.3.2), the use of statistical distance provides little improvement over the use of actual distance. Considering the extra computation involved, it should probably not be used in these cases in associating measurements with tracks.

Even if statistical distance is not used for associating measurements with tracks, it is appropriate for associating tracks with tracks, a process which needs to be performed to eliminate duplicate tracks. In this case the errors <u>are</u> due to primarily measurement errors, and if the covariance of the tracks is available, it should probably be used.

The procedure is as follows: Consider two tracks $(\hat{\underline{x}}_1, P_1)$ and $(\hat{\underline{x}}_2, P_2)$ which have been extrapolated to the same time (see Sec. 2.4.1). The difference between the states is $(\hat{\underline{x}}_1 - \hat{\underline{x}}_2)$ and the covariance of the difference is $(P_1 + P_2)$. The statistical difference d^2 between the states is then

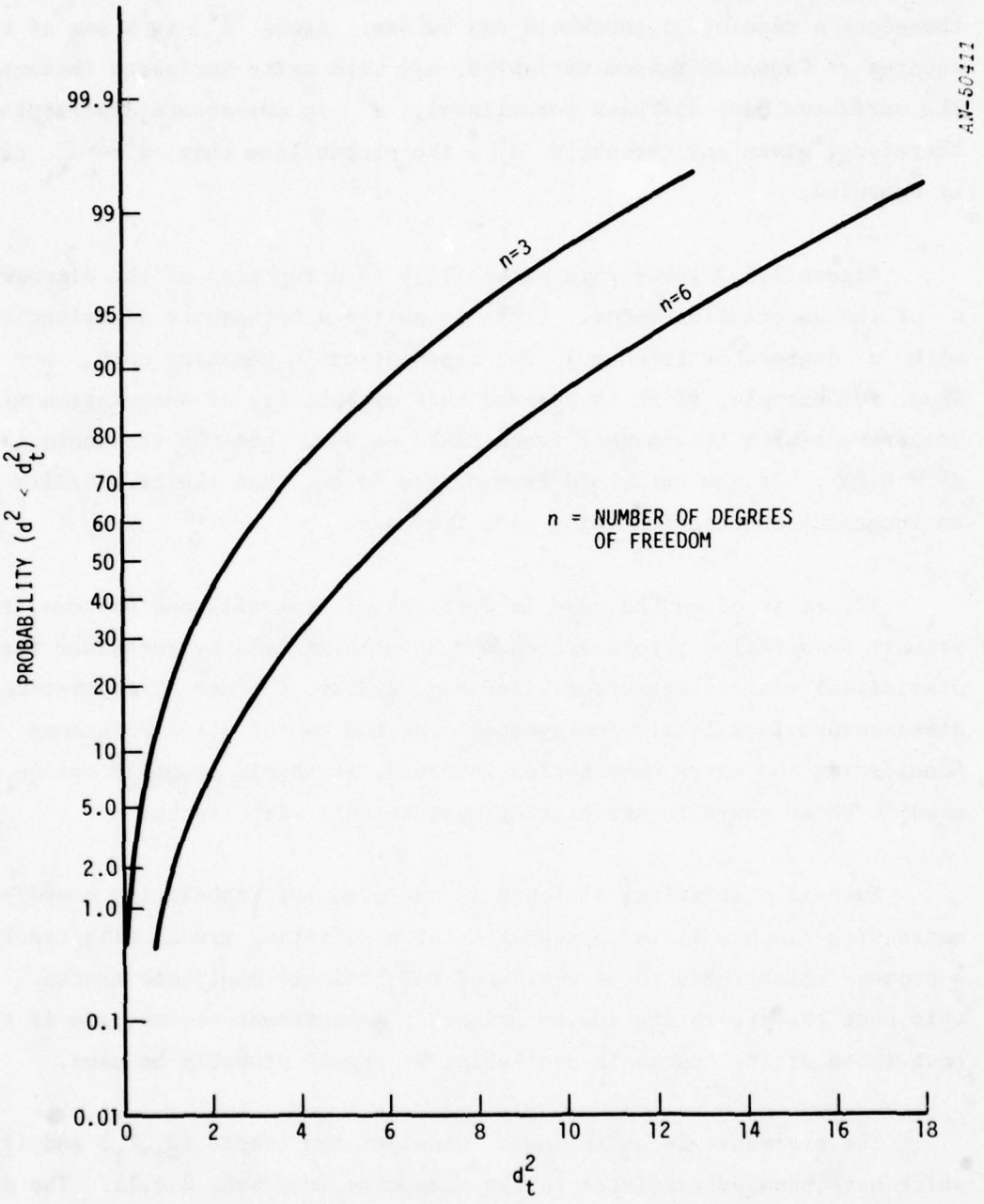


Figure 2.3.3. Chi-Square Distribution

$$d^2 = (\underline{x}_1 - \underline{x}_2)^T (P_1 + P_2)^{-1} (\underline{x}_1 - \underline{x}_2) \qquad (2.3\text{-}2)$$

The test to determine if the two tracks are identical is $d^2 < d_t^2$, where d_t^2 is the threshold. This test is in 6 dimensions if the track consists of position and velocity, in 9 dimensions if acceleration is included. The probability that $d^2 < d_t^2$ if the tracks are indeed the same is shown in Fig. 2.3.3 for the six-dimensional case (n = 6). Of course increasing d_t^2 to increase the probability of correct association also increases the probability that two different tracks will pass the test, thus choosing d_t^2 involves the usual tradeoff between correct and false associations.

2.3.1.2 Assignment Algorithm

When multiple measurements and tracks lie close enough together that they must be considered as a group for association, such as the situation depicted in Fig. 2.3.4 in two dimensions, the distances between measurements and tracks that could be associated with the same target (those which lie within the potential-association volume with respect to one another) can be conveniently displayed and processed as elements of a two-dimensional matrix. Different measures of distance can be used; however, the square of the distance, or statistical-distance squared, which tend to weigh more heavily the larger measurement-track separations, appear to be reasonable measures for this application. The distance-squared matrix for the measurement-target distribution of Fig. 2.3.4 is shown in Fig. 2.3.5.

With the situation described in this way, a reasonable and well-defined procedure for associating each measurement with a track is to determine the set of pairings which minimizes the sum of the squares of the distances. That is, as the solution of the association problem, choose that combination of measurements and tracks, out of all possible combinations, minimizes Σd_{mi}^2, where d_{mi} is the distance from measurement m to track i and the summation is over M or I terms, whichever is

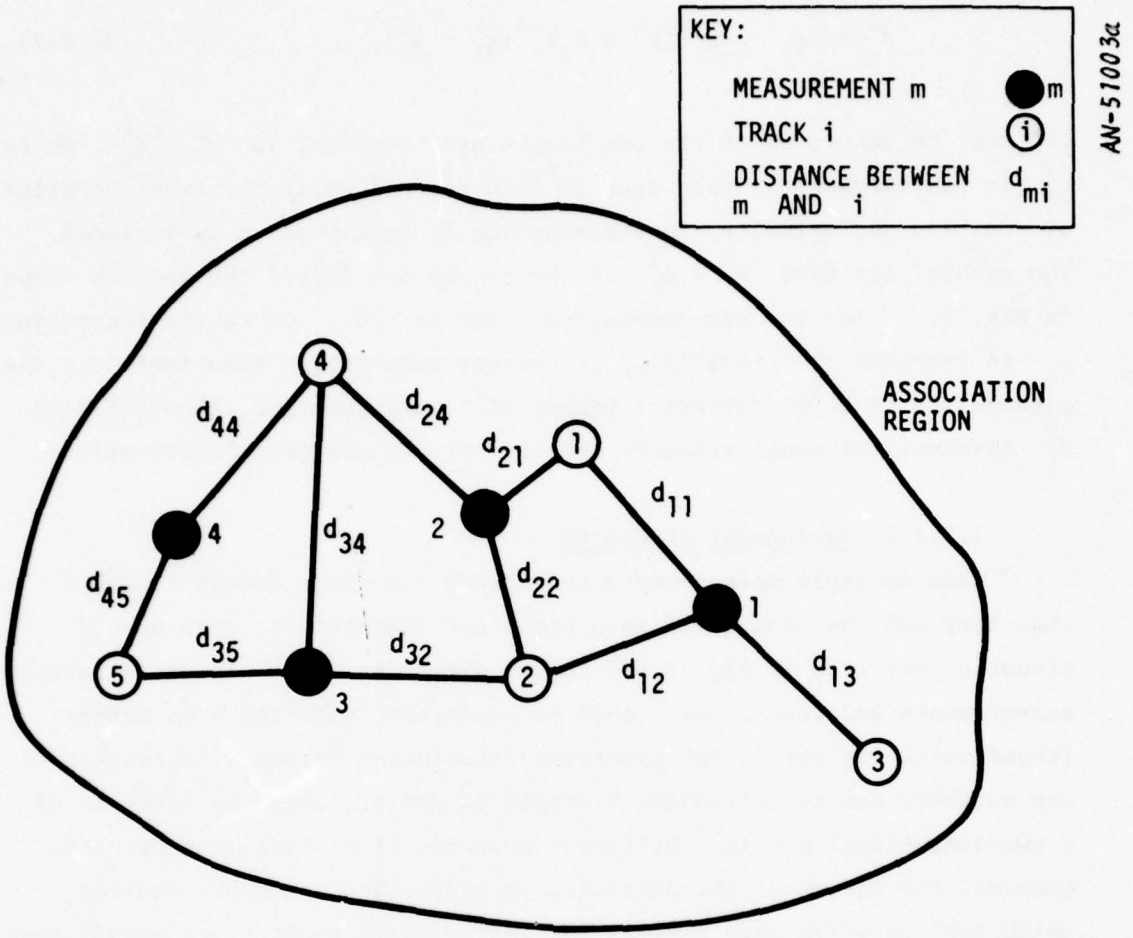


Figure 2.3.4. Multiple-Target Association Geometry

less, and M and I are the number of measurements and tracks being considered. One term is selected from each row and column. For example, in Fig. 2.3.4, one possible combination of distance-squared values is the set of circled elements in the matrix of Fig. 2.3.5. This type of problem is known as the assignment problem in network flow theory, and efficient algorithms for its solution are known.[1] This assignment algorithm

[1] L.R. Ford, Jr., and D.R. Fulkerson, Flow in Networks, The RAND Corporation Report, No. R-375-PR, Princeton University Press, 1962.

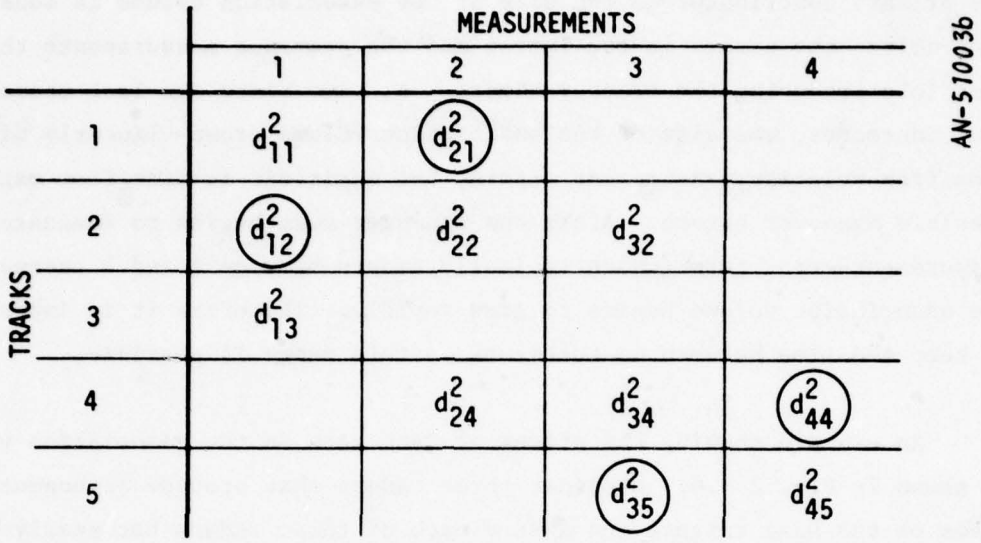


Figure 2.3.5. Matrix of Association Distances-Squares for Geometry of Fig. 2.3.4

---

was implemented in TACRAN3 as discussed in Sec. 3.5. Its track-association performance in particular situations is considered in Sec. 2.3.3.2.

2.3.2 Association Volume

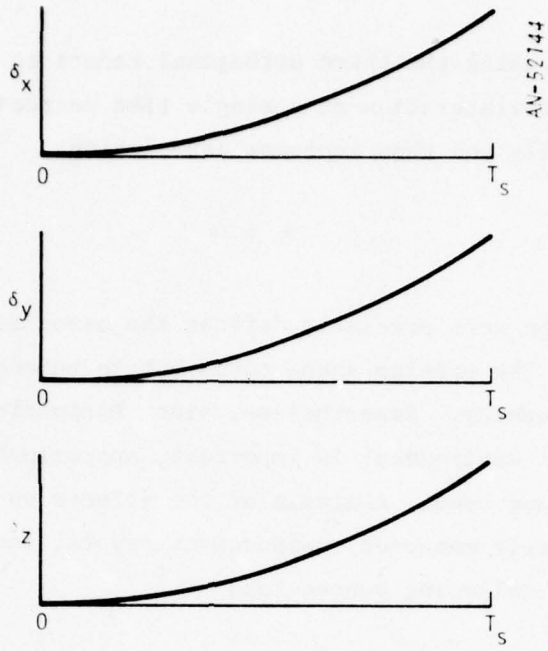
The size and shape of the potential association volume--the volume surrounding the predicted target position within which the next measurement is expected to lie--depends on the maximum possible acceleration of the target since the previous measurement, the radar measurement errors as they affect both the track prediction and the current position measurement, and the model errors introduced by any difference between the trajectory shape assumed in smoothing the tracking data and the actual flight path of the target.

In a dense target environment it is important to minimize the association volume in order to minimize the probability of making a false association. Immediately after updating the track from a new measurement,

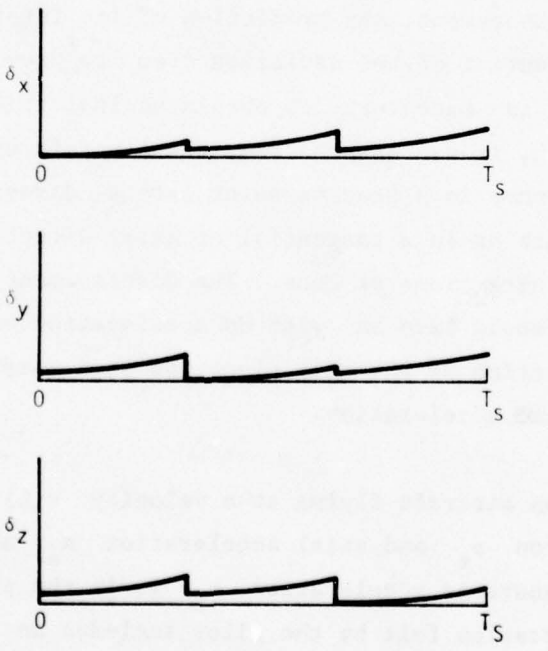
the primary contributor to the size of the association volume is measurement noise--the errors in the latest and the previous measurements that went into producing the track estimate. As time since the last measurement increases, the size of the association volume grows--linearly with time from velocity measurement errors, but quadratic in time from maximum possible maneuver errors. After the maneuver term begins to dominate the measurement-error term (which typically occurs between 1 and 3 seconds), the association volume begins to grow rapidly. Therefore it is important to keep the time between measurements in this range if possible.

An example showing the effect of data rate on the association volume is shown in Fig. 2.3.6. Consider three radars that provide orthogonal views on the same target, and assume each of these radars has nearly perfect accuracy in one (range) dimension and much less accuracy in the other two (angle) dimensions. This is the perfect case for trilateration; by making all of the measurements at the same time and combining them, a nearly perfect measurement can be made. Figure 2.3.6(a) shows the growth of the association volume in the three dimensions during the time after the trilateration measurements before the next measurement T_s seconds later. The plotted curve is for $T_s = 6$ s , a maximum maneuver acceleration of 10g, and $\delta = \frac{1}{2} at^2$, an approximation that will be shown valid in the next subsection. The association error at the next measurement time is over 1,700 meters.

Now consider using the same three measurements equally spaced in time; that is, one radar makes a measurement every $T_s/3 = 2$ seconds . At each measurement time a nearly perfect range measurement and two much worse angle measurements are made. The radar making the measurement at $t = 0$ in Fig. 2.3.6(b) makes a perfect x measurement and imperfect y and z measurements with 100-m standard deviation error. At $t = 2$ s the second radar makes a measurement, this time the perfect measurement in the y direction. The third radar likewise makes a measurement at $t = 4$ s . The maximum association error in this case is less than 350



(a) Three Measurements at Same Time



(b) Three Measurements Equally Spaced

Figure 2.3.6. Growth of Association Volume for Two Different Timings of the Measurements by Multiple Radars

meters. Clearly using the three orthogonal radars to improve data rate rather than for trilateration at a single time decreases the association volume considerably and thus improves association.

* * *

This section more precisely defines the association volume in both size and shape. The precise shape turns out to be concave, and difficult to model in a computer. Nevertheless, since minimizing false associations in a dense target environment is important, approximations to the shape should be found and used. Analysis of the effects on association shape and size of aircraft maneuver, measurement errors, and model errors are presented in the following subsections.

2.3.2.1 Aircraft Acceleration Effects

If an aircraft that is being tracked initiates a maneuver after the last tracking measurement, any prediction of its future position will be in error by the extent of the deviation from its previous flight path that is introduced by the maneuver. To obtain an indication of the magnitude of this deviation, it was assumed that the aircraft undergoes a constant acceleration, either in a transverse or lateral direction perpendicular to its flight path or in a tangential or axial direction along its flight path or in both directions at once. The displacement of the aircraft from the position it would have had with no acceleration was calculated and plotted as a function of the time since the last measurement and the aircraft velocity and acceleration.

Consider an aircraft flying at a velocity $v(t)$ with constant transverse acceleration a_t and axial acceleration a_a as indicated in Fig. 2.3.7. (The transverse acceleration a_t is in the plane of the maneuver; the total acceleration felt by the pilot includes an additional 1g vertical acceleration to counter gravity.) In a time dt, the angle θ changes by an amount

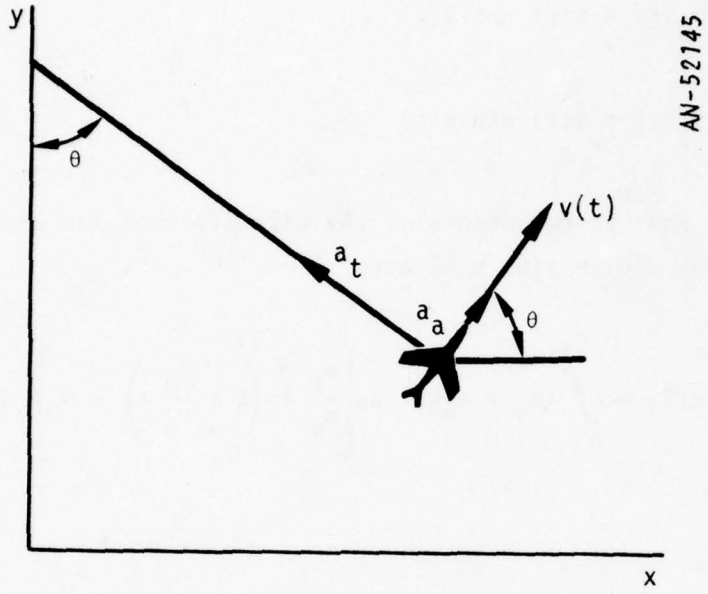


Figure 2.3.7. Maneuvering Aircraft Geometry

$$d\theta = \frac{a_t}{v(t)}\, dt$$

Now $v(t) = v_o + a_a t$, where v_o is the aircraft velocity at time $t = 0$. Thus

$$\theta(t) = \int_0^t \frac{a_t}{v_o + a_a u}\, du + \theta_o$$

$$= \frac{a_t}{a_a} \ln\left(1 + \frac{a_a}{v_o} t\right) + \theta_o$$

where θ_o is the value of $\theta(t)$ at $t = 0$. Since the x and y components of $v(t)$ are

$$v_x(t) = v(t) \cos \theta(t)$$

$$v_y(t) = v(t) \sin \theta(t)$$

then the x and y components of the aircraft position at time T [starting from $x(0) = y(0) = 0$] are

$$x(T) = \int_0^T (v_o + a_a t) \cos\left[\frac{a_t}{a_a} \ln\left(1 + \frac{a_a}{v_o} t\right) + \theta_o\right] dt$$

and

$$y(T) = \int_0^T (v_o + a_a t) \sin\left[\frac{a_t}{a_a} \ln\left(1 + \frac{a_a}{v_o} t\right) + \theta_o\right] dt$$

Approximating the natural logarithm by the first term in its series expansion gives

$$x(T) \simeq \int_0^T (v_o + a_a t) \cos\left(\frac{a_t}{v_o} t + \theta_o\right) dt \quad \text{for} \quad \frac{a_a t}{2v_o} << 1$$

and

$$y(T) \simeq \int_0^T (v_o + a_a t) \sin\left(\frac{a_t}{v_o} t + \theta_o\right) dt \quad \text{for} \quad \frac{a_a t}{2v_o} << 1$$

Carrying out the integrations yields

$$x(T) \simeq \frac{v_o^2}{a_t}\left[\left(1 + \frac{a_a}{v_o} T\right) \sin\left(\frac{a_t}{v_o} T + \theta_o\right) - \frac{a_a}{a_t}\left(1 - \cos\left(\frac{a_t}{v_o} T + \theta_o\right)\right)\right] \qquad (2.3\text{-}3)$$

$$y(T) \simeq \frac{v_o^2}{a_t}\left[1 - \left(1 + \frac{a_a}{v_o} T\right) \cos\left(\frac{a_t}{v_o} T + \theta_o\right) + \frac{a_a}{a_t} \sin\left(\frac{a_t}{v_o} T + \theta_o\right)\right] \qquad (2.3\text{-}4)$$

A special case of Eqs. 2.3-3 and 2.3-4 occurs when the aircraft is maneuvering at constant speed: $a_a = 0$. In this case (and assuming $\theta_o = 0$),

$$x(T) = \frac{v_o^2}{a_t} \sin \frac{a_t}{v_o} T$$

$$y(T) = \frac{v_o^2}{a_t}\left(1 - \cos \frac{a_t}{v_o} T\right)$$

These equations for $x(T)$ and $y(T)$ are the parametric equations of a circle with radius v_o^2/a_t.

To evaluate and plot the flight-path deviations given by Eqs. 2.3-3 and 2.3-4 requires that the aircraft accelerations and velocity be specified. Indications of the maximum lateral acceleration that should be considered are provided in Figs. 2.3.8 and 2.3.9. The plots of acceleration capability versus altitude in Fig. 2.3.8 for a typical high-performance fighter aircraft show that the maximum acceleration at the lower altitudes, where it is limited by the pilot tolerance, is from 5.5g to 8g. The

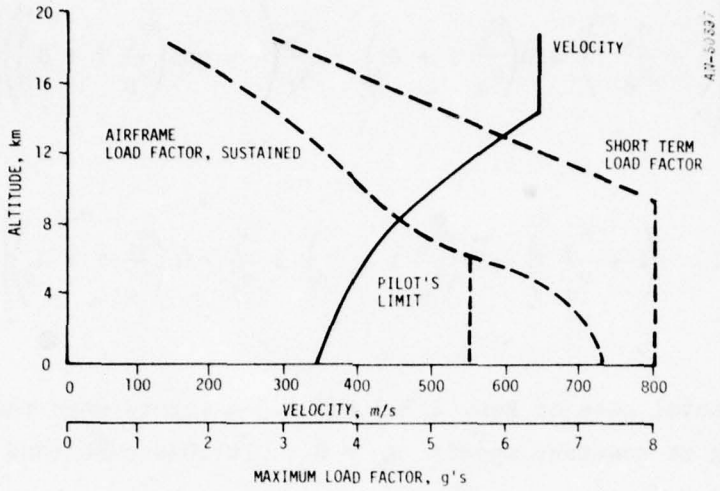


Figure 2.3.8. Flight Envelope of a High-Performance Fighter

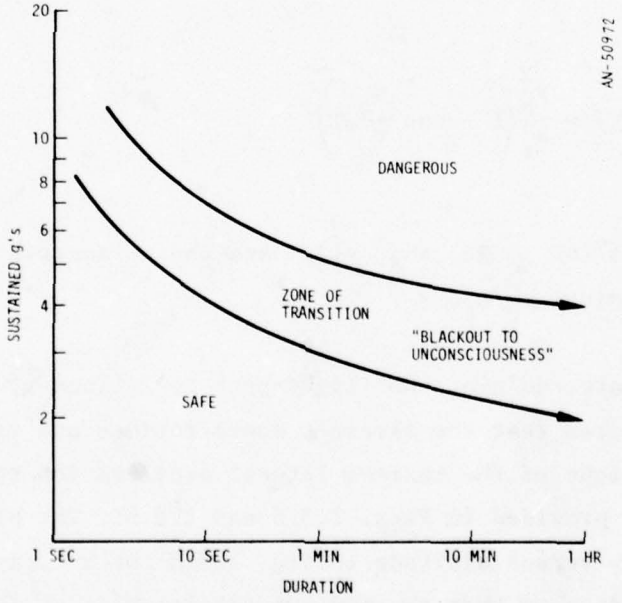


Figure 2.3.9. Effect of Maximum Sustained Accelerations for Human Pilots (from Litton Report No. MS-00678, 2 October 1970)

plots of acceleration magnitude versus duration in Fig. 2.3.9,[1] indicate that pilots can tolerate accelerations of perhaps as much as 10g for a few seconds. In this same Litton report, axial aircraft accelerations of 1g to 2g are mentioned. Based on these data, a lateral acceleration of 10g and an axial acceleration of 2g were selected as conservatively high values for illustrating the effects of target acceleration on the association volume.

The loci of aircraft positions as given by Eqs. 2.3-3 and 2.3-4 for $a_t = \pm 10g$, $a_a = \pm 2g$, and $v_o = 300$ m/s are plotted in Fig. 2.3.10 for three values of the time since the last measurement. The axial acceleration is constant at +2g or -2g along the curved portions of the contours, while the transverse acceleration is constant at +10g or -10g along the straight-line portions. These contours can be considered to be the boundaries of possible locations of the aircraft in the plane of the flight path after the specified time intervals. If the transverse acceleration capability is the same in all directions perpendicular to the flight path, the association volume due to target acceleration is obtained by rotating these contours about the x- (along-track) axis.

The rapid increase in size of the association volume as the time since the last measurement increases is shown graphically in Fig. 2.3.10. At a time of 5 seconds, the volume extends laterally to a distance of about 1,200 meters. The increases curvature of the volume is also apparent; this curvature is insignificant at short time intervals but becomes substantial at long times.

The effect of target velocity on the location and shape of the association volume is illustrated in Fig. 2.3.11, which shows the

---

[1] Final Report--A Parametric Study of the Advanced Forward Area Air Defense Weapon System (AFAADS), Vol. I--Systems Analysis, Data Systems Division, Litton Systems, Inc., Report No. MS-00678, 2 October 1970 (UNCLASSIFIED).

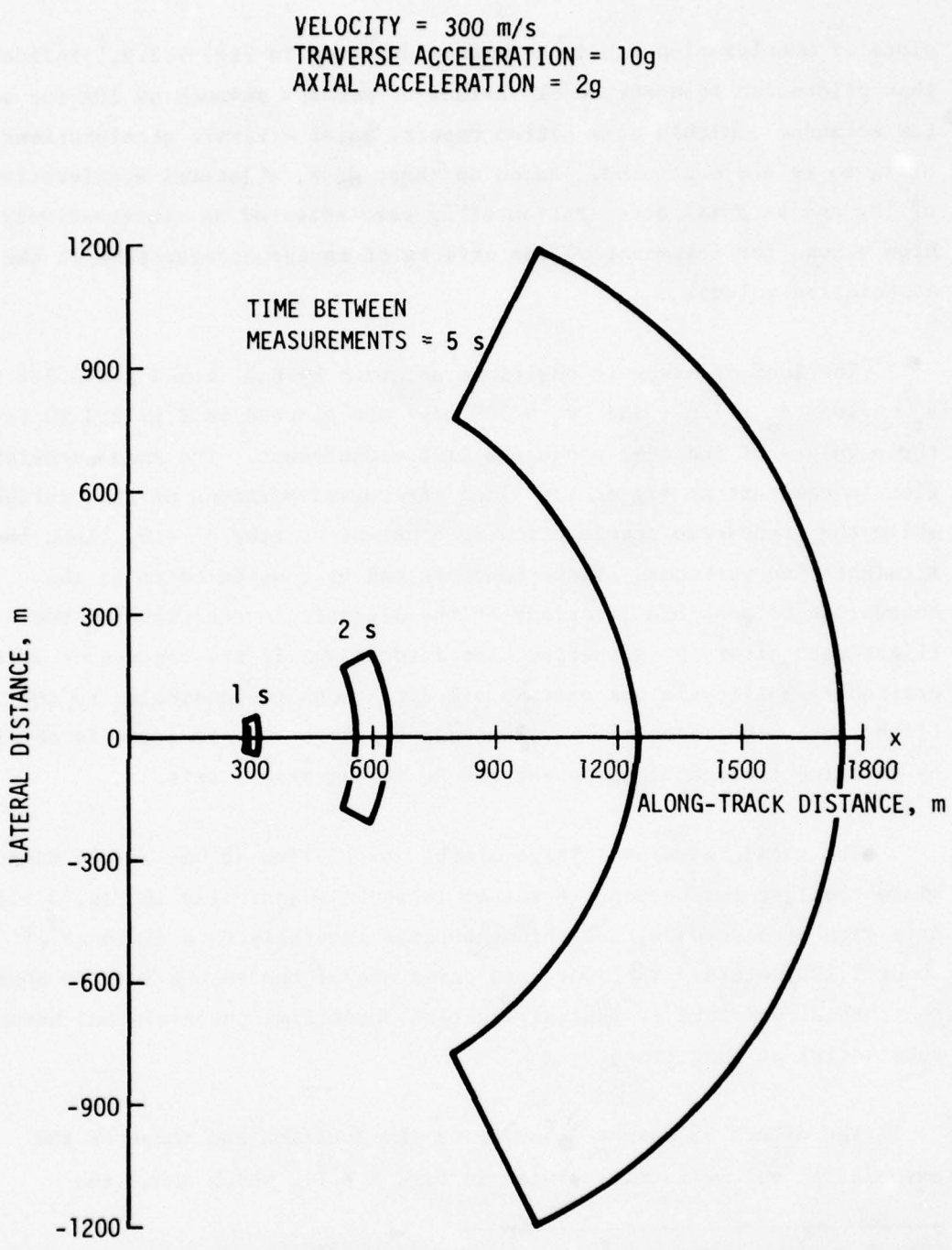


Figure 2.3.10. Projection of Association Volume Due to Possible Target Acceleration

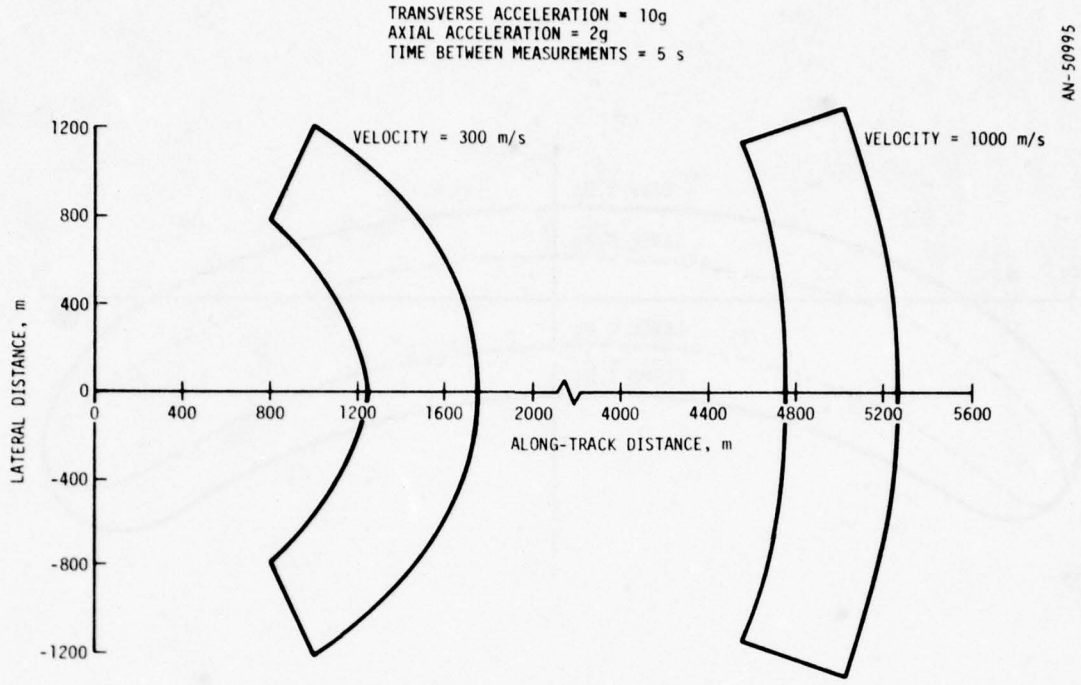


Figure 2.3.11. Projection of Association Volume Due to Possible Target Acceleration

association-volume contour for a velocity v_o of 1,000 m/s at a time of 5 seconds after the last measurement, along with the contour for v_o = 300 m/s from the previous figure. The two uncertainty volumes are about the same size, but the curvature is considerably less at the higher target velocity.

Actual association-volume contours would probably have rounded corners since it is unlikely that an aircraft and pilot could maintain high accelerations both laterally and axially at the same time. An analysis of maneuver capabilities based on energy coupling effects[1] produced a contour similar to the ones in Figs. 2.3.10 and 2.3.11 but with rounded corners; it is shown in Fig. 2.3.12 for purposes of comparison.

[1] Litton report, op. cit.

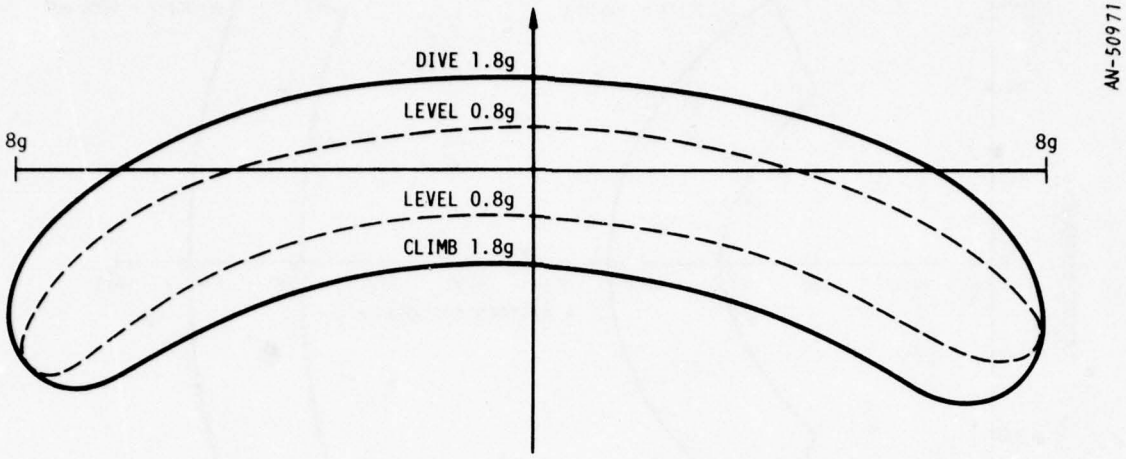
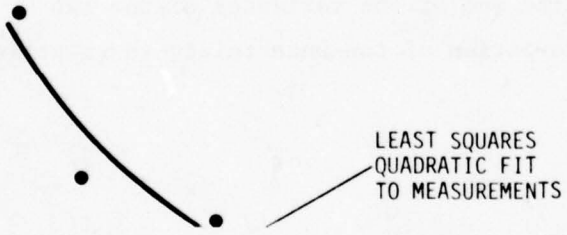


Figure 2.3.12. Relative Acceleration Capability of a High Performance Fighter/Bomber Along and Perpendicular to the Direction of Flight (from Litton Report, op. cit.)

2.3.2.2 Measurement Error Effects

The errors in the radar measurements of target position (and perhaps radial velocity), which are caused by thermal noise and perhaps other environmental or target-induced effects, lead to track estimation and prediction errors. The error in the predicted target position must be taken into account in determining the size and shape of the potential association volume. The error in the measurement that is to be associated with the target track must also be taken into account in defining the association volume since association is based on the distance between the measured and predicted target positions. The measurement and prediction errors of interest are random in nature and can only be described statistically. The association volume, defined with respect to the predicted target position, should be made large enough to ensure that a specified (presumably high) percentage of the correct measurements will lie within its boundaries.

The target prediction error depends on the procedure used to combine and smooth the measurements and to extrapolate the track to future times, as well as on the errors in the measurements. One of the filtering techniques described in Sec. 2.4--the one used in TACRAN3--is to fit a quadratic function to the preceding N measurements so as to minimize the sum of the squares of the distances between the measurements and the quadratic curve--the so-called least-squares fit. The predicted position of the target then lies along this quadratic curve extrapolated into the future. However, for purposes of locating the association volume, a linear extrapolation along the tangent to the quadratic fit at the last measurement is more appropriate. With such an extrapolation, the association volume is essentially symmetric about its center line since the maximum possible lateral acceleration is presumably equal in all directions. This geometry is depicted in Fig. 2.3.13.



AV-52146

The error in the predicted target position due to the random measurement errors was analyzed for the least-squares fit to a quadratic function. For both the extrapolation along the least-squares quadratic fit and the linear extrapolation along the tangent, the standard deviation of the error in predicted position (σ_p) is a multiple of the standard deviation of the measurement error (σ_m), with the value of the multiplying factor being a function of the number of measurements to which the quadratic is fitted (N) and the extrapolation time in relation to the time between measurements (Δ). This factor for linear extrapolation is plotted in Fig. 2.3.14 as a function of N for three extrapolation times. These curves are based on the assumption that the N measurements have the same variance and are uniformly spaced. As N increases from its minimum possible value of three, the σ_p/σ_m ratio falls rapidly at first but then starts to level off at N equal to six or seven, particularly for an extrapolation time of one Δ which is the case of greatest interest. At N = 6 with an extrapolation time of one Δ, for example, $\sigma_p \simeq 1.6\sigma_m$.

Since the errors in the measurement to be associated and the predicted track are independent, the variance of the error in the distance between these positions is just the sum of the variances of the two errors. That is, the standard deviation of the uncertainty in relative position is given by

$$\sigma_u = \sqrt{\sigma_p^2 + \sigma_m^2}$$

For the above example where $\sigma_p = 1.6\sigma_m$, $\sigma_u = 1.9\sigma_m$.

In order to compare the magnitude of this measurement-induced error with the target displacement calculated above, it is necessary to assign a value to σ_m. For the AN/TPS-43 radar, σ_m is of the order of 100 m in both range and azimuth for typical ranges of interest. While the measurement accuracy of other radar designs for the application of interest

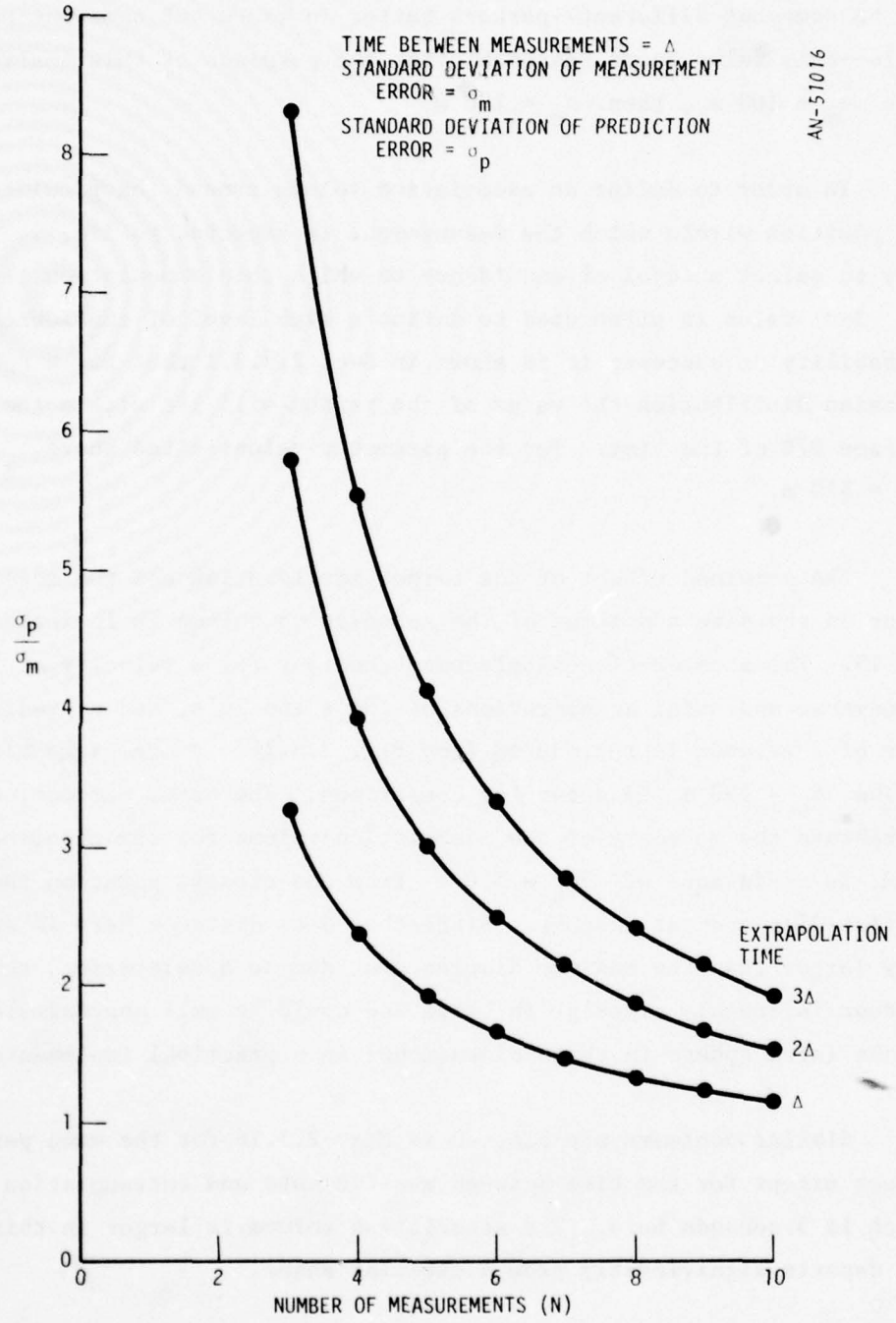


Figure 2.3.14. Error in Predicted Position for Linear Extrapolation From a Quadratic Fit to N Measurements

may be somewhat different--perhaps better in range but somewhat poorer in angle--this value seems reasonable for the purposes of this analysis. With $\sigma_m = 100$ m , then $\sigma_u = 190$ m.

In order to define an association volume around the predicted target position within which the measurement is expected to lie, it is necessary to select a level of confidence to which this expectation is held. The $3\text{-}\sigma$ value is often used to define a high level of confidence or probability of success; it is shown in Sec. 2.3.3.1 that for a spherical Gaussian distribution the value of the random will lie within the $3\text{-}\sigma$ surface 97% of the time. For the parameter values cited above, $3\sigma_u = 570$ m.

The combined effect of the target acceleration and the measurement error on the size and shape of the association volume is indicated in Fig. 2.3.15. The acceleration-displacement contour for a velocity of 300 m/s, transverse and axial accelerations of 10g's and 2g's, and a prediction time of 2 seconds is reproduced from Fig. 2.3.10. A one-sigma circle of radius $\sigma_u = 190$ m is shown for comparison. The outer contour, which represents the boundary of the association volume for the parameter values used, is a distance of $3\sigma_u = 570$ m from the closest point on the acceleration-displacement contour. Since the $3\text{-}\sigma$ distance here is considerably larger than the maximum displacement due to acceleration, this outer contour is roughly circular in shape and could be well approximated by a circle (or a sphere in three dimensions) in a practical implementation.

Similar contours are plotted in Fig. 2.3.16 for the same parameter values except for the time between measurements and extrapolation time, which is 5 seconds here. The association volume is larger in this case and departs significantly from a circular shape.

Association-volume contours for the case where the time between measurements is 2 seconds and the prediction time is 5 seconds are shown in Fig.

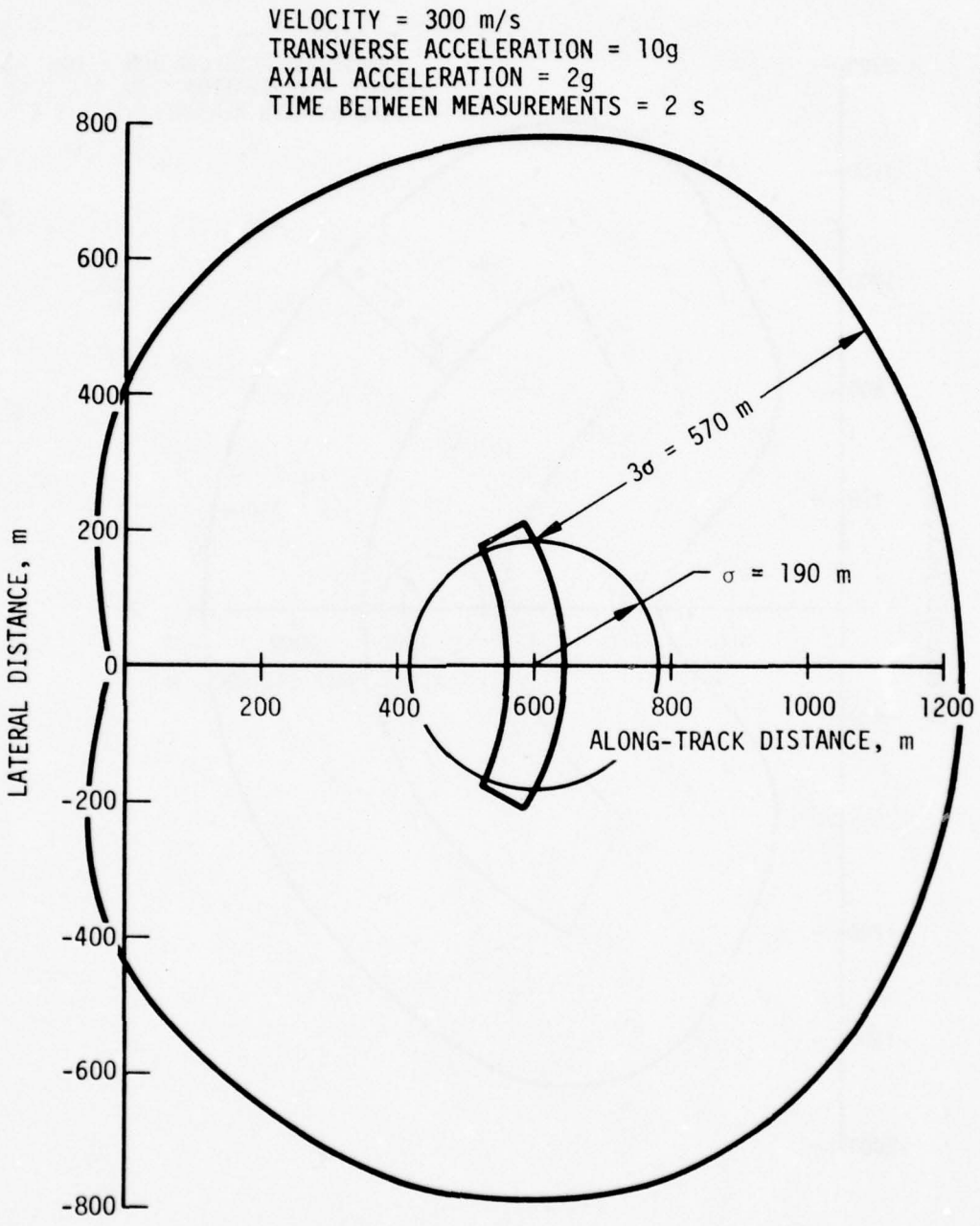


Figure 2.3.15. Association Volume for Track Association

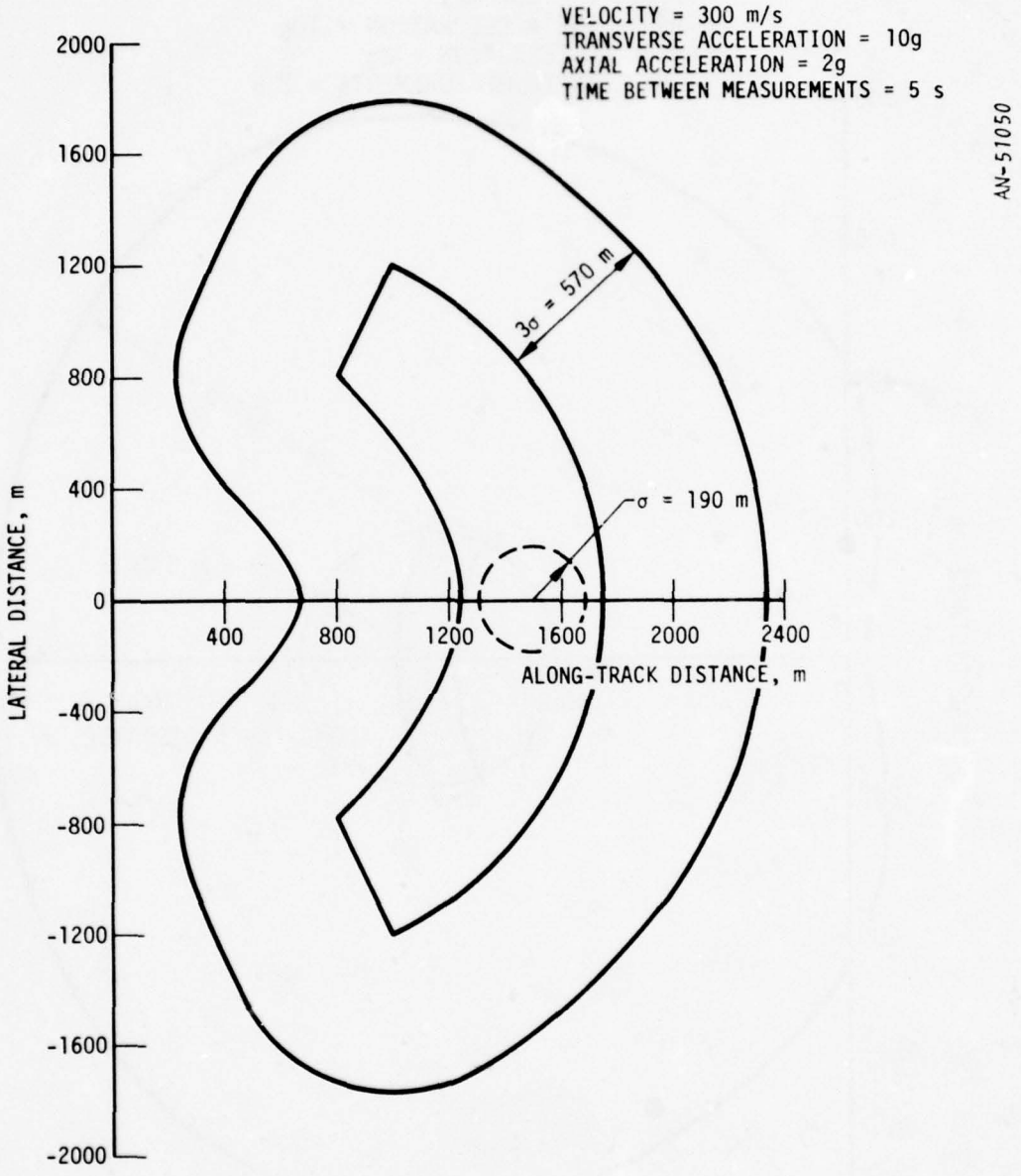


Figure 2.3.16. Association Volume for Track Association

2.3.17. In this case, $\sigma_p = 2.9\sigma_m$ for an extrapolation time of 2.5 measurement intervals, $\sigma_u = 3.1\sigma_m = 310$ m for $\sigma_m = 100$ m, and $3\sigma_u = 930$ m. This association volume is still larger than the previous one but somewhat closer to a circle in shape.

2.3.2.3 Model Error Effects

A quadratic function (rather than the usual linear function) is a reasonable model of aircraft flight in the present air surveillance context where it is desirable to be able to extrapolate some time ahead without having to update the System Track. Even when aircraft are performing constant-acceleration maneuvers, they do not follow flight paths that are quadratic functions, and predictions of their position based on the quadratic model are in error because of this fact. This model error is in addition to the errors due to target maneuvers and measurement noise which were discussed in the preceding sections.

An aircraft performing a constant-acceleration maneuver follows a circular path (assuming the speed remains constant). A circle can be closely approximated by a quadratic function over short distances, but diverges substantially at longer distances. In order to obtain an indication of the magnitude of the model error, an analysis of the case of a quadratic-fit to a circular path was carried out. The quadratic was fitted to three equally spaced points on the circle with the distance between the points as a parameter. The distance between the circle and the quadratic fit one sample time after the last measurement point was then determined, as was the distance between the tangent to the circle and the tangent to the quadratic (a quantity of interest in determining the required size of the association volume). The geometry of these curves and their extrapolations are shown in Fig. 2.3.18.

The distance between the target position on the constant-acceleration circle and the quadratic projection (ε_{mq}) and the distance between the tangents of the two curves ($\varepsilon_{m\ell}$) are functions of the angle θ between measurement points, the target velocity v, and the acceleration a. The

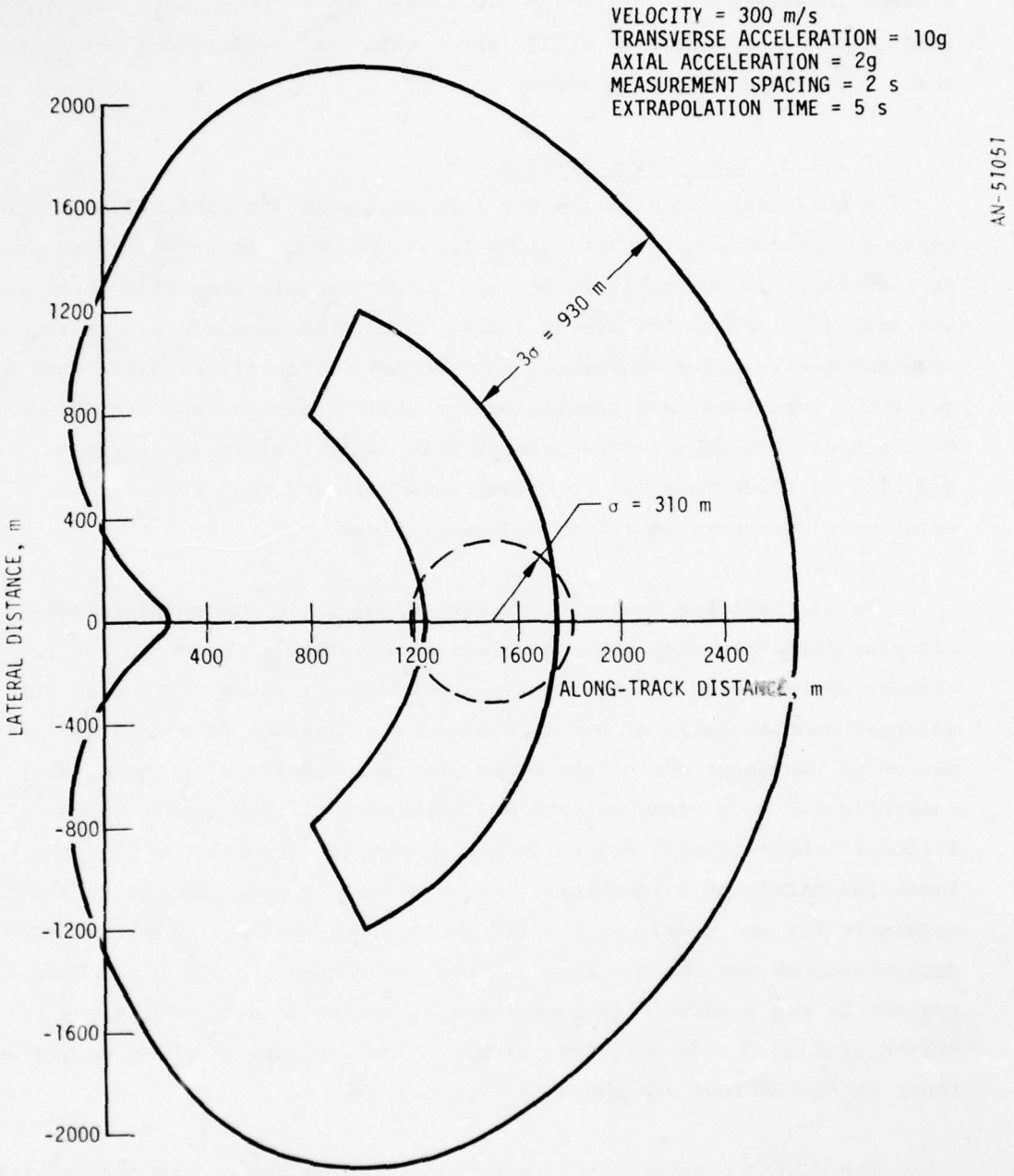


Figure 2.3.17. Association Volume for Track Association

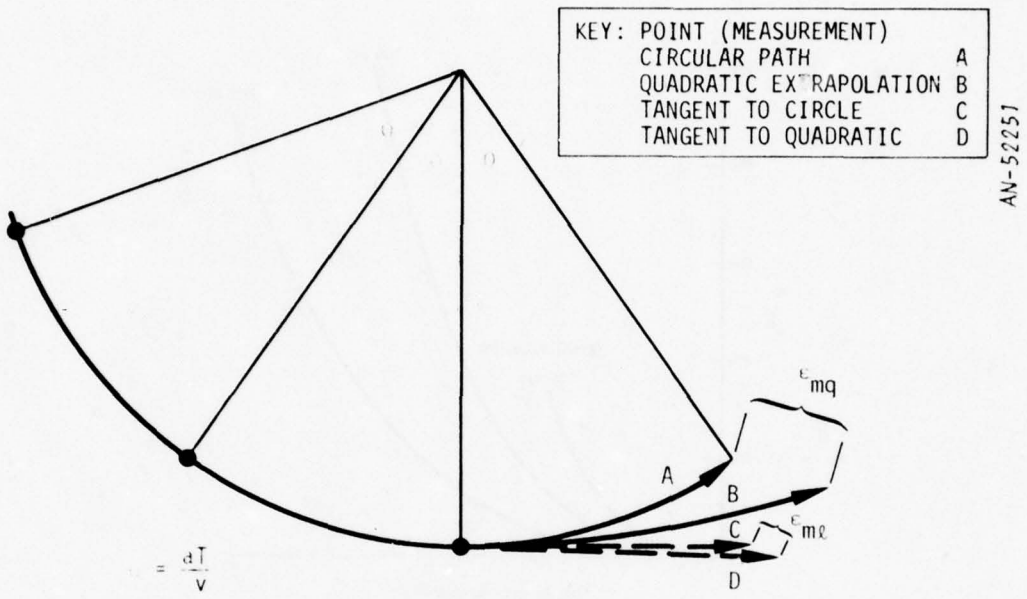


Figure 2.3.18. Quadratic Fit to Three Points on a Circular Path and the Extrapolation of These Curves and Their Tangents

---

sample time T between measurements is related to these parameters by $T = v\theta/a$. Both types of model errors were evaluated and plotted for several combinations of parameter values.

The distance between actual and predicted target positions (ε_{mq}) for an extrapolation time equal to the time between measurements is plotted versus the time between measurements in Fig. 2.3.19 for a transverse target acceleration of 3g, and target velocities of 100 m/s, 300 m/s, and 1,000 m/s. Similar plots for a target acceleration of 10g are provided in Fig. 2.3.20. In all cases, the model error increases slowly from zero at first with increasing time between measurements (and extrapolation time); then beyond some point which depends on the target acceleration and velocity, it starts to increase rapidly. These curves indicate that the model error can be a major contributor to the track-prediction error for high accelerations.

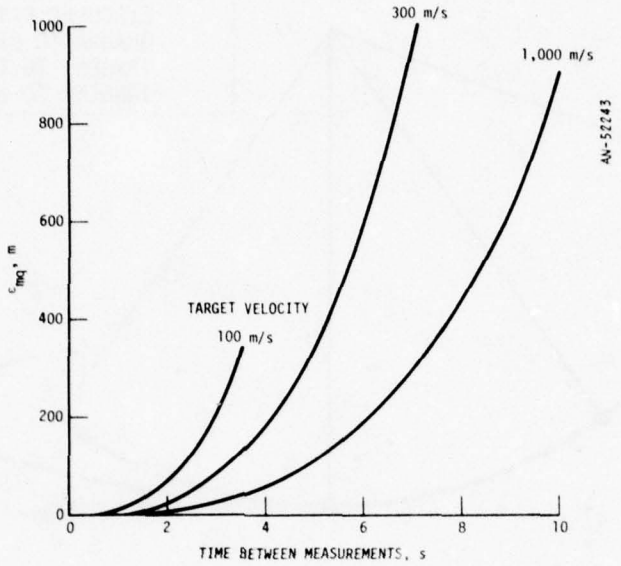


Figure 2.3.19. Model Error: Difference Between Constant 3g Acceleration Circle and Quadratic Prediction, One Sample Time Ahead

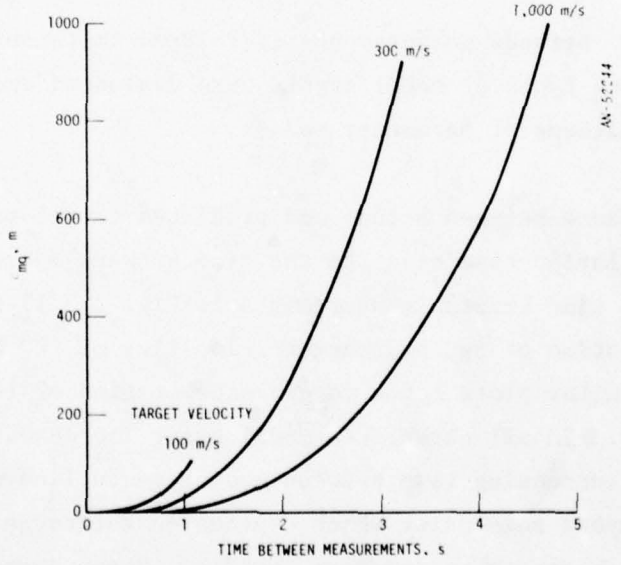


Figure 2.3.20. Model Error: Difference Between Constant 10g Acceleration Circle and Quadratic Prediction, One Sample Time Ahead

Similar plots of the distance between the tangents to the curves ($\varepsilon_{m\ell}$) are presented in Figs. 2.3.21 and 2.3.22. Again, these plots are for an extrapolation time equal to the time between measurements. These curves for the linear prediction error are similar in form to those for ε_{mq}, rising slowly at first and then much more rapidly. These results for both types of model error indicate that the time between measurements should be two seconds or so to avoid large model errors, perhaps even less under some conditions.

The magnitude of the model error as a function of the extrapolation time for a fixed time between measurements is also of interest because it gives an indication of how often the System Track must be updated. Plots of the difference ε_{mq} for a measurement interval of two seconds are shown in Figs. 2.3.23 and 2.3.24.

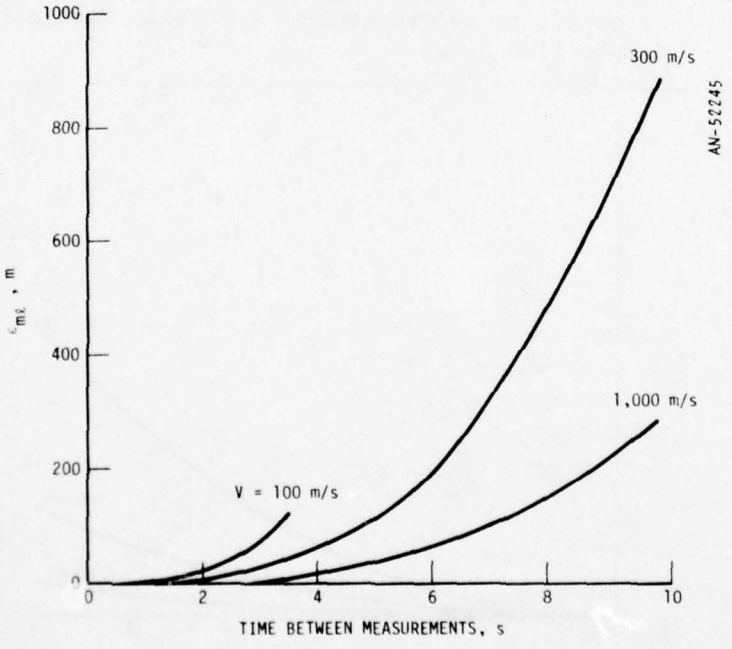


Figure 2.3.21. Model Error: Difference Between Tangent to Constant 3g Acceleration Circle and Tangent to Quadratic Prediction One Sample Time Ahead

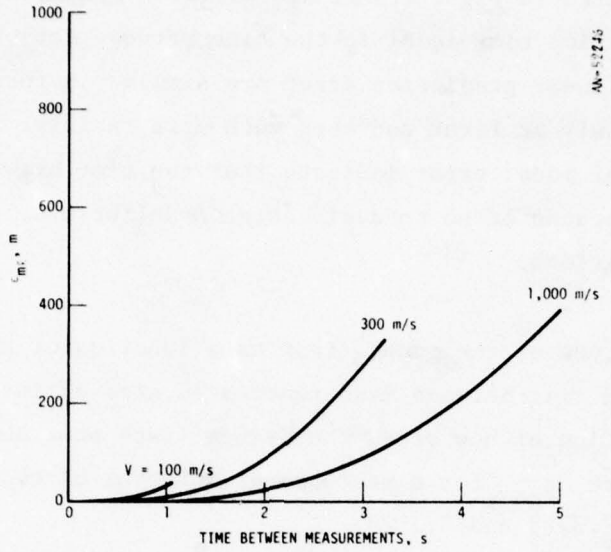


Figure 2.3.22. Model Error: Difference Between Tangent to 10g Acceleration Circle and Tangent to Quadratic Prediction One Sample Time Ahead

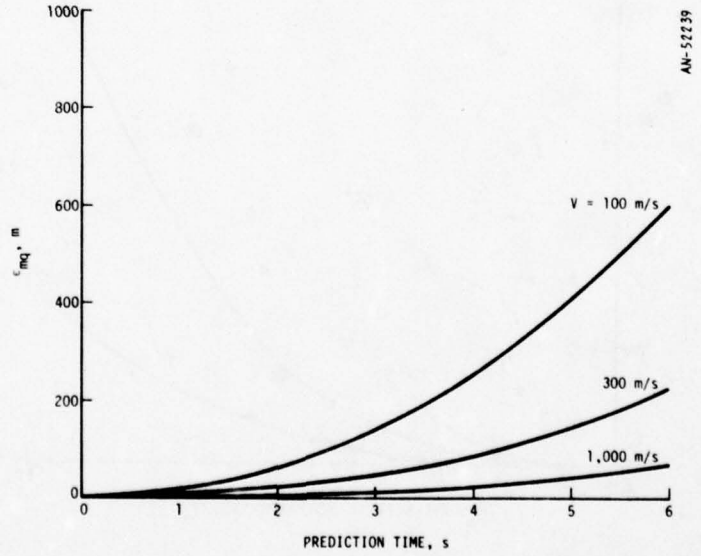


Figure 2.3.23. Model Error: Difference Between Constant 3g Acceleration Circle and Quadratic Prediction for a Time Between Measurements of 2 Seconds

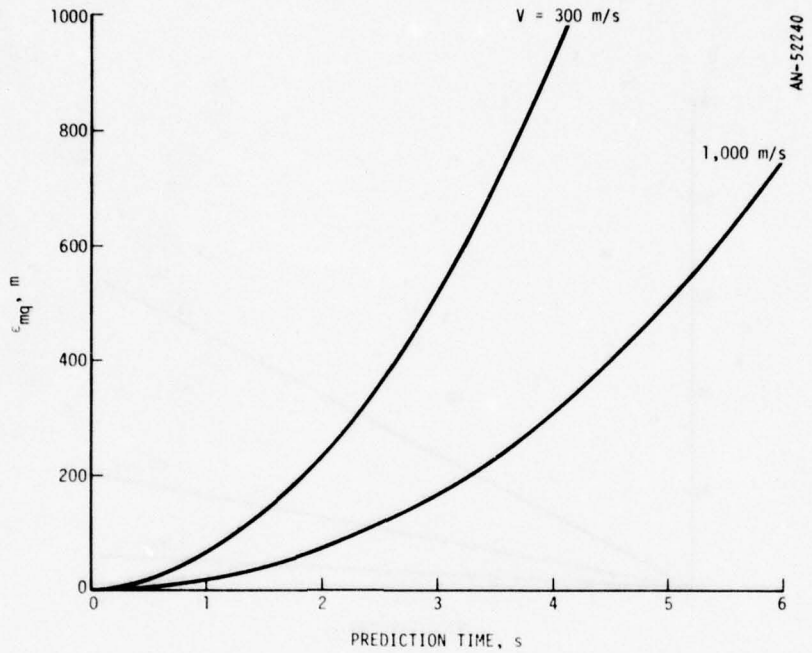


Figure 2.3.24. Model Error: Difference Between Constant 10g Acceleration Circle and Quadratic Prediction for a Time Between Measurements of 2 Seconds

---

Similar sets of curves for the distance $\varepsilon_{m\ell}$ between the linear projections (tangents) as functions of the extrapolation time for a measurement interval of two seconds are plotted in Figs. 2.3.25 and 2.3.26. Since the distance between the tangents increases linearly with time, these curves are straight lines. This type of model error is significant only for high target accelerations for extrapolation times up to several measurement intervals.

2.3.3 Association Performance Analysis

To gain insight into the relationship between parameters describing the association performance and the radar and target parameters, the association process was analyzed using idealized models and considering special cases. These analyses and their results are presented in this section. First the probability of associating a track with a false alarm is derived and evaluated for sets of parameter values of interest, then the

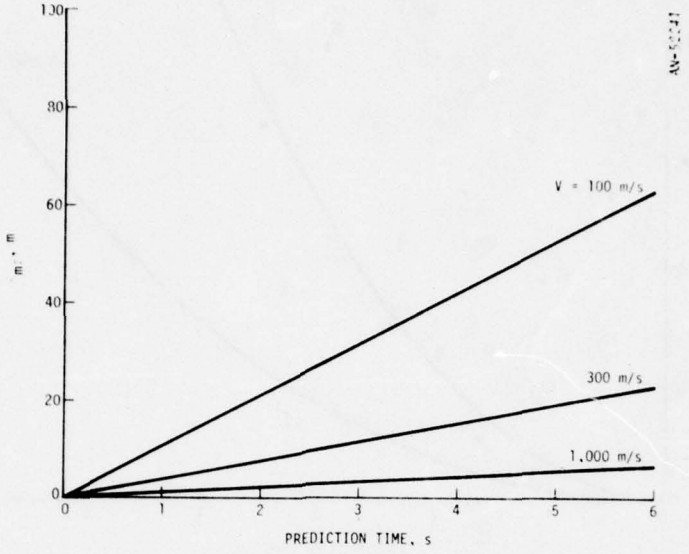


Figure 2.3.25. Model Error: Difference Between Tangent to Constant 3g Acceleration Circle and Tangent to Quadratic Prediction for a Time Between Measurements of 2 Seconds

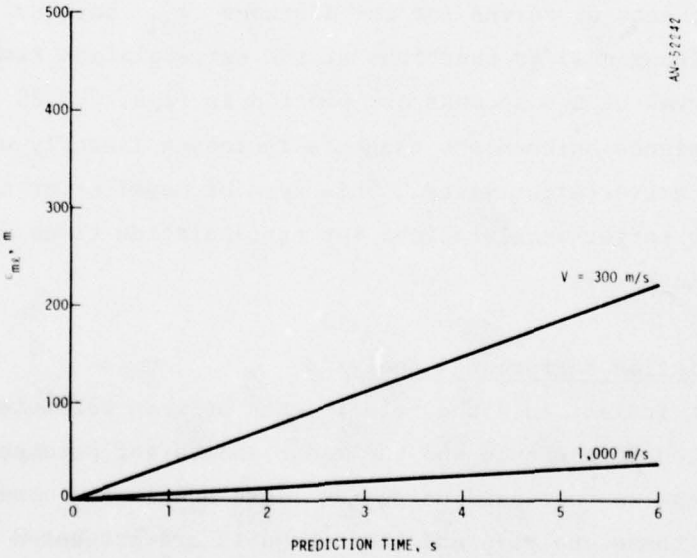


Figure 2.3.26. Model Error: Difference Between Tangent to Constant 10g Acceleration Circle and Tangent to Quadratic Prediction for a Time Between Measurements of 2 Seconds

probabilities of incorrect association of closely spaced targets are determined for certain simple target geometries.

2.3.3.1 False Alarm Associations

If a false alarm occurs in a location close to a projected track, it may be associated with the track. The probability of a false-alarm association is the probability that the distance from the predicted target position to the false alarm location is less than the distance from the predicted target position to the measurement location. With an isolated track (no other nearly targets), it appears that any reasonable algorithm will associate the false alarm with the track if, and only if, it is closer than the target to the track extrapolation.

To evaluate the probability of a false-alarm association, it is first necessary to determine the probability density function of the measured target position relative to the predicted target position. As discussed in the preceding section, this distance is made up of three components: (1) the acceleration displacement, (2) the measurement error, and (3) the model error. The first and third of these can be considered to be bias errors whose magnitude and direction depend on the particular maneuver that the aircraft performs. The measurement error, on the other hand, is random in nature since it is produced by random noise. As a mathematical model of this situation, it is assumed that the distance of the measurement from the track prediction (assumed to be at the origin) consists of a bias term of magnitude A which is equally likely to lie in any direction, plus a random variable whose x, y, and z components are Gaussianly distributed with zero mean and standard deviation σ. This geometry is depicted in Fig. 2.3.27. It can be shown that the probability density function of the distance of a measurement from the origin is given by

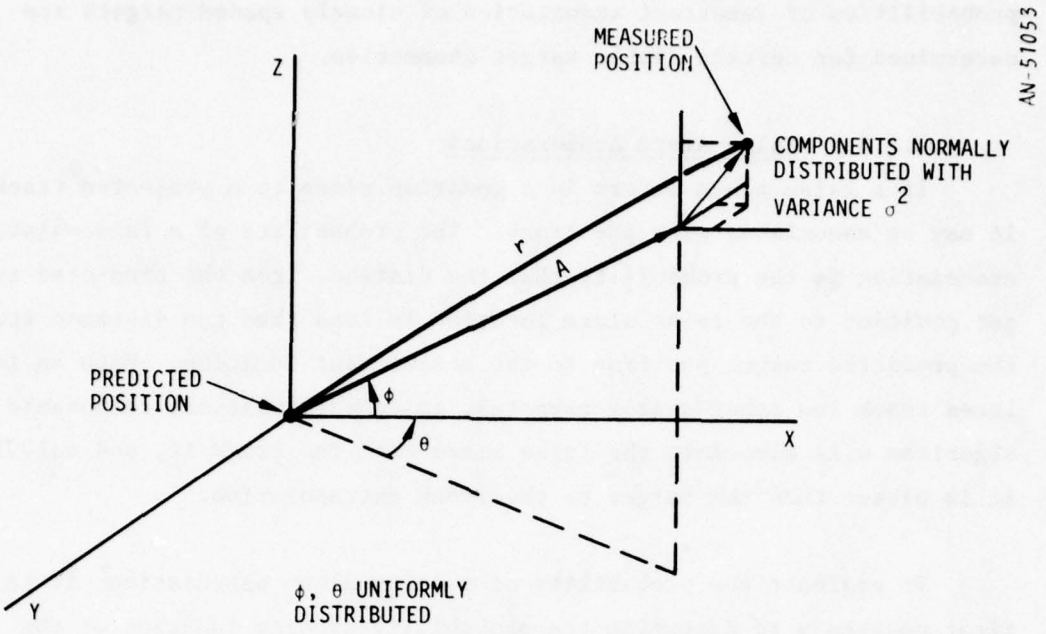


Figure 2.3.27. Error Distribution Model

$$p(r) = \frac{r}{\sqrt{2\pi}\sigma A} e^{-(r^2+A^2)/2\sigma^2}\left[e^{rA/\sigma^2} - e^{-rA/\sigma^2}\right] \qquad (2.3\text{-}5)$$

The probability that r is less than or equal to a specified value R is then

$$p(r \leq R) = \int_0^R p(r)\, dr = \frac{1}{\sqrt{2\pi}A}\left[e^{-(A+R)^2/2\sigma^2} - e^{-(A-R)^2/2\sigma^2}\right]$$

$$+ f_1\left(\frac{A+R}{\sigma}\right) - f_1\left(\frac{A-R}{\sigma}\right)$$

where

$$f_1(X) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{X} e^{-x^2/2}\, dx$$

is a tabulated function. This cumulative probability distribution function is plotted in Fig. 2.3.28 for $\sigma = 200$ m (a rounded-off approximation to the 190 m that was obtained in Sec. 2.3.2.2 as a reasonable value for the combined track-prediction and measurement errors) and three values of A . These values of the bias term are related to possible combinations of acceleration and prediction time in the figure, assuming that model error is zero. For $A = 0$, r is less than 600 m (the $3\text{-}\sigma$ value) with a probability of 0.97%. It is seen that small biases have little effect on the distribution function but that larger biases shift the curves substantially and change their shape somewhat.

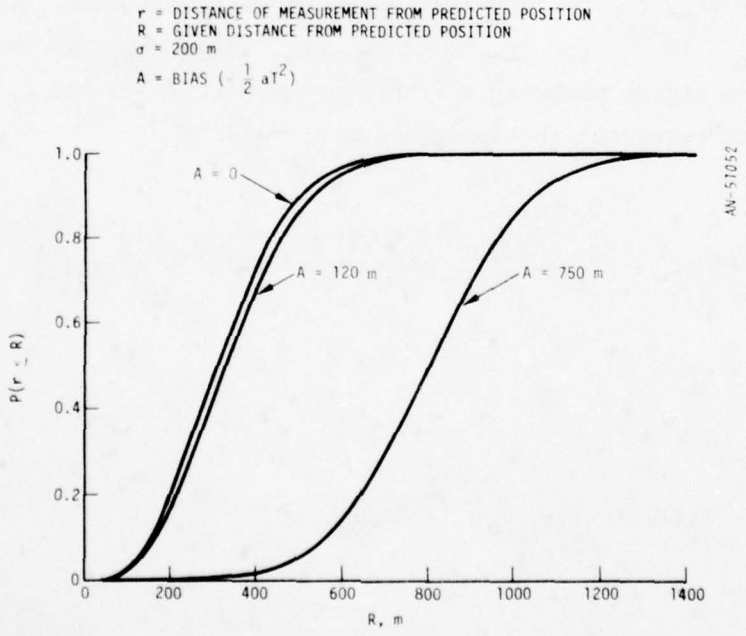


Figure 2.3.28. Probability That a Position Measurement is Less Than a Given Distance From its Predicted Location

Using the probability density function $p(r)$ from Eq. 2.3-5, the probability of association with a false alarm--i.e., the probability that a false alarm occurs at a distance from the predicted target position that is less than the distance to the measurement--can be written as

$$P(FA) = \int_0^\infty p_1(FA/r)p(r)\, dr \tag{2.3-6}$$

where $p_1(FA/r)$ is the probability of association with a false alarm given that the measurement is at a distance r from the predicted target position. Assuming that the false alarms are uniformly distributed throughout the coverage volume, this latter probability is just

$$p_1(FA/r) = Kr^3 \tag{2.3-7}$$

for $Kr^3 << 1$. For the coverage volume specified in Sec. 2.2.2, $K = 1.07N_{FA} \times 10^{-14}$ for r in kilometers, where N_{FA} is the number of random false alarms per scan. Substituting Eqs. 2.3-5 and 2.3-7 in Eq. 2.3-6 and carrying out the integration gives

$$P(FA) = \frac{2K}{\sqrt{2\pi}} (5\sigma^3 + A^2\sigma)e^{-A^2/2\sigma^2} + \frac{K}{A} (3\sigma^4 + 6A^2\sigma^2 + A^4) f_2\left(\frac{A}{\sigma}\right) \tag{2.3-8}$$

where

$$f_2(X) = \frac{1}{\sqrt{2\pi}} \int_{-X}^{X} e^{-x^2/2}\, dx$$

is a tabulated function.

The probability of a false-alarm association per scan as given by Eq. 2.3-8, which is equal to the expected number of false-alarm associations per scan for $P(FA) << 1$, was evaluated for several values of σ and A. The value of A used in these calculations was taken to be equal to $\frac{1}{2} aT^2$ where a is the target acceleration and T is the time since the last measurement (normally equal to the time between measurements).[1] A maximum target acceleration of 6g was used because for any value of T the volume of a sphere of that radius is approximately equal to the uncertainty volume with a 10g transverse acceleration and a 2g axial acceleration as plotted in Fig. 2.3.10. The expected number of false-alarm associations with each track on each scan ($\bar{N}_{FA}$/SCAN) is plotted as a function of the time between measurements in Fig. 2.3.29 for σ = 200 m, a = 6g, and N_F = 100 false alarms per scan. The straight-line asymptotic for $a = 0$ and $\sigma = 0$ are also plotted. In the latter case $\bar{N}_{FA}$/SCAN varies as T^6 since A varies as T^2 and the probability of false-alarm association is proportional to A^3. The curve for σ = 200 m and a = 6g is nearly flat out to a time between measurements of perhaps three seconds, then it starts to rise rather rapidly. For the shorter times, the expected number of false-alarm associations is small: with 1,000 targets in track, the total for all the targets is less than 0.1 per scan.

The expected number of false-alarm associations from Eq. 2.3-8 is plotted in Fig. 2.3.30 for other values of σ and a to provide an indication of the sensitivity to these parameters. The shape of these curves is similar to that of the curve in Fig. 2.3.29 for the nominal values of σ and a. These results indicate that the time between tracking measurements should be less than about three seconds to minimize the number of associations with false alarms.

---

[1] The contribution of the model error to the bias term was not included in these calculations. This error had not been evaluated when the calculations were made and it was assumed to be negligible, as it indeed appears to be in some, but not all, cases based on the results in Sec. 2.3.2.3. Other models of the target trajectory could produce errors which are, in fact, small in all cases of interest.

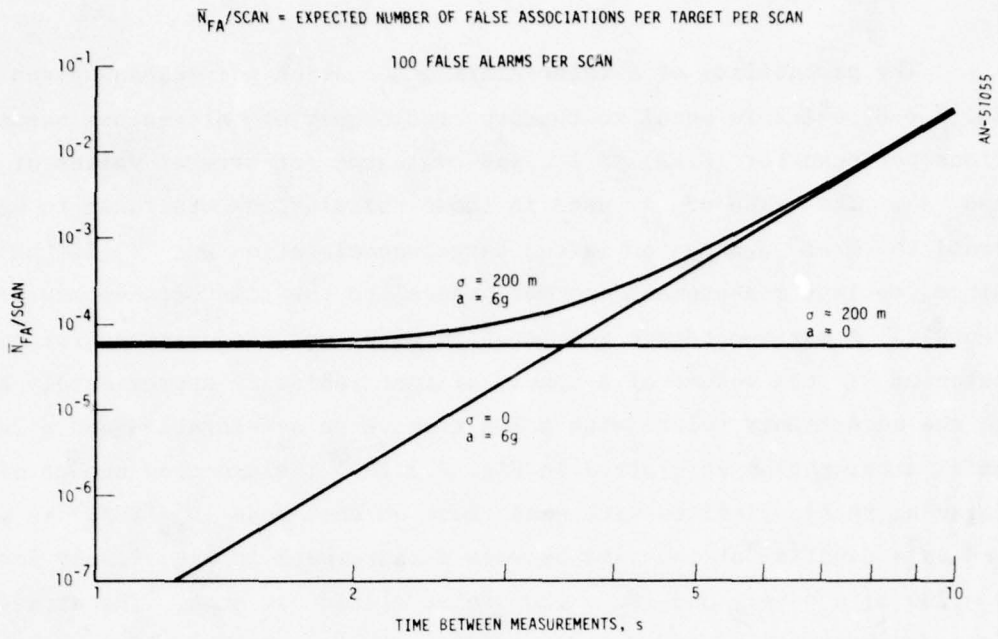


Figure 2.3.29. False Associations With Random Noise

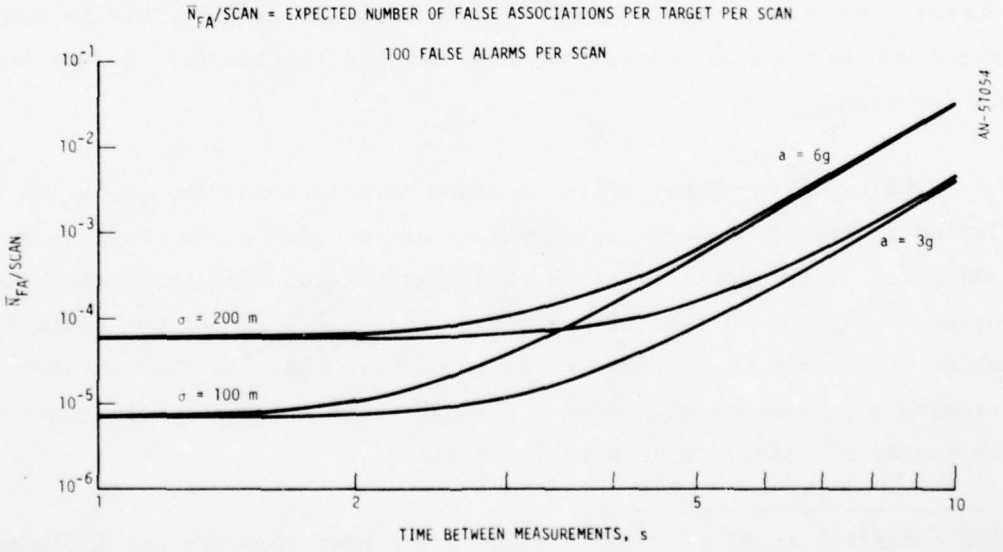


Figure 2.3.30. False Associations With Random Noise

For some purposes, the expected number of false-alarm associations per second, rather than the number per scan, may be of interest. The number per second can be obtained from Figs. 2.3.29 and 2.3.30 simply by dividing by the scan time. If the time between tracking measurements and the scan time are related as in a track-while-scan system, then each value on the curves in Fig. 2.3.30, for example, can be divided by the time between measurements to obtain the curves shown in Fig. 2.3.31. In this case, there is a time between measurements which minimizes the expected number of false-alarm associations per second; its value is in the range of two to three seconds for the parameter values considered.

2.3.3.2 Multiple-Target Associations

With many targets in track, it is possible to incorrectly associate a measurement from one target with the track on another target. The determination of the frequency with which such false associations occur and its dependence on the radar and target parameters is certainly a

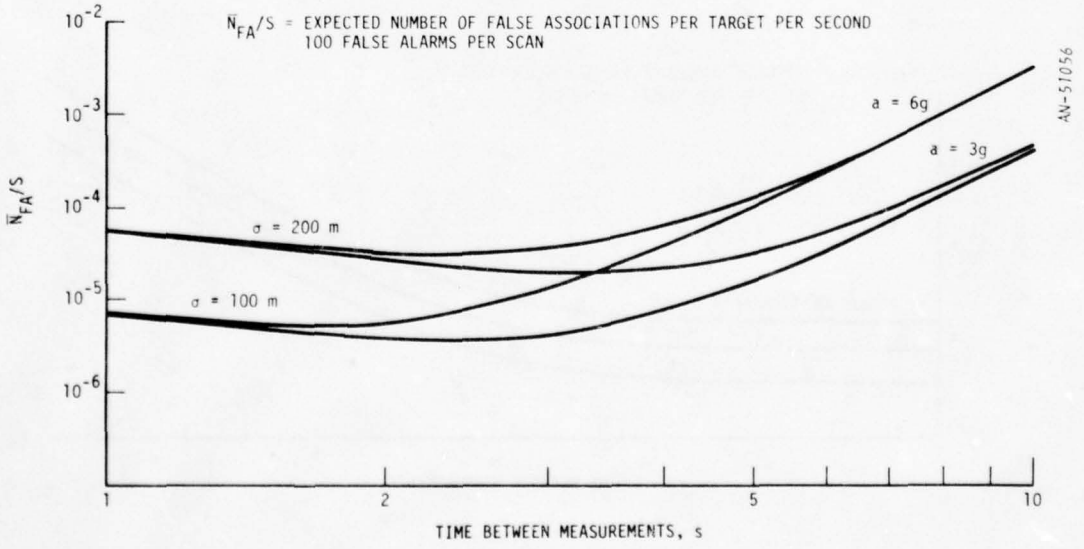


Figure 2.3.31. False Associations With Random Noise

AD-A067 936 GENERAL RESEARCH CORP SANTA BARBARA CALIF F/G 17/9
TACTICAL FORWARD AREA SURVEILLANCE AND CONTROL INTERNETTING STU--ETC(U)
NOV 78 G W DELEY, J H BALLANTINE F19628-78-C-0020
UNCLASSIFIED CR-1-814-VOL-2 ESD-TR-78-180-VOL-2 NL

3 OF 5
ADA
067936



problem of interest. Some comments on and approaches to this problem are offered in this section along with some limited results for special cases.

If the targets are widely separated, there is of course little problem in correctly associating them with their tracks. If, for example, the targets were unformly distributed throughout the coverage volume and could be considered to affect the association process in the same way as random false alarms, then the probability of false association could be analyzed in the same way as that of the false-alarm associations in the preceding section. The resulting expected number of false associations per scan for each target in track are plotted in Fig. 2.3.32 for $\sigma = 200$ m and $a = 6g$. For a time between tracking measurements less than three seconds or so, the number of false associations is small.

Of greater interest is the situation where the local target density is considerably higher--where several aircraft are close together either

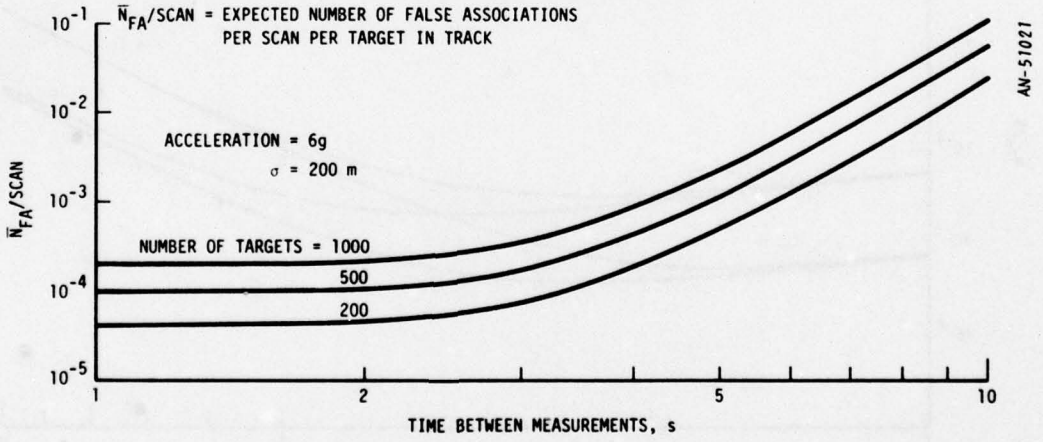
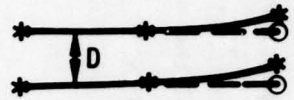
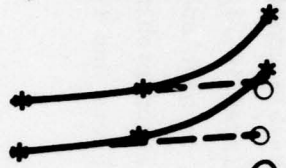
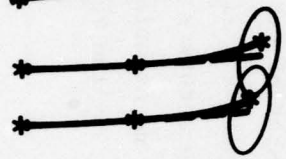
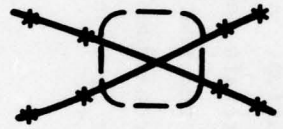
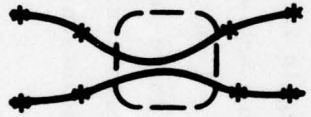
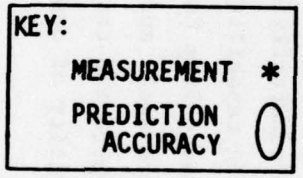


Figure 2.3.32. False Associations With Other Targets

because they are flying in formation or because they happen to be on flight paths which bring them close together for a short time. The assignment algorithm described in Sec. 2.3.1.2 is designed to handle this situation, but under certain conditions, measurements and tracks could be incorrectly associated. A general analysis of this situation is difficult and has not been attempted as part of this study; instead, simpler special cases have been considered to gain insight into the problem.

Consider the case of two closely spaced targets, for example, two aircraft flying in formation. A number of possible combinations of tracking capabilities and aircraft maneuvers are listed in Fig. 2.3.33. They range from Case 1 where the tracking accuracy is high and the maneuvers are small and there is consequently no association problem, to Case 6 where the targets cannot be resolved by the radars and there is no possibility of association.

This last case is important in that it brings out a basic requirement for tracking closely spaced targets, namely that they must be resolved by the radars. (Of course, two or more closely spaced targets that are unresolved could be tracked as a group, but the tracking accuracy would be poor and no information about the number of aircraft would be provided.) If the aircraft can maintain separations of a few tens of meters or so, this requirement for tracking leads to a requirement that the radars have high resolution in range so that at least some of the radars within range of the targets can resolve them. If a radar can resolve two targets, it can measure their positions with an error that is an order of magnitude less than the distance between the targets. Thus, it should always be possible to correctly associate the measurements with the tracks for these targets unless they have performed a maneuver since the last measurement that has produced a displacement comparable to the distance between them.

CASE	TARGETS RESOLVED	PREDICTION ACCURACY	MANEUVER	GEOMETRY	ASSOCIATION CAPABILITY
1	YES	GOOD	SMALL ($\frac{1}{2}aT^2 << D$)		NO PROBLEM
2	YES	GOOD	LARGE		DEPENDS ON ALGORITHM
3	YES	POOR	SMALL OR LARGE		DEPENDS ON ACCURACY AND ACCELERATION
4	YES-NO-YES	GOOD	SMALL		PERHAPS USING PATTERN RECOGNITION
5	YES-NO-YES	GOOD	LARGE		LIMITED
6	NO	----	----		NONE

AN-52247

Figure 2.3.33. Track Association With Closely Spaced Targets

In Cases 4 and 5 in Fig. 2.3.33, the targets pass through a region in which they are temporarily unresolved (or, alternatively, through a region in which proper association is not possible). It may be possible to maintain track using pattern recognition techniques to correctly associate the tracks before and after they become unresolved, but if the targets can perform unknown maneuvers while they are unresolved, such techniques will be of little avail. It may, however, be of little value to distinguish between the aircraft, particularly if they are of the same type, in which case before-and-after track association may not be necessary.

Case 3 represents a situation that could occur only in certain systems or under certain circumstances. It might occur if for some reason no tracking measurement has been made for some time, so that the prediction error has become large even though the targets can be resolved and the measurement accuracy is high. Or it might occur in a track-while-scan system with cooperative tracking by several radars with high range resolution and accuracy, but relatively poor angular resolution. If the current measurement has been preceded by several in which the radar beam was nearly orthogonal to that of the current measurement, then the track prediction accuracy could be poor in the current range direction. In any event, it is assumed that the prediction accuracy is poor, but the measurement accuracy is good in one direction, making the association problem essentially one-dimensional in this case. Assuming that the measurement error is Gaussianly distributed and that the closest-pair algorithm is used, it can be shown that the probability of false association (i.e., of associating a measurement with the wrong track) is given by

$$P(FA) = \frac{1}{2}\left[1 + f_1\left(\frac{A - D/2}{\sigma}\right) - f_1\left(\frac{A + D/2}{\sigma}\right)\right] \qquad (2.3\text{-}9)$$

where

$$f_1(X) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{X} e^{-x^2/2}\, dx$$

and D is the target separation, A is the displacement of both targets due, for example, to target acceleration and equally likely to be in either direction. The probability of false association as given by Eq. 2.3-9 is plotted as a function of the target separation in Fig. 2.3.34

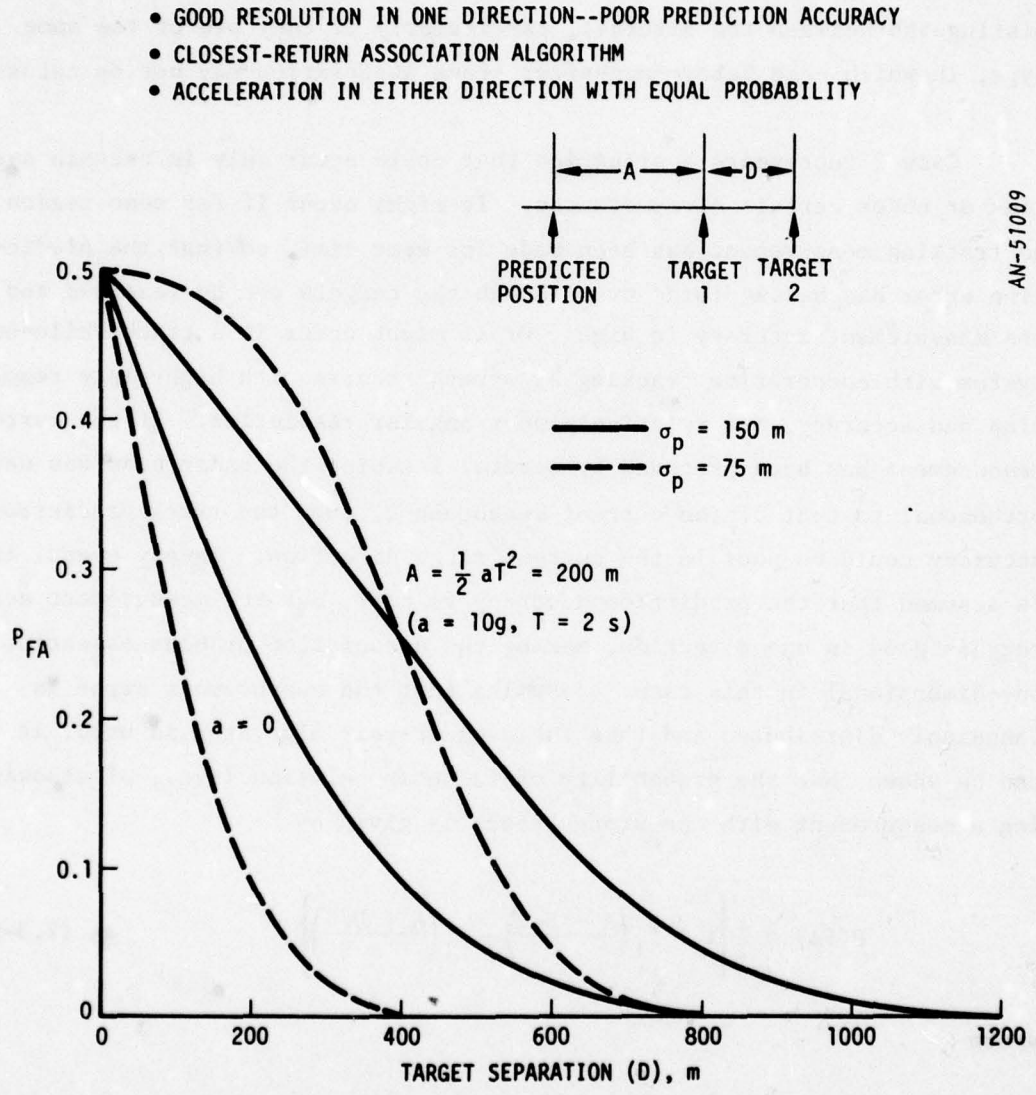


Figure 2.3.34. Probability of False Association--Closely Spaced Targets

for $\sigma_p = 150$ m (the prediction error for $\sigma = 100$ m with no contribution from the current measurement) and $\sigma_p = 75$ m, and for $A = 0$ and $A = 200$ (e.g., $\frac{1}{2} aT^2$ for $a = 10g$ and $T = 2$ seconds). These curves indicate that the probability of false association in this case remains substantial for target separations out to several hundred meters.

Case 2 in Fig. 2.3.33 is one in which the measurement and prediction accuracies are high but the displacement since the last measurement is larger than the target separation. This case is one in which the assignment algorithm can make the correct association while the closest-pair algorithm does not. Consider the situation shown in Fig. 2.3.35. The closest-pair algorithm would associate measurement 2 with track 1 (and probably measurement 1 with track 1 as well). The assignment algorithm using the squares of the distances would properly associate the measurements and tracks, however, since

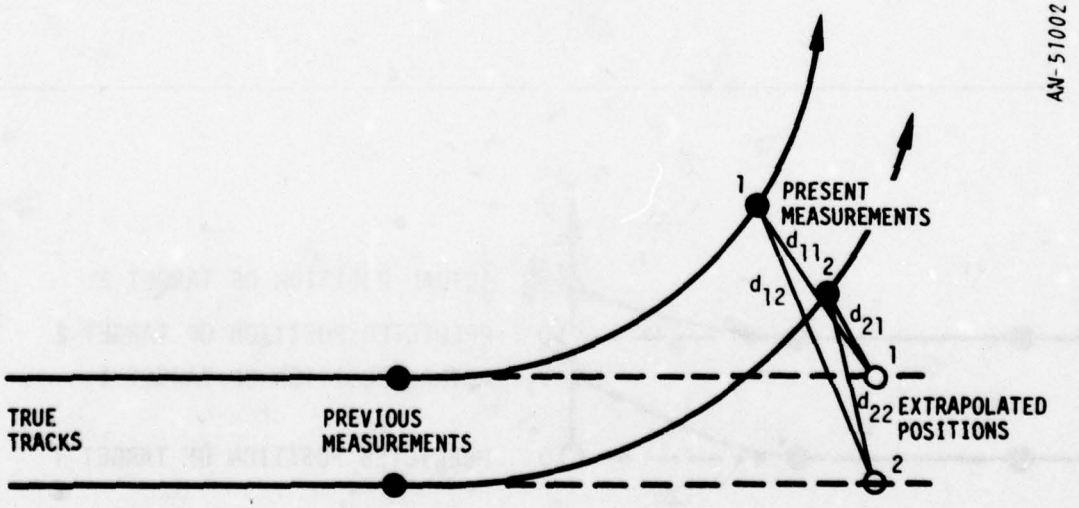


Figure 2.3.35. Maneuvering Targets Geometry

$$d_{11}^2 + d_{22}^2 < d_{12}^2 + d_{21}^2$$

Thus the association performance in this case depends in part on the algorithm that is used.

The association performance for closely spaced targets also depends on the target maneuver capability. To obtain an indication of the effect of maneuvers in a simple case, consider the two-target geometry sketched in Fig. 2.3.36. The two targets are separated by a distance D at the last measurement and then undergo transverse accelerations in the x-y plane. Assume that at the time of the next measurement the targets lie along the y axis but are displaced by amounts $y_1 = \frac{1}{2} a_1 T^2$ and $(y_2 - D) = \frac{1}{2} a_2 T^2$ from their predicted positions. Assume further that position measurement errors are small compared with the distance D between the targets (a reasonable assumption if the targets are resolved in the y direction by the radar). Using the assignment algorithm with the squares of the distances, the measurements will be correctly associated with the tracks if

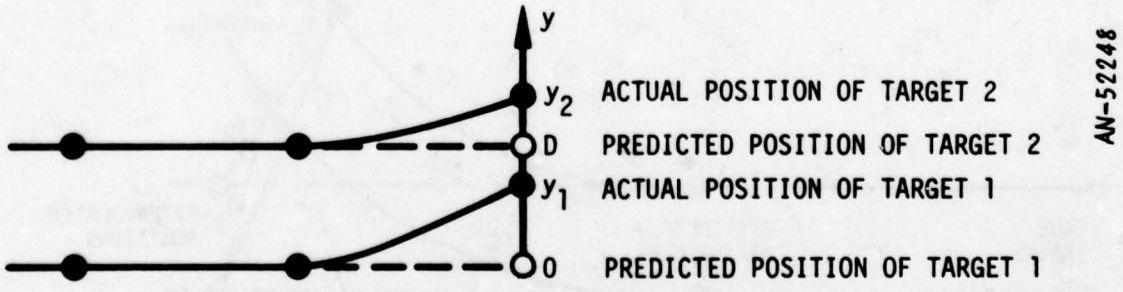


Figure 2.3.36. Target Geometry for One-Dimensional Association Analysis

$$y_1^2 + (y_2 - D)^2 < y_2^2 + (y_2 - D)^2$$

From this inequality it follows that incorrect association will occur if $y_2 < y_1$ or if

$$y_d = y_2 - y_1 < 0 \qquad (2.3\text{-}10)$$

(It is of interest to note that if linear distances rather than distances-squared were used in the assignment algorithm, the measures of association for the two possible target-track combinations--$y_1 + (y_2 - D)$ and $y_2 + (y_1 - D)$--would always be equal, and the pairings would presumably be made randomly with a probability of correct association of 0.5.)

To determine the probability of incorrect association using Eq. 2.3-10, assume that the acceleration a of each target can be described statistically by the probability density functions shown in Fig. 2.3.37--impulses at 0 and at the maximum attainable accelerations of a_m and $-a_m$ with an additional component uniformly distributed between the maximum values--and that these accelerations are uncorrelated. Then the probability density function of y_d is the convolution of $p(y_2)$ and $p(-y_1)$, where $y_1 = \frac{1}{2} a_1 T^2$ and $y_2 = \frac{1}{2} a_2 T^2 + D_1$ and has the form shown

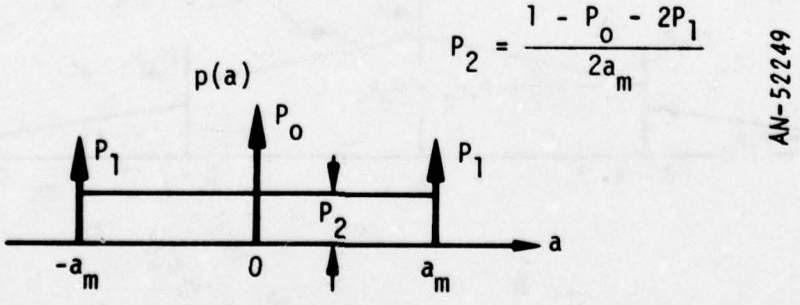


Figure 2.3.37. Assumed Probability Density Function of Target Acceleration

in Fig. 2.3.38 where $A = \frac{1}{2} a_m T^2$. From Eq. 2.3-10, the probability of an incorrect or false association is

$$P(FA) = \int_{-\infty}^{0} p(y_d)\, dy_d \qquad (2.3\text{-}11)$$

This integral was evaluated for several sets of values of the probability-function parameters to obtain the probabilities of false association plotted in Fig. 2.3.39 as functions of the target separation D relative to the maximum target displacement $A = \frac{1}{2} a_m T^2$. In all cases this probability decreases with increasing D , with step changes at $D = A$ and at $D = 2A$ where it falls to zero. [For cases in which $P_2 \neq 0$, the P(FA) plots would include segments of quadratic functions as well.] Note that the maximum probability of false association for D between 0 and A does not occur for $P_1 = 0.5$ as might be expected, but rather for an intermediate value of P_1 . The probability of false association is plotted in Fig. 2.3.40 as a function of the time T between measurements for $D = 100$ m and several sets of values of the

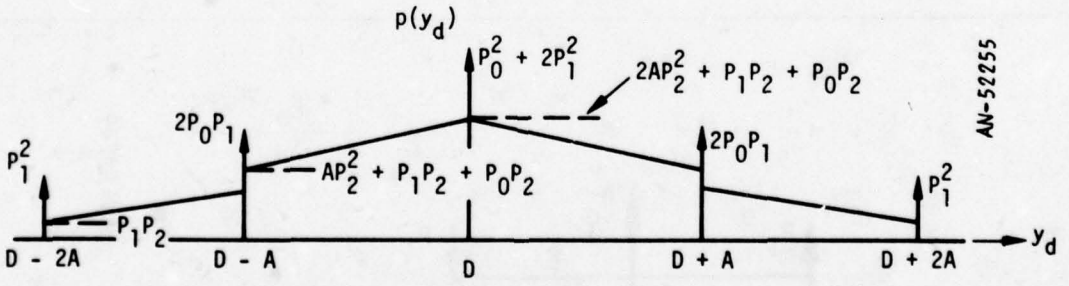


Figure 2.3.38. Probability Density Function of $y_d = y_2 - y_1$

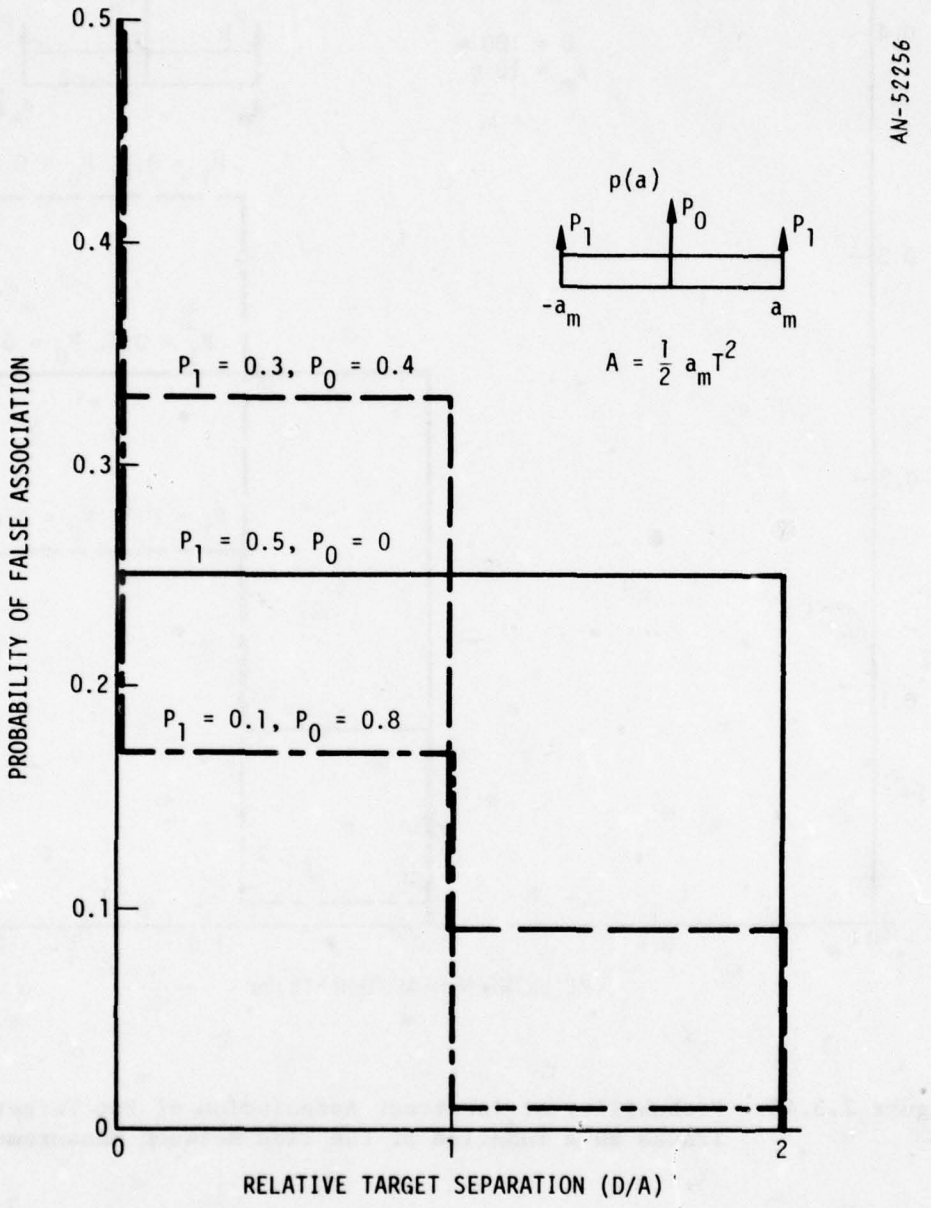


Figure 2.3.39. Probability of Incorrect Association of Two Targets and Tracks as a Function of the Target Separation

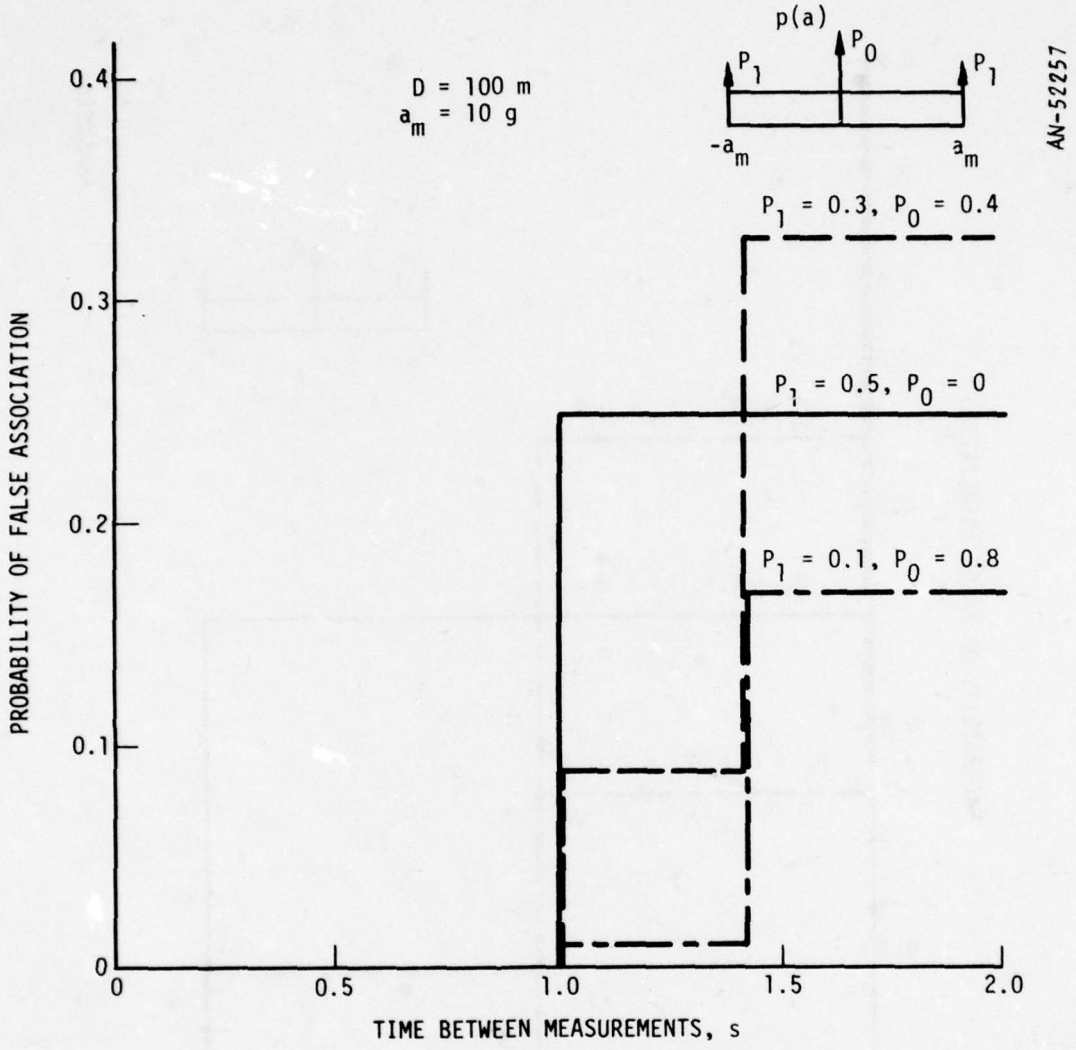


Figure 2.3.40. Probability of Incorrect Association of Two Targets and Tracks as a Function of the Time Between Measurements

probability-function parameters. This probability increases rapidly for T greater than one second, reaching its maximum value at $T = \sqrt{2}$ seconds.

These results reinforce the need for a high tracking rate to maintain the association performance at a high level. However, the problem of associating multiple targets and tracks is a complex one, and further analysis is needed to establish general requirements for various combinations of target spacings, maneuver capabilities, and measurement accuracies for the assignment algorithm and perhaps for other algorithms of interest.

2.4 TRACK FILTERING

The principal objective of the surveillance systems studied is to keep track of targets; very accurate track of the aircraft is not a requirement at this time nor is it foreseen to be in the future. Thus the requirements on track filtering are not as exacting as the requirements on association.

As discussed in Sec. 1.1, the principal output of the system is a file of System Tracks on all aircraft which is available to several or all nodes in the system. To lower communications requirements on the system, it is desired to update the System Tracks as infrequently as possible. This means that there is an advantage in estimating acceleration as well as position and velocity if the acceleration estimate is not too noisy.

Two types of filters were considered: recursive and non-recursive. A recursive filter is one in which past measurements are summarized in the present estimate of the state of the object being tracked (i.e., position, velocity and perhaps acceleration). A new measurement is combined with the state estimate extrapolated to the time of the measurement to obtain a new state estimate. The method of combining the data can be complex, as with the Kalman filter, or simple as with the α-β filter. Other filters, such as the Weiner filter, are intermediate in complexity.

The least-squares filter is an example of a non-recursive filter. In this case a curve is passed through a number of past measurements (which must be saved) so that the sum of the squared errors between the measurements and the curve is minimized.

In the following sections, the Kalman, α-β , and weighted least squares filters are discussed. Additional discussions of these filters appears in the simulation sections (Secs. 3.3, 3.4, and 3.5).

All of the filters considered use Cartesian coordinates (although initially during the study radar coordinates were considered). This is primarily because a common coordinate system is required for the System Track, and Cartesian coordinates are well suited as a common coordinate system. Also, the equations of motion of an aircraft are simple to write in Cartesian coordinates, especially if the aircraft is flying in a straight line, a common case. A straight line in radar (polar) coordinates is complex; the simplifications that are usually employed to avoid this complexity are an additional source of error. In all cases it was therefore decided to immediately transform measurements to Cartesian coordinates and work only in that system.

2.4.1 Kalman Filter

The Kalman filter is the optimal filter for linear systems perturbed by Gaussian noise for several reasonable optimization criteria such as minimum mean square error, maximum likelihood, and minimal variance Bayes' estimate.[1] The extended Kalman filter has proven useful in many non-linear problems as well, and has been widely studied for use in tracking maneuvering aircraft (see Bibliography, p. 245). Several versions of the extended Kalman filter were considered during the study, and one was implemented in the TACRAN2 simulation (Sec. 3.4).

Filter Variations. Several choices exist when selecting a Kalman formulation of a filter for tracking maneuvering aircraft. The first is the dimensionality of the state--specifically whether acceleration is to be estimated along with position and velocity. In essentially all formulations acceleration is handled as a random input. Acceleration may be correlated or uncorrelated between samples. Correlated inputs add accuracy but also augment (i.e., increase the size of) the state vector.[2]

---

[1] A. Gelb et al., Applied Optimal Estimation, M.I.T. Press, Cambridge Massachusetts, 1974.

[2] R.A. Singer, "Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-6, No. 4, July 1979, pp. 473-483.

Another option is to include association errors in the filter formulations.[1] This considerably complicates the filter.

Acceleration is usually introduced into the filter formulation as "maneuver noise." (How this is done is described later.) The filter tracks best when the aircraft maneuver and the maneuver noise are the same. However, usually a compromise is made and a single value of maneuver noise is input to cover all possible maneuvers. Such a compromise degrades performance somewhat, and several adaptive (to maneuver) filters have been designed.[2] These usually involve carrying along several filters for each aircraft in track, a computational burden.

For the present study it was decided that the simplest Kalman filter would suffice, since the accuracy required of the track is not great.

Advantages and Disadvantages. Besides the optimality of the filter, the Kalman formulation offers some advantages over simpler algorithms. Thus, for example, it provides a state error covariance matrix which can be used to properly combine tracks from separate radars into a single System Track. This same covariance matrix can also be used for statistical association between tracks. Also provided is the measurement-residual covariance matrix for use in statistical association between measurements and tracks (Sec. 2.3.1.1).

The disadvantages of the Kalman filter are (1) the large amount of data processing resources required to implement the algorithm, and (2) more importantly in the present context, the increased communications bandwidth required to transmit the state error covariance matrix.

---

[1] See Bibliography, p. 245, numbers 4-7.

[2] See Bibliography, p. 245, numbers 8, 10, and 11.

Kalman Filter Equations. The formulation of the Kalman filter used in the TACRAN2 simulation (Sec. 3.4) is described next. Using state-space notation the motion of an aircraft can be modeled as

$$\dot{\underline{x}}(t) = F\underline{x}(t) + G\underline{w}(t) \tag{2.4-1}$$

where $\underline{x}(t)$ = system state vector
$\dot{\underline{x}}(t)$ = time derivative of $\underline{x}(t)$
$\underline{w}(t)$ = random forcing function

For the problem at hand using Cartesian coordinates, Eq. 2.4-1 becomes

$$\underbrace{\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \\ \ddot{x}(t) \\ \ddot{y}(t) \\ \ddot{z}(t) \end{bmatrix}}_{\dot{\underline{x}}} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \end{bmatrix}}_{\underline{x}} + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{G} \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ a_x(t) \\ a_y(t) \\ a_z(t) \end{bmatrix}}_{\underline{w}(t)}$$

This model assumes the state consists of position and velocity (which are to be estimated) which are perturbed by the unknown and random accelerations (a_x, a_y, a_z).

Since measurements will be made at discrete times, the discrete form of Eq. 2.4-1 is required. Let $\underline{x}(k)$ be the value of the state vector at sample time t_k. Then

$$\underline{x}(k) = \Phi(k-1)\underline{x}(k-1) + \Gamma(k-1)\underline{w}(k-1) \tag{2.4-2}$$

where $\Phi(k-1)$ is the state transition matrix that takes the state at sample time t_{k-1} to time t_k, and

$$\Gamma(k-1) = \int_{t_{k-1}}^{t_k} \Phi(k-1) G \underline{w} \, d\tau$$

In the above equation $\Phi(k-1)$ is a function of τ. Equation 2.4-2 written out is

$$\begin{bmatrix} x(k) \\ y(k) \\ z(k) \\ \dot{x}(k) \\ \dot{y}(k) \\ \dot{z}(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(k-1) \\ y(k-1) \\ z(k-1) \\ \dot{x}(k-1) \\ \dot{y}(k-1) \\ \dot{z}(k-1) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & T^2/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & T^2/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & T^2/2 \\ 0 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 0 & 0 & T \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ a_x(k-1) \\ a_y(k-1) \\ a_z(k-1) \end{bmatrix}$$

$$\underline{x}(k) = \Phi(k-1)\,\underline{x}(k-1) + \Gamma(k-1)\,\underline{w}(k-1)$$

where $T = t_k - t_{k-1}$. The random disturbances a_x, a_y, a_z are assumed to be constant during the interval (t_k, t_{k-1}).

The measurement model is given by

$$\underline{z}(k) = h\left(\underline{x}(k)\right) + \underline{v}(k)$$

where $\underline{z}(k)$ = measurement vector (in radar coordinates)

$h\left(\underline{x}(k)\right)$ = transformation from state (Cartesian) coordinates to measurement coordinates

$\underline{v}(k)$ = measurement noise (zero mean Gaussian)

For a three-dimensional radar,

$$\underline{z}(k) = \begin{bmatrix} r \\ A \\ E \end{bmatrix} = \begin{bmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{bmatrix}$$

For a four-dimensional radar with range-rate (Doppler)

$$\underline{z}(k) = \begin{bmatrix} r \\ A \\ E \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \\ \text{range-rate} \end{bmatrix}$$

The coordinate transformation is

$$\underline{h}(\underline{x}) = \begin{bmatrix} \sqrt{x^2 + y^2 + z^2} \\ \tan^{-1}(y/x) \\ \tan^{-1}\left(z/\sqrt{x^2 + y^2}\right) \\ (x\dot{x} + y\dot{y} + z\dot{z})/\sqrt{x^2 + y^2 + z^2} \end{bmatrix} \qquad (2.4\text{-}3)$$

The Kalman filter equations are given by

$$\hat{\underline{x}}(k|k) = \hat{\underline{x}}(k|k-1) + K(k)\tilde{\underline{z}}(k) \qquad (2.4\text{-}4)$$

where $\underline{x}(k|k)$ = minimum mean-square estimate of the state $\underline{x}(k)$ given k measurements

$\underline{x}(k|k-1)$ = minimum mean-square estimate of $\underline{x}(k)$ given $(k-1)$ measurements

$K(k)$ = Kalman gain matrix (defined later)

$\tilde{\underline{z}}(k)$ = measurement residual

and

$$\hat{\underline{x}}(k|k-1) = \Phi(k-1)\hat{\underline{x}}(k-1|k-1)$$

$$\tilde{\underline{z}}(k) = \underline{z}(k) - \underline{h}\left(\hat{\underline{x}}(k|k-1)\right)$$

Note that $\hat{\underline{x}}(k|k)$ is the filtered estimate of $\underline{x}(k)$ at time t_k, whereas $\hat{\underline{x}}(k|k-1)$ is the prediction (or extrapolation) of $\underline{x}(k)$ one sample ahead. A block diagram and summary of the Kalman filter equations are given in Fig. 2.4.1.

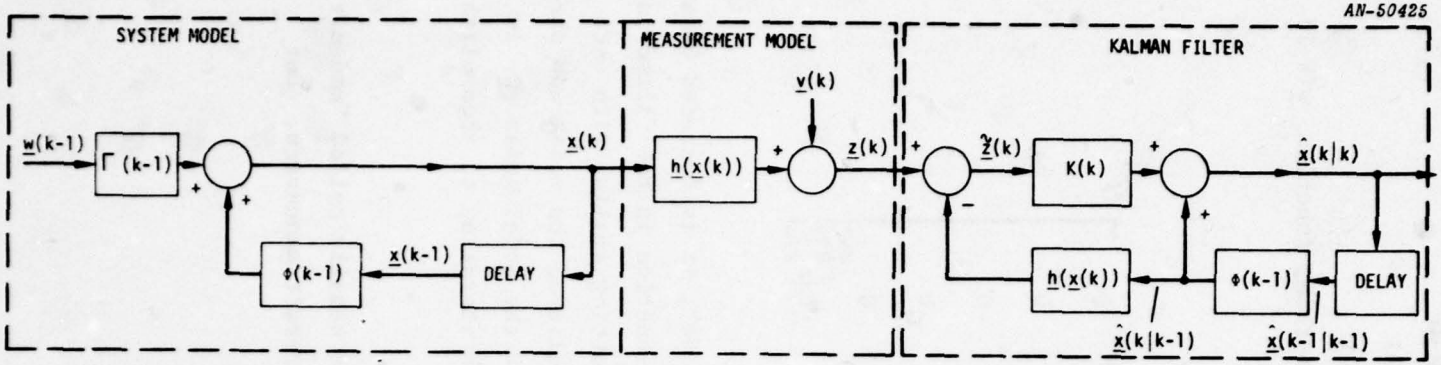
Thus far everything about the filter is defined except for the Kalman gain matrix $K(k)$. The primary difference between the Kalman filter and more simple filters is that in simple filters the gain matrix is a constant (in the Weiner filter) and uncoupled (in the α-β filter), whereas the Kalman gain matrix varies with time. Actually, most of the computation of the Kalman filter goes into the calculation of $K(k)$. Several useful by-products come out of the computation, however, including error covariances and measurement residual covariances. (In error analysis studies these "by-products" are often the whole reason for using the Kalman formulation.)

The covariance matrix of the error in the estimation is denoted by $P(k|k)$:

$$P(k|k) \equiv E\left\{\left(\underline{x}(k) - \hat{\underline{x}}(k|k)\right)\left(\underline{x}(k) - \hat{\underline{x}}(k|k)\right)^T\right\}$$

where $E\{\cdot\}$ is the expectation operator. The extrapolation of the covariance matrix from time t_{k-1} to t_k is given by

$$P(k|k-1) = \Phi(k-1)P(k-1|k-1)\Phi^T(k-1) + \Gamma(k-1)Q(k-1)\Gamma^T(k-1) \quad (2.4\text{-}5)$$



- SYSTEM MODEL

$$\underline{x}(k) = \Phi(k-1)\underline{x}(k-1) + \Gamma(k-1)\underline{w}(k-1)$$

$\underline{x}(k)$ = SYSTEM STATE AT TIME k

$\Phi(k)$ = STATE TRANSITION MATRIX

$\underline{w}(k)$ = MODEL (OR MANEUVER) NOISE $\sim N(\underline{0}, Q(k))$

$N(\underline{m}, Q)$ = NORMAL DISTRIBUTION WITH MEAN $\underline{m}$ AND COVARIANCE Q

- MEASUREMENT MODEL

$$\underline{z}(k) = h(\underline{x}(k)) + \underline{v}(k)$$

$\underline{z}(k)$ = MEASUREMENT AT TIME k

$\underline{h}(\underline{x}(k))$ = SYSTEM STATE IN MEASUREMENT COORDINATES

$\underline{v}(k)$ = MEASUREMENT NOISE $\sim N(\underline{0}, M(k))$

- FILTER EQUATIONS

$$\hat{\underline{x}}(k|k) = \hat{\underline{x}}(k|k-1) + K(k)\tilde{\underline{z}}(k)$$

$\hat{\underline{x}}(k|k)$ = ESTIMATE OF x(k) GIVEN k MEASUREMENTS

$\hat{\underline{x}}(k|k-1)$ = ESTIMATE OF $\underline{x}(k)$ GIVEN k - 1 MEASUREMENTS

$\tilde{\underline{z}}(k)$ = MEASUREMENT RESIDUAL

$K(k)$ = KALMAN GAIN MATRIX

$$\tilde{\underline{z}}(k) = \underline{z}(k) - \underline{h}(\hat{\underline{x}}(k|k-1))$$

$$\hat{\underline{x}}(k|k-1) = \Phi(k-1)\hat{\underline{x}}(k-1|k-1)$$

Figure 2.4.1. Kalman Filtering (Discrete Form)

where $P(k-1|k-1)$ = covariance of $\hat{\underline{x}}(k-1|k-1)$

$P(k|k-1)$ = covariance of $\hat{\underline{x}}(k|k-1)$

$\Phi(k-1)$ = state transition matrix

$Q(k-1)$ = covariance of random forcing function $\underline{w}(k-1)$

$$Q(k-1) = E\left\{\underline{w}(1-1)\underline{w}^T(k-1)\right\}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & E\left(a_x^2\right) & 0 & 0 \\ 0 & 0 & 0 & 0 & E\left(a_y^2\right) & 0 \\ 0 & 0 & 0 & 0 & 0 & E\left(a_z^2\right) \end{bmatrix}$$

The quantity $\underline{w}$ is often called the "model noise"; it is included because the model of the process is not perfect. Its function in the filter is to keep the error covariance matrix P from getting smaller with each measurement, which it would do if the filter believed the model was perfect. Note that in Eq. 2.4-5 the covariance of the model noise $\underline{w}$ is added to the extrapolated error covariance each iteration to accomplish this objective.

When tracking aircraft the model noise is usually called "maneuver noise," because it mainly represents random aircraft maneuvers. Let

$$E\left(a_x^2\right) = E\left(a_y^2\right) = E\left(a_z^2\right) = \sigma_m^2$$

Then it can be shown that

$$\Gamma(k-1)Q(k-1)\Gamma^T(k-1) = \sigma_m^2 \left[\begin{array}{ccc|ccc} T^4/4 & 0 & 0 & T^{3/2} & 0 & 0 \\ 0 & T^4/4 & 0 & 0 & T^3/2 & 0 \\ 0 & 0 & T^4/4 & 0 & 0 & T^3/2 \\ \hline T^3/2 & 0 & 0 & T^2 & 0 & 0 \\ 0 & T^3/2 & 0 & 0 & T^2 & 0 \\ 0 & 0 & T^3/2 & 0 & 0 & T^2 \end{array}\right] \qquad (2.4\text{-}6)$$

where $T = t_k - t_{k-1}$.

The covariance $V(k)$ of the measurement residual $\tilde{\underline{z}}(k)$ is given by

$$V(k) = H(k)P(k|k-1)H^T(k) + M(k) \qquad (2.4\text{-}7)$$

where

$$H(k) = \left[\frac{\partial \underline{h}}{\partial \underline{x}}\right] = \text{Jacobian of } \underline{h}(\underline{x})$$

$$H(k) = \left[\begin{array}{cccccc} x/r & y/r & z/r & 0 & 0 & 0 \\ -y/(x^2+y^2) & x/(x^2+y^2) & 0 & 0 & 0 & 0 \\ \dfrac{-xz}{r^2\sqrt{x^2+y^2}} & \dfrac{-yz}{r^2\sqrt{x^2+y^2}} & \dfrac{\sqrt{x^2+y^2}}{r^2} & 0 & 0 & 0 \\ \left(\dfrac{\dot{x}}{r} - \dfrac{cx}{r^3}\right) & \left(\dfrac{\dot{y}}{r} - \dfrac{cy}{r^3}\right) & \left(\dfrac{\dot{z}}{r} - \dfrac{cz}{r^3}\right) & \dfrac{x}{r} & \dfrac{y}{r} & \dfrac{z}{r} \end{array}\right] \qquad (2.4\text{-}8)$$

$$r = \sqrt{x^2 + y^2 + z^2}$$

$$c = x\dot{x} + y\dot{y} + z\dot{z}$$

and (x,y,z) are at the origin of the radar making the measurement. The last row is omitted if there is no Doppler measurement.

M(k) is the covariance matrix of the measurement (in radar coordinates):

$$M(k) = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_A^2 & 0 \\ 0 & 0 & \sigma_E^2 \end{bmatrix}$$

where σ_r^2 = variance of range measurement

σ_A^2 = variance of azimuth measurement

σ_E^2 = variance of elevation measurement

The measurement-residual covariance matrix V(k) (Eq. 2.4-7) can be used in performing a Chi-square statistical test of association, as described in Sec. 2.3.1.1.

The Kalman gain matrix K(k) can now be calculated from

$$K(k) = P^T(k|k-1)H^T(k)V^{-1}(k) \qquad (2.4\text{-}9)$$

and the error covariance matrix at sample time t_k, P(k|k) , is computed by

$$P(k|k) = [I - K(k)H(k)]P(k|k-1) \qquad (2.4\text{-}10)$$

The above covariance equations are summarized in Table 2.4.1.

Smoothing Equations. The tracks from two or more radars can be combined to provide a single track whose quality is better than that of any of the individual radars. Consider two radars whose states and error covariances have been extrapolated to the same time. Denote these extrapolated quantities by $(\hat{\underline{x}}_1, P_1)$ for Radar 1 and $(\hat{\underline{x}}_2, P_2)$ for Radar 2. The combined (smoothed) error covariance is

$$P_c = \left[P_1^{-1} + P_2^{-1}\right]^{-1} \qquad (2.4\text{-}11)$$

and the combined states are given by

$$\hat{\underline{x}}_c = P_c\left[P_1^{-1}\hat{\underline{x}}_1 + P_2^{-1}\hat{\underline{x}}_2\right] \qquad (2.4\text{-}12)$$

A two-dimensional example is shown in Fig. 2.4.2. The first state, $\hat{\underline{x}}_1$, has an error ellipse that is largest in a direction that is nearly orthogonal to the direction of the largest error in the second state, $\hat{\underline{x}}_2$. The combined state, $\hat{\underline{x}}_c$, has an error ellipse P_c that is smaller than either P_1 or P_2, showing the benefit of combining states in this manner.

Computer Results. Figures 2.4.3 through 2.4.6 show the track on a 2,000-km/hr aircraft for different amounts of maneuver noise, σ_m. (The measurements were taken by three track while scan radars each with a 6-second scan period, as described in Sec. 3.5.3. The data was pooled, providing an average 2-second data interval.)

Figure 2.4.3 shows the results for σ_m = 4g maneuver noise. The track begins with initiation by two measurements 6 seconds apart. Then

TABLE 2.4.1

KALMAN FILTERING COVARIANCE EQUATIONS

- Error Covariance Extrapolation

$$P(k|k-1) = \Phi(k-1)P(k-1|k-1)\Phi^T(k-1) + \Gamma(k-1)Q(k-1)\Gamma^T(k-1)$$

where $P(k-1|k-1)$ = covariance of $\hat{\underline{x}}(k-1|k-1)$

$P(k|k-1)$ = covariance of $\hat{\underline{x}}(k|k-1)$

$\Phi(k-1)$ = state transition matrix

$Q(k-1)$ = covariance of model (or maneuver) noise

- Measurement Residual Covariance

$$V(k) = H(k)P(k|k-1)H^T(k) + M(k)$$

where $V(k)$ = covariance of $\tilde{\underline{z}}(k)$

$H(k) = \left[\frac{\partial \underline{h}}{\partial \underline{x}}\right]$ = Jacobian of $\underline{h}(\underline{x})$

$M(k)$ = covariance of measurement

- Kalman Gain Matrix

$$K(k) = P(k|k-1)^T H^T(k)V^{-1}(k)$$

where $K(k)$ = Kalman gain matrix

- Error Covariance Update

$$P(k|k) = [I - K(k)H(k)]P(k|k-1)$$

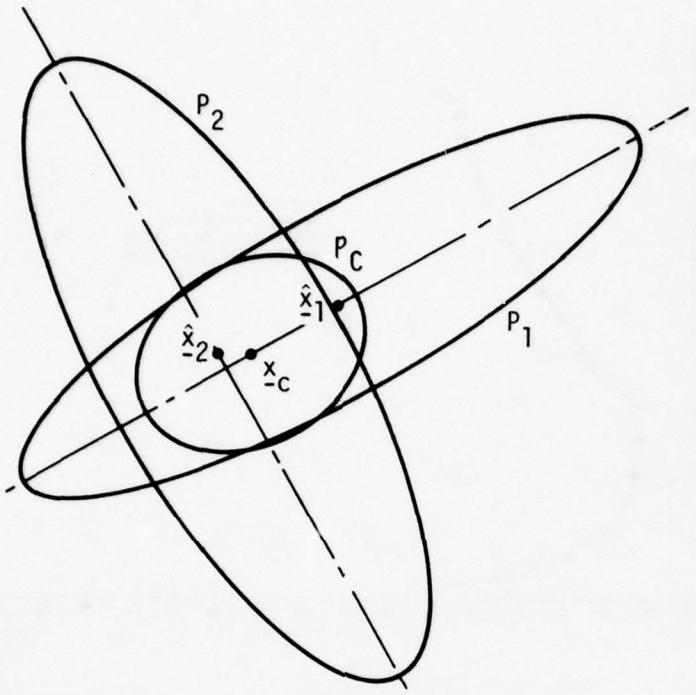


Figure 2.4.2. Example of Combining Two Tracks Using Covariance

other measurements are added approximately every two seconds. The specific measurements and their errors are given in Table 2.4.2.

The track in Fig. 2.4.3 is sufficiently good that the symbols in the plot are nearly on top of each other. Figure 2.4.4 shows a blowup of the last portion of the track.

Figure 2.4.5 shows the result if the maneuver noise is increased to 8g. There is little difference and, for the system being studied, no significant difference.

However, when there is no maneuver noise the result is quite different. Figure 2.4.6 shows that the track is very poor, and track would be lost if association were not forced to occur as it was during these

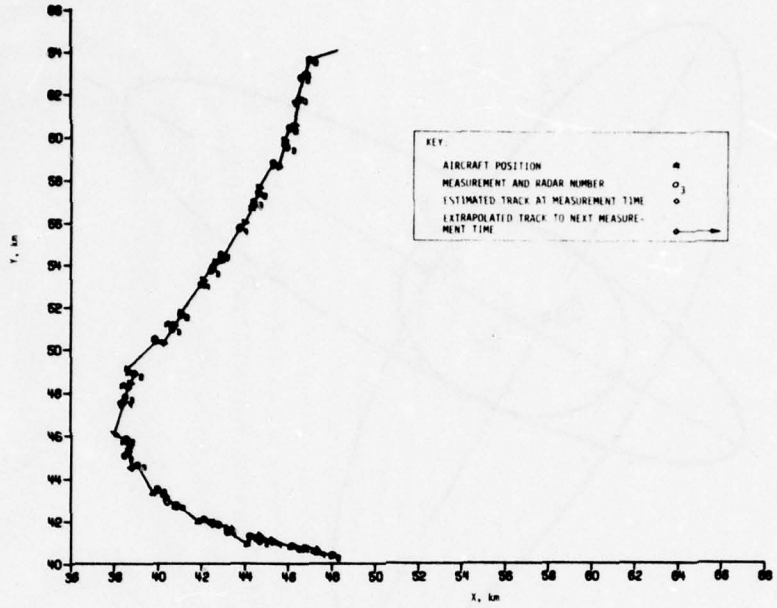


Figure 2.4.3. Kalman Filter Tracking With 4g of Maneuver Noise

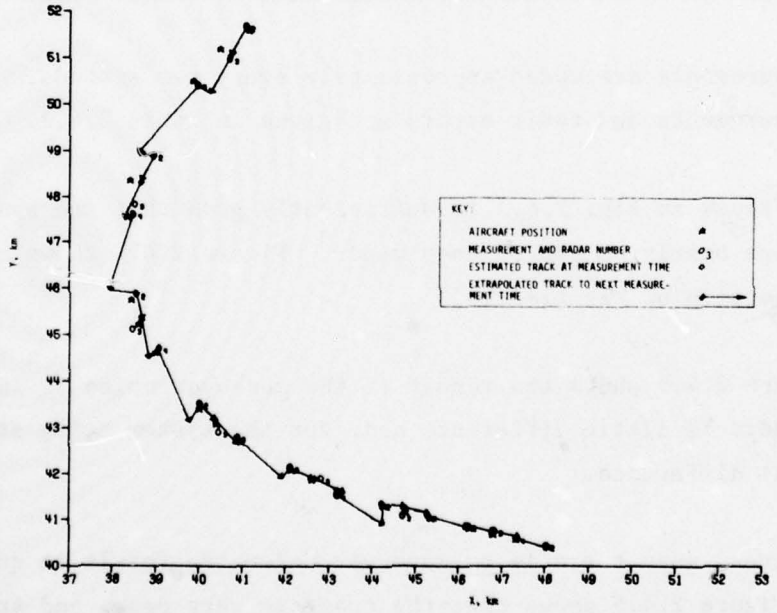


Figure 2.4.4. Enlargement of Kalman Filter Tracking With 4g of Maneuver Noise

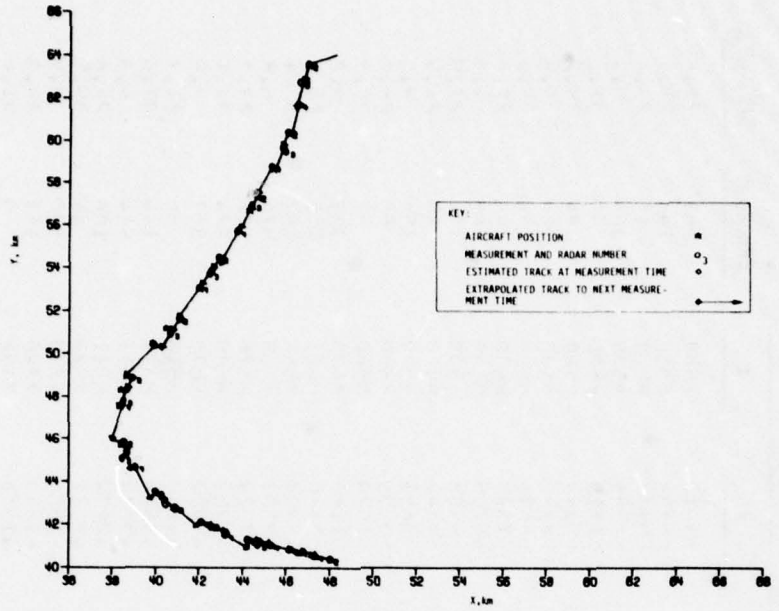


Figure 2.4.5. Kalman Filter Tracking With 8g of Maneuver Noise

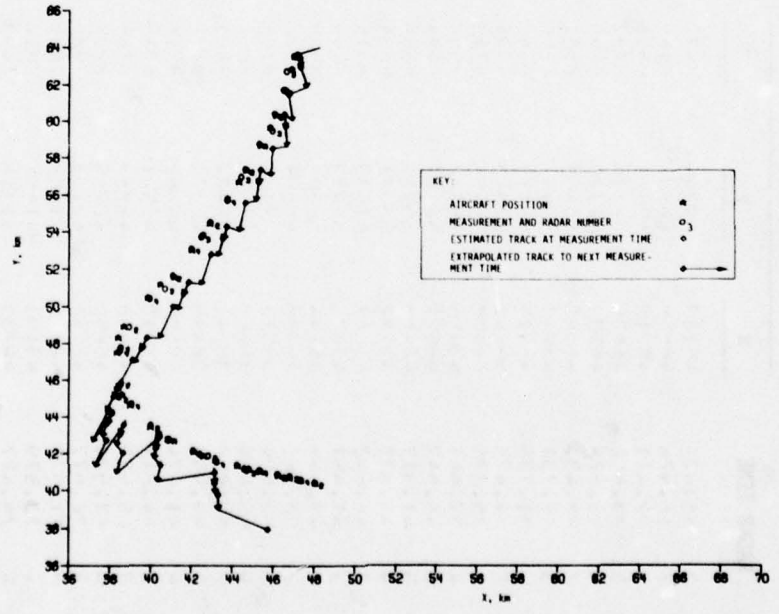


Figure 2.4.6. Kalman Filter Tracking With No Maneuver Noise

TABLE 2.4.2

MEASUREMENTS USED IN KALMAN FILTER RUNS (FROM TACRAN3)

RADAR	TIME	MEASUREMENT, m			AIRCRAFT POSITION, m			ERROR, m	S/N, dB
		x	y	z	x	y	z		
2	19.473	45117	59749	8476	45250	59813	8500	151	20.6
2	25.426	46496	58536	8469	46841	58499	8500	148	21.8
2	32.411	47917	57351	8506	47024	63577	8500	10	23.4
3	33.686	46631	62736	8767	46865	62856	8500	386	4.9
4	35.829	46523	61726	8499	46609	61715	8500	90	31.0
2	38.430	46260	60409	8501	46127	60349	8500	84	25.2
3	39.731	45947	59491	8286	45829	59697	8500	315	5.9
4	41.735	45314	58774	8451	45374	58671	8500	131	24.2
2	44.476	44643	57473	8494	44657	57327	8500	48	27.0
3	45.775	44428	56964	8719	44240	56705	8500	366	4.7
4	47.647	43755	55732	8544	43764	55808	8500	115	27.5
2	50.547	42921	54482	8444	42906	54444	8500	44	29.9
3	51.817	42466	53716	8554	42523	53851	8500	156	9.5
4	53.570	41942	53081	8530	41992	53035	8500	58	25.0
2	56.042	41167	51833	8515	41126	51578	8500	82	30.7
3	57.457	40676	50933	8445	40451	51180	8493	337	9.2
4	59.494	39944	50186	8528	39830	50486	8501	154	24.6
2	62.749	38832	48959	8436	38667	48937	8544	282	32.4
3	63.894	38513	47754	8704	38393	48333	8568	574	8.8
4	65.454	38162	47571	8662	38320	47529	8576	176	23.0
2	68.858	38541	45834	8525	38412	45733	8585	174	34.7
3	69.933	38449	45173	8270	38589	45207	8570	357	8.9
4	71.474	39159	44062	8516	38991	44604	8516	98	21.7
2	74.819	40310	43501	8319	39993	43404	8374	109	37.3
3	75.959	40452	42883	8284	40387	43074	8311	514	9.3
4	77.524	40898	42705	8200	40951	42479	8211	106	20.8
2	80.877	42198	42117	7986	42115	42077	7977	44	39.5
3	81.975	42828	41858	7635	42599	41866	7873	330	9.4
4	83.579	43193	41496	7812	43194	41640	7740	161	20.3
2	86.477	44233	41316	7493	44240	41310	7492	9	41.0
3	87.984	44718	41149	7606	44773	41162	7361	275	10.4
4	89.630	45306	41046	7216	45345	41014	7217	50	20.0
2	92.265	46213	40942	7014	46229	40805	6990	47	41.7
3	93.990	46827	40734	6990	46789	40680	6845	159	10.9
4	95.675	47340	40552	6682	47324	40563	6705	33	19.7
2	98.072	48020	40401	6510	48046	40410	6515	28	41.8

runs. The reason for such poor behavior is that without maneuver (or model) noise, the filter assumes the straight-line model is perfect, and after several measurements begins to believe that its state estimate is much better than the measurement (i.e., the error covariance becomes small), and the measurements are nearly ignored.

The last tracking example shows a case where the measurements are far apart (every 6 seconds), the signal-to-noise ratio is poor (10 dB) and the maneuver is much less than the maneuver noise (σ_m = 8g). Under these rather adverse conditions, Fig. 2.4.7 shows that the track is still adequate. (The aircraft speed is 2,000 km/hr.)

Effect of Maneuver Noise. It should be noted that model noise when entered as a random acceleration (i.e., as maneuver noise) has a more sophisticated effect on the filter than simply keeping the error covariance

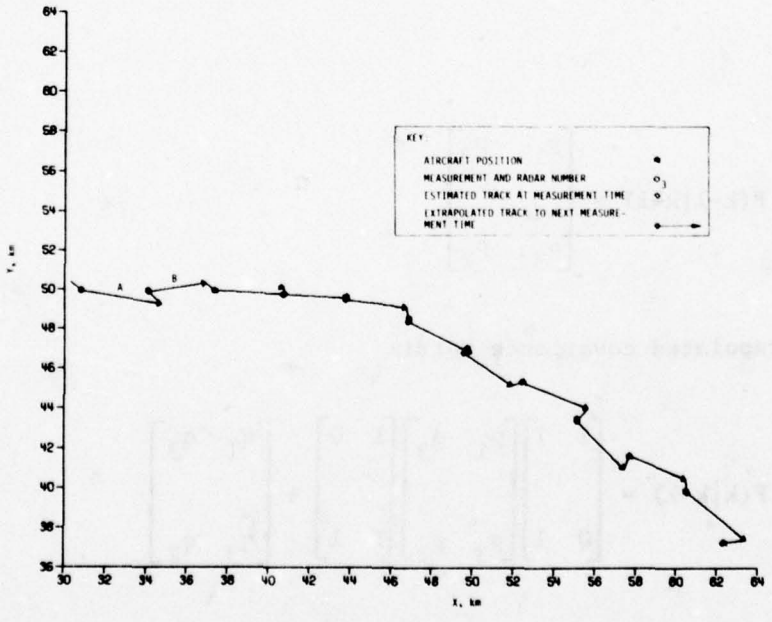


Figure 2.4.7. Kalman Filter Tracking Example

from becoming too small. It also affects the Kalman gain matrix in such a way that it tells the filter how much of the measurement residual is caused by an unknown acceleration. (The rest of the residual is caused by noise.) Large amounts of maneuver noise tell the filter that most of the residual is caused by an acceleration. The filter responds by doubling the part of the residual due to acceleration and adding this to the extrapolated state. Thus if the maneuver noise is large, but the residual is actually due to noise rather than acceleration, the filter will overcompensate.

This effect can be shown mathematically in the one-dimensional case. Let the state vector consist of position x and velocity $\dot{x}$. At time t_{k-1} the error covariance matrix of the estimated state

$$\underline{\hat{x}}(k-1|k-1) = \begin{bmatrix} \hat{x}(k-1|k-1) \\ \hat{\dot{x}}(k-1|k-1) \end{bmatrix}$$

is given by

$$P(k-1|k-1) = \begin{bmatrix} p_1 & p_3 \\ p_3 & p_2 \end{bmatrix}$$

and the extrapolated covariance matrix

$$P(k|k-1) = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_1 & p_3 \\ p_3 & p_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ T & 1 \end{bmatrix} + \begin{bmatrix} q_1 & q_3 \\ q_3 & q_2 \end{bmatrix}$$

The Kalman gain matrix $K(k)$ as given by Eqs. 2.4-7 and 2.4-9 is

$$K(k) = P(k|k-1)H^T\left[HP(k|k-1)H^T + M\right]^{-1}$$

where for the one-dimensional problem

$$H = [1 \quad 0]$$

$$M = \sigma_n^2$$

From Eq. 2.4-6:

$$q_1 = \sigma_m^2 T^4/4 \quad , \quad q_2 = \sigma_m^2 T^2 \quad , \quad q_3 = \sigma_m^2 T^3/2$$

A few matrix multiplications then give

$$K(k) = \begin{bmatrix} \dfrac{\sigma_m^2 T^4/4 + p_1 + 2Tp_3 + T^2 p_2}{\sigma_m^2 T^4/4 + p_1 + 2Tp_3 + T^2 p_2 + \sigma_n^2} \\ \\ \dfrac{\sigma_m^2 T^3/2 + p_3 + Tp_2}{\sigma_m^2 T^4/4 + p_1 + 2Tp_3 + T^2 p_2 + \sigma_n^2} \end{bmatrix}$$

where the p_1 are the elements of the previous error covariance matrix, σ_n^2 is the variance of the measurement noise, and σ_m^2 is the maneuver noise.

Now assume that the maneuver noise is set very high, and dominates the other terms. In this case the gain matrix reduces to

$$K(k) \simeq \begin{bmatrix} 1 \\ \frac{2}{T} \end{bmatrix}$$

The top element of the matrix multiplies the measurement residual to determine the position estimate; the unity says to use all of the residual, that is, the best position estimate is simply the measurement. (In the next section on α-β filtering this is equivalent to $\alpha = 1$.) The $2/T$ term is the gain that determines the velocity estimate; it says to double the residual, which is correct if the residual is actually caused by an acceleration. (Note that this is equivalent to $\beta = 2$ in an α-β filter, a value usually not permitted.) This effect of too much maneuver noise can be seen in Fig. 2.4.7, where the first velocity vector A is too long, which is overcompensated producing velocity vector B which is too short.

A simple argument helps explain the gain of 2. Consider a track where the measurement residual is d. Then if the maneuver noise is large the filter says all of d is due to acceleration a: $d = \frac{1}{2} aT^2$, or $a = 2d/T^2$. The velocity increment ΔV due to the acceleration is aT. Therefore, the velocity increment is $\Delta V = 2d/T$, which explains the factor 2.

Conclusion. The Kalman tracking filter tracks aircraft adequately as long as the maneuver noise is non-zero and is set to a reasonable value such as 4g. The complexity of the filter, however, is such that simpler filters are probably more desirable in the present context. The next two sections describe filters that are easier to implement: the α-β filter and the weighted least squares filter.

2.4.2 α-β Filter

The α-β filter is a greatly simplified recursive filter with a constant and decoupled gain matrix. It is equivalent to the Kalman formulation with a gain matrix

$$K(k) = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & \alpha \\ \beta/T & 0 & 0 \\ 0 & \beta/T & 0 \\ 0 & 0 & \beta/T \end{bmatrix}$$

Since the gain matrix is constant, none of the covariance equations need be calculated.

The simplicity of the α-β filter has resulted in widespread use and investigation.[1] In the present study it was the first filter to be implemented (in the TACRAN1 simulation--Sec. 3.3).

In the α-β filter the state vector consists of position and velocity:

$$\underline{\hat{x}} = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix}$$

$$\underline{\hat{\dot{x}}} = \begin{bmatrix} \hat{\dot{x}} \\ \hat{\dot{y}} \\ \hat{\dot{z}} \end{bmatrix}$$

These are extrapolated from time t_{k-1} to time t_k by

$$\underline{\hat{x}}(k|k-1) = \underline{\hat{x}}(k-1|k-1) + \underline{\hat{\dot{x}}}(k-1|k-1)T \qquad (2.4\text{-}13)$$

[1] See Bibliography, p. 245, numbers 14-19.

$$\dot{\underline{\hat{x}}}(k|k-1) = \dot{\underline{\hat{x}}}(k-1|k-1) \tag{2.4-14}$$

where $\hat{\underline{x}}(k|k-1)$ = extrapolated position vector at time t_k using measurements through time t_{k-1}

$\dot{\underline{\hat{x}}}(k|k-1)$ = extrapolated velocity vector

$\hat{\underline{x}}(k-1|k-1)$ = estimated position at time t_{k-1}

$\dot{\underline{\hat{x}}}(k-1|k-1)$ = estimated velocity at time t_{k-1}

$T = t_k - t_{k-1}$

The position is updated by

$$\hat{\underline{x}}(k|k) = \hat{\underline{x}}(k|k-1) + \alpha\left[\underline{x}_m(k) - \hat{\underline{x}}(k|k-1)\right] \tag{2.4-15}$$

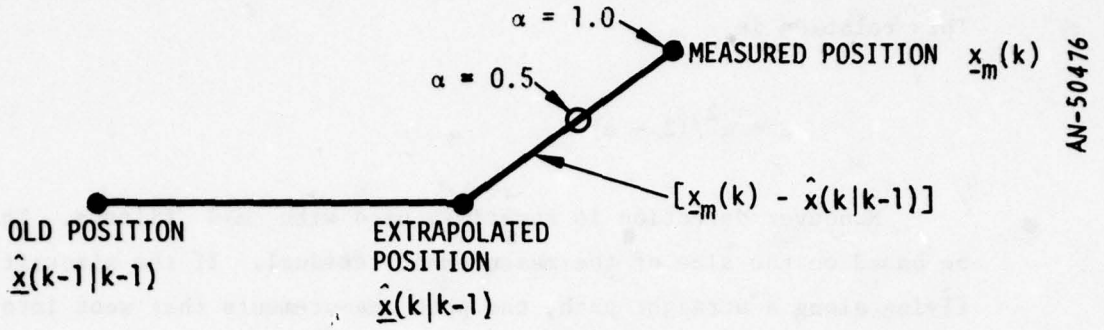
where $\underline{x}_m(k)$ is the measurement in Cartesian coordinates at time t_k and α is a constant between 0 and 1. Note that the quantity $[\underline{x}_m(k) - \hat{\underline{x}}(k|k-1)]$ is the measurement residual in Cartesian coordinates.

If $\alpha = 0$, the measurement is ignored and the new estimate is the extrapolated position given by Eq. 2.4-13. If $\alpha = 1$, the new estimate is the measurement, and the extrapolated position is ignored. Usually α is somewhere between, and some of both the extrapolated position and the measurement are used; this is illustrated in Fig. 2.4.8(a).

The velocity is updated by

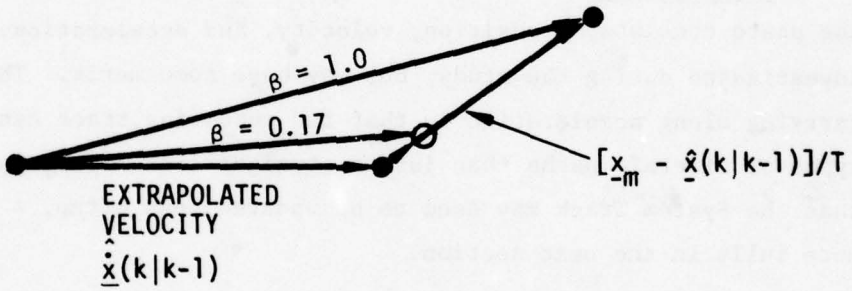
$$\dot{\underline{\hat{x}}}(k|k) = \dot{\underline{\hat{x}}}(k|k-1) + \frac{\beta}{T}\left[\underline{x}_m - \hat{\underline{x}}(k|k-1)\right] \tag{2.4-16}$$

where β is a constant between 0 and 1. Note that $[\underline{x}_m - \hat{\underline{x}}(k|k-1)]/T$ is the velocity residual. The quantity β determines how much of this residual will be added to the extrapolated velocity.



- $\hat{\underline{x}}(k|k) = \hat{\underline{x}}(k|k-1)] + \alpha[\underline{x}_m(k) - \hat{\underline{x}}(k|K-1)]$

(a) Position



- $\hat{\dot{\underline{x}}}(k|k) = \hat{\dot{\underline{x}}}(k|k-1) + \beta[\underline{x}_m - \hat{\underline{x}}(k|k-1)]/T$

NON-MANEUVERING FILTER

$\alpha = 0.5 \qquad \beta = \frac{\alpha^2}{2 - \alpha} = 0.17$

MANEUVERING FILTER

$\alpha = 1.0 \qquad \beta = 1.0$

(b) Velocity

Figure 2.4.8. α-β Filter Illustration

Many years ago it was shown[1] that there is an optimum value of β in terms of α ; thus there is only one free parameter to choose from. This relation is

$$\beta = \alpha^2/(2 - \alpha)$$

Maneuver detection is sometimes used with α-β filters. This can be based on the size of the measurement residual. If the aircraft is flying along a straight path, the past measurements that went into the extrapolated position are still valid, and a small value of α can be used. If a maneuver is detected, the present measurement should be weighted more heavily, and a value of α near 1.0 is used.

α-β-γ Filter. This is an extension of the α-β filter used when the state consists of position, velocity, and acceleration. It was not investigated during the study, but may have some merit. The advantage in carrying along acceleration is that the resulting track can follow more types of aircraft paths than just a straight line. This in turn may mean that the System Track may need to be updated less often, a point discussed more fully in the next section.

2.4.3 Weighted Least-Squares Filter

The least squares filter fits a curve to the last N points so as to minimize the sum of the squares of the distance between the points and the curve. This filter is attractive for tracking targets where due to maneuvers only the last few points are likely to be on the present trajectory.

In order to update the System Track as seldom as possible, a fit to a quadratic curve (rather than a linear curve) was used in TACRAN3 (Sec. 3.4). Higher derivatives were expected to be too noisy to use.

---

[1] T.R. Benedict and G.W. Bordner, "Synthesis of an Optimal Set of Radar Track-While-Scan Smoothing Equations," IRE Trans. Automatic Control, Vol. AC-7, July 1962, pp. 27-32.

The tracking equation in each of three dimensions is

$$x(t) = a + bt + ct^2 \quad \text{(position)} \tag{2.4-18}$$

A similar equation holds for the y and z dimensions. Velocity and acceleration are thus

$$\dot{x}(t) = b + 2ct \quad \text{(velocity)} \tag{2.4-19}$$

$$\ddot{x}(t) = 2c \quad \text{(acceleration)} \tag{2.4-20}$$

Equation 2.4-18 is used to determine the x-coordinate of the track at any time t. The parameters (a,b,c) determine the track, and will be recomputed each time the track is updated.

The parameters (a,b,c) are determined by a set of N measurements at times t_i, $i = 1, 2, \ldots, N$. Using the measurements and the t_i in Eq. 2.4-18 provides a set of N equations which can be solved for the parameters (a,b,c):

$$x(t_i) \equiv x_i = a + bt_i + ct_i^2 \quad ; \quad i = 1, \ldots, N \tag{2.4-21}$$

If N is greater than 3, then the (weighted) least-squares solution to the set of Eq. 2.4-21 is desired, that is, the solution that minimizes

$$J = \sum_{i=1}^{N} W_i \left(a + bt_i + ct_i^2 - x_i\right)^2 \tag{2.4-22}$$

where the W_i are the weights.

Equation 2.4-22 is minimized in the usual manner by taking the partial derivatives of J with respect to the parameters (a,b,c) and setting them equal to zero:

$$\frac{\partial J}{\partial a} = \sum_{i=1}^{n} 2W_i\left(a + bt_i + ct_i^2 - x_i\right) = 0$$

$$\frac{\partial J}{\partial b} = \sum_{i=1}^{n} 2W_i\left(a + bt_i + ct_i^2 - x_i\right)t_i = 0 \qquad (2.4\text{-}23)$$

$$\frac{\partial J}{\partial c} = \sum_{i=1}^{n} 2W_i\left(a + bt_i + ct_i^2 - x_i\right)t_i^2 = 0$$

Solving the set of Eqs. 2.4-23 gives the parameters (a,b,c). In matrix form this is

$$\underbrace{\begin{bmatrix} \sum W_i & \sum W_i t_i & \sum W_i t_i^2 \\ \sum W_i t_i & \sum W_i t_i^2 & \sum W_i t_i^3 \\ \sum W_i t_i^2 & \sum W_i t_i^3 & \sum W_i t_i^4 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} a \\ b \\ c \end{bmatrix}}_{\underline{s}} = \underbrace{\begin{bmatrix} \sum x_i \\ \sum t_i x_i \\ \sum t_i^2 x_i^2 \end{bmatrix}}_{\underline{r}} \qquad (2.4\text{-}24)$$

where the sums are over i, i = 1, ..., N . The solution is then simply

$$\underline{s} = A^{-1}\underline{r} \qquad (2.4\text{-}25)$$

The optimum weight W_i is equal to $1/\sigma_i^2$, the inverse of the variance of the ith measurement. A common model of the variance is

$$\sigma_i^2 = \frac{A^2}{S/N_{ii}} + B^2 \tag{2.4-26}$$

where A and B are constants. Of the first term, which depends on the signal-to-noise ratio S/N_i of the ith measurement, dominates the second term, which is a random bias term, then to a good approximation the optimum weight is proportional to the signal-to-noise ratio:

$$W_i \propto S/N_i \tag{2.4-27}$$

Tracking Errors. Tracking errors arise from three sources: (1) measurement noise, (2) maneuver, and (3) model error. The variance σ_x^2 of the estimated position x due to measurement noise is

$$\sigma_x^2 = E[dx^2] = E\left[\left(da + dbt + dct^2\right)^2\right]$$

$$= E(da^2) + E(db^2)t^2 + E(dc^2)t^4 + 2\left[E(dadb)t\right.$$

$$\left. + E(dadc)t^2 + E(dbdc)t^3\right] \tag{2.4-28}$$

Similarly, the variances of the velocity and acceleration errors are (from Eqs. 2.4-19 and 2.4-20)

$$\sigma_{\dot{x}}^2 = E[d\dot{x}^2] = E\left[(db + 2dct)^2\right]$$

$$= E(db^2) + 4E(dbdc)t^2 + 4E(dc^2)t^2 \tag{2.4-29}$$

$$\sigma_{\ddot{x}}^2 = E[d\ddot{x}^2] = 4E(dc^2) \tag{2.4-30}$$

The covariances of the parameters (a,b,c) are derived as follows:

$$E[d\underline{s}d\underline{s}^T] = E\begin{bmatrix} da^2 & dadb & dadc \\ dbda & db^2 & dbdc \\ dcda & dcdb & dc^2 \end{bmatrix} \tag{2.4-31}$$

$$= A^{-1}E[d\underline{r}d\underline{r}^T]A^{-T} \tag{2.4-32}$$

where the last equation follows from Eq. 2.4-25.

From Eq. 2.4-24

$$E[d\underline{r}d\underline{r}^T] = \begin{bmatrix} \sum\sum W_iW_jE[dx_idx_j] & \sum\sum W_iW_jE[dx_idx_j]t_i & \sum\sum W_iW_jE[dx_idx_j]t_i^2 \\ \sum\sum W_iW_jE[dx_idx_j]t_i & \sum\sum W_iW_jE[dx_idx_j]t_it_j & \sum\sum W_iW_jE[dx_idx_j]t_it_j^2 \\ \sum\sum W_iW_jE[dx_idx_j]t_i^2 & \sum\sum W_iW_jE[dx_idx_j]t_i^2t_i & \sum\sum W_iW_jE[dx_idx_j]t_i^2t_j^2 \end{bmatrix} \tag{2.4-33}$$

where all sums are from 1 to N . Since the measurements x_i are uncorrelated

$$E[dx_idx_j] = \begin{cases} \sigma_i^2 & ; \quad i = j \\ 0 & ; \quad \text{otherwise} \end{cases} \tag{2.4-34}$$

and Eq. 2.4-33 becomes

$$E[\underline{dr}\underline{dr}^T] = \begin{bmatrix} \sum_i W_i^2\sigma_i^2 & \sum_i W_i^2\sigma_i^2 t_i & \sum_i W_i^2\sigma_i^2 t_i^2 \\ \sum_i W_i^2\sigma_i^2 t_i & \sum_i W_i^2\sigma_i^2 t_i^2 & \sum_i W_i^2\sigma_i^2 t_i^3 \\ \sum_i W_i^2\sigma_i^2 t_i^2 & \sum_i W_i^2\sigma_i^2 t_i^3 & \sum_i W_i^2\sigma_i^2 t_i^4 \end{bmatrix} \qquad (2.4\text{-}35)$$

Thus, substituting Eqs. 2.4-24 and 2.4-35 into Eq. 2.4-32 gives $E[\underline{ds}\underline{ds}^T]$.

The variance σ_x^2 in the extrapolated position due to measurement noise (Eq. 2.4-28) is shown in Fig. 2.4.9 for the case where the measurement noise $\sigma_i = \sigma_n$ is the same for each measurement (and therefore the weights $W_i = 1$ are also the same for each measurement). This figure shows that the extrapolation error due to noise increases rapidly with time when small numbers of points, N, are used. For example, if only four sample points are used, the extrapolated position error is nearly ten times the measurement error when the position is extrapolated only three sample periods ahead.

Two other sources of error are model error and maneuver error. Model error is caused by the fact that aircraft flying a constant-g trajectory (except for zero-g) do not fly a path that is described by a second-degree polynomial in Cartesian coordinates. A true trajectory has higher degree terms. This effect is illustrated in Fig. 2.4.10, where a quadratic has been fit through three points to a constant-g trajectory (which is a circle). The predicted trajectory does not continue to follow the true trajectory; the difference is the model error.

Figure 2.4.10 also shows a case where the aircraft maneuvers after the last measurement. This difference between the previous trajectory and the new trajectory is the maneuver error.

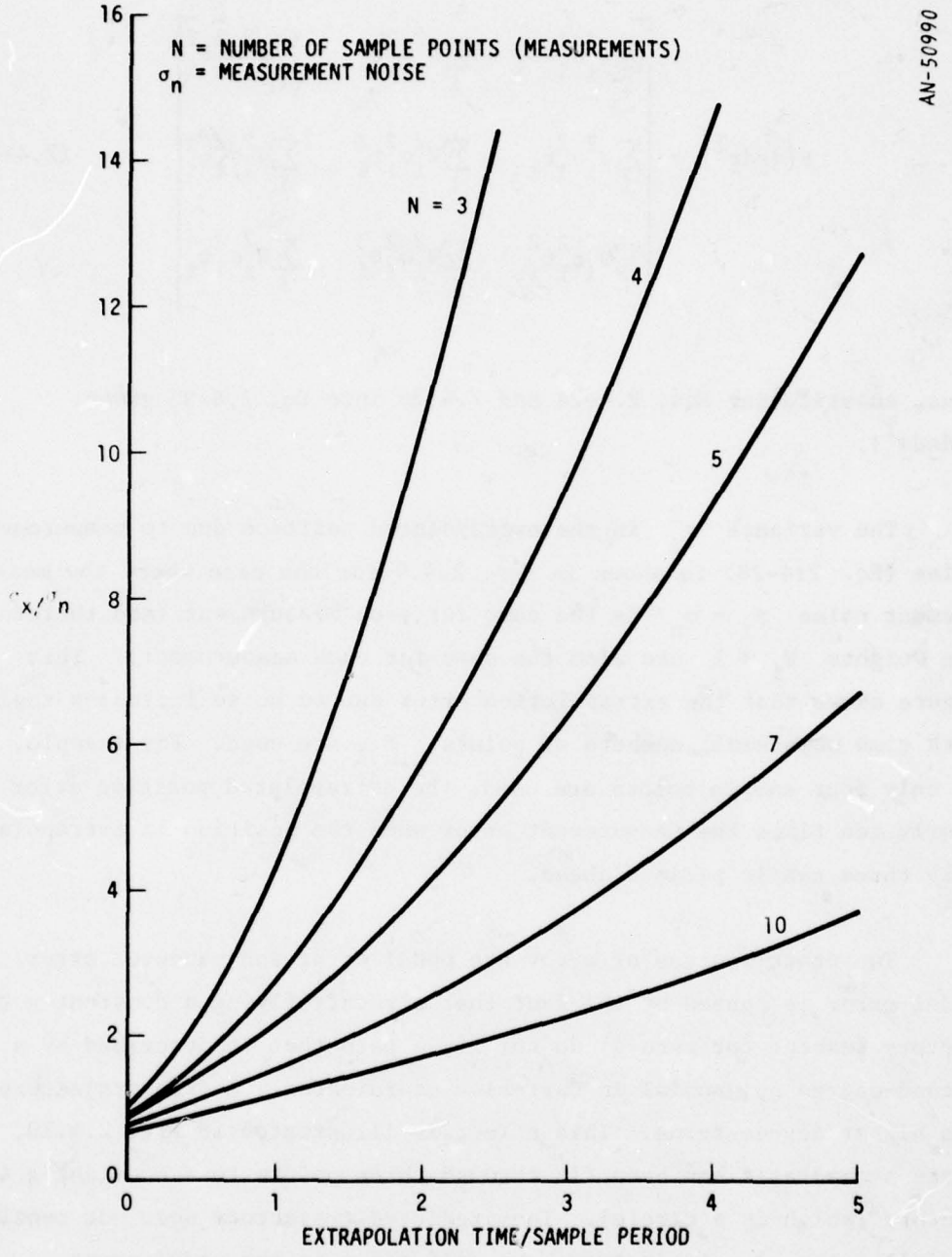


Figure 2.4.9. Extrapolation Error Due to Measurement Error for Least-Squares Fit to Second-Degree Polynomial

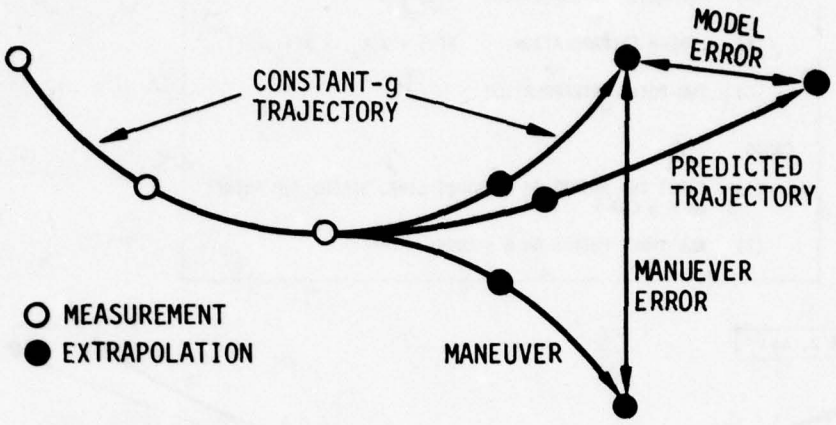


Figure 2.4.10. Model and Maneuver Error in Second-Degree Polynomial Extrapolation

---

Quadratic extrapolation is not best for all purposes, as discussed in Sec. 2.3 on association. Figure 2.4.11 shows two different examples of fitting a second-degree (quadratic) polynomial to three points (Cases 1 and 2), and three ways of extrapolating the resulting polynomials. The quadratic extrapolation (method A) gives the best estimate of the extrapolated position assuming the aircraft does not maneuver. The linear extrapolation (method B) is best for association, because it gives the center point of where the aircraft could be considering all of its maneuver possibilities. A simple two-point extrapolation (method C) is included for comparison.

Computer Results. Numerous TACRAN3 runs were made using the weighted least squares filter; the results of these runs are presented in Sec. 3.5.3. Figure 2.4.12 repeats one of those results (from Fig. 3.5.37). The aircraft, flying at 2,000 km/hr, makes about a 6g turn. The data points are pooled from three radars (as described in Sec. 3.5), and the average time between measurements is 2 seconds. The measurements are weighted by their signal-to-noise ratios.

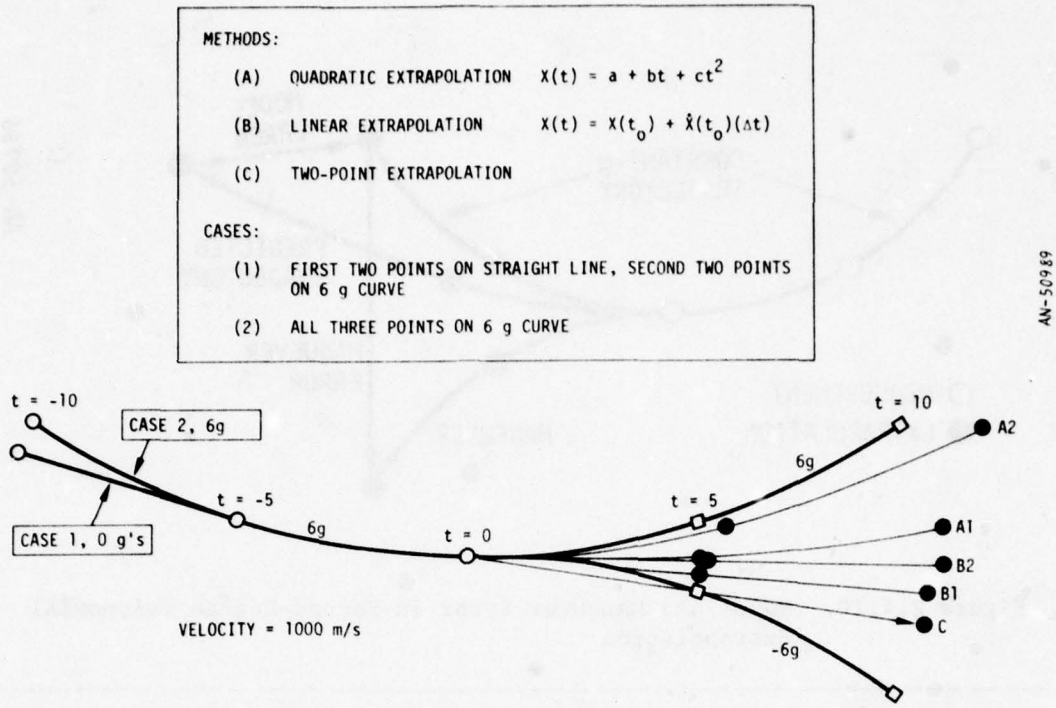


Figure 2.4.11. Extrapolation Methods for Second-Degree Polynomial Fit to Three Points

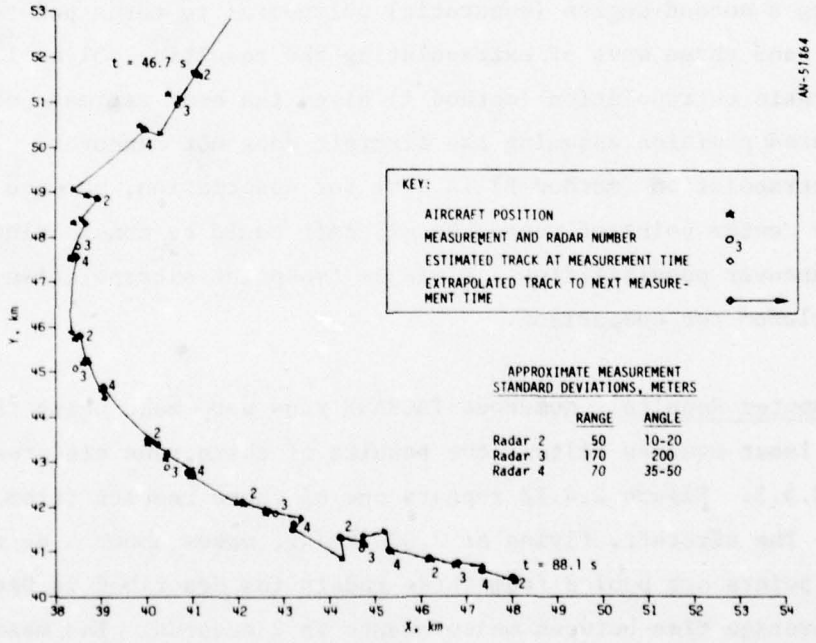


Figure 2.4.12. Distributed Local Track Performance for Aircraft No. 3

The measurements used in the weighted least squares filter are exactly the same as those used in the Kalman filter shown in Fig. 2.4.4. The primary differences appear to be from the fact that the least-squares filter state includes position, velocity and acceleration, whereas the Kalman state includes only position and velocity. The quadratic extrapolation of the least squares filter permits the curved trajectory to be followed more closely. Of course, the Kalman filter could also have been designed to estimate an acceleration term; however, the resulting 9-dimensional covariance matrices would have required substantial additional data processing and communications loads.

Additional computer runs were made to see how well the least-squares filter tracks a very highly maneuvering target. A trajectory was set up as shown in Fig. 2.4.13. At about 4 seconds into the trajectory the aircraft, which is traveling at 1,000 km/hr (280 m/s), makes a 180-degree counterclockwise 9g turn. Immediately thereafter, at about 13 seconds the aircraft begins about a 9g clockwise turn. It then turns counterclockwise and begins a relatively straight flight.

The runs in TACRAN3 generally used five measurements spaced an average of 2 seconds apart. Therefore these parameters were used in the initial high-g runs. Three runs were made, each with different measurement noise (i.e., with different random number kernels in the random number generator). The results are shown in Figs. 2.4.14 through 2.4.16. While as expected the different noise samples do affect the track, the effect is not large. The largest association distance is about 600 meters for Cases 1 and 2, 700 meters for Case 3.

Figures 2.4.17 and 2.4.18 show the effect of using three and seven points, respectively, in the least-squares filter. Three points can be seen to be insufficient (the largest association distance is over 900 meters), while seven points gives results similar to five points (600-m association distance).

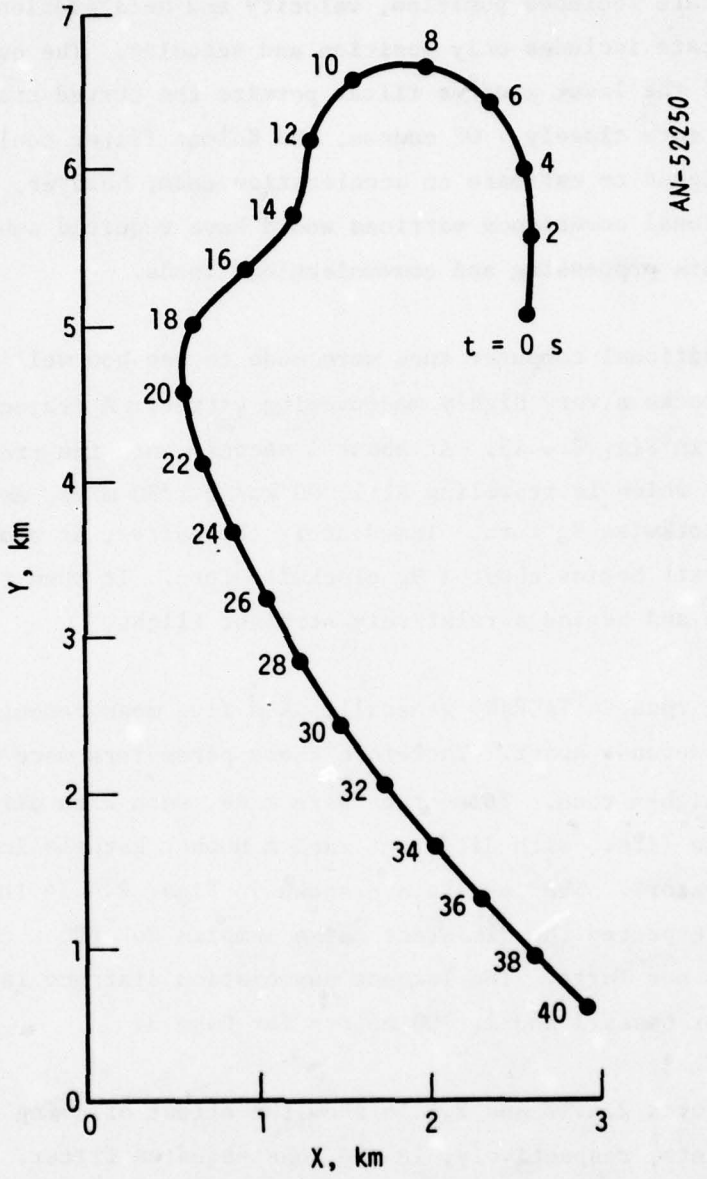


Figure 2.4.13. Trajectory for High-g Tracking Runs

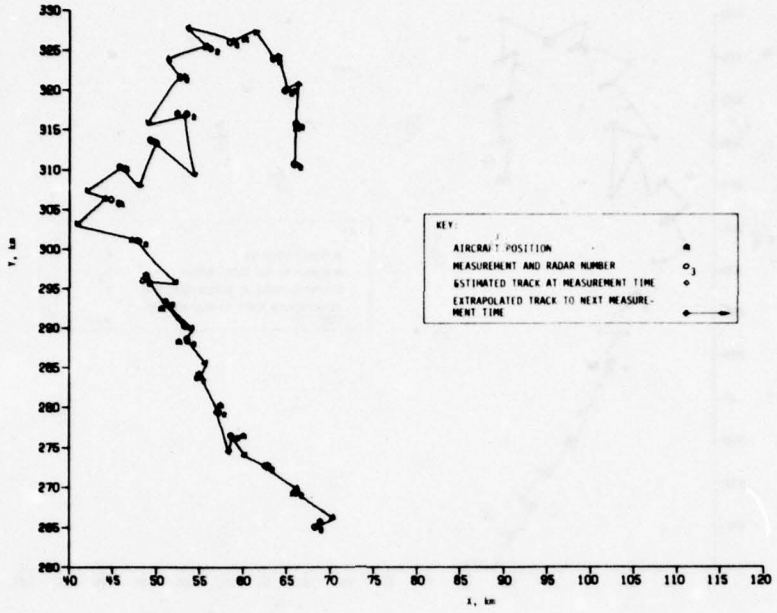


Figure 2.4.14. Track With Five Points, 2-Second Measurement Interval (Case 1)

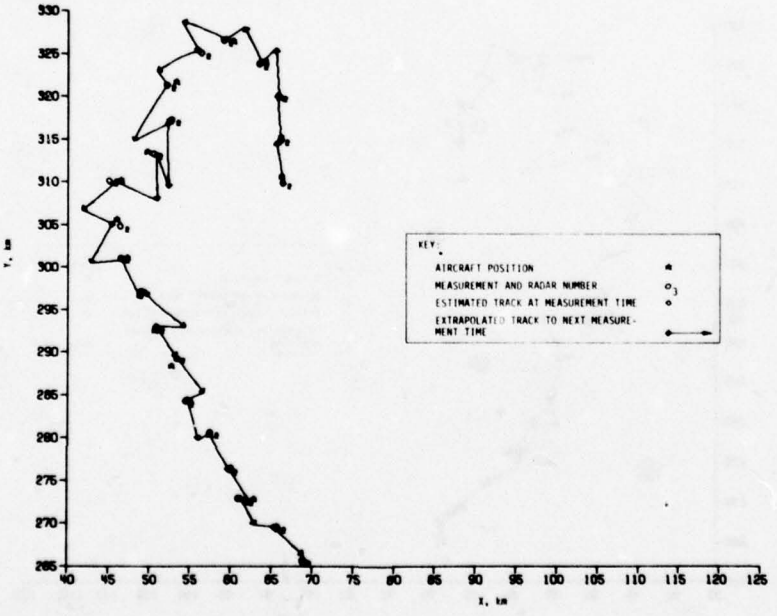


Figure 2.4.15. Track With Five Points, 2-Second Measurement Interval (Case 2)

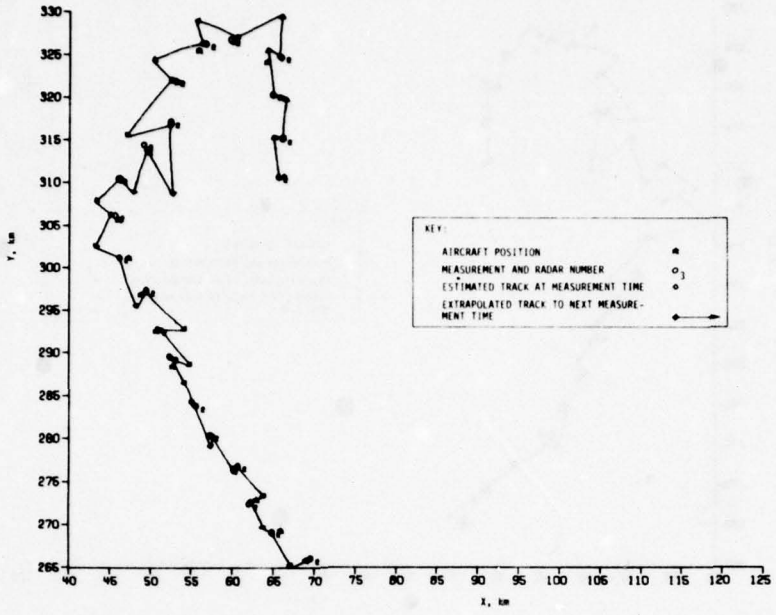


Figure 2.4.16. Track With Five Points, 2-Second Measurement Interval (Case 3)

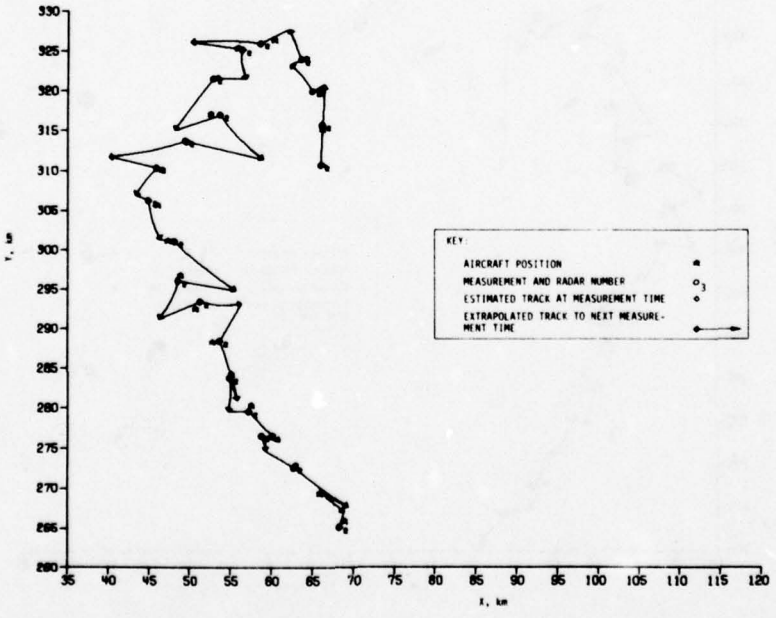


Figure 2.4.17. Track With Three Points, 2-Second Measurement Interval

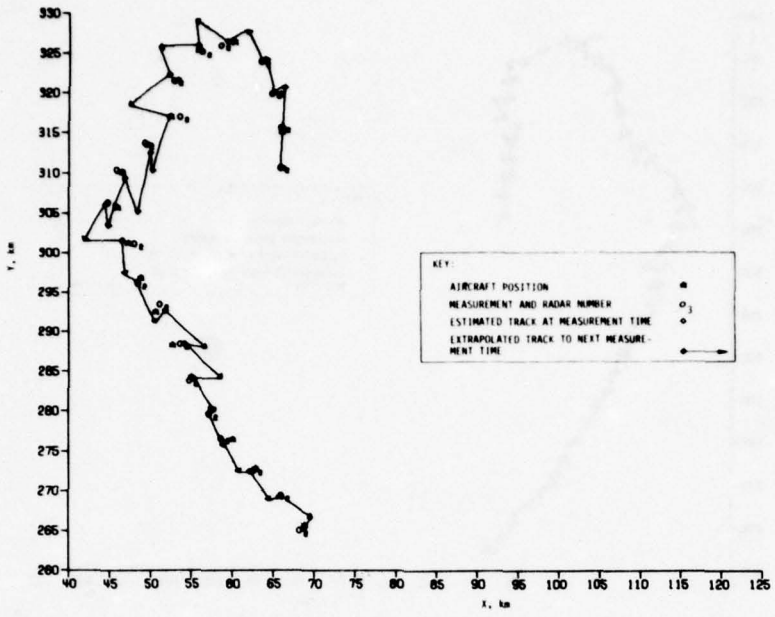


Figure 2.4.18. Track With Seven Points, 2-Second Measurement Interval

---

Finally, Fig. 2.4.19 shows the effect of doubling the measurement rate, with a measurement every second. As expected, the track is considerably improved; the largest association distance is approximately 300 m.

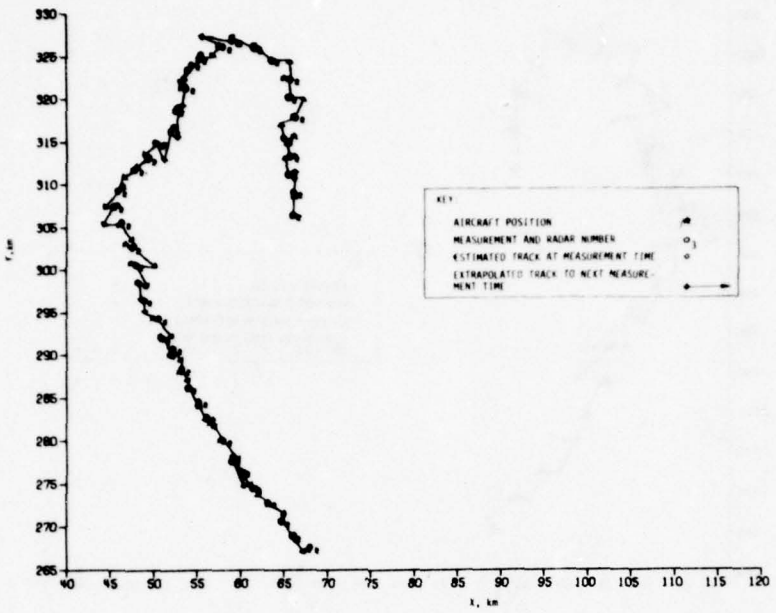


Figure 2.4.19. Track With Five Points, 1-Second Measurement Interval

BIBLIOGRAPHY

ASSOCIATION (CORRELATION) REFERENCES (Sec. 2.3)

1. R.W. Sittler, "An Optimal Data Association Problem in Surveillance Theory," IEEE Trans. Military Electronics, April 1964, pp. 125-139.

2. R.G. Sea, "Optimal Correlation of Sensor Data With Tracks in Surveillance Systems," Proc. 6th Int. Conf. System Science, Honolulu, Hawaii, January 1973, pp. 424-426.

3. J.J. Stein and S.S. Blackman, "Generalized Correlation of Multi-Target Track Data," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-11, No. 6, November 1975, pp. 1207-1217.

KALMAN FILTERING REFERENCES (Sec. 2.4.1)

4. R.A. Singer, "Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-6, No. 4, July 1970, pp. 473-483.

5. R.A. Singer and K.W. Behnke, "Real-Time Tracking Filter Evaluation and Selection for Tactical Applications," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-7, No. 8, January 1971, pp. 100-110.

6. R.A. Singer and R.G. Sea, "A New Filter for Optimal Tracking in Dense Multitarget Environments," Proc. 9th Allerton Conf. on Circuit and System Proceedings, Urbana, Illinois, October 1971, pp. 201-211.

7. R.A. Singer and J.J. Stein, "An Optimal Tracking Filter for Processing Sensor Data of Imprecisely Determined Origin in Surveillance Systems," Proc. IEEE Conf. Decision and Control, Miami Beach, Florida, December 1971, pp. 171-175.

8. R.A. Singer and R.G. Sea, "New Results in Optimizing Surveillance System and Data Correlation Performance in Dense Multitarget Environments," IEEE Trans. Automatic Control, Vol. AC-18, No. 6, December 1973, pp. 571-582.

9. R.A. Singer, R.G. Sea, and K.B. Housewright, "Derivation and Evaluation of Improved Tracking Filters for Use in Dense Multitarget Environments," IEEE Trans. Information Theory, Vol. IT-20, No. 4, July 1974, pp. 423-432.

10. J.S. Thorp, "Optimal Tracking of Maneuvering Targets," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-9, No. 4, July 1973, pp. 512-519.

11. B. Friedland, "Optimum Steady-State Position and Velocity Estimation Using Noisy Sampled Position Data," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-9, No. 6, November 1973, pp. 906-911.

12. R.J. McAulay and E. Denlinger, "A Decision-Directed Adaptive Tracker," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-9, No. 2, March 1973, pp. 229-236.

13. G.C. Ricker and J.R. Williams, "Adaptive Tracking Filter for Maneuvering Targets," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-14, No. 1, January 1978, pp. 185-193.

α-β FILTER REFERENCES (Sec. 2.4.2)

14. T.R. Benedict and G.W. Bordner, "Synthesis of an Optimal Set of Radar Track-While-Scan Smoothing Equations," IRE Trans. Automatic Control, Vol. AC-7, July 1962, pp. 27-32.

15. H.R. Simpson, "Performance Measures and Optimization Condition for a Third-Order Sampled-Data Tracker," IEEE Trans. Automatic Control, Vol. AC-8, April 1963, pp. 182-183.

16. S.R. Neal, "Discussion on 'Parametric Relations for the α-β-ν Filter Predictor'," IEEE Trans. Automatic Control, Vol. AC-12, No. 3, June 1967, pp. 315-317.

17. R.L.T. Hampton and J.R. Cooke, "Unsupervised Tracking of Maneuvering Vehicles," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-9, No. 2, March 1973, pp. 197-207.

18. B.K. Bhagavan and R.J. Polge, "Performance of the g-h Filter for Tracking Maneuvering Targets," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-10, No. 6, November 1974, pp. 864-866.

19. C.C. Schouler, "Optimal α-β Filters for Systems With Modeling Inaccuracies," IEEE Trans. Aerospace and Electronic Systems, Vol. AES-11, No. 6, November 1975, pp. 1300-1306.

3 DISTRIBUTED NETWORK SIMULATION

3.1 OVERVIEW

A digital computer simulation of netted systems was constructed during the study as an analysis tool. This simulation models four major elements:

1. Radar targets and environment
2. Radars
3. Data processors
4. Communications network

The interconnection of these elements is shown in Fig. 3.1.1. At each radar node in the network the radar simulation interacts with the radar target and environment simulation to obtain radar measurements. These in turn are passed to the radar's data processor simulation, which performs the usual radar functions of track initiation, association and filtering as well as other required radar data processor logic.

Other types of data processor nodes can also exist in the simulation. The figure shows two of these: (1) intermediate nodes, which are intermediate communications nodes, and (2) user nodes, where data from the system is displayed for use by operations planners and controllers.

All data processors are interconnected through the communications network simulation, which permits complete flexibility of the interconnections.

The Distributed Network Simulation is completely modular to facilitate interchange of simulation models. Any number of radar types, data processor types, radar target types and communication types can be simultaneously modeled. Each type can be replicated as often as required using the same code module. Since dynamic storage allocation with overflow to secondary storage is used throughout, there are no upper limits to the

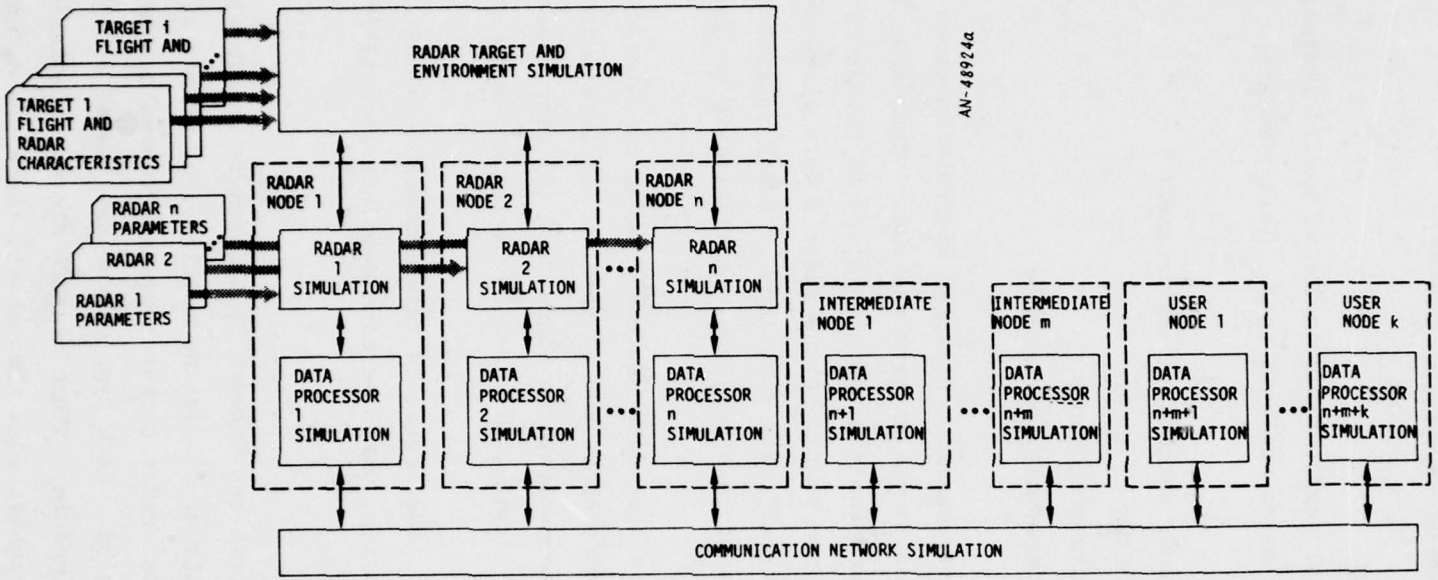


Figure 3.1.1. Distributed Network Simulation

numbers of targets, radars, data processors, communications messages, etc., that can be simulated. (There may, however, be some practical limits in terms of computer running time and cost.)

The simulation relies on a number of simulation systems and tools developed and refined at GRC during the past ten years. These systems and tools, described in Sec. 3.6, include IFTRAN to simplify and clarify model (code) preparation; dynamic storage allocation (DSA) to provide dynamic assignment of data storage; PRINTOUT to simplify data type specification; FLEXREAD, to provide a highly flexible data input system; Action Sequence Chains to provide the simulation capabilities of event processing, time delays and other useful facilities; a postprocessor that permits the assimilation, selection, ordering, printing and plotting of the voluminous output data typical of major simulations; and the TRAID system to provide trajectory models, matrix and vector routines, and input/output routines.

The specific versions of the Distributed Network Simulation that will be described later were developed to aid in the evaluation of specific netted systems for use in tactical air control and are called the TACRAN (Tactical Air Control Radar Net) simulations. Before describing the different versions of TACRAN in detail, the specific models of the radar, target (aircraft) communication simulation models common to all versions will be described, and the data processing models will be summarized.

3.2 TACRAN SIMULATION MODELS

Radar Simulation Model. The TACRAN simulation models track-while-scan radars, providing noisy radar returns in up to four dimensions (range, azimuth, elevation, and doppler). The radar simulation is entered separately for each radar in the net (Fig. 3.2.1). Each time it is entered all aircraft returns and false alarms are generated for the next scan sector. (The size of the sector is defined as input; typically about a 20-degree sector has been used.) First, all of the false alarms

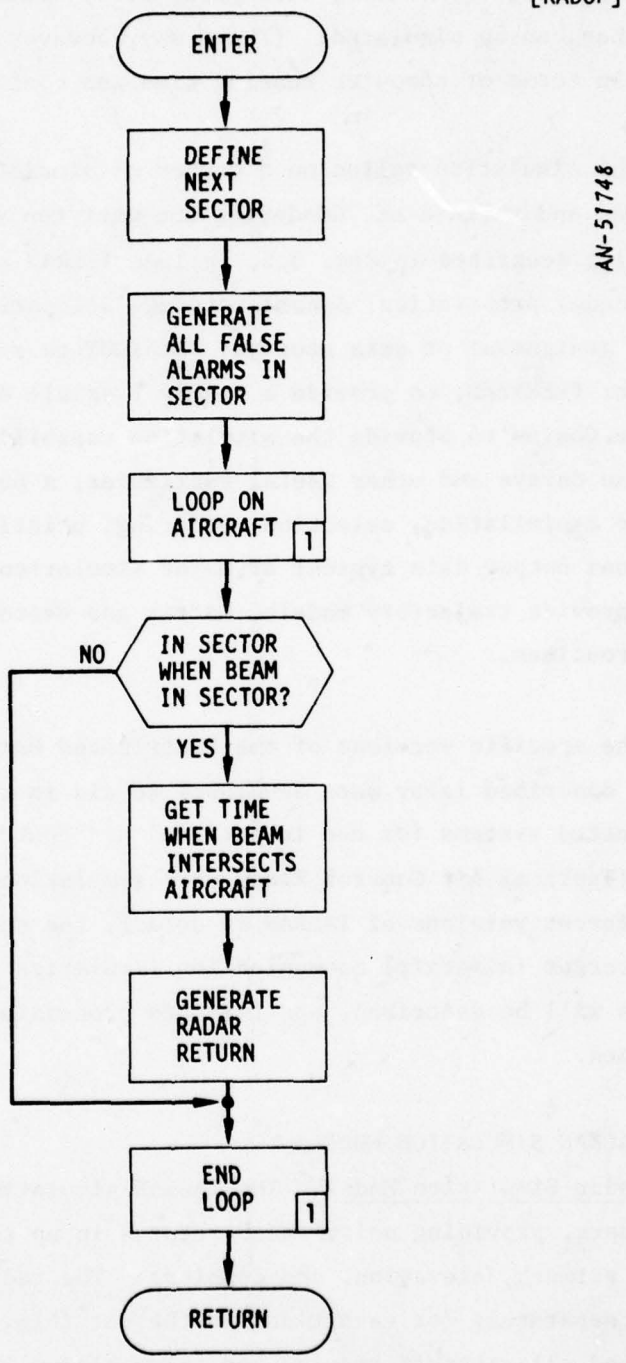


Figure 3.2.1. TACRAN Radar Simulation

in the sector are generated. Next the intersection time and position is determined iteratively for each aircraft in the sector and a noisy return is generated. The simulation system passes the returns to the radar data processor in the proper order at the proper time.

The radar is defined by a number of parameters as described in Table 3.2.1.

Aircraft Flight Simulation Model. Aircraft are defined by their mass, area, drag, thrust, and maximum accelerations. Flight paths, as depicted in Fig. 3.2.2, are described by an initial state and a few points along the path. Each point defines a desired position and speed. The actual flight path is determined by proportional navigation on the next point. As an aircraft nears a point, it begins flying towards the point beyond it; thus aircraft do not necessarily fly precisely through the points.

TABLE 3.2.1
RADAR SIMULATION PARAMETERS

For Each Radar Type:

- Radar Scan Period
- Range on 1 square meter
- Detection Threshold
- Range Resolution
- Azimuth Resolution
- Elevation Resolution
- Maximum Elevation Angle
- Average Number of False Alarms Per Scan

For Each Radar:

- Cartesian Position
- Initial Pointing Direction
- Pointer to Associated Data Processor

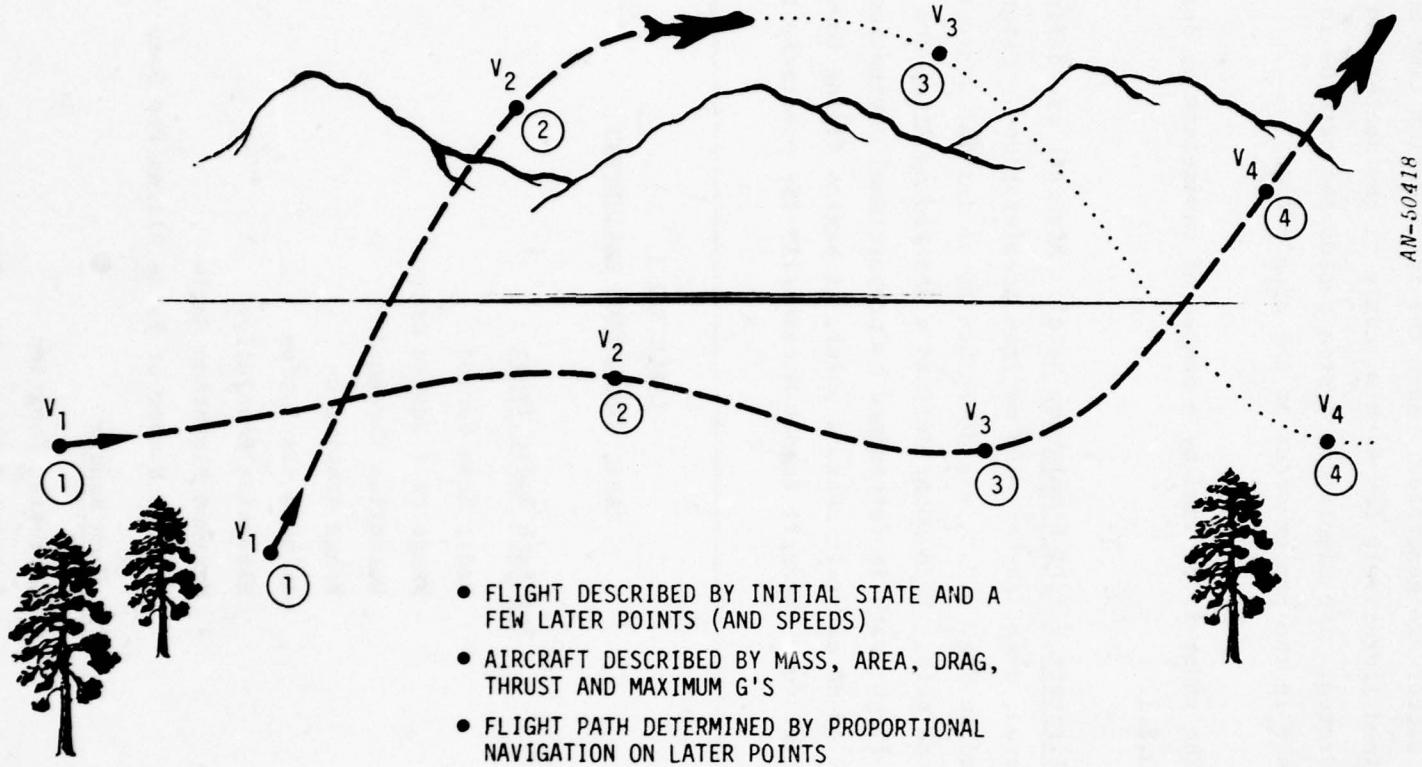


Figure 3.2.2. Aircraft Flight Simulation

Communications Simulation. The communications simulation interconnects all of the data processors in the network by point-to-point communications. The interfaces between the data processor simulations and the communications simulations are realistic; that is, a sending data processor sets up a message to one or more other data processors (Fig. 3.2.3) and turns it over to the communications simulation, which in turn routes the message over the appropriate communication links with appropriate queuing and delays, and delivers it to the receiving data processor.

The simulation can send three types of messages: (1) directed, (2) system (addressed), and (3) system (unaddressed). A directed message is one which is sent to one or more specified nodes; it is always an addressed message since it contains the address(es) of the receiving node(s). A system message is a message that is sent to all other nodes in the system. These can be either addressed or unaddressed. Addressed system messages initially contain the address of every node that is to receive the message; thus it is handled exactly the same as a directed message. The only difference is that for a system message the sending data processor does not append the receiving addresses; this is done by the communications simulation.

Unaddressed system messages are transmitted by a scheme due to Otto Wech of ESD (XRT). Here the sender transmits the system message over all the connected links (e.g., links 1-4 in Fig. 3.2.3). The algorithm for any node receiving such a system message is to retransmit it over all its links except the one it was received on, unless it has received this message before. In the latter event the message is discarded. The result of this algorithm is that shortly after all nodes receive the message, it is no longer retransmitted.

The unaddressed system message has some advantages and one disadvantage over the addressed message. Its advantages are that it is a simpler scheme to implement and that it is more resilient to link

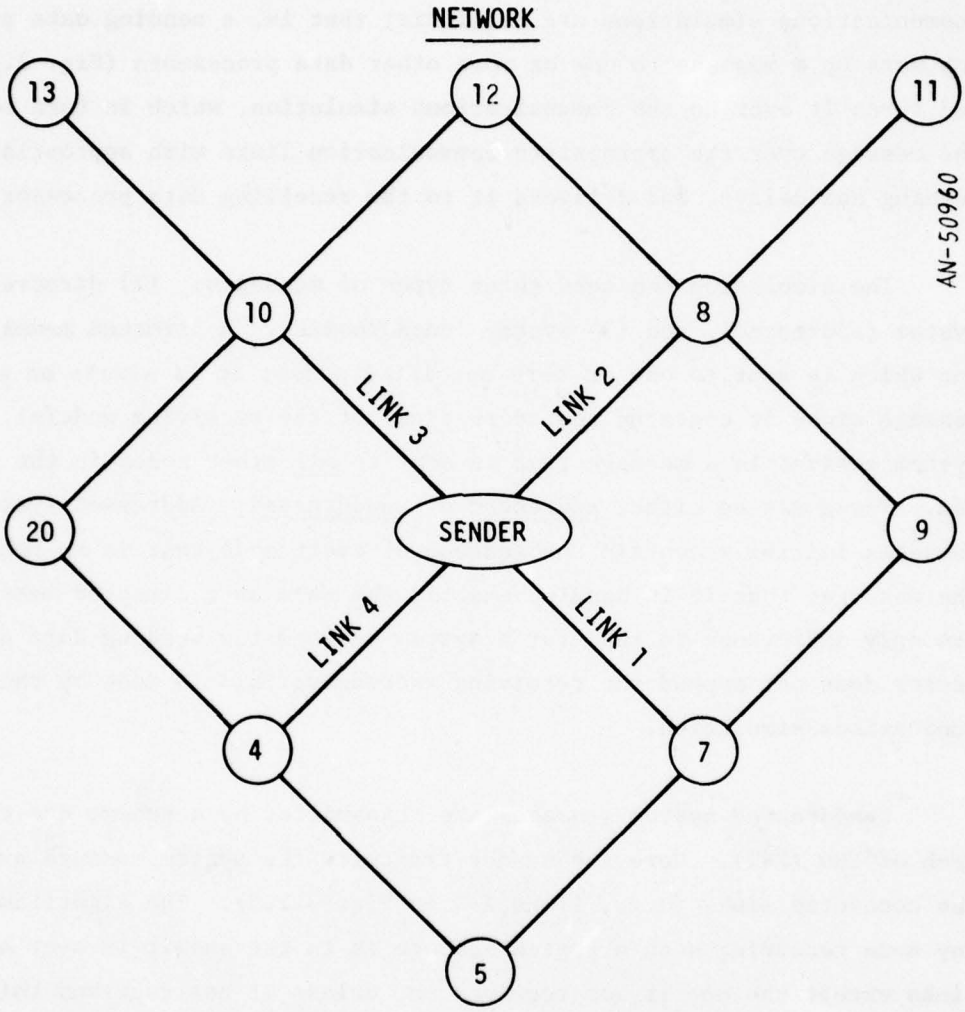


Figure 3.2.3. Communications Simulation

failures. Its disadvantage is that it requires more communications bandwidth.*

Addressed messages are delivered according to route tables at each node. The route table defines the link on which a message to a given destination node is to be transmitted. Thus, for example, Fig. 3.2.4 shows that if node i has originated or received a message to node 5, it should be sent out on link 4.

All messages consist of a header, text, and a cyclic redundancy check code (CRC) for error checking. The header contains data required by the communication simulation to route the message and data required by the receiving data processor to identify the message. Thus the header contains such data as an identification label of the originator of the message, a message type, message priority, and (in the case of addressed messages) a list of identification labels (addresses) of the nodes to receive the message.

The technique of routing addressed messages is illustrated in Fig. 3.2.5. Assume a message received by node 2 is addressed to nodes 2, 3, 8, and 10. This message and the list of receivers attached to the header is depicted at the top of the figure. The relevant part of the network

---

*It can be shown that in a network with N nodes and L two-way links, an addressed system message requires (N - 1) links to completely propagate through a network, whereas an unaddressed message requires L links (plus an additional number to take care of the case where copies of the message cross on the same link). A network where all nodes have k links connected has L = kN/2 links. Thus the ratio of links used for an unaddressed message to link used for an addressed message is

$$\frac{\text{unaddressed message links}}{\text{addressed message links}} = \frac{kN}{2(N-1)}$$

For large N and k = 3 links/node, this ratio is 1.5. For k = 4 it is 2. Since an addressed system message is longer (because of the addresses) than an unaddressed message, the ratio of average bandwidths required is somewhat less than the above ratio.

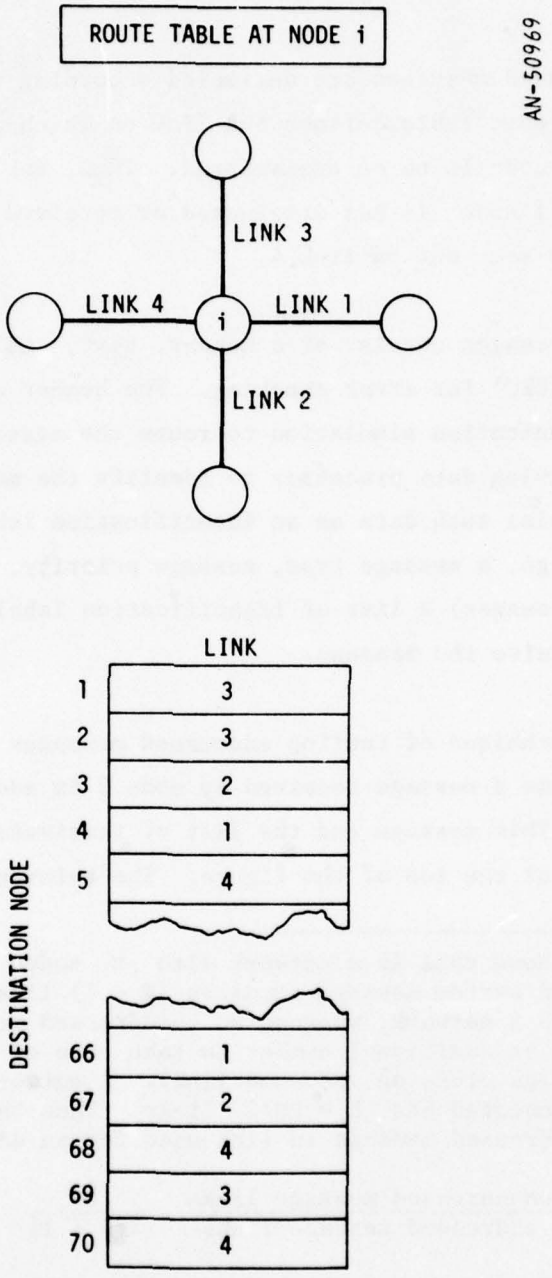


Figure 3.2.4. Route Table Structure

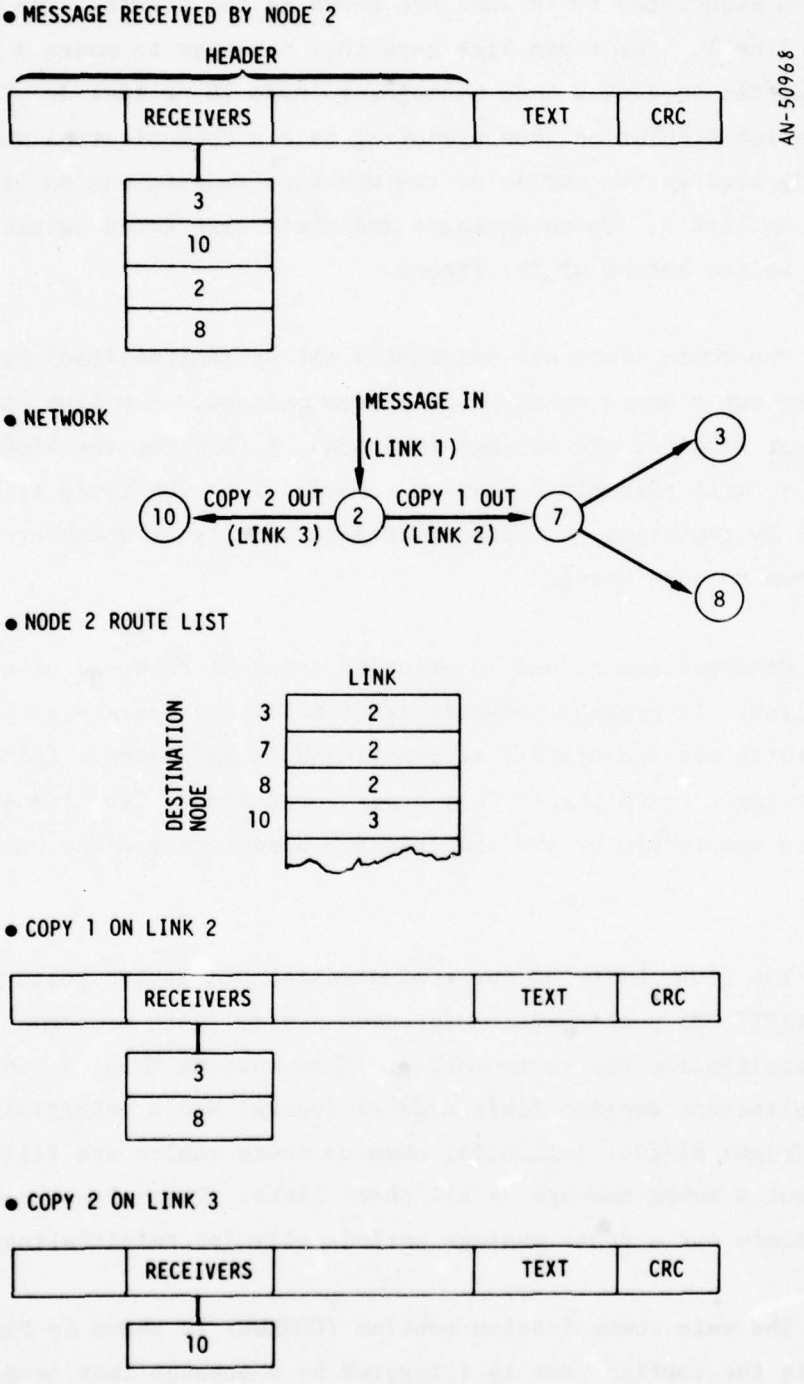


Figure 3.2.5. Example of Addressed Message Routing

and the associated route list are shown in the figure. The message comes in on link 1. The route list says that messages to nodes 3 and 8 are transmitted on link 2, and messages to node 10 on link 3. Thus the communication simulation (now operating as the communication subsystem of node 2) creates two copies of the message, sending one on link 2, the other on link 3. These messages and their associated receiver lists are shown at the bottom of the figure.

The route lists are originally set up (initialized) by each node sending out a short unaddressed system message. The link on which node j first receives the message from node i defines the link on which node j will transmit a message to node i . The route lists are maintained by repeating this procedure periodically or whenever the network is known to have changed.

Messages are queued in priority order at each end of a transmission link. At present messages are transmitted one way at a time over a link, with message traffic alternating between queues. (Simple changes to the logic would permit full duplex operation.) The transmission duration is determined by the transmission bandwidth and the number of bits sent.

The flow charts of the communication simulation begin with the routine (INITCOM) that sends unaddressed system route messages to initialize and reinitialize the route tables. This routine (Fig. 3.2.6) has an initialization section (left side of figure) and a reinitialization section (right side). Initially, when no route tables are filled, all nodes send out a route message on all their links. Thereafter, each node in turn sends out a route message periodically for reinitialization.

The main communication routine (COMMUN) is shown in Fig. 3.2.7. This is the routine that is triggered by a message that needs retransmitting;

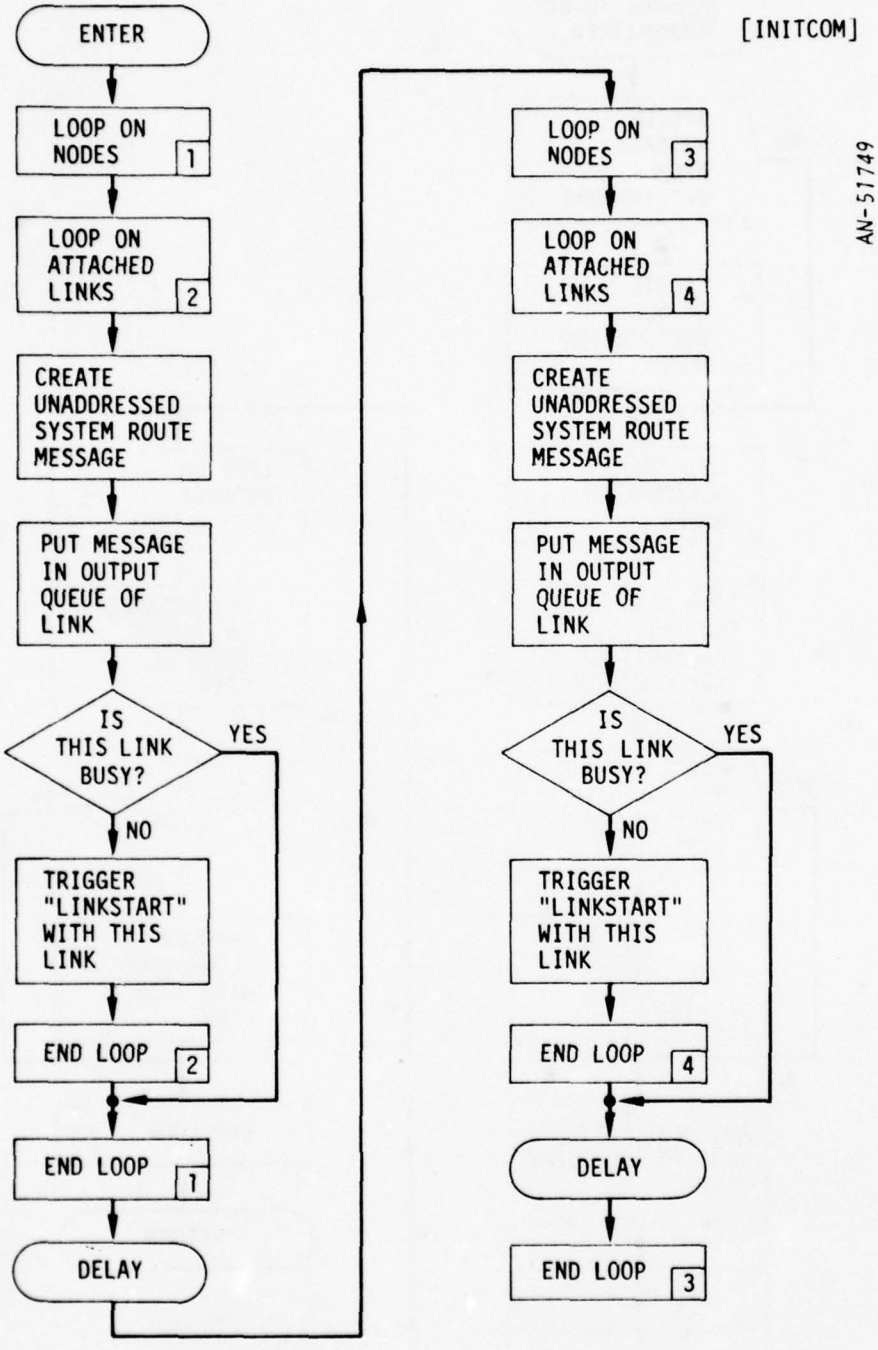


Figure 3.2.6. Sending of Unaddressed System Route Messages

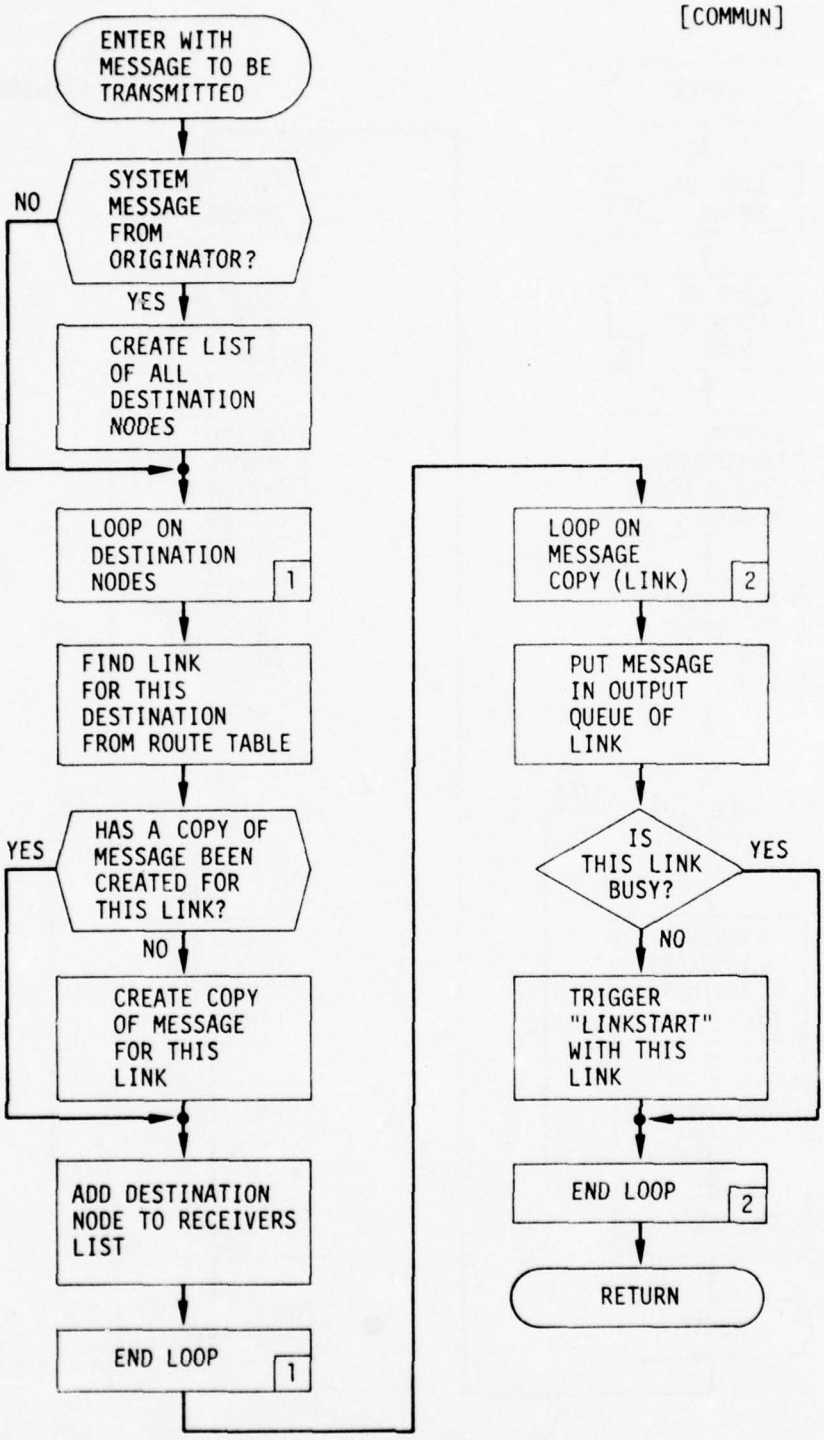


Figure 3.2.7. Main Communications Routine

it represents the communications section of each node. This routine is entered when a node originates a message and whenever a node retransmits a message.

All messages sent by COMMUN are addressed. (Unaddressed system route messages are created and handled by INITCOM.) Directed messages are given to COMMUN with the destination nodes already attached. System messages initially have no addresses attached; these are added by COMMUN.

Next, during a loop on destination nodes, copies of the message are made as required, and the destination nodes are apportioned among the copies (see Fig. 3.2.5). Finally, the message copies are put into the output queues of the appropriate links. If a link is "busy", it means its output queue is not empty and another message is presently being sent. The routine that transmits messages (LNKMNGT) will eventually get to this message, so no further action is necessary. However, if the link is not busy (i.e., "free"), the transmission must be initiated by triggering LINKSTART, a part of LNKMGMT, and setting the link status to "busy".

LINKSTART is shown in Fig. 3.2.8. Its purpose is to "transmit" the first message of a previously free link. Messages are transmitted by computing the transmission time ΔT , and then triggering "End of Transmission" (EØT) to occur ΔT seconds later.

End of Transmission (EØT), shown in Fig. 3.2.9, is entered at the end of transmission of a message on a link. The two types of messages are handled differently. Addressed messages only concern themselves with whether or not this node is to receive the message, and if so, is it the only receiver (destination node) on the list. If this node is to receive the message, a copy is given to the node by triggering RCVCOMM, which is the routine (action sequence chain) in the data processor simulation that handles received messages.

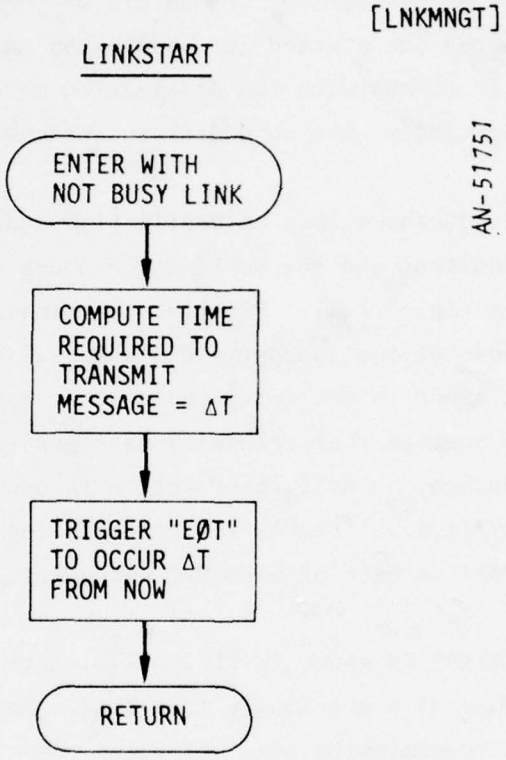


Figure 3.2.8. Start a Not-Busy Link

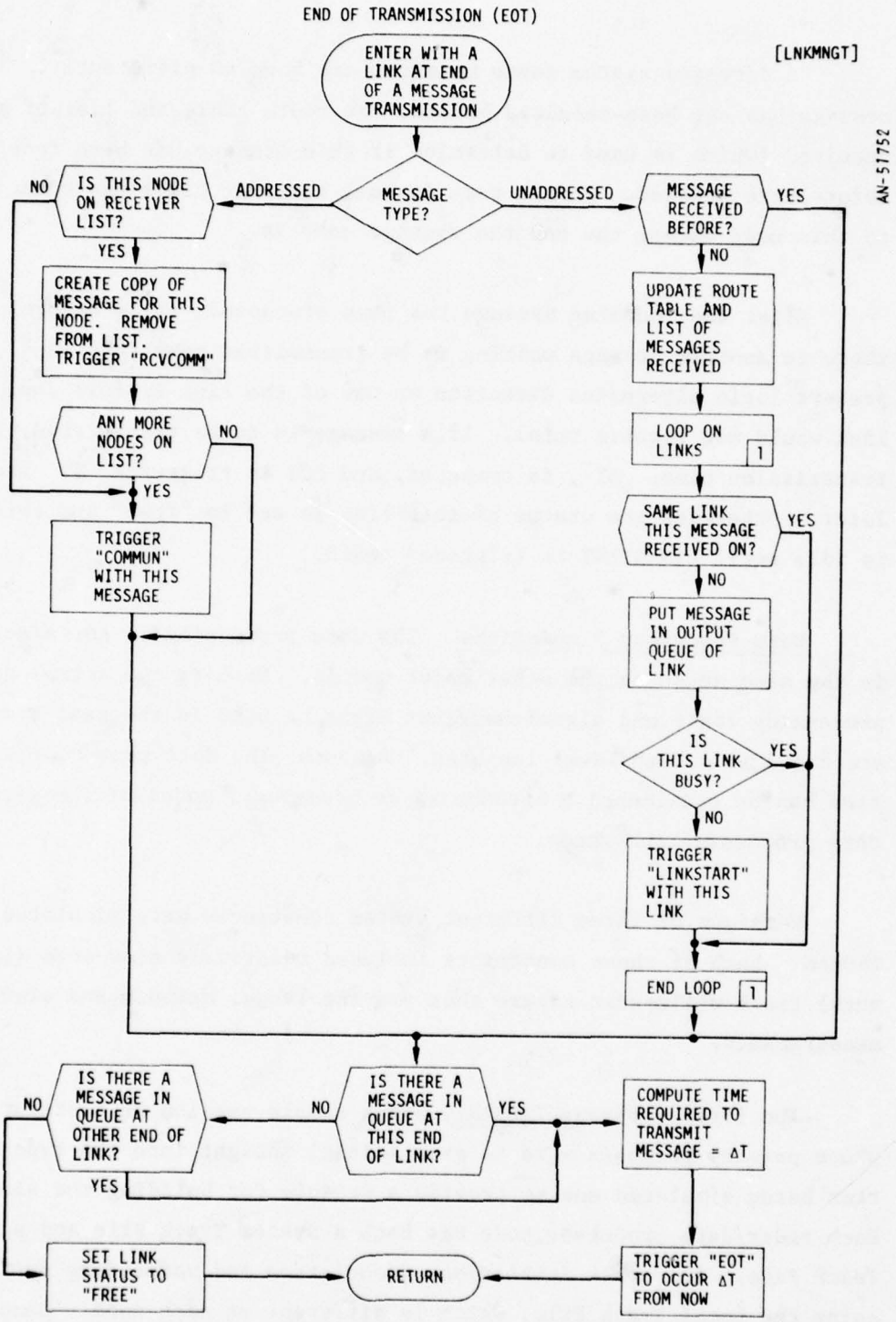


Figure 3.2.9. Management of a Busy Link

Unaddressed system route messages are handled differently. If this message has not been received before, the route table and list of messages received (which is used to determine if this message has been received before) are updated. Then a copy is sent out over all links connected to this node except the one the message came in.

After the incoming message has been processed, it is determined if there is another message waiting to be transmitted over this link. The present logic alternates direction on use of the link (a full duplex link would not require this). If a message is to be transmitted, its transmission time, ΔT , is computed, and EØT is triggered ΔT seconds later. Otherwise the status of this link is set to "free" and this link is idle until LINKSTART is triggered again.

Data Processor Simulations. The data processing is not simulated in the same sense as the other major models. Rather, the actual data processing logic and algorithms that might be used in the real system are coded in a high level language. As such, the data processor simulation can be considered a breadboard or brassboard model of the real-time data processing subsystem.

Versions of three different system constructs were simulated in TACRAN. Each of these constructs included relatively slow-scan (in azimuth) track-while-scan radars that provide range, azimuth and elevation measurements.

The first version, TACRAN1, was a simple version of Configuration 1 whose primary purposes were to gain initial insight into the type of systems being simulated and to provide a vehicle for building the simulation. Each radar/data processor node has both a System Track File and a Local Track File. All track initiation, association and update are performed using the Local Track File, which is different at each node. Maneuver detection is used to update the System Track File, which is replicated (to the degree possible) at each node. TACRAN1 is described in Sec. 3.3.

TACRAN2 was developed as a sequel to TACRAN1. The basic Configuration 1 system concept is the same, but the track association and update algorithms are much more sophisticated. Associations are performed with chi-square tests, and updating is by Kalman filtering. TACRAN2 is described in Sec. 3.4.

Both of the first two versions of TACRAN had conceptual and algorithm problems that were uncovered while using the simulations. Therefore a third system construct (Configuration 2) was developed to overcome these deficiencies, which included poor association probability due to the low scan rate and large communication and data processing loads. TACRAN3, described in Sec. 3.5, simulates this new construct. This version pools data from several radars to effectively increase the data rate on each aircraft, associates using a more sophisticated algorithm, and tracks using a weighted least-squares quadratic fit to the last few pooled measurements.

3.3 SIMPLE-ALGORITHM SYSTEM (TACRAN1)

Much was learned about the nature of the distributed tactical air surveillance and control systems investigated during the study from the construction and running of the first Configuration 1 version of the distributed network simulation, TACRAN1. While this version is not a recommended system, it is described here for the insights gained.

A flow chart of the TACRAN1 Simulation (Fig. 3.3.1) shows N radar/data processor nodes, all connected through the communications network simulation. Each radar makes measurements on aircraft whose characteristics are determined by the aircraft flight simulation and provides radar returns (including false alarms) to its associated data processor.

The data processing logic and algorithms at each node are identical. A simplified flow chart of this logic is shown in the dashed box labelled "data processor #1". When a new return is received from the radar, it is first associated with the Local Track File. If it does not associate, it

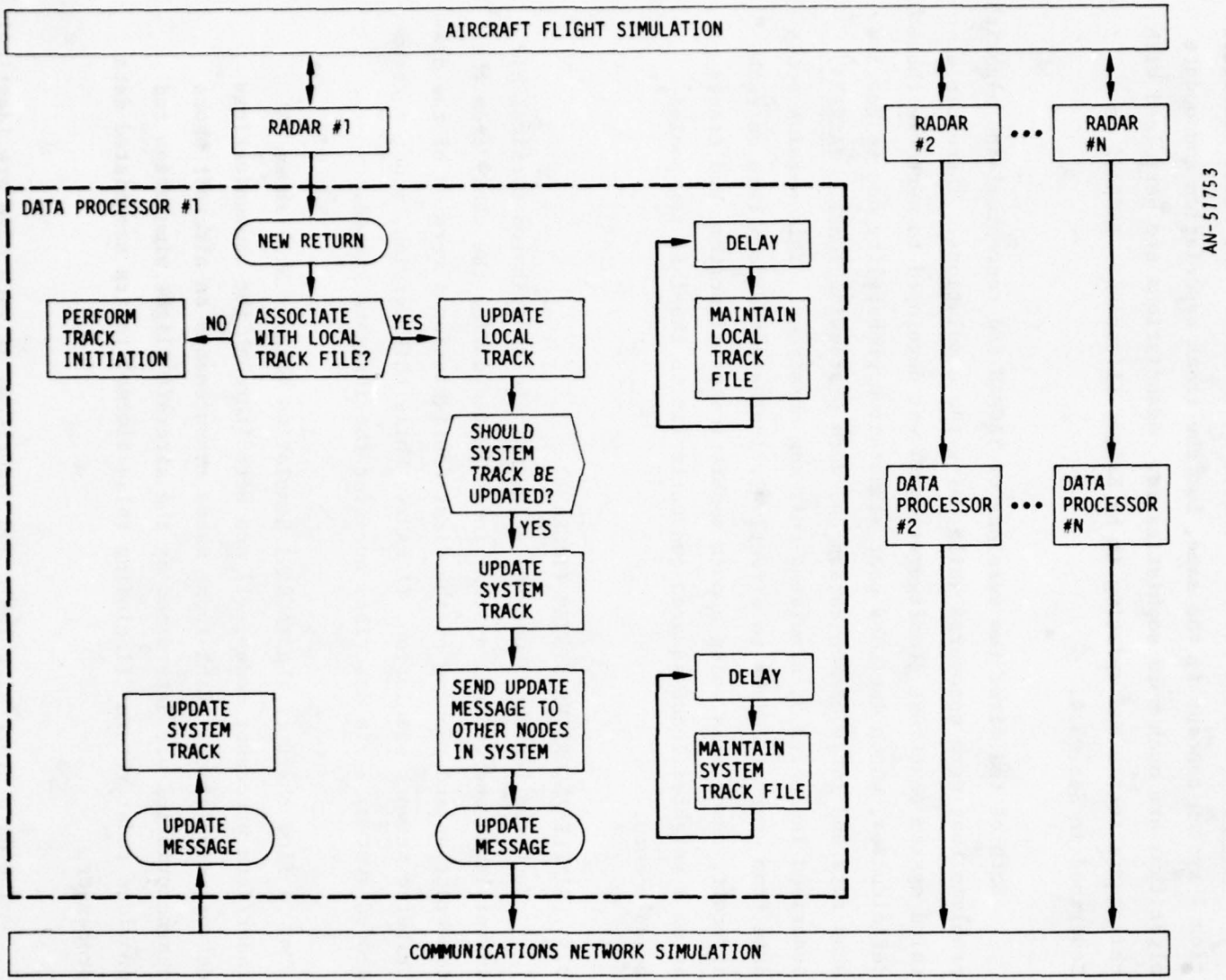


Figure 3.3.1. Flow Chart of TACRAN1 Simulation

becomes a candidate for track initiation. If it associates with a track in the Local Track File, the local track is updated. If a maneuver is detected, the associated system track is updated and an (unaddressed) system message containing the updated system track is sent to all other nodes. When a system update is received from another node, it is used to update the corresponding system track.

Both the Local and System Track Files are maintained periodically. This maintenance includes purging aged tracks that have not been updated recently from the files, and checking the files for duplicates.

Association With Local Track File (Fig. 3.3.2). Association is based on the computed distance between the measurement and each track (extrapolated to the measurement time). If more than one track associates, the one with the smallest distance is chosen. Two association thresholds are used. If the distance-squared is less than the first threshold, A1, then the measurement associates and the aircraft is not maneuvering. If the distance-squared is between A1 and A2, then the measurement associates and the aircraft is maneuvering. Otherwise there is no association.

Track Initiation (Fig. 3.3.3). If a return does not associate with any track in the Local Track File, track initiation is attempted. Initiation into the Local Track File requires only that two measurements on successive scans form a straight-line track with a credible speed.

Local Track Update (Fig. 3.3.4). Track update is performed by a simple α-β recursive filter by the Cartesian equations:

$$\underline{x}(k|k) = \underline{x}(k|k-1) + \alpha(\underline{x}_m - \underline{x}(k|k-1))$$

$$\dot{\underline{x}}(k|k) = \dot{\underline{x}}(k|k-1) + \frac{\beta}{T}(\underline{x}_m - \underline{x}(k|k-1))$$

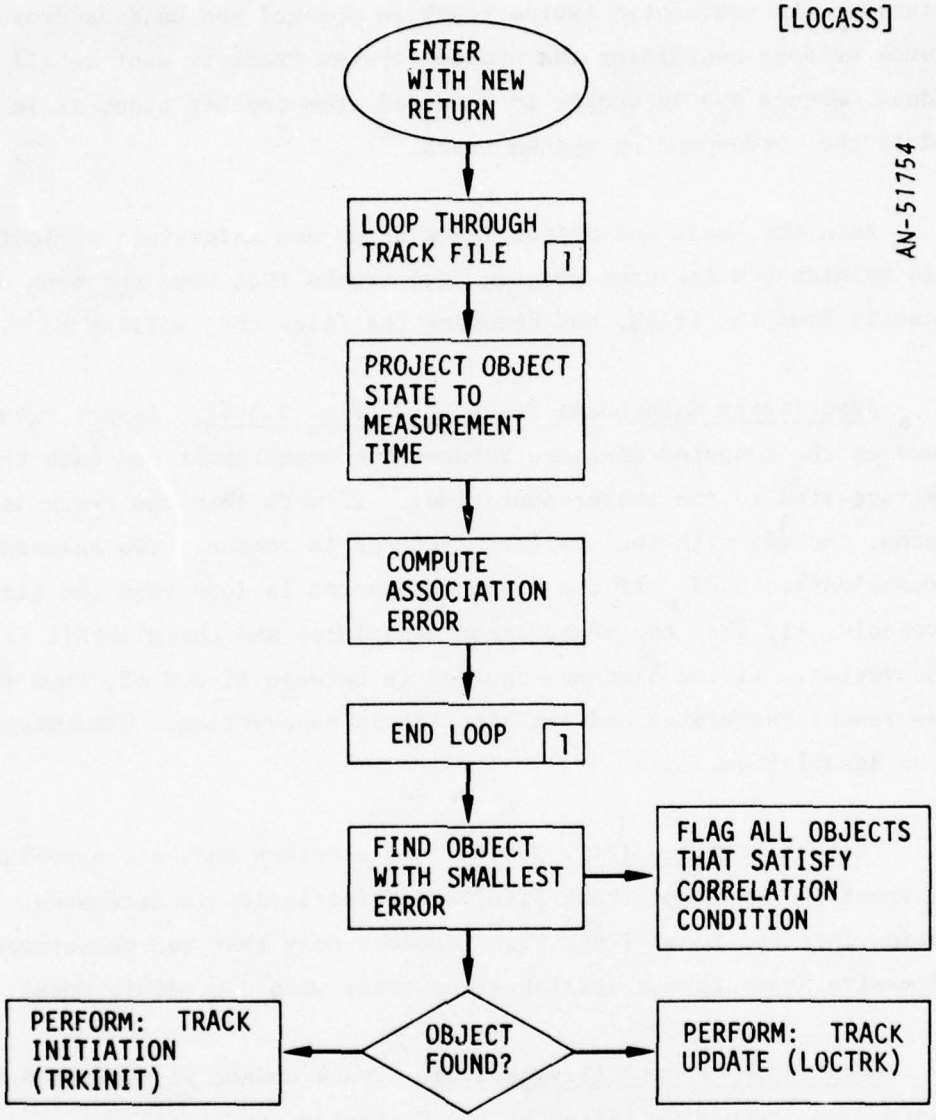


Figure 3.3.2. Association With Local Track File

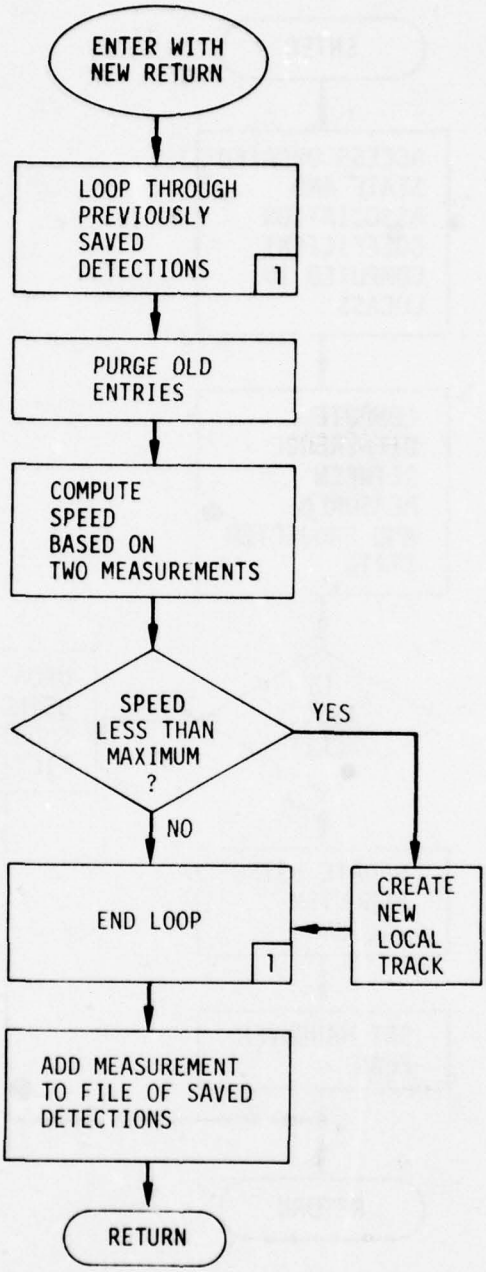


Figure 3.3.3. Track Initiation

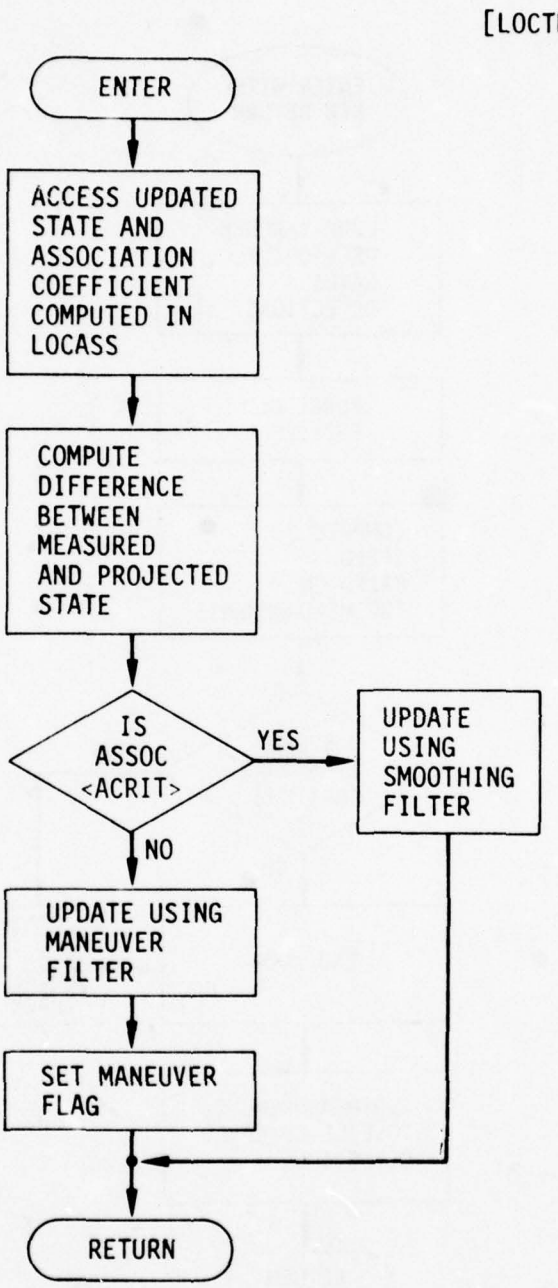


Figure 3.3.4. Local Track Update

where $\underline{x}(k|k)$ = estimated position state at sample time k based on measurements through time k

$\dot{\underline{x}}(k|k)$ = estimated velocity state

$\underline{x}(k|k-1)$ = extrapolated position state at sample time k based on measurements through time k-1

$\dot{\underline{x}}(k|k-1)$ = extrapolated velocity state

$\underline{x}_m$ = measurement position vector

T = time between kth and (k-1)th measurements

The extrapolated states are given by

$$\underline{x}(k|k-1) = \underline{x}(k-1|k-1) + \dot{\underline{x}}(k-1|k-1)T$$

$$\dot{\underline{x}}(k|k-1) = \dot{\underline{x}}(k-1|k-1)$$

If the association distance is small enough that the aircraft is deemed to be not maneuvering, then the "smoothing filter" is used with $\alpha = 0.5$, $\beta = \alpha^2/(2 - \alpha) = 0.17$. Otherwise the "maneuver filter" is used with $\alpha = \beta = 1.0$.

System Track Update Test (Fig. 3.3.5). A new two-measurement local track does not become a new system track until a third measurement associates. This "three-measurement" system track initiation greatly reduces the number of false tracks in the System Track File. If this is the fourth or greater measurement on this track, the system track is updated only if the aircraft is maneuvering based on the test performed during association. It was additionally determined that a maximum interval between updates must be established to prevent the system track from slowly diverging from the local track when the aircraft is not maneuvering or maneuvering very slowly.

Update System Track File from Local Track (Fig. 3.3.6). The system track is updated simply by replacing the old system track with the local

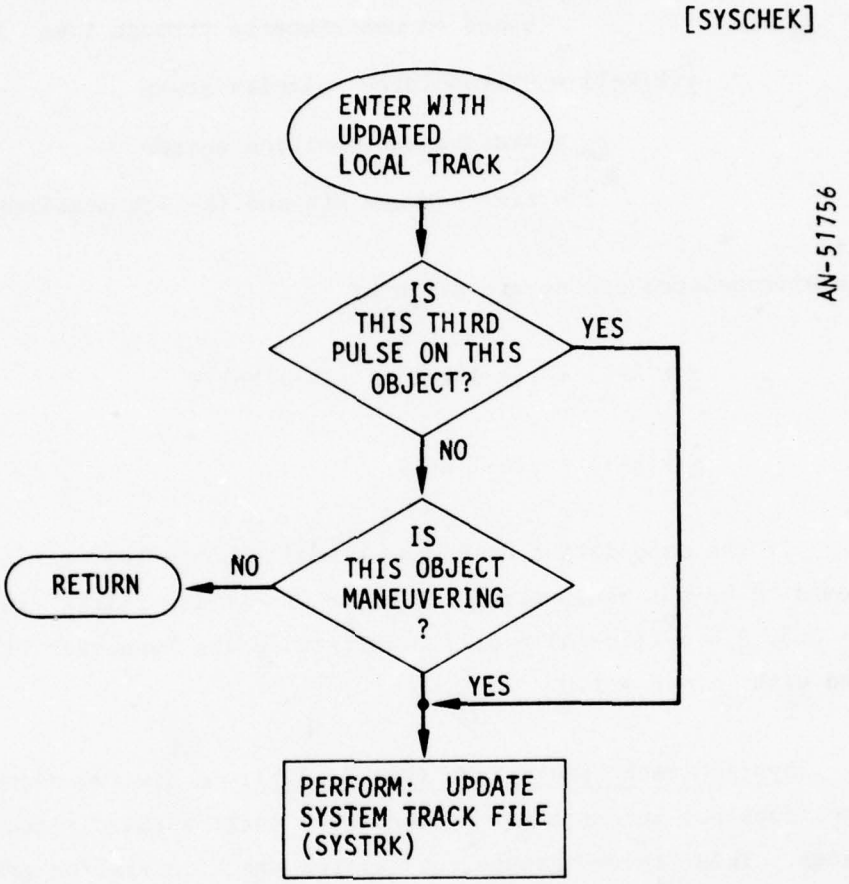


Figure 3.3.5. System Track Update Test

[SYSTRK]

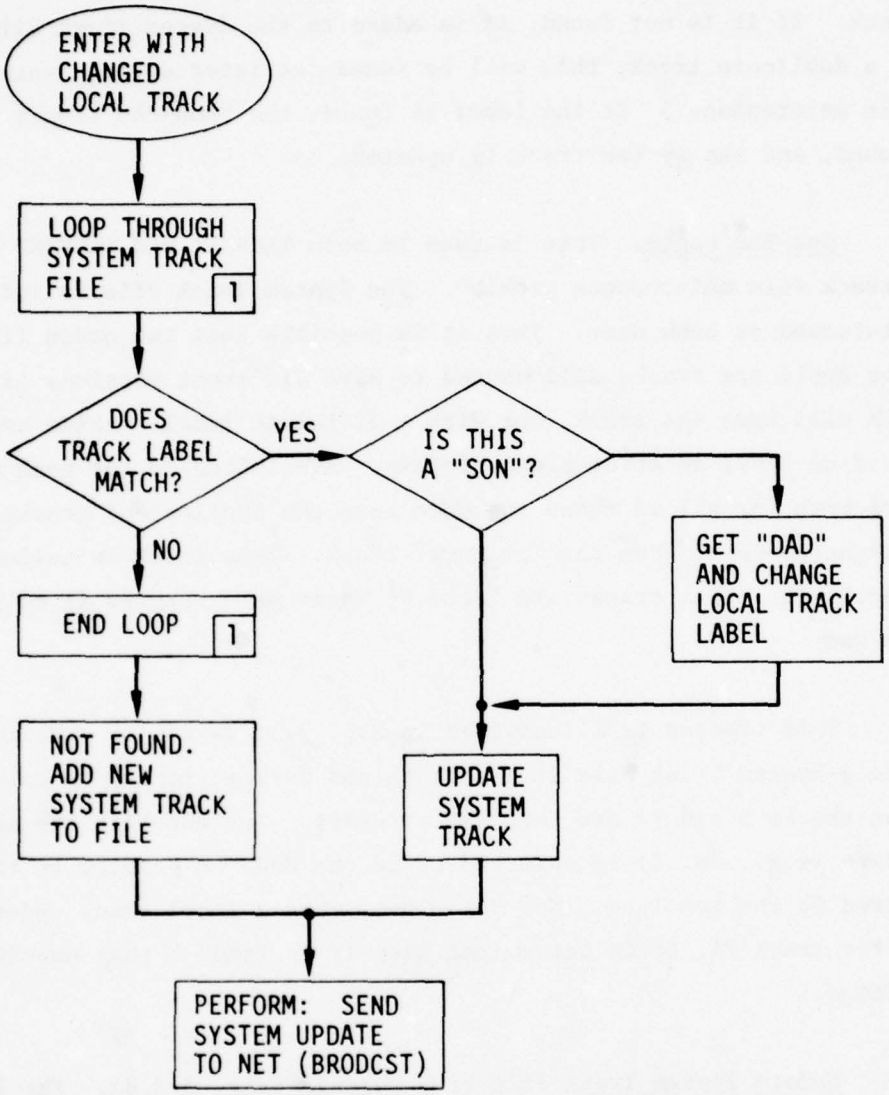


Figure 3.3.6. Update System Track File From Local Track

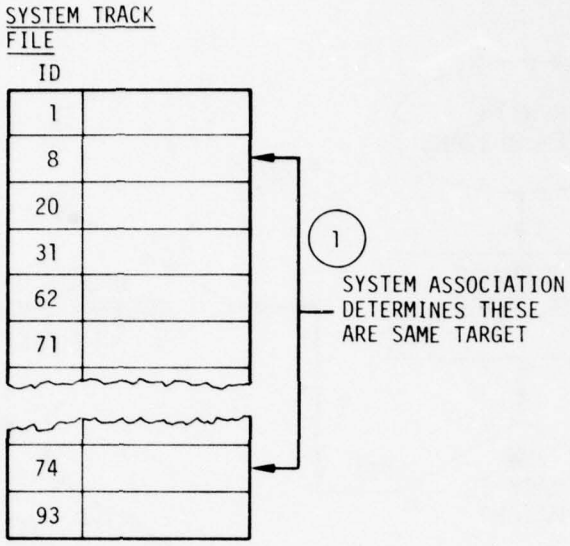
track. First, it is determined if a match between system and local tracks can be made on the basis of the unique track label assigned to each track. If it is not found, it is added to the System Track File. (If it is a duplicate track, this will be found out later during System Track File maintenance.) If the label is found, the "son-dad" logic is performed, and the system track is updated.

Son-Dad Logic. This is used in both TACRAN1 and TACRAN2 to solve a track file maintenance problem. The System Track File is independently maintained at each node. Thus it is possible that two nodes finding the same duplicate tracks will choose to save different versions of the tracks; each will have the track, but with a different label, making association based on label an error-prone process. A solution is for each node to save both (or all if there are more than two duplicates) tracks, and to designate one of them the "primary" track. This track is called the "dad." The other tracks are "sons." Their only purpose is to point to the dad.

This process is illustrated in Fig. 3.3.7. Assume that in one node's System Track File it is determined during track file maintenance that tracks 8 and 74 are the same aircraft. The one with the most recent update (e.g., No. 8) is selected to be the dad. A pointer to the dad is stored in the son (e.g., No. 74). Now, when a local track update comes in for track 74, it is determined that it is track 8 that should be updated.

Update System Track File from Message (Fig. 3.3.8). The logic for updating a system track from a message is similar to updating from a local track. The difference is that there is a possibility that the system track may have already been updated to a more recent time than the update in the message. Therefore, this is checked first.

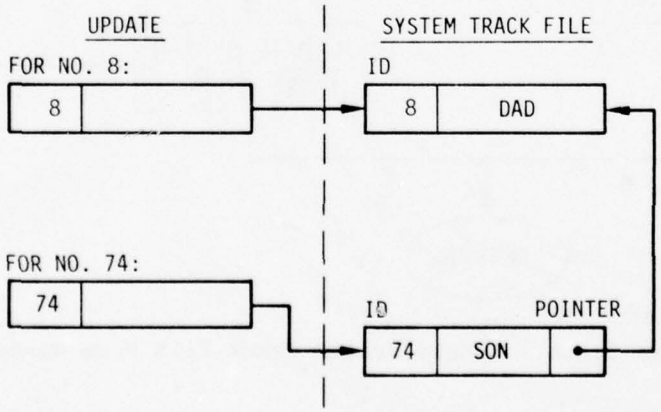
Simulation Test Run. A test run of TACRAN1 was made primarily for the purpose of testing the overall TACRAN simulation structure, and



(2) NEWEST ONE IS SELECTED AS ONE TO BE KEPT IN SYSTEM TRACK (E.G., TRACK NO. 8). CALL THIS ONE THE "DAD".

(3) OLDEST ONE CANNOT BE DISCARDED BECAUSE ITS ID MAY BE KNOWN TO REST OF SYSTEM. SAVE IT AND CALL IT A "SON".

(4) STORE A POINTER TO DAD IN SON:



- UPDATE FOR NO. 8 GOES DIRECTLY TO NO. 8
- UPDATE FOR NO. 74 GOES TO NO. 74, FINDS POINTER TO 8, AND GOES TO NO. 8

Figure 3.3.7. "Son" and "Dad" Relationship

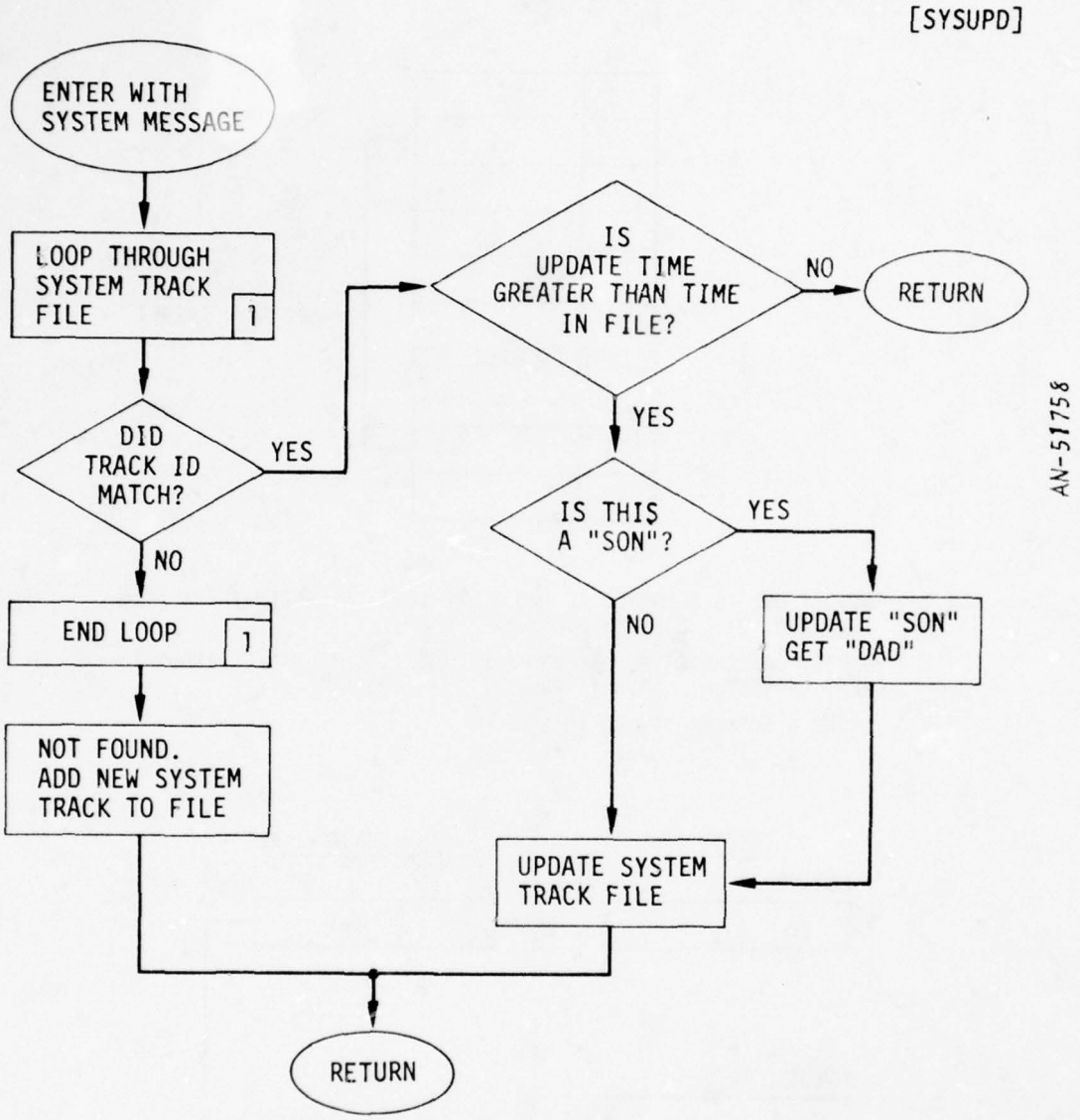


Figure 3.3.8. Update System Track File From Message

secondarily to test the tactical data processor logic. A run consisting of three modes, three communication links, and two aircraft was made. The results showed that the simulation worked well, but some deficiencies showed up in the logic. These deficiencies were in the areas of association, track file maintenance, and the system track updating algorithm. Rather than spend time correcting the deficiencies, it was decided to construct a new version, which is now called TACRAN2.

3.4 KALMAN-FILTER SYSTEM (TACRAN2)

The original purpose of the second version of the TACRAN simulation was to create a standard by which to compare other systems with simpler algorithms. Thus the best known algorithms were included in TACRAN2, including chi-square statistical tests for association and Kalman filtering for track update. System tracks are updated by combining all existing data weighted by covariances, rather than by simply replacing old data with the newest data, as in TACRAN1. Other improvements were added such as a short delay between association and update so that the "best" of several measurements could be used for update, rather than simply the first to associate.

Otherwise, the basic system concepts in TACRAN2 are the same as in TACRAN1: All track initiation, association and update are performed using the Local Track File. System Tracks, which are replicated at all nodes, are updated only whenever a maneuver is detected. Thus, TACRAN2 is also a Configuration 1 system.

The system configuration simulated by TACRAN2 has certain flaws that make it also not a recommended construct. Both the communication bandwidths and data processing loads are excessive, the data rate is too low for effective association of measurements with tracks, the chi-square tests are somewhat superfluous when the major association errors arise from a basically non-statistical source (maneuvering), and the "nearest return" association algorithm does not properly handle multiple association.

Overview. A flow chart of TACRAN2 is shown in Fig. 3.4.1. The data processing logic is described in the dashed box labelled "data processor #1". This logic is very similar to that used in TACRAN1.

The concept of an "action sequence chain" (A.S.C.) is introduced in the figure. This concept, described in Sec. 3.6.5, permits a sequence of actions to be performed on a single dataset (a measurement or a message in the present context) with delays permitted during the sequence.

A new radar return is processed by the NEWRET A.S.C. Association with the Local Track File is first. If unsuccessful, track initiation is attempted. Otherwise (after a 0.2-second delay to permit other candidate association with the same track to occur), the NEWUPD A.S.C. is entered to process the track.

The local track is updated using a Kalman filter on the measurement that is "closest" (using a chi-square distributed statistical distance) to the track. Next, it is determined if the system track needs updating, based primarily on maneuver detection. If so, it is updated and the new system track is sent to all other nodes for updating.

When an update message is received, the RCVCOMM A.S.C. is entered, and the system update is performed.

Both Local Track File and System Track File are performed periodically in the TLMAINT A.S.C. and the TSMAINT A.S.C., respectively.

Local Track File Association (Fig. 3.4.2). A new measurement is first transformed from radar coordinates (range, azimuth, elevation) to Cartesian coordinates (x,y,z), the system in which all computations are performed. Each target in the Local Track File is extrapolated to the measurement time, after which a coarse association test is performed. This test is passed if

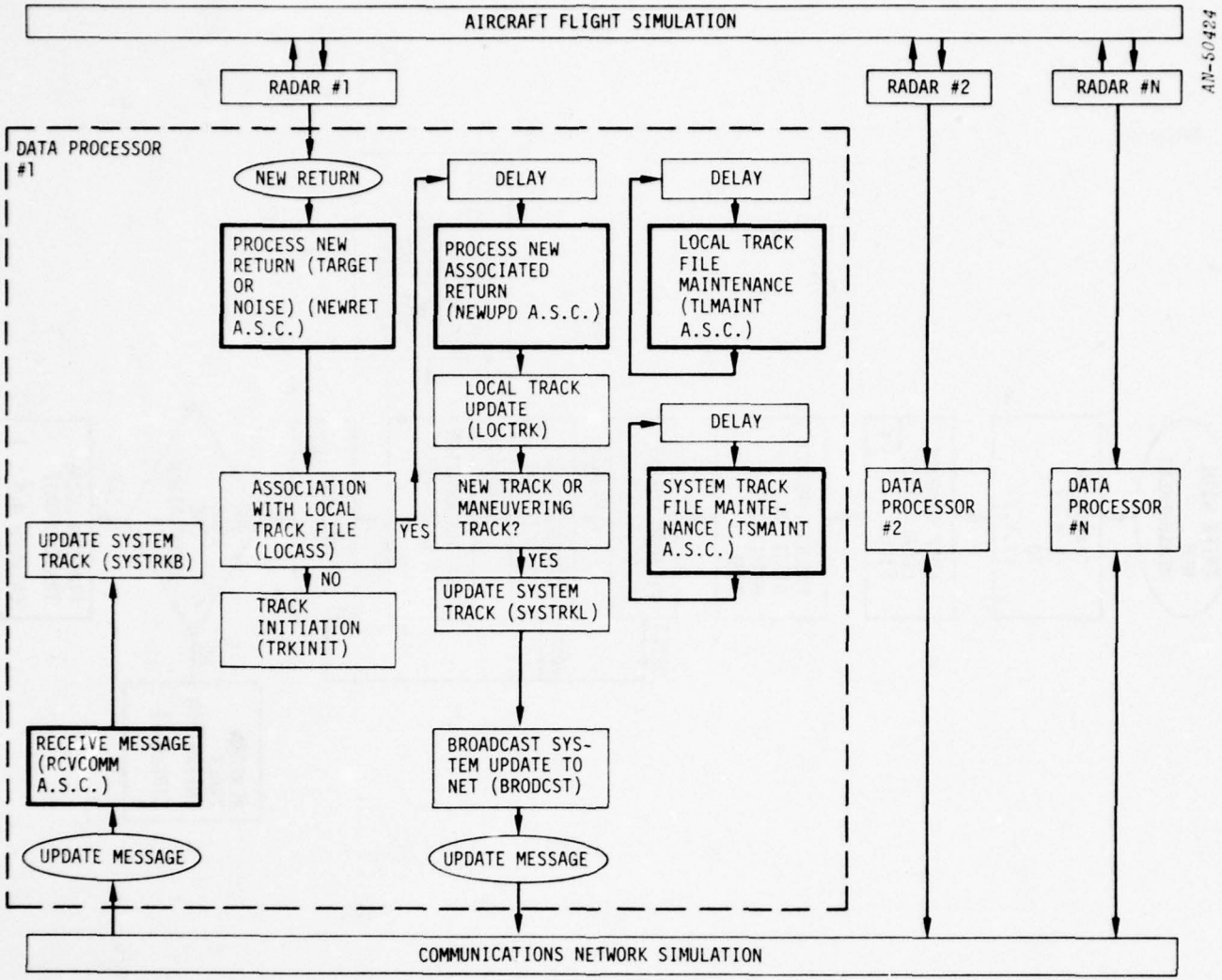


Figure 3.4.1. Flow Chart of TACRAN2 Simulation

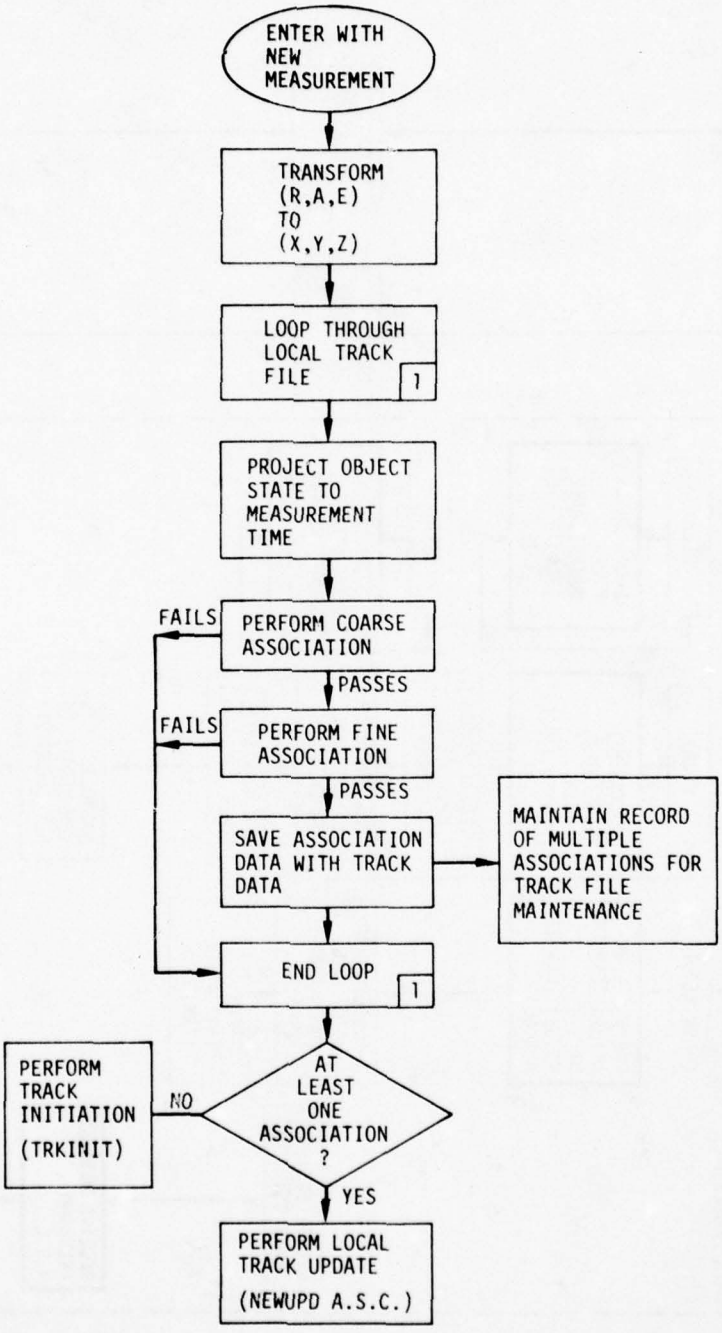


Figure 3.4.2. Association With Local Track File

$$|x_m - \hat{x}_p| < \Delta x$$

$$|y_m - \hat{y}_p| < \Delta y$$

$$|z_m - \hat{z}_p| < \Delta z$$

where (x_m, y_m, z_m) is the measurement $(\hat{x}_p, \hat{y}_p, \hat{z}_p)$, is the extrapolated position of the track, and $(\Delta x, \Delta y, \Delta z)$ is the threshold.

If the coarse test is passed, a fine test is made:

$$d^2 = \underset{\sim}{r}^T V^{-1} \underset{\sim}{r} < d_t^2$$

where d^2 is the chi-square distributed "statistical distance," $\underset{\sim}{r}$ is the measurement residual vector given by

$$\underset{\sim}{r} = \underline{r}_m - \underline{\hat{r}}_p \qquad \text{(measurement residual)}$$

$$\underline{r}_m = \begin{bmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{bmatrix} \qquad \text{(measurement)}$$

$$\hat{r}_p = \begin{bmatrix} \text{extrapolated range} \\ \text{extrapolated azimuth} \\ \text{extrapolated elevation} \end{bmatrix} \qquad \text{(extrapolated position in radar coordinates)}$$

and V is the covariance matrix of the measurement residual (defined later). The quantity d_t^2 is the threshold. Since the probability distribution of d^2 is known (chi-square), the threshold can be set so that a given probability of passing the test can be selected, as described in Sec. 2.3.1.1.

Each successful association is saved with the track data so that all successful associations with a given track will be available when that track is updated.

Track Initiation. The logic for track initiation is the same as described on page 269 (Fig. 3.3.3) for TACRAN1. Two measurements are required for local track initiation. In Cartesian coordinates these measurements are

$$\underline{s}_m(1) = \begin{bmatrix} x_m(1) \\ y_m(1) \\ z_m(1) \end{bmatrix} = \text{first measurement}$$

$\underline{s}_m(2)$ = second measurement T seconds later

The estimated state at sample time 2 given two measurements is

$$\hat{\underline{s}}(2|2) = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} x_m(2) \\ y_m(2) \\ z_m(2) \\ \left(x_m(2) - x_m(1)\right)/T \\ \left(y_m(2) - y_m(1)\right)/T \\ \left(z_m(2) - z_m(1)\right)/T \end{bmatrix}$$

The covariance matrix of the estimated state is given by

$$P(2|2) = \left[\Phi^{-T}H^T(1)M(1)H(1)\Phi^{-1} + H^T(2)M(2)H(2)\right]^{-1}$$

where

$$\Phi = \text{state transition matrix} = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H(k) = \text{Jacobian} = \begin{bmatrix} x/r & y/r & z/r & 0 & 0 & 0 \\ \dfrac{-y}{(x^2 + y^2)} & \dfrac{x}{(x^2 + y^2)} & 0 & 0 & 0 & 0 \\ \dfrac{-xz}{r^2\sqrt{x^2 + y^2}} & \dfrac{-yz}{r^2\sqrt{x^2 + y^2}} & \dfrac{\sqrt{x^2 + y^2}}{r^2} & 0 & 0 & 0 \end{bmatrix}$$

$$r = \sqrt{x^2 + y^2 + z^2}$$

(x,y,z) = estimated states at time k centered at the radar

$$M(k) = \text{measurement covariances (at time k)} = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_A^2 & 0 \\ 0 & 0 & \sigma_E^2 \end{bmatrix}$$

$(\sigma_r^2, \sigma_A^2, \sigma_E^2)$ = measurement variances of range, azimuth, elevation

Local Track Update (Fig. 3.4.3). If several measurements associate with this track, the one with the smallest statistical distance is selected

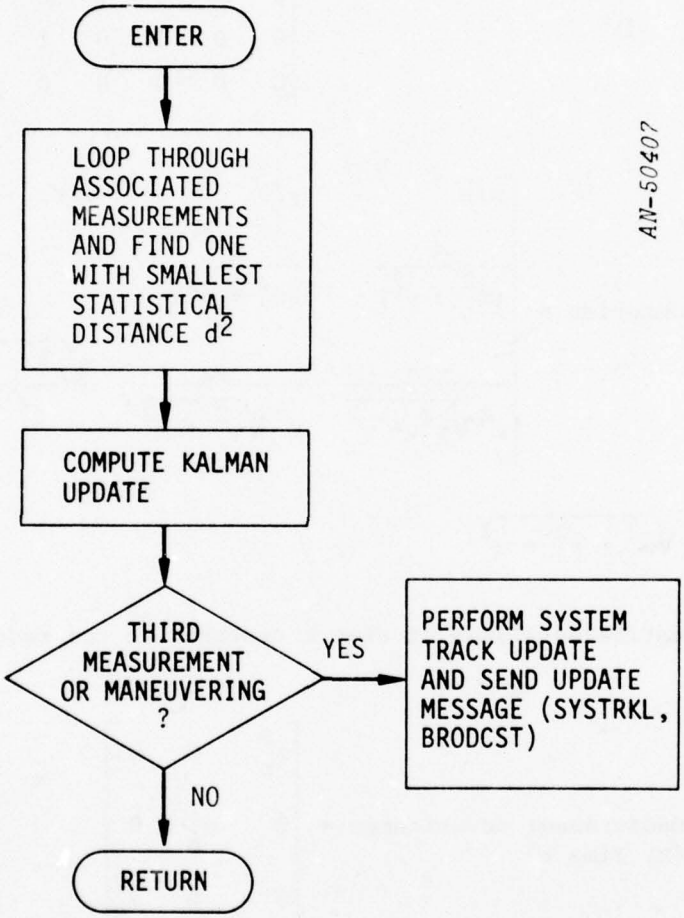


Figure 3.4.3. Local Track Update

for track update. Track update is by an extended Kalman filtering of position and velocity, with model (or "maneuver") noise added, as discussed in Sec. 2.4.1.

The state update equation is

$$\underline{\hat{s}}(k|k) = \underline{\hat{s}}(k|k-1) + K(k)\underline{\tilde{r}}(k)$$

where $\underline{\hat{s}}(k|k)$ = estimate of true state $\underline{s}(k)$ given k measurements

$\underline{\hat{s}}(k|k-1)$ = estimate of $s(k)$ given $k-1$ measurements

$K(k)$ = Kalman gain matrix

$\underline{\tilde{r}}(k)$ = measurement residual (defined previously)

The quantity $\underline{\hat{s}}(k|k-1)$ is the extrapolation of the state estimate at time $k-1$, $\underline{\hat{s}}(k-1|k-1)$. It is given by

$$\underline{\hat{s}}(k|k-1) = \Phi(k-1)\underline{\hat{s}}(k-1|k-1)$$

where Φ is the state transition matrix defined previously.

The Kalman gain matrix $K(k)$ is computed from the extrapolated error covariance $P(k|k-1)$ by the following sequence of computations:

$$P(k|k-1) = \Phi(k-1)P(k-1|k-1)\Phi^T(k-1) + \Gamma Q(k-1)\Gamma^T$$

where $P(k-1|k-1)$ = covariance of $\underline{\hat{s}}(k-1|k-1)$

$P(k|k-1)$ = covariance of $\underline{\hat{s}}(k|k-1)$

$$\Gamma Q \Gamma^T = \sigma_m^2 \begin{bmatrix} T^4/4 & 0 & 0 & T^3/2 & 0 & 0 \\ 0 & T^4/4 & 0 & 0 & T^3/2 & 0 \\ 0 & 0 & T^4/4 & 0 & 0 & T^3/2 \\ T^3/2 & 0 & 0 & T & 0 & 0 \\ 0 & T^3/2 & 0 & 0 & T & 0 \\ 0 & 0 & T^3/2 & 0 & 0 & T \end{bmatrix}$$

σ_m^2 = variance of the model (or maneuver) noise

Next the covariance of the measurement residual, $V(k)$, is computed:

$$V(k) = H(k)P(k|k-1)H^T(k) + M(k)$$

These matrices were previously defined. The Kalman gain matrix is then computed by

$$K(k) = P(k|k-1)^T H^T(k) V^{-1}(k)$$

Finally, the error covariance is updated by

$$P(k|k) = [I - K(k)H(k)]P(k|k-1)$$

The last step in local track update is to determine if the system track needs updating. As in TACRAN1, system update is performed if it is a new system track (i.e., a third measurement has been received on a just-initiated local track) or the aircraft is deemed to be maneuvering, determined (incorrectly) by the statistical distance being greater than a threshold.

Update System Track from Local Track (Fig. 3.4.4). If the system track is to be updated from the local track, it is first determined if this is a new 3-measurement local track (track label = 0). If so, an association with the System Track File is performed. If this association (described later) is successful, this aircraft is already in the System Track File, and the data is used to update the system track. Otherwise a new system track is created.

If this is not a new track (track label $\neq$ 0), it is determined if this track label is in the System Track File. If so, the "son-dad" algorithm (described previously in Sec. 3.3) is performed and the system track is updated. Otherwise an association is attempted.

The equations for updating the System Track File are used to smooth the two tracks together, rather than simply replace the system track with a newer track. First the older track $\hat{\underline{s}}_1$ and its covariance P_1 are extrapolated to the time of the newer track, represented by state $\hat{\underline{s}}_2$ and covariance P_2. Next the covariances are combined:

$$P_c = \left[P_1^{-1} + P_2^{-1}\right]^{-1}$$

and finally the states are combined:

$$\hat{\underline{s}}_c = P_c\left[P_1^{-1}\hat{\underline{s}}_1 + P_2^{-1}\hat{\underline{s}}_2\right]$$

Association With System Track File (Fig. 3.4.5). Association of a local track with the System Track File involves both a coarse and a fine test. The coarse test compares the magnitude of the difference in each of the six state dimensions to a threshold:

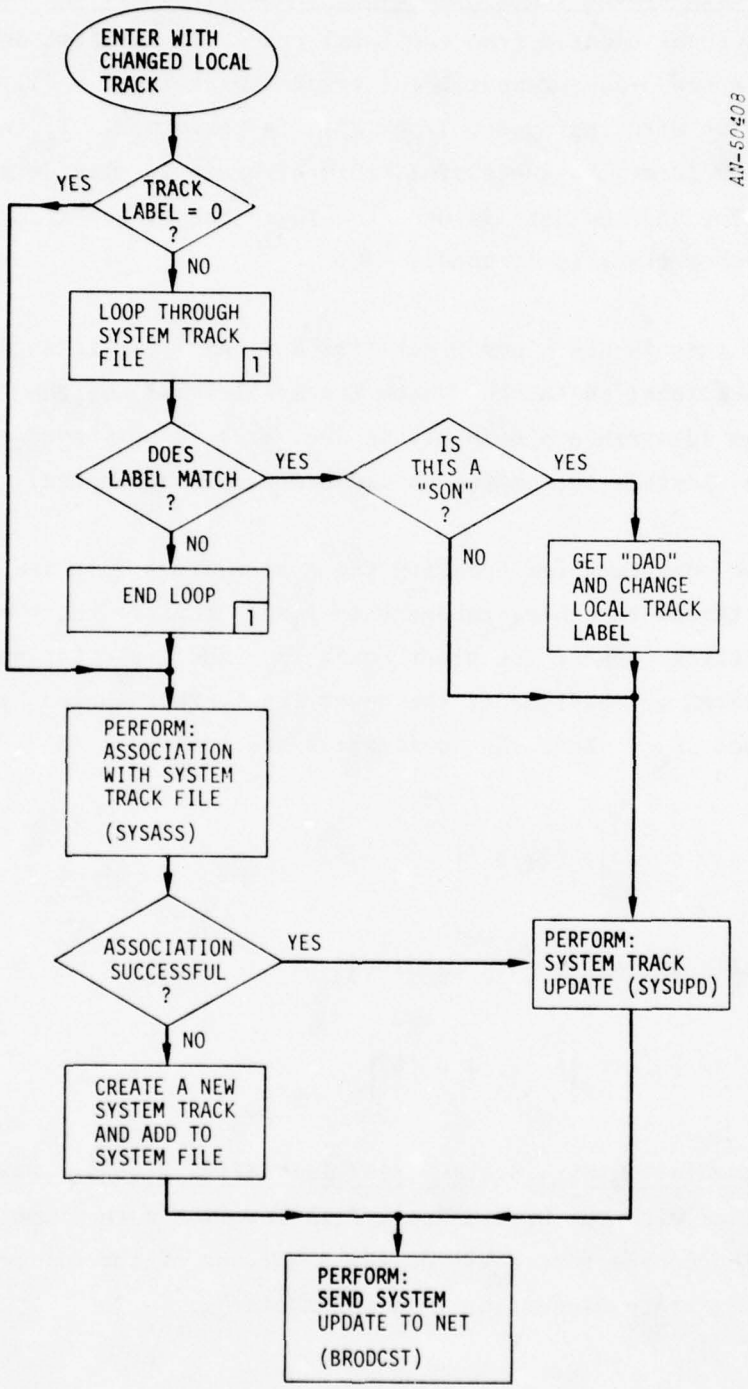


Figure 3.4.4. Update System Track From Local Track

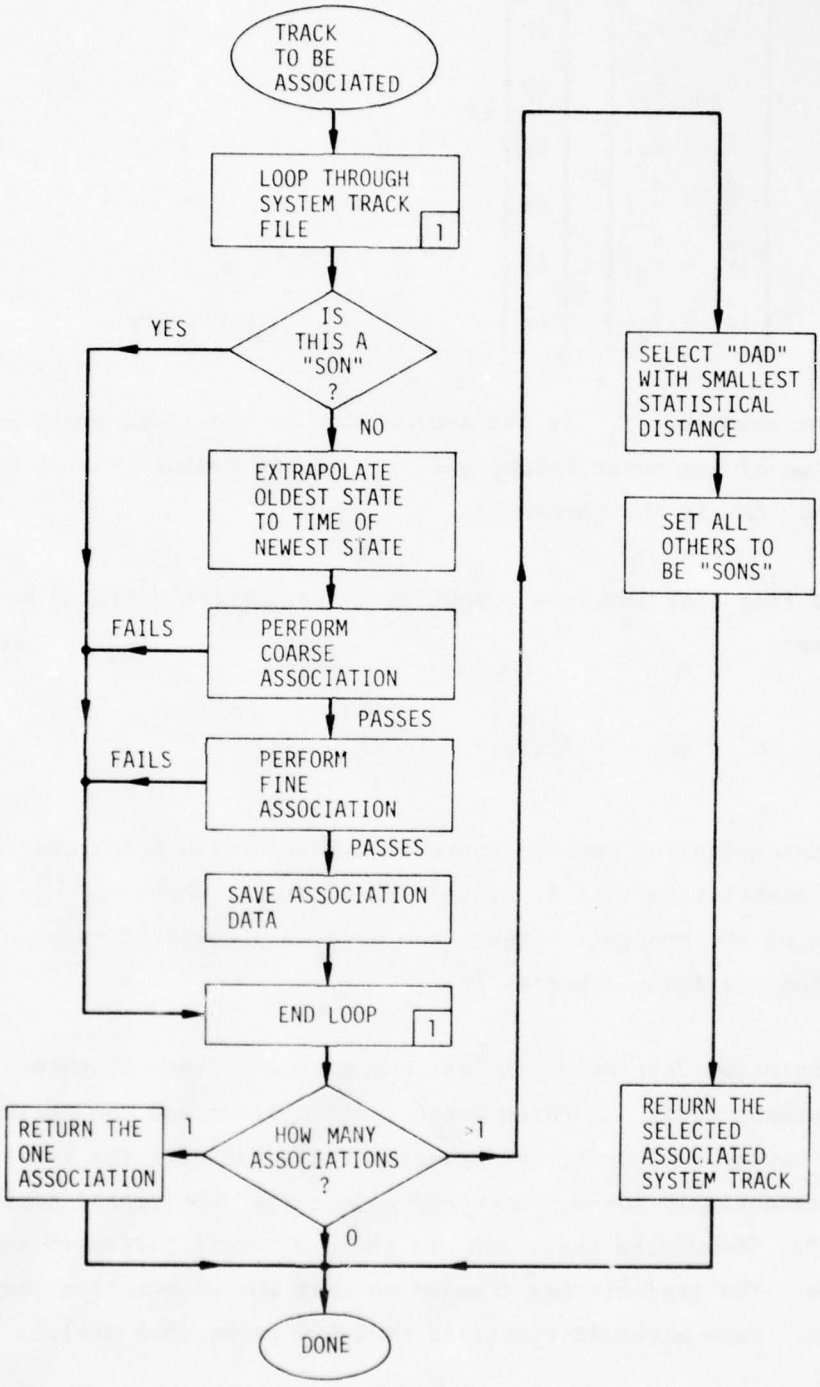
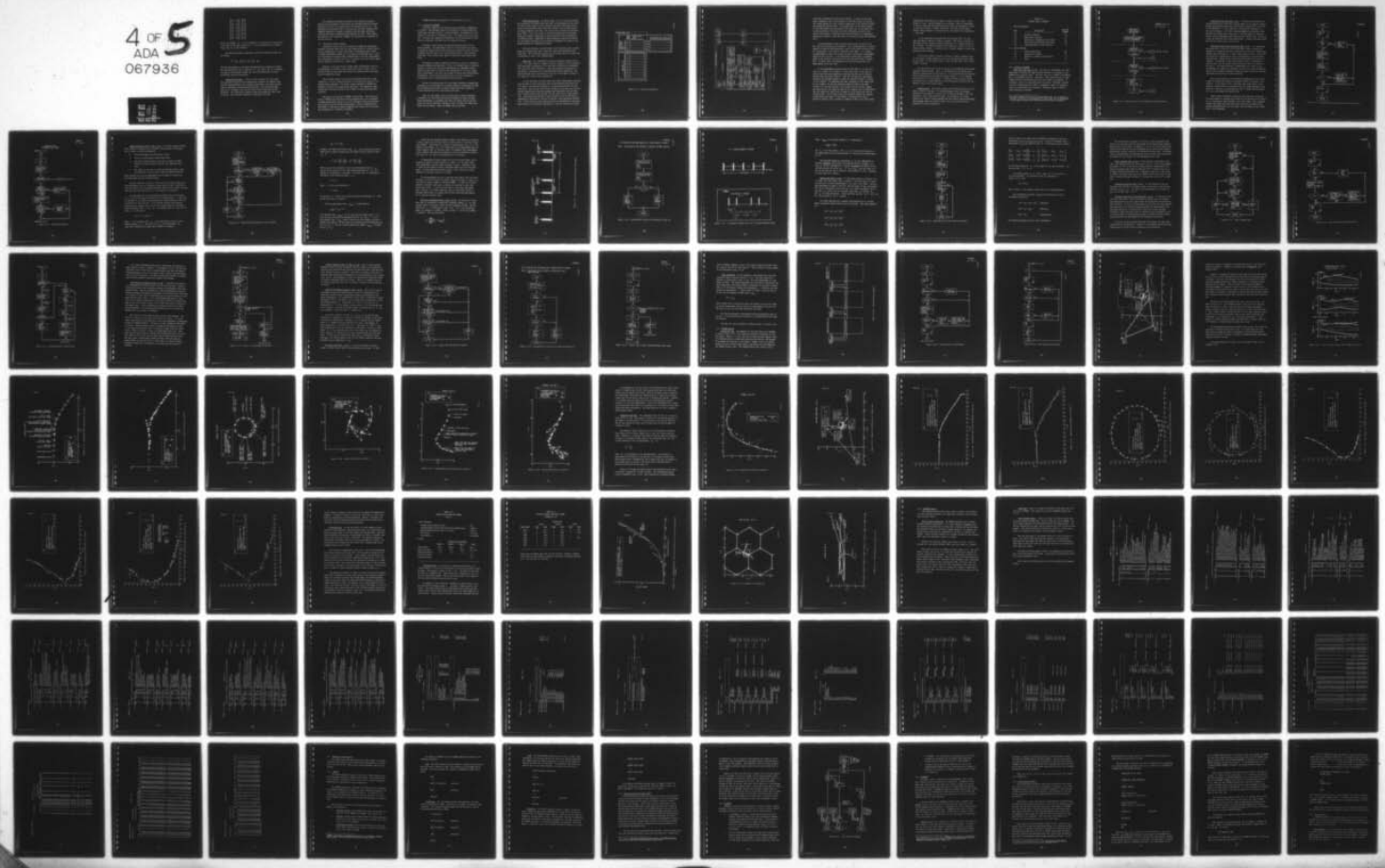


Figure 3.4.5. Association With System Track File

AD-A067 936 GENERAL RESEARCH CORP SANTA BARBARA CALIF F/G 17/9
TACTICAL FORWARD AREA SURVEILLANCE AND CONTROL INTERNETTING STU--ETC(U)
NOV 78 G W DELEY, J H BALLANTINE F19628-78-C-0020
UNCLASSIFIED CR-1-814-VOL-2 ESD-TR-78-180-VOL-2 NL

4 OF 5
ADA
067936



$$\begin{bmatrix} |\hat{x}_1 - \hat{x}_2| \\ |\hat{y}_1 - \hat{y}_2| \\ |\hat{z}_1 - \hat{z}_2| \\ |\hat{\dot{x}}_1 - \hat{\dot{x}}_2| \\ |\hat{\dot{y}}_1 - \hat{\dot{y}}_2| \\ |\hat{\dot{z}}_1 - \hat{\dot{z}}_2| \end{bmatrix} < \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta \dot{x} \\ \Delta \dot{y} \\ \Delta \dot{z} \end{bmatrix}$$

where, for example, $\hat{x}_1$ is the x-dimension of the older track extrapolated to the time of the newer track, and $\hat{x}_2$ is the x-dimension of the newer track, and Δx is the threshold.

The fine test involves computing a statistical distance between the two states:

$$d^2 = (\hat{\underline{s}}_1 - \hat{\underline{s}}_2)^T (P_1 + P_2)^{-1} (\hat{\underline{s}}_1 - \hat{\underline{s}}_2)$$

This distance-squared is chi-square distributed with 6 degrees of freedom. The fine association test is passed if $d^2 < d_t^2$, where d_t^2 is a threshold set based on the tradeoff between the desired probabilities of correct association and false association.

Simulation Test Runs. A test run with a network of three radar/data processor nodes and three communication links was run against three aircraft having different, but crossing trajectories. The first aircraft flew an essentially non-maneuvering trajectory. The second performed about a 5g, 360-degree turn, and the third aircraft performed several maneuvers. The trajectories crossed so that the association logic could be tested. Each aircraft travelled at 2,000 km/hr (556 m/s).

The simulation performed as expected, with additional insights gained into the problems of association and tracking maneuvering targets. Tracking results with the Kalman filter are presented in Sec. 2.4. The weaknesses of association with a low data rate, the large communications bandwidth required to transmit covariances, and the large processing loads dictated that an improved system approach be taken. Thus, again a new version of the simulation, TACRAN3, was constructed, as described in the next section.

3.5 POOLED-DATA SYSTEM (TACRAN3)

Experience with the first two versions of TACRAN and considerable parametric analysis showed that the only viable method for greatly improving the probability of correct association of measurements with tracks of highly maneuverable targets is to increase the measurement rate. Since the track-while-scan radars being considered have a low scan (measurement) rate (measurement intervals from 4 to 12 seconds), data from several radars must be combined (pooled) into a single track.

However, in the basic system concept under consideration, there is an embarassment of riches; so many radars can see most targets that if all of the data on each target were pooled the communications requirement would be undesirably large.

Thus a system concept is required that simultaneously has a high probability of correct association while at the same time requires a low communications bandwidth. The system concept simulated by TACRAN3 was designed to satisfy these conflicting objectives. The simulation demonstrated that this major objective and a number of other objectives were satisfied.

Association is improved in TACRAN3 in other ways. The statistical tests in TACRAN2 were discarded as being inappropriate in an environment of highly maneuverable targets and in the assignment algorithm (described later) used to resolve multiple associations. Track is simplified and performance is improved by using a weighted least squares fit of the last few measurements to a quadratic polynomial.

TACRAN3 simulates Configuration 2 as defined in Sec. 1.3.2.

3.5.1 Overview of TACRAN3

The basic elements of the system concept simulated in TACRAN3 are as follows: The principal output of the system is a System Track File (STF) whose required accuracy, a system parameter, is assumed to be low, perhaps on the order of a kilometer or so. All nodes have a copy of the STF. To reduce the communication bandwidth, the STF is updated as infrequently as possible consistent with the required accuracy.

Tracking of each aircraft is performed cooperatively by a few selected radars. The measurements from the selected radars are pooled to provide a single track on each aircraft at an effective data rate that is higher than the scan rate of the individual radars. Each tracking radar has a copy of this track, which is called the "Distributed Local Track."

The number of radars tracking each aircraft is kept to a minimum to minimize communication bandwidth. The set of tracking radars is carefully selected to ensure that the pooled measurements are reasonably evenly spaced in time, since it does not help association if all of the radars make a measurement at the same time causing an entire scan period between measurements. Other factors described later are also considered in selecting the set of tracking radars for each aircraft.

The set of tracking radars for each target dynamically changes with time based on criteria which are evaluated whenever a measurement is made. The determination and control of which radars track which targets is totally distributed with no centralized controller.

Thus, for each target there are three types of nodes (radar/data processors); (1) those that are actively tracking the target, pooling their measurements to accomplish this task, (2) those that can see the target but are not presently tracking it, and (3) those that cannot presently see the target. Nodes can change type at any time.

Track File Structure. As shown in Fig. 3.5.1, at each node (radar/data processor) the System Track File (STF) is partitioned into two parts: (1) those tracks being actively tracked by this node [this partition is called the "Distributed Local Track File" (DLTF)], and (2) those tracks not being tracked by this node. The latter file is called the "Non-Track File" (NTF). All of the tracks in the entire system are in the System Track File; however, membership in the two partitions will be different at different nodes. Regardless of which partition a particular track is in, the system track data will be identical throughout the system to the extent permitted by communication delays.

The first partition, the Distributed Local Track File, has, in addition to the system track data, other data called the Distributed Local Track data. The latter data contains the up-to-date track on a target and is identical at each of the tracking nodes.

Logic Flow. The (simplified) system flow diagram is shown in Fig. 3.5.2. The figure depicts a number of radars connected to data processors. Each radar makes measurements on aircraft whose positions are determined by the Aircraft Flight Simulation. Each data processor is connected to the network, represented by the Communications Network Simulation. The data processing logic implemented at each node of the system is shown in the dashed box labeled "Data Processor #1."

At each radar/data processor node a new return (measurement) obtained from the radar is first associated with the Distributed Local Track File. If it doesn't successfully associate it is next associated with the Non-Track File. This two-part association, designed to save both data processing and communication resources, assumes that association with the DLTF, which is more accurate than the NTF, will be correct with a high probability. Association with the NTF is more difficult since the NTF track is sometimes not sufficiently accurate to provide a high association probability. If more than one track associates with the measurement (a situation which should occur relatively infrequently), then this node does not have

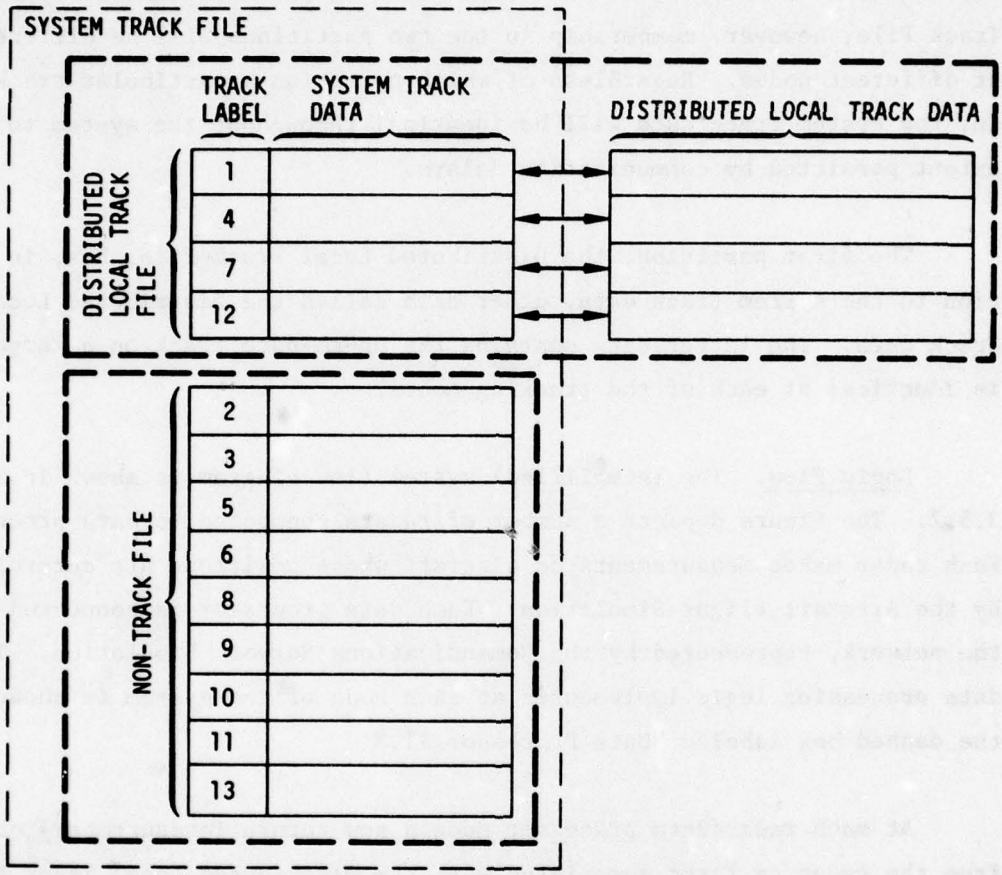


Figure 3.5.1. Track File Structure

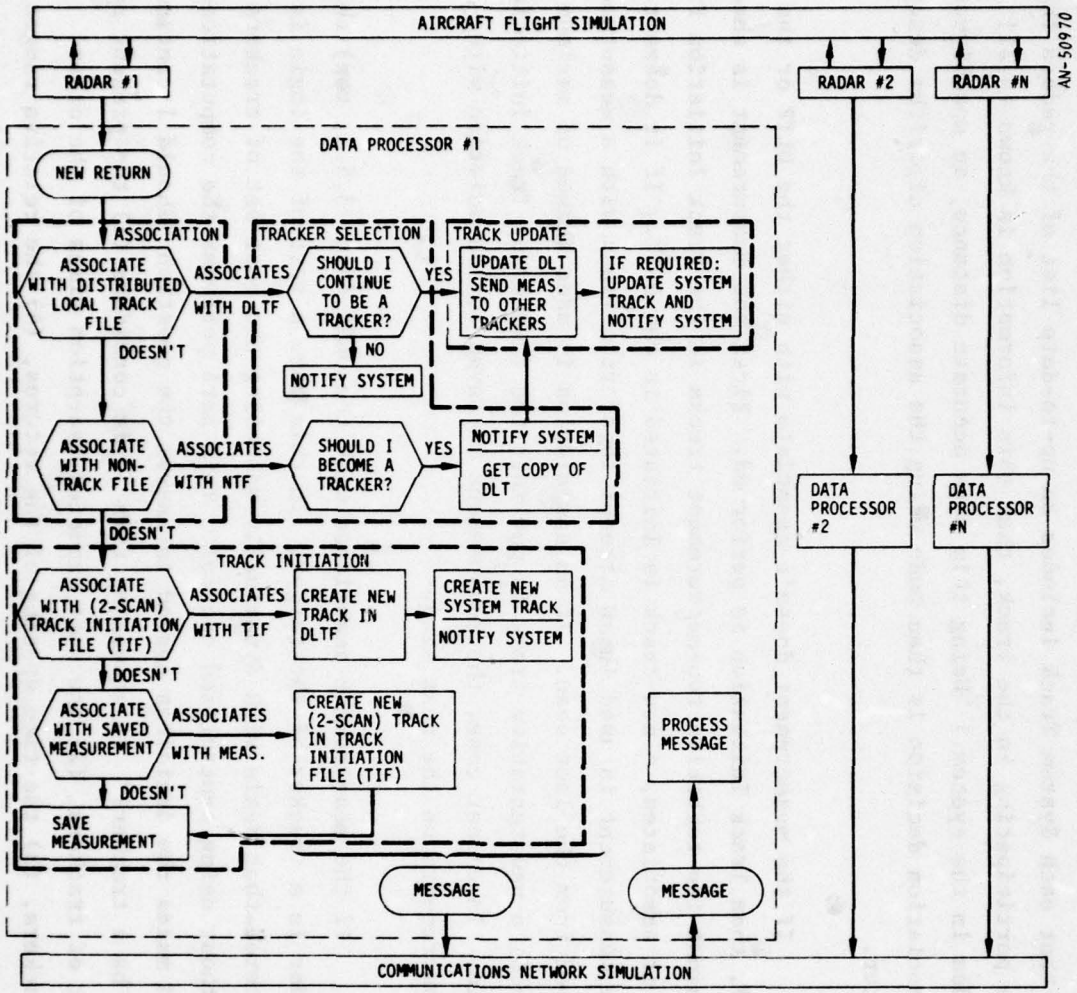


Figure 3.5.2. Flow Chart of TACRAN3 Simulation

sufficient information to resolve the conflict. In this case the node sends a message to one of the trackers of each track within the association volume asking for a more accurate computation of the distance between the measurement and the track. (An integral part of the system concept is that each System Track includes an up-to-date list of the radars that are participating in the track, thus this information is known to all nodes in the system.) Using this more accurate distance, an unambiguous association decision is then made using the association algorithm described later.

If the measurement doesn't associate with either the DLTF or the NTF, then Track Initiation is performed. First the measurement is checked against the tentative two-measurement tracks in the Track Initiation File. If it associates, a new track is initiated in the DLTF. If it doesn't, the measurement is used in an attempt to start a track with a measurement saved from the last scan. If an association is made (based on maximum speed) a new tentative track is put into the (two-scan) Track Initiation File. In either case, the measurement is saved for association with measurements on the next scan.

If the measurement associates with the DLTF (Fig. 3.5.2, top) this radar is a tracker of the target. In this case a part of the logic is entered that deals with dynamically selecting the best set of trackers without using centralized control. This part performs the computations and makes the decisions needed to answer the question "Should I continue to be a tracker?" In answering this logic considers (1) the present number of trackers, (2) the next expected reception times of the other trackers, (3) the time spacing of the returns, (4) the relative ranges and range rates of the target as seen by all trackers, and (5) expectation that the target will soon be out of range or in the blind cone over the tracking radar. All of these factors are evaluated without any communications with other nodes. If the answer to the question is "No, I shouldn't continue to be a tracker," a message is sent to all other nodes

advising that this radar is no longer a tracker of this target. If the answer is "Yes, I should continue to be a tracker," then the measurement is used to update the Distributed Local Track and the measurement is sent to the other trackers to update their DLTs. The node can also send a special "Help" message if it would like for some reason to be replaced by another node.

If the measurement associates with the Non-Track File, the node asks the question "Should I become a tracker?" This logic is similar to the logic used by a present tracker to determine if it should continue to be a tracker. The node may decide to add itself as a tracker, or to replace a present tracker. In either case, it sends a message to every node advising of this decision.

If the node decides to become a tracker, it sends a message to one of the present trackers asking for a copy of the Distributed Local Track on the target. On receiving the DLT it uses the same logic as if it were already a tracker.

The Distributed Local Track is updated by the weighted least-squares algorithm described later. Next, it is determined if the System Track should be updated based on two criteria: (1) the System Track has deviated from the Distributed Local Track by a given distance, or (2) a maximum time (a minute or so) has elapsed. If either of these criteria are met the System Track is made identical to the Distributed Local Track and the update is sent to the entire system.

Communications. The system concept just described requires two different classes of messages: (1) System Messages, which are messages delivered to all nodes in a system, and (2) Directed Messages, which are delivered to one or more specific nodes. As described, the concept requires four distinct System Messages and five distinct Directed Messages. These message types are listed in Table 3.5.1; they are more fully described in the next subsection.

TABLE 3.5.1

MESSAGE TYPES IN TACRAN3

	Type	Description	Text Bit Length
I	Directed Messages		
	01	Send Me a Distance	85
	02	Reply With a Distance	32
	03	Send Me Your Distributed Local Track	24
	04	Reply With a Distributed Local Track	439
	05	Send Measurement to Other Trackers	77
II	System Messages		
	11	Update System Track	206
	12	Drop Me as a Tracker	24
	13	Add Me as a Tracker [and Drop Him]	47
	14	Help	40

3.5.2 Details of TACRAN3

Main Logic Flow (Fig. 3.5.3). The logic for processing a new radar measurement, including the various delays required, is performed in the NEWRET action sequence chain.* The first step is to transform the measurement to system-wide Cartesian coordinates. Next association of the measurement with the Distributed Local Track File (DLTF) is attempted. If successful, logic to be described later is entered. Otherwise, association with the Non-Track File (NTF) is attempted. Again, if successful, logic to be described later is entered. Otherwise, logic to perform track initiation is entered.

*An action sequence chain (A.S.C.), described in Sec. 3.6.5, permits a sequence of actions to be performed on a single data set (a measurement in the present context) with delays permitted during the sequence.

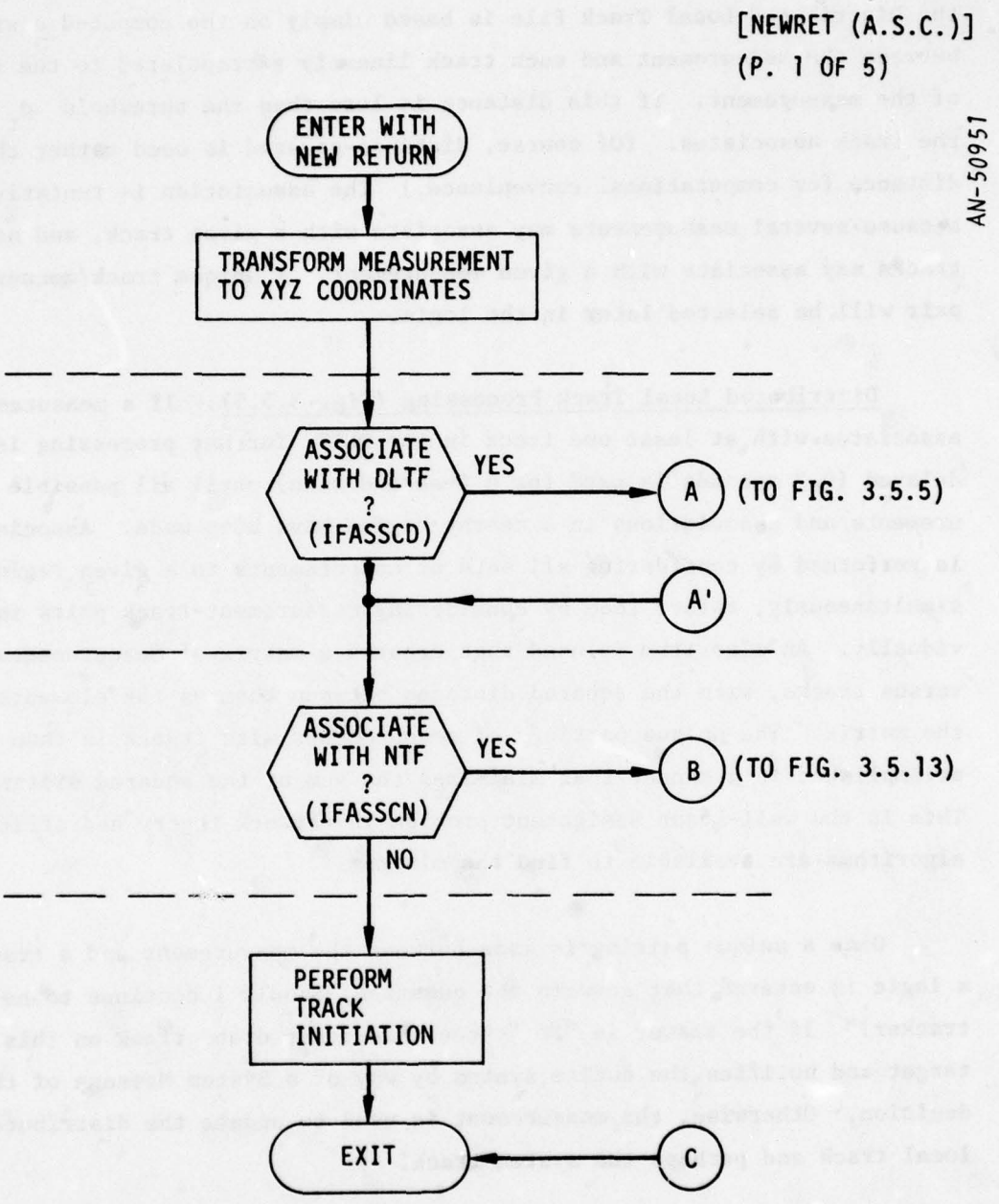


Figure 3.5.3. Main Logic Flow for Processing a New Radar Return

Association with DLTF (Fig. 3.5.4). A tentative association with the Distributed Local Track File is based simply on the computed distance between the measurement and each track linearly extrapolated to the time of the measurement. If this distance is less than the threshold d_t, the track associates. (Of course, distance-squared is used rather than distance for computational convenience.) The association is tentative because several measurements may associate with a given track, and several tracks may associate with a given measurement. A unique track/measurement pair will be selected later in the logic.

Distributed Local Track Processing (Fig. 3.5.5). If a measurement associates with at least one track in the DLTF, further processing is delayed (0.2 seconds is used for a 6-second scan) until all possible measurements and associations in a nearby region have been made. Association is performed by considering all sets of measurements in a given region simultaneously, rather than by considering measurement-track pairs individually. An algorithm is used that creates a matrix of measurements versus tracks, with the squared distance between them as the elements of the matrix. The unique pairings of measurements with tracks is then accomplished in a manner that minimizes the sum of the squared distances. This is the well-known assignment problem in network theory and efficient algorithms are available to find the minimum.

Once a unique pairing is made between the measurement and a track, a logic is entered that answers the question "Should I continue to be a tracker?" If the answer is "No," then this radar drops track on this target and notifies the entire system by way of a System Message of this decision. Otherwise, the measurement is used to update the distributed local track and perhaps the system track.

Under certain unfavorable conditions, this radar may decide that its participation in tracking this target will be difficult in the future. If so, it may send a "help" message to all other nodes. Such a message informs other nodes that this radar would like to be relieved of its tracking responsibilities on this target, and thus speeds up the likelihood of this happening.

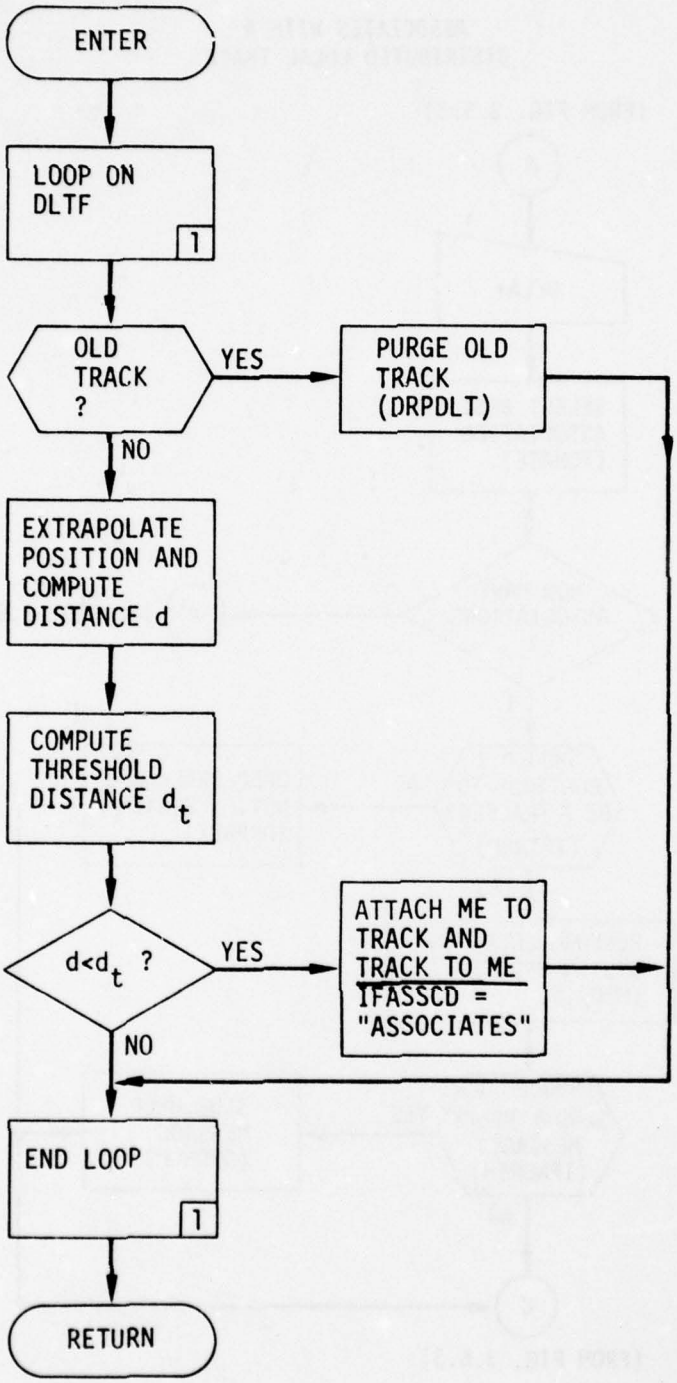


Figure 3.5.4. Association With the Distributed Local Track File

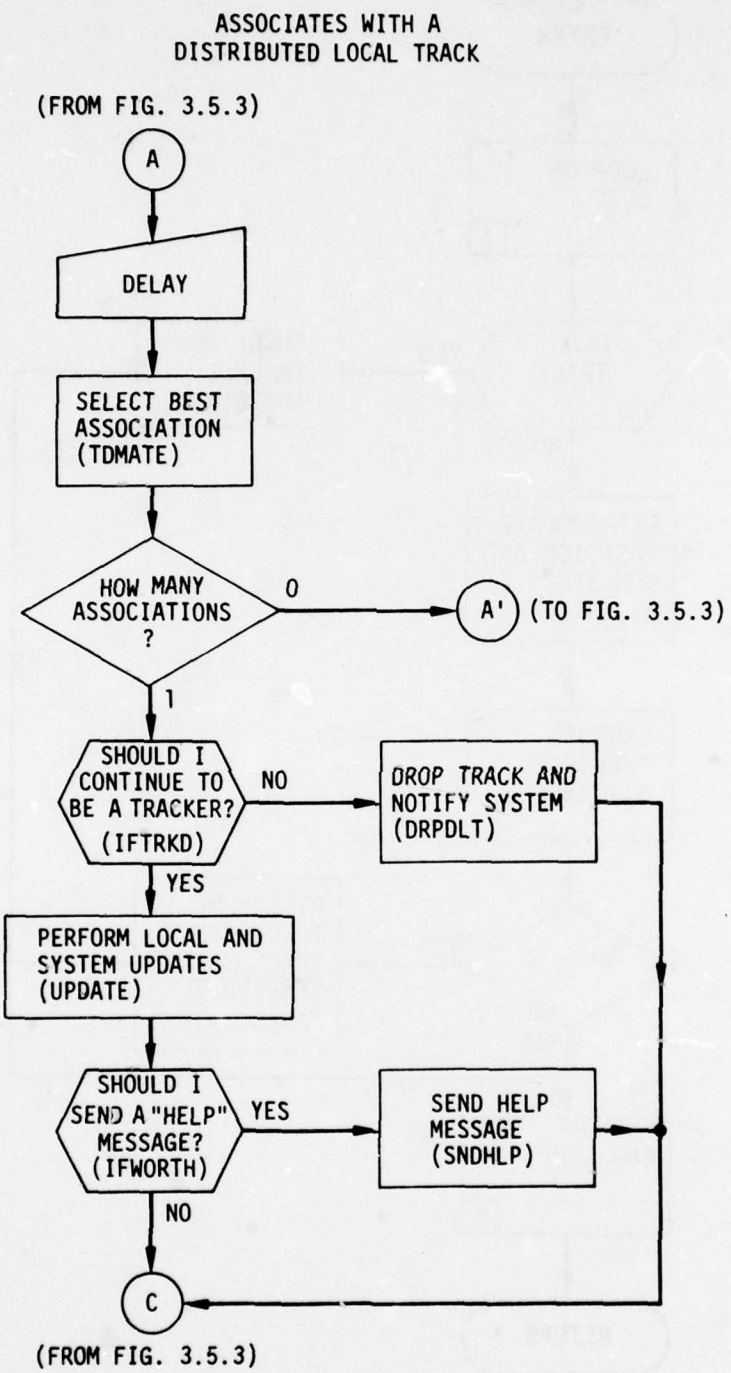


Figure 3.5.5. DLT Track Processing

Tracker Selection Logic I (Fig. 3.5.6). This logic answers the question, "Should I continue to be a tracker?" The criteria for choosing a set of radars to perform tracking are:

- Their measurements are well spaced in time
- Each has a high signal-to-noise (S/N) ratio
- Each has a signal-to-noise ratio that is likely to remain high in the future (based on present S/N, range and range rate)
- The target is not about to exceed the maximum range or enter the blind cone over the radar (caused by maximum elevation)

Other criteria that are not implemented but that might be included are data processing load and terrain blinding.

The most important criterion is that the set of tracking radars provide measurements that are reasonably evenly spaced in time. This calculation is based on the last known measurement time of each tracker, the scan rate, and the known trajectory (i.e., the track) of the aircraft.

The calculation of the next expected reception time is as follows: Let t_i be the last known reception time of tracking radar i. This time is available to all radars in the system and is stored along with the list of trackers for this aircraft. (It is updated at least every time the system track is updated, and in the set of tracking radars it is updated every measurement time.) First, N_s, the integer number of scans plus one since t_i is calculated

$$N_s = [(t - t_i)/T_s] + 1$$

where t is the present time, T_s is the scan period, and $[x]$ is the largest integer in x. Next the time t_{in} of the measurement N_s scans later (assuming the target has not moved) is computed:

[IFTRKD]

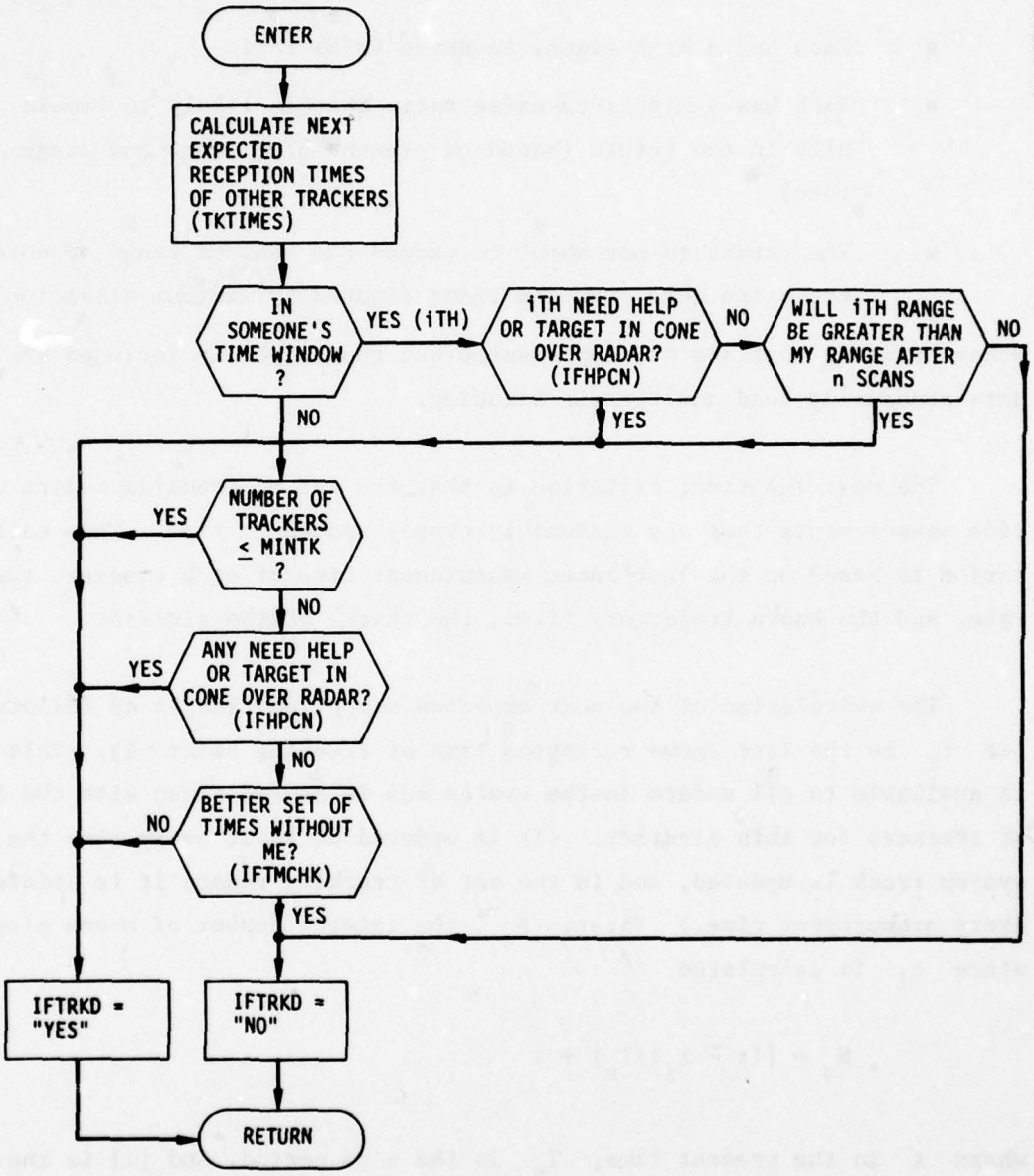


Figure 3.5.6. Should This Node Continue to be a Tracker?

$$t_{in} = t_i + N_s T_s$$

Actually, the target has moved in time t_{in}, and the additional azimuth angle (plus or minus) through which the target actually travels in time t_{in} is

$$\theta = \tan^{-1} \frac{(y_2 - y_R)}{(x_2 - x_R)} - \tan^{-1} \frac{(y_1 - y_R)}{(x_1 - x_R)}$$

where (x_1,y_1) is the position of the track extrapolated back to t_i, (x_2,y_2) is the position of the track extrapolated ahead to t_{in}, and (x_R,y_R) is the position of the radar. The extra time Δt required for the radar to scan through an angle θ is

$$\Delta t = \theta/\omega$$

where ω is the scan rate given by

$$\omega = 2\pi/T_s$$

Of course in Δt seconds the aircraft has moved an additional $\Delta\theta$; this second order effect is ignored.

The next measurement time, t_{inext}, is thus given by

$$t_{inext} = t_{in} + \Delta t$$

It is possible that t_{inext} will be less than the present time t; in this case the calculation is performed over for $N_s = N_s + 1$. It is also possible that t_{inext} might be greater than the present time plus a complete scan period. In this case the calculation could be performed over for $N_s = N_s - 1$, but for no particularly good reason t_{inext} is simply set to $t_s - T_s$.

After the next expected reception times of all trackers are computed, a check is made to see if this radar's reception time is in another radars time "window". The concept of a window is illustrated in Fig. 3.5.7. The purpose of the window, T_W, is to ensure that no two measurements will be closer together than T_W seconds, under the assumption that measurements closer together than this do not materially help the association problem. (The value for T_W used in the simulation is 1.0 seconds.)

If measurements from two radars will be in the same window, then a decision is made as to which should be a tracker. If the other (ith) radar needs help, then this radar continues to track. Otherwise a look n scans into the future is made to determine which radar has the lesser range. If this radar has the lesser range, it will continue to track. Otherwise it will drop itself as a tracker.

If two measurements are not in the same time window, then this radar continues to track as long as the present number of trackers is equal to or less than the desired number of trackers (MINTK). If there are too many trackers at present, then logic is entered to determine if this radar should drop itself. It will not drop itself if any other radar needs help. However, if no radar needs help, then a calculation is made to determine which subset of trackers has the best (most uniformly spaced) set of expected reception times.

Best Set of Reception Times I (Fig. 3.5.8). The best set of reception times is determined by computing a "goodness" measure G_T for each possible subset of reception times, and then choosing the best. The calculation of G_T is illustrated in Fig. 3.5.9. Let N be the desired number of trackers (N ≡ MINTK). Select a subset N of the total number of trackers. Then for this subset

$$G_T = \sum_{i=1}^{N} \left(\Delta T_i - \Delta T_{BEST} \right)^2$$

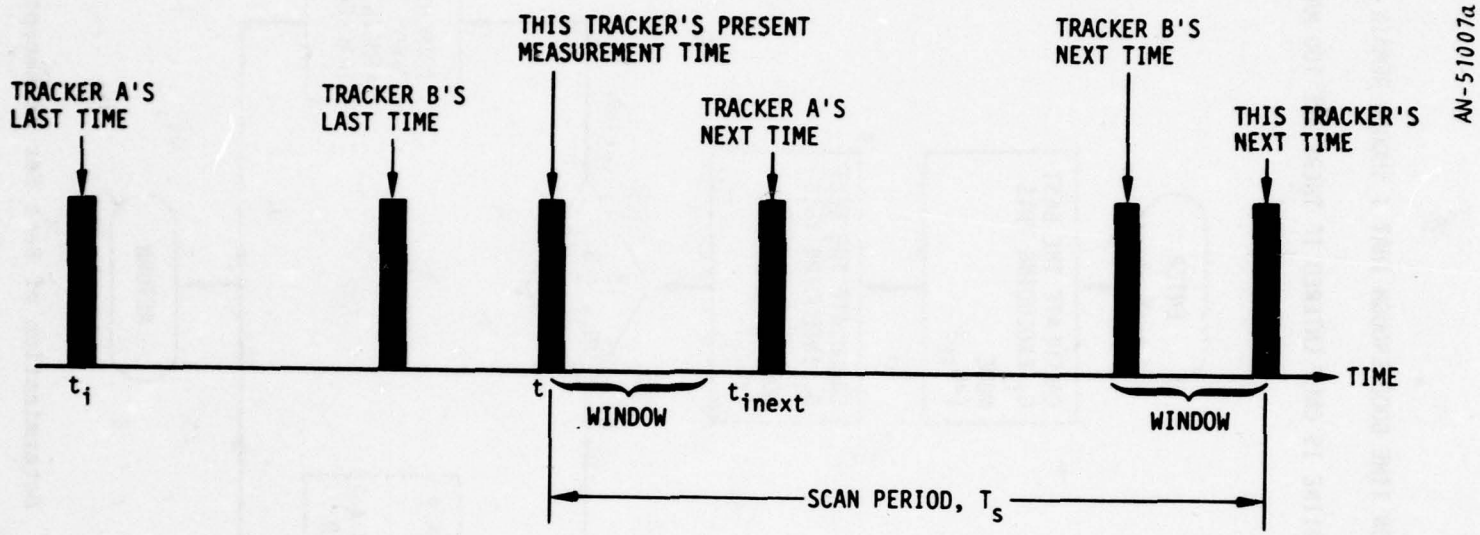


Figure 3.5.7. Next Expected Reception Times and Time Window

[IFTMCHK]

AN-50946

IS MY RECEPTION TIME GOOD ENOUGH THAT I SHOULD REMAIN A TRACKER?

(NOTE: THIS ROUTINE IS ONLY ENTERED IF THERE ARE TOO MANY TRACKERS)

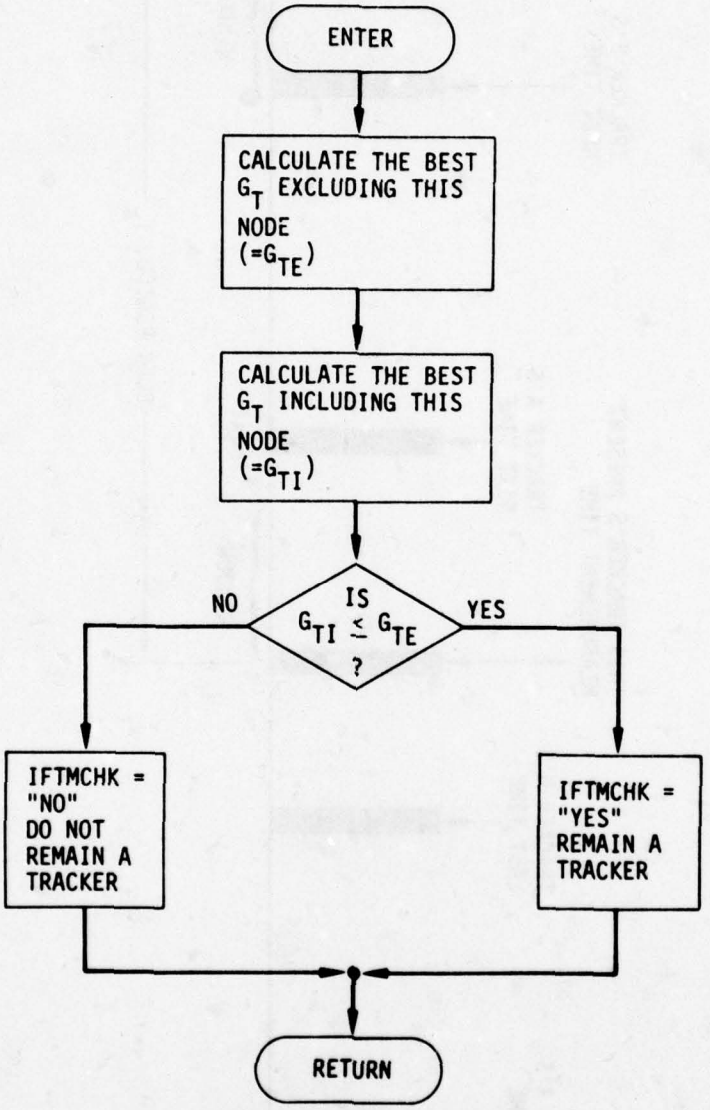
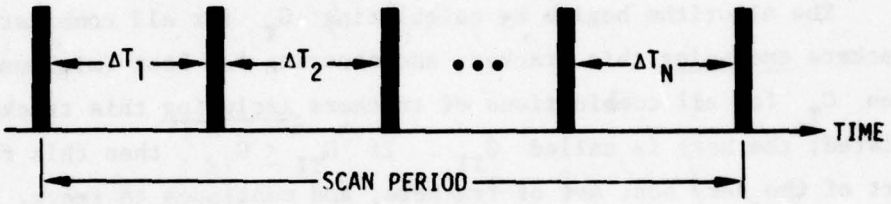


Figure 3.5.8. Determination of Best Set of Reception Times (I)

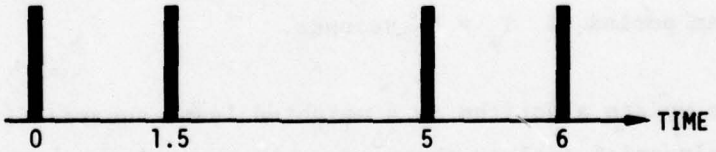
- N = DESIRED NUMBER OF TRACKERS



EXAMPLE

SCAN PERIOD = 6 SECONDS

N = 3



$\Delta T_{BEST} = 2$

$G_T = (1.5 - 2)^2 + (3.5 - 2)^2 + (1 - 2)^2$

$= (0.25) + (2.25) + (1)$

$= 3.5$

Figure 3.5.9. A "Goodness" Measure for a Set of Tracker Reception Times

where T_{BEST} is the ideal spacing of N measurements,

$$\Delta T_{BEST} = T_s/N$$

and T_s is the scan period. Thus G_T is the sum square deviation of the measurements from the ideal spacing; for equally spaced measurements, $G_T = 0$.

The algorithm begins by calculating G_T for all combinations of trackers excluding this tracker, and choosing the best (minimum) G_{TE}. Then G_T for all combinations of trackers including this tracker is calculated; the best is called G_{TI}. If $G_{TI} \leq G_{TE}$, then this radar is a part of the very best set of trackers, and continues to track. Otherwise it drops itself as a tracker.

Update Track (Fig. 3.5.10). If this radar decides to continue to be a tracker, the track update logic is entered. The first step is to maintain the list of points (measurements) to be used in the update. The present measurement is added to the list, and old points, if any, are dropped. The list is maintained so that there are (1) at least two points, (2) no more than N_p points, and (3) no points older than T_p seconds. N_p and T_p are input parameters. Values typically used are $N_p = 5$, $T_p = 14$ s for a scan period of $T_s = 6$ seconds.

The update algorithm is a weighted least-squares fit to a second degree polynomial (unless there are only two points). The track equations are

$$x(t) = x_0 + x_1 t + x_2 t^2$$

$$y(t) = y_0 + y_1 t + y_2 t^2$$

$$z(t) = z_0 + z_1 t + z_2 t^2$$

AN-50948

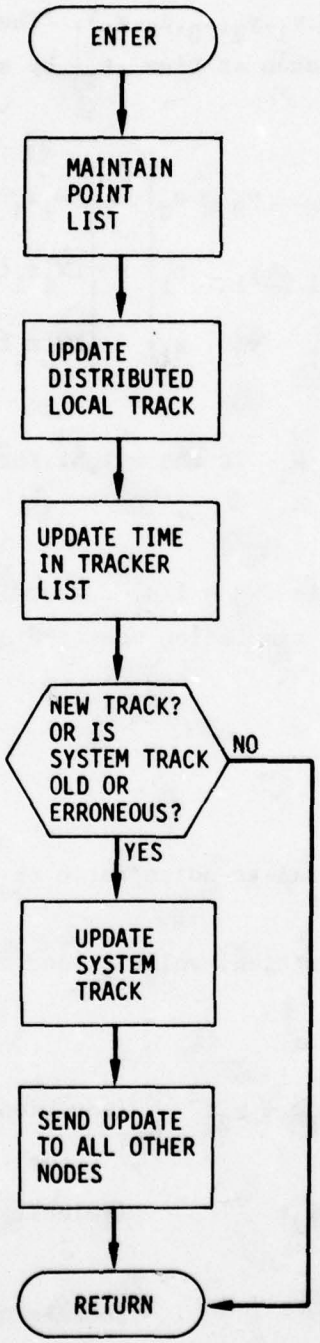


Figure 3.5.10. Distributed Local and System Track Update

Thus the track at any given time is completely determined by the nine coefficients $(x_0,x_1,x_2,y_0,y_1,y_2,z_0,z_1,z_2)$. These are determined from N measurements (x_i,y_i,z_i) taken at time t_i by solving the matrix equation

$$\begin{bmatrix} \Sigma W_i & \Sigma W_i t_i & \Sigma W_i t_i^2 \\ \Sigma W_i t_i & \Sigma W_i t_i^2 & \Sigma W_i t_i^3 \\ \Sigma W_i t_i^2 & \Sigma W_i t_i^3 & \Sigma W_i t_i^4 \end{bmatrix} \begin{bmatrix} x_0 & y_0 & z_0 \\ x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \end{bmatrix} = \begin{bmatrix} \Sigma W_i x_i & \Sigma W_i y_i & \Sigma W_i t_i \\ \Sigma W_i x_i t_i & \Sigma W_i y_i t_i & \Sigma W_i z_i t_i \\ \Sigma W_i x_i t_i^2 & \Sigma W_i y_i t_i^2 & \Sigma W_i z_i t_i^2 \end{bmatrix}$$

for the nine parameters. W_i is the weight for the ith measurement. All sums are from $i = 1$ to N.

The optimum weight is $W_i = 1/\sigma_i^2$, where σ_i^2 is the variance of the ith measurement. The simulation uses the approximation $(S/N)_i \propto 1/\sigma_i$; thus

$$W_i = (S/N)_i$$

where $(S/N)_i$ is the signal-to-noise ratio of the ith measurement.

The extrapolated position, velocity and acceleration in the x-coordinate are given by

$$x(t) = x_0 + x_1 t + x_2 t^2 \qquad \text{(position)}$$

$$\dot{x}(t) = x_1 + 2x_2 t \qquad \text{(velocity)}$$

$$\ddot{x}(t) = 2x_2 \qquad \text{(acceleration)}$$

with similar equations in the y- and z-coordinates.

The system track is updated if (1) this is a new distributed local track, (2) if the time since the last update of the system track exceeds a maximum time (on the order of a minute), or (3) if the extrapolated system track position deviates from the distributed local track by some predetermined system constant (typically one kilometer has been used). The system track is updated simply by replacing the old set of nine coefficients with the newest set just computed for the distributed local track.

"Help" Messages (Fig. 3.5.11). After updating a distributed local track, the updating radar asks the question "Should I send a help message?" The answer is "Yes" if the aircraft being tracked is extrapolated to be beyond the maximum radar range or above the radar's maximum elevation at the next expected measurement time. A message is also sent if the target is beyond a threshold range and if the signal-to-noise ratio (S/N) is less than a threshold S/N and the target is receding from the radar at a rapid rate.

Association With NTF (Fig. 3.5.12). If the measurement does not associate with the DLTF, an attempt is made to associate it with the Non-Track File (Fig. 3.5.3). The logic for accomplishing this is similar to that for the DLTF (Fig. 3.5.4).

Non-Track Association Processing (Fig. 3.5.13). If association with the NTF is successful, further processing is delayed until all possible measurement/track pairs that might affect the association of this measurement are known. If after the delay there are more than one unique measurement/track pair, the calculated distance between the measurement(s) and the system tracks are too inaccurate to select the correct pairings. Therefore a logic (described later) is entered in which a message is sent to a known tracker of each associated target to obtain the more accurate distance between the measurement and the distributed local track.

After all the more accurate distances are received, the best association with the measurement is decided by the assignment algorithm, which either selects a unique track or decides no track associates.

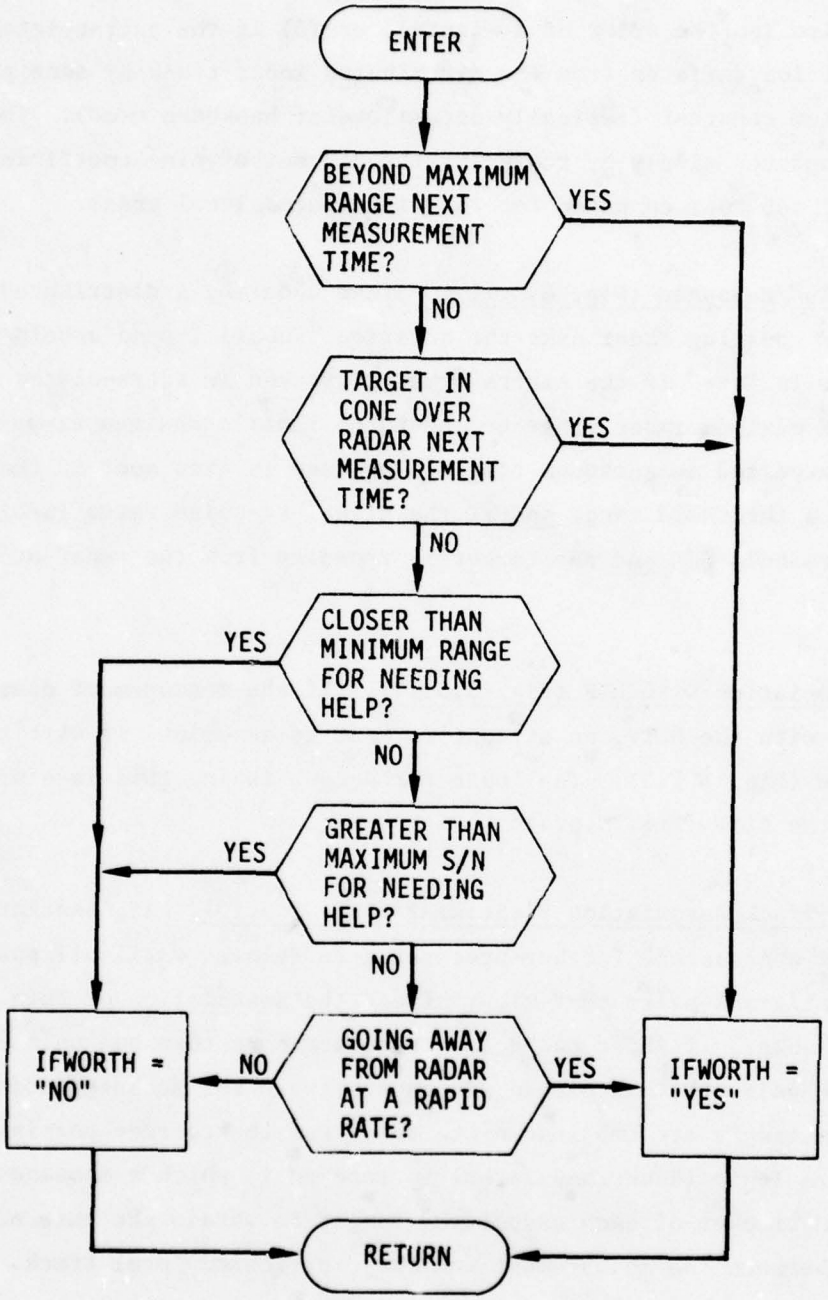


Figure 3.5.11. "Help" Message Logic

[IFASSCN]

AN-50966

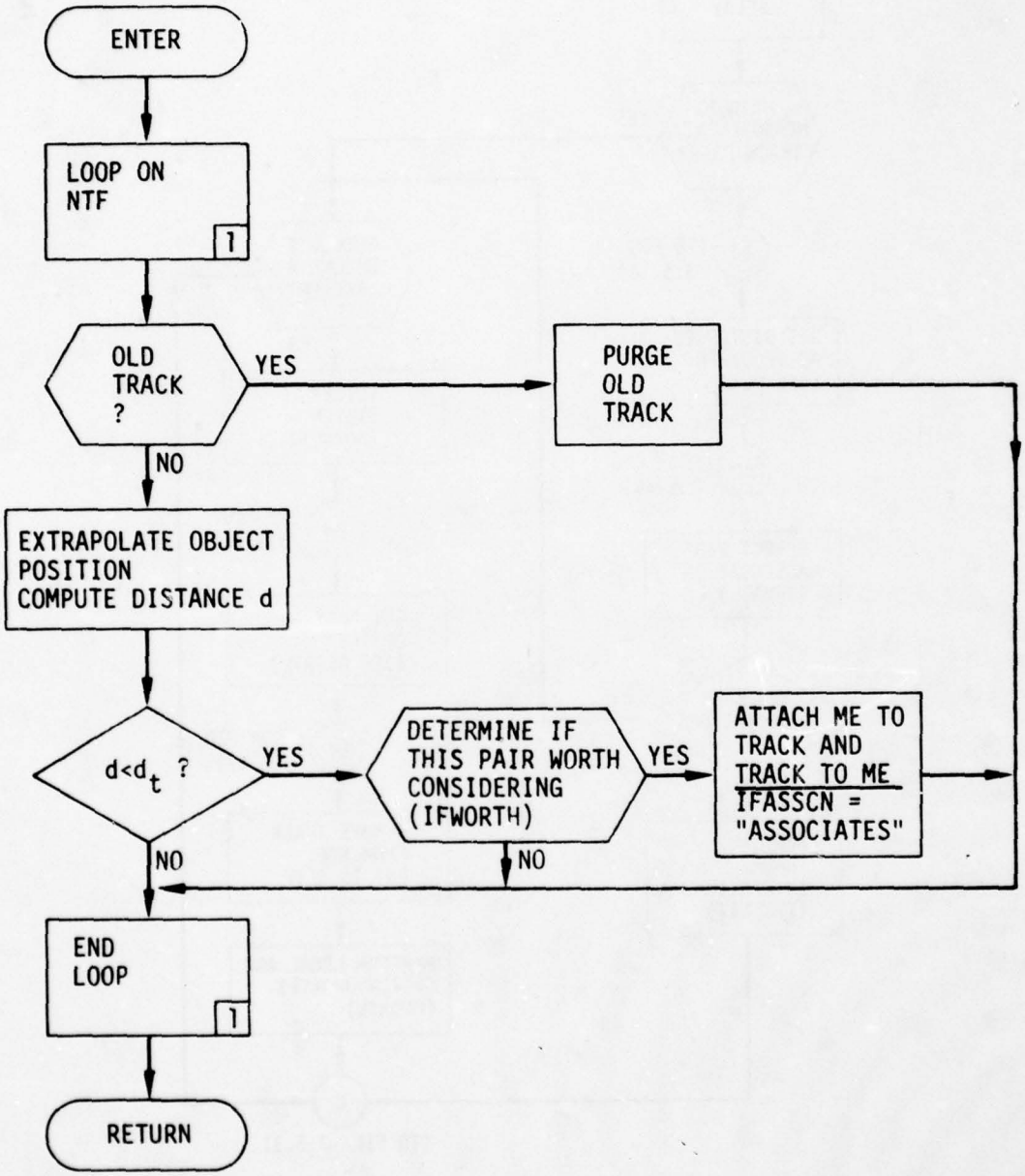
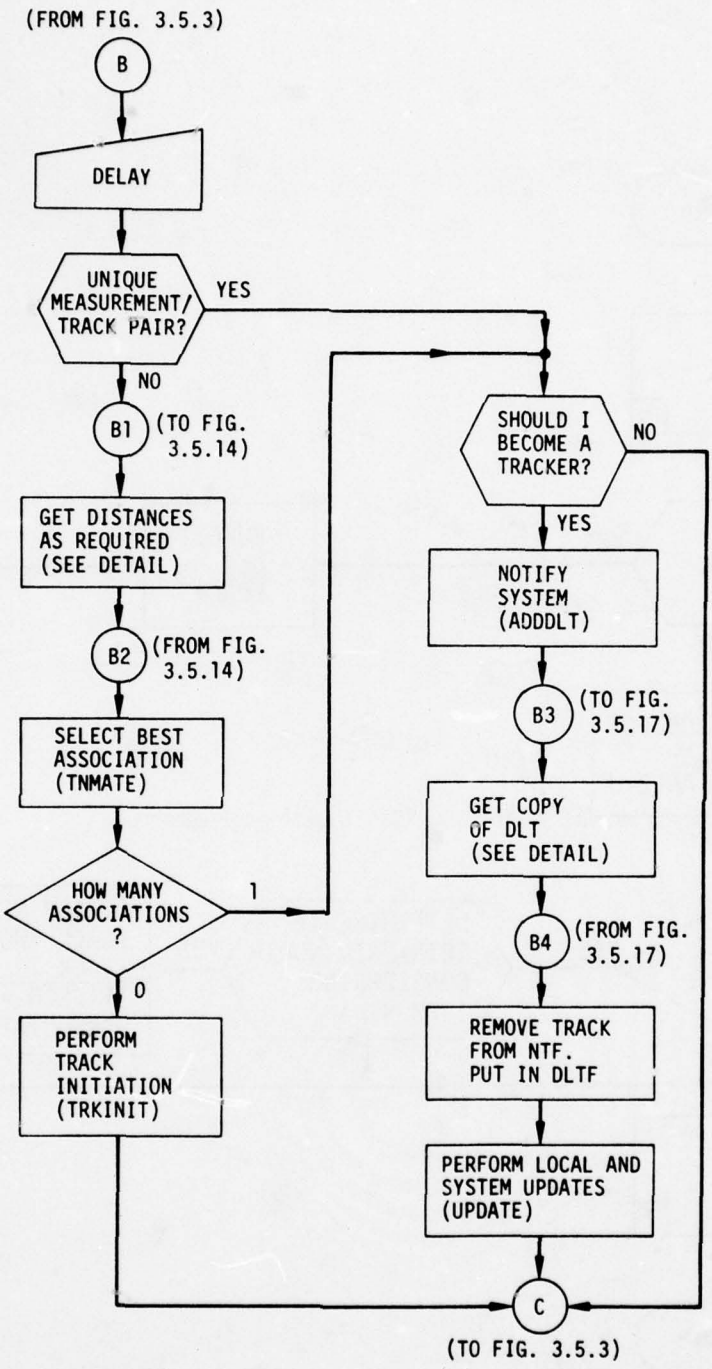


Figure 3.5.12. Association With the Non-Track File



AN-50953

Figure 3.5.13. Association With a System Track

If a unique measurement/track pair is associated, the question is asked "Should I become a tracker?" If the answer is "Yes," all nodes are notified of this fact by way of a system message, and this node sends a directed message to one of the trackers to get a copy of the distributed local track on this target. The track is moved from the NTF to the DLTF, and local and (perhaps) system track updates are performed.

Get Distances As Required (Fig. 3.5.14). If multiple associations of a measurement and system tracks in the NTF have been made, a logic is entered which gets accurate distances between the measurements and tracks using the DLTF. This is accomplished for each measurement/track pair by sending a message containing the measurement and the associated track label to a tracker of that track and requesting that an accurate distance be computed and sent back. First a list of measurement/track pairs is made. This list is looped through and for each pair an "send me a distance" message is sent to the first tracker on the tracker list that is attached to the system track. (A more communications-efficient logic will combine distance-requests to the same tracker and find the set of trackers that minimizes the total number of messages.)

Next there is a delay while waiting for the return messages. The "WAIT" is a special form of delay with two ways to resume. If a reply event occurs before the maximum delay occurs, then processing is immediately resumed at "RESUME ON REPLY". Otherwise the logic "RESUME ON TIME LIMIT" is entered. Resume on reply requires that all of the replies are back. If the resume-on-time-limit logic is entered, it means this node failed to get a reply on time for at least one measurement/track pair. Those pairs that did get a reply are deleted from the duplicate-track list that was created at the top of the flow chart, and for those pairs that did not get a reply, a new distance request is made, this time using the second tracker in the tracker list. Again there is a want for replies. Eventually (hopefully) all distances are received and processing can continue.

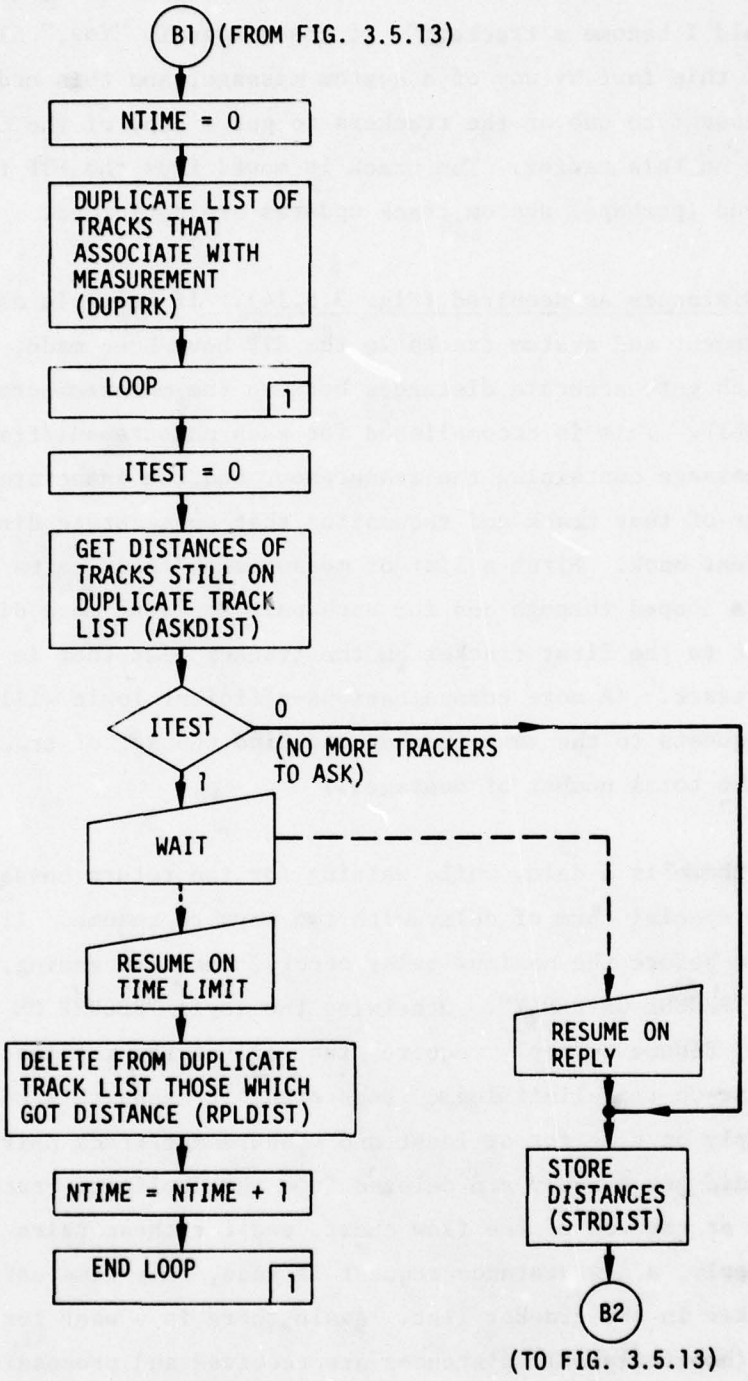


Figure 3.5.14. Detail: Get Distances as Required

Tracker Selection Logic II (Fig. 3.5.15). After a unique measurement/track pair have been associated, this node (which is not presently a tracker since association was made with the Non-Track File), asks the question, "Should I become a tracker?" This logic is similar to the logic a tracking radar enters when determining if it should continue to be a tracker (Fig. 3.5.6). The thresholds on these two logics must be carefully chosen to prevent "thrashing" among the trackers. That is, a change in the set of tracking radars must not occur too easily to prevent it from occurring too often.

Best Set of Reception Times II (Fig. 3.5.16). Part of the logic of determining if this node should become a tracker is determining if its replacing a present tracker significantly improves the spacing of the pooled measurements. The computations in this logic are identical to those performed in Fig. 3.5.8; however, the threshold logic is different. First the goodness measure G_T (described on p. 306) is calculated for all combinations of N of the present trackers, where N is the desired number of trackers. (Usually the present number of trackers will be N, but it is possible to have more than N temporarily.)

If the best (lowest) of these G_T ($= G_{TE}$) is less than an input threshold, the present set of trackers is spaced well enough so that there is no further consideration. Otherwise G_T is computed for all combinations of N possible trackers including this node. (For example, if there are presently 3 trackers and $N = 3$, then there are $\binom{4}{3} = 4$ possible combinations, given by the binomial coefficient.) The ratio of the lowest G_T ($= G_{TI}$) including this node to the lowest G_T excluding this node, G_{TI}/G_{TE}, is compared with an input threshold (which is less than unity). If the ratio is sufficiently small (G_{TI} is sufficiently lower than G_{TE}), this tracker replaces (one of) the present tracker(s) that was excluded from the computation of G_{TI}.

Get Copy of DLT (Fig. 3.5.17). If this node decides to become a tracker, it must obtain a copy of the Distributed Local Track. It does

[IFTRKN]

AN-50956

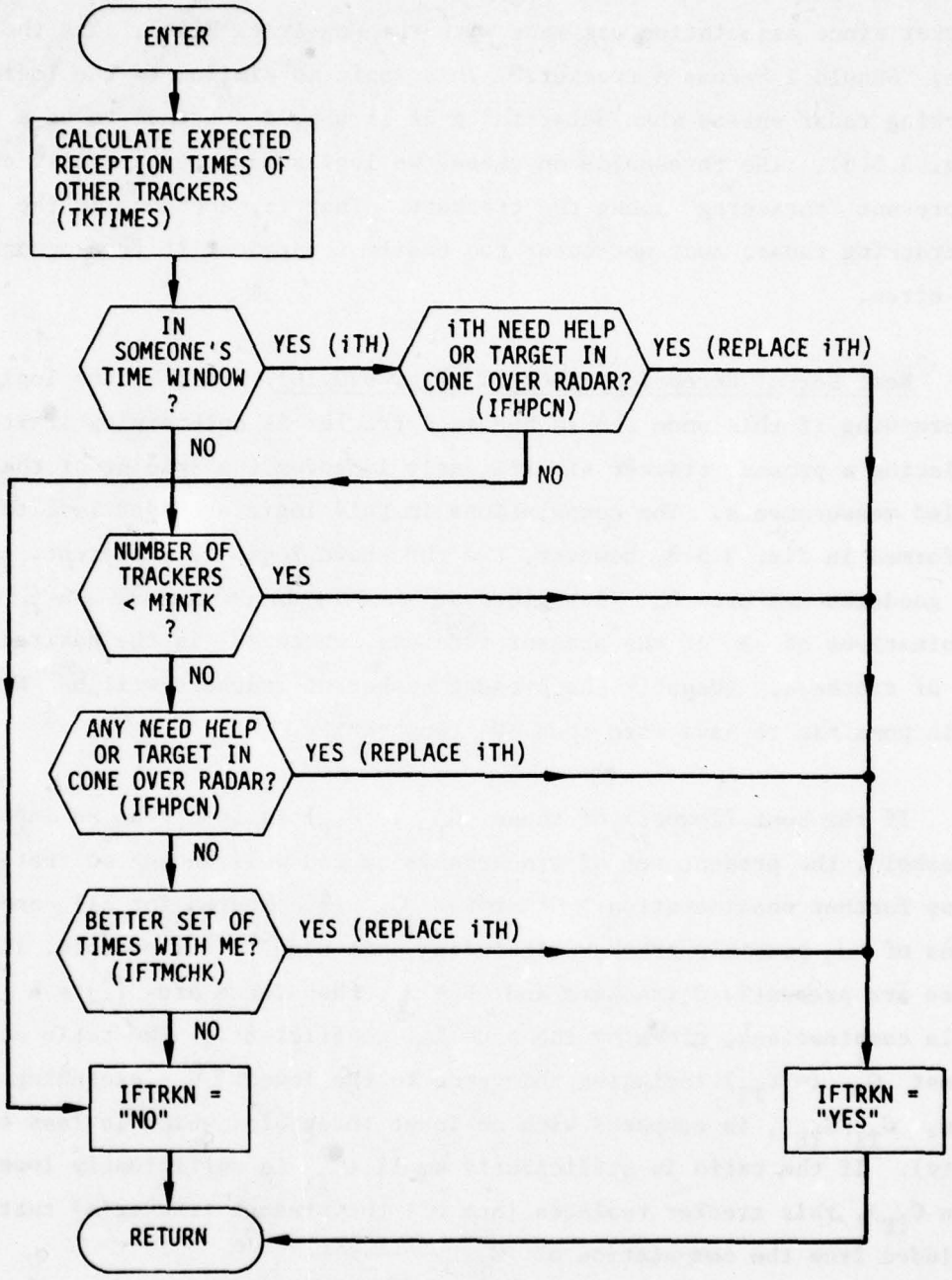


Figure 3.5.15. Should This Node Become a Tracker?

[IFTMCHK]

IS MY RECEPTION TIME GOOD ENOUGH THAT I SHOULD REPLACE A TRACKER?

(NOTE: THIS ROUTINE IS ONLY ENTERED IF THERE ARE AT LEAST ENOUGH TRACKERS)

AN-50958

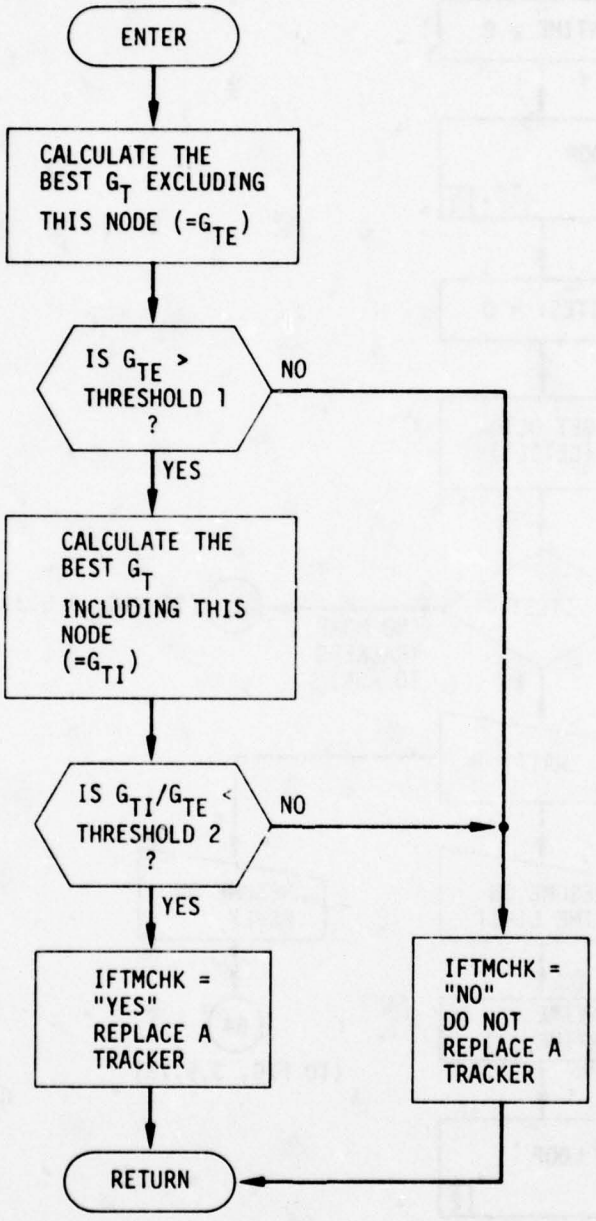


Figure 3.5.16. Determination of Best Set of Reception Times (II)

[NEWRET]
(P. 5 OF 5)

AN-50961

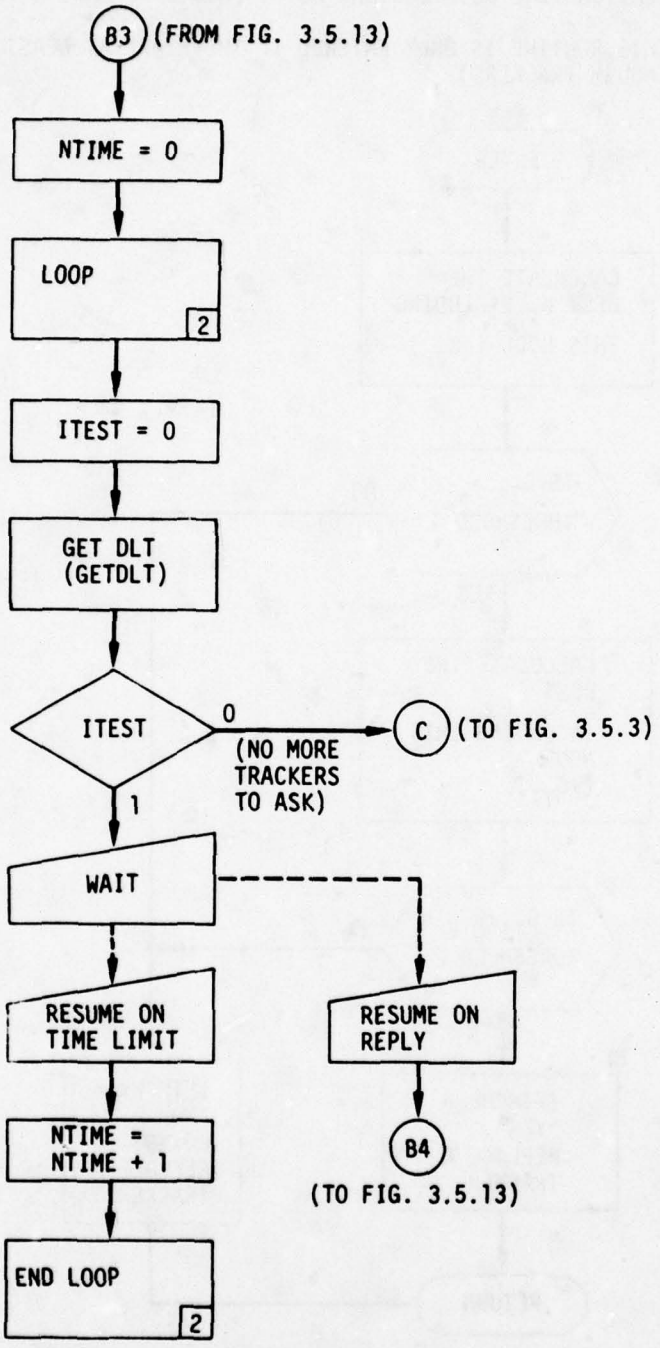


Figure 3.5.17. Detail: Get a Copy of the Distributed Local Track

this by sending a message to one of the present trackers that says "Send me a copy of your DLT on track label n." This is done in a similar manner to that described in Fig. 3.5.14.

Track Initiation. If all attempts at association fail, the track initiation logic is entered (see Fig. 3.5.3). The track initiation sequence is illustrated in Fig. 3.5.18. The first measurement on a new target is saved. The second measurement one scan later is paired with the first measurement (and perhaps others erroneously) on the basis of speed; that is, if the distance between the measurements d divided by the time between the measurements T (which should be close to the scan time T_s) is less than the maximum credible speed, V_{max} :

$$d/T < V_{max}$$

Then a linear track is started in which the position (x,y,z) is the same as the last measurement and the velocity is determined by (x/T,y/T,z/T). The result is put into the Track Initiation File (TIF).

The third measurement is associated with the extrapolated track in the TIF. If the association is successful, a new Distributed Local Track is initiated.

The logic for track initiation is shown in Figs. 3.5.19 and 3.5.20.

3.5.3 TACRAN3 Results

Initial Test Run. The geometry for the first test run of TACRAN3 is shown in Fig. 3.5.21. Four nodes were selected to represent all the node types: Radars 1, 2, and 3 can see the three aircraft targets, Radar 4 is representative of a node that cannot see the aircraft (the maximum 80 km ranges are indicated on the figure). Radars 1 and 3 see targets during only part of their flight paths; they operate at long range with low signal-to-noise ratio. The communications paths (shown in heavy

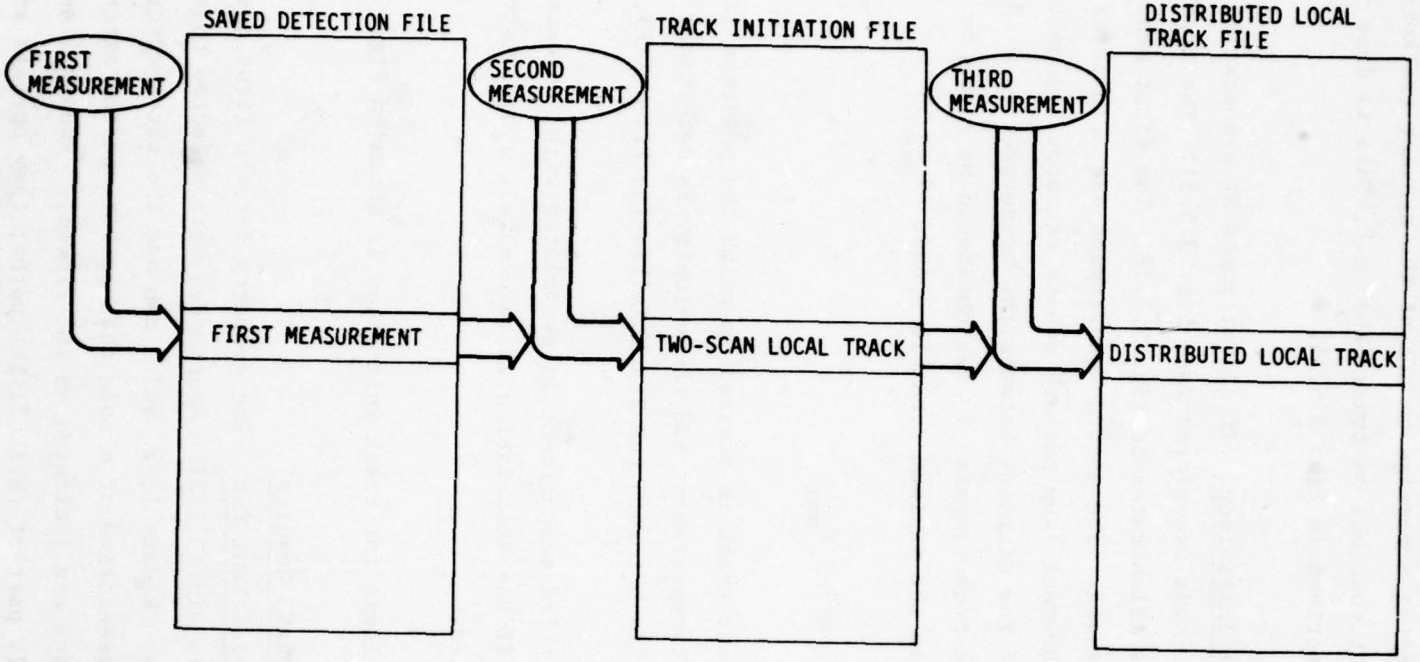


Figure 3.5.18. Track Initiation Sequence

AN-50964

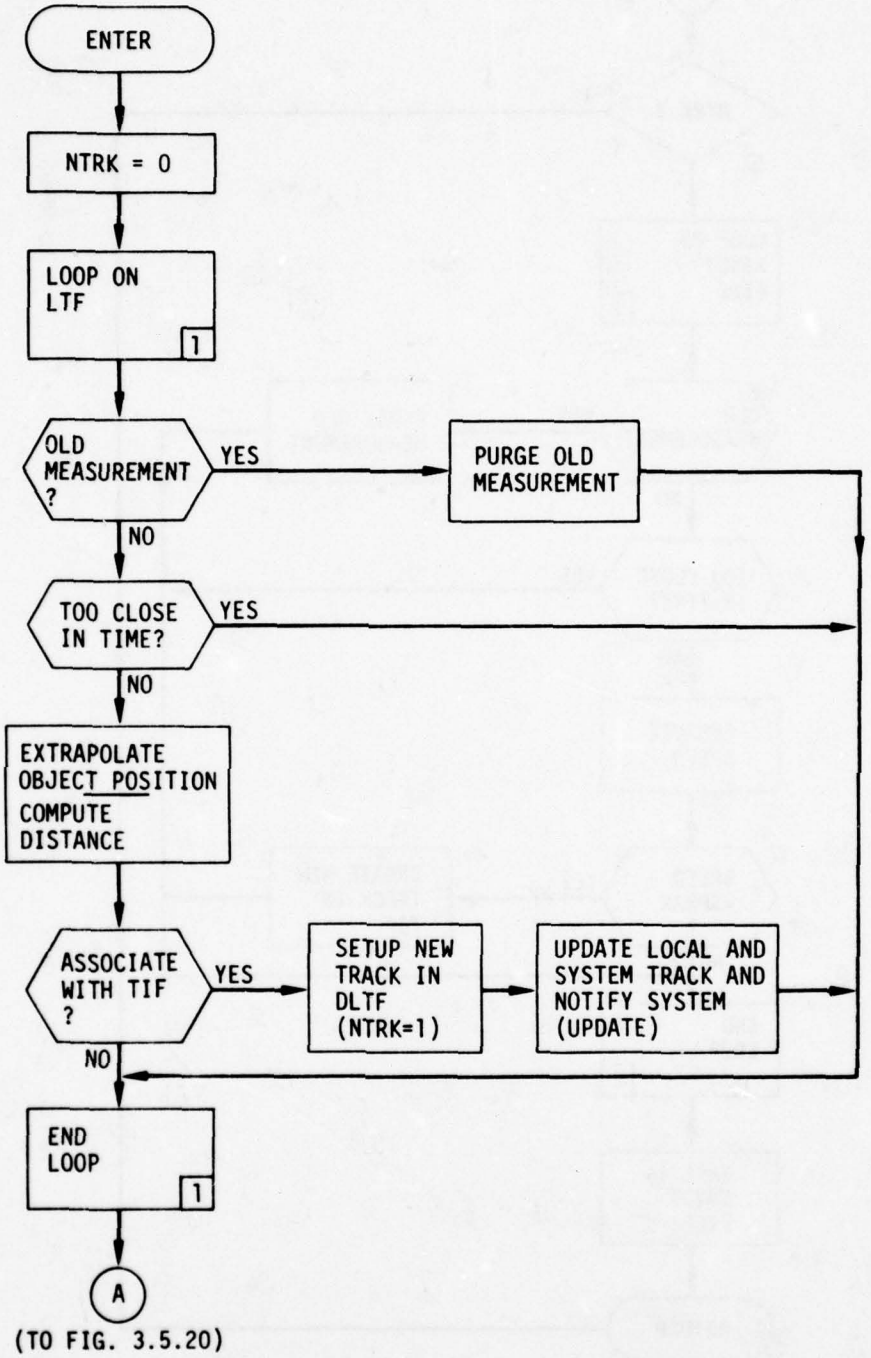


Figure 3.5.19. Track Initiation (Three-Scan)

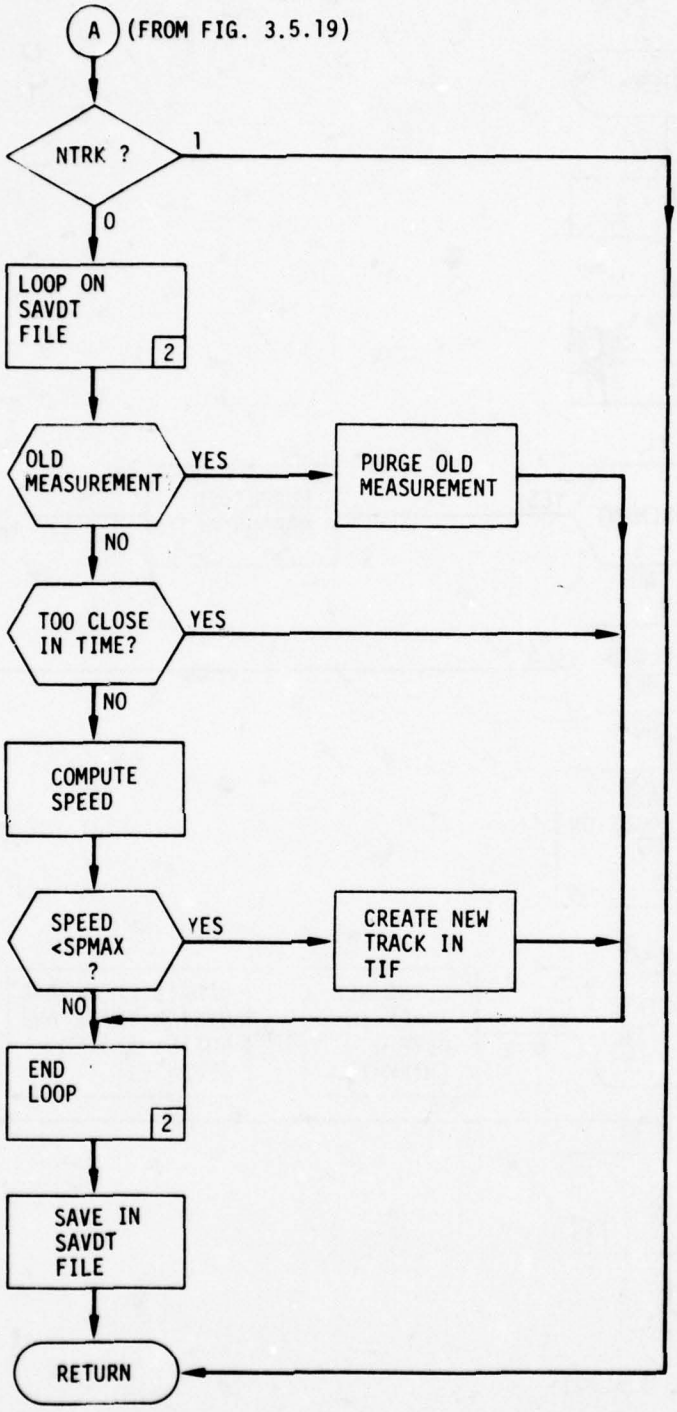


Figure 3.5.20. Track Initiation (Two-Scan)

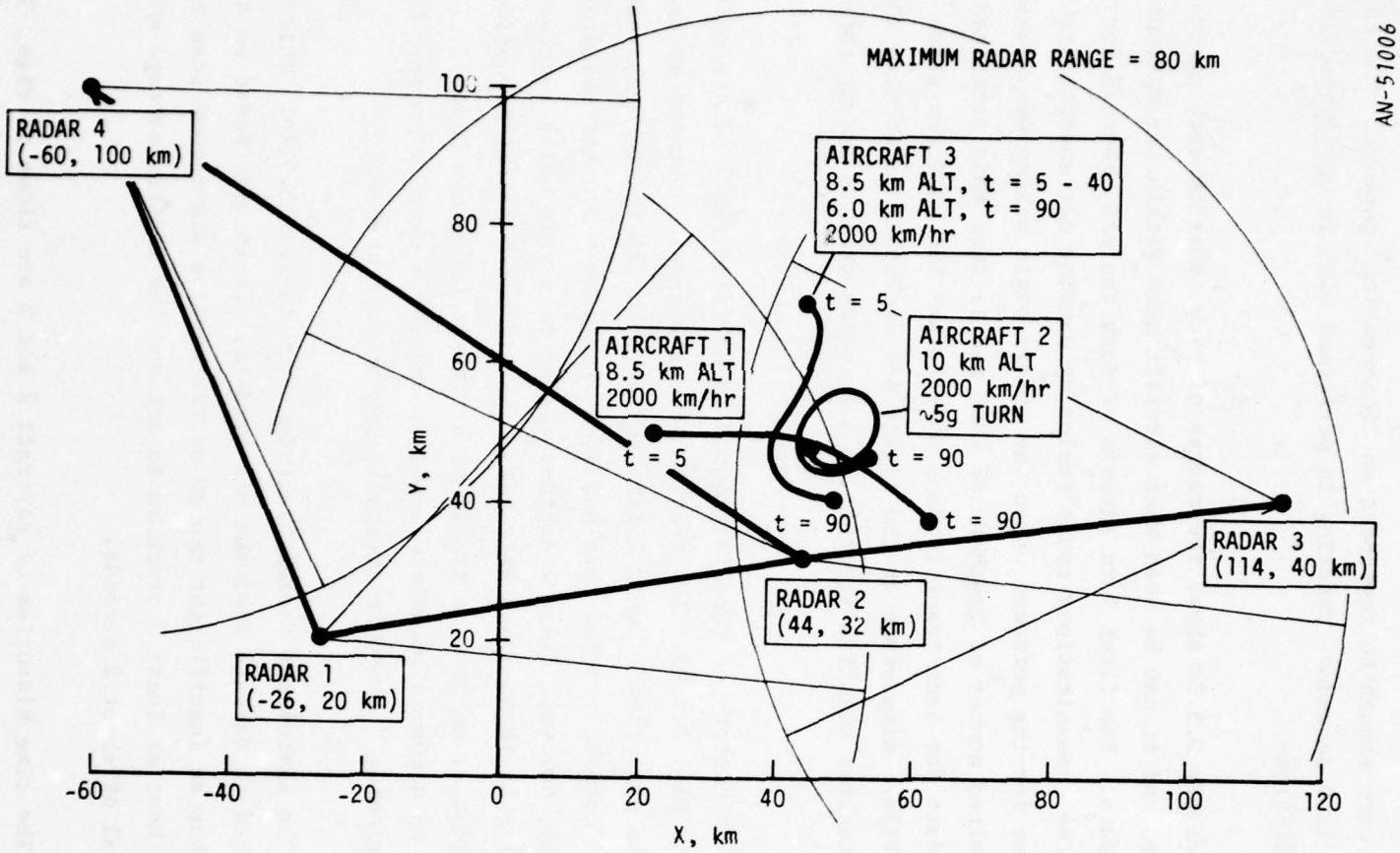


Figure 3.5.21. Geometry for TACRAN3 Test Run 1

lines) were chosen to represent an "interesting" network. The radar scan period is 6 seconds. Tracking is performed with an unweighted least squares filter.

Figure 3.5.22 shows the ranges of each radar to each aircraft during the run, and it can be seen when aircraft were within the maximum range of the radars. The first five seconds of each run are reserved for initializing the communications route tables by sending out unaddressed system messages for the purpose. Then the radars begin making measurements. The desired number of trackers in this run is two; this permitted one of the radars for each track to be a non-tracker and have the track in its Non-Track File. Also shown in the figure are dashed lines showing which radars were trackers of which aircraft as a function of time during the simulation run.

The history of the Distributed Local Track (DLT) on Aircraft 1 is shown in Fig. 3.5.23. The tracker selection logic operates as expected with one exception: Radar 1 asks for help on its fourth measurement. Radar 3 correctly notes this and adds itself as a tracker and drops Radar 1. Next, however, Radar 1 decides (based on a very noisy range-rate calculation) that it does not need help, that it has a better reception time than Radar 2, so it adds itself as a tracker and drops Radar 2. This points up a common threshold problem, and shows that to prevent this kind of "thrashing," a double-threshold system must be used.

The associated System Track is shown in Fig. 3.5.24. This track was permitted to deviate from the DLT by 3 km. It is also based on a DLT which has an insufficient number of trackers; a six-second scan period should have at least 3 trackers to achieve the desired average measurement interval of about 2 seconds.

The time histories of Aircraft 2 and 3 are shown in Figs. 3.5.25 through 3.5.28.

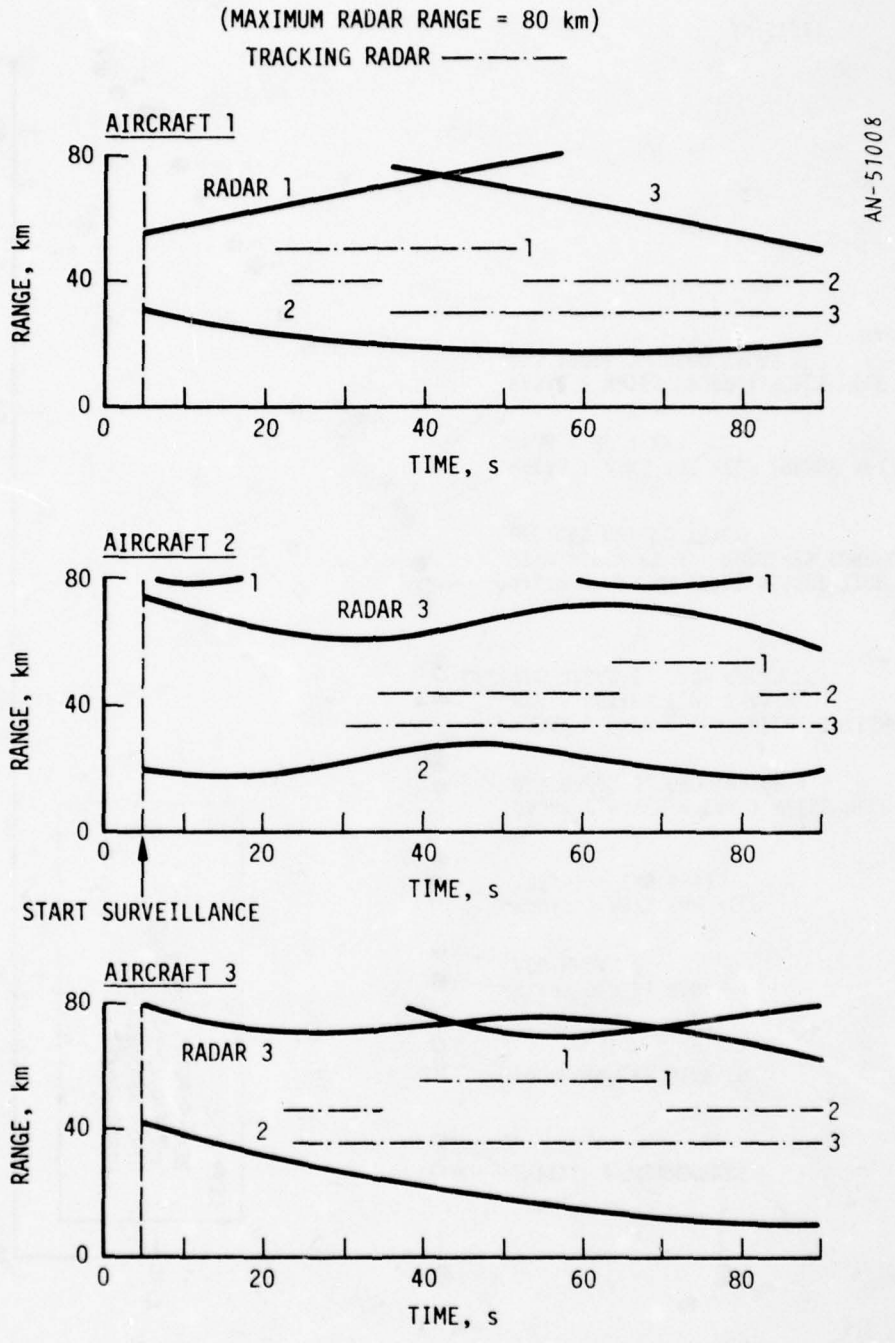


Figure 3.5.22. Aircraft Ranges Versus Time, TACRAN3, Test Run 1

AN-5101

Figure 3.5.23. Distributed Local Track History, Aircraft 1

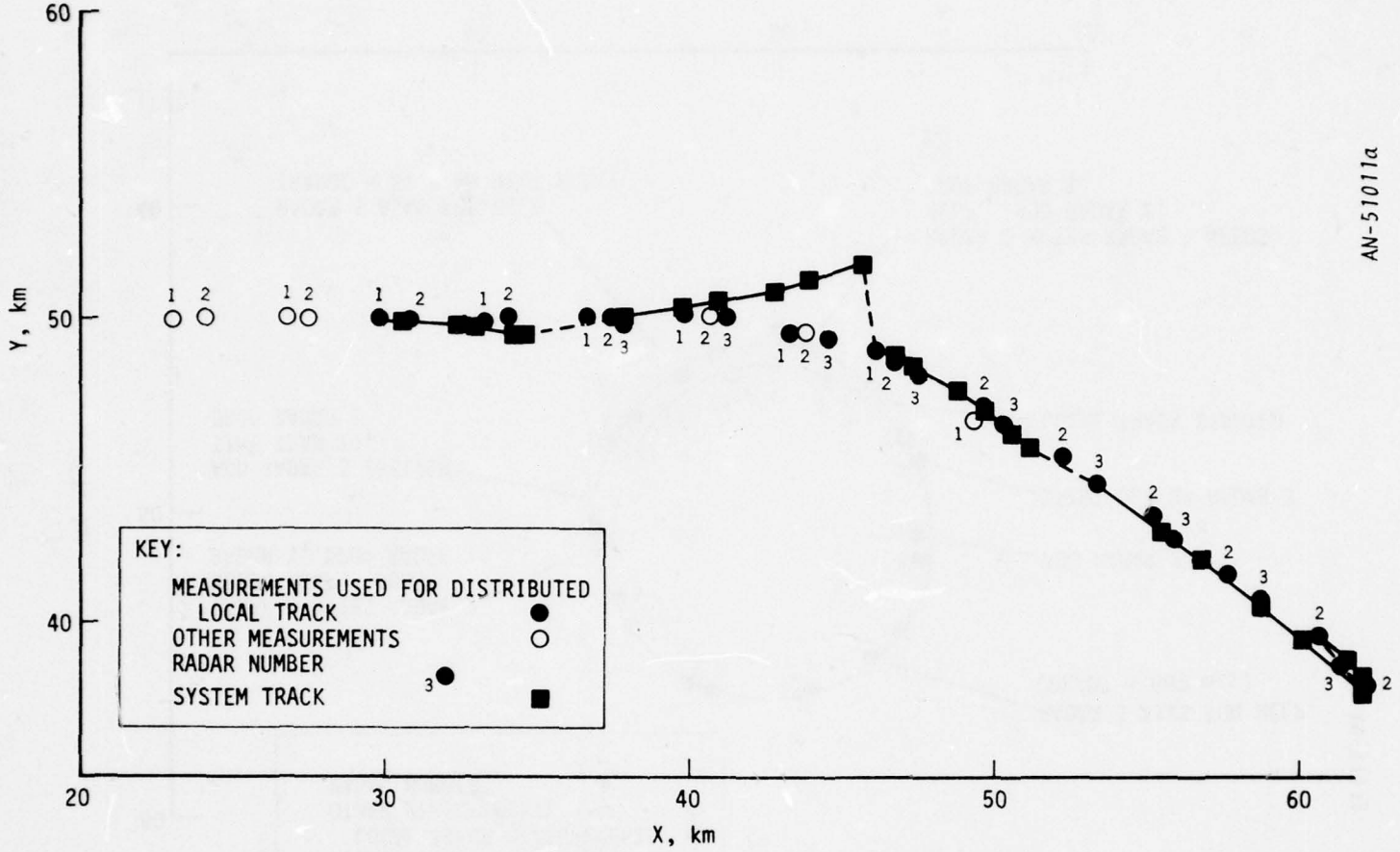


Figure 3.5.24. System Track History, Aircraft 1

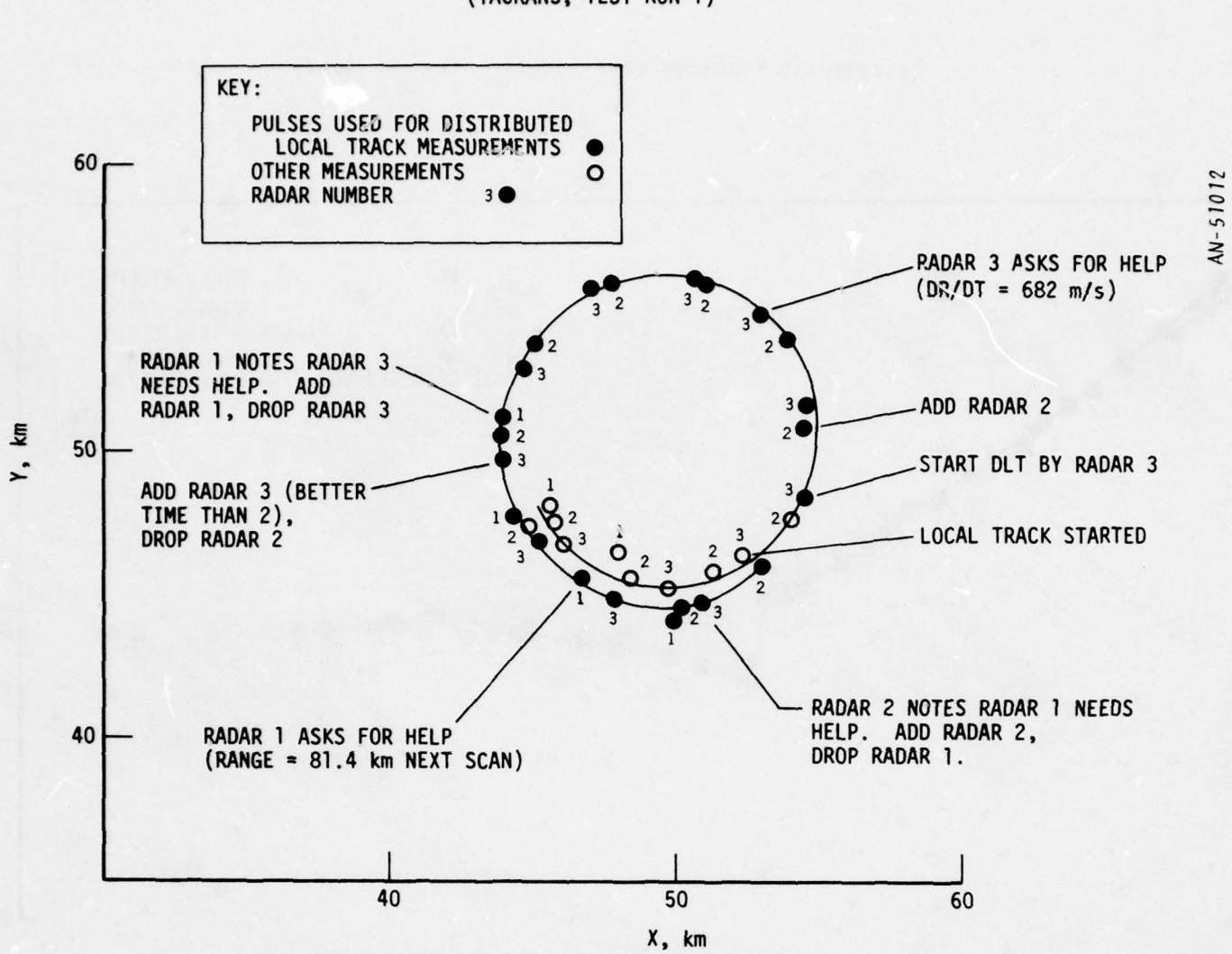


Figure 3.5.25. Distributed Local Track History, Aircraft 2

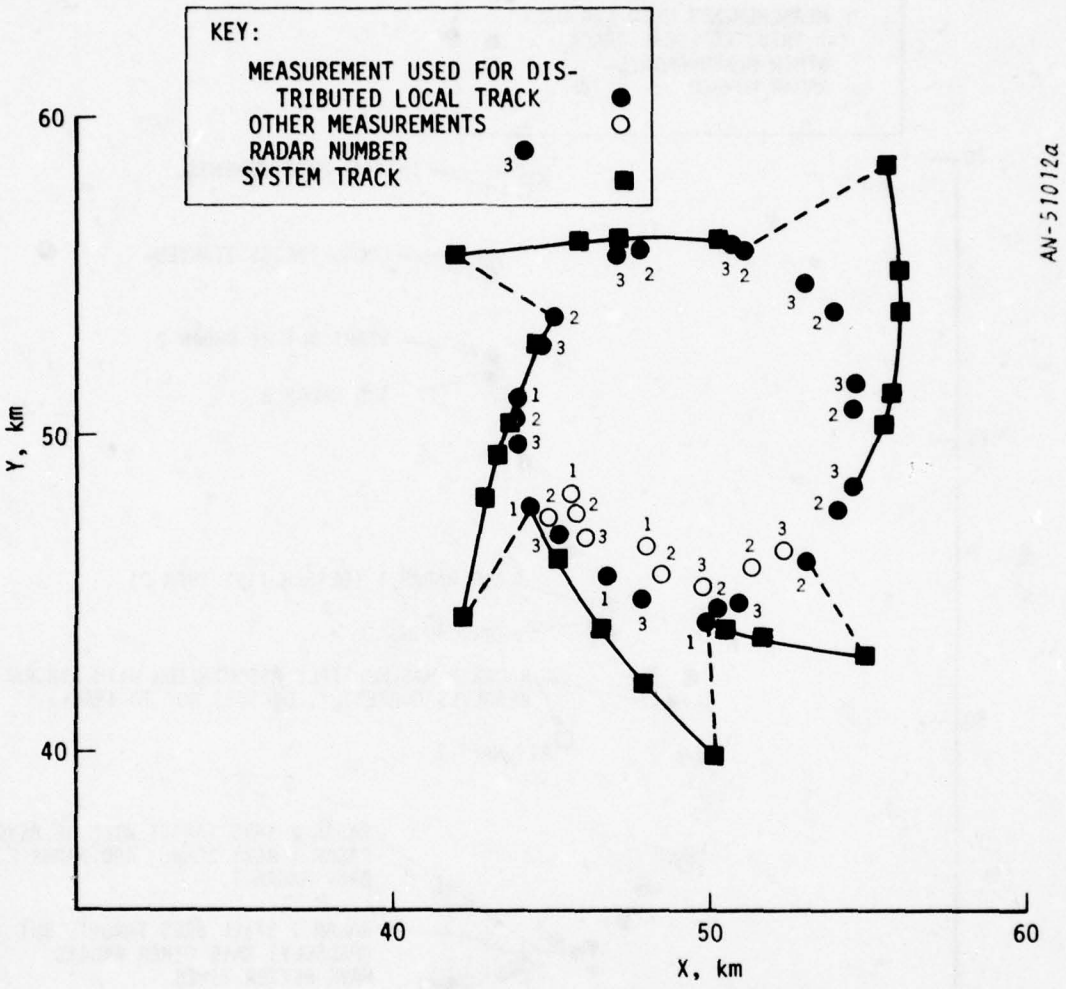


Figure 3.5.26. System Track History, Aircraft 2

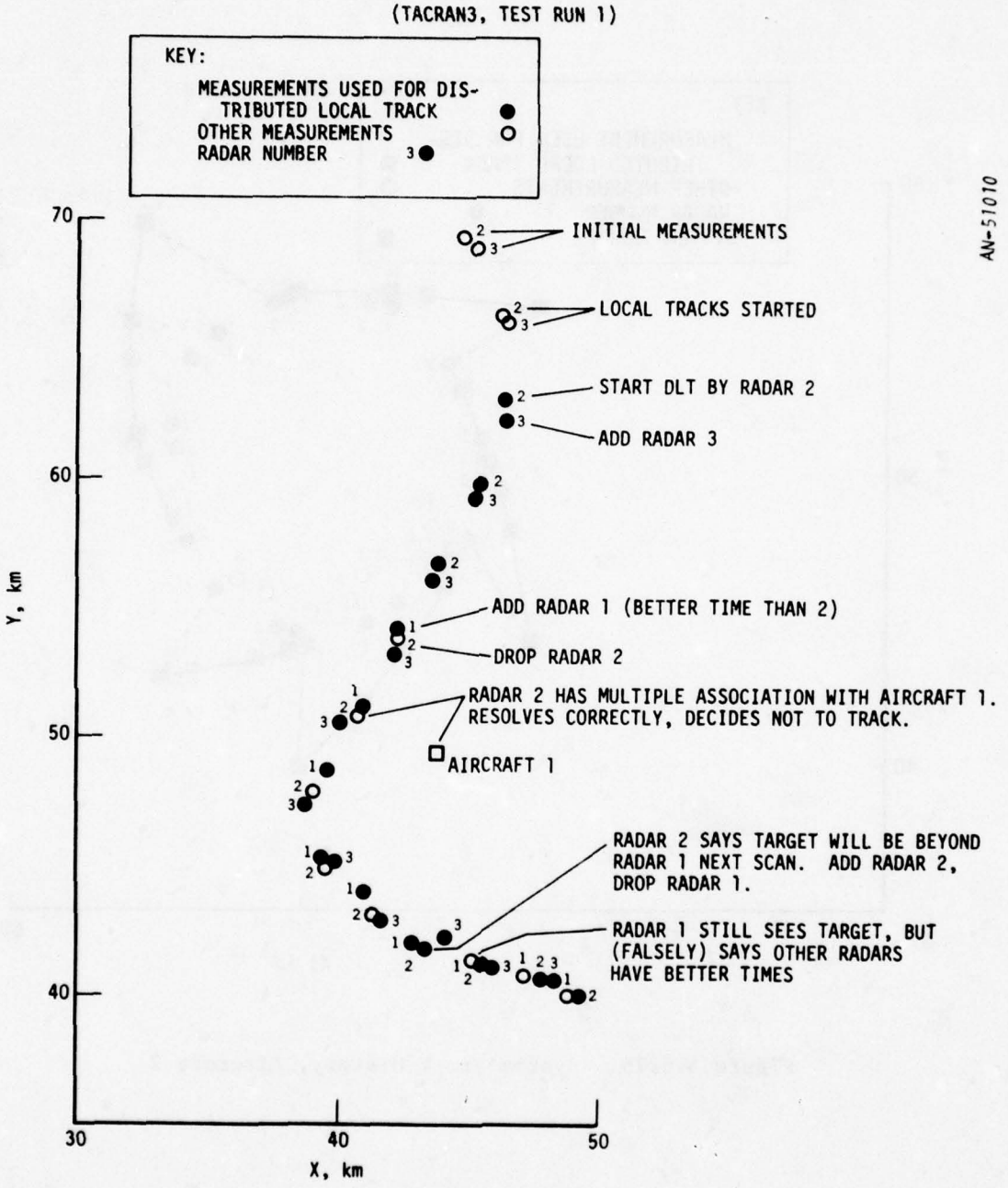


Figure 3.5.27. Distributed Local Track History, Aircraft 3

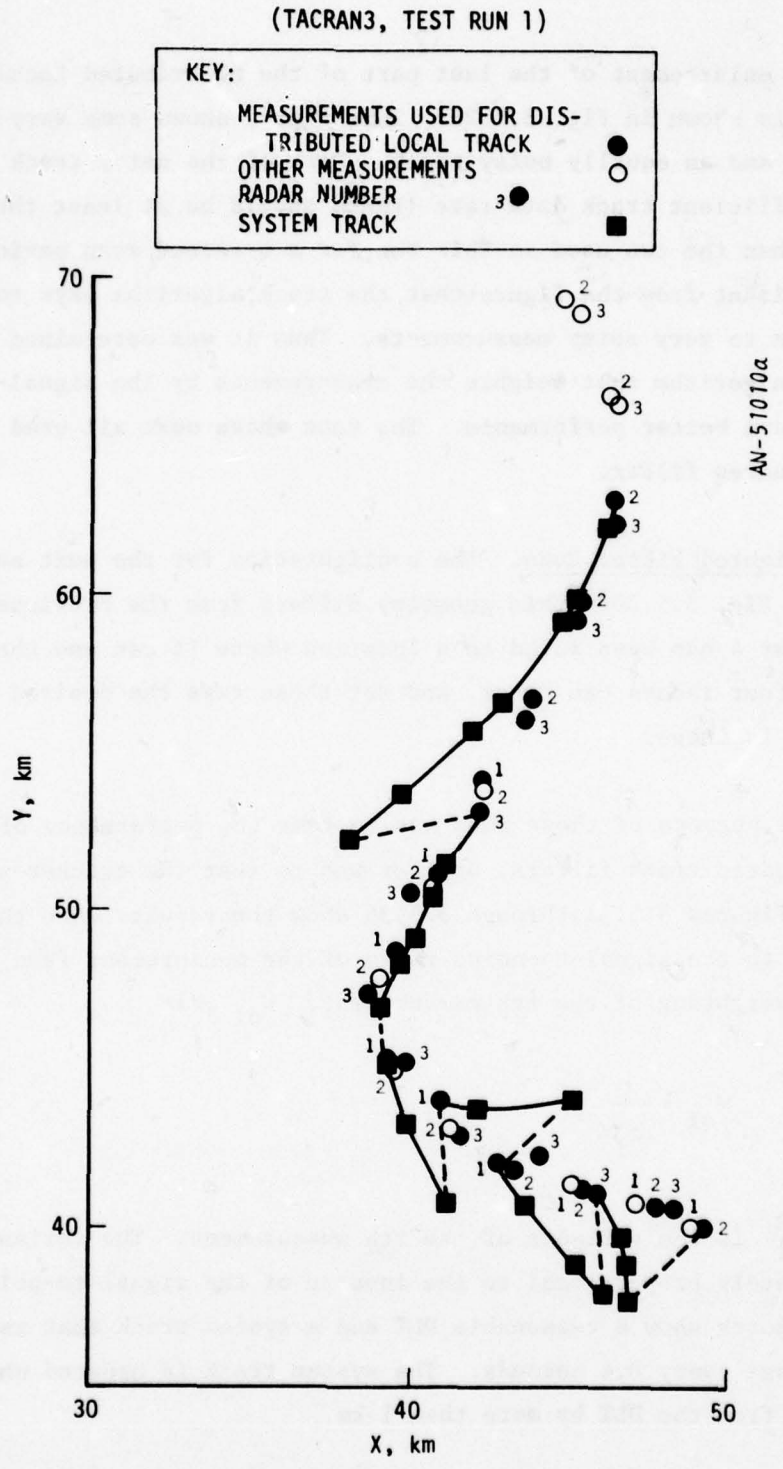


Figure 3.5.28. System Track History, Aircraft 3

An enlargement of the last part of the Distributed Local Track on Aircraft 3 is shown in Fig. 3.5.29. This figure shows some very noisy measurements and an equally noisy track. Part of the noisy track is caused by the insufficient track data rate (there should be at least three trackers rather than the two used in this run for a 6-second scan period). However, it is evident from the figure that the track algorithm pays too much attention to very noisy measurements. Thus it was determined that a least-squares algorithm that weights the measurements by the signal-to-noise ratio should have better performance. The runs shown next all used a weighted least-squares filter.

Weighted Filter Runs. The configuration for the next set of runs is shown in Fig. 3.5.30. This geometry differs from the previous geometry in that Radar 4 has been moved to a location where it can see the aircraft. Now all four radars can track, and for these runs the desired number of trackers is three.

One purpose of these runs was to test the performance of weighted least squares track filters, another was to test the tracker-selection logic. Figures 3.5.31 through 3.5.36 show the results when the weighting is equal to the signal-to-noise ratio of the measurement (Run 11). The optimum weighting of the ith measurement, W_{oi}, is

$$W_{oi} = \frac{1}{\sigma_i^2}$$

where σ_i^2 is the variance of the ith measurement. The variance is approximately proportional to the inverse of the signal-to-noise ratio. These figures show a reasonable DLT and a system track that is updated on the average every 8.4 seconds. The system track is updated when it deviates from the DLT by more than 1 km.

Figure 3.5.37 shows an expanded view of the distributed local track on Aircraft 3 between 46.7 and 88.1 seconds. The corresponding system track is expanded in Fig. 3.5.38. This illustrates a potential problem

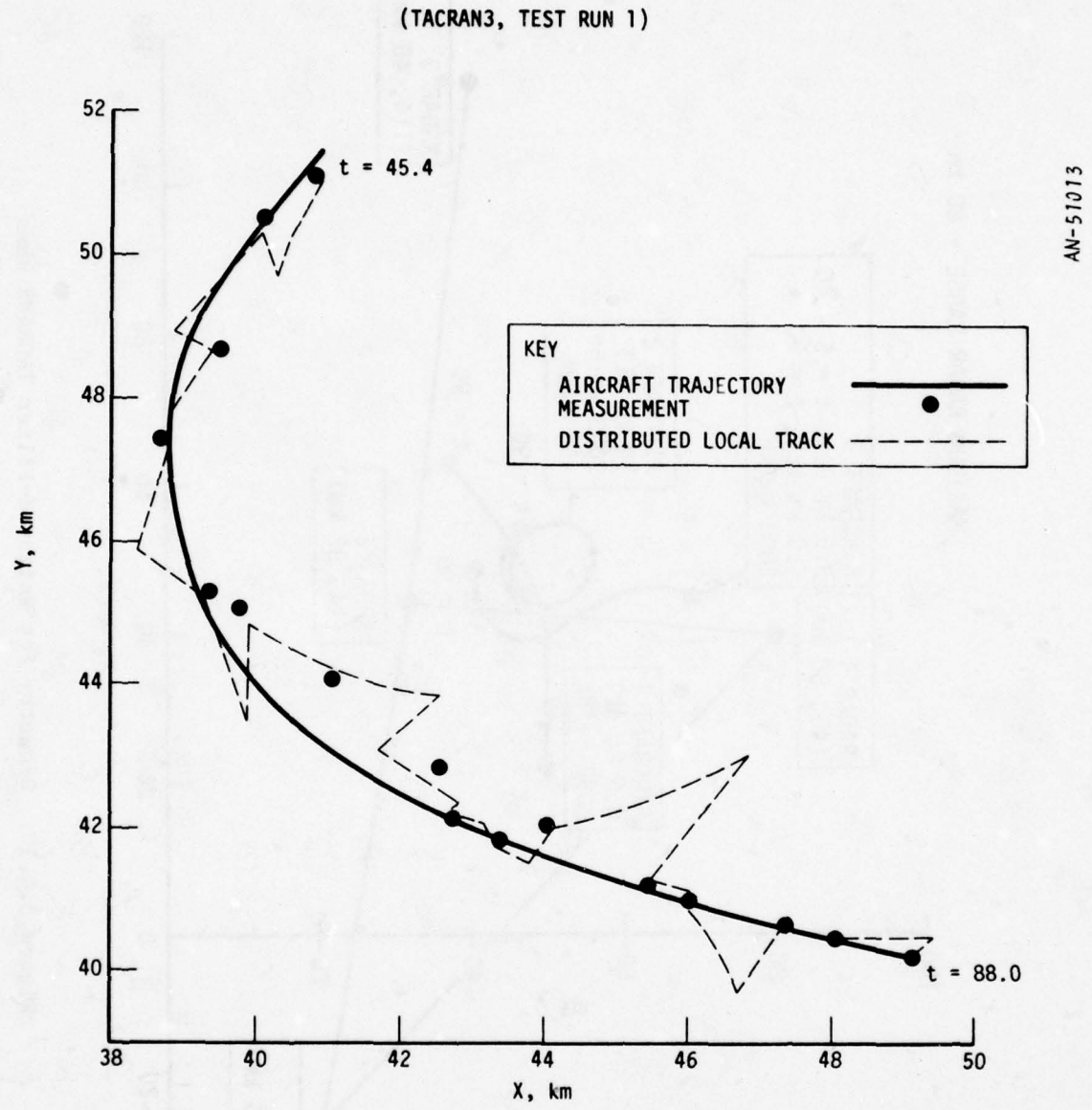


Figure 3.5.29. Detailed Track History, Aircraft 3

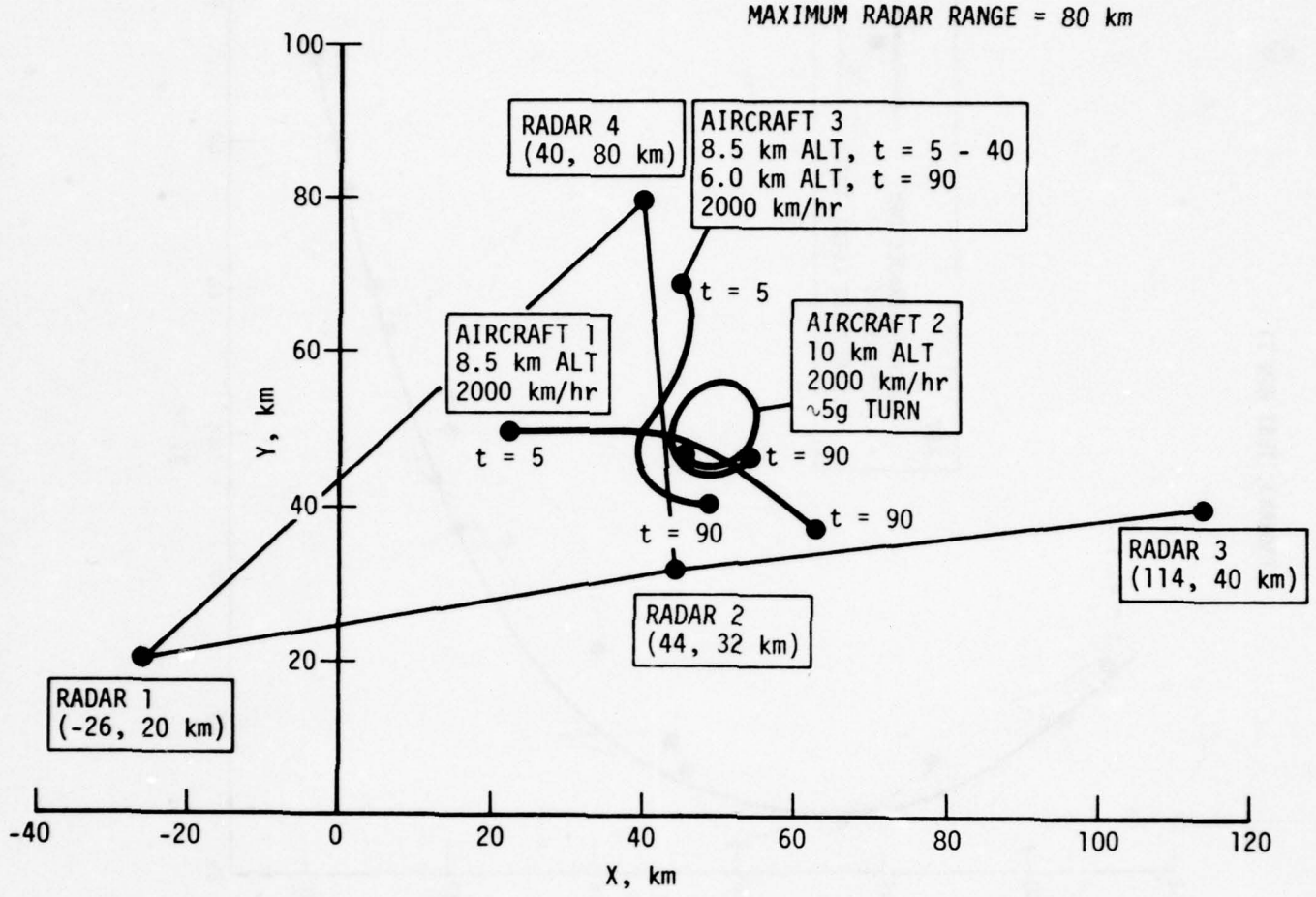


Figure 3.4.30. Geometry for Weighted-Filter TACRAN3 Runs

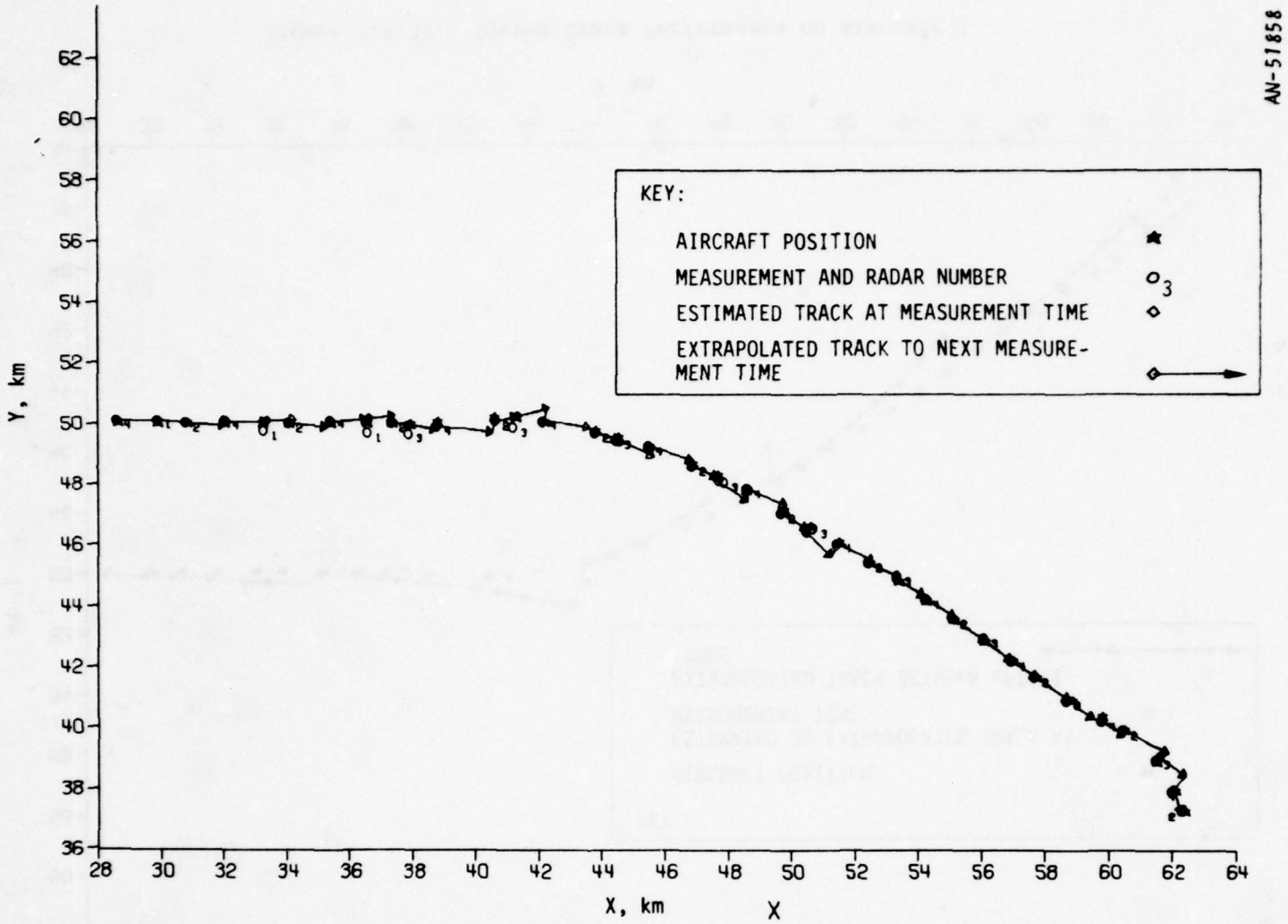


Figure 3.5.31. Distributed Local Track Performance on Aircraft 1

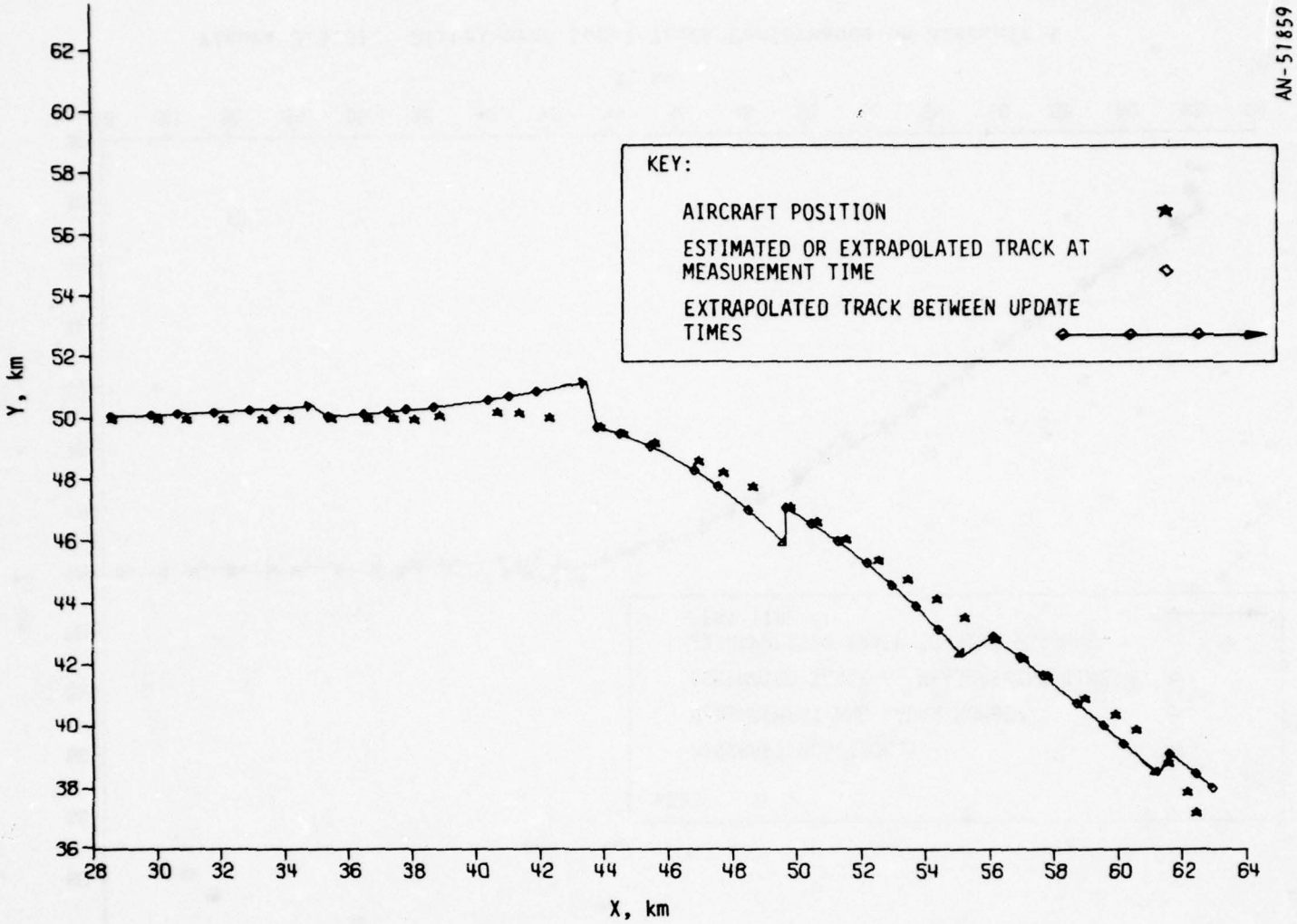


Figure 3.5.32. System Track Performance on Aircraft 1

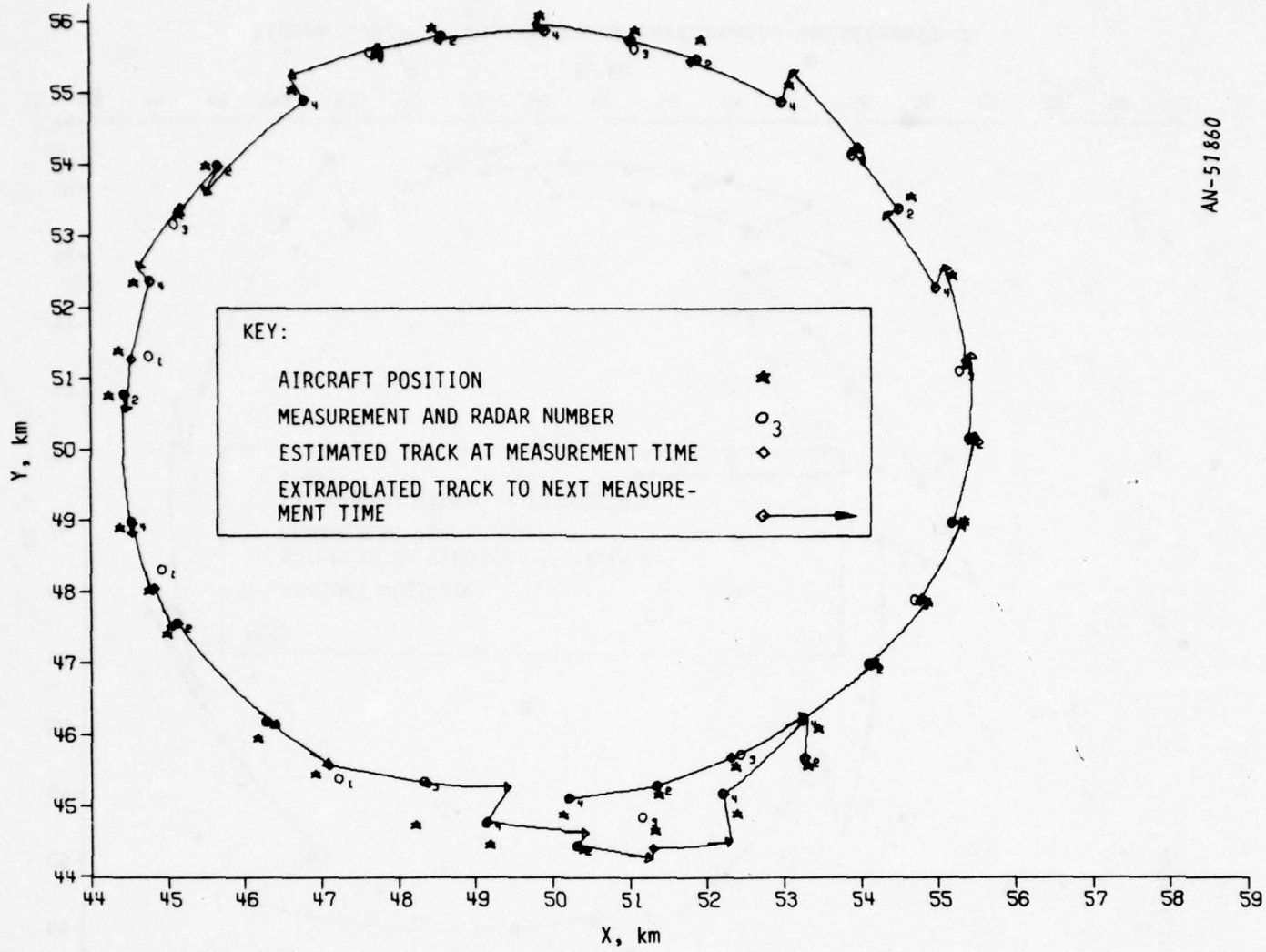


Figure 3.5.33. Distributed Local Track Performance on Aircraft 2

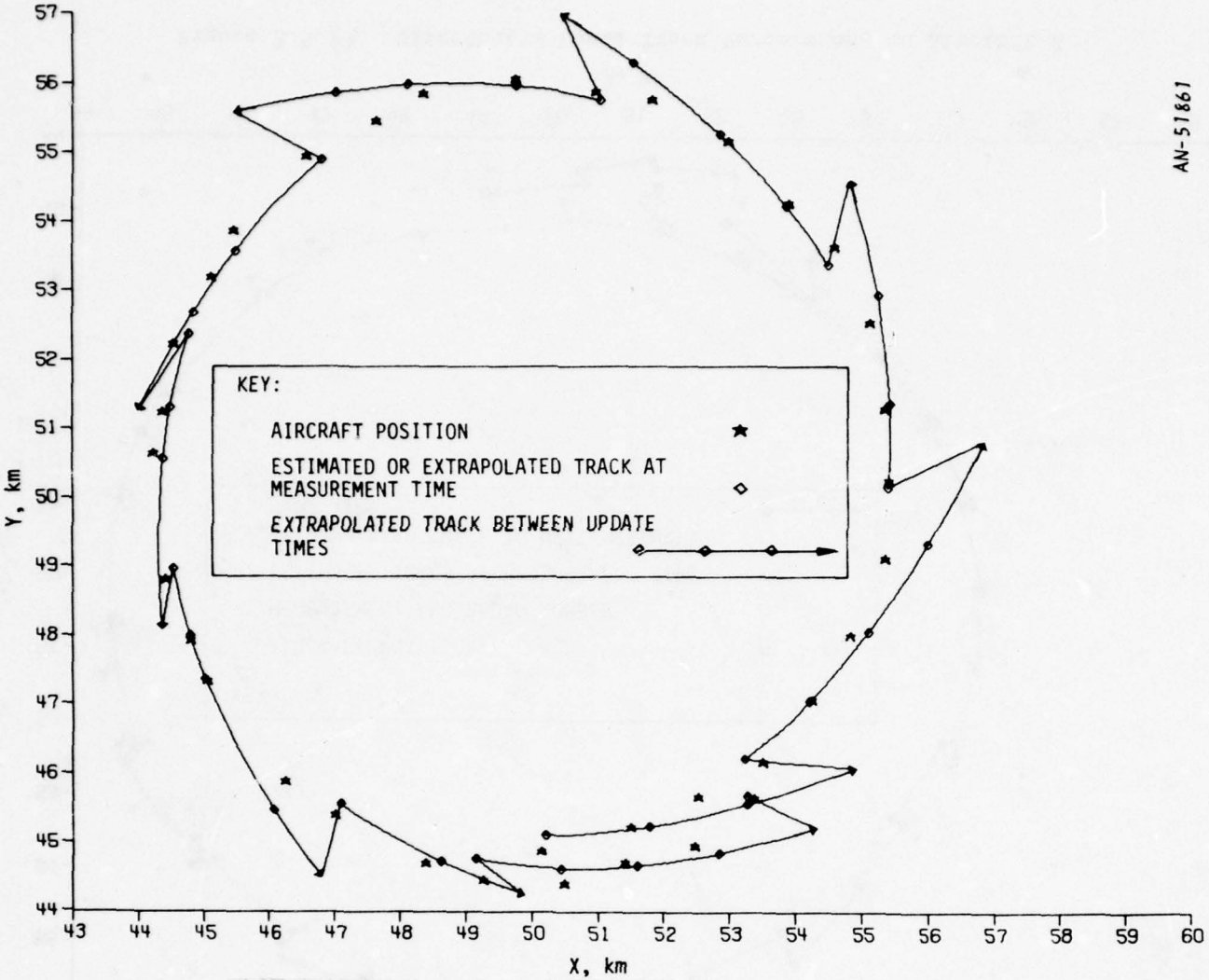


Figure 3.5.34. System Track Performance on Aircraft 2

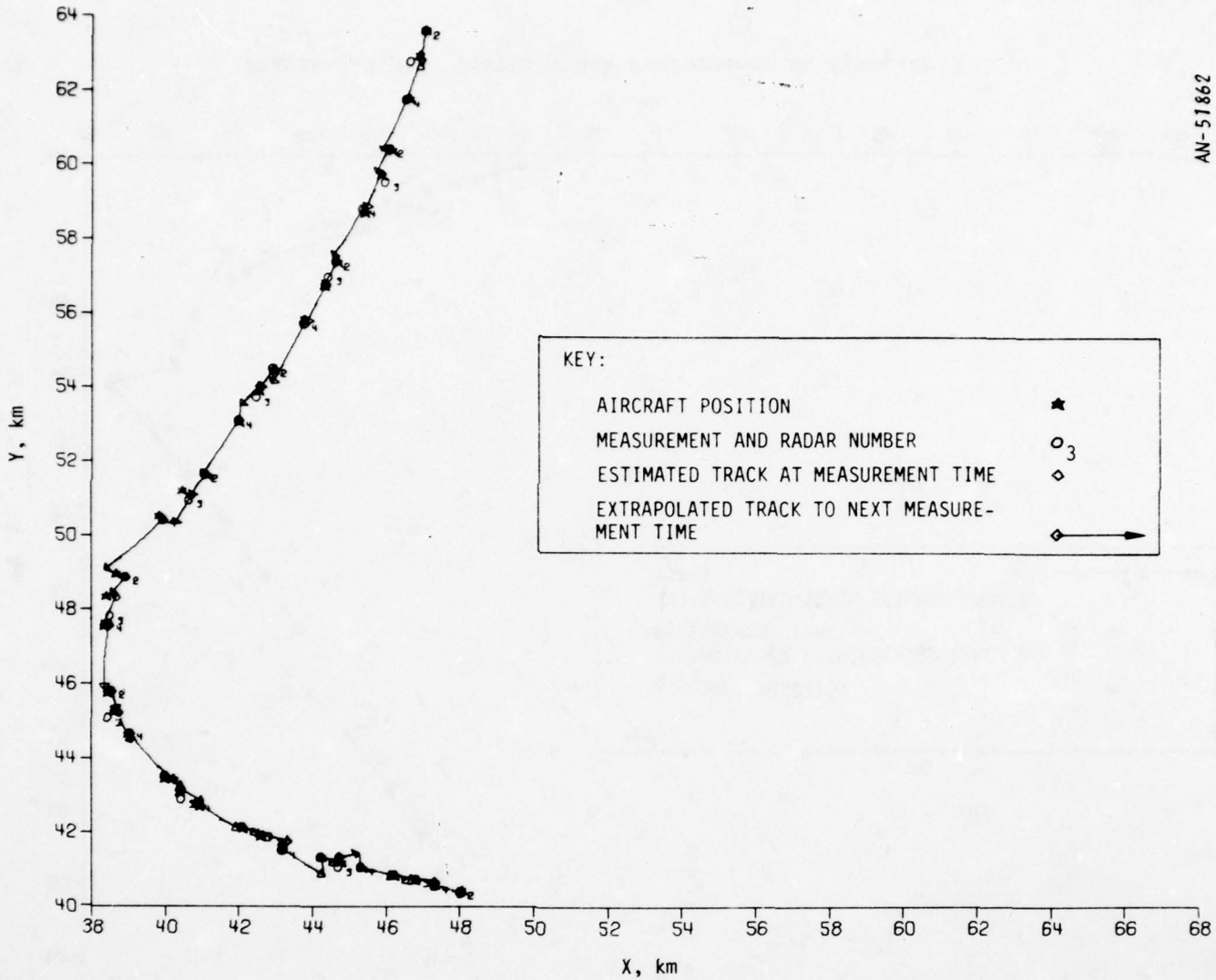


Figure 3.5.35. Distributed Local Track Performance on Aircraft 3

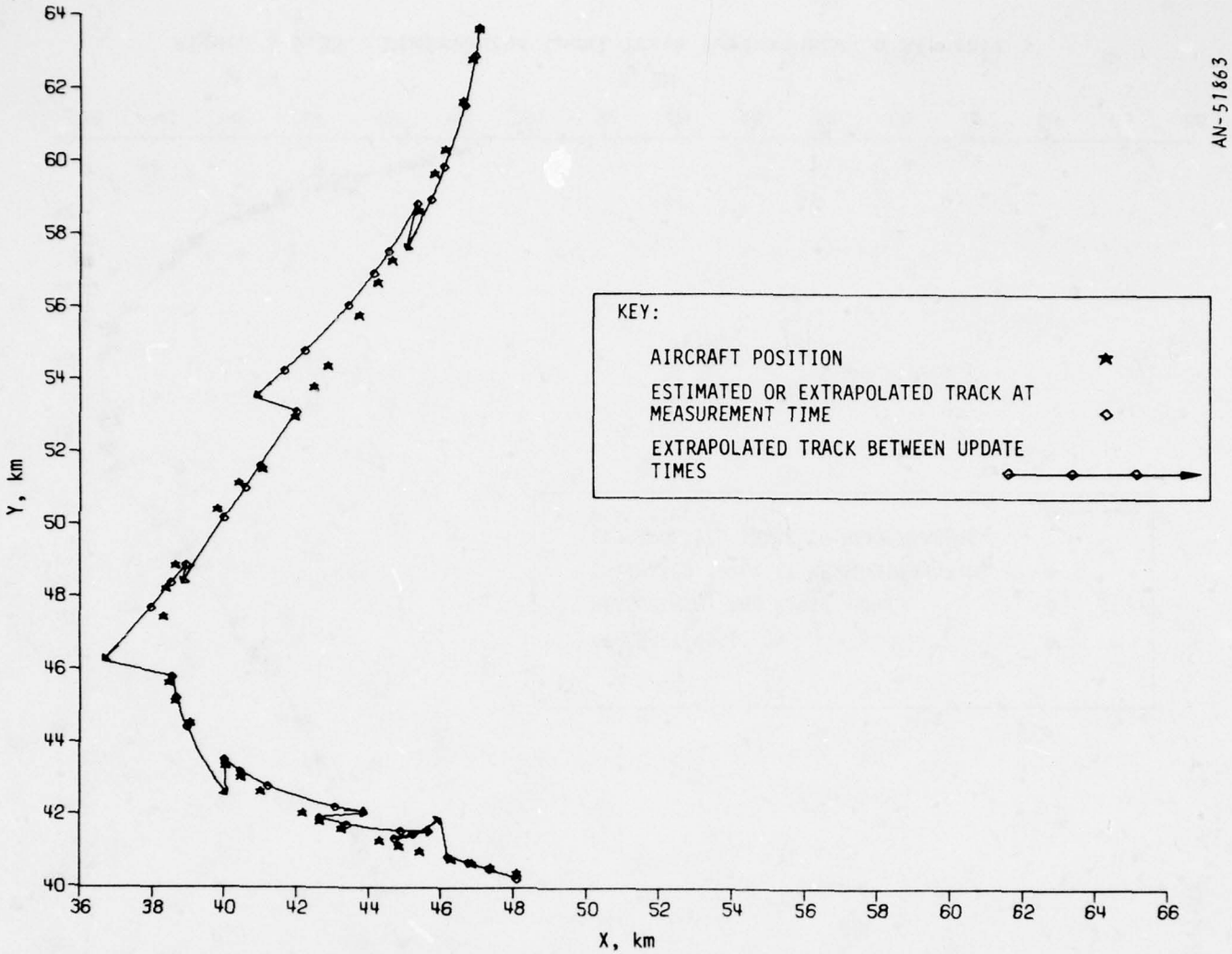


Figure 3.5.36. System Track Performance on Aircraft 3

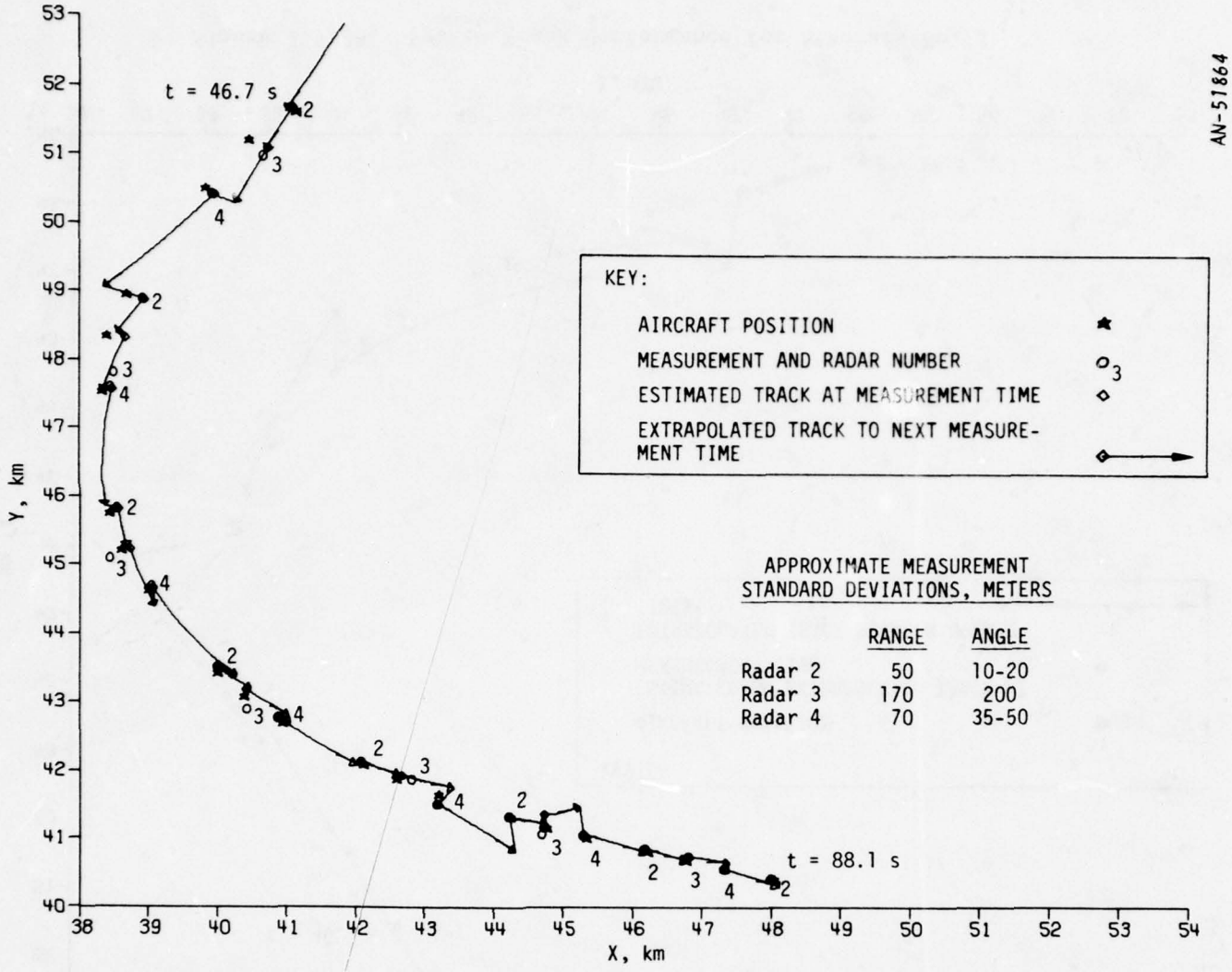


Figure 3.5.37. Distributed Local Track Performance for Aircraft No. 3

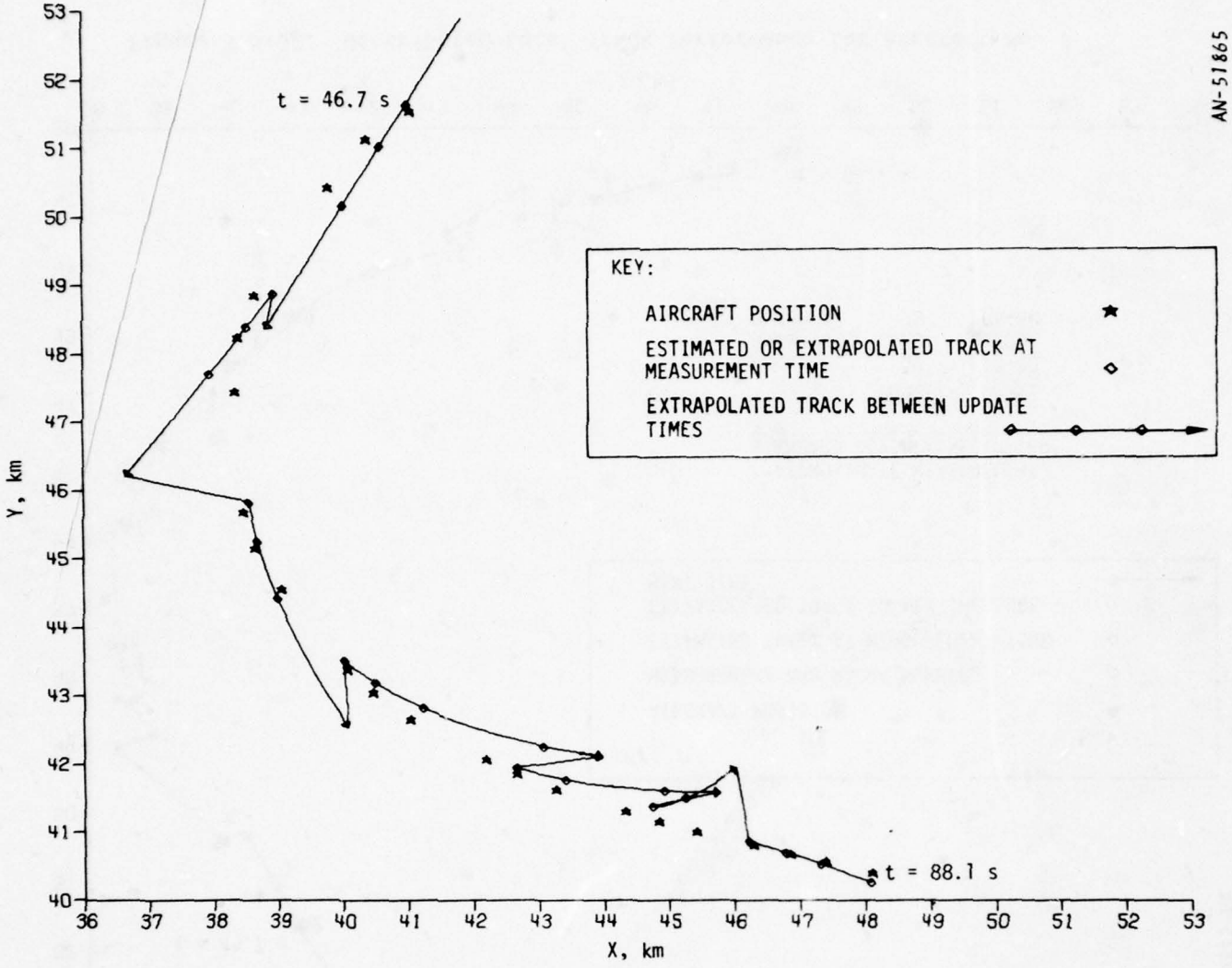


Figure 3.5.38. System Track Performance for Aircraft No. 3

in the concept of having a system track that is updated infrequently and suddenly. Certainly a display of as much as 40 seconds of a past track would be confusing with its discontinuities. However, a display of an aircraft's present estimated position and velocity vector should not be confusing, even with occasional discontinuities.

False-Alarm Run. To test the effect of a large number of false alarms on track initiation, association and track, a run (Run 9) was made with an average number of 100 false alarms per scan per radar. Two-scan track initiation occurred when two measurements produced a credible track with a speed less than 2,500 kt (1,286 m/s). The maximum association distance for initiating track with a third measurement was 3,300 m. The scan period was 6 seconds, the run duration 60 s, maximum range 80 km, maximum altitude 50 kft (15 km).

The results are summarized in Table 3.5.2. The actual average number of false alarms per scan was 98.8. This produced an average of 50.5 two-scan tracks per scan per radar. (This compares satisfactorily with the 41 predicted by the parametric analysis described in Sec. 2.2.) Distributed local tracks (three-scan) were initiated an average of 2.1 times per scan per radar. Seven tracks were sustained (using a 3,300-m association distance) through a fourth measurement; three through a fifth measurement. A history of these sustained tracks is shown in Table 3.5.3.

The association distance of 3,300 m is larger than would be required under some circumstances (better radar accuracy, less possible maneuver). Figure 3.5.39 shows a graph of the average number of three-measurement tracks per scan per radar that would have been initiated as a function of maximum association volume. As expected, it is a steep function of distance. The solid line in Fig. 3.5.39 is a plot of the number of track initiations on false alarms predicted for the three-measurement algorithm by the parametric analysis described in Sec. 2.2. The agreement between the simulation and analytic results is quite good.

TABLE 3.5.2

TACRAN3 FALSE-ALARM RUN SUMMARY

(Run 9)

Input Parameters

Average False Alarms per Scan	100
Two-Measurement (Scan) Track Initiation Maximum Speed	2,500 kt
Maximum Association Distance (6 seconds)	3,300 m
Scan Period	6 seconds
Run Duration	60 seconds

Results

	Average Scan per Radar			
	Radar 1	Radar 2	Radar 3	Total
False Alarms	99.8	103.0	93.7	98.8
Two-Scan Tracks	52.6	53.4	45.6	50.5 (29)
Three-Scan Tracks	2.3	2.4	1.8	2.1
Four-Scan Tracks				0.3
Five-Scan Tracks				0.2

Fifteen-Node Run. In an effort to understand the behavior of a larger network, a run (Run 10) consisting of 15 radar/data processor nodes was made. The geometry is shown in Fig. 3.5.40. The nodes are spaced 30 km apart on a hexagonal grid. These nodes can be considered to be a subset of a much larger network. Four of the five aircraft fly in pairs to test the track initiation and association logic.

An example of track initiation performance is shown in Fig. 3.5.41. Both aircraft fly straight paths, with Aircraft 1 slightly ahead of Aircraft 3. The lines indicate the six distributed local tracks that were created from the first three measurements made by one radar (Radar 12) on each aircraft. Clearly the four spurious tracks will eventually die out,

TABLE 3.5.3

SUSTAINED TRACKS FROM FALSE ALARMS

(TACRAN3 Run 9)

	Measurement					
	Third		Fourth		Fifth	
Track Label	Node	Time	Node	Time	Node	Time
2013	2	21.3	2	27.3	2	33.2
2774	3	26.0	3	38.0	2	50.0
2769	3	26.0	3	38.1		
4700	2	37.7	2	49.9	2	62.1
5095	1	39.4	1	51.3		
5561	1	41.9	3	45.4		
7575	2	52.4	2	64.5		

since they are heading away from the true aircraft. However, troubles could occur if these tracks accidentally associate with another aircraft (i.e., one that was not simulated).

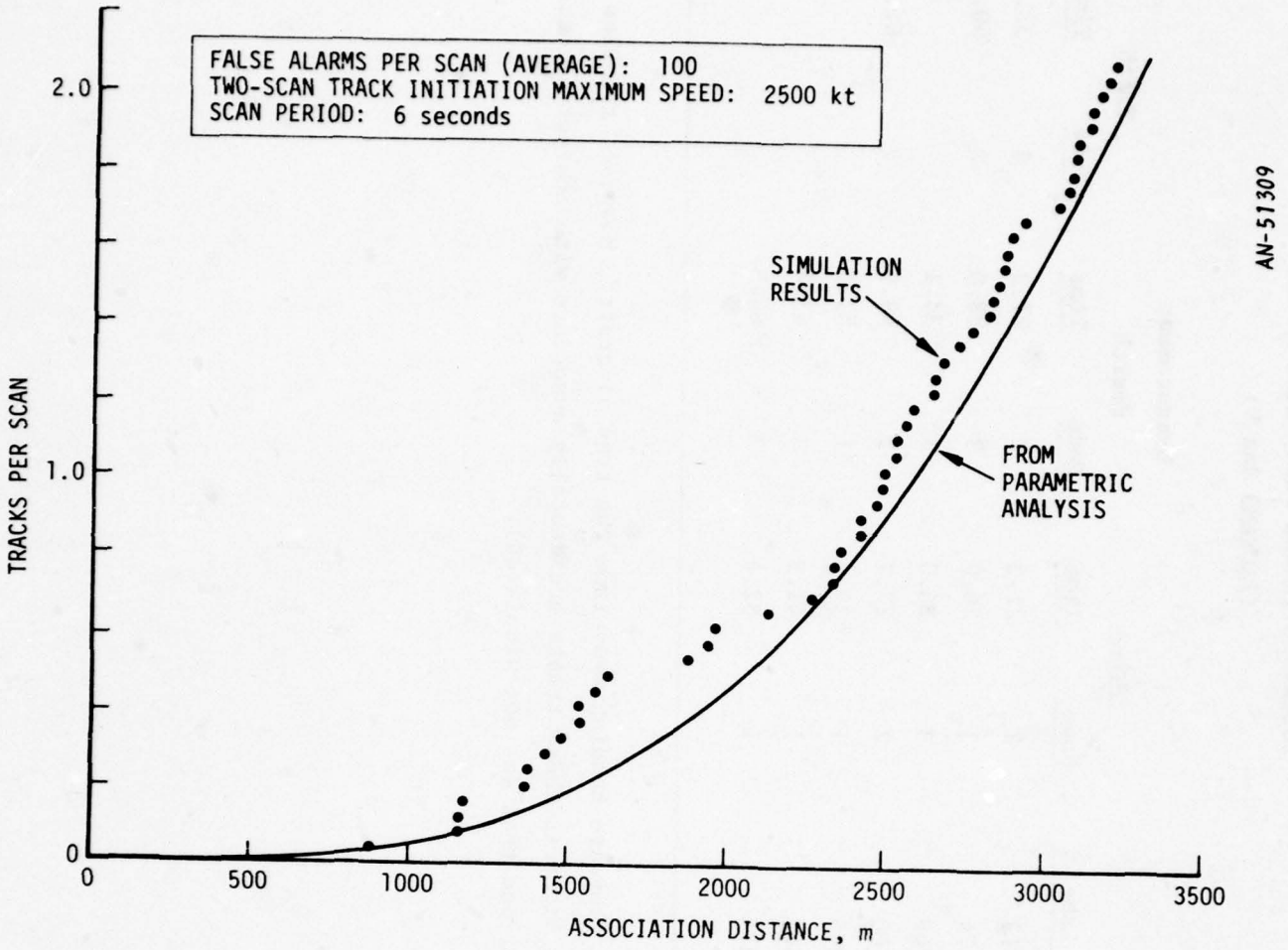


Figure 3.5.39. Three-Measurement (Scan) Track Initiations Per Scan Versus Association Distance

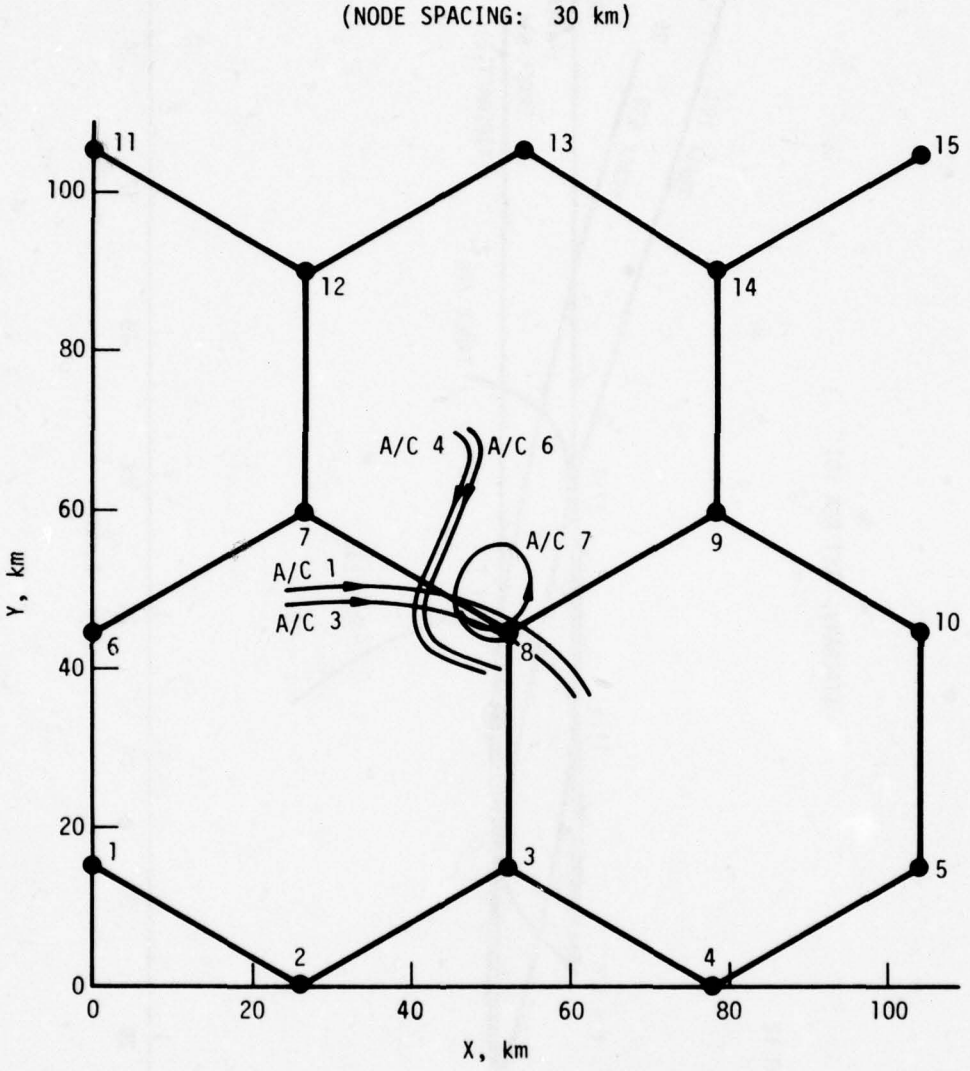


Figure 3.5.40. Geometry for 15-Node Run

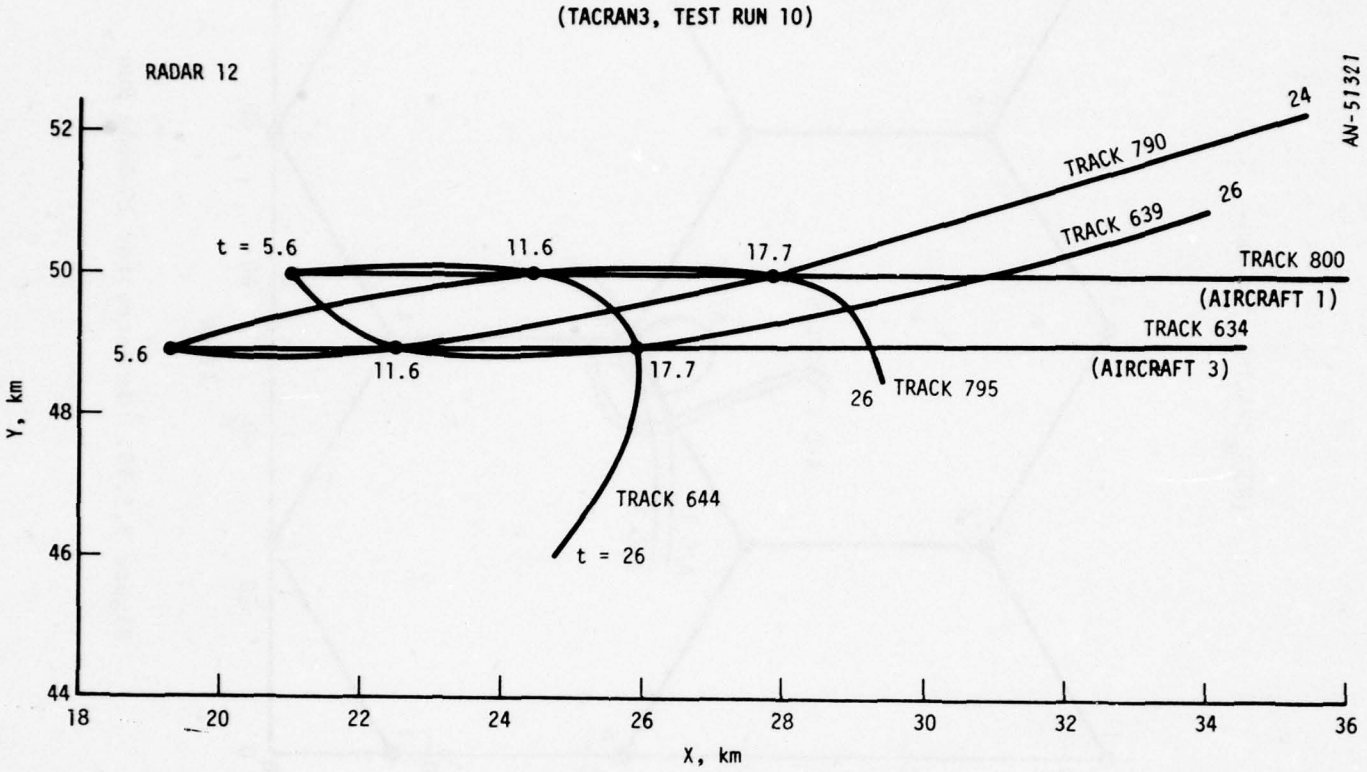


Figure 3.5.41. Track Initiation of Aircraft 1 and 3

3.5.4 TACRAN3 Printout

This appendix section shows three types of printout from TACRAN3: (1) data structure definition, (2) sample data structure, and (3) sample post processor output.

Data Structure Definition. The TACRAN simulation uses a dynamic storage allocation system (Sec. 3.6.2) which consists of datasets which are created as required either at input time or during the run, and which are destroyed during the run when no longer required. For example, each System Track at each node is represented by the data in a System Track Dataset. These individual datasets are accessed through Data Set Pointers (DSP), which may themselves reside in a dataset.

Datasets are collected together into lists, or files. A list is accessed by a List Header Variable (LHV), which may reside in a dataset.

The dataset structure in TACRAN3 is shown in Table 3.5.4. The first (basic) dataset (BO) is set up at input time and basic data required to link the simulation data together. It also contains some constants used by the Communication Simulation. Other data sets that follow describe the common radar parameters (RT), individual radar parameters (RD), individual data processor parameters (DP), measurements (ME), (two-scan) track initiation data (TL), system track data (TS), distributed local track data (TD), message data (MS), a series of message text datasets, communication link description (LK), aircraft data (AC, L2), and others required for running the simulation.

Input Data. Table 3.5.5 shows the printout of the input data for a test run of TACRAN3. This printout is from the FLEXREAD program (Sec. 3.6.4).

Post Processor Output. Table 3.5.6 shows the selected output from the post processor program (Sec. 3.6.6) for the test run of TACRAN3. The first output summarizes all measurements made by all radars (nodes). The data is ordered first by node, next by time aircraft, last by time of measurement. (An aircraft labeled zero means this return is a false alarm.) Table 3.5.6 shows the first page of the measurement printout.

Next the first page of the summary printout of track initiation is shown. The "TYPE" column describes what happened to the measurement. SAVE means the measurement was saved for future use in track initiation, LOCAL means a (two-scan) track was initiated, DLT means a new Distributed Local Track was created.

The third and fourth pages of Table 3.5.6 summarizes the track history of the Distributed Local Tracks. Also shown is the System Track position at each measurement time.

Other output (not shown because of its size) summarizes the message traffic.

TABLE 3.5.4

DATA STRUCTURE FOR THIRD VERSION OF TACRAN--TACTICAL AIR CONTROL RADAR NET

(BD) ---BASIC DATASET STRUCTURE NUMBER 1

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	LHV		EVENT LIST	
2	DSP	OUTSM BD	SUMMARY OUTPUT DATASET	
3	DSP	OVLY BD	OVERLAY STRUCTURE DATASET	
4	DSP	INOUT BD	INTERNAL OUTPUT FLAGS BY EVENT TYPE	
5	DSP	ASCO BD	A.S.C. OVERLAYING INSTRUCTIONS	
6	LHV	ASCL BD	LIST OF (TIME ORDERED) ACTION SEQUENCE CHAINS (THE ACTIVE ONES)	
7	LHV	RADLS BD	LIST OF ALL ACTIVE RADARS	
8	LHV	OBLIS BD	LIST OF ALL AIRCRAFT	
9	LHV	DPSYS BD	LIST OF DP WITH SYSTEM FILES	DP
10	LHV	CLINK BD	LIST OF DATALINKS	
11	R	SUPT BD	TIME INTERVAL BETWEEN SYS. STATUS OUTPUTS (SEC)	
12	I	L01 BD	BIT LENGTH OF DISTANCE REQUEST MESSAGE	
13	I	L02 BD	BIT LENGTH OF DISTANCE REPLY MESSAGE	
14	I	L03 BD	BIT LENGTH OF DLT REQUEST MESSAGE	
15	I	L04 BD	BIT LENGTH OF DLT REPLY MESSAGE	
16	I	L05 BD	BIT LENGTH OF MEASUREMENT POINT MESSAGE	
17	I	L11 BD	BIT LENGTH OF UPDATE SYSTEM TRACK MESSAGE	
18	I	L12 BD	BIT LENGTH OF DROP TRACKER MESSAGE	
19	I	L13 BD	BIT LENGTH OF ADD TRACKER MESSAGE	
20	I	L14 BD	BIT LENGTH OF HELP MESSAGE	
21	I	ML01 BD	BIT LENGTH OF MESSAGE HEADER AND CRC	
22	I	ML02 BD	BIT LENGTH OF EACH NODE IDENTITY IN A DESTINATION LIST	
23	R	HPRIO BD	HIGHEST MESSAGE PRIORITY	
24	R	WDELY BD	DELAY BETWEEN SUCCESSIVE NODE ROUTING MESSAGES	

TOTAL NUMBER OF WORDS IN DATASET IS 24

(RT) ---RADAR TYPE DATASET NUMBER 2

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	R	SCNTM RT	RADAR SCAN PERIOD	
2	R	SCAN RT	SCAN RATE FOR THE RADAR (RADIANS/SEC)	
3	R	RMSQ RT	RANGE ON 1 SQUARE METER	
4	R	THRES RT	DETECTION THRESHOLD (POWER MULTIPLIER)	
5	R (4)	ACOEF RT	SIGNAL/NOISE DEPENDENT PORTION OF MEASUREMENT ERRORS	
9	R (4)	BCOEF RT	THE SIGNAL/NOISE INDEPENDENT PORTIONS OF THE ERROR THE FOUR COEFFICIENTS ARE FOR R, A.E. AND RDOT, RESPECTIVELY	
13	R	FARAT RT	FALSE ALARM RATE (NUMBER OF FALSE ALARMS PER SCAN)	
14	R	FADIS RT	AVERAGE DISTANCE FROM RADAR FOR A FALSE ALARM	
15	R	RGRES RT	RANGE RESOLUTION	
16	R	AZRES RT	AZIMUTH RESOLUTION	
17	R	ELRES RT	ELEVATION RESOLUTION	
18	R	ELMAX RT	MAXIMUM RADAR ELEVATION ANGLE	
			DATA PROCESSING PARAMETERS (RADAR DEPENDENT)	
19	R	AMAXN RT	MAXIMUM A/C NORMAL ACCELERATION (IFASSCD)	
20	R	AMAXT RT	MAXIMUM A/C TANGENTIAL ACCELERATION (IFASSCD)	
21	R	DRDTT RT	DR/DT THRESHOLD (IFWORTH)	
22	R	DTD RT	DELAY IN PROCESSING DLTF (NEWRET)	
23	R	DTN RT	DELAY IN PROCESSING NTF (NEWRET)	
24	R	DTW RT	MAXIMUM WAIT FOR REPLY MESSAGE (NEWRET)	
25	R	MAXPT RT	MAXIMUM NUMBER OF POINTS FOR LEAST SQUARES (UPDATE)	

TABLE 3.5.4 (Cont.)

DATA STRUCTURE FOR THIRD VERSION OF TACRAN - TACTICAL AIR CONTROL RADAR NET

(RT) ---RADAR TYPE DATASET NUMBER 2

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
26	I	MINTK RT	DESIRED MINIMUM NUMBER OF TRACKERS PER TRACK (PFTRKD.N)	
27	R	RMAX RT	MAXIMUM RANGE OF RADAR (IFWORTH)	
28	R	RWRTH RT	MINIMUM RANGE FOR WHICH WORTH CONSIDERED (IFWORTH)	
29	R	SCNMX RT	MAXIMUM NUMBER OF SCANS FOR RANGES CROSSING (IFTRKD)	
30	R	SIGMA RT	NOMINAL MEASUREMENT CARTESIAN POSITION VARIANCE (IFASSCD)	
31	R	SPMAX RT	MAXIMUM CREDIBLE SPEED FOR LOCAL TWS INITIATION (TRKINIT)	
32	R	SWRTH RT	MAXIMUM S/N FOR WHICH WORTH CONSIDERED (IFWORTH)	
33	R	TDEAD RT	DELAY IN REMOVING TRACKER PRESUMED DEAD (IFTRKD)	
34	R	TIMAX RT	MAXIMUM TIME INTERVAL FOR LOCAL TWS INITIATION (TRKINIT)	
35	R	TIMIN RT	MINIMUM TIME INTERVAL FOR LOCAL TWS INITIATION (TRKINIT)	
36	R	TLASS RT	DISTANCE THRESHOLD FOR 3-PULSE TRACK INITIATION (TRKINIT)	
37	R	TMCK1 RT	THRESHOLD ON PRESENT SET OF TRACKERS (IFTMCHK)	
38	R	TMCK2 RT	THRESHOLD TO REPLACE A TRACKER (IFTMCHK)	
39	R	TMIN RT	MINIMUM SPACING BETWEEN TRACK PULSES (IFTRKD.IFTRKN)	
40	R	TNASS RT	SYSTEM TRACK ASSOCIATION DISTANCE (IFASSCN)	
41	R	TPD RT	DELAY IN PURGING OLD DISTRIBUTED TRACKS (IFASSCD)	
42	R	TPL RT	DELAY IN PURGING OLD MEASUREMENTS IN LTF (TRKINIT)	
43	R	TPN RT	DELAY IN PURGING OLD SYSTEM TRACKS (IFASSCN)	
44	R	TPTMX RT	MAXIMUM ALLOWED AGE OF OLD MEASUREMENT POINT (UPDATE)	

TOTAL NUMBER OF WORDS IN DATASET IS 44

(RD) ---INDIVIDUAL RADAR NUMBER 3

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL RD	RADAR LABEL NUMBER (SAME AS DP LABEL NUMBER)	
2	R	X RD	RADAR X-COORDINATE	
3	R	Y RD	RADAR Y-COORDINATE	
4	R	Z RD	RADAR Z-COORDINATE	
5	R	PONT RD	INITIAL POINTING DIRECTION OF BORESIGHT (RADS)	
6	DSP	TYPE RD	RADAR TYPE PARAMETERS	
7	DSP	DPROC RD	POINTER TO ASSOCIATED DATA PROCESSOR	
8	R	FANEX RD	NEXT FALSE ALARM TIME (OR ZERO IF NONE)	
9	DSP	NTARG RD	POINTER TO THE NEXT A/C TO PRODUCE A RETURN	

TOTAL NUMBER OF WORDS IN DATASET IS 9

(DP) ---DATA PROCESSOR NUMBER 4

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL DP	DATA PROCESSOR LABEL NUMBER (SAME AS RADAR LABEL NUMBER)	
2	R (3)	LOC DP	LOCATION OF DATA PROCESSOR	
5	H	TYPE DP	TYPE OF DATA PROCESSOR	
6	LHV	SAVDT DP	LIST OF SAVED DETECTIONS	
7	LHV	LTF DP	LOCAL (TWO-PULSE) TRACK FILE	MF
8	LHV	DLTF DP	DISTRIBUTED LOCAL TRACK FILE	TL
9	LHV	NTF DP	NON TRACK FILE (DLTF + NTF = SYSTEM TRACK FILE)	TS
10	LHV	COMM DP	LIST OF DIRECTLY CONNECTED DP	TS
11	LHV	CLINK DP	LIST OF DATALINKS	DP
12	DSP	RADAR DP	POINTER TO ASSOCIATED RADAR	RD
13	DSP	TEXT DP	POINTER TO MESSAGE TEXT	

TOTAL NUMBER OF WORDS IN DATASET IS 13

TABLE 3.5.4 (Cont.)

DATA STRUCTURE FOR THIRD VERSION OF TACRAN - TACTICAL AIR CONTROL RADAR NET

(ME) ---STANDARD RADAR MEASUREMENT DATASET — NUMBER 5

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL ME	UNIQUE LABLE FOR EACH MEASUREMENT	
2	R	TIME ME	TIME AT WHICH MEASUREMENT WAS TAKEN	
3	R	RANGE ME	RANGE OF MEASUREMENT	
4	R	AZIM ME	AZIMUTH	
5	R	ELEV ME	ELEVATION	
6	R	RDOT ME	RANGE RATE (IF MEASURED)	
7	R	SN ME	SIGNAL/NOISE RATIO	
8	R	RCS ME	EQUIVALENT RADAR CROSS SECTION	
9	DSP	RADAR ME	POINTER TO THE RADAR MAKING THE MEASUREMENT	
10	I	IFFID ME	SPACE RESERVED FOR IFF RETURN IF APPLICABLE	
11	R	X ME	MEASURED X COMPONENT	
12	R	Y ME	MEASURED Y COMPONENT	
13	R	Z ME	MEASURED Z COMPONENT	
14	DSP	IDTRU ME	POINTER TO AIRCRAFT	AC
			THE FOLLOWING ARE NOT A PART OF THE MEASUREMENT	
15	I	NTRK ME	NUMBER OF TRACKS THAT ASSOCIATE	
16	LHV	TKLST ME	LIST OF TRACKS THAT ASSOCIATE	TS
17	I	NWT ME	LENGTH OF WTLST	
18	LHV	WTLST ME	LIST OF MESSAGES WAITING FOR DISTANCE REPLY	WT
			OR LIST OF TRACKERS THAT HAVE BEEN ASKED FOR A DLT $DP	
19	I	NTIME ME	NUMBER OF RETRIES TO GET DISTANCES	
20	I (2)	IFLG ME	FLAGS FOR NTF PROCESSING (NEWRET)	
22	LHV	TKL ME	LIST OF TRACKS THAT STILL NEED DISTANCES	TS

TOTAL NUMBER OF WORDS IN DATASET IS 22

(WT) ---LIST OF GET-DISTANCE TEXTS — NUMBER 6

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	ID WT	WAIT ID FOR USE BY POST	
2	I	KREF WT	UNIQUE LABEL FOR RETURN MESSAGE	
3	DSP	DSPTS WT	POINTER TO TRACK	TS
4	DSP	DSPDP WT	POINTER TO TRACKER THAT REQUEST WAS SENT TO	DP
5	DSP	LDDSP WT	POINTER TO REPLY LD TEXT. IF 0, NO REPLY YET. IF -N, NEW	LD
			REQUEST ON THIS TRACK IN NTH POSITION ON LIST.	

TOTAL NUMBER OF WORDS IN DATASET IS 5

(TL) ---LOCAL (TWO-PULSE) TRACK FILE DATASET — NUMBER 7

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL TL	UNIQUE LABEL FOR EACH DATASET	
2	R	TIM1 TL	TIME OF FIRST PULSE	
3	R	X1 TL	MEASURED X COMPONENT (FIRST PULSE)	
4	R	Y1 TL	MEASURED Y COMPONENT (FIRST PULSE)	
5	R	Z1 TL	MEASURED Z COMPONENT (FIRST PULSE)	
6	I	LBLA1 TL	TRUE A/C LABEL NUMBER (FIRST PULSE)	
7	R	TIM2 TL	TIME OF SECOND PULSE	
8	R	X2 TL	MEASURED X COMPONENT (SECOND PULSE)	
9	R	Y2 TL	MEASURED Y COMPONENT (SECOND PULSE)	
10	R	Z2 TL	MEASURED Z COMPONENT (SECOND PULSE)	
11	I	LBLA2 TL	TRUE A/C LABEL NUMBER (SECOND PULSE)	
12	R	XDOT TL	ESTIMATED X VELOCITY	
13	R	YDOT TL	ESTIMATED Y VELOCITY	

TABLE 3.5.4 (Cont.)

DATA STRUCTURE FOR THIRD VERSION OF TACRAN - TACTICAL AIR CONTROL RADAR NET

(TL) ---LOCAL (TWO-PULSE) TRACK FILE DATASET — NUMBER 7

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
14	R	ZDOT TL	ESTIMATED Z VELOCITY	

TOTAL NUMBER OF WORDS IN DATASET IS 14

(TS) ---SYSTEM TRACK FILE DATASET — NUMBER 8

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL TS	OBJECT LABEL NUMBER	
2	R	TUPD TS	TIME OF LAST UPDATE	
3	R (9) C	TS	STATE ESTIMATE COEFFICIENTS= X0,X1,X2,Y0,Y1,Y2,Z0,Z1,Z2. X(T) = X0 + X1*T + X2*T**2	
12	I	NTKRS TS	PRESENT NUMBER OF TRACKING DATA PROCESSORS	
13	LHV	TKLST TS	LIST OF DATASETS THAT DESCRIBE TRACKING DATA PROCESSORS	TK
14	DSP	DLT TS	POINTER TO DISTRIBUTED LOCAL TRACK DATASET (0 IF NONE)	TD
15	I	LPLAC TS	TRUE A/C LABEL NUMBER	
16	I	NASSC TS	NUMBER OF MEASUREMENTS THAT ASSOCIATE	
17	LHV	ASLST TS	LIST OF MEASUREMENTS THAT ASSOCIATE	AS

TOTAL NUMBER OF WORDS IN DATASET IS 17

(TK) ---TRACKING DATA PROCESSOR DATASET — NUMBER 9

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	DSP	DSPDP TK	POINTER TO DATA PROCESSOR	DP
2	R	TIME TK	TIME OF LAST KNOWN UPDATE	
3	I	IHELP TK	1 IF THIS TRACKER NEEDS HELP. 0 OTHERWISE	

TOTAL NUMBER OF WORDS IN DATASET IS 3

(TD) ---DISTRIBUTED LOCAL TRACK DATASET — NUMBER 10

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	R	TMEAS TD	TIME OF LAST MEASUREMENT	
2	R (9) C	TD	STATE ESTIMATE COEFFICIENTS= X0,X1,X2,Y0,Y1,Y2,Z0,Z1,Z2. X(T) = X0 + X1*T + X2*T**2	
11	I	NPRPT TD	NUMBER OF POINTS (DATASETS) IN PTLST	
12	LHV	PTLST TD	LIST OF MEASUREMENT POINTS	PT
13	DSP	DSPTS TD	POINTER TO SYSTEM TRACK DATASET	TS

TOTAL NUMBER OF WORDS IN DATASET IS 13

(PT) ---MEASUREMENT POINT DATASET — NUMBER 11

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	R	TMEAS PT	TIME OF MEASUREMENT	
2	R	X PT	MEASURED X COORDINATE	
3	R	Y PT	MEASURED Y COORDINATE	
4	R	Z PT	MEASURED Z COORDINATE	

TOTAL NUMBER OF WORDS IN DATASET IS 4

(AS) ---ASSOCIATION DATASET — NUMBER 12

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	DSP	DSPME AS	POINTER TO MEASUREMENT	ME
2	R	DSQ AS	DISTANCE-SQUARED	

TOTAL NUMBER OF WORDS IN DATASET IS 2

(MS) ---MESSAGE HEADER DATASET — NUMBER 13

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	DSP	SNDR MS	POINTER TO ORIGINATOR OF MESSAGE	DP
2	LHV	RCVRS MS	LIST OF NODES (DATA PROCESSORS) TO RECEIVE MESSAGE	DP

CONTAINS 0 IF ALL NODES ARE TO RECEIVE MESSAGE (SYSTEM MESSAGE)
--NOTE--ON RECEPTION CONTAINS POINTER TO RECEIVER

TABLE 3.5.4 (Cont.)

DATA STRUCTURE FOR THIRD VERSION OF TACRAN - TACTICAL AIR CONTROL RADAR NET

(MS) ---MESSAGE HEADER DATASET — NUMBER 13

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
3	DSP	CNDR MS	POINTER TO SENDER OF THIS COPY (FILLED IN BY SYSTEM)	DP
4	I	LNTH MS	NUMBER OF BITS IN TEXT	
5	I	NBIL MS	NUMBER OF BITS IN THE RCVRS LIST (FILLED IN BY SYSTEM)	
6	I	MTYP MS	TYPE OF MESSAGE	
7	I	LABL MS	UNIQUE LABEL FOR EACH MESSAGE	
8	P	PRIO MS	PRIORITY OF MESSAGE	
9	I	KREF MS	ID OF WAIT MESSAGE	
10	DSP	TEXT MS	POINTER TO MESSAGE TEXT	

TOTAL NUMBER OF WORDS IN DATASET IS 10

(MD) ---DISTRIBUTED LOCAL TRACK MEASUREMENT TEXT DATASET — NUMBER 14

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL MD	OBJECT LABEL NUMBER	
2	R	TMEAS MD	TIME OF MEASUREMENT	
3	R	X MD	MEASURED X COORDINATE	
4	R	Y MD	MEASURED Y COORDINATE	
5	R	Z MD	MEASURED Z COORDINATE	

TOTAL NUMBER OF WORDS IN DATASET IS 5

(DT) ---DROP-TRACKER TEXT DATASET — NUMBER 15

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL DT	OBJECT LABEL NUMBER	
2	DSP	DROP DT	POINTER TO TRACKER TO BE DROPPED	DP

TOTAL NUMBER OF WORDS IN DATASET IS 2

(AT) ---ADD-TRACKER TEXT DATASET — NUMBER 16

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL AT	OBJECT LABEL NUMBER	
2	DSP	ADD AT	POINTER TO TRACKER TO BE ADDED	DP
3	R	TIME AT	TIME OF NEW TRACKERS LAST UPDATE	
4	DSP	DROP AT	POINTER OF TRACKER TO BE DROPPED (0 IF NONE)	DP

TOTAL NUMBER OF WORDS IN DATASET IS 4

(DL) ---REQUEST FOR DISTANCES TEXT DATASET — NUMBER 17

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL DL	OBJECT LABEL NUMBER	
2	R	TMEAS DL	TIME OF MEASUREMENT	
3	R	X DL	MEASURED X COORDINATE	
4	R	Y DL	MEASURED Y COORDINATE	
5	R	Z DL	MEASURED Z COORDINATE	
6	DSP	DSPTS DL	POINTER TO OBJECT TRACK	TS
7	DSP	DSPME DL	POINTER TO MEASUREMENT DATASET	ME

TOTAL NUMBER OF WORDS IN DATASET IS 7

(LD) ---REPLY WITH DISTANCES TEXT DATASET — NUMBER 18

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	DSP	DSPTS LD	POINTER TO OBJECT TRACK	TS
2	DSP	DSPME LD	POINTER TO MEASUREMENT	ME
3	R	DSQ LD	DISTANCE SQUARED	

TOTAL NUMBER OF WORDS IN DATASET IS 3

(DS) ---REQUEST FOR DISTRIBUTED LOCAL TRACK TEXT DATASET — NUMBER 19

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL DS	OBJECT LABEL NUMBER	

TABLE 3.5.4 (Cont.)

DATA STRUCTURE FOR THIRD VERSION OF TACRAN - TACTICAL AIR CONTROL RADAR NET

(DS) ---REQUEST FOR DISTRIBUTED LOCAL TRACK TEXT DATASET — NUMBER 19

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
2	DSP	DSPTS DS	POINTER TO OBJECT TRACK	TS

TOTAL NUMBER OF WORDS IN DATASET IS 2

(MH) ---HELP MESSAGE TEXT DATASET — NUMBER 20

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL MH	OBJECT LABEL NUMBER	
2	DSP	DSPDP MH	POINTER TO DATA PROCESSOR	DP
3	R	TIME MH	TIME OF LAST KNOW UPDATE	
4	I	IHELP MH	1 IF THIS TRACKERS NEEDS HELP, 0 OTHERWISE	

TOTAL NUMBER OF WORDS IN DATASET IS 4

(LK) ---DATALINK DESCRIPTION — NUMBER 21

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	DSP	NODE1 LK	POINTER TO ONE OF THE -DP- CONNECTED TO THIS LINK	
2	DSP	NODE2 LK	POINTER TO THE OTHER DP	
3	R	BAUD LK	TRANSMISSION RATE FOR THIS LINK (IN BITS/SEC)	
			THE FOLLOWING DATA DO NOT HAVE TO BE PROVIDED AT INPUT TIME	
4	I	BUSY LK	LINK STATUS AND QUE IDENT	
5	LHV	QUE1 LK	QUEUE FOR UNDELIVERED MESSAGES IN ONE DIRECTION	
6	LHV	QUE2 LK	QUEUE FOR UNDELIVERED MESSAGES IN THE OTHER DIRECTION	

TOTAL NUMBER OF WORDS IN DATASET IS 6

(AC) ---OBJECT BEING LOOKED AT (USUALLY AN AIRCRAFT) — NUMBER 22

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	LABL AC	A/C LABEL NUMBER	
2	DSP	LOCN AC	POINTER TO LOCATION MODEL FOR THIS A/C	
3	R	XSEC AC	RADAR CROSS SECTION OF THE OBJECT	
4	I	NCPT AC	INDEX OF CURRENT AIM POINT	
5	A (3)	STAT AC	ST VEC--T,X,Y,Z,XD,YD,ZD,TH,PSI,THD,PSID,VELD	
18	R	FTIM AC	TIME OF CREATION OF THIS A/C	

TOTAL NUMBER OF WORDS IN DATASET IS 18

(L1) ---STRAIGHTLINE FLIGHT MODEL FOR A/C MOTION — NUMBER 23

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	H	KTYP L1	TYPE OF MODEL REPRESENTED BY DATA BELOW	
			STRAIGHT = CONSTANT SPEED ALONG STRAIGHT LINE	
2	R	TBEG L1	BEGINNING TIME FOR THIS MODEL	
3	R (3)	PBEG L1	LOCATION OF THE BEGINNING POINT	
6	R	TEND L1	ENDING TIME FOR THIS MODEL	
7	R (3)	PEND L1	LOCATION OF END POINT OF A/C TRAJECTORY	

TOTAL NUMBER OF WORDS IN DATASET IS 9

(L2) ---POINT MASS AIRCARFT MODEL,G-LIMITED PROP. NAVIGATION — NUMBER 24

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	H	KTYP L2	TYPE OF MODEL REPRESENTED BY BELOW DATA 6HPTMASS	
2	R	DELT L2	MAXIMUM TIME STEP	
3	A (0)	AIMP L2	X,Y,Z,SPEED AIM POINTS --MAX OF 10	

TOTAL NUMBER OF WORDS IN DATASET IS 42

(JM) ---JAMMING SOURCE — NUMBER 25

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	R (3)	LOC JM	LOCATION OF THE JAMMING SOURCE	
4	H	TYPE JM	TYPE OF EMITTER	

TOTAL NUMBER OF WORDS IN DATASET IS 4

TABLE 3.5.4 (Concl.)

DATA STRUCTURE FOR THIRD VERSION OF TACRAN - TACTICAL AIR CONTROL RADAR NET

(XA) ---TRANSACTION DATASET FOR GENERAL ACTION SEQUENCE CHAIN — NUMBER 26

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	H	NASC XA	HOLLERITH IDENTIFIER OF A.S.C. INVOLVED	
2	R	TIME XA	TIME FOR NEXT EVENT	
3	I	NUMR XA	COUNTER UNIQUELY IDENTIFYING THIS TRANSACTION	
4	H	NEXN XA	INTERNAL LOCATION WHERE LEFT OFF PROCESSING	
5	DSP	DATA XA	POINTER TO (USER CONSTRUCTED) DATASET CONTAINING ANY OTHER DATA	

TOTAL NUMBER OF WORDS IN DATASET IS 5

(XW) ---SYSTEM GENERATED -WAIT- DATASET FOR WAIT AND POST ROUTINES — NUMBER 27

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	DSP	IXA XW	POINTER TO THE WAITING TRANSACTION	
2	R	TIME XW	TIME AT WHICH TRANSACTION IS TO BE REACTIVATED	
3	I	ID XW	DISTINGUISHING -ID- NUMBER FOR THIS -WAIT-	
4	H	ITIM XW	EVENT IN ASC TO RESTART AT IN THE -TIMES UP- CASE	
5	H	IPOS XW	EVENT IN ASC TO RESTART AT IN THE -POST- CASE	

TOTAL NUMBER OF WORDS IN DATASET IS 5

(CF) ---CAPTURABLE FACILITY DATASET — NUMBER 28

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	H	NAME CF	IDENTIFYING NAME OF THE FACILITY	
2	LHV	USLS CF	CURRENT LIST OF WAITING REQUESTORS	
3	DSP	KDSP CF	POINTER TO TRANSACTION CURRENTLY USING THE FACILITY	

TOTAL NUMBER OF WORDS IN DATASET IS 3

(CL) ---THE -WAITING- DATASET FOR THE USER OF A CAPTURABLE FACILITY — NUMBER 29

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	DSP	KDSP CL	POINTER TO ASSOCIATED -WAITING- TRANSACTION	

TOTAL NUMBER OF WORDS IN DATASET IS 1

(SF) ---DATA STRUCTURE FOR A SHARED FACILITY — NUMBER 30

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	H	NAME SF	IDENTIFYING NAME OF THE FACILITY	
2	R	CAPY SF	TOTAL CAPACITY OF THE FACILITY	
3	LHV	USLS SF	LIST OF THE CURRENT USERS OF THE FACILITY	
4	H	LAWS SF	IDENTIFICATION OF THE LAWS GOVERNING FACILITY UTILIZATION	
5	R	DMND SF	CURRENT TOTAL DEMAND FOR THE SERVICES OF THE FACILITY	

TOTAL NUMBER OF WORDS IN DATASET IS 5

(SL) ---STRUCTURE OF THE DATASETS ON THE -SHARE LIST- OF A SHARED FACILITY — NUMBER 31

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	R	AMRQ SL	AMOUNT REQUESTED BY THIS USER OF THE FACILITY	
2	R	ATGO SL	AMOUNT OF THE FACILITY STILL REQUIRED	
3	R	TIME SL	TIME AT WHICH THE ABOVE CALCULATION WAS VALID	
4	R	PELT SL	PREDICTED RELEASE TIME FOR THIS USER	
5	DSP	XADS SL	POINTER TO THE TRANSACTION DATASET FOR THIS USER	

TOTAL NUMBER OF WORDS IN DATASET IS 5

(E0) ---START/STOP EVENT — NUMBER 32

WORD	TYPE	MNEMONIC	--ITEM IDENTIFICATION--	-POINTS TO-
1	I	KTYP E0	EVENT TYPE INDICATOR (ALWAYS = 1)	
2	R	START E0	START TIME FOR THE EVENT	
3	R	STOP E0	SIMULATION STOPPING TIME	

TOTAL NUMBER OF WORDS IN DATASET IS 3

TABLE 3.5.5

INPUT DATA FOR TACRAN3


```
TACRAN-    RUN 03          1-ST RUN OF VERSION 3

                  .-------------------------------------------.
                  :                                           :
                  : THESE FIRST FEW CARDS DEFINE THE BASIC DATASET :
                  :                                           :
                  '-------------------------------------------'
                  BASIC DATA SET                                          BASIC

                  .-------------------------------------------.
                  :                                           :
                  :   SOME SCALING CONSTANTS                  :
                  :                                           :
                  '-------------------------------------------'
                  DEFINITION OF -BITS-          BITS        1.0000
                  SCALING OF RAD/SEC            RAD/SEC     1.0000    SCALE
                  DEFINITION OF SQ-M            SQ-M        1.0000    SCALE
                  SCALE BIT/SEC                 BITS/SEC    1.0000    SCALE
                  SCALE FOR REVOLUTIONS/SEC     RPS         6.2832    SCALE
                  DEFINITION OF HOUR            HOUR        3600.0    SCALE
                  DEFINITION OF MINUTE          MIN         60.000    SCALE
                                                M/S         1.0000    SCALE
                  M/S**2                        M/S**2      1.0000    SCALE

                  .-------------------------------------------.
                  :                                           :
                  : THE BASIC DATASET                         :
                  :                                           :
                  '-------------------------------------------'
SET---BASIC DATA SET
                  REFER     EVENT LIST                                    REF SET
                  REFER     SUMMARY OUTPUT DATASET                        REFER
                  REFER     EVENT PROCESSING OVERLAYS                     REFER
                  REFER     INTERNAL OUTPUT INSTRUCTIONS BY EVENT         REFER
                  REFER     OVERLAYING INSTRUCTIONS BY A.S.C.             REFER
                  REFER     CURRENTLY ACTIVE A.S.C.                       REFER
                  REFER     LIST OF ALL ACTIVE RADARS                     REFER
                  REFER     LIST OF ALL A/C IN THE PROGRAM                REFER
                  REFER     LIST OF DP CARRYING SYSTEM FILES              REFER
                  REFER     LIST OF ALL COMMUNICATIONS LINKS              REFER
                  SYST. STATUS OUTPUT INTERVAL          30.000 SEC
                  L01                                   79.000 INT
                  L02                                   36.000 INT
                  L03                                   24.000 INT
                  L04                                   397.00 INT
                  L05                                   71.000 INT
                  L11                                   176.00 INT
                  L12                                   24.000 INT
                  L13                                   41.000 INT
                  L14                                   34.000 INT
                  ML01                                  44.000 INT
                  ML02                                  8.0000 INT
                  HPRIO                                 99.000 INT
                  WDELY                                 60.000 SEC
```

TABLE 3.5.5 (Cont.)


```
TACRAN-    RUN 03          1-ST RUN OF VERSION 3

                 ----------------------------------------
                 *                                      *
                 * MISCELLANEOUS ENTITIES ALWAYS ATTACHED TO BASIC DATASET *
                 *                                      *
                 ----------------------------------------
LIST--EVENT LIST
              REFER     STOPPER EVENT TO CONTROL SIMULATION DURATION          REG LIST
SET---SUMMARY OUTPUT DATASET                                                  REFER
SET---EVENT PROCESSING OVERLAYS                                               REG SET
              EVENT 1 CALLS OVERLAY                  2.0000 INT               REG SET
SET---INTERNAL OUTPUT INSTRUCTIONS BY EVENT
              DEBUG OUTPUT, EVENT 1..          NO                           REG SET
SET---OVERLAYING INSTRUCTIONS BY A.S.C.
              FOR A.S.C. NAMED                 SHAR                         REG SET
                   CALL OVERLAY...                1.0000 INT
              FOR A.S.C. NAMED                 NEWRET
                   CALL OVERLAY....               3.0000 INT
              FOR A.S.C. NAMED                 RCVCOMM
                   CALL OVERLAY....               3.0000 INT
              FOR A.S.C. NAMED                 COMM
                   CALL OVERLAY....               3.0000 INT
              FOR A.S.C. CALLED                RADOP
              CALL OVERLAY.....                   3.0000 INT
              FOR A.S.C. CALLED                STATUS
              CALL OVERLAY.....                   3.0000 INT
              FOR A.S.C CALLED                 LNKMNGT
              CALL OVERLAY.....                   3.0000 INT
              FOR A.S.C CALLED                 INITCOM
              CALL OVERLAY.....                   3.0000 INT
LIST--CURRENTLY ACTIVE A.S.C.
                                                                            REG LIST
```

TABLE 3.5.5 (Cont.)


```
TACPAN-    RUN 03          1-ST RUN OF VERSION 3

                  ----------------------------------------
                  *                                      *
                  * AND NOW FOR THE PROBLEM DEPENDENT DATASETS
                  *                                      *
                  ----------------------------------------
SET---CENTER OF THE LOCAL COORDINATE SYSTEM                                                    REG SET
              GEOGRAPHICAL COORDINATE CENTER        0.      KM ALT      22.000 DEG E LONG   50.000 DEG N LAT GEOGR
                  ----------------------------------------
                  *                                      *
                  *  INITIALIZATION EVENTS, ETC.         *
                  *                                      *
                  ----------------------------------------
SET---STOPPER EVENT TO CONTROL SIMULATION DURATION                                              REG SET
              EVENT TYPE 1 (INITIALIZATION AND STOP)    1.0000 INT
              TIME TO START SIMULATION                  .10000 SEC
              TIME TO STOP SIMULATION                   90.000 SEC
```

TABLE 3.5.5 (Cont.)

TACRAN- RUN 03 1-ST RUN OF VERSION 3

```
                    *RADAR DATA
LIST--LIST OF ALL ACTIVE RADARS                                                    REG LIST
                    REFER     RADAR NUMBER 1                                       REFER
                    REFER     RADAR NUMBER 2                                       REFER
                    REFER     RADAR NUMBER 3                                       REFER
                    REFER     RADAR NUMBER 4                                       REFER
SET---RADAR NUMBER 1                                                               REG SET
                    LABEL                                  1.0000 INT
                    LOCATION                             -26.000 KM EAST    20.000 KM NORTH   0.    KM VRTCL  LOCXYZ
                    INITIAL POINTING DIRECTION           -90.000 DEG
                    REFER     RADAR PARAMETERS                                     REFER
                    REFER     DATA PROCESSOR 1                                     REFER
                    FIRST F.A. TIME                        5.1000 SEC
                    POINTER TO NEXT A/C                  0.
SET---RADAR NUMBER 2                                                               REG SET
                    LABEL                                  2.0000 INT
                    LOCATION                              44.000 KM EAST    32.000 KM NORTH   0.    KM VRTCL  LOCXYZ
                    INIT PNT DIR                         -90.000
                    REFER     RADAR PARAMETERS                                     REFER
                    REFER     DATA PROCESSOR 2                                     REFER
                    FIRST F.A. TIME                        5.1000 SEC
                    POINT TO A/C                         0.
SET---RADAR NUMBER 3                                                               REG SET
                    LABEL                                  3.0000 INT
                    LOCATION                              114.00 KM EAST    40.000 KM NORTH   0.    KM VRTCL  LOCXYZ
                    INIT PNT DIR                         -90.000 DEG
                    REFER     RADAR PARAMETERS                                     REFER
                    REFER     DATA PROCESSOR 3                                     REFER
                    FIRST F.A. TIME                        5.1000 SEC
                    NEXT A/C                             0.
SET---RADAR NUMBER 4                                                               REG SET
                    LABEL                                  4.0000 INT
                    LOCATION                             -60.000 KM EAST   100.000 KM NORTH   0.    KM VRTCL  LOCXYZ
                    INIT PNT DIR                         -90.000 DEG
                    REFER     RADAR PARAMETERS                                     REFER
                    REFER     DATA PROCESSOR 4                                     REFER
                    FIRST F.A. TIME                        5.1000 SEC
                    NEXT A/C                             0.
SET---RADAR PARAMETERS                                                             REG SET
                    SCNTM                                  6.0000 SEC
                    SCAN                                   1.0472 RAD/SEC
                    RANGE ON 1 SQUARE METER CSSEC          40000. M
                    SIGNAL DETECTION THRESSHOLD           10.0000 DB
                    SN DEPENDENT ERRORS-RANGE              500.00 M
                                       AZIMUTH             .50000 DEG
                                       ELEVATION           .50000 DEG
                                       RDOT                0.
                    NON SN DEPENDENT   -RANGE              50.000 M
                                       AZIMUTH             5.00E-02 DEG
                                       ELEVATION           5.00E-02 DEG
```

TABLE 3.5.5 (Cont.)

TACPAN- RUN 03 1-ST RUN OF VERSION 3

	RDOT	0.
FAPAT		2.0000
AVERAGE FALSE ALARM DISTANCE		40.000 KM
RANGE RESOLUTION		1000.00 M
AZIMUTH RESOLUTION		1.0000 DEG
ELEVATION RESOLUTION		1.0000 DEG
FLMAX		40.000 DEG
AMAXN		78.400 M/S**2
AMAXT		19.600 M/S**2
DRDTT		500.00 M/S
DTD		.20000 SEC
DTN		.20000 SEC
DTW		.50000 SEC
MAXPT		5.0000 INT
MINTK		2.0000 INT
RMAX		80.000 KM
RWPTH		60.000 KM
SCNMX		3.0000
SIGMA		600.00 M
SPMAX		1286.0 M/S
SWPTH		20.000
TDEAD		13.200 SEC
TIMAX		7.8000 SEC
TIMIN		4.2000 SEC
TLASS		1500.0 M
TMCK1		5.0000
TMCK2		.70000
TMIN		1.0000 SEC
TNASS		3.0000 KM
TPD		13.800 SEC
TPL		7.8000 SEC
TPN		40.000 SEC
TPTMX		14.000 SEC

TABLE 3.5.5 (Cont.)


```
TACRAN-    RUN 03          1-ST RUN OF VERSION 3

                  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                  •                                                                               •
                  •DATA PROCESSOR DESCRIPTIONS FOLLOW HERE                                        •
                  •                                                                               •
                  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
SET---DATA PROCESSOR 1                                                                            REG SET
                  LABEL                                    1.0000 INT
                  LOCATION                               -26.000 KM EAST      20.000 KM NORTH   0.      KM VRTCL  LOCXYZ
                  TYPE, 4 LHV                                      5.0000                                          ZEROS
                  REFER     COMM LIST FOR DP 1                                                                    REFER
                  LIST OF DATALINKS                    0.
                  REFER     RADAR NUMBER 1                                                                        REFER
                  MESG TEXT DSP                        0.
SET---DATA PROCESSOR 2                                                                            REG SET
                  LABEL                                    2.0000 INT
                  LOCATION                                44.000 KM EAST      32.000 KM NORTH   0.      KM VRTCL  LOCXYZ
                  TYPE, 4 LHV                                      5.0000                                          ZEROS
                  REFER     COMM LIST FOR DP 2                                                                    REFER
                  LIST OF DATA LINKS                   0.
                  REFER     RADAR NUMBER 2                                                                        REFER
                  MSG TEXT DSG                         0.
SET---DATA PROCESSOR 3                                                                            REG SET
                  LABEL                                    3.0000 INT
                  LOCATION                               114.00 KM EAST      40.000 KM NORTH   0.      KM VRTCL  LOCXYZ
                  TYPE, 4 LHV                                      5.0000                                          ZEROS
                  REFER     COMM LIST FOR DP 3                                                                    REFER
                  LIST OF DATA LINKS                   0.
                  REFER     RADAR NUMBER 3                                                                        REFER
                  MSG TEXT DSP                         0.
SET---DATA PROCESSOR 4                                                                            REG SET
                  LABEL                                    4.0000 INT
                  LOCATION                               -60.000 KM EAST     100.000 KM NORTH   0.      KM VRTCL  LOCXYZ
                  TYPE, 4 LHV                                      5.0000                                          ZEROS
                  REFER     COMM LIST FOR DP 4                                                                    REFER
                  LIST OF DATA LINKS                   0.
                  REFER     RADAR NUMBER 4                                                                        REFER
                  MSG TEXT DSP                         0.

                  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                  •                                                                               •
                  • FUNCTIONAL LISTS OF DP-S                                                      •
                  •                                                                               •
                  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
LIST--LIST OF DP CARRYING SYSTEM FILES                                                            REG LIST
                  INSERT    LIST OF DP CARRYING LOCAL FILES                                       INSERT
LIST--LIST OF DP CARRYING LOCAL FILES                                                             REG LIST
                  REFER     DATA PROCESSOR 1                                                      REFER
                  REFER     DATA PROCESSOR 2                                                      REFER
                  REFER     DATA PROCESSOR 3                                                      REFER
                  REFER     DATA PROCESSOR 4                                                      REFER
```

TABLE 3.5.5 (Cont.)


```
TACPAN-    RUN 03          1-ST RUN OF VERSION 3

                  ----------------------------------------
                  *                                      *
                  * AND ASSOCIATED COMMUNICATIONS LISTS   *
                  *                                      *
                  ----------------------------------------
LIST--COMM LIST FOR DP 1
                  REFER     DATA PROCESSOR 2
                  REFER     DATA PROCESSOR 4
LIST--COMM LIST FOR DP 2
                  REFER     DATA PROCESSOR 1
                  REFER     DATA PROCESSOR 3
                  REFER     DATA PROCESSOR 4
LIST--COMM LIST FOR DP 3
                  REFER     DATA PROCESSOR 2
LIST--COMM LIST FOR DP 4
                  REFER     DATA PROCESSOR 1
                  REFER     DATA PROCESSOR 4

                  ----------------------------------------
                  *                                      *
                  * AND COMMUNICATION DATA LINKS          *
                  *                                      *
                  ----------------------------------------
LIST--LIST OF ALL COMMUNICATIONS LINKS
                  REFER     LINK 1,2
                  REFER     LINK 1,4
                  REFER     LINK 2,3
                  REFER     LINK 2,4
SET---LINK 1,2
                  REFER     DATA PROCESSOR 1
                  REFER     DATA PROCESSOR 2
                  BAUD                              10000.0 BITS/SEC
SET---LINK 1,4
                  REFER     DATA PROCESSOR 1
                  REFER     DATA PROCESSOR 4
                  BAUD                              10000.0 BITS/SEC
SET---LINK 2,3
                  REFER     DATA PROCESSOR 2
                  REFER     DATA PROCESSOR 3
                  BAUD                              10000.0 BITS/SEC
SET---LINK 2,4
                  REFER     DATA PROCESSOR 2
                  REFER     DATA PROCESSOR 4
                  BAUD                              10000.0 BITS/SEC
```

TABLE 3.5.5 (Cont.)


```
TACRAN-    RUN 03          1-ST RUN OF VERSION 3

                  .-----------------------------------.
                  :                                   :
                  :   AIR CRAFT DATA                  :
                  :                                   :
                  '-----------------------------------'

LIST--LIST OF ALL A/C IN THE PROGRAM                                                    REG LIST
                  REFER     A/C NUMBER 1                                                REFER
                  REFER     A/C NUMBER 4                                                REFER
                  REFER     A/C NUMBER 3                                                REFER
SET---A/C NUMBER 1                                                                      REG SET
                                                    1.0000 INT                          REFER
                  REFER     LOCATION MODEL - A/C 1
                  RADAR CROSS SECTION             10.0000 SQ-M
                  INDEX OF AIM POINT               1.0000 INT
                  TIME                           1.00E-02
                  INITIAL X Y Z                    18.000 KM EAST     50.000 KM NORTH     8.5000 KM VRTCL  LOCXYZ
                  X-DOT                            555.60
                                                          8.0000                        ZEROS
                  TIME OF CREATION FOR THIS AC   1.00E-02
SET---A/C NUMBER 2                                                                      REG SET
                                                    2.0000 INT                          REFER
                  REFER     LOCATION MODEL - A/C 1
                  RADAR CROSS SECTION             10.0000 SQ-M
                  INDEX OF AIM POINT               1.0000 INT
                  TIME                            10.0000
                  INITIAL X Y Z                    20.000 KM EAST     50.000 KM NORTH     8.5000 KM VRTCL  LOCXYZ
                  X-DOT                            555.60
                                                          8.0000                        ZEROS
                  TIME OF CREATION FOR THIS AC    10.0000
SET---A/C NUMBER 3                                                                      REG SET
                                                    3.0000 INT                          REFER
                  REFER     LOCATION MODEL - A/C 2
                  RADAR CROSS SECTION             10.0000 SQ-M
                  INDEX OF AIM POINT               1.0000 INT
                  TIME                           1.00E-02
                  INITIAL X Y Z                    40.000 KM EAST     72.000 KM NORTH     8.5000 KM VRTCL  LOCXYZ
                  XDOT                            0.
                  Y-DOT                           -555.60
                  Z-DOT                           0.
                  THETA                           0.
                  PSI                             0.
                  REMAINDER OF STATE
                                                          4.0000                        ZEROS
                  TIME OF CREATION FOR THIS AC   1.00E-02
SET---A/C NUMBER 4                                                                      REG SET
                                                    4.0000 INT                          REFER
                  REFER     LOCATION MODEL - A/C 3
                  RADAR CROSS SECTION             10.0000 SQ-M
                  INDEX OF AIM POINT               1.0000 INT
                  TIME                              .50000
                  INITIAL X Y Z                    43.600 KM EAST     52.500 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  X-DOT                           0.
                  Y-DOT                           -555.60
                  Z-DOT                           0.
                  THETA                           0.
```

TABLE 3.5.5 (Concl.)


```
TACRAN-    RUN 03          1-ST RUN OF VERSION 3

                  PSI                                    -1.5700
                  THETA-DOT                              0.
                  PSI-DOT                                9.00E-02
                  VELOCITY-DOT                           0.
                  NUMBER G S                             0.
                  TIME OF CREATION FOR THIS AC             .50000 SEC
SET---LOCATION MODEL - A/C 1                                                    REG SET
                  MODEL TYPE                             PTMASS
                  DELTA T                                  .10000
                  AIM POINT 1                            40.000 KM EAST    50.000 KM NORTH    8.5000 KM VRTCL  LOCXYZ
                  AP V 1                                 555.60
                  AIM POINT 2                            60.000 KM EAST    40.000 KM NORTH    8.5000 KM VRTCL  LOCXYZ
                  AP V 2                                 555.60
                  AIM POINT 3                            50.000 KM EAST    20.000 KM NORTH    8.5000 KM VRTCL  LOCXYZ
                  AP V 3                                 555.60
SET---LOCATION MODEL - A/C 2                                                    REG SET
                  MODEL TYPE                             PTMASS
                  DELTA T                                  .20000
                  AIM POINT 1                            40.000 KM EAST    50.000 KM NORTH    8.5000 KM VRTCL  LOCXYZ
                  AP V 1                                 555.60
                  AIM POINT 2                            50.000 KM EAST    40.000 KM NORTH    6.0000 KM VRTCL  LOCXYZ
                  AP V 1                                 200.60
                  AIM POINT 3                            50.000 KM EAST    20.000 KM NORTH    6.0000 KM VRTCL  LOCXYZ
                  VELOCITY 3                             400.00
SET---LOCATION MODEL - A/C 3                                                    REG SET
                  MODEL TYPE                             PTMASS
                  DELTA T                                  .10000
                  AIM POINT 1                            45.500 KM EAST    48.000 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  2                           50.000 KM EAST    46.100 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  3                           54.500 KM EAST    48.000 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  4                           56.400 KM EAST    52.500 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  5                           54.500 KM EAST    57.000 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  6                           50.000 KM EAST    58.900 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  7                           45.500 KM EAST    57.000 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  8                           43.600 KM EAST    52.500 KM NORTH    10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  AIM POINT  9                           60.600 KM EAST    0.       KM NORTH  10.0000 KM VRTCL  LOCXYZ
                  VELOCITY                               555.60
                  EOF
```

TABLE 3.5.6

POST PROCESSOR OUTPUT FROM TACRAN3

FOR TITLER RUN 03 1-ST RUN OF VERSION 3 1 of 4

TACRAN POST PROCESSOR

	NODE	A/C	TIME SEC	ME LABEL	NO.OF ASSNS	MEASUREMENTS X M	Y M	Z M	TRUE--- X	-(M)--- Y	Z	TRUE- XDOT	(M/S) YDOT	ZDOT	RANGE M	AZIM DEG	ELEV DEG	S/N DB	RCS DBSM
1	1	0	5.100	16	0	26402	-52125	8812	0	0	0	0	0	0	89586	306	6	10.8	14.8
2	1	0	6.415	28	0	-80	32016	1871	0	0	0	0	0	0	28631	25	4	10.8	-5.0
3	1	0	19.412	100	0	-21801	65516	9299	0	0	0	0	0	0	66645	85	11	10.8	3.5
4	1	0	24.379	186	0	10116	35122	11104	0	0	0	0	0	0	40698	23	16	10.8	1.1
5	1	0	25.203	188	0	-19603	39867	6150	0	0	0	0	0	0	21759	72	16	10.8	-9.8
6	1	0	28.733	225	0	-13691	-21352	4725	0	0	0	0	0	0	42879	284	6	10.8	2.0
7	1	0	30.178	255	0	26533	29933	12472	0	0	0	0	0	0	54899	11	13	10.8	6.3
8	1	0	31.977	283	0	-90841	138851	16859	0	0	0	0	0	0	136433	119	7	10.8	22.1
9	1	0	35.127	315	0	-5847	-6140	4704	0	0	0	0	0	0	33340	308	8	10.8	-2.4
10	1	0	36.261	343	0	-20317	21593	1247	0	0	0	0	0	0	6032	16	12	10.8	-32.1
11	1	0	41.504	459	0	492	4640	7378	0	0	0	0	0	0	31402	330	14	10.8	-3.4
12	1	0	43.215	491	0	-4470	89919	6739	0	0	0	0	0	0	73468	73	5	10.8	11.4
13	1	0	46.188	517	0	-36397	-10661	4512	0	0	0	0	0	0	32689	251	8	10.8	-2.7
14	1	0	52.612	622	0	-24329	5760	3868	0	0	0	0	0	0	14850	277	15	10.8	-16.4
15	1	0	55.538	688	0	-27394	54657	621	0	0	0	0	0	0	34691	92	1	10.8	-1.7
16	1	0	55.890	690	0	-59669	97814	19556	0	0	0	0	0	0	87012	113	13	10.8	14.3
17	1	0	57.515	705	0	-89373	-17939	10415	0	0	0	0	0	0	74592	211	8	10.8	11.6
18	1	0	60.855	771	0	-15026	33694	6310	0	0	0	0	0	0	18648	51	20	10.8	-12.5
19	1	0	62.409	779	0	-39656	29735	5151	0	0	0	0	0	0	17544	145	17	10.8	-13.5
20	1	0	64.003	835	0	-34113	5842	2427	0	0	0	0	0	0	16497	240	8	10.8	-14.6
21	1	0	71.094	1006	0	-24172	17452	282	0	0	0	0	0	0	3149	306	5	10.8	-43.4
22	1	0	71.978	1008	0	42149	18439	11857	0	0	0	0	0	0	69191	354	10	10.8	10.3
23	1	0	76.319	1061	0	-30649	-4260	5745	0	0	0	0	0	0	25361	259	13	10.8	-7.1
24	1	0	79.010	1096	0	-17911	34354	4731	0	0	0	0	0	0	17142	61	16	10.8	-13.9
25	1	0	80.113	1102	0	-26463	20619	278	0	0	0	0	0	0	821	127	20	10.8	-66.7
26	1	0	81.372	1117	0	-236529	-66450	48426	0	0	0	0	0	0	232683	202	12	10.8	31.4
27	1	0	87.046	1226	0	-55710	18563	8071	0	0	0	0	0	0	30821	183	15	10.8	-3.7
28	1	0	87.948	1237	0	-52403	-20455	8235	0	0	0	0	0	0	49006	237	10	10.8	4.3
29	1	1	9.522	40	0	23099	49971	8554	23285	50000	8500	556	0	-0	58156	31	8	13.4	10.0
30	1	1	15.495	71	0	26767	50004	8373	26603	50000	8500	556	0	-0	61275	30	8	12.6	10.0
31	1	1	21.470	106	0	29832	49949	8535	29923	50000	8500	556	0	-0	63929	28	8	11.8	10.0
32	1	1	27.448	194	1	33395	49660	8536	33244	50000	8500	556	0	-0	66935	27	7	11.1	10.0
33	1	1	33.427	265	1	36566	50051	8252	36566	50000	8500	556	0	-0	69898	26	7	10.3	10.0
34	1	1	39.408	372	1	39838	50126	8349	39892	50092	8500	555	-15	-0	72883	25	7	9.6	10.0
35	1	1	45.387	493	1	43398	49530	8218	43200	49752	8500	537	-141	-0	75866	23	6	8.9	10.0
36	1	1	51.361	581	1	46299	48939	8273	46358	48731	8500	507	-226	-0	78314	22	6	8.3	10.0
37	1	1	57.331	692	0	49416	46736	8570	49334	47265	8500	481	-279	-0	80473	20	6	7.8	10.0
38	1	3	39.443	374	1	42373	54238	8178	42646	54356	8500	-285	-477	-0	76903	27	6	8.6	10.0
39	1	3	45.420	495	1	40910	51068	8270	40927	51515	8500	-290	-474	-0	74234	25	6	9.2	10.0
40	1	3	51.395	583	1	39501	48539	8873	39066	48671	8526	-98	-517	3	72037	24	7	9.8	10.0
41	1	3	57.358	694	1	39417	45288	8342	39098	45583	8509	224	-421	-32	70629	21	7	10.1	10.0
42	1	3	63.322	775	1	41088	44058	8196	40781	43394	8244	349	-256	-66	71741	20	7	10.0	10.0
43	1	3	69.296	913	1	42752	42108	7973	42937	42076	7808	363	-150	-84	72658	18	6	9.6	10.0
44	1	3	75.278	1032	1	45130	41444	7323	45087	41289	7291	342	-101	-87	74652	17	6	9.2	10.0
45	1	3	81.264	1100	1	47167	40652	6984	47090	40732	6773	313	-80	-83	76346	16	5	8.8	10.0
46	1	3	81.264	1100	0	47167	40652	6984	47090	40732	6773	313	-80	-83	76346	16	5	8.8	10.0
47	1	3	87.252	1216	0	48944	39925	6473	48917	40270	6288	284	-71	-75	77817	15	5	8.4	10.0
48	1	4	9.357	36	0	45556	48083	10132	45284	47903	9983	344	-437	2	77534	21	8	8.5	10.0

TABLE 3.5.6 (Cont.)

FOR TITLER RUN 03 1-ST RUN OF VERSION 3

TACRAN POST PROCESSOR

TRACK INITIATION

NODE	A/C		TIME SEC	TYPE	ME LABEL	TL LABEL	FIRST ME LABEL	SPEED M/S	TS LABEL	DISTANCE M
1	1	0	5.100	SAVE	16	0	0	0	0	0
2	1	0	6.415	SAVE	28	0	0	0	0	0
3	1	0	19.412	SAVE	100	0	0	0	0	0
4	1	0	24.379	SAVE	186	0	0	0	0	0
5	1	0	25.203	SAVE	188	0	0	0	0	0
6	1	0	28.733	SAVE	225	0	0	0	0	0
7	1	0	30.178	SAVE	255	0	0	0	0	0
8	1	0	31.977	SAVE	263	0	0	0	0	0
9	1	0	35.127	SAVE	315	0	0	0	0	0
10	1	0	36.261	SAVE	343	0	0	0	0	0
11	1	0	41.504	SAVE	459	0	0	0	0	0
12	1	0	43.215	SAVE	491	0	0	0	0	0
13	1	0	46.188	SAVE	517	0	0	0	0	0
14	1	0	52.612	SAVE	622	0	0	0	0	0
15	1	0	55.538	SAVE	688	0	0	0	0	0
16	1	0	55.890	SAVE	690	0	0	0	0	0
17	1	0	57.515	SAVE	705	0	0	0	0	0
18	1	0	60.855	SAVE	771	0	0	0	0	0
19	1	0	62.409	SAVE	779	0	0	0	0	0
20	1	0	64.003	SAVE	835	0	0	0	0	0
21	1	0	71.094	SAVE	1006	0	0	0	0	0
22	1	0	71.978	SAVE	1008	0	0	0	0	0
23	1	0	76.319	SAVE	1061	0	0	0	0	0
24	1	0	79.010	SAVE	1096	0	0	0	0	0
25	1	0	80.113	SAVE	1102	0	0	0	0	0
26	1	0	81.372	SAVE	1117	0	0	0	0	0
27	1	0	87.046	SAVE	1226	0	0	0	0	0
28	1	0	87.948	SAVE	1237	0	0	0	0	0
29	1	1	9.522	SAVE	40	0	0	0	0	0
30	1	1	15.495	LOCAL	71	82	40	615	0	0
31	1	1	15.495	SAVE	71	0	0	0	0	0
32	1	1	21.470	DLT	106	82	0	0	116	701
33	1	1	57.331	SAVE	692	0	0	0	0	0
34	1	3	81.264	SAVE	1100	0	0	0	0	0
35	1	3	87.252	LOCAL	1216	1230	1100	332	0	0
36	1	3	87.252	SAVE	1216	0	0	0	0	0
37	1	4	9.357	SAVE	36	0	0	0	0	0
38	1	4	15.321	LOCAL	69	79	36	493	0	0
39	1	4	15.321	SAVE	69	0	0	0	0	0
40	2	0	5.100	SAVE	18	0	0	0	0	0
41	2	0	5.494	SAVE	24	0	0	0	0	0
42	2	0	6.370	SAVE	30	0	0	0	0	0
43	2	0	7.057	SAVE	34	0	0	0	0	0
44	2	0	8.405	SAVE	46	0	0	0	0	0
45	2	0	8.941	SAVE	48	0	0	0	0	0
46	2	0	12.210	SAVE	62	0	0	0	0	0

TABLE 3.5.6 (Cont.)

FOR TITLER RUN 03 1-ST RUN OF VERSION 3

TACRAN POST PROCESSOR

DISTRIBUTED LOCAL TRACKS

A/C	TIME SEC	NODE	TRACK LABEL	X M	Y M	Z M	XDOT M/S	YDOT M/S	ZDOT M/S	XACC M/S2	YACC M/S2	ZACC M/S2	PREDICTED X	POSITION Y	Z	SYSTEM-TRACK X	Y	Z	PREDICTED PTS	NBR TKR
1	21.470	1	116	29832	49949	8535	462	-16	56	-17	-2	10	30437	50036	8192	29832	49949	8535	3	1
1	23.103	2	116	30752	49936	8550	488	-15	46	-11	-2	7	30564	49919	8638	30564	49919	8638	4	2
1	27.448	1	116	33401	49665	8533	627	-75	-8	12	-8	-4	32873	49869	8750	32292	49808	9039	4	2
1	29.980	2	116	34173	49849	8522	591	-4	-12	6	1	-3	34363	49549	8521	32825	49757	9224	5	2
1	33.427	1	116	36552	50068	8260	520	76	-74	-7	11	-9	36801	49829	8470	34147	49579	9887	5	2
1	34.838	2	116	37341	50035	8406	533	53	-15	-4	7	-1	37286	50175	8156	37341	50035	8406	5	2
1	35.875	3	116	37947	49779	8602	589	-183	142	11	-46	34	37891	50094	8390	37891	50094	8390	5	2
1	39.408	1	116	39839	50083	8402	538	53	-16	-3	9	-1	39737	50350	8328	39737	50350	8328	5	3
1	41.569	3	116	41264	50371	8269	575	175	-123	5	32	-27	41163	50214	9363	40996	50581	8340	5	2
1	45.387	1	116	43399	49595	8218	623	-222	-23	10	-39	2	43285	50985	7837	42758	50985	8303	5	2
1	47.872	3	116	44726	49280	8703	574	-245	149	1	-34	24	44947	49042	8162	43975	51323	8143	5	2
1	51.361	1	116	46311	48903	8380	443	-155	-35	-17	-7	-7	46728	48425	9222	46311	48903	8380	5	2
1	52.337	2	116	46792	48651	8406	431	-138	-50	-18	3	-12	46735	48749	8342	46735	48749	8342	5	2
1	53.883	3	116	47600	48145	8471	497	-330	-48	1	-39	-15	47458	48438	8328	47374	48489	8368	5	2
1	58.152	2	116	49777	47098	8558	522	-293	63	5	-13	16	49722	46738	9268	48926	47685	7473	5	2
1	59.902	3	116	50434	46517	8236	397	-292	-202	-21	-4	-46	50690	46585	8668	49473	47320	7814	5	2
1	63.960	2	116	52418	45294	8475	467	-324	30	-2	-8	7	52044	45334	7415	50541	46388	7360	5	2
1	65.926	3	116	53508	44556	8501	517	-361	49	5	-10	10	53336	44656	8534	53508	44556	8501	5	2
1	69.766	2	116	55186	43528	8563	453	-271	34	-4	6	4	55483	43169	8687	55530	43094	8758	5	2
1	71.954	3	116	55967	42821	8590	362	-299	0	-17	1	-5	56178	42934	8637	56715	42193	8969	5	2
1	75.583	2	116	57714	41634	8550	450	-323	-9	1	-2	-2	57283	41736	8592	58733	40592	9420	5	2
1	77.985	3	116	58837	40777	8698	490	-369	26	9	-9	1	58795	40859	8529	60105	39458	9789	5	2
1	81.420	2	116	60343	39639	8515	452	-339	-43	0	-1	-7	60519	39510	8787	60343	39639	8515	5	2
1	84.025	3	116	61359	38693	8413	388	-361	-55	-10	-3	-7	61521	38757	8402	61521	38755	8376	5	2
1	87.259	2	116	62178	37083	8482	245	-499	-4	-23	-20	1	62613	37527	8237	62985	37649	8141	5	2
3	22.431	2	133	46321	63091	8501	-90	-534	16	-35	-3	3	47572	63181	8396	46321	63091	8501	3	1
3	23.696	3	133	46269	62270	8559	-108	-579	30	-32	-7	4	46180	62413	8523	46180	62413	8523	4	2
3	28.453	2	133	45406	59730	8444	-253	-540	-31	-31	1	-5	45757	59517	8702	45148	59329	8447	4	2
3	29.741	3	133	45090	59112	8713	-283	-518	46	-30	3	4	45080	59034	8404	44733	58919	8493	5	2
3	34.501	2	133	43935	56791	8583	-284	-465	-30	-13	8	-6	43743	56645	8933	42696	56461	8982	5	2
3	35.735	3	133	43637	56037	8536	-265	-517	-39	-7	-1	-6	43570	56195	8545	42011	55734	8969	5	2
3	39.443	1	133	42385	54220	8151	-364	-504	-144	-17	0	-19	39743	53641	9189	39743	53641	9189	5	3
3	41.827	3	133	42089	53177	8603	-230	-453	83	4	7	18	41516	53019	7808	42089	53177	8603	5	2
3	45.420	1	133	40975	51111	8353	-281	-564	-24	-2	-10	-2	41263	51551	8900	41288	51597	9016	5	2
3	47.867	3	133	40119	50368	8544	-334	-414	38	-8	9	6	40288	49731	8293	40771	50574	9428	5	2
3	51.395	1	133	39400	48706	8829	-251	-429	71	2	6	5	38941	48908	8680	40066	49174	10212	5	2
3	53.907	3	133	38834	47483	8534	-171	-454	-3	18	2	1	38769	47628	9008	39593	48232	10906	5	2
3	57.358	1	133	39317	45263	8260	157	-684	-138	49	-32	-22	38245	45916	8524	39317	45263	8260	5	2
3	59.942	3	133	39829	44943	8432	286	-283	-4	52	31	3	39723	43495	7903	39888	43389	7829	5	2
3	63.322	1	133	41124	44140	8289	493	-139	10	58	41	8	40794	43985	8420	41124	44140	8289	5	2
3	65.964	3	133	41675	43109	8217	323	-251	-47	13	15	-6	42426	43774	8314	42628	43915	8342	5	2
3	69.296	1	133	42746	41990	7977	337	-394	-82	9	-18	-8	42751	42273	8061	42746	41990	7977	5	2
3	70.559	2	133	43294	41813	7737	355	-230	-116	6	17	-9	43179	41479	7867	43179	41479	7867	5	2
3	71.977	3	133	44068	42033	7596	551	82	-158	47	74	-19	43797	41487	7573	43681	40871	7728	5	2
3	74.351	2	133	45488	41295	7223	343	-94	-86	-7	11	3	46476	42394	6907	45340	38775	7198	5	2
3	77.986	3	133	45949	40971	7178	235	-245	-32	-30	-28	13	46049	41141	7082	45949	40971	7178	5	2
3	82.147	2	133	47357	40630	6721	314	-107	-113	-4	2	-5	46927	39950	7045	46668	39709	7154	5	2
3	83.993	3	133	48041	40531	6202	351	-34	-201	3	16	-13	47937	40432	6513	46820	38996	7212	5	2
3	87.969	2	133	49133	40234	6184	271	-59	-20	-8	5	14	49437	40397	5404	49133	40234	6184	5	2

TABLE 3.5.6 (Concl.)

FOR TITLER RUN 03 1-ST RUN OF VERSION 3

TACRAN POST PROCESSOR

A/C	TIME SEC	NODE	TRACK LABEL	X M	Y M	Z M	XDOT M/S	YDOT M/S	ZDOT M/S	XACC M/S2	YACC M/S2	ZACC M/S2	PREDICTED POSITION X	Y	Z	SYSTEM-TRACK-- X	Y	Z	PREDIC PTS	NBR TYR
4	29.863	3	241	54499	48381	10567	277	413	54	-26	26	1	55415	47462	10515	54499	48381	10567	3	1
4	33.997	2	241	54916	50806	11031	-3	689	138	-50	50	12	55425	50309	10804	55425	50309	10804	3	2
4	35.812	3	241	54710	51679	11245	-134	625	141	-55	30	10	54911	52056	11280	55693	51294	10918	4	2
4	40.097	2	241	53809	53845	11576	-324	493	72	-50	-8	-5	54137	54357	11950	55988	53958	11199	4	2
4	41.775	3	241	53104	54726	11618	-434	499	42	-53	-5	-8	53266	54671	11697	55974	55130	11317	5	2
4	46.222	2	241	51111	55809	12030	-545	196	85	-38	-37	1	51174	56946	11905	51111	55809	12030	5	2
4	47.765	3	241	50479	56015	12158	-532	84	92	-30	-48	3	50270	56111	12160	50225	56067	12161	5	2
4	52.347	2	241	47882	55775	12630	-608	-173	113	-21	-53	4	48043	56399	12582	47057	56316	12563	5	2
4	53.784	3	241	47006	55560	12728	-647	-208	89	-24	-46	-1	47007	55527	12792	45898	56235	12693	5	2
4	58.447	2	241	45179	53663	12923	-376	-541	29	20	-60	-7	43989	54588	13145	45179	53663	12923	5	2
4	59.822	3	241	44764	52954	12893	-286	-591	-6	33	-55	-11	44663	52919	12963	44682	52862	12956	5	2
4	63.399	1	241	44096	51186	12997	-120	-565	16	39	-26	-3	43567	50249	12976	43567	50249	12976	5	2
4	64.497	2	241	44018	50518	13012	-72	-582	15	39	-21	-2	43964	50565	13015	43277	49293	12963	5	2
4	65.867	3	241	44135	49590	12872	101	-684	-73	67	-38	-20	42949	47997	12934	42949	47997	12934	5	3
4	69.360	1	241	44482	47650	12929	200	-601	-13	47	-8	-4	44489	47202	12619	44482	47650	12929	5	3
4	71.907	3	241	45258	46909	12740	351	-249	-52	51	61	-6	44992	46119	12895	45146	46093	12883	5	2
4	75.323	1	241	46748	45670	12257	527	-265	-162	52	32	-19	46456	46057	12562	46519	43924	12747	5	2
4	77.928	3	241	47989	44881	11880	580	-251	-197	43	22	-19	48121	44981	11834	47940	42206	12485	5	2
4	81.301	1	241	49972	44116	11648	639	-212	-89	30	15	4	49945	44032	11217	49972	44116	11648	5	2
4	82.069	2	241	50285	44464	11419	598	16	-130	21	52	-1	50472	43957	11581	50472	43957	11581	5	2
4	83.925	3	241	50963	44811	11091	416	250	-183	-19	81	-12	51396	44493	11178	51752	43611	11429	5	2
4	87.960	2	241	52864	46068	10451	415	469	-180	-14	68	-6	52643	45821	10354	52864	46068	10451	5	2

CREATED ITEM NUMBER 4 8 BY 285

3.6 SIMULATION SYSTEM DETAILS

The TACRAN simulation was constructed using a number of existing simulation facilities and tools that have been developed over the past ten or so years at GRC. This section briefly describes some of these systems.

3.6.1 IFTRAN*

The TACRAN simulation system is written in a GRC-developed structured language extension of FORTRAN called IFTRAN. This extension permits GO-TO-less programming, which is believed to increase programmer productivity by reducing errors and testing time.

The IFTRAN preprocessor translates the IFTRAN specific statements into standard FORTRAN while passing all other statements unchanged. The result is then compiled by the FORTRAN compiler. In addition to the translation, the preprocessor checks the control structure for proper use of IFTRAN control structures and issues error messages if violations occur.

The preprocessor provides the following additional features to improve code production:

1. Indented listing of the IFTRAN source code, and optional indentation of the FORTRAN card images on the output file
2. Editing functions which include character string replacement, insertion of saved blocks of source text, limited macro capability, and in-line comments.
3. Input/output controls which include selection of input and output files, selective page ejection, and selective suppression of the source listings.

---

*IFTRAN: Structured Programming Preprocessors for FORTRAN, IFTRAN-3 User's Guide, General Research Corporation, January 1978.

The subset of IFTRAN used in the TACRAN simulation consists of the following statements:

Loop. The LOOP statement permits indefinite looping; the EXIT statements are used to escape the loop. The predicate is any FORTRAN logical statement. The dots represent any sequence of FORTRAN and IFTRAN statements.

```
LOOP
   :
EXIT IF (predicate)      [optional]
   :
EXIT                     [optional]
   :
END LOOP
```

Conditional. The IF statement permits code sequences to be executed if a predicate is TRUE. There may be any number of (optional) ORIF statements. The (optional) ELSE statement will be executed if none of the preceding predicates are TRUE.

```
IF (predicate)
   :
ORIF (predicate)         [optional]
   :
ORIF (predicate)         [optional]
   :
ELSE                     [optional]
   :
END IF
```

Case. The CASE statement permits one of a set of "cases" (code sequences) to be executed based on the value of an integer expression. For example, when the integer expression = m, the code following the CASE (m) statement will be executed. The (optional) code between the CASE OF and the first CASE statements is executed for every case.

```
CASE OF (integer expression)
   :
CASE (i)
   :
CASE (j, k, l)
   :
CASE (m)
   :
CASE ELSE                          [optional]
   :
END CASE
```

Sequential. The INVOKE statement permits a sequence of code between a BLOCK statement and an END BLOCK to be executed at the point in the code where the INVOKE statement appears. There may be any number of INVOKEs of the same BLOCK of code. This statement acts like a subroutine, except that the BLOCK of code appears (anywhere) in the same routine as the INVOKE statements. Therefore no arguments are required. The block name is a character string, which permits descriptive phrases to be used.

```
INVOKE (block name)
   :
INVOKE (block name)
   :
BLOCK (block name)
   :
END BLOCK
```

The complete GRC IFTRAN language permits a number of additional statements, including DO...END DO, FOR...END FOR, REPEAT...UNTIL, and WHILE...END WHILE. All IFTRAN sequences may be nested.

3.6.2 Dynamic Storage Allocation (DSA)*

The GRC-developed DSA (Dynamic Storage Allocation) system provides a system of utility routines for data management, so that the programmer who uses the system sees the machine as having an infinite "virtual memory" for data storage. Rather than the usual organization of data into large multiple-dimensioned arrays, which are accessed by means of indexing and searched by means of the FORTRAN DO-loop, data in the DSA mode of operation is organized into individual dataset instances, which are relatively short collections of sequential words (the maximum length tends to be a few tens of words), and which are organized into lists. System subroutines are then provided to enable the programmer to search a given list, to access a dataset whose identity is known, and in general to perform all the operations on datasets that can be performed on the more standard dimensioned arrays.

Two new kinds of data words have been defined: the List Header Variable, and the Data Set Pointer, which serve the functions, respectively,

---

*R.S. Stone, A Dynamic Storage Allocation System for FORTRAN Programs, General Research Corporation IMR-1249, January 1970.

of identifying a list of datasets, thus enabling the program to access its members, and of identifying an individual dataset, enabling the program to access that dataset. This capability allows the model designer to put together extremely complex data structures without making use of large arrays.

Figure 3.6.1 shows the structure of linked lists in the DSA system. A list is available to the user through the List Header Variable (LHV). DSA data management routines are used to access sequentially, either forward or backward, the datasets on the list. Data sets are accessed through the Data Set Pointer (DSP) word, which remains in a fixed storage location as long as the dataset exists. The actual dataset may not remain at the same set of addresses during a run, but may be moved around as space requires, or may even be put out on secondary storage when not being accessed. The actual location of the dataset at any given time is kept in the DSP word, and the address of the first word of a dataset (called the dataset index) can be obtained by using a data management routine.

3.6.3 PRINTOUT

PRINTOUT is used with the DSA system to provide a method of easily defining a large dataset and file structure. It accepts dataset template information punched on cards and produces three items as output:

1. A programmer's notebook, containing the definition of all datasets sorted by dataset type, and including the FORTRAN variable name associated with each field. A cross-reference listing permits the user to locate the dataset in which a particular variable has been defined. Table 3.5.4 in Sec. 3.5.4 (p. 355) is the PRINTOUT output defining the datasets for TACRAN3.

2. A set of "equivalence blocks" for input into a software maintenance system (such as Control Data's UPDATE package) which in turn allows the programmer to access dataset variables with the names provided in the dataset definition input deck

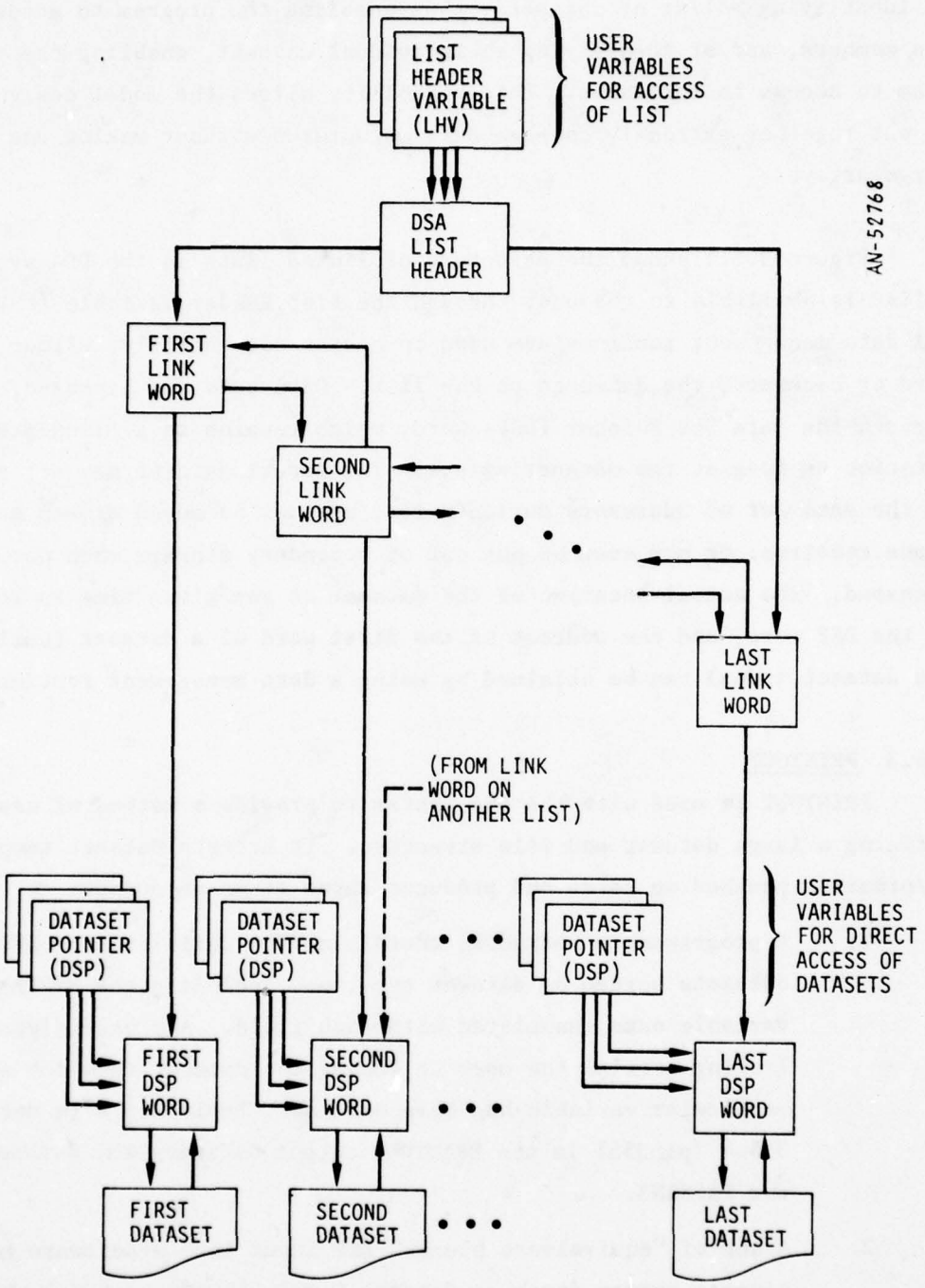


Figure 3.6.1. List Internal Linkages

to PRINTOUT. The equivalence block guarantees that the data accesses are consistent with the dataset definitions and ensures that all references are with a common mnemonic.

3. A compact form of the dataset definition for use by the OUTPUT-DATASET (OTPTDS) routine, which allows the contents of datasets to be printed in a readable form during the running of a simulation.

3.6.4 FLEXREAD*

The power of DSA is augmented by program FLEXREAD, which creates data structures under the direction of input cards. Information in the input deck causes dataset instances to be created, filled with initial data values, and assigned Data Set Pointers or List Header Variables for structure definition. FLEXREAD provides an extremely readable and flexible form of input, with a 40-character identification for each variable and automatic conversion of units for those values that are input in a system different than that used internally.

A major strength of the FLEXREAD system is that it permits a programmer to write his code with no concern about the quantities of the various elements being simulated. He manipulates a list of items of arbitrary length, using system routines. FLEXREAD allows a user to define the list elements at execution time. He may wish to simulate a single radar, or a hundred, and accomplishes this by supplying the proper number of data definition cards.

FLEXREAD requires the definition of a "Basic" dataset from which all other elements of the input data structure may be accessed. In typical use, one word of the Basic dataset is assigned to point to a sub-data structure for a particular model. Models may be represented by modules of different fidelity which might require more or less elaborate data

---

*J.A. Bardens and L.R. Ford, Jr., FLEXREAD User's Manual; A General Purpose Input Program for the Dynamic Storage Allocation System, General Research Corporation RM-1447, August 1972.

structures. However, each module's data would still be input through that model's assigned word in the Basic dataset. Data structure definition for more than one module may be included in the input deck; however, FLEXREAD uses only those datasets which are reachable from the Basic dataset in much the same way as a program loader will load only those programs reachable from the main program.

Table 3.5.5 in Sec. 3.5.4 (p. 362) shows the printout from FLEXREAD for a TACRAN3 run.

3.6.5 Action Sequence Chains

The GRC-developed simulation system used in TACRAN is event based, with several features which permit considerable flexibility. (This same simulation system is used in the TAC Assessor simulation[1] being developed by GRC for Air Force Studies and Analyses. This simulation is the testbed and forerunner to the future CASM--Combined Arms Simulation Model--simulation.)

Each event to occur in the future is represented as a dataset which contains the time at which the event is to occur in the simulation as well as other data describing the particular event. The events are attached to a list in time order. The event-processing logic removes the first event dataset from the list and schedules whatever event it specifies. As examples, an event might be a radar return to be processed, or a message received to be processed. After the event is processed (i.e., "occurs") the event processor goes on to the next event.

Each event is an "Action Sequence Chain" (A.S.C.), which is a sequence of actions (code) that may include delays. If a delay occurs during an A.S.C., a new event is added to the event list whose time is the time when the processing of this A.S.C. should resume after the delay. The event processor then processes events that occur before the delay is over, and at the appropriate time resumes processing the A.S.C. In the

---

[1] D. Cooper, S. Kiselewich, and L. Ford, TAC Assessor Final Report, General Research Corporation CR-5-792, December 1978.

usual simulation run, many such A.S.C.'s with delays will be being processed during the same period of time.

The processing of delays in an A.S.C. is identical to the processing of the CASE statement described in Sec. 3.6.1. The A.S.C. is imbedded in a RESUME FLAG...END RESUME sequence:

```
SUBROUTINE [A.S.C. name]
   :
RESUME FLAG (NEXN XA(NEVENT))
   :
RESUME (4HBGIN)
   :
(Delay Statement 1)
RESUME (value 2 or variable 2)
   :
(Delay Statement 2)
RESUME (value 2 or variable 2)
   :
RESUME ELSE                          [optional]
   :
END RESUME
   :
RETURN
END
```

When the A.S.C. (i.e., "event") is first entered (i.e., the event "occurs"), the RESUME FLAG expression, which always has the FORTRAN variable name NEXN XA(NEVENT), is set to the Hollerith value 4HBGIN. (NEVENT is the dataset index of this A.S.C.'s dataset.) The subroutine is entered at the top (as usual in a FORTRAN subroutine), the code between the top

and the RESUME FLAG statement is executed, and any code between the RESUME FLAG and first RESUME statement is executed. When the first RESUME statement is reached, the value of NEXN XZ(NEVENT) is checked to see to which RESUME statement control should jump to. In the beginning it is the statement RESUME (4HBGIN), therefore code following this statement is executed.

When a delay statement (described below) is reached, a new event is created ΔT seconds in the future (where ΔT is the delay specified in the delay statement), and NEXN XA(NEVENT) is set to the value of the argument in the RESUME statement where control is to resume after the delay is up. When the first RESUME statement after the delay statement is reached (a RESUME usually directly follows a delay statement), control jumps to the code after the END RESUME. This code may simply be a RETURN, and control passes back to the event processor for initiating the next event on the event list.

When control passes back to this A.S.C. ΔT seconds in the future (the event processor treats this as a new event), the value of NEXN XA(NEVENT) is the value saved. Again the subroutine is entered at the top, all instruction down to the first RESUME statement are executed, and then control jumps to the RESUME statement with the value or variable value that is the same as NEXN XA(NEVENT).

This process can continue through as many delay and RESUME statements as required.

Two different delay statements are used in TACRAN: (1) DELAY, and (2) WAIT. DELAY is a FORTRAN subroutine that permits a simple delay of ΔT seconds. Its form is

```
CALL DELAY(ΔT, ARG)
```

where ARG is the value which is used in the RESUME statement to which control is to return after the delay of ΔT.

WAIT is a FORTRAN function that permits control to resume at two places under two different conditions. It can resume after a specific event has occurred (such as the recepit of an expected message) or after a specified delay has occurred (in case the message is not received within a given time), whichever occurs first. WAIT is implemented as a function so that it reads well when used with an IF statement:

```
IF (WAIT(ID,ΔT,ARG1,ARG2) .EQ. ARG1)
RESUME (ARG1)
   :
ELSE
RESUME (ARG2)
   :
END IF
```

The ID tells which dataset (e.g., which message) the system is waiting for. If the dataset appears before ΔT passes, then control is passed to RESUME (ARG2). If ΔT passes before the dataset appears, control is passed to RESUME (ARG1).

There exists several other features of the simulation system that were not specifically used in TACRAN and therefore are not described here.

3.6.6 Post Processor

A TACRAN run produces a prodigous amount of output, and it would be cumbersome to print it all when only selected parts of it may be required. Also it may be convenient to see the data processed in different ways, perhaps some of it plotted.

Data Gathering. The simulation uses input flags to allow the user to select run data that is to be saved for post processing. Some data is always saved, but other data is optional. Data which might be saved during a TACRAN run includes all measurements from all radars, all track records, and all messages.

AD-A067 936 GENERAL RESEARCH CORP SANTA BARBARA CALIF F/G 17/9
TACTICAL FORWARD AREA SURVEILLANCE AND CONTROL INTERNETTING STU--ETC(U)
NOV 78 G W DELEY, J H BALLANTINE F19628-78-C-0020
UNCLASSIFIED CR-1-814-VOL-2 ESD-TR-78-180-VOL-2 NL

5 OF 5
ADA
067936



END
DATE
FILMED
6-79
DDC

The data is stored unformatted on a file in records; each data type has its own record type number so that it can be easily retrieved and collected for post processing.

Post Processor Run. A post processor run is directed by input data which, in a specified format, tells the post processor what it is to do with the data on the file created during the simulation run. For example, it might be desired to print out all measurements from Radars 1, 2, and 6 separately and in time order.

Selection, ordering, and formatting are the usual functions performed in the post processor. Examples of output from TACRAN that have been selected, ordered, and formatted are shown in Table 3.5.6 in Sec. 3.5.4 (p. 371).

A plot package is also a part of the post processor. All of the machine-made plots shown in this report [e.g., Figs. 3.5.31 through 3.5.38 in Sec. 3.5.3 (p. 339)] were made using the post processor.

Data Selection and Processing Options. The post processor has considerable flexibility to manipulate data. Data is gathered either directly from the saved file or as some derivative of that. Data is grouped into conglomerations called Items, which are of arbitrary length. Items can be viewed as groups of records (these can be records straight from the file but need not be) strung out as a matrix. The matrix can be a one-record vector or a ten thousand-record clump of data.

Items which come directly from the file are of two kinds--short Items which contain statistics on data from a certain file record type, or else actual raw data gathered from that record type.

Not all data of a given record type need be gathered from the file. Skipping can be specified (take every third record 3) or complex logic

can be applied. However, all the data must be contained in one record type.

Several means of data massaging are available. In addition to scaling, concatenating, and mathematically combining, data can be aggregated by sorting an Item on several keys and doing statistics on the resulting groups of rows from the original Item. Each group produces one row in the new Item.

Row differencing is available. This is especially useful for link data gathered at sampling intervals; this data is not reinitialized at each sample so needs to be differenced. The difference number is given, so rows any number apart can be differenced as long as the original Item has that many rows.

A new Item can be built from one or two old ones by specifying that series of binary operations be performed using specified words from the old or new Items to create words of the new Item. As an example, one could specify that each word across a row of the new Item be the corresponding word of one old Item multiplied by word 8 of the other old Item, then divided by 60. Any number of operations can be performed, forming a sort of macro function which can be performed on one or on several words of a row. Several functions can be specified and stacked in order.

Any Item can be sorted on several keys. The original conformation of the Item is not changed, and the record of the sort is kept separately.

Data Presentation. Once the data has been selected and processed, the user must decide how it is to be presented. Several options are available. For example, various forms of printout are available. Items can be printed in their entirety using "canned" formats. Each word of each record type of the input file has a standard format and title associated with it. Often when a new item is formed from an old one, these

formats are passed along with the corresponding words. However, if the user wishes to specify his own formats, he may do so. Words from several different Items can be printed in the same table on the page if the items are the same length.

Printer plots are available and can be used in conjunction with pen-and-ink plots if desired. Whenever a pen plot is requested, a printer plot is also generated. Plots are either histograms or regular curve plots. Curves can appear as individual points or as lines. Up to four ordinates can be plotted against one abscissa, and scaling can be communal, individual or mixed on a curve-by-curve basis. The ordinates and abscissa need not come from the same item, but they must have the same length.

Data can be automatically presented as a sort of histogram table. The starting value and increment are input, and for each interval the number of points being in that interval are printed, where "point" refers to collections for all rows of an Item of values for a specific word in the row--all of the word 5's, for instance, for Item 7.

At the end of a post-processor run, a summary is printed listing the minimum mean, standard deviation, and number of points for each Item.

3.6.7 TRAID[1]

The GRC TRAID (TRajectory AID) system is a family of library subroutines designed to allow rapid, easy, and accurate construction of simulations and models. TRAID can be thought of as a language of higher order than FORTRAN although in the strictest sense it is not a syntactical language but a system of user-called subroutines. It can be used to fly aircraft, perform orbital-mechanics calculations of various types, and is particularly useful as part of sensor and system simulations which involve the movement of objects in space or in the earth's atmosphere.

---

[1] T. Plambeck, The TRAIDsman, General Research Corporation IMR-1131/1, July 1969 (revised March 1973).

The TRAID system contains subroutines which move objects under various conditions and constraints, model the effects of the atmosphere, and perform various auxiliary computations. More specifically, these subroutines:

1. Place and evaluate Keplerian orbits over a fixed or rotating spherical earth, with or without drag
2. Calculate and integrate powered and guided trajectories of aircraft and missiles
3. Manipulate vectors and matrices
4. Perform coordinate transformations
5. Carry out input/output functions
6. Perform miscellaneous functions such as plotting, interpolation, produce debugging output, control iteration processes, calculate random numbers

The greatest value of the TRAID system is the ease with which simulations of aircraft, satellites, reentry objects, interceptors, radars, and optical sensors may be put together. The analyst finds that the desired simulation program falls into place with speed and few inherent errors. Furthermore, he can join the "building block" routines in the order that they occur in his thought processes. Using TRAID, it is quite common to construct useful simulations in a matter of hours. Very complex simulations can often be constructed in weeks.