DNA 3964F-4

THE ROSCOE MANUAL

Volume IV – Systems Models

General Research Corporation
P.O. Box 3587
Santa Barbara, California 93105

October 1975

Final Report for Period 1 March 1974—30 September 1975

CONTRACT No. DNA 001-74-C-0182

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED.

THIS WORK SPONSORED BY THE DEFENSE NUCLEAR AGENCY UNDER RDT&E RMSS CODES B322074464 S99QAXHC06428 H2590D AND B322075464 S99QAXHC06432 H2590D.

Prepared for
Director
DEFENSE NUCLEAR AGENCY
Washington, D. C. 20305

Destroy this report when it is no longer needed. Do not return to sender.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: DNA 3964F-4	2. GOVT ACCESSION NO.
	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): THE ROSCOE MANUAL, Volume IV, Systems Models	5. TYPE OF REPORT & PERIOD COVERED: Final Report for Period 1 Mar 74—30 Sep 75
	6. PERFORMING ORG. REPORT NUMBER: CR-1-520, Volume 4
7. AUTHOR(s): H. S. Ostrowsky, P. J. Redmond, J. R. Garbarino, R. E. Rein	8. CONTRACT OR GRANT NUMBER(s): DNA 001-74-C-0182
9. PERFORMING ORGANIZATION NAME AND ADDRESS: General Research Corporation, P. O. Box 3587, Santa Barbara, California 93105	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: NWED Subtask S99QAXHC064-28-32
11. CONTROLLING OFFICE NAME AND ADDRESS: Director, Defense Nuclear Agency, Washington, D.C. 20305	12. REPORT DATE: October 1975
	13. NUMBER OF PAGES: 216
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

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

CR-1-520-Vol-4

18. SUPPLEMENTARY NOTES

This work sponsored by the Defense Nuclear Agency under RDT&E RMSS Codes B322074464 S99QAXHC06428 H2590D and B322075464 S99QAXHC06432 H2590D.

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

Nuclear Effects
Radar
Optical Sensors
Computer Program
Simulation
Ballistic Missile Defense

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

The ROSCOE computer code is designed specifically to be the "laboratory standard" for evaluating nuclear effects on radar systems. The program provides a means for (1) evaluating radar acquisition, discrimination, and tracking performance in a nuclear environment, (2) measuring various propagation error sources, and (3) computing specific phenomenological data.

(Continued)

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

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

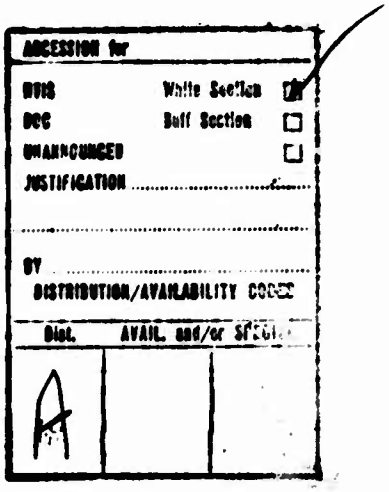
402754

20. ABSTRACT (Continued)

The ROSCOE users manual is divided into four volumes:

Volume I: Program Description
Volume II: Sample Cases
Volume III: Program Structure
Volume IV: Systems Models

Volume I provides the user with a brief overview of the code, a description of the input and output, and some sample input decks. Volume II contains illustrative examples showing the use of the code in various modes. Volume III presents structural details of the code (such as the subroutine structure and dataset structure) for the programmer-user. Volume IV describes the systems content of the code.



CONTENTS

CONTENTS (Cont.)

CONTENTS (Cont.)

CONTENTS (Cont.)

ILLUSTRATIONS

ILLUSTRATIONS (Cont.)

ILLUSTRATIONS (Cont.)

1 INTRODUCTION

This volume describes the systems content of the ROSCOE code. The systems model, as defined here, consists of these major blocks: attack generation, object observables, sensor platforms, track filter, radar event logic, radar signal processing, and radar propagation.

The objective in building the systems model was to provide a very generalized set of routines which could be used by a variety of users for a variety of applications. As a result, the user can select from among a number of options to simulate a particular engagement. For example, any number of different object types, radar types, and weapon types can be specified in a single run. In addition, launch and impact points can be mixed to generate complex attacks.

Section 2 of this volume briefly describes the ROSCOE models of attack generation and object motion, object radar observables, sensor platforms, and radar track filter. References are given for fuller descriptions.

Section 3 describes the radar logic in terms of the radar "events": search/detect/verify, track initiation, track, and discrimination. Assumptions are explicitly identified; procedures are described in detail; and annotated flow charts are presented.

Section 4 describes the ROSCOE models of the signal processing that is applied to each radar pulse (radar look event). These models account for the effects of the environment (by calling the propagation routines, Sec. 5) and of radar characteristics such as monopulse processing, range gating, and signal design (ambiguity function).

Section 5 describes the radar propagation models. These models link the systems part of ROSCOE with the environment part by calculating the effects on radar signals of absorption, fireball noise, refraction, clutter, and multipath.

Appendixes supply the background for some of the signal processing algorithms.

2 MISCELLANEOUS MODELS

2.1 ATTACK GENERATOR AND OBJECT MOTION

The attack generator for ROSCOE is taken from General Research Corporation's BAG 14 code.[1] It consists of the following set of programs:

ATKGEN	Attack generation program
LAUNCH	Creates objects and object position arrays
PHIMP	Creates burst arrays at impact

The programs use the following subroutines to generate the object trajectories and find specific points on them:

ORB2	Determines the elements of a Keplerian orbit joining two points*
ORB1	Determines the orbital elements corresponding to a given state*
ORBP	Determines the state vector of a point on a Keplerian orbit*
BLLSTIC	Generates an endo-atmospheric trajectory with slowdown
DRORB2	De-orbit routine which modifies an orbit to meet a specified impact point
GENORB	Corrects an orbit to meet a specified impact time
ADVANCE	Advances the state and covariance of an object to the current time
KUTTA	Trajectory integration routine
PREDLOC	Returns a predicted object location
WHERE	Finds object state at a specified time

*These routines are part of the General Research Corporation TRAID system.[2]

Figure 2.1 is a flow diagram of the main program in this module (ATKGEN). The input consists of a set of launch points (dataset LP) and a set of target points (dataset TG) specified by the user. These datasets may have repeated entries and may be in quite arbitrary order. The attack generator takes each target point in order. Associated with the target point as an input is the number of reentry objects which the target point is to receive. This number of objects is assigned to the target point one at a time by a sequential pass through the list of launch points. If the end of the list is reached before all the target points are satisfied, the program cycles back to the front of the list of launch points and continues. Thus, in a simple attack with a single launch point, one entry on the list is sufficient. It should be noted parenthetically that object type is peculiar to launch point; thus to mix RVs and decoys requires mixing launch points on the launch point list.

The timing of the attack and the type of trajectory are specified with the target point (dataset TG). These data include the desired arrival time for the first booster, the desired time separation between objects, the sigma of arrival time errors, and the CEP of position errors. An alternative form of timing specification allows the user to specify a fixed time interval inside which the reentry objects assigned to a particular target point arrive in random order.

The trajectory is specified as in TRAID[2] and is calculated with drag on a round, rotating earth. For each reentry object, the attack generator generates a launch event at the time at which the Keplerian position of the orbit leaves the launch point. The storage for each object will be set aside when the launch event is processed--all that is saved during the attack generation is the computed orbital elements, which are appended to the launch event.

The launch event creates an object and names it. (The object LP-5 7 is the seventh launch from the launch point called LP-5.) Object and object position arrays are created. Orbital element and reentry

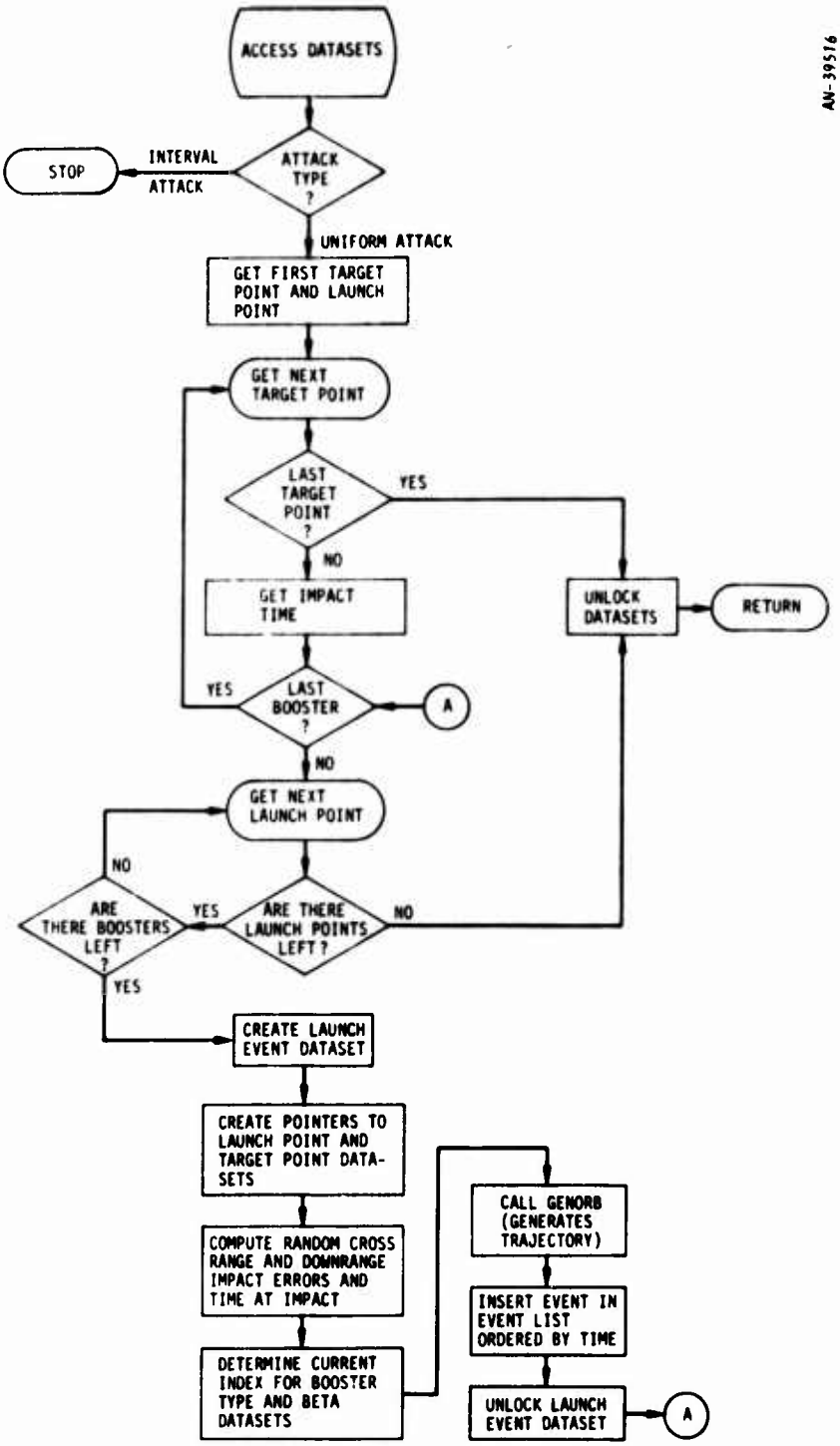


Figure 2.1. Program ATKGEN

position state vectors are constructed, as is a summary output array for that object. Future events associated with the object are constructed and placed on the event list. These are

- The impact event (Program PHIMP)
- A potential acquisition event (Program RADAR) for each radar in the defense system

The impact event is created at the same time the object is created and occurs at the moment that the object trajectory passes through its designated target point. In particular, this means that if an airburst (or a precursor) is desired, the target point should be specified at a non-zero altitude.

2.2 OBJECT OBSERVABLES

This section describes the following object observables models:

1. Ballistic coefficient
2. Tumbling dynamics
3. Radar cross section

The models are taken from General Research Corporation's SETS model.[3]

In the following sections, flow diagrams are shown for each routine with explanatory notes to define the variables and describe the computations.

2.2.1 Ballistic Coefficient

The ballistic coefficient (beta) is found by calling Function BETAGT (Fig. 2.2) with the object state and a dataset pointer (DSP) to the ballistic coefficient model type desired. There are three model types: type 1 uses a constant value of beta for all altitudes; type 2 uses an N-point altitude-beta table, resulting in a piecewise-linear relation between beta and altitude; and type 3 relates the beta history of an object to the dynamics of a theoretical conical vehicle of constant mass and shape.

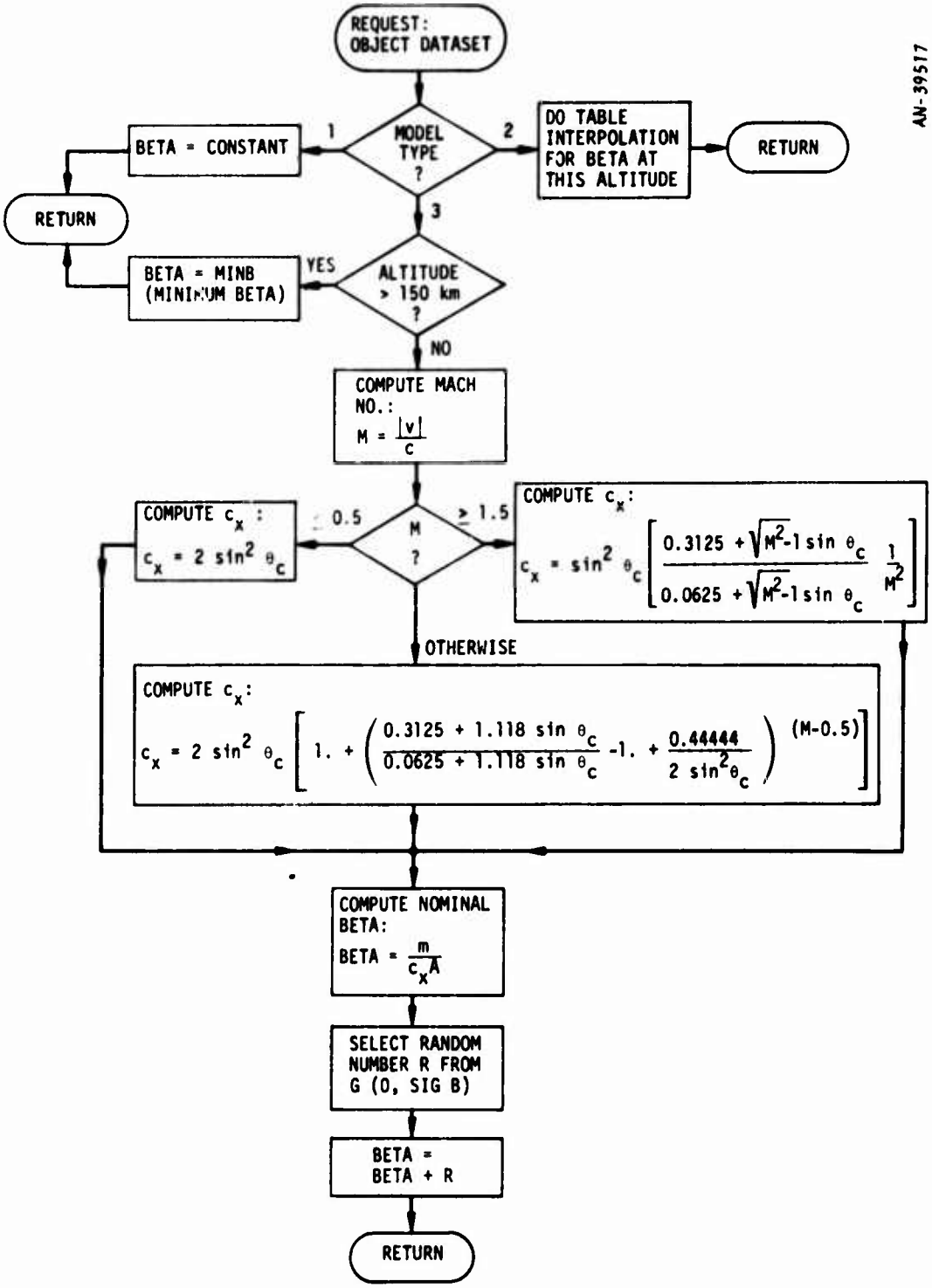


Figure 2.2. Function BETAGT

In type 2, the altitude-beta pairs are given in order of decreasing altitude. For object altitudes above the first in the list, the beta value for the highest altitude is used. For object altitudes below the last in the list, the last beta value is used.

In type 3, the quantities shown in Fig. 2.2 are defined as follows:

m = mass of vehicle, lb

A = aerodynamic reference area, ft^2

c_x = axial drag coefficient

θ_c = theoretical cone angle, deg

M = mach number

c = local speed of sound, m/s

v = current vehicle velocity, m/s

At altitudes above 150 km, type 3 uses an input minimum beta, with no further computation required.

The ballistic coefficient computed by model type 3 may be randomized by adding a number drawn from a Gaussian distribution G(0, SIGB) with zero mean and an input standard deviation SIGB.

2.2.2 Tumbling

The tumbling model (Subroutine TUMBLR, Fig. 2.3) is called with the object state vector and a DSP word denoting the model type, and returns a vector (AVEC) describing the motion of the object about its point mass trajectory.

There are three tumbling models available. They all assume that the object is a solid of revolution with one axis of rotational symmetry (body axis). It is assumed that the cone swept out by the tumbling body axis differs negligibly from a plane (tumble plane). Type 1 keeps the object oriented along its current velocity vector; therefore, no data is

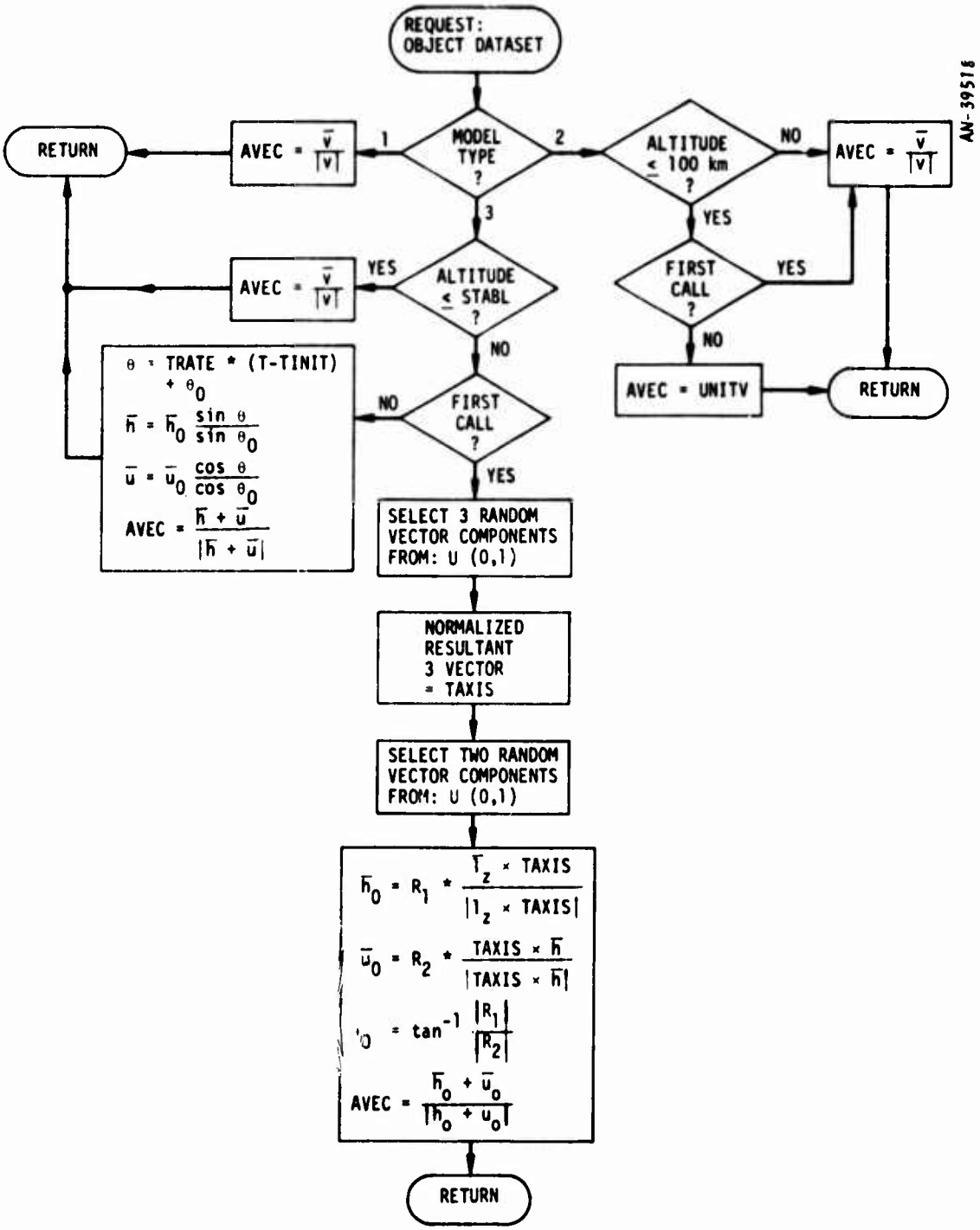


Figure 2.3. Subroutine TUMBLR

required for this model. Type 2 orients the body along its initial reentry velocity vector (computed at 100 km, or at the object's current altitude if the initial call occurs below 100 km). Type 3 is a stochastic model which allows an object to tumble at a given rate until its body axis stabilizes along the velocity vector at a given (input) stabilization altitude. The initial body orientation and tumble plane are derived as follows:

1. Three random components are chosen from a uniform random distribution.

2. A three-vector formed by normalizing these components defines a tumble axis (TAXIS).

3. A tumble plane normal to this axis is formed and a random vector in the plane chosen by selecting two more random numbers from the uniform distribution. This latter vector is called the body axis (AVEC).

4. After initialization, the body axis is updated by using the input tumble rate and the definition of the tumble plane ($\bar{h}_o, \bar{u}_o$).

2.2.3 Radar Cross Section

Four models are available for modeling an object's RCS (Function RCSMODL, Fig. 2.4). Type 1 assumes the RCS is constant and requires no other input data. The other three models compute RCS as a function of the aspect angle (defined as the angle between the radar line-of-sight and the current orientation of the body axis as given by the tumbling model).

In the type 2 model, the RCS is found by interpolation of an RCS vs aspect angle table. Type 3 is intended for tank-like objects (cylinders), while type 4 is for RVs and decoys.

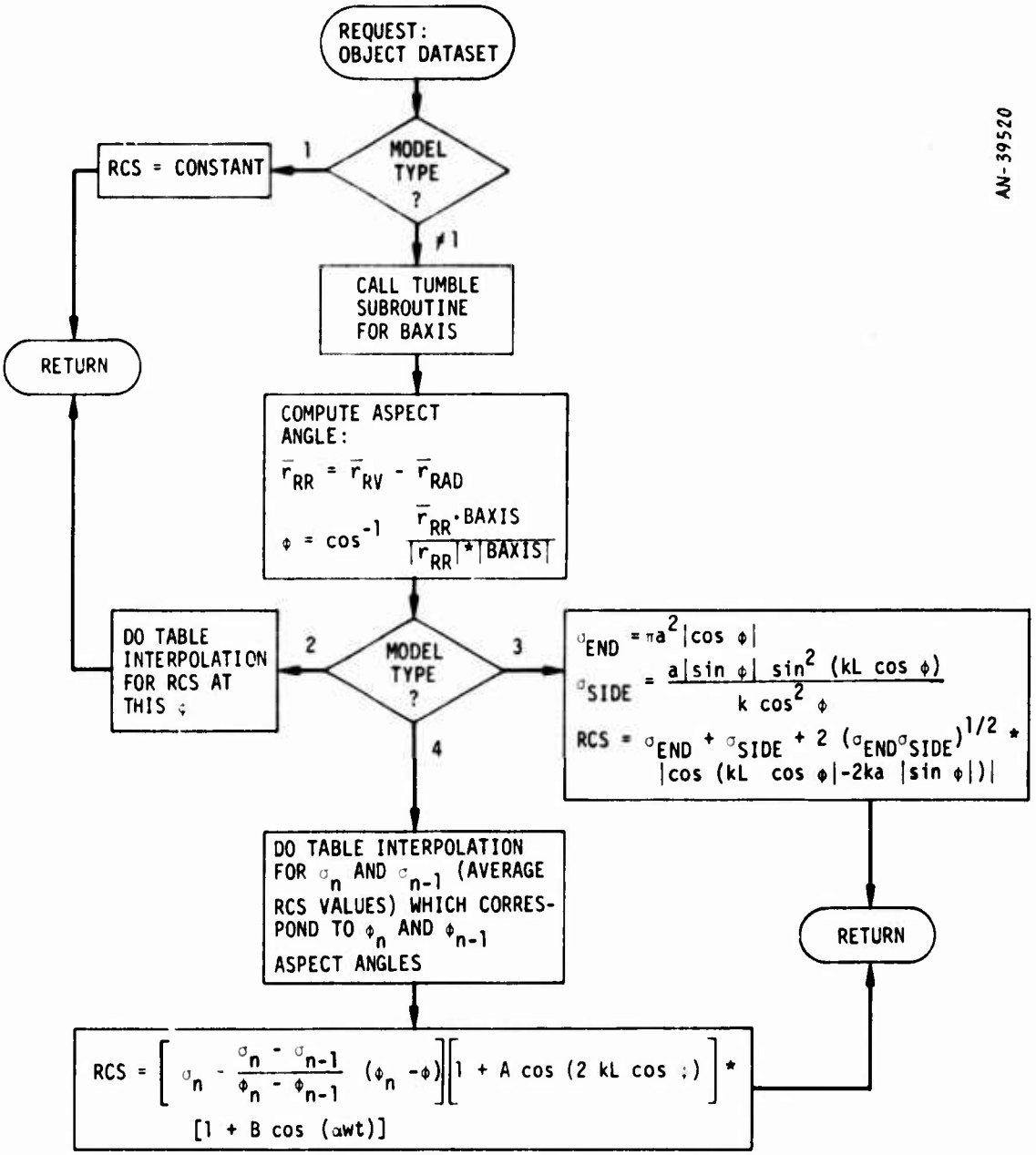


Figure 2.4. Function RCSMODL

In Fig. 2.4, the variables used in type 3 are defined as follows:

a = radius of tank, m

L = length of tank, m

ϕ = aspect angle, deg

$k = 2\pi/\lambda$

λ = wavelength, m

and the variables used in type 4 are defined as follows:

w = spin rate, rad/s

α = number of RCS peaks per revolution

A,B = constants for vehicle in question, as used in the equation shown in Fig. 2.4

2.3 SENSOR PLATFORMS

Three sensor platform models are currently available in ROSCOE: fixed, orbital, and circular-orbit. They are specified by placing the proper dataset (P1, P2, or P3, respectively) in the input deck. The platform (radar) location is found by calling subroutine PLTFRM with a pointer to the dataset containing the platform position information. The routine returns the position through the calling sequence.

The "fixed" platform is simply described by a position vector. The "orbital" platform is described by a TRAID ten-vector giving its orbital elements. The "circular-orbit" platform is described by a simplified set of orbital elements defining a circular orbit.

2.4 TRACK FILTER

The GRC BAG 14 tracking filter has been suitably modified for use in ROSCOE. It is a fully coupled seven-state Kalman filter with an option for exponential memory decay.

The filter module is made up of five subroutines:

FILTER	Initializes covariance and updates filter predictions (Fig. 2.5)
CHOLSKI	Inverts an N × N matrix
JACOBIAN	Sets up Jacobian matrix for filter routine
XFORM	Inverts Gamma matrix to Covariance and vice versa
KALMAN	Kalman filter

The formulation of the tracking filter, expressed in radar face coordinates, is as follows. If we define the three-dimensional vector $\vec{x}$ by

$$x_1 = r \sin \alpha$$

$$x_2 = r \sin \beta$$

$$x_3 = r(1 - \sin^2 \alpha - \sin^2 \beta)^{1/2}$$

where r, α, β are range and radar angular coordinates, respectively, then the seven-dimensional state vector estimated by the filter is given by

$$\vec{X} = (\vec{x}, \dot{\vec{x}}\Delta, \lambda)$$

where Δ is the time between pulses for a given track file and λ is the natural logarithm of the ballistic coefficient. The equations of motion giving the time history of $\vec{X}$ are assumed to be

$$\left.\begin{aligned} \ddot{\vec{x}} &= \frac{-\rho g(e)^{-\lambda}}{2} |\dot{\vec{x}}| \dot{\vec{x}} - \frac{GM\vec{R}}{|R|^3} + \vec{\omega} \times \dot{\vec{x}} + \vec{\omega} \times (\vec{\omega} \times \vec{R}_e) \\ \vec{R} &= \vec{x} + \vec{R}_e \\ \dot{\lambda} &= \mu(t) \end{aligned}\right\} \quad (2.1)$$

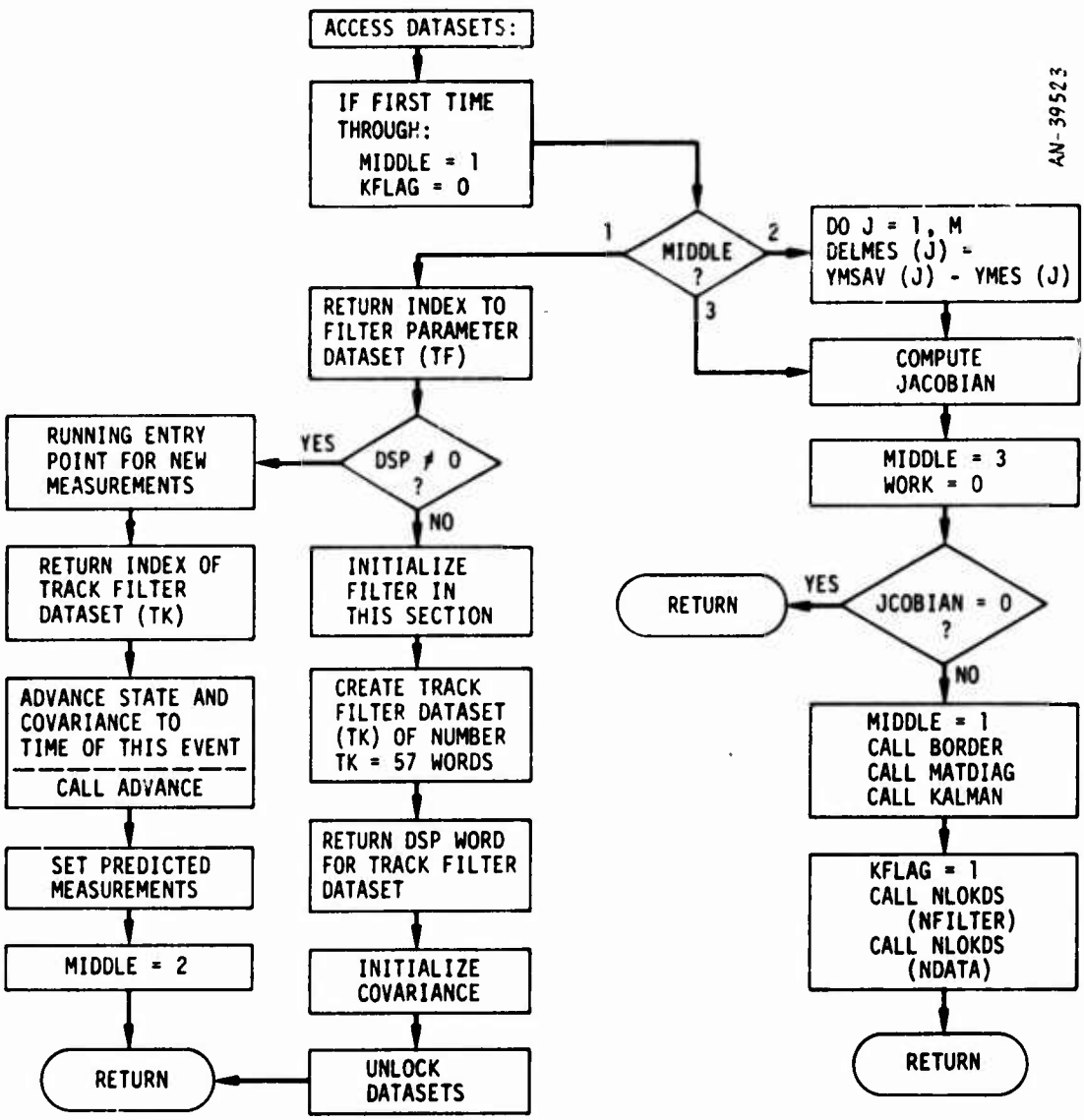


Figure 2.5. Subroutine FILTER

where $\vec{R}_e$ = vector from earth's center to radar

$\vec{\omega}$ = earth's angular rotation vector

μ = stationary, uncorrelated random variable with mean zero and standard deviation σ_μ

G = universal gravitational constant

ρ = atmospheric density

M = mass of earth

This formulation differs from the standard three-degree-of-freedom equation for reentry motion in that no lift is considered, and the centripetal acceleration term is taken as a constant.

The ROSCOE tracking filter is basically an extended Kalman filter and, as such, gives a running estimate of the error covariance matrix C associated with $\vec{X}$. The filter can be separated into three distinct operations--initiation, prediction, and correction--of which only the last two are repetitive. To distinguish the estimates of the state vector and covariance matrix resulting from the prediction and correction steps, they are given the subscript p or c, respectively.

The filter can be initiated either through the track initiation logic or through handover from another radar. In both cases, it is assumed that the first six components of $\vec{X}_p$ are given at the time of the first tracking pulse, along with some estimate of variance and covariance between them. The initial estimate of λ is currently $\ln(100\ \mathrm{lb/ft^2}) = 4.605$, and the initial variance is equal to 4. The covariances between λ and the remaining components are initiated as zero.

Having an estimate of $\vec{X}$ and C at some time t, the filter must be able to predict the best estimate of these quantities at some future time, usually to determine the radar gating and beam pointing for the next

pulse. The prediction step is always carried out over one cycle time Δ. The subvectors of $\vec{X}$ are predicted by the equations

$$\left.\begin{aligned} \vec{x}_p(t+\Delta) &= \vec{x}_c(t) + \dot{\vec{x}}_c(t)\Delta + \ddot{\vec{x}}_c(t)\Delta^2/2 \\ \dot{\vec{x}}_p(t+\Delta) &= \dot{\vec{x}}_c(t) + \ddot{\vec{x}}_c(t)\Delta \\ \lambda_p(t+\Delta) &= \lambda_c(t) \end{aligned}\right\} \quad (2.2)$$

where $\ddot{\vec{x}}_c$ is evaluated from Eq. 2.1. For the first prediction, the initial values are used in Eq. 2.2. The covariance matrix is predicted through the use of an approximate transition matrix $\phi(t+\Delta,t)$. The usual approximation starts with the equations of motion written in the form

$$\dot{\vec{X}} = \vec{f}(\vec{X})$$

Then the equations are linearized about some nominal value of the state vector

$$\delta\dot{\vec{X}} = (\dot{\vec{X}} - \dot{\vec{X}}_{nom}) \simeq \frac{\partial\vec{f}}{\partial\vec{X}}(\vec{X}_{nom})\delta\vec{X} = F[\vec{X}_{nom}(t)]\delta\vec{X}$$

where $\vec{X}_{nom}$ is usually taken to be the best estimate $\vec{X}_c$. The linearized equations then furnish a transition matrix through the matrix formula (to first order in Δ):

$$\phi(t+\Delta,t) = \exp \int_t^{t+\Delta} F(t')\,dt' \simeq I + F(t)\Delta$$

In the ROSCOE tracking filter, the further approximation is made in F that

$$\frac{\partial\ddot{\vec{x}}}{\partial\vec{x}} = \frac{\partial\ddot{\vec{x}}}{\partial\dot{\vec{x}}} = 0$$

so that only the partials with respect to λ contribute to the last four rows of ϕ. Once ϕ is evaluated, C is predicted through the formula

$$C_p(t+\Delta) = \phi(t\ \Delta,t)C_c(t)\phi^T(t+\Delta,t) + Q \qquad (2.3)$$

where Q is a matrix whose only non-zero element is in the last row and column and is equal to σ^2_{μ}.*

When a measurement is made, it is incorporated into the estimate through the correction step. For numerical reasons, a pseudo-observation vector $\vec{Y} = (x,y,r)$ is computed from the measured values of r, sin α, and sin β. The single-pulse variance estimates for r, sin α, and sin β are computed from the measured signal-to-noise ratio and the pulse type. If the matrix ψ is defined as

$$\psi = \begin{bmatrix} \sin\alpha & r & 0 \\ \sin\beta & 0 & r \\ 1 & 0 & 0 \end{bmatrix}$$

then the estimated error covariance matrix for $\vec{Y}$ is computed through the formula

$$R = \psi \begin{bmatrix} \sigma^2_r & 0 & 0 \\ 0 & \sigma^2_{\sin\alpha} & 0 \\ 0 & 0 & \sigma^2_{\sin\beta} \end{bmatrix} \psi^T$$

The 3×7 matrix M is defined by

$$M_{ij} = \frac{\partial Y_i(\vec{X})}{\partial x_j}\bigg|_{\vec{X} = \vec{X}_c}$$

*Currently σ^2_{μ} is taken as 0.01Δ for altitudes less than 250 kft and zero otherwise. The Q matrix keeps the filter memory finite within the atmosphere.

Then the correction formulas are given by

$$\vec{X}_c(t) = \vec{X}_p(t) - W[\vec{Y}(t) - M\vec{x}_p(t)]$$

$$C_c(t) = C_p(t) - WMC_p(t)$$

$$W = C_c M^T [MC_c M^T + R]^{-1}$$

If for some reason no measurement is available, the filter sets $\vec{X}_c = \vec{X}_p$ and $C_c = C_p$ in preparation for the next prediction step.

The use of the Q matrix in Eq. 2 3 keeps the filter memory finite within the atmosphere and is employed only below 250 kft altitude. Another useful technique for decreasing memory length in tracking filters is through the multiplication of the covariance matrix by a "fading factor"

$$C'_p = \Gamma C_p \Gamma^t$$

where Γ is a diagonal matrix having non-zero elements greater than or equal to one. In the ROSCOE filter, a variation of this approach is used wherein the fading factor is applied only to the diagonal elements of the covariance matrix. The elements along the diagonal of Γ are given by $(1,1,1,\gamma,\gamma,\gamma,1)$ where $\gamma = e^{-\tau}$ and τ is a constant which can be varied with altitude.

3 RADAR EVENT LOGIC

The ROSCOE radar model may be divided into two subsets: radar event logic and radar signal processing.

The radar event logic is responsible for setting up and simulating the search, verification, track initiation, track, and discrimination events. Each event must contain the logic for selecting the appropriate signals, transmitting them to the appropriate points, calling for the correct data processing of any return, deciding (based on the latest return) what further events are necessary, and finally setting up future calls for these events at the appropriate time.

This type of event logic is essentially independent of how the radar signal processing is simulated. The latter therefore constitutes an independent segment of the radar model, and will be discussed separately in Sec. 4.

The event logic is described in the subsections which follow. Each subsection describes the assumptions which underlie the logic, the process which is being modeled, and any limitations of the model. Flow charts are presented and discussed.

3.1 SEARCH/DETECTION/VERIFICATION EVENTS

3.1.1 Assumptions

During these events, the system is assumed to operate without monopulse. The target angular position is assumed to be that of the commanded axis of the receive beam. Many real or proposed radar systems operate without monopulse in the search mode, so this is a realistic model.

Bulk filtering, which is a part of the search/verify procedure in many real radar systems, is not modeled in ROSCOE.

It is assumed that the "real target" is known and distinguished from other targets. In particular, when multipath is present, only one of the target images is detected by this procedure; the method by which this target is to be selected is described later in this discussion. When real multiple targets are present in the beam(s), the association and filtering necessary to distinguish the correct new target are not modeled, but are assumed to be done perfectly. Extraneous targets, whether real or images, appear principally in the form of "noise" affecting the return from the "real" target.

The search sector is a sector (between specified azimuths) of a volume of revolution whose vertical cross section has the shape shown in Fig. 3.1, defined by maximum and minimum ranges, altitudes, and elevation angles. Note that these quantities cannot all be independently specified; to prevent a gap at the range/altitude transition, it is required that

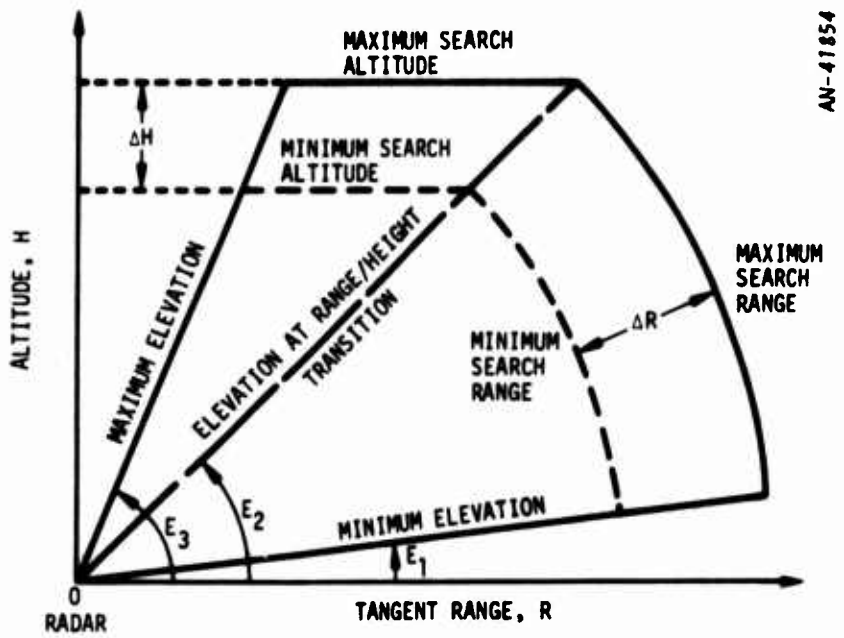


Figure 3.1. Vertical Slice Through Search Sector with Range-Height Transition

$$\frac{\text{max. alt.}}{\text{min. alt.}} = \frac{\text{max. range}}{\text{min. range}}$$

The range/altitude transition occurs at an elevation angle

$$E_2 = \sin^{-1} \frac{\text{max. alt.}}{\text{max. range}}$$

If this angle exceeds the maximum elevation, or if maximum altitude $\geq$ maximum range, there is no flat portion of the search sector. Such a search sector might be used by an area-defense radar; compare, for example, the radars in the two scenarios specified in the appendixes of Vol. 1 (Search Mode Parameters Dataset). An airborne or orbiting radar might have a negative minimum elevation angle for search. It should go without saying that the angular extent of the search sector for any radar face should lie within the field-of-view specified for that face. In the program, it is implicitly assumed that this is the case.

If the target appears within the search sectors of more than one radar face, detection is first attempted by that face for which the target has the smallest off-boresight angle at the time it crosses the outer boundary of the search sector.

The time of flight of radar signals is not modeled.

3.1.2 Procedures

The user may elect to make the first try at detection as soon as the target crosses the outer boundary of the search sector, thus ignoring the random orientation of the search frame relative to the target position. Alternatively, he may select the option of modeling this "frame randomization" by making the first try at detection at some randomly-selected instant during the first search frame period after the target crosses the outer boundary.

If the first try at detection fails, succeeding tries are made at intervals of the search frame time, T_s, after the first try, until either a successful detection is made or the target crosses the inner boundary of the search sector. In the latter case, detection fails to occur.

The search frame time, T_s, cannot be chosen completely at random. It must be short enough, compared with the target velocity and the range window (ΔR), so that the radar gets <u>at least</u> two chances at any target before it crosses the inner boundary. Since the first chance might (because of frame randomization) occur as much as T_s after the target crosses the outer boundary, this means approximately that

$$2T_s < \Delta R/V_t$$

where V_t is the target velocity at the time it crosses the outer boundary. This inequality is not exact and is, in fact, rather conservative; in many cases, considerably larger values of T_s would be quite satisfactory.

When successful detection is accomplished, a verification pulse is ordered to be transmitted at time T_v after the successful detection pulse. The verification pulse uses the same beams and waveform as the search pulse, but the beam axis is along the commanded position of the beam axis for the successful search pulse. If the verification return is above the search-return threshold, verification is assumed to be accomplished and track initiation is begun. If the verification is unsuccessful, search continues at intervals of T_s after the previous "successful" detection.

If the search process were being explicitly modeled, with its raster scan, it would usually be the case that the target appears somewhere off the axes of the transmit and receive beams. This is another form of frame randomization, and it is modeled in ROSCOE by offsetting the axes of the

search-mode transmit and receive beams with respect to the target position. The amount and direction of offset are randomly generated in such a way that the target will be somewhere within the 3 dB beamwidths of these beams, allowing for refraction.

Refraction. The term "allowing for refraction" is interpreted in the following ways under the stated conditions:

1. When only one ray path exists. The ray path from the true target position to the radar is computed. The true azimuth and elevation at the radar are used for the apparent values A' and E' ; the apparent range, R' , is computed with allowance for both the added length of the curved ray path and the decreased value of c in plasma.

2. Multiple paths through a single fireball or stria. When multiple paths are present, one of them will always be the analytic continuation of the single ray-path occurring at large distances from the plasma. This ray-path is the one to use for both outgoing and incoming signals; its azimuth and elevation at the radar give A' and E' , and the apparent range along it gives R' ; this is called the "apparent position of the true target."

3. Multiple paths through a striated region. In ROSCOE, a "fuzz-ball" approximation is used (see Sec. 4.3 for further details), with R', A', E' being essentially the center of the apparent target image distribution which is the "fuzz-ball."

3.1.3 Flow Charts

Definitions. The flow charts refer to several sets of object position coordinates. In general, R, A, E refer to range, azimuth, and elevation with respect to the radar, and R, u, v refer to "sine-space" coordinates, where u and v are the sines of two orthogonal angles

measured from the radar boresight (see Appendix A). These coordinates are subscripted or primed to indicate the following:

R',A',E' or R',u',v'	Apparent target coordinates allowing for refraction but not including radar measurement errors
R_s,u_s,v_s	Measured coordinates from search return
R_v,u_v,v_v	Measured coordinates from verification return
$(u_o,v_o)_T$	Sine-space coordinates of transmit beam axis
$(u_o,v_o)_R$	Sine-space coordinates of receive beam axis

Other symbols used in the flow charts are:

θ	Angle between line-of-sight to target and boresight
acU(x,y)	Random number chosen from uniform distribution between x and y

Initialization (Fig. 3.2). The initialization procedure is not an event, but takes place before any actual events have occurred. Its purpose is to set up the first search event for each target. An implicit assumption is that search and verification are being performed by the radar, rather than by some other kind of sensor.

In the initialization procedure, the time for the first attempt at detection, t_o, is computed in vacuum, without reference to refractive effects. This of course will not give the strictly "correct" time, but it permits the search events to be time-ordered in advance without calling upon nuclear-effects and propagation subroutines. The time error involved is not likely ever to be significant in the context of program applications, and the approximation makes it possible to maintain a completely event-based structure.

Search (Fig. 3.3). The search event is first set up by the initialization procedure, and itself sets up the appropriate succeeding event.

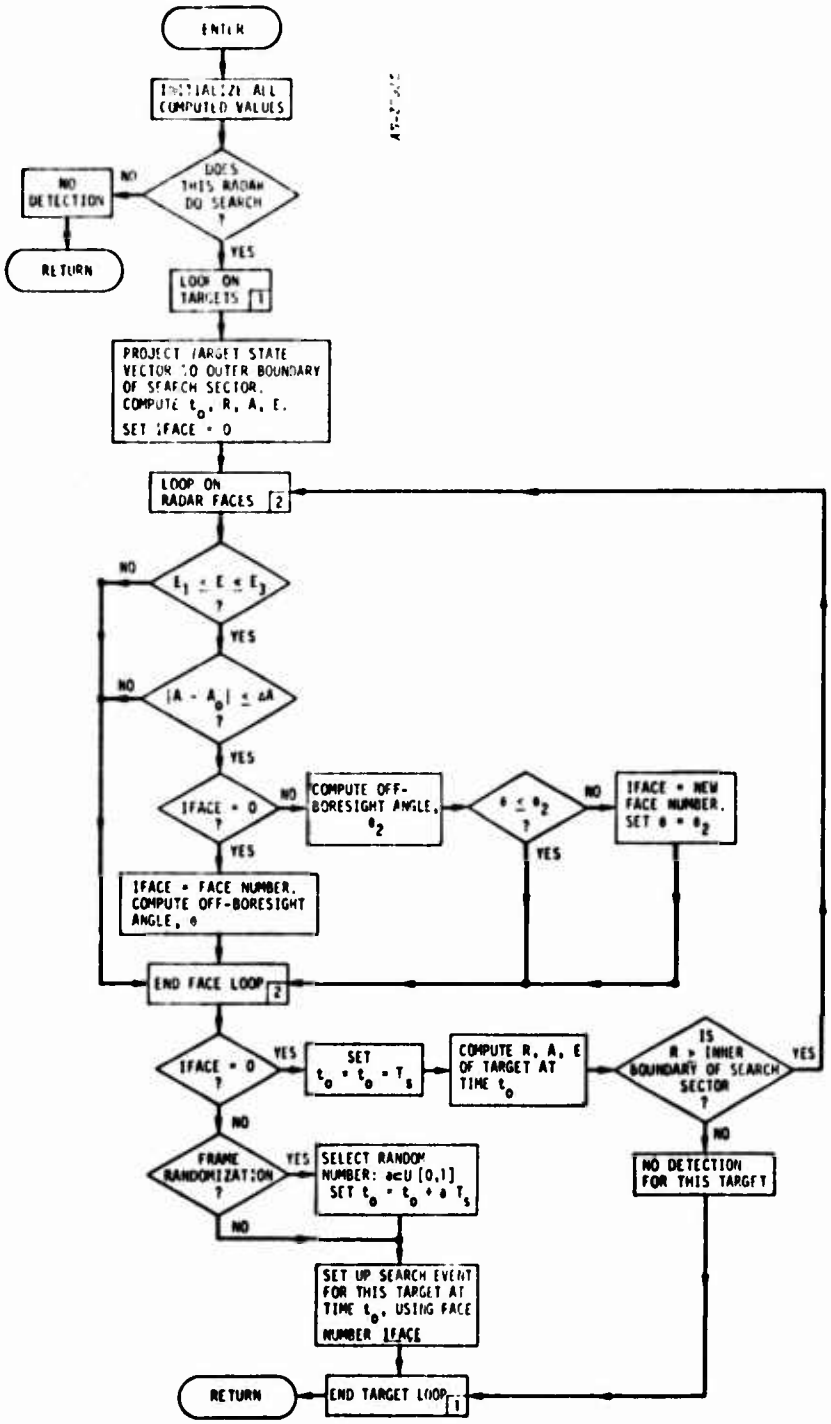


Figure 3.2. Search Initialization Flow Chart

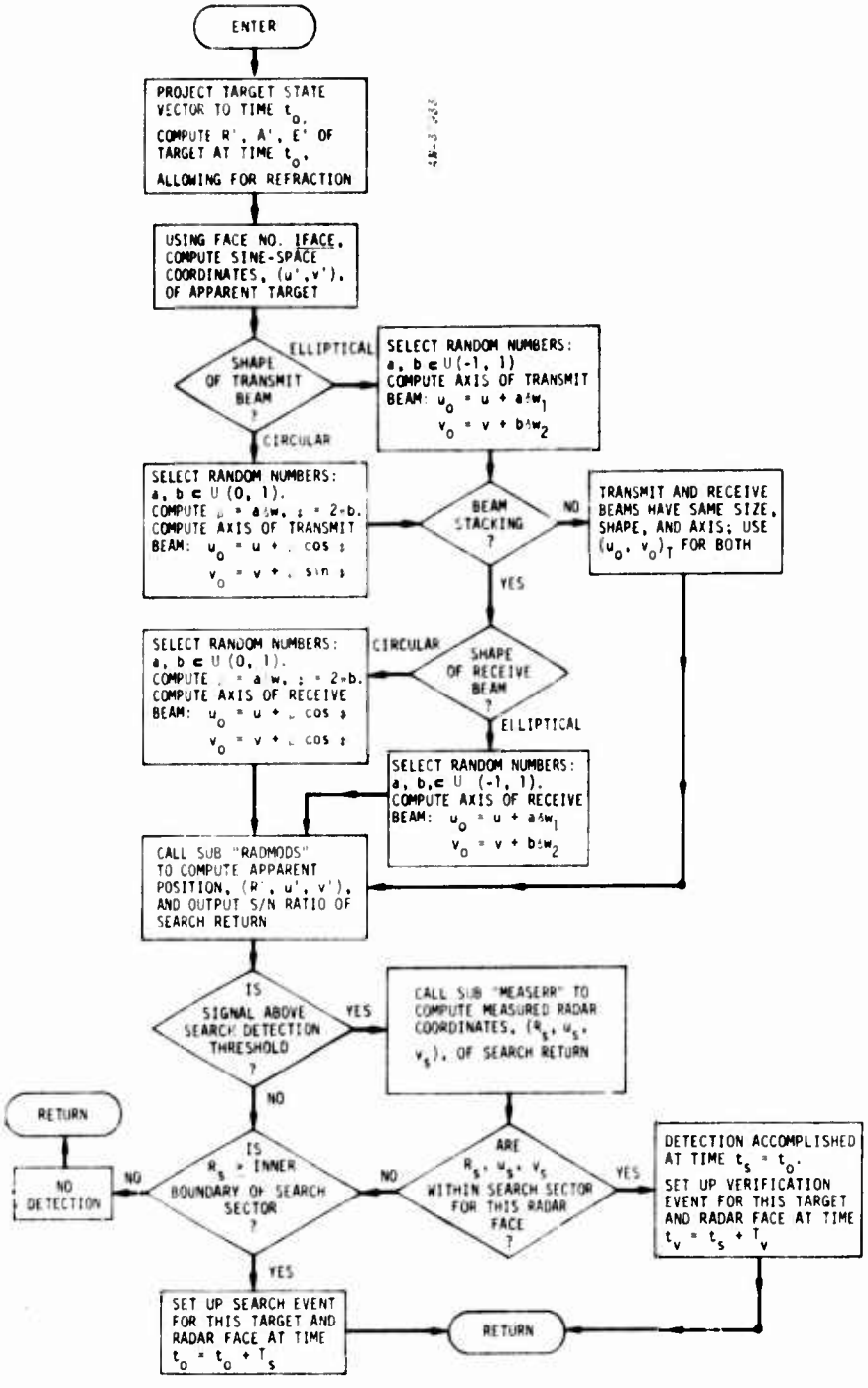


Figure 3.3. Search Flow Chart

Verification (Fig. 3.4). The verification event is set up by a successful search event.

3.2 TRACK INITIATION EVENT

3.2.1 Assumptions

Only one radar--the same one that was used for search--is used for track initiation.

Because of the event structure, any number of targets may pass (one at a time) through the track initiation procedure.

The track initiation procedure uses its own pre-designated waveform and beamshape, which may (but need not) be the same as those for the search event. For track initiation, the beam size, shape, and axis are the same for both transmit and receive beams. The waveform is usually chirped.

Monopulse is used in the track initiation procedure.

A general rule, applicable to this as well as all other radar events, is that each separate transmitted radar signal is a separate event. (A "signal" may, of course, consist of more than one pulse, provided that they are sent as a unit under a single command--e.g., pulse pairs or pulse bursts.) This event structure may appear to be less efficient than lumping together several transmissions which are a logical unit (such as the search and succeeding verification pulse, or the various track initiation pulses to be described below), but it avoids a possible problem arising from such lumped events. This problem would occur when an event related to a different target or radar should be scheduled for a time in the midst of the lumped period occupied by, for example, the five possible track initiation pulses. This would distort the proper time-ordering of events so necessary to the smooth operation of ROSCOE.

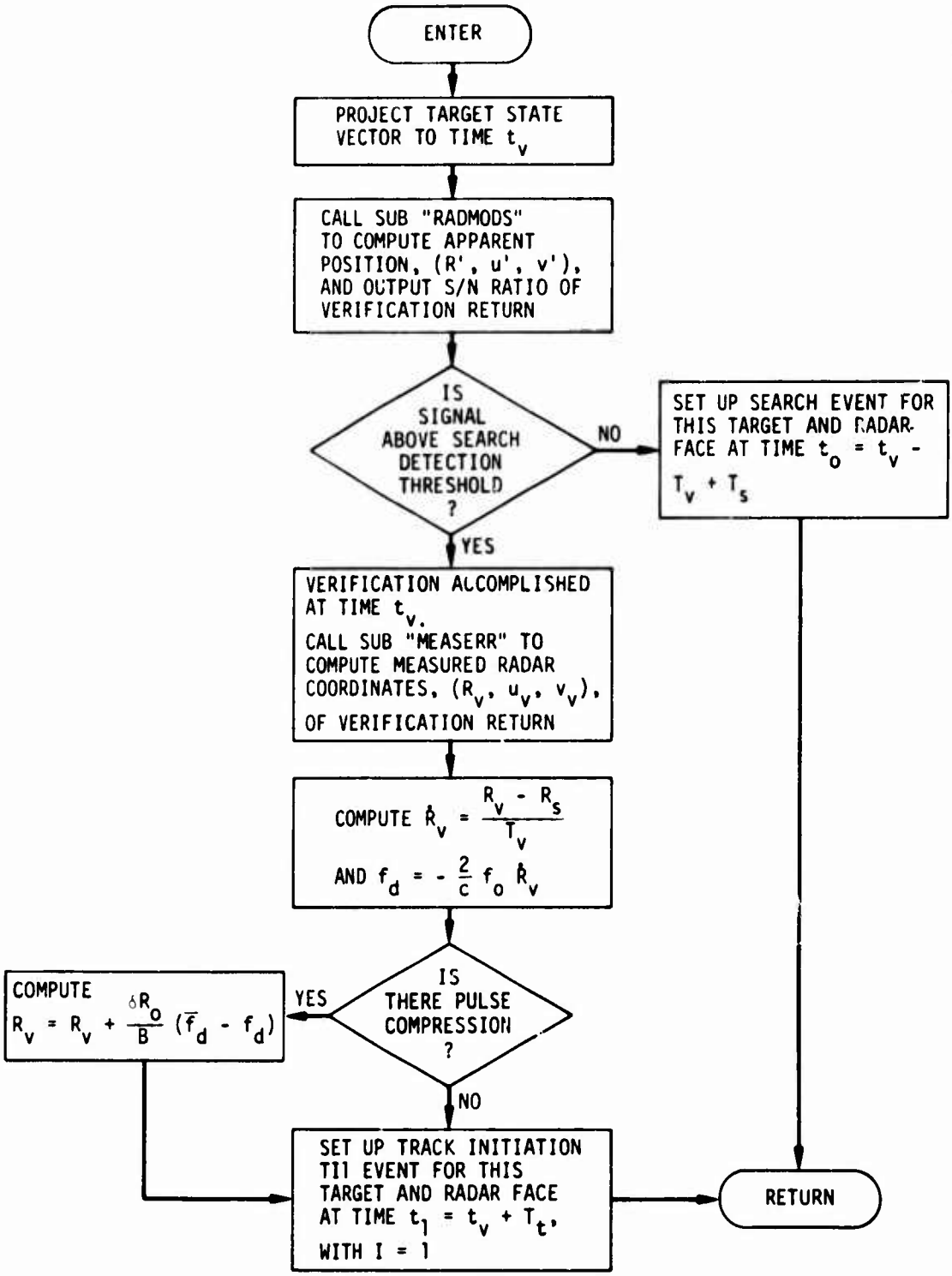


Figure 3.4. Verification Flow Chart

Another general rule is that each event is coincident in time with the instant of transmission of the relevant radar signal. The time-of-flight of the signal between the radar and the target is ignored.

3.2.2 Procedure

The first track initiation pulse is transmitted T_t seconds after the successful verification pulse, where T_t is the period between track pulses (an input quantity). Succeeding track initiation pulses will be transmitted at intervals of T_t, whether the first one was successful or not.

Two good track initiation returns are needed to start track. If the first track initiation pulse is not successful, a second attempt is made T_t seconds later. These two pulses are referred to as TI1 pulses. If neither of them is successful, the program reverts to the search process in the regular schedule.

If there is a successful TI1 return, another track initiation pulse is sent to the appropriate place (see below); this pulse is referred to as a TI2 pulse, although it is exactly the same in waveform and beamshape as the TI1 pulses. If a successful return is not obtained from the first TI2 pulse, a second is sent after T_t; if that is not successful, a third attempt is made after the same interval. At this point, failure to obtain a successful return causes the program to revert to the search procedure in the regular schedule. If there is a successful TI2 return, a track event is scheduled T_t seconds later.

Thus, under the best circumstances one TI1 pulse and one TI2 pulse are transmitted, whereupon the track process can begin. Under the worst circumstances, two TI1 pulses and three TI2 pulses could be transmitted before admitting defeat and reverting to the search procedure.

In a real system, the radar would continue to detect the object with search pulses as long as it remained within the search sector, even after track had commenced. Such detections would be compared with existing track files, and ignored when an association had been established. In ROSCOE, it is assumed that such an association process is always perfectly performed, and no additional search events are scheduled for an object once track initiation has been completed. In other words, possible errors in association leading to multiple track files on a single object are not modeled.

On the other hand, if two track initiation returns are not successfully completed, ROSCOE reverts to the search process.

The first TI1 pulse is transmitted in the direction of the commanded axis of the verification pulse, since (in the absence of monopulse for the verification procedure) that is the best estimate of the target's angular coordinates.

The receiver range gate for this pulse has width W_1 (an input quantity) and is centered on the estimated range

$$R_e = R_v + \left(\frac{R_v - R_s}{T_v}\right)T_t$$

where R_v = measured range from the verification return

R_s = measured range from the preceding successful search return

T_v = time interval between the search and verify pulses

T_t = time interval between the verify pulse and the first TI1 pulse

The second TI1 pulse, if needed, has its range gate centered at the estimated range

$$R_e = R_v + 2\left(\frac{R_v - R_s}{T_v}\right)T_t$$

The gate width remains the same.

The first TI2 pulse is transmitted in the direction indicated by the monopulse return of the successful TI1 pulse. The receive gate again has width W_1, and is centered at the estimated range

$$R_e = R_1 + \left(\frac{R_1 - R_s}{t_1 - t_s}\right)T_t$$

where R_1 = measured range from the successful TI1 pulse

R_s = measured range from the last successful search pulse

t_1 = transmission time of the successful TI1 pulse

t_s = transmission time of the last successful search pulse

If needed, the second and third TI2 pulses have range gates centered at

$$R_e = R_1 + \left(\frac{R_1 - R_s}{t_1 - t_s}\right)nT_t$$

respectively, where $n = 2$ or 3 as required.

In computing the return from track initiation and track pulses, all applicable object, radar, environmental, and propagation effects are taken into account. These include refraction, multipath, focusing or defocusing,

absorption, dispersion, noise, and clutter. The "signal-to-noise" ratio, S/N, referred to in several places in the flow charts, should more properly be called the "signal-to-noise-plus-clutter" ratio, $S/(N + C)$, since clutter is included; see Sec. 4.5.

Initializing the Tracking Filter. The tracking filter is initialized after a successful TI2 pulse. The state of the target as handed over to the filter is in the Geocentric Rectangular Cartesian (G.R.C.) coordinate system (see Appendix A). It consists of the following elements:

t_2 = time for which the target state was determined (transmission time of the successful TI2 pulse)

$\vec{T}$ = measured position vector of the target at time t_2, with components (T_1, T_2, T_3)

$\dot{\vec{T}}$ = calculated rate-of-change of target position at time t_2, with components $(\dot{T}_1, \dot{T}_2, \dot{T}_3)$

$\tilde{\sigma}_T$, $\tilde{\sigma}_{\dot{T}}$ = estimates of the measurement errors in position and velocity of the target at time t_2

It must be emphasized here that the values of $\tilde{\sigma}$ are decidedly not supposed to be the actual radar measurement errors computed by ROSCOE for the TI2 pulse, but rather are estimates which a system would make on the basis of available information. The computation of these estimates is discussed in Sec. 3.2.3.

Setting Up Discrimination Events. Once the tracking filter has been initialized and a track file has been opened on the target, it is possible to set up future discrimination events (see Sec. 3.4).

3.2.3 Details

3.2.3.1 Interactions with the Tracking Filter Subroutine

The tracking filter subroutine, FILTER, has a number of variant calling sequences depending on its application at the calling time. The

subroutine is assumed to know what the value of the calling time is; in the present case, it is t_2 . The calling sequence for the initialization call is

(PV, SIGMAT, DUMMY, DUMMY, DSPWRD, DUMMY)

The dataset pointer DSPWRD is set to zero, indicating that this is the initial call to the tracking filter for this target.

The Cartesian vector PV is a six-dimensional vector containing the position and velocity of the target at time t_2 , as measured and calculated from the two TI pulses and transformed into G.R.C. coordinates; i.e., the vector $(T_1, T_2, T_3, \dot{T}_1, \dot{T}_2, \dot{T}_3)$.

The covariance matrix SIGMAT is a 6×6 matrix which contains the initial estimate, in G.R.C. coordinates, of the errors in PV. For reasons that will be discussed in Sec. 3.2.3.2, only a rough estimate of the diagonal elements of this matrix can be made; the off-diagonal elements are not known at all and are set equal to zero. The components of SIGMAT are:

$$\text{SIGMAT}(I,J) = 0 \quad \text{for} \quad I \neq J$$

$$\text{SIGMAT}(1,1) = \text{SIGMAT}(2,2) = \text{SIGMAT}(3,3) = \tilde{\sigma}_T^2$$

$$\text{SIGMAT}(4,4) = \text{SIGMAT}(5,5) = \text{SIGMAT}(6,6) = \tilde{\sigma}_{\dot{T}}^2$$

The track initiation procedure also initializes two parameters that are maintained in the track event: TLAST, which indicates the time of the last successful track return for this radar/target combination, and ILT, which indicates the number of successive unsuccessful track pulses for this radar/target combination. This information is needed

in case of lost track on the target, when it is desired to keep a record of how long it has been since the target was last seen. Both parameters are initialized at zero by the track initiation procedure.

After the initial call to subroutine FILTER, the data needed for the track file to be established for this target has been prepared, and their location is indicated by the returned value of the DSP word DSPWRD.

Finally, a first track event is set up for this radar/target combination at time

$$t_3 = t_2 + T_t$$

The symbol "t_3" will be used consistently to represent the time of a track event.

3.2.3.2 System Estimates of Measurement Errors

The radar modeling in ROSCOE of course introduces measurement errors, both bias and random, in each of the radar coordinates. In calculating the values of these errors and applying them to the "apparent" coordinates to yield the "measured" coordinates, the ROSCOE model is playing the part of Nature, whose errors are completely known to herself but can only be imperfectly estimated by the radar system. This modeling of Nature's errors is a part of the radar measurements simulation, and will be fully discussed in Sec. 4.7.

But ROSCOE needs another kind of measurement error model also: this is the model which the radar system would employ in estimating, given imperfect information, what the measurement errors might be for any particular return. Here ROSCOE plays the part not of Nature but of the radar engineer. He has certain formulas, developed according to various theoretical models, for estimating the probable error inherent in a given measurement. Such estimated errors are used by the track filter to decide

how to weight each new measurement in comparison with the old ones. They are also needed for properly initializing the track filter. This section discusses a simple model which is used in ROSCOE for making these error estimates.

The track filter requires estimates of the measurement errors in each of the primary coordinates: R, u, v . Range or angular rates are not directly used in this filter, so their errors need not be estimated. Furthermore, since the filter uses them in the same coordinate system in which they are measured, there is no need to transform the error estimates into another coordinate system.

For track initiation, however, things are not so simple. Not only the position but also the velocity of the target is needed to start off the tracking filter, and these must be expressed in the G.R.C. coordinate system (see Appendix A). The corresponding error estimates must therefore also be calculated and transformed into that frame. Procedures for doing this are described below.

Position Error Estimates. Three types of errors are usually distinguished in each measured coordinate: a bias error, a jitter error component which is independent of signal-to-noise ratio, and a jitter error component which is signal-to-noise dependent. The bias error is represented in the "Nature's model" part of ROSCOE, but cannot be estimated by the system (if it could, being constant, it would be removed). The jitter error in each radar coordinate is modeled as being Gaussian distributed with zero mean and a standard deviation, $\tilde{\sigma}$, of the general form

$$\tilde{\sigma}^2 = \tilde{\sigma}_1^2 + \tilde{\sigma}_2^2$$

where $\tilde{\sigma}_1$ is a constant and $\tilde{\sigma}_2 = \frac{K}{\sqrt{S/N}}$

Here K is a constant, dependent on the beamshape or waveform parameters, which is given in all standard texts on radar theory.

Although there are a number of good theoretical formulas for the constant K, there are no general theoretical models for the constant $\tilde{\sigma}_1$, because this is supposed to be dependent on specific mechanical and electronic limitations in the radar itself. However, it has been observed[4] that in many existing or planned radars the ratio $K/\tilde{\sigma}_1$ falls within relatively narrow limits (possibly because the designers all use the best current state-of-the-art). By using a mean value for this ratio, γ (which is different for linear and angular coordinates), $\tilde{\sigma}_1$ can be estimated given K. The resulting formula for $\tilde{\sigma}$ accordingly takes the form

$$\tilde{\sigma}^2 \simeq K^2 \left(\frac{1}{\gamma^2} + \frac{1}{S/N} \right)$$

The constant K is usually quoted in terms of the altazimuth coordinate system, i.e., in range and angle. The angular error, however, then depends on the off-boresight angle and the consequent effective beam broadening as well as beamshape distortion. These effects do not arise in the sine-space (R, u, v) coordinate system, which is one of the main reasons why ROSCOE uses it. In R, u, v coordinates, the following values of K are used:

$$K_R = \frac{c}{\pi B}$$

$$K_u = \delta w_1$$

$$K_v = \delta w_2$$

where c = speed of light in vacuum

B = effective signal bandwidth

$\delta w_i = \sin(\delta\theta_i/2)$

$\delta\theta_i$ = one-way 3-dB beamwidths of receive beam in the u and v directions

Velocity Error Estimates. The velocities $\dot{R}$, $\dot{u}$, $\dot{v}$ are determined for track initiation purposes by taking differences of the measurements made with the TI1 and TI2 pulses, as previously discussed.

The error in computing the rate-of-change of a measured coordinate from a series of n independent measurements of that coordinate has been estimated by Manasse.[4] Although his derivation is not strictly applicable to the present case where $n = 2$, ROSCOE uses a modified form of his result and allows for the smallness of n by dropping the factor $n^{-1/2}$ from his equation. The result is

$$\tilde{\sigma}_{\dot{p}} \simeq \frac{2\sqrt{3}}{T} \tilde{\sigma}_p$$

where p is the coordinate involved (R, u, or v as the case may be), T is the time interval between the two measuring pulses, and $\tilde{\sigma}_p$ is the estimated error in measurement of p. Accordingly, ROSCOE uses

$$\tilde{\sigma}_{\dot{R}} \simeq \frac{2\sqrt{3}}{T_t} \tilde{\sigma}_R$$

$$\tilde{\sigma}_{\dot{u}} \simeq \frac{2\sqrt{3}}{T_t} \tilde{\sigma}_u$$

$$\tilde{\sigma}_{\dot{v}} \simeq \frac{2\sqrt{3}}{T_t} \tilde{\sigma}_v$$

where T_t, as before, is the interval between track initiation pulses.

Transformation to G.R.C. Coordinates. It is not in general possible to make a meaningful transformation of error sigmas from one coordinate system to another when the coordinate transformation is nonlinear. The actual error ellipsoid remains the same, of course, so in effect the error sigmas in the transformed coordinates can have any values between the smallest and the largest linear radar measurement errors. The obvious simple and safe course is to make the most conservative estimate, which is to assume that the error sigmas in G.R.C. coordinates (W_1, W_2, W_3) are all the same and equal to the largest linear measurement sigma (and similarly for the velocities). This leads to the following expressions:

$$\tilde{\sigma}_{W_1} = \tilde{\sigma}_{W_2} = \tilde{\sigma}_{W_3} \simeq \max\left[\tilde{\sigma}_R, R\tilde{\sigma}_u, R\tilde{\sigma}_v\right]$$

$$\tilde{\sigma}_{\dot{W}_1} = \tilde{\sigma}_{\dot{W}_2} = \tilde{\sigma}_{\dot{W}_3} \simeq \max\left[\tilde{\sigma}_{\dot{R}}, R\tilde{\sigma}_{\dot{u}}, R\tilde{\sigma}_{\dot{v}}\right]$$

Thus only the diagonal elements of the 6×6 error covariance matrix can be estimated, and those only roughly. The off-diagonal elements are not even guessed at, but are set equal to zero.

3.2.4 Flow Charts

Flow charts for the TI1 and TI2 events are given in Figs. 3.5 and 3.6.

3.3 TRACK EVENT

3.3.1 Assumptions

3.3.1.1 Logical Flow

ROSCOE generally implements a simplified, straight-line logical flow from one procedure to the next. Only one radar face does Search/Detection/Verification; if the radar has more than one face, the S/D/V initialization routine selects for each target the radar face which

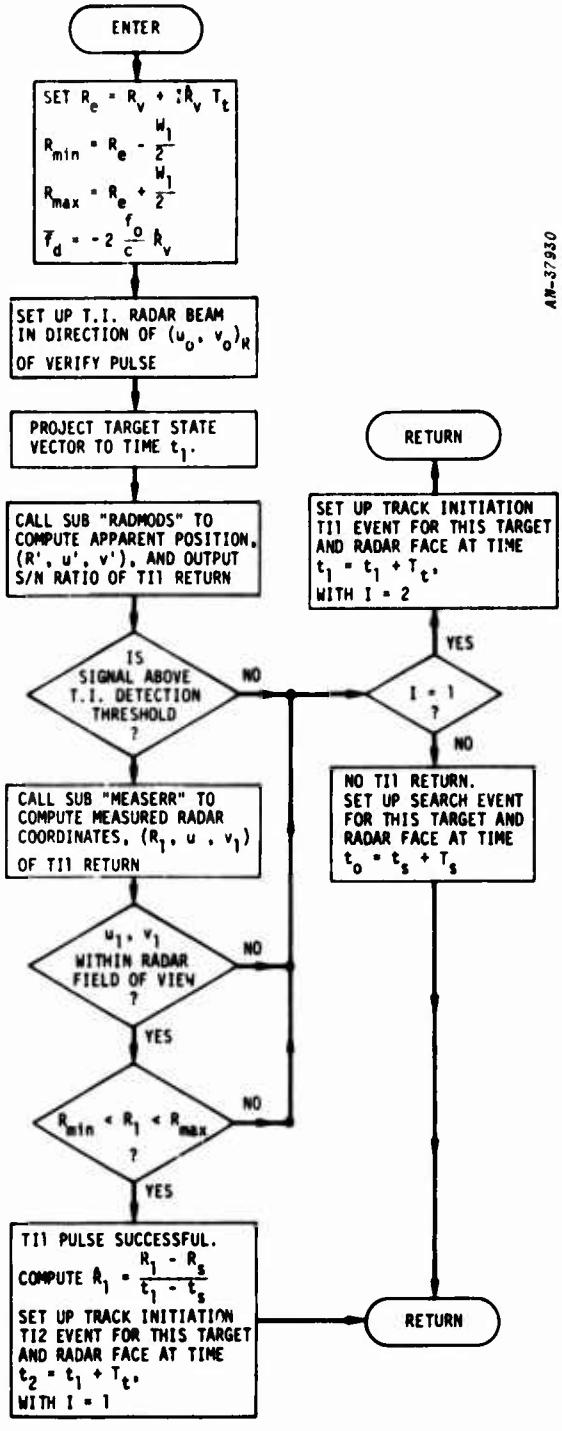


Figure 3.5. Track Initiation (TI1 Pulses) Flow Chart

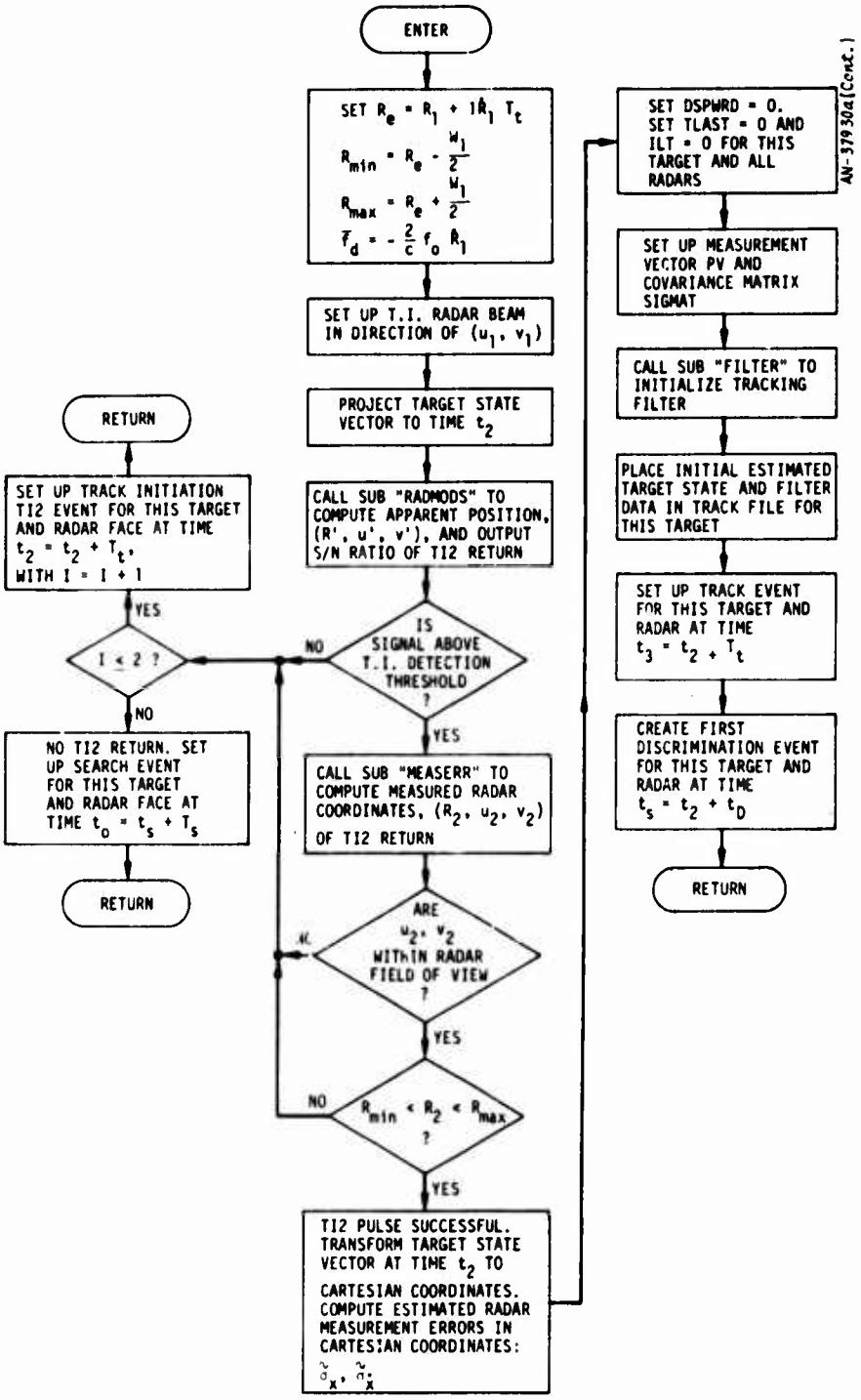


Figure 3.6. Track Initiation (TI2 Pulses) Flow Chart

would detect it first. The user can also control which radar face does the initial detection by specifying that only one of them can search.

The logic next assumes that the same radar that did the initial detection also performs the track initiation on the detected target. This is a realistic assumption, provided that one is only concerned with targets for which no track file has been established until detected by a radar. Ultimately, however, ROSCOE will have to consider situations in which initial detection and track initiation were performed by optical sensors before any radar detection could take place. The radar logic would then have to bypass the radar search, verification, and track initiation routines and start right in with tracking on this target whose track file has already been established. This would constitute a "multiple-origin," or non-straight-line logical flow. The system here described does not yet handle such a situation.

3.3.1.2 The Tracking Filter

Tracking filters, in general, can suffer from stability problems; filters which work well in a normal environment can become unstable in a nuclear-disturbed environment. In such situations, certain filter parameters may have to be "tuned" in order to get a filter to perform reasonably well. Or one can turn off the filter (i.e., refuse to accept further data) for a certain period after a nuclear burst in the vicinity of the line-of-sight. Or one can allow the filter time constant and/or data weighting factors to be continuously adjustable parameters in response to some observable which (it is hoped) is correlated with the degree of environmental degradation being encountered. Ultimately, one may attempt to design special tracking filters for use in refractive environments--for example, filters which include refraction error parameters among their fitted variables.

All of these schemes, and others, have been under active consideration in the community. Some users may wish to use ROSCOE as a vehicle to

test out the effectiveness of their own filters. If so, the modular arrangement of ROSCOE will permit them simply to substitute their filter for the standard ROSCOE filter with a minimum of difficulty. But it must be clearly understood that the standard ROSCOE filter is not intended to be general enough to permit all these contemplated adjustments. The ROSCOE filter is a more-or-less standard Kalman filter, with only such adjustable parameters as are described below.

Only the requirements on, and the external features of, the track filter are given in this subsection; the internal operational details of the filter were discussed in Sec. 2.4.

Filter Inputs. ROSCOE's filter is a Kalman filter operating with seven target state variables: position, velocity, and ballistic coefficient. The input data may be two-dimensional measurements (e.g., angles-only from an optical sensor), three-dimensional measurements (e.g., R, u, v from a radar), four-dimensional measurements (e.g., R, u, v, and $\dot{R}$), or whatever is available. Any convenient measurement coordinate system can be used; its nature must be known to the sensor but need not be known to the filter. It is necessary, however, that the sensor module know how to transform a predicted target position given in G.R.C. coordinates into the measurement system. The present discussion assumes that all radar measurements are made in radar face coordinates (R, u, v).

In the filter, the input measurements are weighted in accordance with their estimated error sigmas, which must also be input to the filter. Section 3.2.3.2 explains how to calculate $\tilde{\sigma}_R$, $\tilde{\sigma}_u$, and $\tilde{\sigma}_v$.

The time at which the measurement was made must also be an input.

Filter Outputs. The filter outputs needed by the track event routines are:

- The estimated target state vector at the time of the measurement, in G.R.C. coordinates.

- The filter's current estimates of the errors in these coordinates, derived from the filter's internal covariance matrix.

Filter Memory Decay. It is convenient, though not absolutely necessary, that the filter has an exponential memory decay. Without such decay, older data may be given too much weight in comparison with new data, leading to unnecessarily prolonged propagation of bias or start-up errors, errors resulting from temporary refractive effects, or errors due to imperfectly known atmospheric models and ballistic coefficients. On the other hand, if the decay time is too short the target may be lost during prolonged periods of missing returns (such as those due to blackout) or refractive errors. Thus the filter decay constant τ (if present) is one of the more sensitive parameters in terms of which the filter can be "tuned" to function best under given conditions.

The following two-stage scheme is used for defining the filter decay time constant: the user inputs an altitude, H_τ, and two values of the decay constant, τ_1 and τ_2. The filter uses

$$\tau = \begin{cases} \tau_1 & \text{when } H_p \geq H_\tau \\ \tau_2 & \text{when } H_p < H_\tau \end{cases}$$

where H_p is the target altitude as determined from the filter output (not from raw measured data).

Note that the user can set $H_\tau = 0$ and $\tau_1 = \infty$ (i.e., some very large number) if he wishes to use no filter decay. Similarly, setting $H_\tau = 0$ permits the user to specify only a single decay constant for the entire track. It might also be pointed out that the above decay constant definition scheme is for the filter itself and hence applies uniformly to all radars in the system.

Other Adjustments. The standard ROSCOE filter weights input data in accordance with the estimated measurement error sigmas, but does not apply any sort of special weighting based on measured or assumed environmental conditions.

ROSCOE does not have any provision for "turning off" the filter in response to nuclear bursts: neither an "exclusion time" nor an "exclusion volume" nor any other such process is employed.

3.3.2 Procedure

3.3.2.1 Multiple Radars

In a real system, several radars may detect the same target during search. Each one will start a separate track file on it, but presumably the system logic will at some point make the appropriate associations and coalesce them all into a single track file. Thereafter all radar data on this target would go into the same tracking filter file, regardless of which radar took the measurements.

Environmental effects could certainly have an adverse effect on this association process, probably resulting in the prolonged maintenance of multiple track files on a single target. But although this is indeed an interesting problem and one related to nuclear effects, its solution lies in the realm of system logic and is outside the purview of ROSCOE. Accordingly, ROSCOE ignores the track association problem and maintains a separate track file for each radar/object pair.

Other radars (if present) may join in the track process, all maintaining separate track files. The track event logic is structured in such a way that any radar may be making the present measurement, according to a previously-set-up track event which told it where and when to transmit its signal. The logic also schedules the next track event for this radar/target combination.

3.3.2.2 Track Cut-Off Range

Some radars have a minimum range inside of which they cannot track a target. This option is included in the model: the user specifies R_{tmin} for each radar. Then, as soon as the range of a target (as obtained from the tracking filter projection) is less than R_{tmin}, no further track events are scheduled for that radar on that target.

3.3.2.3 Tracking Range Gates

A real radar system listens for a return only during some specified time interval spanning the estimated time for the track pulse return. This interval is called the range gate, and is usually expressed in range units rather than time units. It needs to have sufficient width to achieve a high probability of receiving the target return, yet be small enough to eliminate many extraneous echoes and minimize the amount of time the radar has to spend in listening. To facilitate this, the width of the range gate is usually given as a function of the estimated range variance as obtained from the tracking filter.

Before the filter has been started (i.e., during track initiation), the range gate width is a specified input value, W_1. Thereafter, the range gate is defined as follows. The user inputs parameters k_1 and k_2 for each radar. On the basis of all previous track pulses, an estimated (Cartesian) state vector for the target has been computed by the tracking filter, along with an estimate of the covariance matrix of the state parameters. These are projected ahead to the time t_o for the current track event, and the predicted range R_p and range variance $\sigma^2_{R_p}$ are calculated for t_o. Then the track return is listened for in the range window

$$R_p - \frac{W_2}{2} < R < R_p + \frac{W_2}{2}$$

where

$$W_2 = k_1 + k_2 \sigma_{R_p}$$

If the measured range, after allowing for refraction and noise errors, does not fall within this window, the return is recorded as "not received."

3.3.2.4 Lost Targets

In a real radar system, it may happen that a number of consecutive track pulses fail to produce a return--a "lost track." If all radars that were tracking the target lost track on it, then the system is said to have "lost target." In this event, a real radar system will initiate some special course of action (such as a limited search scan about the predicted position) in an attempt to recover the target. Such system logic is not a proper concern for ROSCOE, and such specific responses are not modeled.

In ROSCOE, the following procedure is used. If no successful track returns have been received on a given target for a time period ΔT_{LT} (an input parameter), the target is assumed to be "lost" and a new search pulse is scheduled. These will continue until either the target is reacquired or it ceases to exist (impacts or detonates).

The method described above gives the user a step-by-step picture of what the nuclear environment is doing to his target tracking attempts.

3.3.3 Definitions

(R,u,v) represent true vacuum coordinates of the target relative to the radar face, with no propagation effects or radar measurement errors included. They are computed by transforming the true target state vector from G.R.C. to radar face coordinates.

(R',u',v') represent the apparent target coordinates relative to the radar face, allowing for all propagation effects but not including radar measurement errors. They are computed by applying appropriate propagation interactions to (R,u,v) and then calculating what the radar "thinks" it sees, taking into account the beamshape and the commanded position of the beam axis.

(R_m, u_m, v_m) represent the measured target coordinates relative to the radar face, including all radar measurement errors. They are computed from (R', u', v') by taking into account the current value of the signal-to-noise-plus-clutter ratio.

(R_p, u_p, v_p) represent the predicted target coordinates relative to the radar face, calculated from the estimated target state vector from the track file, updated to time t_3. The calculation includes no propagation effects or radar measurement errors; it is made by transforming the updated estimated state vector from G.R.C. to radar face coordinates.

$(\bar{R}, \bar{u}, \bar{v})$ represent the "expected" target coordinates relative to the radar face, for use by the track filter. After the filter has absorbed the current measurements (R_m, u_m, v_m) it returns a Cartesian state vector CARTEZ giving its estimate of where the target should now be; $(\bar{R}, \bar{u}, \bar{v})$ are calculated directly from this vector with no propagation effects or radar measurement errors included, as above. This follows the procedure laid down for using the track filter; the coordinates $(\bar{R}, \bar{u}, \bar{v})$ are fed back into the filter to be compared with (R_m, u_m, v_m) in order to compute the next iteration of the filter variables. This may have to be done several times before the filter is satisfied with the result. For further details, see below, and also Sec. 2.4.

3.3.4 Inputs

The following data are required, either as inputs or in datasets, for the track event procedure:

- The radar identification (IR) and location
- The radar face number (IFACE)
- The target identification (IT)
- The transmission time of this track pulse (t_3)
- Dataset pointers to
    - The true trajectory, ballistic coefficient, and radar cross section parameters of the target

- The track filter estimates of the target trajectory and ballistic coefficient parameters
- The current track filter data
- The radar waveform and beamshape parameters for this radar
- The boresight vectors for all faces of this radar, and their fields-of-view

- The track detection threshold (K_T)
- The minimum tracking range for this radar (R_{tmin})
- Parameters for computing the range gate width (k_1,k_2)
- The current values of the parameters TLAST(IR,IT) and ILT(IR,IT) for this radar/target combination
- Decay-time parameters (τ_1,τ_2,H_τ) for the track filter
- The time period between successive track pulses on a given target (T_t)
- The lost-target parameter for this system (ΔT_{LT})
- Flags to indicate
  - That this target has been "killed"
  - That this radar cannot see this target with any of its faces

3.3.5 Outputs

Possible outputs of this track event procedure are:

- An updated track file on this target and the scheduling of the next track event for this radar/target combination
- The scheduling of the next track event for this radar/target combination without updating the track file on this target
- Indication that this radar has lost track on this target

3.3.6 Logical Outline of the Flow Charts

3.3.6.1 Basic Procedure (Fig. 3.7)

The basic flow through this event, assuming a successful track return, is found in Boxes 1 through 24.

If the target has been "killed" between the time this event was scheduled and the time the track pulse was to be transmitted, the procedure is aborted, no pulse is sent, and no new track event for this target and radar is set up (Box 1).

If the predicted target cannot be seen within the fields-of-view of any of the radar's faces, no pulse is sent at this time (Boxes 25 through 30), and the procedure goes into the "no return" mode (Boxes 32 through 38; see below).

If the predicted target range is less than the minimum tracking range (if any) for this radar, no pulse is sent and no future pulses are scheduled; a "dropped track" flag is set (Box 31).

If a pulse is transmitted but no successful return is received, either because the return did not exceed the track threshold (Box 11) or because the measured return (including refraction and errors) fell outside the field-of-view (Box 13) or because the measured range fell outside the receive range gate (Box 14), the procedure goes into the "no return" mode (Boxes 32 through 38; see below).

The interaction between this event and the track filter subroutine is represented in Boxes 16 through 23. This procedure is described more fully below.

When no return is received, for one reason or another, the logical flow branches to the column headed by the symbol Ⓐ (Boxes 32 through

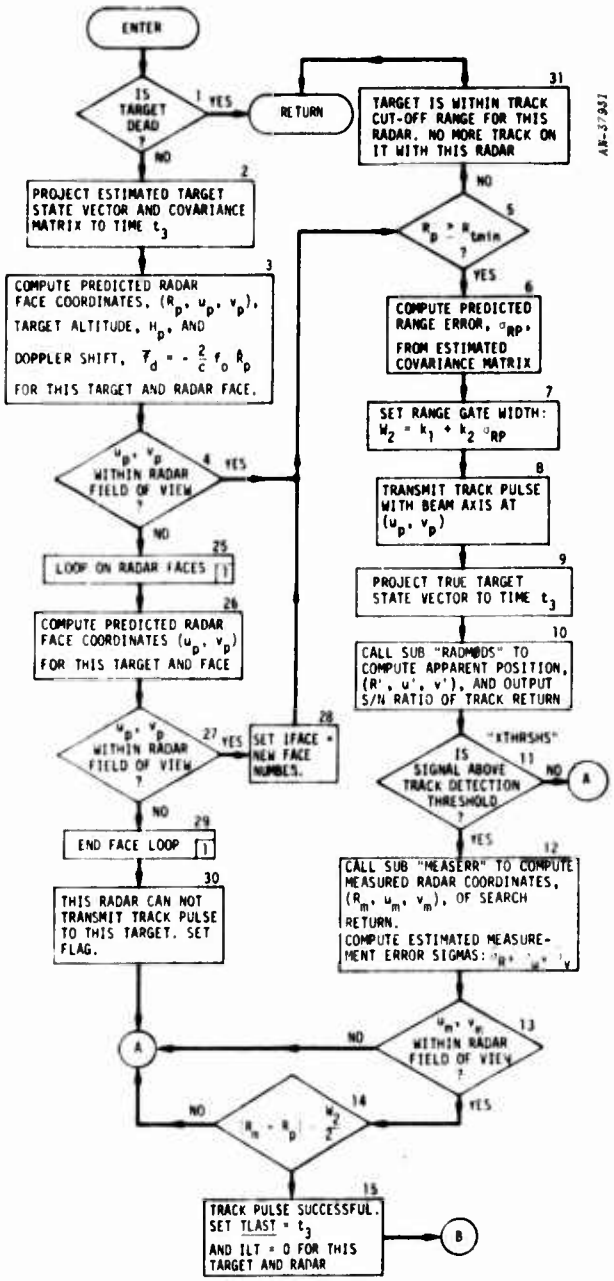


Figure 3.7. Track Flow Chart

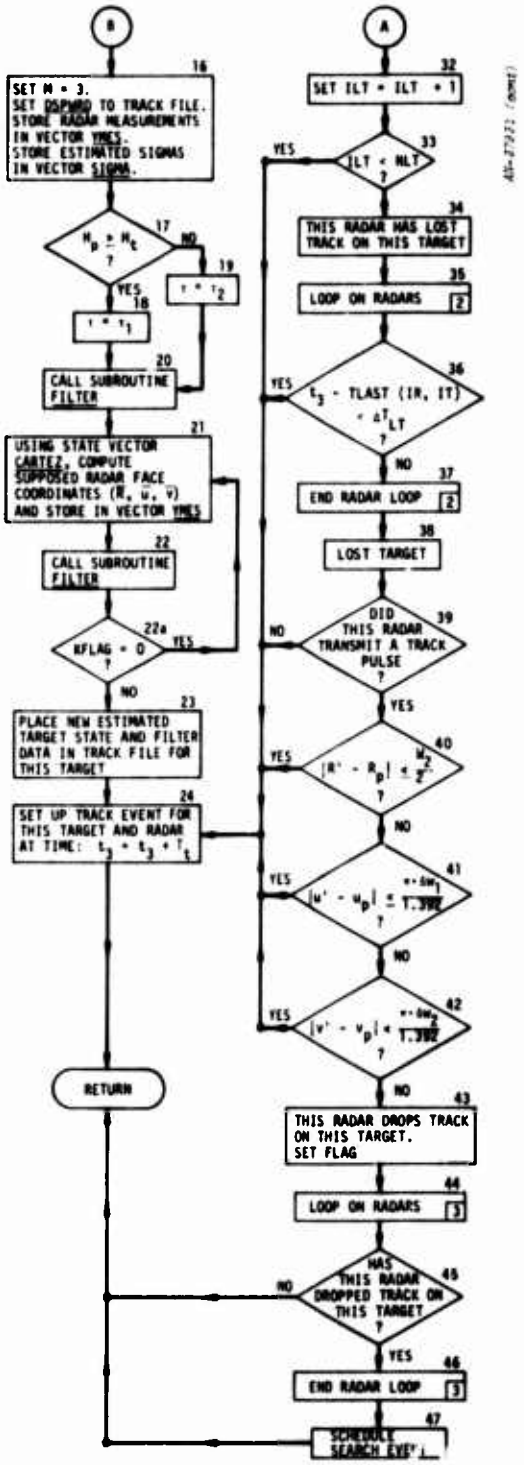


Figure 3.7 (Cont.)

38). This is the "no-return" mode. It decides whether to define a "lost target" state, then continues to attempt track as previously described.

3.3.6.2 Interactions with the Tracking Filter

Data Set Pointer DSPWRD. DSPWRD, set in Box 16, is the DSP word for the track file into which the current measurement is to be incorporated. It must not be zero, since the FILTER routine will assume this to be an initialization call if zero.

Integer M (Box 16). The integer M is the dimensionality of the measurement vector; it is always 3 for radar measurements in the present ROSCOE.

Vector YMES (Box 16). Vector YMES is the three-vector giving the current radar measurement in radar face coordinates (R_m, u_m, v_m).

Vector SIGMA (Box 16). Vector SIGMA is the three-vector giving the estimated radar measurement errors $(\tilde{\sigma}_R, \tilde{\sigma}_u, \tilde{\sigma}_v)$ associated with the current measurement. These are computed (in Box 12) in radar face coordinates and stored in SIGMA.

First Call to Subroutine FILTER (Box 20). After setting up the input variables as described above, and setting the value of τ, subroutine FILTER is called. It returns an "expected state vector" CARTEZ and an integer KFLAG, which are used as described below.

Integer KFLAG (Box 22a). The integer KFLAG indicates when the FILTER subroutine is finished absorbing the current measurement. As long as it is returned by the subroutine with the value zero, a new iteration of the "expected" state vector (CARTEZ) is being returned to be converted into radar face coordinates and fed back into FILTER. When the filter is satisfied with the current iteration it feeds back

KFLAG = 1 , whereupon the current estimated state vector is ready to be put into the track file (Box 23). Several iterations may be required.

Vector CARTEZ (Box 21). Vector CARTEZ, a standard 10-dimensional state vector in G.R.C. coordinates, is returned by subroutine FILTER. It represents the current iteration's estimate of the target state vector, which must be compared with the current radar measurement in order to perform the next iteration. Since the FILTER subroutine does not know the position of the radar, this vector must be converted by the radar module into a form matching that of the measurements. This is done by letting

$$W(I) = CARTEZ(I + 1),\ I = 1,2,3$$

be the expected target position vector in G.R.C. coordinates, and converting to radar face coordinates. The resulting radar face coordinates $(\bar{R},\bar{u},\bar{v})$ are stored in vector YMES, and subroutine FILTER is called again (Box 22).

3.4 DISCRIMINATION EVENT

3.4.1 Assumptions

1. Only one radar--the same one that was used for search and track initiation--is used for discrimination.
2. The initial discrimination event is created by the track initiation event and thus follows it in the time ordered processing of events.
3. Subsequent discrimination events are created in Subroutine DISCRIM (see flow diagrams) at time intervals t_D over a period T_{ID} (input quantities).
4. The discrimination process will proceed independently of the other radar events once it has been initialized.

5. Here, as in the other radar events, the transmission time of the signal is ignored; the transmission and reception of the signal occur at the event time.

3.4.2 Procedure

Two types of discriminants are available for simulation in ROSCOE: Fine-Frequency Length and Wide-Bandwidth Length.

The Fine-Frequency Length discriminant involves making measurements of the target's body length by using a special waveform. For reasons which will not be discussed here, this measurement cannot be made if the target is below a specified altitude. While the actual limiting altitude depends somewhat on the nature of the target, which cannot be assumed to be known, it is possible to define an altitude H_{FF} above which the measurement can be made for all targets. This quantity is an input to the program.

The DISCRIM routine first checks whether the user has specified the use of this discriminant. If so, the target state vector is projected ahead to the current time for this event and the target altitude, H, is computed. Then, if $H > H_{FF}$, the Fine-Frequency Length discrimination is performed. If $H \leq H_{FF}$, this measurement cannot be made.

Subroutine DISCRIM then checks whether the user has specified the use of the Wide-Bandwidth Length measurement (independent of whether a Fine-Frequency Length measurement has been made). This discriminant also requires a special waveform, and cannot be performed if the target is too far away. Once again, although the limiting range depends somewhat on the target and radar characteristics, a value of R_{WB} can be defined in the input such that a measurement can be made whenever the target's slant range is less than R_{WB}. If the user has specified this discriminant, the target range at the current time is computed and compared with R_{WB}. If $R < R_{WB}$, the Wide-Bandwidth Length calculation is made and a summary output dataset created and put on the output list.

Finally, the total elapsed discrimination time (T_T) is computed and checked against the input time limit (T_{ID}). If the elapsed time has not exceeded the limit, a new discrimination event is created at a time interval t_D in the future. If the limit has been reached, no new discrimination events will be set up.

Figure 3.8 is a flow diagram of the discrimination event. The following subsections describe the specific models used to implement the Fine-Frequency Length and Wide-Bandwidth Length discriminants.

3.4.3 Fine-Frequency Length Model

If we imagine a simple "dumbbell" target with point scatterers of cross section σ_f and σ_r at the front and rear, then the total radar cross section is given by

$$\sigma_{net} = \sigma_f + \sigma_r + 2\sqrt{\sigma_f \sigma_r}\cos\left(\frac{4\pi f}{c} L|\cos\theta|\right)$$

where L is the length of the body, θ is the look angle, f is the radar frequency, and c is the speed of light.

If the cross section is measured at four frequencies, f_1, f_2, f_3, f_4, such that $f_2 = f_1 + \Delta f$, $f_3 = f_1 + 2\Delta f$, and $f_4 = f_1 + 3\Delta f$, then one can calculate the quantity

$$K = \frac{\sigma_4 - \sigma_1}{\sigma_3 - \sigma_2} = \frac{\sin 3\left[\frac{2\pi\Delta f L|\cos\theta|}{c}\right]}{\sin\left[\frac{2\pi\Delta f L|\cos\theta|}{c}\right]}$$

which reduces to

$$K = 3 - 4\sin^2\left[\frac{2\pi\Delta f|L\cos\theta|}{c}\right]$$

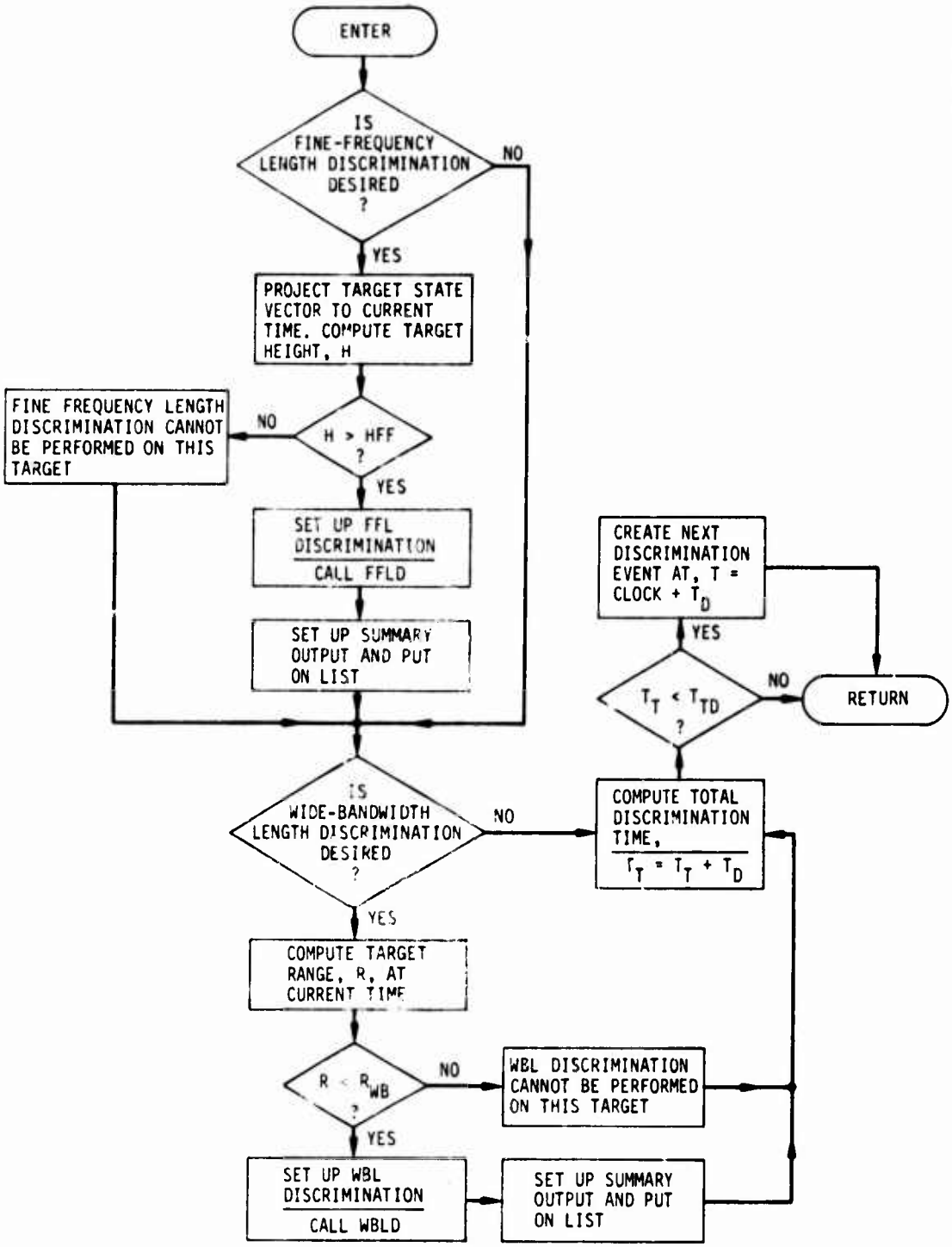


Figure 3.8. Discrimination Flow Chart

This can be solved for the body length, L :

$$L = \frac{c}{2\pi\Delta f\,|\cos\,\theta|}\,\sin^{-1}\left[\frac{3 - K}{4}\right]^{1/2}$$

3.4.4 Wide-Bandwidth Length Model

If a body is illuminated by a short pulse, the amplitude of the reflected signal, f(t), can provide a measure of the length. Specifically, twice the standard deviation of f(t) , i.e.,

$$T \equiv 2(\overline{f^2} - \bar{f}^2)^{1/2}$$

$$\equiv 2\sqrt{\frac{\int t^2 f(t)\,dt}{\int f(t)\,dt} - \left(\frac{\int t f(t)\,dt}{\int f(t)\,dt}\right)^2}$$

is called the pulse stretching in this method, and the length is approximated as cT/2.

As an example consider a target with front and rear point scatterers of cross sections σ_f and σ_r , separated by a distance L . If the inherent spread of the radar pulse is such that the return from a perfect point scatterer is

$$f(t) \propto e^{-t^2/2\sigma^2}$$

then the above approximation will give a "length"

$$\text{"L"} = L \times \frac{2(\sigma_f\sigma_r)^{1/4}}{\sqrt{\sigma_f} + \sqrt{\sigma_r}}\left[1 + \frac{\left(\sqrt{\sigma_f} + \sqrt{\sigma_r}\right)^2}{4\sqrt{\sigma_f\sigma_r}}\left(\frac{c\sigma}{L}\right)^2\right]^{1/2}$$

The true length L is modified by two factors, the first an error caused by the fact that the two scatterers have different cross sections and the second caused by the non-zero variance of the radar pulse.

If the radar waveform has no inherent spread (i.e., the variance $\sigma^2 = 0$), the square root term goes away, but the estimated L is still slightly in error by a factor

$$\frac{2(\sigma_f \sigma_r)^{1/4}}{\sqrt{\sigma_f} + \sqrt{\sigma_r}}$$

If the two scatterers are equal, this factor is unity and there is no error. If they are unequal it differs from 1, but only very weakly. If one scatterer has a cross section 10 dB larger than the other the error is only 15%; indeed, if one cross section is 30 times the other (15 dB difference), the error is only about 28%. It is doubtful whether cross sections with that great a dB difference could be measured anyway.

Because of the finite bandwidth of the radar signal the variance is not zero, but as long as the inherent width of the pulse ($c\sigma$) is much less than the target spread (L), the term in the square root is negligible. If plasma dispersion increases the variance so that $c\sigma$ is on the same order as L (or greater), the measured length "L" depends on the length of the pulse rather than the target characteristics; if $\sigma c >> L$, then "L" = $c\sigma$. In this case, a short decoy would appear to be as long as an RV.

3.4.5 Flow Diagrams

The discrimination models described above have been coded into two subroutines, FFLD and WBLD. Flow diagrams for these routines are given in Figs. 3.9 and 3.10.

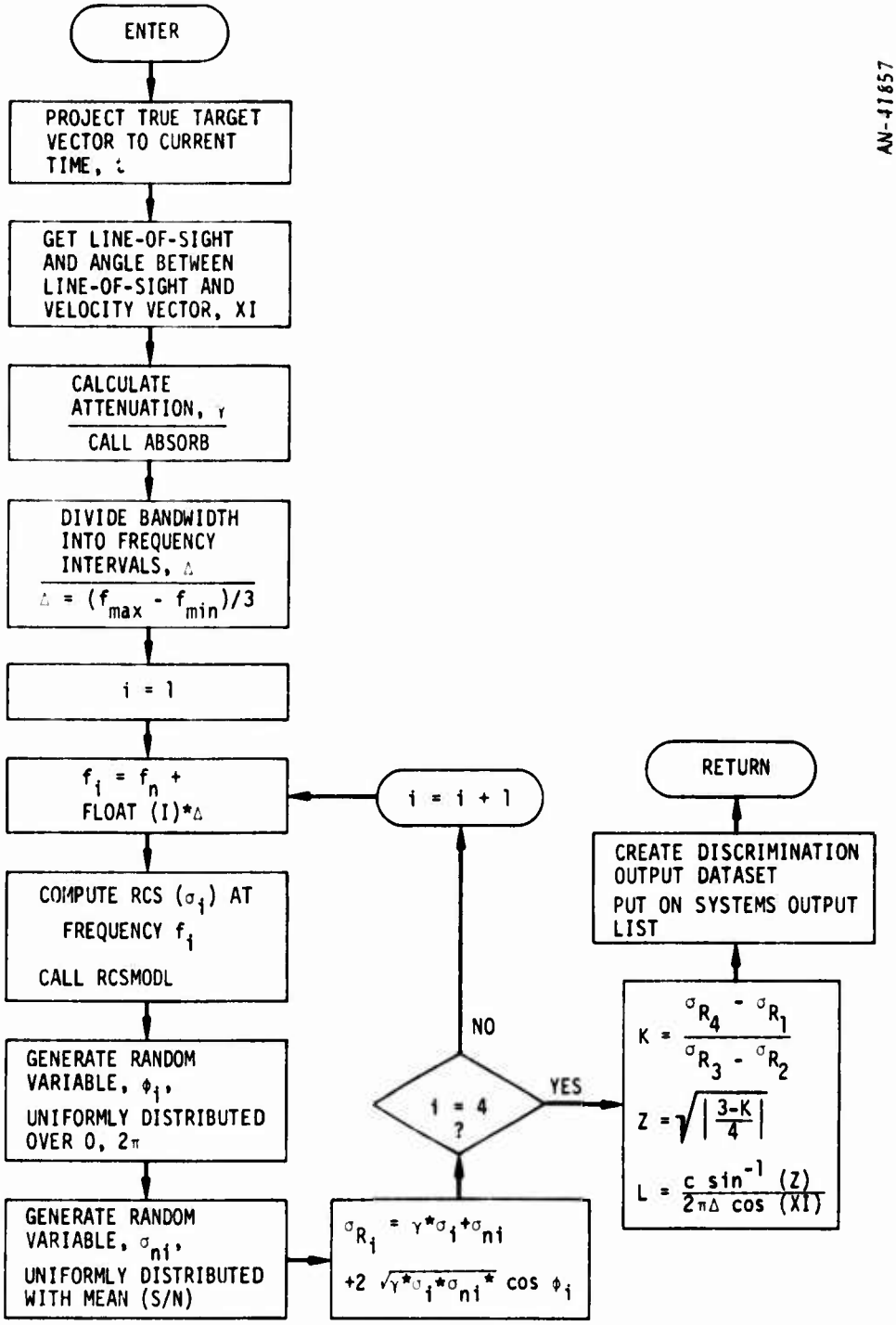


Figure 3.9. Fine-Frequency Length Discrimination Flow Chart

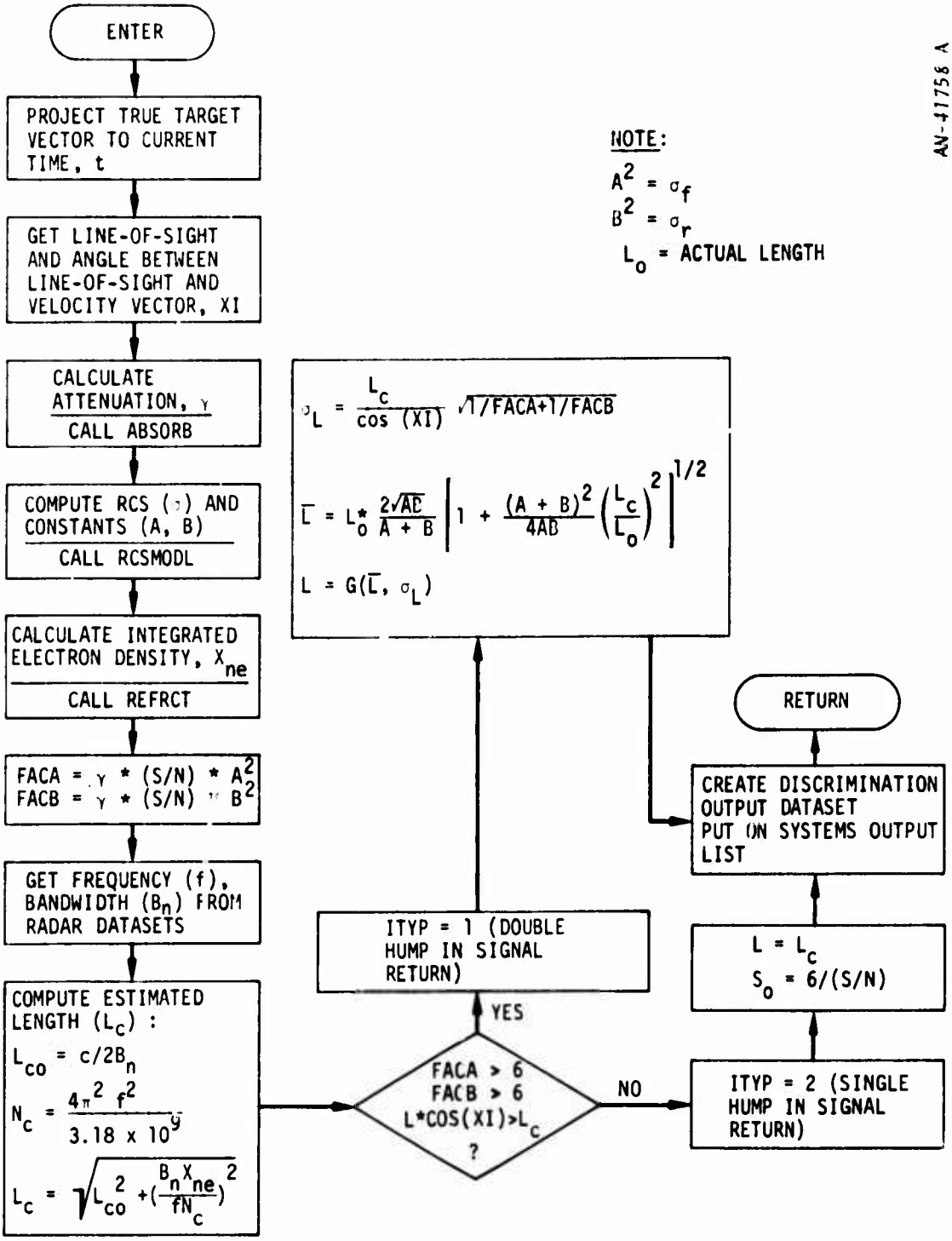


Figure 3.10. Wide-Bandwidth Length Discrimination Flow Chart

4 RADAR SIGNAL PROCESSING

4.1 OVERALL STRUCTURE

The signal processing applied to each radar pulse (radar look event) is simulated by a group of subroutines called by a controlling subroutine, RADMODS. The order in which these subroutines are called is designed to minimize the required computation by calculating the simplest effects first. For example, a target that cannot be detected because of absorption is deleted from the list for further processing, so that neither the other environmental effects nor the effects of radar characteristics need be calculated for that target.

Figure 4.1 shows the hierarchy of subroutines called by RADMODS. The processing begins when RADMODS is called by one of the radar event subroutines (Sec. 3) to calculate the apparent position and signal-to-noise ratio for a particular target. RADMODS calls subroutines to carry out these calculations in four steps:

1. Targets that might interfere with the desired target are listed, taking into account the possible effects of refraction.

2. The effects of absorption, noise, refraction, dispersion, and Faraday rotation are computed for each target on the list.

3. A revised target list is prepared, taking into account the foregoing environmental effects as well as any false targets due to reflection from fireballs.

4. For each target on the revised list, the apparent position R', u', v' and the signal-to-noise-plus-clutter ratio Z_o are calculated, taking into account the radar monopulse processing and range gating and the effects of clutter. As explained earlier, the apparent position R', u', v' does not include radar measurement errors.

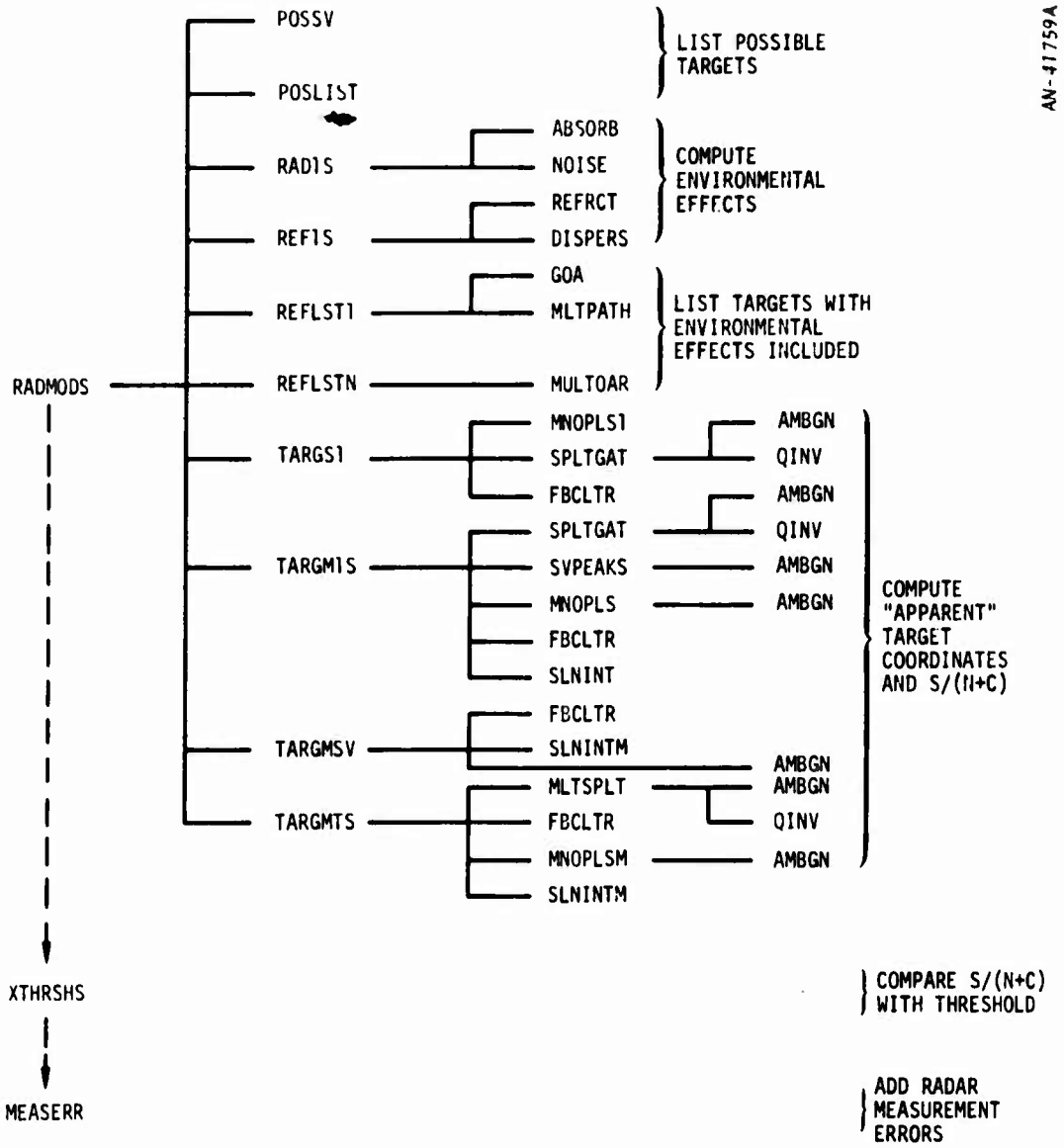


Figure 4.1. Radar Signal Processing Subroutine Hierarchy

RADMODS then returns to the calling event with the values R', u', v', and Z_o. Two further subroutines are called by the event rather than by RADMODS: XTHRSHS compares Z_o with the threshold signal-to-noise-plus-clutter ratio specified for that event type, and MEASERR adds radar measurement errors (bias and random) to R', u', v' to give the measured target position R_m, u_m, v_m.

Figure 4.2 is a flow chart of subroutine RADMODS, showing the sequence in which it calls the other subroutines. The first decision block in Fig. 4.2 chooses subroutine POSSV or subroutine POSLIST to construct the list of possible targets, POSLST, depending on whether the event type is search/verify (no range gating) or track initiate/track (with range gating). In the latter case, it may happen that there is no target (not even the desired one) that might be in the range gate; in that case, the length of POSLST is zero (NLST = 0), and RADMODS sets the "no detection" flag (IFDET = 0) and returns.

Subroutines RAD1S and REF1S are next called to compute the environmental effects; in each case, if none of the targets on the list can be detected, a flag IFAIL is set to a value other than zero and the "no detection" return is made.

The revised target list is prepared in two different ways (LISTA or LISTB), depending on whether there is a single (direct or refracted) path from radar to target (N_p = 1) or many. In the former case, unless the revised list is empty (MLST = 0), the subroutines to compute R', u', v', and Z_o are called, the choice of subroutine depending on whether there is more than one target on the revised list. In the latter case, the multiple refraction is assumed to produce "spread target clouds" of images, which are treated statistically in computing R', u', v', and Z_o.

Inputs: an indicator of the calling event type

Outputs: IFDET, (R', u', v'), Z_o *

*Because many of the signal processing subroutines have numerous inputs and outputs, they are listed in symbolic form only. See Appendix G for definitions of symbols used here and in flow charts.

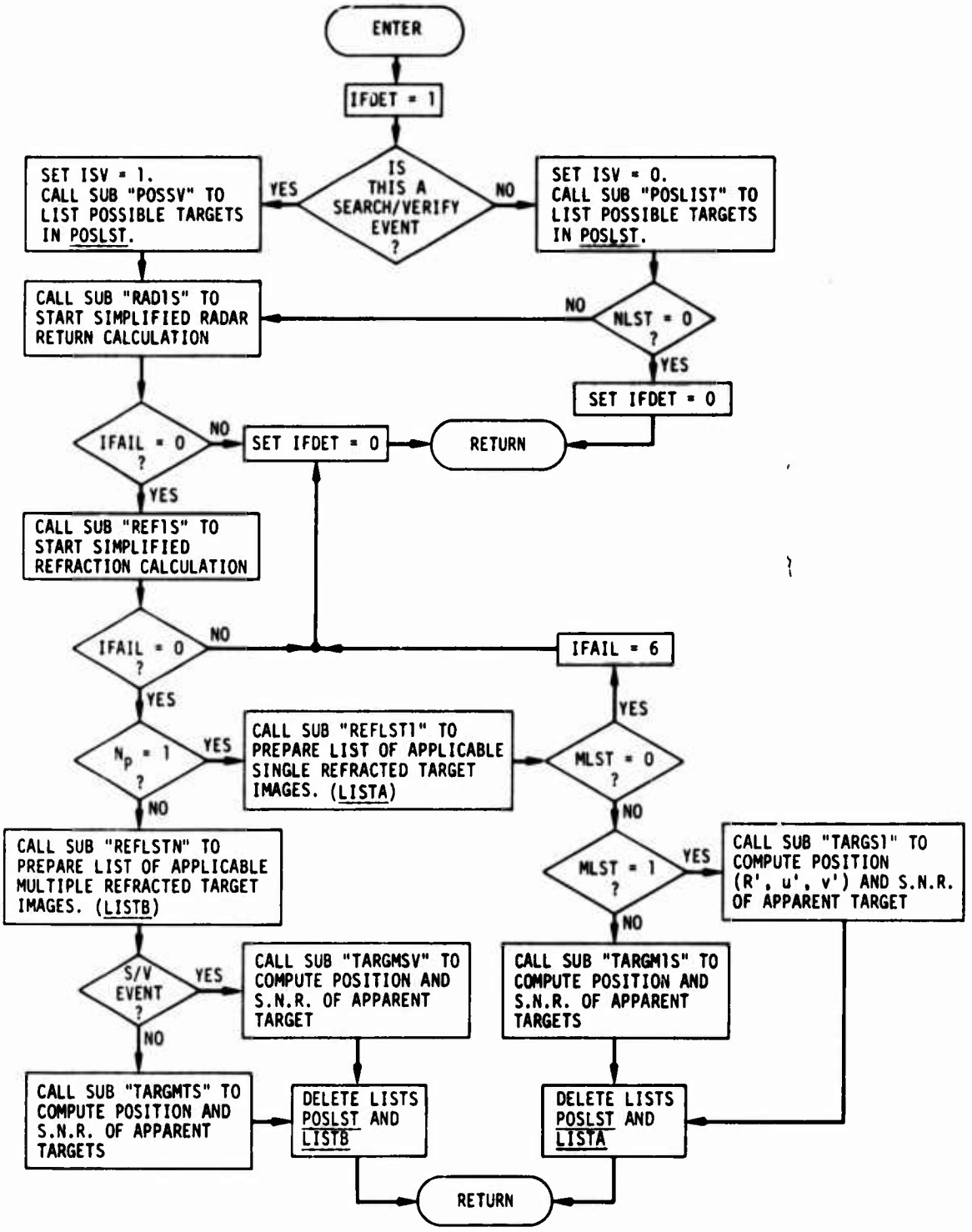


Figure 4.2. Subroutine RADMODS

Before going on to describe the processing steps in detail, a few comments ought to be made concerning features that are not simulated (some of which have been mentioned earlier).

- Only a single radar beamshape is used: circular with a constant or tapered sidelobe level. No beam stacking is modeled: transmit and receive beams have identical parameters and axis locations.
- No monopulse is used in search and verification events. The angular position output by these procedures is that of the center of the radar beam. The range output is that of the highest peak within the resolution cell centered on the target of interest.
- For all events, pulse-compression (chirped) signals may be used. A single model of the matched filter output (ambiguity function) is provided for pulse-compression signals: this model is an approximation falling between the shapes for Hamming-weighted and Hanning-weighted waveforms, and has a constant range sidelobe level.
- Dispersive distortion of the ambiguity function is taken into account, but the model used is strictly correct only for unchirped waveforms. For pulse-compression signals, the model used is based on a heuristic approach.
- A single range marking system is provided for events other than search and verification: the split-gate range tracker.
- It is assumed that when refraction is present, only two cases exist: either there is a single ray path between the radar and the target, or there are very many direct paths (striated debris case) producing a "spread target cloud" of many images. The former case is treated deterministically, the latter case stochastically. No intermediate case (e.g., three direct paths) is provided in the present version of ROSCOE.

- The method of dealing with monopulse and range-tracking in the "spread target" case is a simple random phase approximation, which is considered adequate for most practical cases.
- False alarms are not modeled.

4.2 POSSIBLE-TARGETS LIST

4.2.1 Subroutine POSSV (Fig. 4.3)

This subroutine performs two tasks needed to initiate the signal processing for search and verification events. It is used only with those events, since it makes certain assumptions which are only correct for them: that there is one particular (specified) target of concern to the event at this time, that the event is not interested in detecting any other targets except those which might cause direct interference with the given target, and that no range gate is defined or used by this event.

Subroutine POSSV first finds the current parameters and true state vector of the given target, and determines its doppler shift, f_d . It then finds the value of $\overline{f}_d$, the center frequency of the doppler "bin" in which the return from this target will be found; here it is assumed that for search/verify functions the receiver has an evenly spaced bank of doppler filters so that all targets can be detected even though no a priori information on their velocity is available; these doppler bins will be sufficiently narrow and numerous that any reasonable target velocity will cause the target to appear in one of the bins with no more than 1 dB of doppler mismatch loss.

The subroutine then goes on to prepare a list of all the real targets which might be of concern to the present event and beam position: the list [called POSLST] developed by this subroutine contains only the given target of interest plus any other targets which might later turn out to be within the radar beam and in the same resolution cell. The region of interest is conservatively defined with the possibility of later refractive effects in

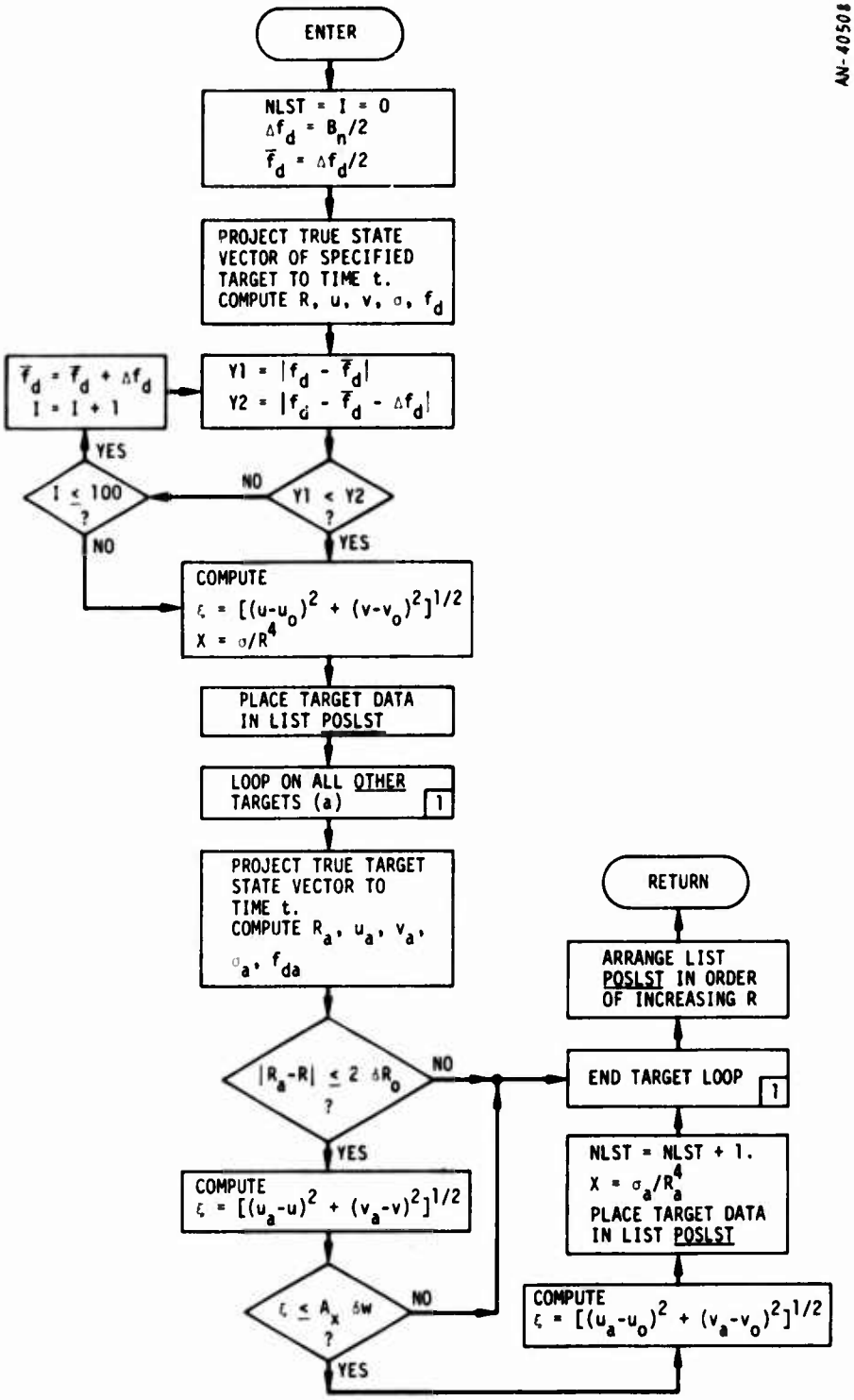


Figure 4.3. Subroutine POSSV

mind, by including all targets within A_x half-beamwidths of the desired target. Currently $A_x = 4$.

Inputs: t, (u_o, v_o), δw, δR_o, NT_o, B_n

Outputs: $\bar{f}_d$, NLST, POSLST

4.2.2 Subroutine POSLIST (Fig. 4.4)

This subroutine prepares a list of those real targets which might be of concern to the present event, beam position, and range gate. It is called by the events (track initiation and track) which specify a range gate center and gate width in addition to the beam position. The list (also called POSLST) developed by this subroutine contains all real targets which might later turn out to lie within the radar beam and within one range resolution distance on either side of the range gate limits. This region may or may not contain the target of particular concern to the present event at the given instant. The region is conservatively defined with the possibility of later refractive effects in mind, by using the constant A_x defined for POSSV and a range constant R_x (currently 3000 m).

Inputs: t, (R_c, u_o, v_o), W, δw, δR_o, NT_o

Outputs: NLST, POSLST

4.2.3 List POSLST

This list is created (before refraction calculations) in either POSSV or POSLIST, depending on the calling event. The list contains NLST entries (if NLST = 0 there is no list), each of which is a dataset pointer (DSP) to a dataset containing the following parameters for one target:

IT	serial number of this target in the list ($1 \leq IT \leq NLST$)
NT	corresponding target identification
R,u,v	current true coordinates of the target
σ	current target cross section
f_d	current target doppler shift

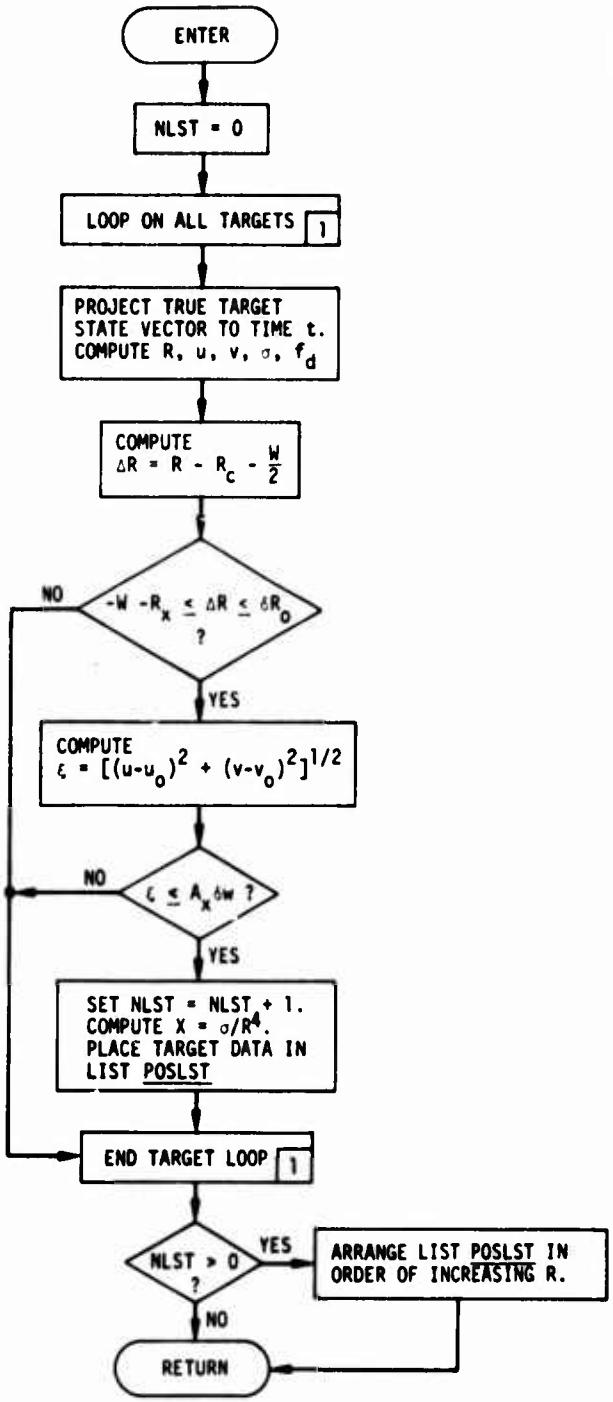


Figure 4.4. Subroutine POSLIST

ξ current true angle off the beam axis for this target in sine-space:

$$\xi^2 \equiv (u - u_o)^2 + (v - v_o)^2$$

X range-weighted cross section, σ/R^4

IFIT flag to indicate whether this is (IFIT = 1) or is not (IFIT = 0) the target of primary interest to this event

4.3 ENVIRONMENTAL EFFECTS

4.3.1 Subroutine RAD1S (Fig. 4.5)

This subroutine begins the job of calculating the radar return. Selecting an appropriate target from among those in POSLST, it conservatively estimates the maximum amount of absorption along the path to this point, and the maximum amount of noise in the beam position, which could exist and still allow a threshold crossing to be possible. The actual absorption and noise are then computed by calling the propagation subroutines ABSORB and NOISE (see Sec. 5). If these exceed the estimated maxima, the computation is aborted with a message to indicate the reason for no return. If the absorption and noise are less than these limits, their values are recorded and certain parameters of the radar equation are calculated and output.

The "conservative estimate" of maximum allowable noise and absorption is accomplished by multiplying the target cross section by a factor α. This factor is a predesignated constant; the value $\alpha = 10$ is presently specified in a data statement in subroutine RAD1S. This allows for focusing and other enhancement effects.

This subroutine operates in terms of a form of the radar equation which calculates the pre-detection signal-to-noise-plus-clutter ratio, Z_i, in terms of four input radar parameters and several factors computed in this subroutine and others:

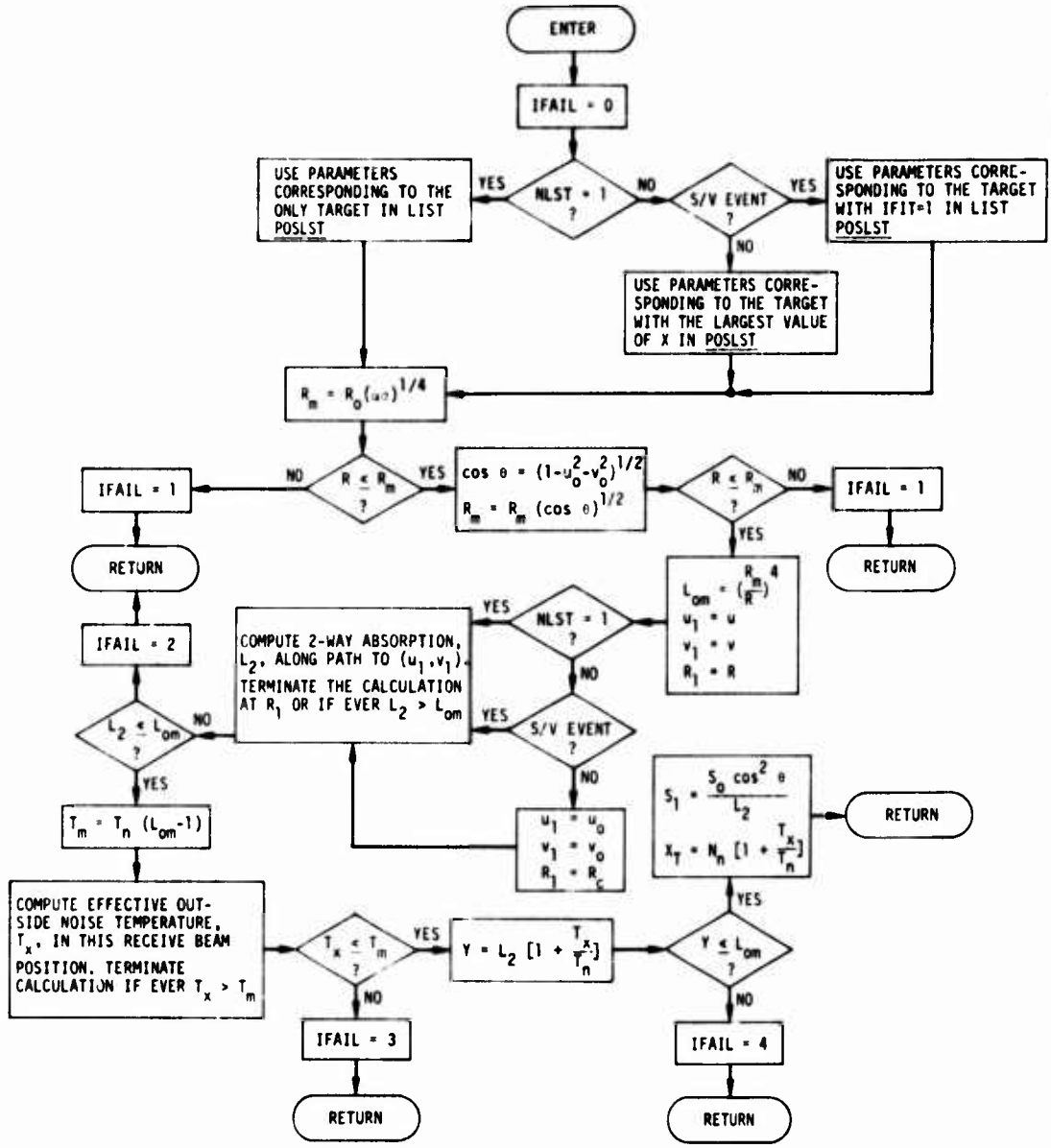


Figure 4.5. Subroutine RAD1S

$$Z_i = (S/N)_{Ti}(R_o/R)^4 \frac{(\cos^2\theta)F_T F_R \sigma}{L_o[1 + (T_x/T_n) + (C/N_n)]}$$

where

$^*(S/N)_{Ti}$ = pre-detection signal-to-noise threshold

$^*R_o$ = range at which a 1 m^2 target will just exceed this threshold under nominal conditions**

R = current range to target

θ = off-boresight angle of beam axis

F_T = relative gain factor due to target's position off-axis of transmit beam

F_R = relative gain factor due to target's position off-axis of receive beam (may differ from F_T because transmit and receive paths to the target need not be the same)

σ = target cross section

$^*T_n$ = nominal system noise temperature

N_n = nominal system noise power (kB_nT_n)

$^*B_n$ = noise bandwidth for the transmitted signal ($B_n \approx 1/\text{pulse length}$)

L_o = total outside loss factor, including absorption (L_2), dispersive loss (L_D), and Faraday rotation loss (L_F), if applicable

T_x = total effective outside noise temperature

C = total effective clutter power

---

*Factors marked by an asterisk are input radar parameters.

**Target on the beam axis at radar boresight, and no outside losses, noise, or clutter.

For some applications, it is convenient to exhibit directly the decomposition of Z_i into the ratio of a signal power, S, to a noise-plus-clutter power, X_T:

$$S = S_o \frac{(\cos^2\theta)F_T F_R \sigma}{L_o R^4}$$

$$X_T = N_n[1 + (T_x/T_n) + (C/N_n)]$$

where

$$S_o = (S/N)_{Ti} R_o^4 N_n$$

is computed in advance for each radar waveform.

IFAIL is an index to indicate whether a threshold crossing may be possible, and if not, why not.

Value of IFAIL	Meaning
0	Possible
1	Impossible--range too great
2	Impossible--too much absorption
3	Impossible--too much outside noise
4	Impossible--absorption plus noise
5	Impossible--dispersive and Faraday rotation losses
6	Impossible--no targets in LISTA (Sec. 4.4.1)

Inputs: NLST, POSLST, (u_o, v_o), δw, R_o, R_c, S_o, T_n, N_n, α

Outputs: IFAIL, (R_1, u_1, v_1), ITO, $\cos\theta$, S_1, X_T, Y, L_{om}, L_2

4.3.2 Subroutine REF1S (Fig. 4.6)

This subroutine calculates further propagation effects: refraction, dispersion, and Faraday rotation.

To begin with, the user may define a limiting altitude, H_{REF}, below which no refractive effects are to be computed. The idea is that for ray paths lying entirely below H_{REF}, significant refraction will always be accompanied by overwhelming absorption, so it would be a waste of time to calculate the refraction. Note that both the target and the radar must lie below H_{REF} in order for this condition to be applicable. The user may, of course, select $H_{REF} = 0$, in which case this test becomes irrelevant.

Refraction. It is assumed in ROSCOE that the number of propagation paths between the radar and the target (N_p) is either one or much greater than one. The latter case occurs only when there is an intervening striated region. When $N_p >> 1$, the number of apparent target images is given by N_p^2.

Subroutine REF1S computes only one set of refraction parameters: those affecting the previously-defined target whose coordinates are (R_1, u_1,v_1). These same parameters are then assumed to apply unchanged to all targets within the region of interest. In other words, the range and angular extent of the region is assumed to be small enough so that the same refraction affects all targets within it. This does not necessarily mean that all these targets suffer the same refractive displacements, since there may be uncertainties associated with each bias component.

Subroutine REFRCT (Sec. 5.3) is called to compute the refraction parameters. It first determines whether there is a striated region between the radar and the target point (R_1,u_1,v_1). If so, it estimates the number of propagation paths, N_p. If not, it sets $N_p = 1$. In either case, it computes the integrated electron density,

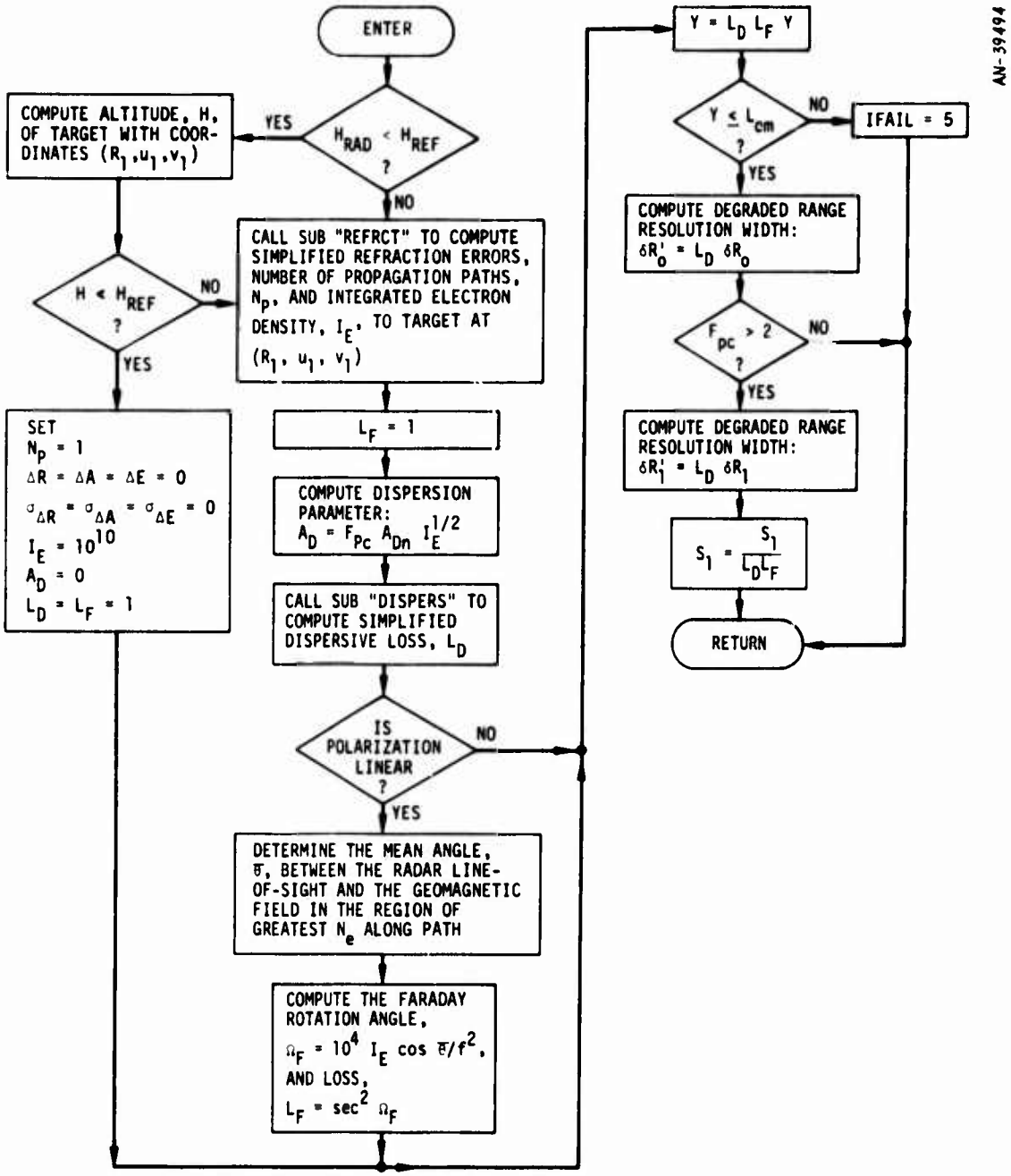


Figure 4.6. Subroutine REFlS

$$I_E = \int N_e ds$$

along the direct path between the radar and this target. This value of I_E (conventionally given in units of electrons/cm^2) is assumed in this version to apply to all targets in the region of interest.

When $N_p = 1$, the properties of the refraction are given by six parameters:

$$\Delta R,\ \Delta A,\ \Delta E,\ \sigma_{\Delta R},\ \sigma_{\Delta A},\ \sigma_{\Delta E}$$

The first three parameters give the refractive bias errors which must be added to the true target coordinates (R,A,E) to give the apparent target coordinates (R',A',E'). The last three parameters give the uncertainties (if any) of these bias errors, in terms of their standard deviations. The use of these parameters is described in Sec. 4.4.1, Subroutine REFLST1.

When $N_p > 1$ (striated-debris refraction), there are many false target images scattered over a three-dimensional region of space. In this case, subroutine REFRCT provides statistical information on the spatial distribution of these images, and on their effective cross-section distribution.

The spatial distribution of the set of N_p^2 target images is assumed to have the form of a probability ellipsoid. The dimensions and orientation of this ellipsoid are approximately determined from the phenomenology. For the target whose true position is (R_1,u_1,v_1), the centroid of this ellipsoid is located at $(R_1 + \Delta R,\ u_1 + \Delta u,\ v_1 + \Delta v)$; the values of $\Delta R,\Delta u,\Delta v$ are among the outputs of REFRCT. The 3×3 covariance matrix defining the error ellipsoid in Cartesian space is projected onto the radar line-of-sight to produce the range-error standard deviation $\sigma_{\Delta R}$. The matrix is also projected onto the sine-space plane, using a linearized approximation based

on the assumption that the dimensions of the ellipsoid are absolutely small; the result is an error ellipse in sine-space, defined by its semimajor axis (σ_1), its semiminor axis (σ_2), and the angle (μ) between the major axis and the u-axis of the sine-space coordinate system (to eliminate possible ambiguities, the model selects $0 \leq \mu < \pi$).

The effective cross sections of the target images will be affected by focusing and defocusing, so that they differ from the true cross section of the real target and in fact will have a distribution of values. The subroutine provides the mean ($\bar{\sigma}$) and standard deviation (σ_σ) of this cross-section distribution under the assumption that it is Gaussian.

Thus, for $N_p > 1$, the output of subroutine REFRCT is the following set of parameters:

$$\Delta R,\ \Delta u,\ \Delta v,\ \sigma_{\Delta R},\ \sigma_1,\ \sigma_2,\ \mu,\ \bar{\sigma},\ \sigma_\sigma$$

In using these parameters, ROSCOE makes the following simplifying assumptions:

1. The errors in R,u,v are uncorrelated
2. The standard deviations are all sufficiently small that linearization is permissible and sine-space coordinates can be treated additively.
3. The random phase approximation can be employed when combining the signal returns from this set of false target images.

Dispersion. This is treated by a method derived from an analysis which is described in Appendix B.

The dispersion parameter, A_D, for a signal of bandwidth B, is defined to be

$$A_D = 4\gamma B$$

where

$$\gamma^2 \equiv \frac{1}{c\omega^3} \int \omega_N^2 \left[1 - (\omega_N^2/\omega^2)\right]^{-3/2} ds$$

and where

$$\omega = 2\pi f$$

$$\omega_N^2 \simeq 3.182 \times 10^9 N_e \text{ rad/s}$$

In most practical cases, γ can be approximated in terms of the integrated electron density, I_E, by the formula

$$\gamma^2 \simeq 4.28 \times 10^{-4} I_E/f^3$$

where f is the radar frequency in Hz. Therefore, we write

$$A_D \simeq 8.28 \times 10^{-2} (I_E/f^3)^{1/2} B$$

This formula was actually derived only for rectangular unmodulated pulses, where $B \equiv B_n \simeq 1/\tau$. For these we can write

$$A_D \simeq A_{Dn} I_E^{1/2}$$

where

$$A_{Dn} \simeq 0.0828 B_n f^{-3/2}$$

For chirped pulses, the effective bandwidth is

$$B \simeq F_{pc} B_n$$

where F_{pc} is the pulse-compression ratio. Therefore, for such signals

we use for the dispersion parameter the form

$$A_D \simeq F_{pc} A_{Dn} I_E^{1/2}$$

The parameter A_{Dn} is precomputed for each waveform and stored in the appropriate dataset.

In terms of this dispersion parameter, the dispersive mismatch loss (L_D) at the peak of the output of the receiver filter is given by a complicated expression involving Fresnel integrals and trigonometric functions. Since this is only approximately valid, however, especially in the case of chirped signals, subroutine DISPERS (Sec. 4.3.3) computes an approximation to L_D by using a piecewise curve fit to a plot of the complicated expression mentioned above.

Faraday Rotation. If the radar uses a linearly polarized signal, Faraday rotation has to be taken into account. Because of this rotation, some of the return signal may arrive in the crossed polarization and hence not contribute to the received power; this results in the "Faraday rotation loss," L_F .

These two losses, L_D and L_F , are combined with the other known losses and noise factors (excluding clutter, which will be calculated later) and compared with a conservatively estimated value of the return signal power (as in RADlS) to see if a threshold crossing is still possible. If not, the subroutine terminates the computation and returns with an appropriate diagnostic index.

Finally, the degraded range resolution widths, $\delta R_o'$ and (if chirped signals are used) $\delta R_1'$, are estimated. The estimate is made by assuming that the ambiguity function in the range dimension is triangular (as it is for an unchirped signal in the absence of dispersion), with its peak amplitude lowered by $\sqrt{L_D}$ and its width correspondingly widened so as to satisfy

the principle of energy conservation before and after dispersion. By this method, the degraded range resolution turns out to be simply L_D times the nominal resolution width. This is a good approximation for an unmodulated rectangular pulse, for which comparisons have been made with the exact calculation. ROSCOE also uses the same technique for a chirped pulse, though no estimates of the goodness of the approximation have been made for this case.

The nominal range-resolution width for an unchirped rectangular pulse is

$$\delta R_o = c\tau/2$$

This is also the half-extent of the region in which the range sidelobes are important for a chirped pulse.

The nominal range-resolution width for a chirped pulse is

$$\delta R_1 = 2\delta R_o/F_{pc}$$

The numerical factor takes roughly into account the spreading which results from the weighting usually applied to reduce the magnitude of the range sidelobes.

Inputs: (R_1,u_1,v_1), H_{REF}, H_{RAD}, F_{pc},f, A_{Dn}, δR_o, δR_1, polarization indicator, S_1, L_{om}, Y, IFAIL

Outputs: N_p, L_E, L_D, L_F, Ω_F, S_1, $\delta R_o'$, $\delta R_1'$, IFAIL

Refraction Parameters: ΔR, ΔA, ΔE, $\sigma_{\Delta R}$, $\sigma_{\Delta A}$, $\sigma_{\Delta E}$: if $N_p = 1$

ΔR, Δu, Δv, $\sigma_{\Delta R}$, σ_1, σ_2, μ, $\bar{\sigma}$, σ_σ: if $N_p > 1$

4.3.3 Subroutine DISPERS (Fig. 4.7)

This subroutine computes the loss factor, L_D , due to dispersive effects along the line-of-sight between the radar and the target. This

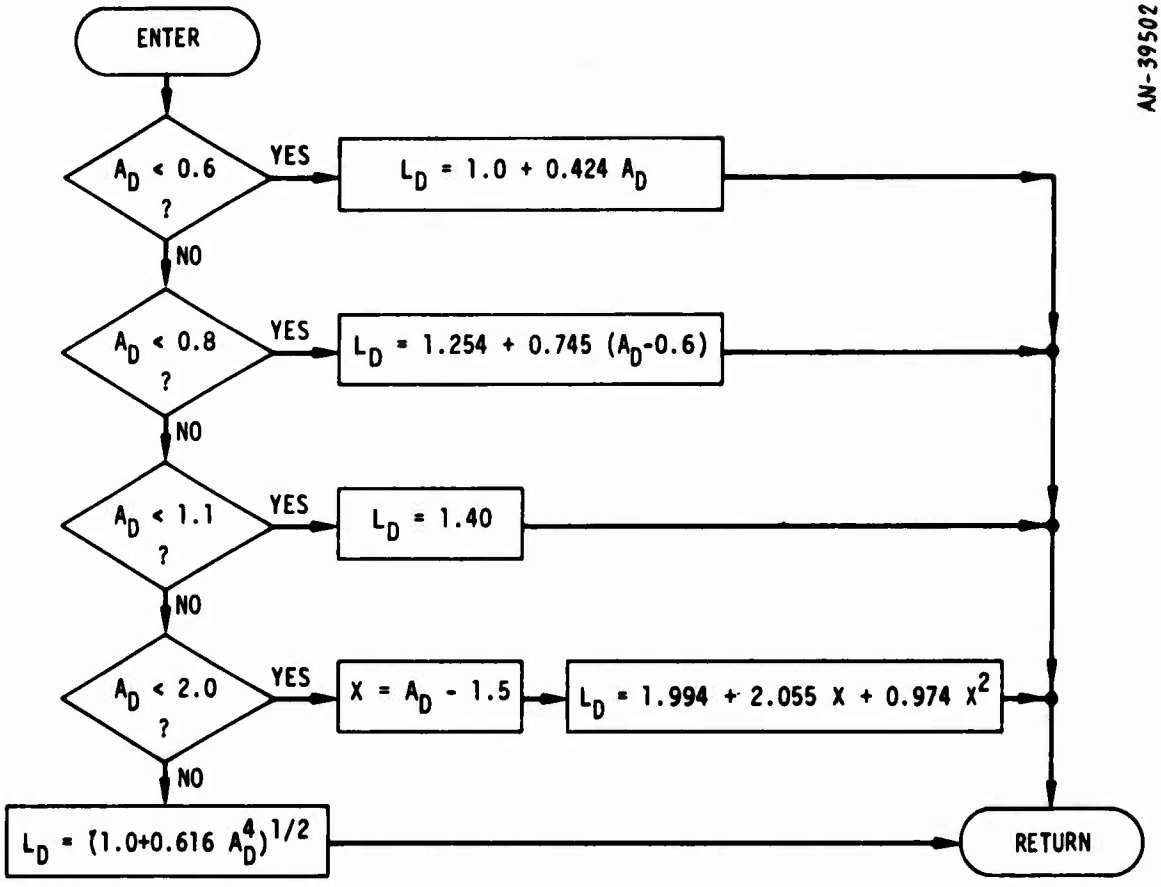


Figure 4.7. Subroutine DISPERS

loss factor represents the power loss at the peak of the ambiguity function, and is computed by a fit to a simplified model which, while strictly correct only for unchirped waveforms, is approximately applicable to chirped waveforms. The input to this subroutine is a dispersion parameter, A_D , which depends on the radar and signal parameters and on the integrated electron density along the line-of-sight. See Appendix B for the derivation of this parameter.

Input: A_D

Output: L_D

4.4 REVISED TARGET LIST

4.4.1 Subroutine REFLST1 (Fig. 4.8)

This subroutine is called to prepare a list of targets which will affect the radar return in those cases when there is only a single refracted (or direct) ray path between the radar and each real target ($N_p = 1$). This list, called LISTA, is constructed by applying the refractive parameters derived in subroutine REF1S to the targets in list POSLST, and making use of the known properties of the radar waveform, beam parameters, and receiver range gate (if any) to determine which of the target images will contribute to the return for the particular radar event and target presently under consideration.

As shown in Fig. 4.8, the procedure for adding the refractive parameters is to select random numbers from Gaussian distributions with the means (ΔR, ΔA, ΔE) and standard deviations ($\sigma_{\Delta R}$, $\sigma_{\Delta A}$, $\sigma_{\Delta E}$, which may be zero) calculated by REFRCT, and add them to the previously computed target coordinates. Note that conversion from sine-space coordinates (R,u,v) to angular-space coordinates (R,A,E) is required because angles do not add in sine space.

The apparent range of a target is also affected, when a chirp waveform is used, by range-doppler coupling (see Appendix B). This is accounted for by the term $\alpha(f_D - \bar{f}_D)$ in the range expressions of Fig. 4.8.

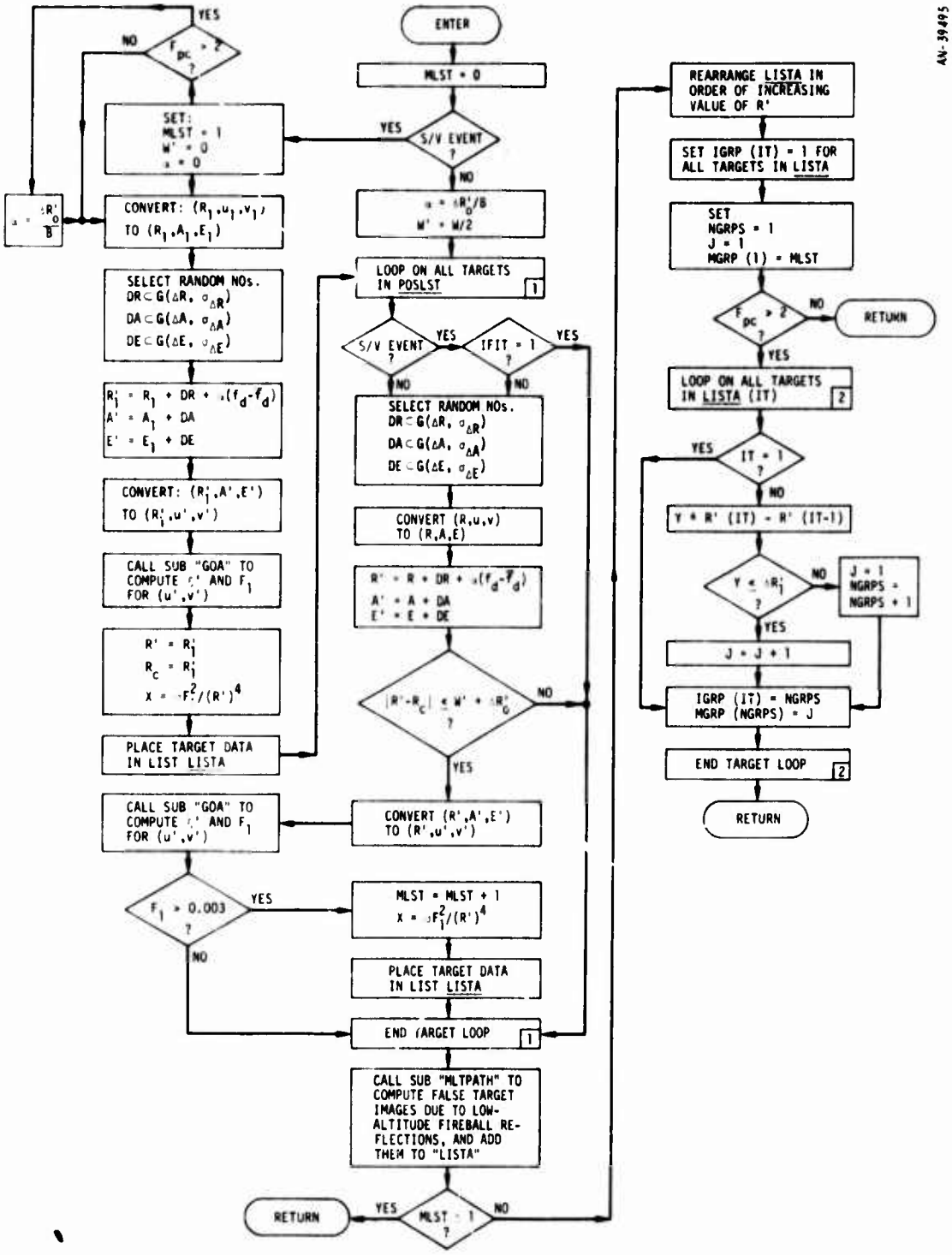


Figure 4.8. Subroutine REFLST1

The resulting refracted target coordinates are tested to see whether they are within a range resolution distance ($\delta R_o'$) of the desired target (or of the range gate if one is used) and within the 25-dB one-way radar beamwidth ($F_1 > 0.003$). (The off-axis gain parameter F_1 is computed by subroutine GOA, Sec. 4.4.2.) Targets within these limits are placed on LISTA. Note that, when a range gate is used, the desired target itself may not be on the list.

To this list are then added all significant "false target" images caused by signal reflection from low-altitude fireballs; the properties of these multipath images are computed in subroutine MLTPATH (Sec. 5). Once these "false" images have been inserted into LISTA, all distinction between "real" and "false" targets is thenceforth disregarded as being irrelevant; all images are treated the same way in the succeeding subroutines. A final step in subroutine REFLST1 indexes all target images in LISTA into subgroups which are mutually unresolvable by the current radar waveform; this grouping is not used in the present ROSCOE signal processing model, but may be needed in later versions.*

Inputs: NLST, POSLST, F_{pc}, W, R_c, ITO, (R_1, u_1, v_1), $\{\Delta R, \Delta A, \Delta E, \sigma_{\Delta R}, \sigma_{\Delta A}, \sigma_{\Delta E}\}$, $\delta R_o'$, $\delta R_1'$, B

Outputs: MLST, NGRPS, MGRP, LISTA

4.4.2 Subroutine GOA (Fig. 4.9)

This subroutine computes the one-way gain factor, F_1, for a target that may be located off the axis of the radar beam. The factor is normalized to unity on the axis. Only a circular mainlobe with constant or tapered sidelobes is provided for. The calculation is done in sine-space coordinates.

*For example, it is not inconceivable that some user may wish to examine his simulated radar returns with a precision which demands doing actual phasor sums on each group of unresolved images.

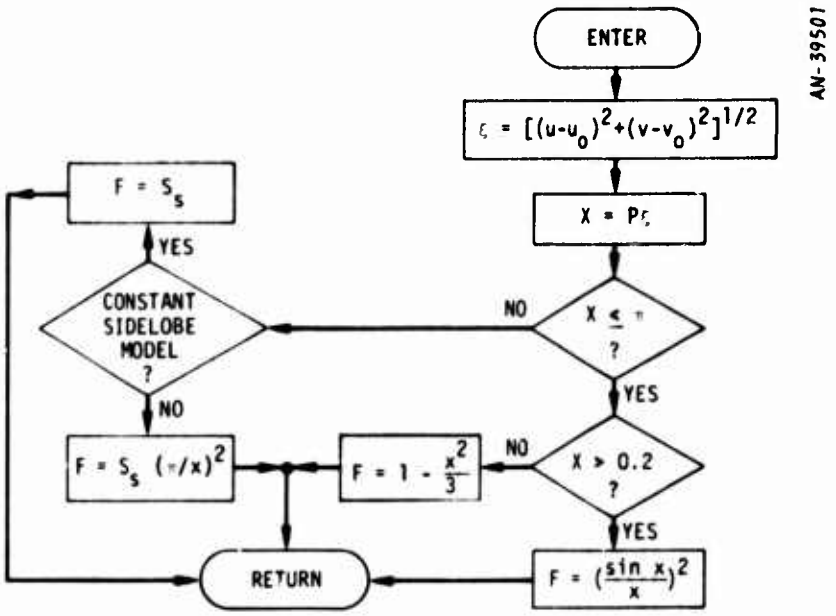


Figure 4.9. Subroutine GOA

Besides F_1, this subroutine provides the value of ξ, the target's angle off the beam axis in sine space.

Inputs: (u,v), (u_o,v_o), p, S_s, D_s

Outputs: F_1, ξ

4.4.3 Subroutine REFLSTN (Fig. 4.10)

This subroutine is called to prepare a list of targets which will affect the radar return in those cases when there are many refracted ray paths between the radar and each real target ($N_p > 1$). As mentioned earlier, the radar model assumes that this case involves a "spread target cloud" of many images, produced by an intervening striated debris region, and these images are defined and treated statistically. The list, called LISTB, is constructed by applying the statistical refractive parameters derived in subroutine REFlS to the targets in list POSLST, and making use of known properties of the radar waveform, beam parameters, and receiver range gate (if any) to determine which of the target image clouds will

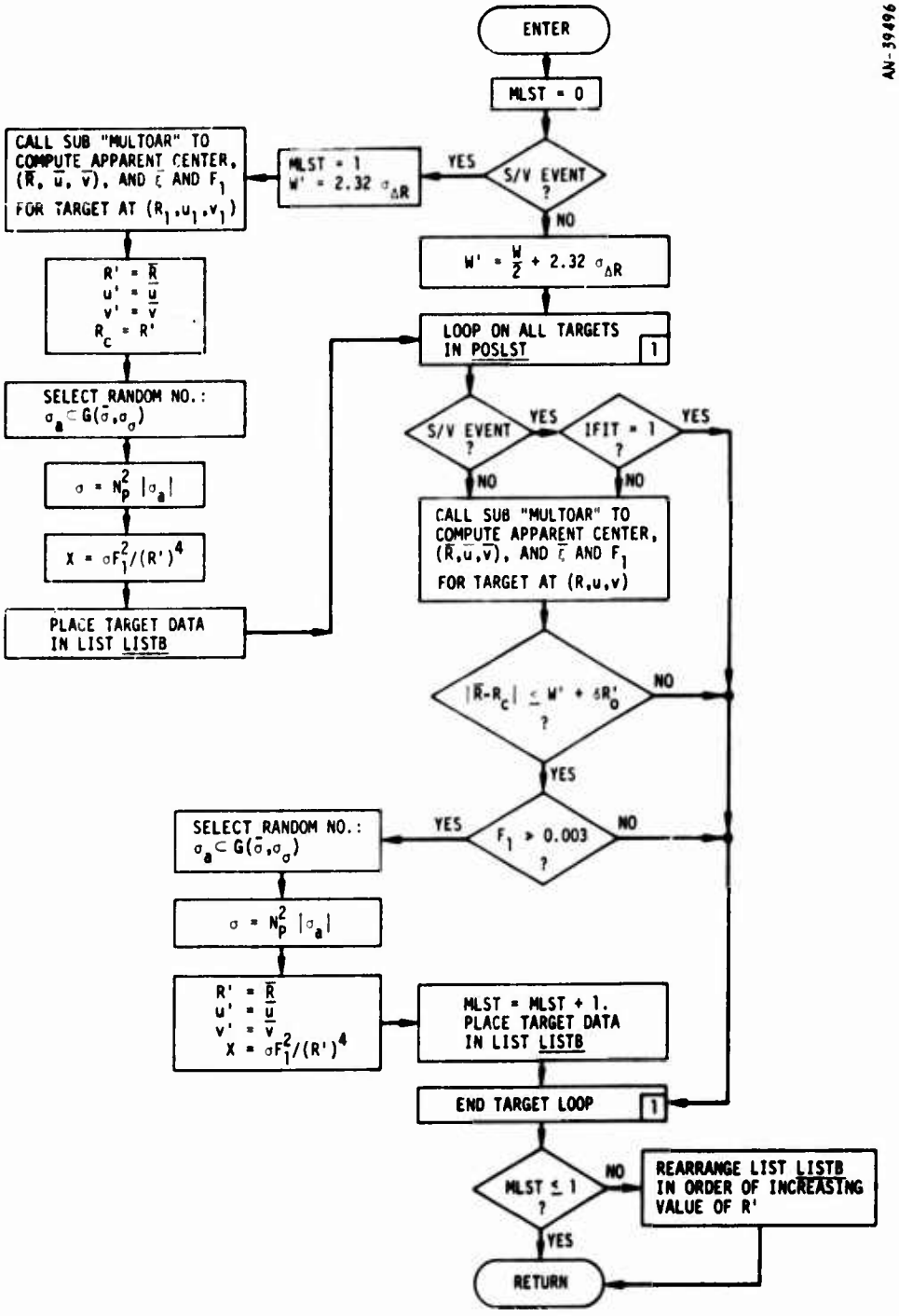


Figure 4.10. Subroutine REFLSTN

contribute to the return for the particular radar event and target presently under consideration. Because there are so many target images that they must be dealt with statistically in this case, it is assumed that a few additional images produced by low-altitude fireball multipath would be of negligible importance, so these are not included in the calculation in this subroutine. "Close-target" groupings are also not included. The effective parameters of the image clouds as viewed by the radar are computed in subroutine MULTOAR (Sec. 4.4.4).

Inputs: NLST, POSLST, W, R_c, (R_1,u_1,v_1), N_p, $\delta R_o'$, $\sigma_{\Delta R}$, $\bar{\sigma}$, σ_σ

Outputs: MLST, LISTB

3

4.4.4 Subroutine MULTOAR (Fig. 4.11)

This subroutine performs a similar function for a "spread target cloud" to that which subroutine GOA performs for a single target point. It combines the statistical parameters of the image cloud with the beam position and waveform parameters to compute the effective position of the weighted center of the cloud $(\bar{R},\bar{u},\bar{v})$ with respect to the beam, and then calculates the target's apparent off-axis angle (in sine-space), ξ, and the effective one-way relative gain factor, F_1.

Inputs: (R,u,v), ΔR, Δu, Δv, σ_1, σ_2, μ, f_d, F_{pc}, (u_o,v_o), p, $\delta R_o'$, B, $\bar{f}_d$, S_s

Outputs: $(\bar{R},\bar{u},\bar{v})$, $\bar{\xi}$, F_1

4.4.5 LISTA and LISTB

These are the lists of target points which contribute to the received signal. LISTA applies in the "single direct path" case ($N_p = 1$), and LISTB in the "striation" case ($N_p > 1$). False target images due to low-altitude fireball reflections are included in LISTA (but not in LISTB) in the same way as "real" targets. Either list contains MLST entries (if MLST = 0 there is no list), each of which is a DSP to a dataset containing the following parameters for one apparent target (or target cloud).

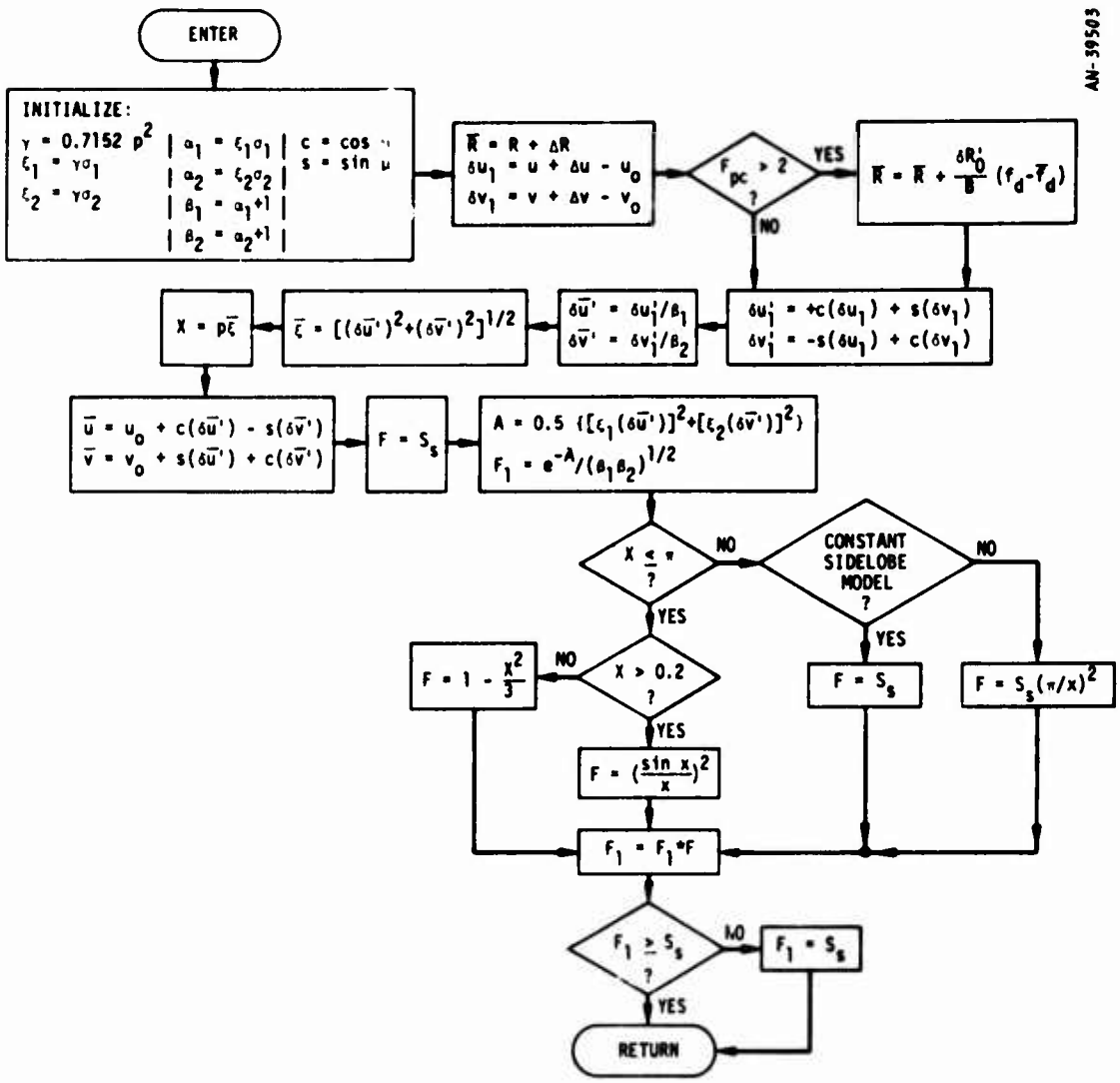


Figure 4.11. Subroutine MULTOAR

4.4.5.1 LISTA

IT	serial number of this target in the list ($1 \leq IT \leq MLST$)
NT	corresponding target identification (may be the same for several "targets" if some of them are reflected images of the same real target)
R',u',v'	current apparent coordinates of the target
σ	current effective target cross section
f_d	current target doppler shift
ξ'	current apparent angle off the beam axis for this target in sine space: $\xi' = [(u' - u_o)^2 + (v' - v_o)^2]^{1/2}$
F_1	one-way relative gain off-axis for this apparent target
X	cross-section parameter for this target, $\sigma F_1^2/(R')^4$
IFIT	flag to indicate whether this is (IFIT = 1) or is not (IFIT = 0) the target of primary interest to this event
IGRP	index number which is the same for all targets belonging to a given close-target group

4.4.5.2 LISTB

IT	serial number of this "target" in the list ($1 \leq IT \leq MLST$)
NT	identification of the real target of which this cloud is the refracted image
R',u',v'	current apparent coordinates of the effective center of this cloud ($R' = \bar{R}, u' = \bar{u}, v' = \bar{v}$)
σ	effective cross-section of this entire cloud as calculated using the random phase approximation
f_d	current effective doppler shift of the effective center of this cloud

ξ' current angle off the beam axis in sine-space for the effective center of this cloud ($\xi' \equiv \bar{\xi}$)

F_1 effective one-way relative gain off-axis for this cloud

X cross-section parameter for this cloud, $\sigma F_1^2/(R')^4$

IFIT flag to indicate whether this cloud is (IFIT = 1) or is not (IFIT = 0) the image of the target of primary interest to this event

4.5 TARGET POSITION AND SIGNAL-TO-NOISE-PLUS-CLUTTER RATIO

The final portion of the radar signal processing calculations controlled by subroutine RADMODS is to calculate the "apparent" target coordinates (R',u',v') and output signal-to-noise-plus-clutter ratio (Z_o), using the list of interfering targets and their properties (LISTA or LISTB) that has resulted from the previous steps.

One of four subroutines is used, depending on the situation:

TARGS1	$N_p = 1$, only one target on LISTA
TARGM1S	$N_p = 1$, more than one target on LISTA
TARGMSV	$N_p > 1$, no range gating
TARGMTS	$N_p > 1$, range gated signals

These subroutines in turn call other subroutines (Sec. 4.6) to calculate the effects of monopulse, range gating, and targets in the range sidelobes, and the propagation routine (Sec. 5.4) that computes clutter.

4.5.1 Subroutine TARGS1 (Fig. 4.12)

This subroutine is called to compute the apparent position (R',u',v') and output signal-to-noise-plus-clutter ratio (Z_o) for the case when there is single refraction ($N_p = 1$) and only one real target is found to be in LISTA. Although this case is basically simple, monopulse and split-gate simulations (MNOPLS1, SPLTGAT) must be called into play when appropriate, and the clutter must be computed (FBCLTR, Sec. 5.4).

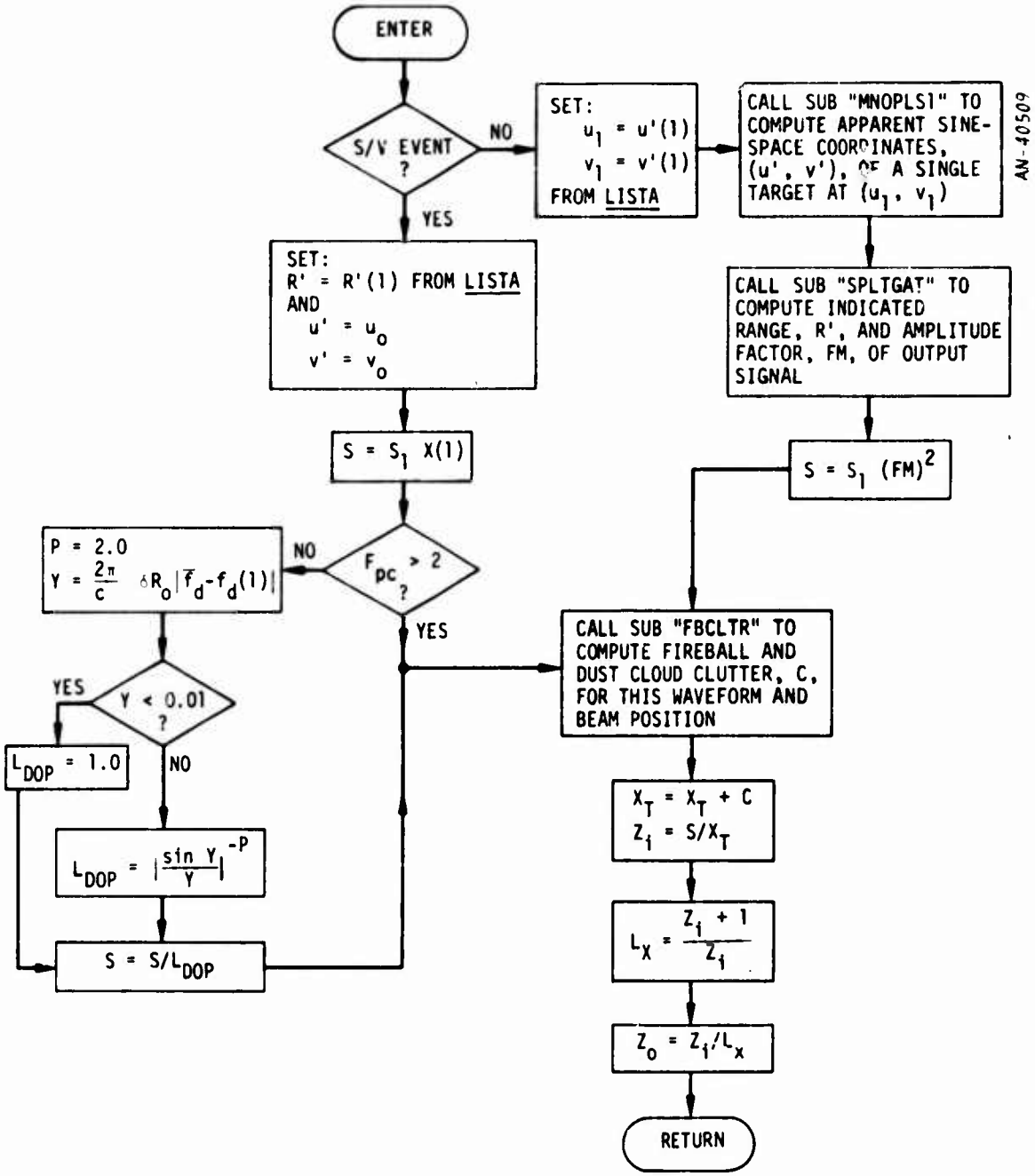


Figure 4.12. Subroutine TARGS1

Inputs: MLST, LISTA, (u_o,v_o), R_o, F_{pc}, $\bar{f}_d$, S_1, X_T

Outputs: (R',u',v'), Z_o

4.5.2 Subroutine TARGM1S (Fig. 4.13)

This subroutine is called to compute R', u', v', and Z_o for the case when there is single refraction ($N_p = 1$) but more than one target is found to be in LISTA. Direct interference between targets in the same resolution cell, as well as equivalent noise due to targets in the range sidelobes, where applicable (subroutine SLNINT), is taken into account.

Inputs: MLST, LISTA, (u_o,v_o), S_1, F_{pc}, X_T

Outputs: (R',u',v'), Z_o

4.5.3 Subroutine TARGMSV (Fig. 4.14)

This subroutine is called to compute R', u', v', and Z_o for the case of refraction through a striated region ($N_p > 1$); it is applicable only to search and verification events (no range gating), since it makes the same assumptions as were mentioned in Sec. 4.2.1 in connection with subroutine POSSV. A simplified statistical model of the "spread target cloud" is used to compute the apparent target range R' ; since monopulse is not used, the system estimate of the target's angular position is just that of the beam axis. A special subroutine (SLNINTM) is used to compute equivalent noise from spread targets in the range sidelobes (applied only to those cases where pulse compression is used in search and verification events) with a statistical model.

Inputs: (u_o,v_o), F_{pc}, $\delta R'_o$, $\delta R'_1$, $\sigma_{\Delta R}$, MLST, LISTB, ITO, S_1, X_T

Outputs: (R',u',v'), Z_o

4.5.4 Subroutine TARGMTS (Fig. 4.15)

This subroutine is called to compute R', u', v', and Z_o for the case of striated refraction ($N_p > 1$) when the event type is one that defines a range gate position and width (track initiation, track). As

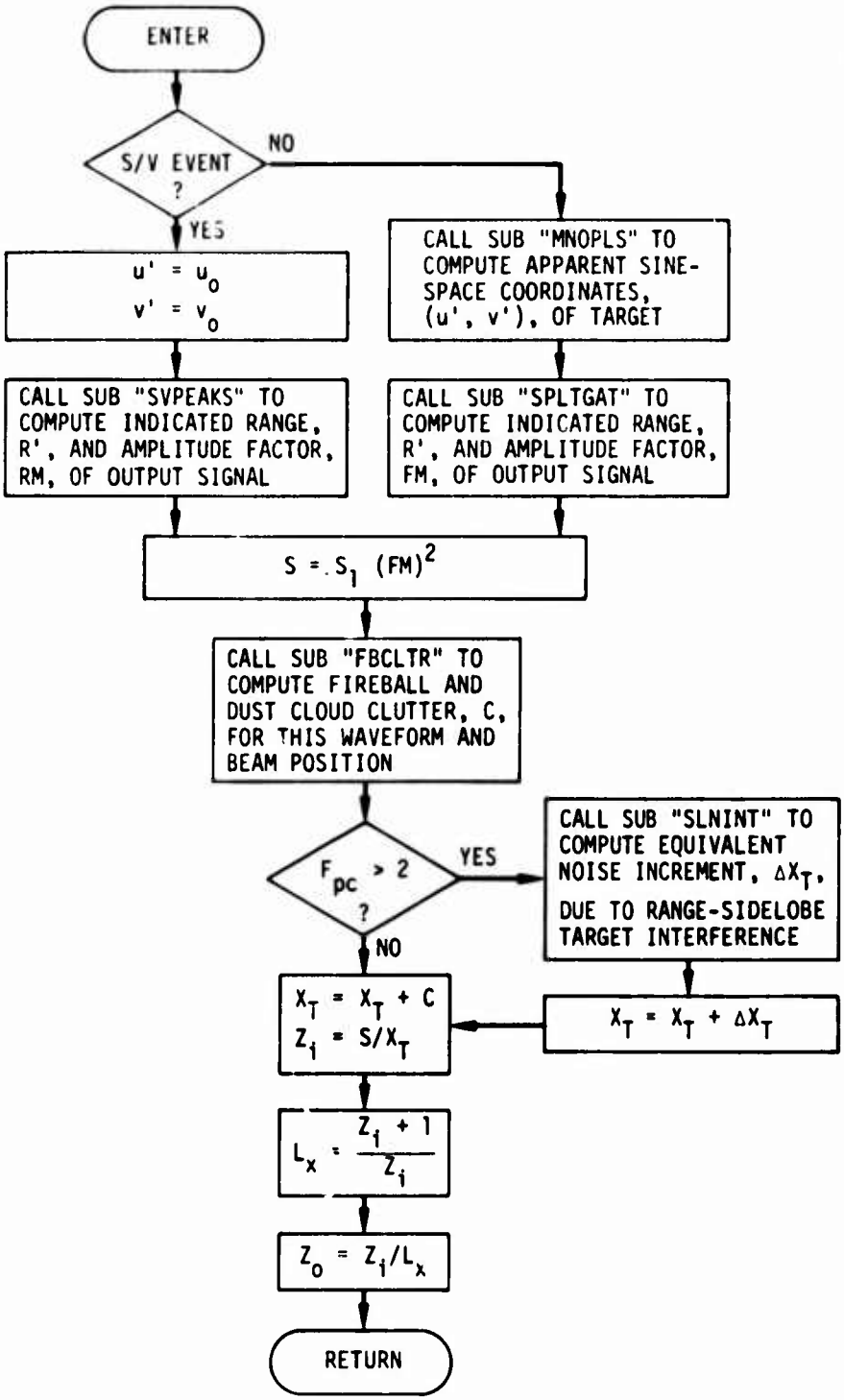


Figure 4.13. Subroutine TARGM1S

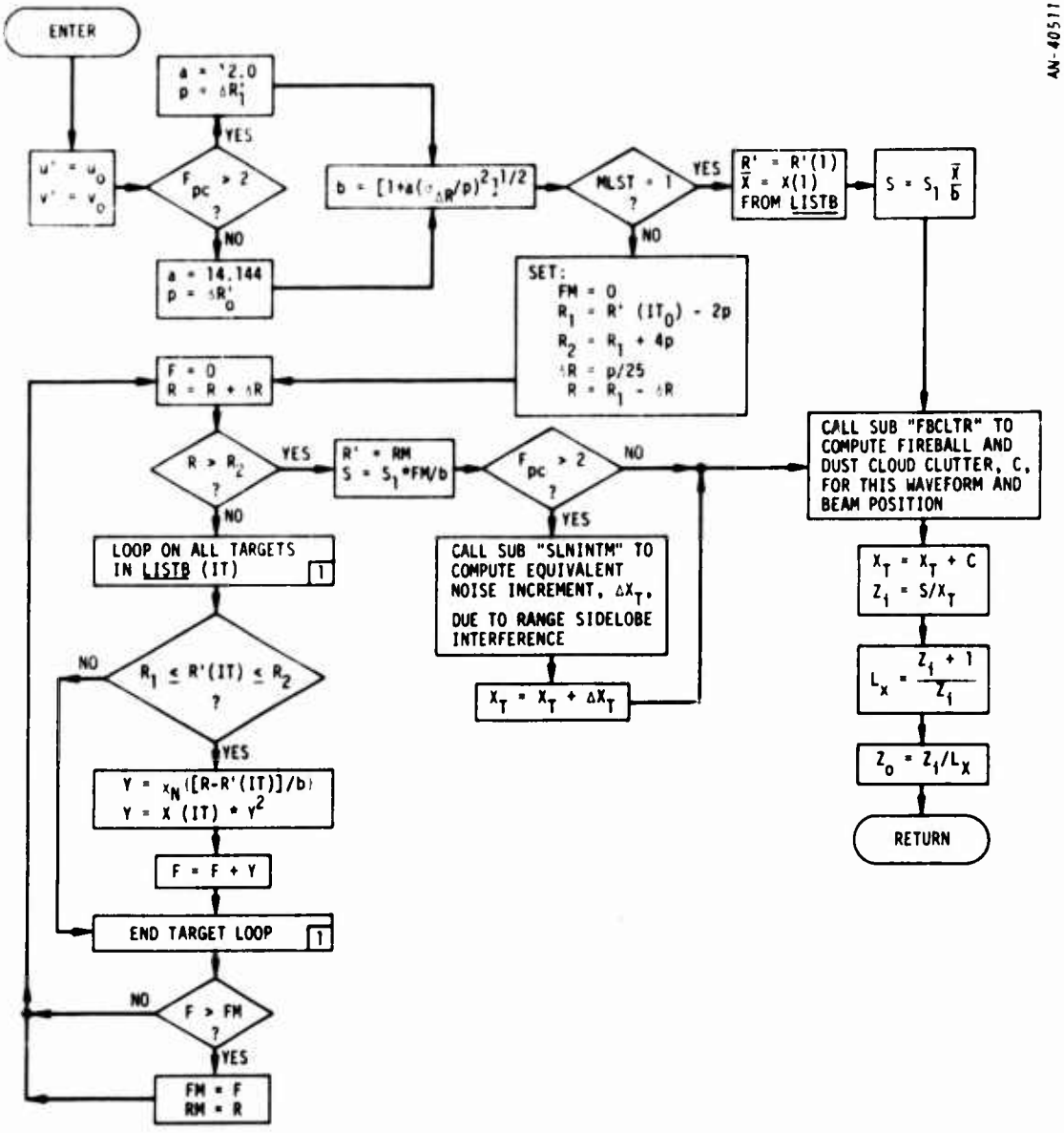
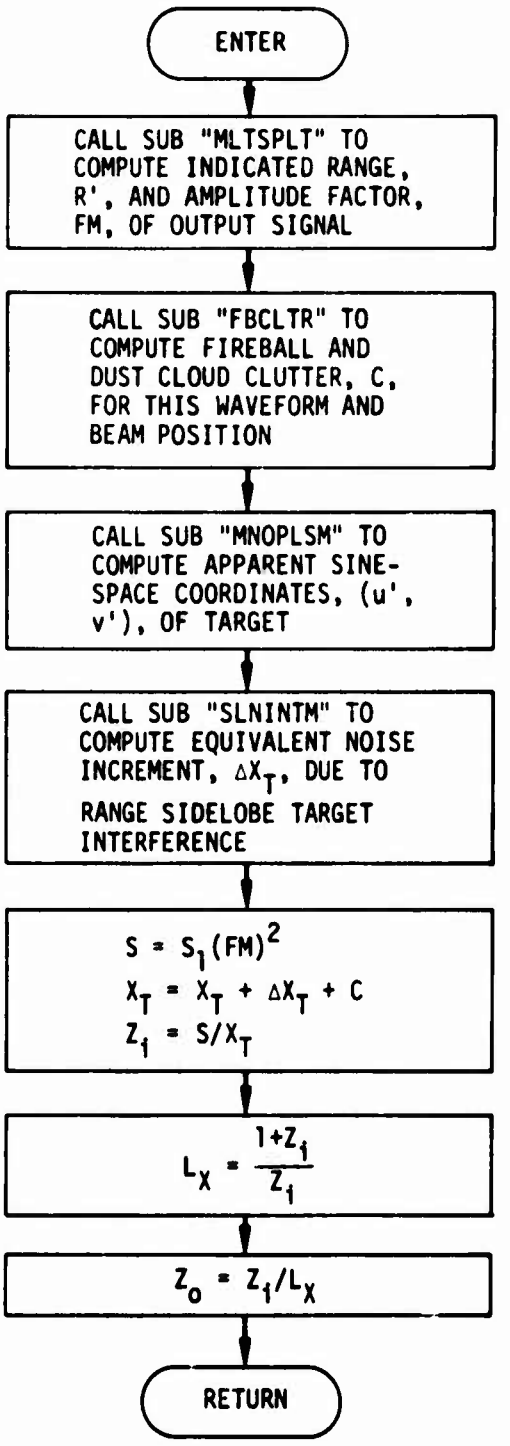


Figure 4.14. Subroutine TARGMSV



AN-39500

Figure 4.15. Subroutine TARGMTS

above, a simplified statistical model of the interaction between the "spread target cloud" and the radar is used. Monopulse and split-gate range marking are simulated in this way, and range-sidelobe interference is modeled by an equivalent noise increment.

Inputs: S_1, X_T

Outputs: (R',u',v'), Z_o

4.6 SUBROUTINES CALLED BY TARGET-POSITION SUBROUTINES

As Fig. 4.1 and the preceding flow charts showed, the "TARG" subroutines call a number of other subroutines to help compute the apparent target position and signal-to-noise-plus-clutter ratio. These may be grouped as follows:

- Monopulse angle output: MNOPLS1, MNOPLS, MNOPLSM
- Split-gate range output: SPLTGAT, MLTSPLT
- Range output for ungated signals: SVPEAKS
- "Noise" due to targets in range sidelobes: SLNINT, SLNINTM

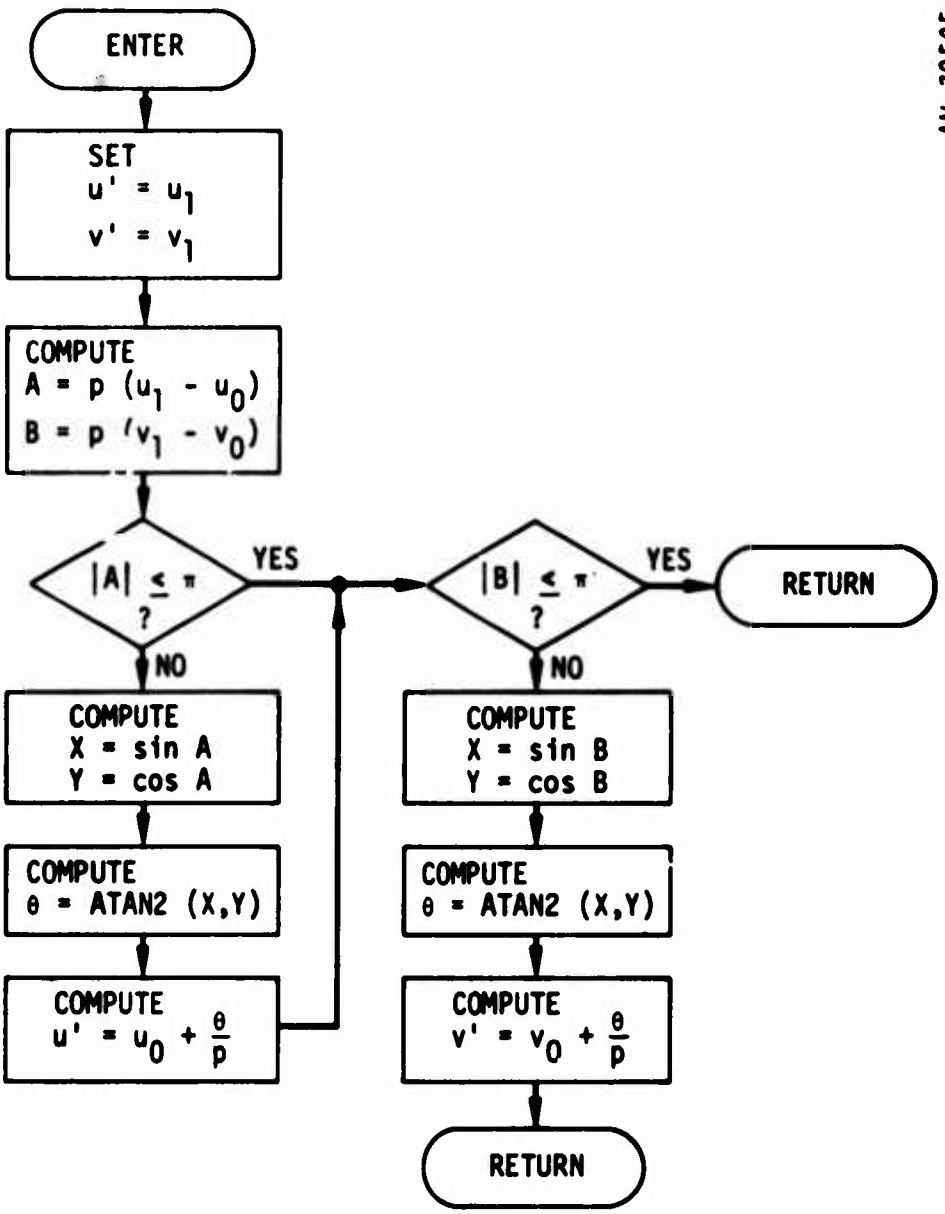
Functions AMBGN and QINV are in turn called by some of these subroutines. The following sections describe these subroutines and functions in the order just given. Subroutine FBCLTR, which computes the clutter contribution to Z_o, is described with the propagation routines, in Sec. 5.4.

4.6.1 Subroutine MNOPLS1 (Fig. 4.16)

This subroutine computes the apparent sine-space coordinates output by the monopulse system when only a single target is present in LISTA. If the target point lies within the antenna pattern mainlobe, the monopulse indication is identical with the target's angular position. However, if the target point lies outside the mainlobe, the monopulse system would indicate a false position within the mainlobe, and this subroutine computes this apparent indicated position.

Inputs: (u_1,v_1), p, (u_o,v_o)

Outputs: (u', v')



AN-39505

Figure 4.16. Subroutine MNOPLS1

4.6.2 Subroutine MNOPLS (Fig. 4.17)

This subroutine simulates the operation of the radar monopulse when there is more than one target in LISTA. It computes the in-phase and quadrature components of the sum and difference channel signals contributed by each target, adds them, and calculates the apparent sine-space coordinates of the indicated "target." (Appendix C gives details.) Only one beam shape is provided for: a circular beam with tapered sidelobe level.

Inputs: MLST, LISTA, R_o', $\delta R_1'$, δw, (u_o, v_o), p, f_o

Outputs: (u', v')

4.6.3 Subroutine MNOPLSM (Fig. 4.18)

This subroutine performs the same function in the case of refraction through a striated region ($N_p > 1$) as is performed by subroutine MNOPLS for the single-refraction case ($N_p = 1$). Using a simplified statistical model of the "spread target cloud," it simulates the operation of the radar monopulse on all the target clouds in LISTB. Because of the statistical nature of the target model, the sum and difference channel signals are calculated by using a random phase approximation for the individual contributions of the target elements (see Appendix C). The outputs are the apparent sine-space coordinates of the indicated "target." As with MNOPLS, only one beam shape is provided for: circular with tapered sidelobes.

Inputs: MLST, LISTB, R_o', $\delta R_1'$, δw, $\sigma_{\Delta R}$, p, (u_o, v_o)

Outputs: (u', v')

4.6.4 Subroutine SPLTGAT (Fig. 4.19)

This subroutine simulates the operation of a split-gate range tracker on the set of individual target images in LISTA. Since split-gate range tracking is used only in those events which also employ a pulse-compression waveform, the subroutine assumes that the signal processing is matched to the known form of the received signal return from a single target unaffected by propagation errors. However, if there are several target images in LISTA, their contributions to the received video signal output are all computed,

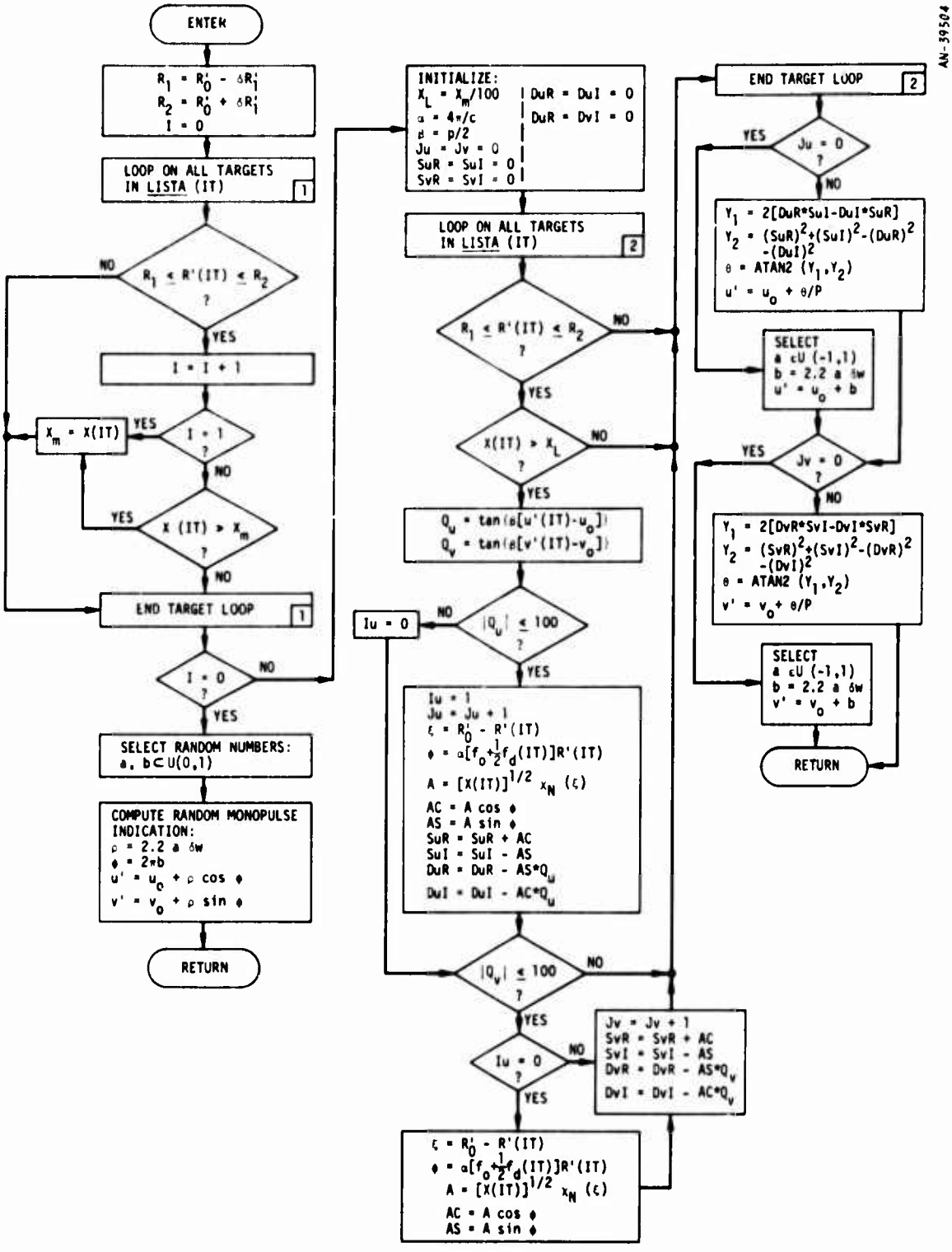


Figure 4.17. Subroutine MNOPLS

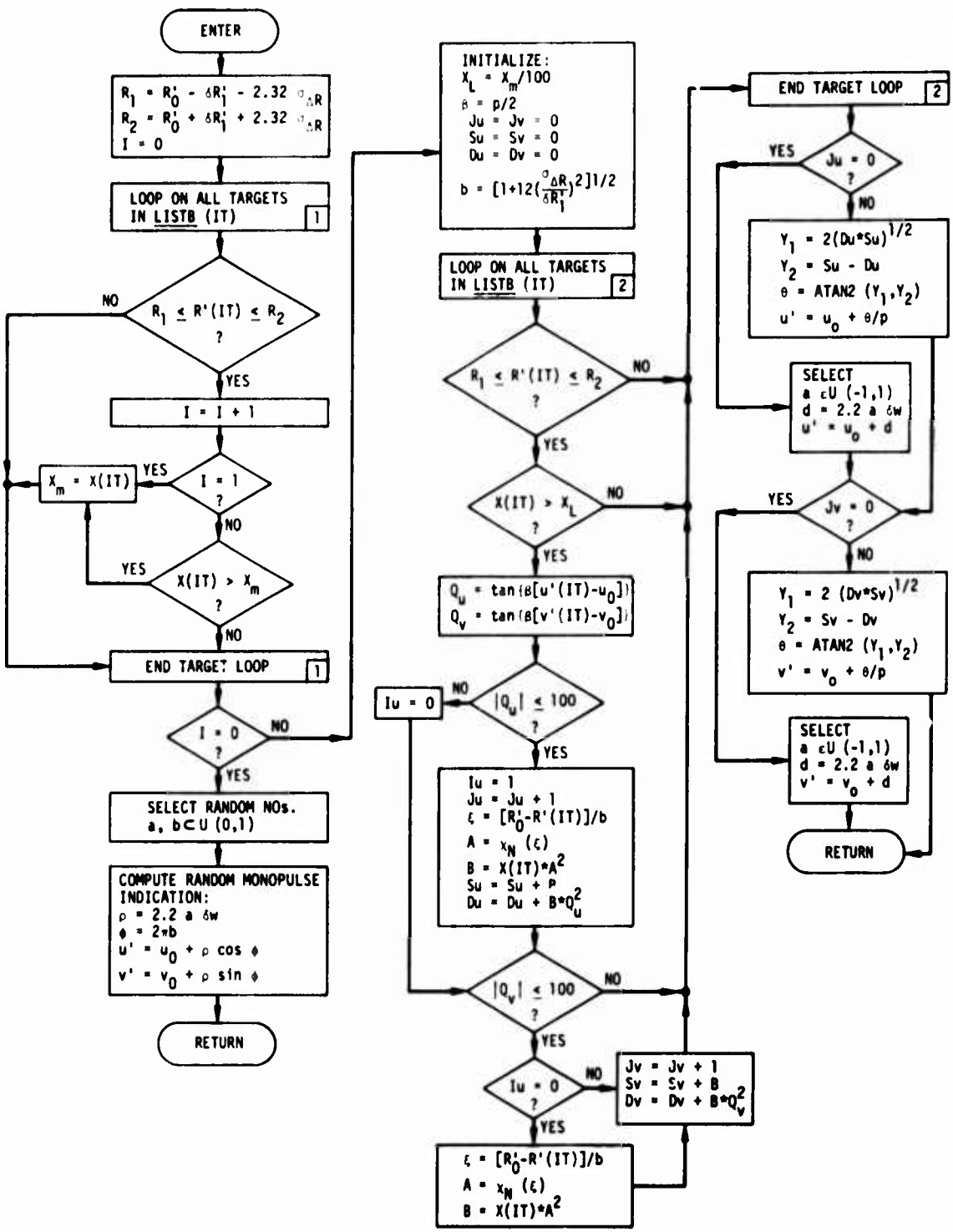


Figure 4.18. Subroutine MNOPLSM

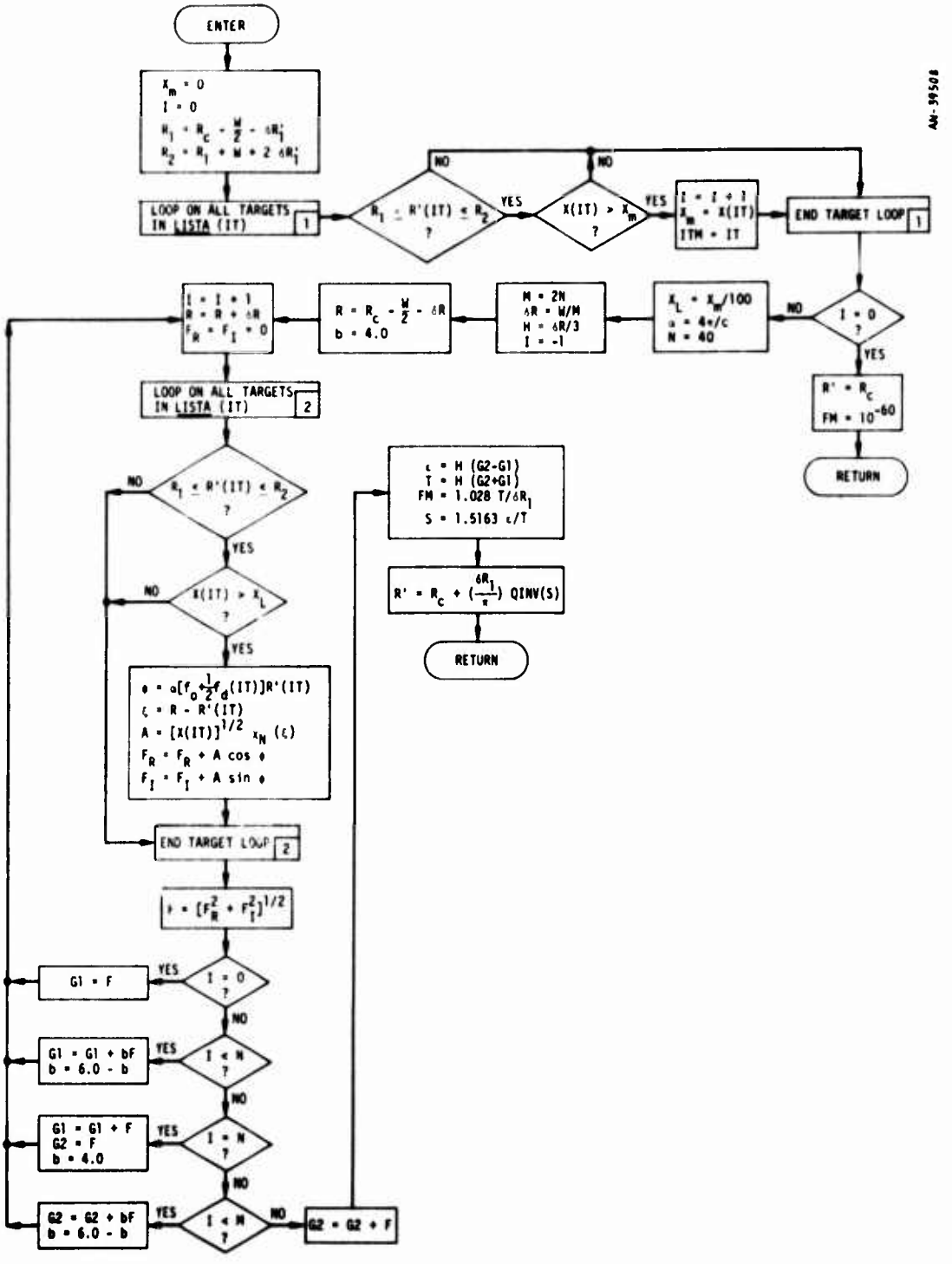


Figure 4.19. Subroutine SPLTGAT

taking into account their proper phase relationships and any dispersive effects which may have distorted the received waveform. The resulting video envelope may therefore no longer be matched to the split-gate receiver function, leading to an error in the indicated range of the "target." Since this is what would be happening in a real radar under the same conditions, the indicated range output is precisely what one desires to simulate, errors and all. The subroutine also computes the indicated signal amplitude which would be output by the split-gate system, which can be immediately converted into an indicated "peak" signal power for use in calculating the effective SNR.

Inputs: MLST, LISTA, W, R_c, $\delta R_1'$, f_o, δR_1

Outputs: R', FM

4.6.5 Subroutine MLTSPLT (Fig. 4.20)

This subroutine performs the same function in the case of refraction through a striated region ($N_p > 1$) as is performed by subroutine SPLTGAT for the single-refraction case ($N_p = 1$). Using a simple statistical model of the "spread target cloud," it simulates the operation of the split-gate range tracker on all the target clouds in LISTB. Because of the statistical nature of the target model, the random phase approximation is used in combining the contributions of different targets to the video signal. Dispersive effects on the return waveform are taken into account. The outputs of this subroutine are the indicated range and amplitude of the "target" return.

Inputs: MLST, LISTB, W, R_c, $\delta R_1'$, $\sigma_{\Delta R}$, δR_1

Outputs: R', FM

4.6.6 Subroutine SVPEAKS (Fig. 4.21)

This subroutine computes the indicated target range and peak amplitude for search and verification events, which do not use a range gate. In this case, the video signal output due to the contributions of all the targets in LISTA, combined in their correct phase relationships and taking

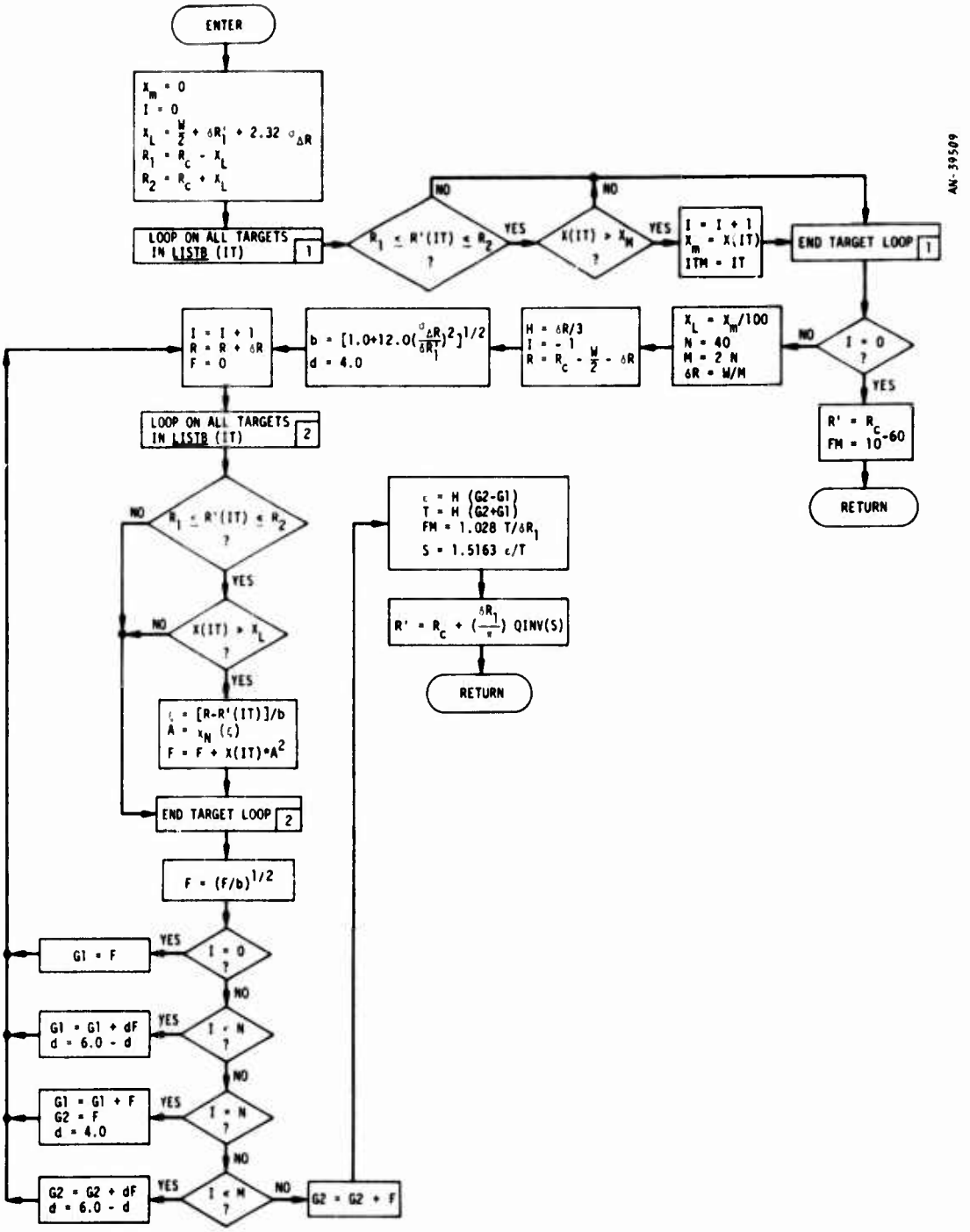


Figure 4.20. Subroutine MLTSPLT

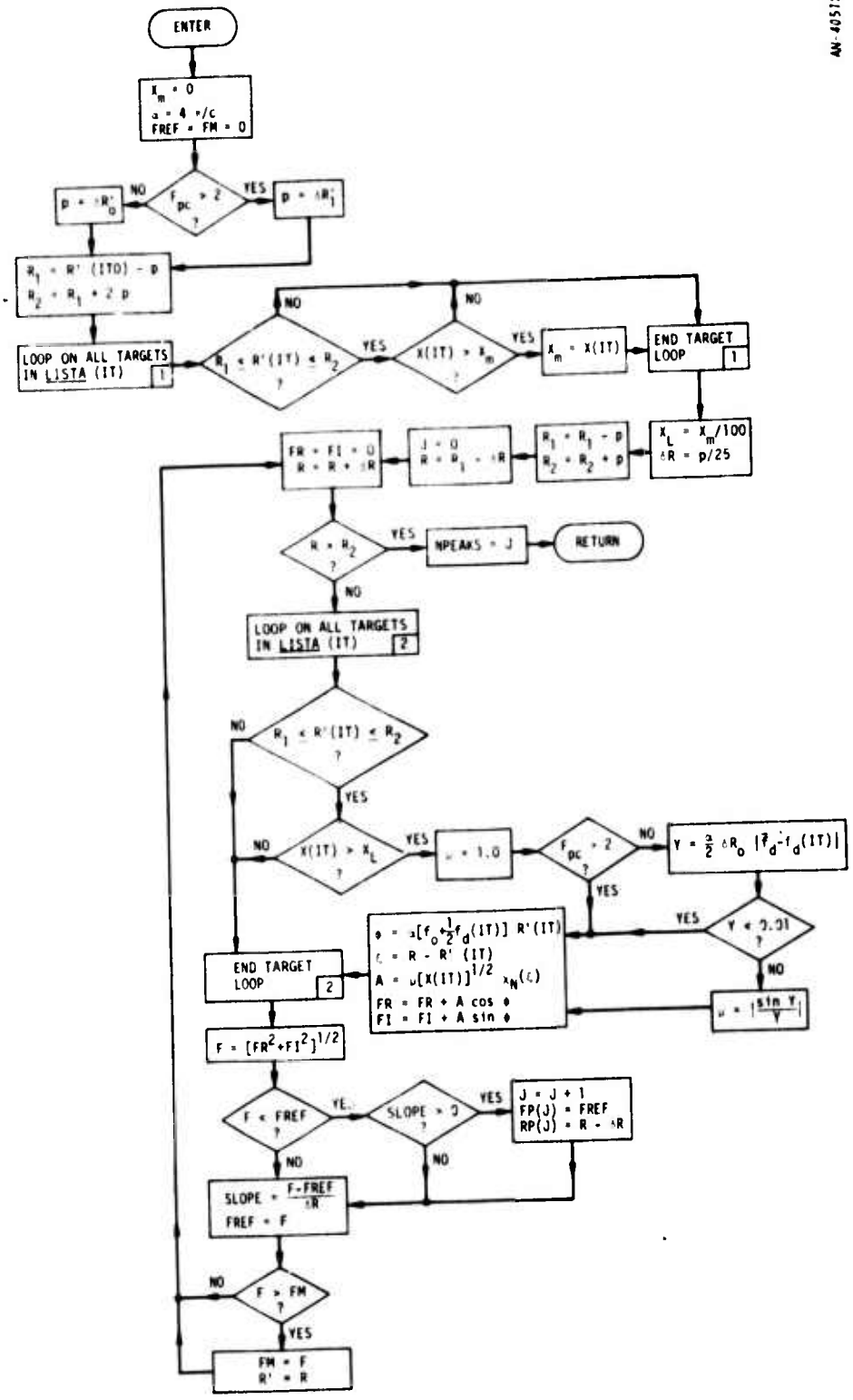


Figure 4.21. Subroutine SVPEAKS

into account any dispersive distortions of the received waveform, is computed and sampled at intervals within one range resolution distance on either side of of the primary target of current interest. The position and amplitude of the highest peak thus discovered are the outputs.

The subroutine also computes the positions and amplitudes of all other video signal peaks within the indicated region; these outputs are not used in the current version of ROSCOE but may be required by more sophisticated versions to follow.

Inputs: MLST, LISTA, ITO, $\delta R_1'$, δR_o, F_{pc}, f_o, $\bar{f}_d$

Outputs: R', FM

4.6.7 Subroutine SLNINT (Fig. 4.22)

This subroutine computes the equivalent noise increment, ΔX_T , which represents the effects of all individual target images in the range sidelobes. Such targets are considered to contribute too little to be worth including them in the calculation of the video signal; instead, they are each assumed to add to the noise a term proportional to their peak amplitude factor, modified by the signal's current mean sidelobe-to-mainlobe ratio (taking dispersive effects, if any, into account). It is only used, of course, for signals which have range sidelobes.

Inputs: R_o', MLST, LISTA, L_D, S_{SL}, S_1, R_c, W, $\delta R_o'$, $\delta R_1'$

Outputs: ΔX_T

4.6.8 Subroutine SLNINTM (Fig. 4.23)

This subroutine performs the same function in the case of refraction through a striated region ($N_p > 1$) as is performed by subroutine SLNINT for the single-refraction case ($N_p = 1$). Using a simple statistical model of the "spread target cloud," it computes the equivalent noise increment, ΔX_T , which represents the effects of all those clouds or parts of clouds which lie in the range sidelobes. It is only used, of course, for signals which have range sidelobes.

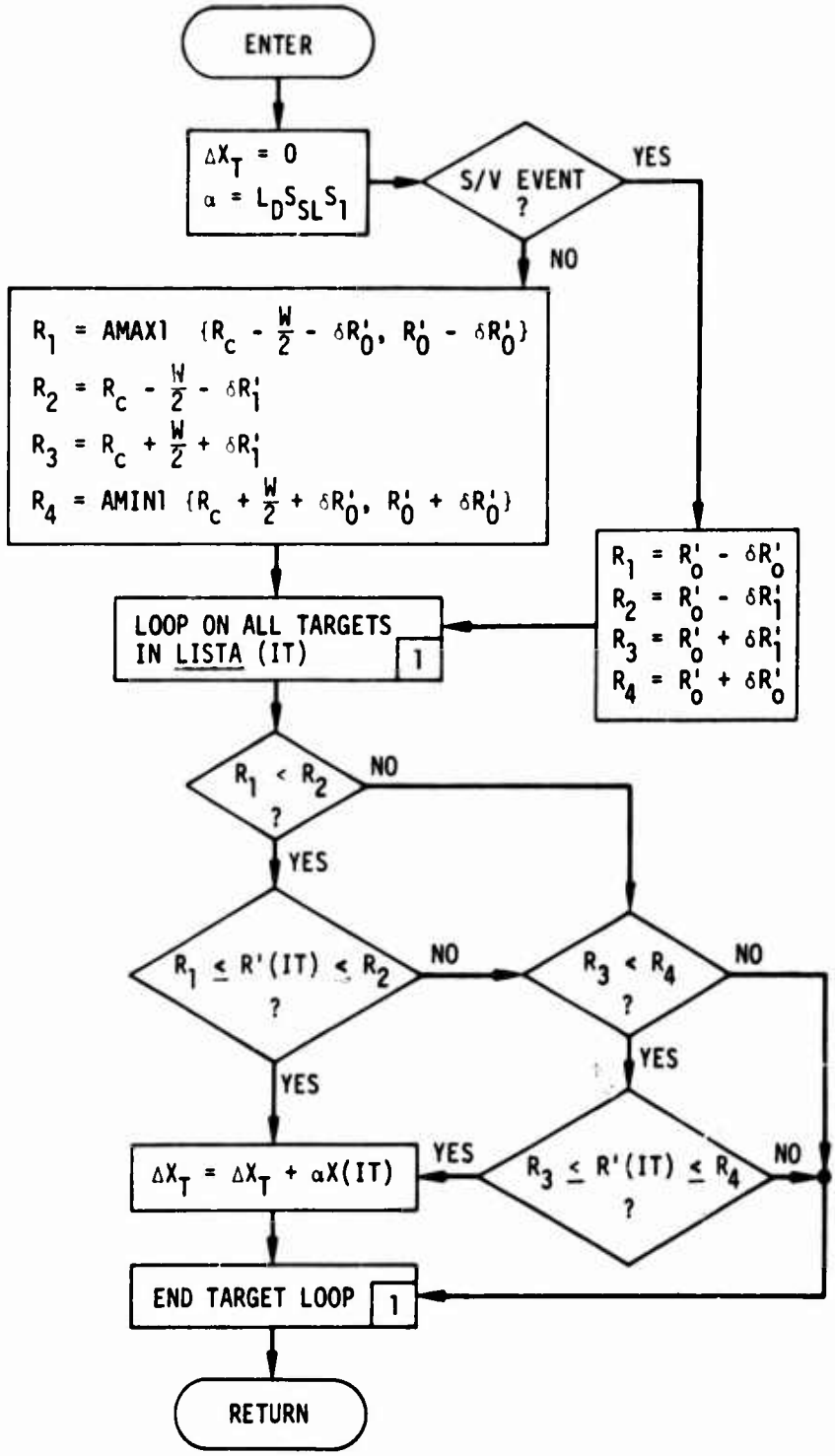


Figure 4.22. Subroutine SLNINT

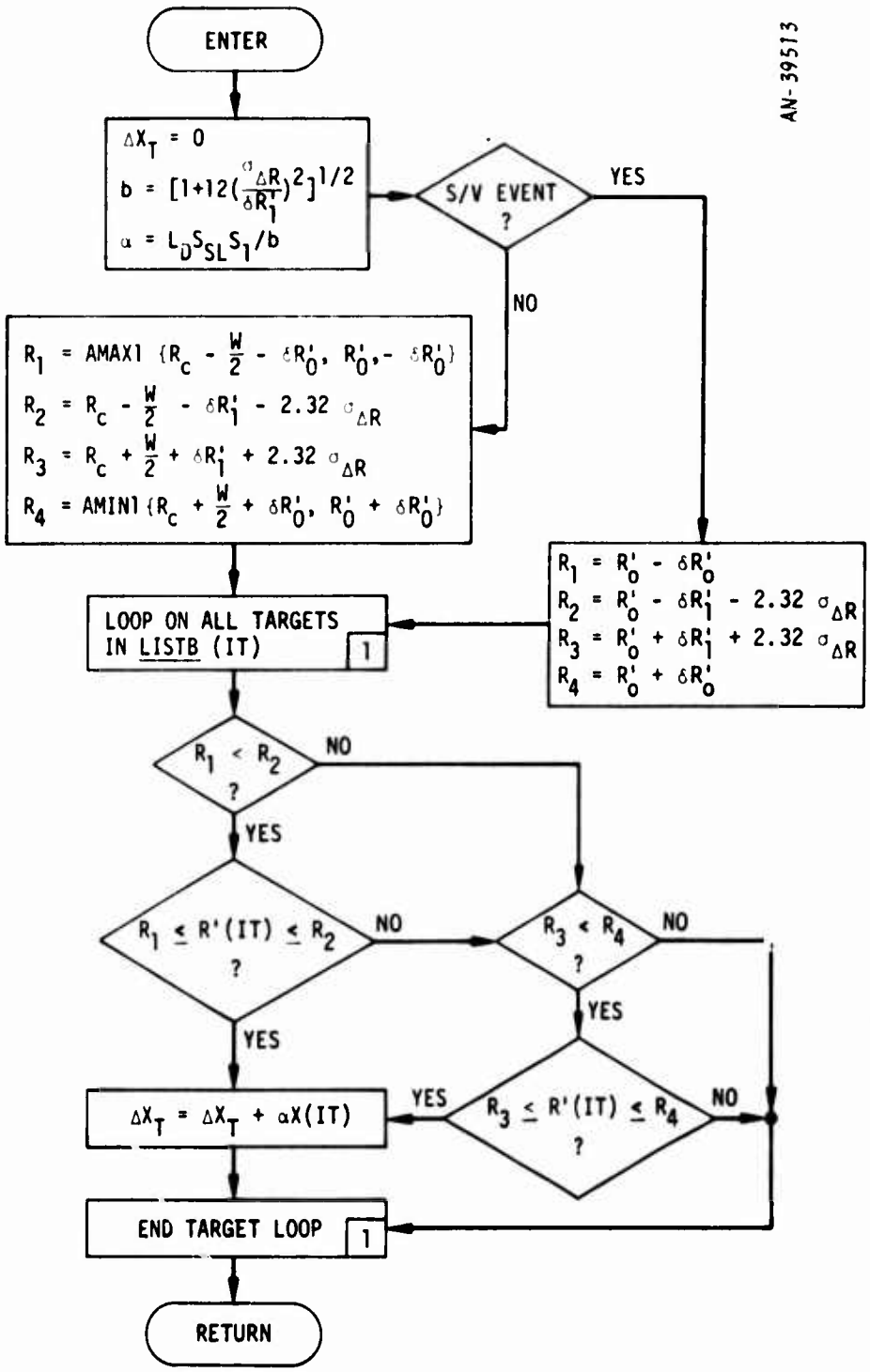


Figure 4.23. Subroutine SLNINTM

Inputs: R'_o, MLST, LISTB, L_D, S_{SL}, S_l, $\sigma_{\Delta R}$, $\delta R'_o$, $\delta R'_l$, W, R_c

Outputs: ΔX_T

4.6.9 Function AMBGN(X) (Fig. 4.24)

This function computes the normalized ambiguity function, $\chi_N(X)$, for a given waveform, as a function of range, X, from the center of the function. Dispersive effects are included, and it is normalized so that $\chi_N(0) \equiv 1$. Two different kinds of waveforms are defined: a simple rectangular pulse without frequency modulation ($F_{pc} = 1$), and a pulse-compression signal ($F_{pc} >> 1$) whose characteristics fall roughly between those of Hamming- and Hanning-weighted waveforms. It should also be noted that the ambiguity function is computed only as a function of range: in effect, it is assumed that ϕ (the doppler frequency difference) = 0; wherever doppler mismatch effects are required they are calculated separately. The reason for this is that function AMBGN attempts to take into account the waveform distortions caused by dispersion along the propagation path(s). Although this can be done for the range coordinate, an exact formulation for the dispersive effects on the doppler coordinate has yet to be derived; an approximation, which separates the two dimensions of χ_N, is currently used.

Inputs: X, F_{pc}, $\delta R'_o$, $\delta R'_l$, A_D, L_D, S_{SL}

Outputs: χ_N

4.6.10 Function QINV(s) (Fig. 4.25)

This function is used by the split-gate range tracking subroutines, SPLTGAT and MLTSPLT, to locate the apparent target point with respect to the center of the range gate. This function is specifically matched to the weighted waveform ($F_{pc} >> 1$) mentioned above, and would give the correct range of the target point if there were only a single target in the resolution cell and if there were no dispersive distortion of the signal. In the presence of either or both of these "anomalies," an incorrect position is computed for the target, which is just what a real split-gate range tracker would do under the same circumstances. See Appendix C for details.

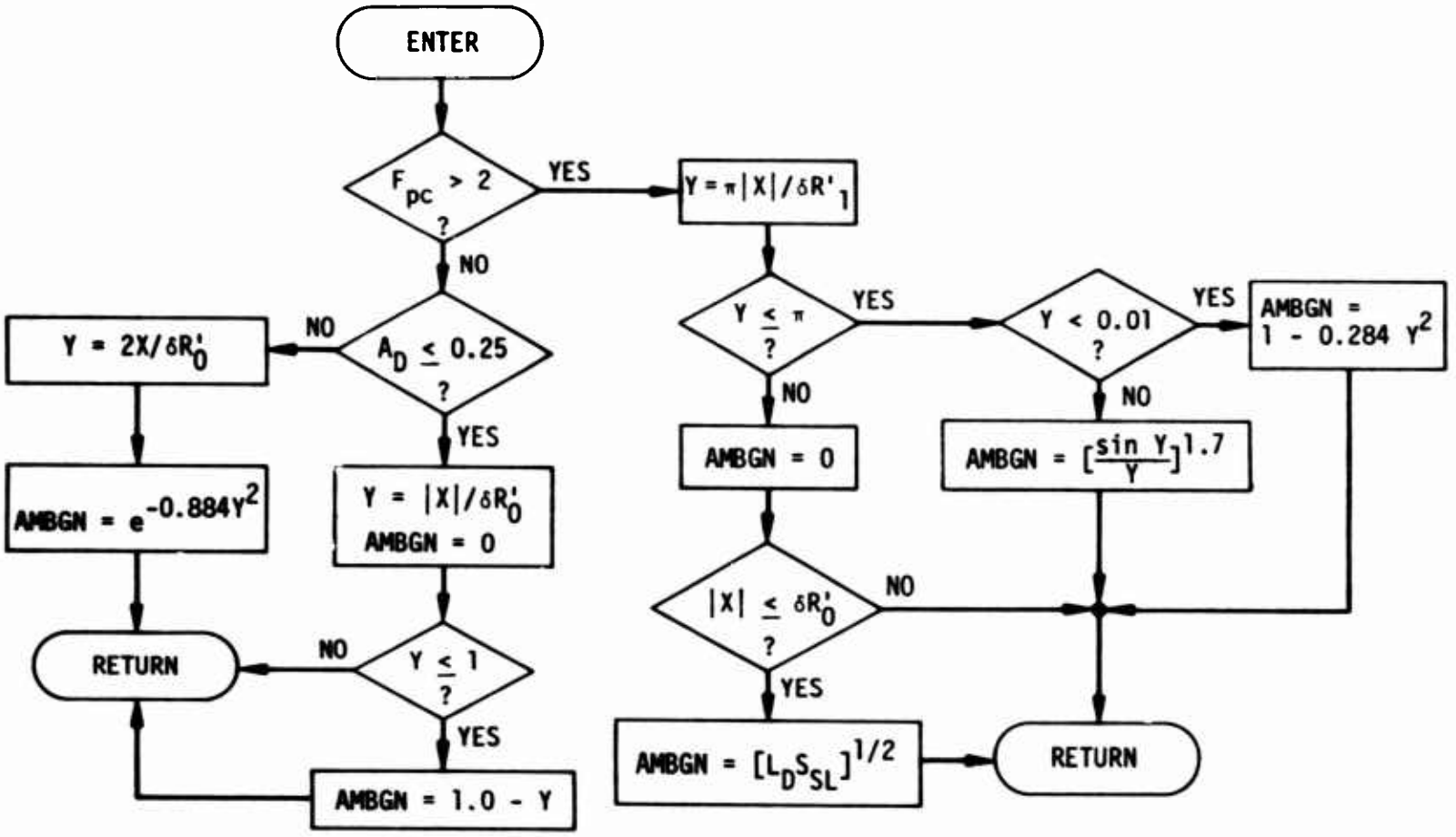


Figure 4.24. Function AMBGN

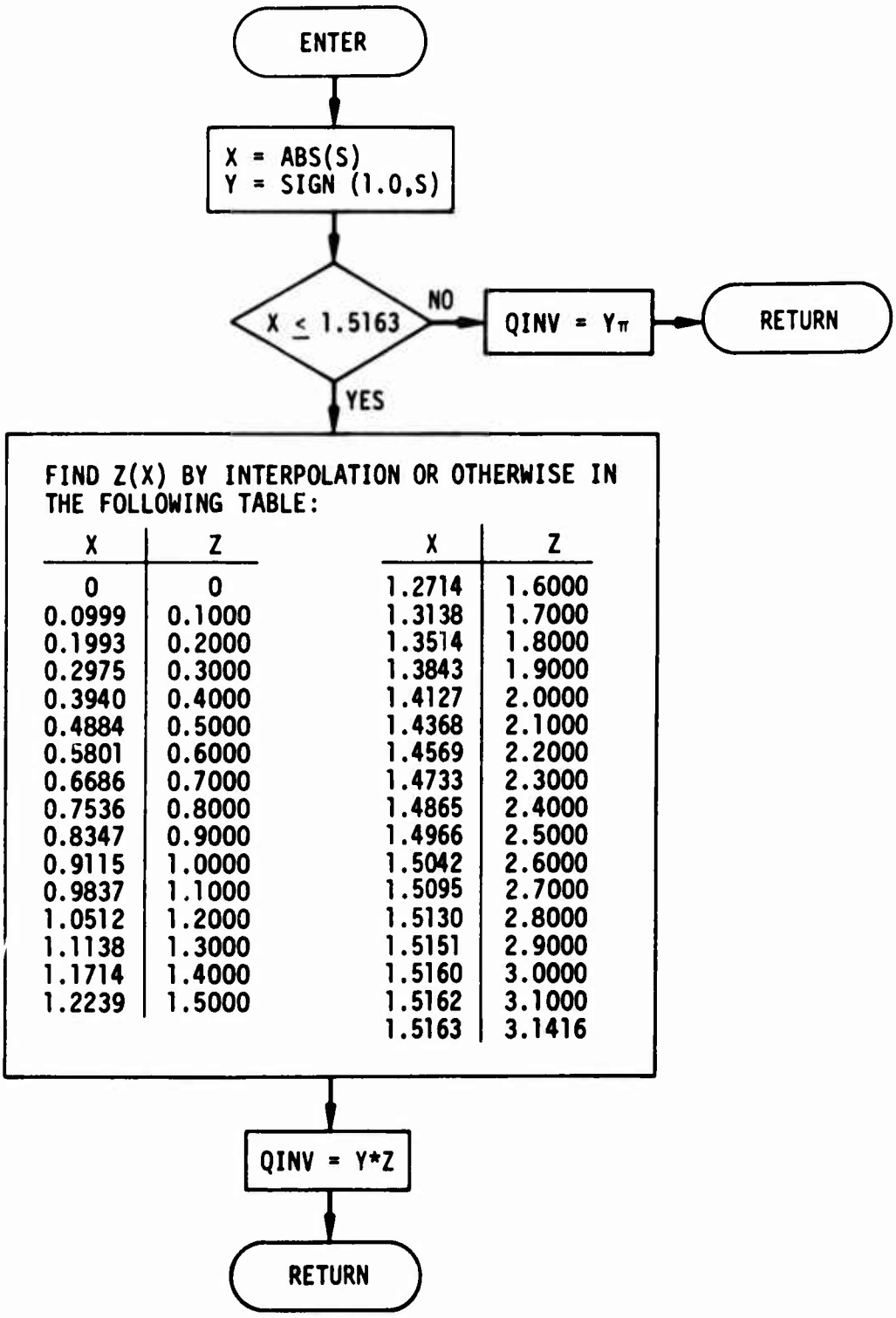


Figure 4.25. Function QINV

Inputs: s

Outputs: QINV

4.7 THRESHOLDING AND RADAR MEASUREMENT ERRORS

As explained earlier, the output of subroutine RADMODS is the "apparent" target coordinates (R',u',v'), which include the effects of the environment on the radar's measurements but do not include the inherent errors of the radar itself. These are added by another subroutine, MEASERR, called not by RADMODS but by the event subroutine that called RADMODS.

A final subroutine, XTHRSHS (also called by the event subroutine), compares the RADMODS output Z_o with a threshold signal-to-noise-plus-clutter ratio.

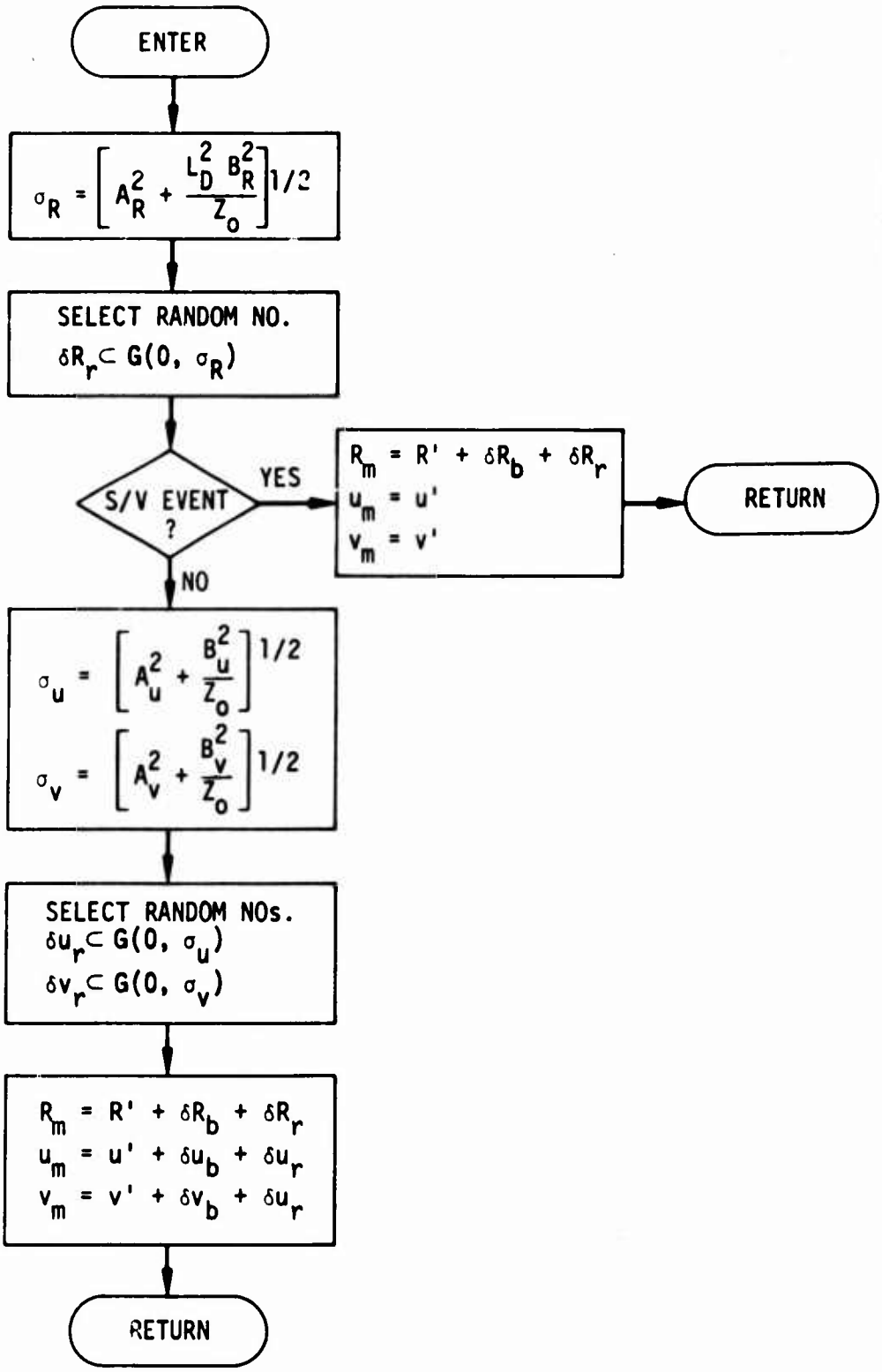
4.7.1 Subroutine MEASERR (Fig. 4.26)

This subroutine just adds radar measurement and bias errors to the apparent target coordinates which are output by subroutine RADMODS. The calculation involves selecting random samples from appropriate zero-mean distributions having variances given by a standard radar formula, and adding fixed biases as required.

Since the angular coordinates are not measured in search and verification events, but are merely set equal to the commanded receive beam axis coordinates (u_o,v_o), subroutine MEASERR does not compute any measurement errors for these coordinates for these events. Then we have just

$$u_m \equiv u' \quad \text{and} \quad v_m \equiv v'$$

The bias errors, δR_b, δu_b, δv_b, remain constant <u>for each radar</u> throughout a given computer run. They may be input directly by the user. An alternative procedure is to input the standard deviations of their distributions for each radar type; then at the beginning of each run the values of δR_b, δu_b, δv_b to be used for each radar during that run may be drawn



AN-39515

Figure 4.26. Subroutine MEASERR

as random samples from the appropriate zero-mean Gaussian distributions. In either case, the values of these bias errors for each radar are stored in appropriate datasets for use in subroutine MEASERR.

The random measurement errors, δR_r, δu_r, δv_r, are sampled each time from zero-mean Gaussian distributions with standard deviations σ_R, σ_u, σ_v which are computed within subroutine MEASERR. The variances of these distributions are found from a formula of standard type:

$$\sigma_q^2 = A_q^2 + \frac{\rho_q^2 B_q^2}{Z_o}$$

where "q" stands for R, u, or v, respectively. Here Z_o is the output signal-to-noise-plus-clutter ratio from RADMODS. The six constants $\{A_q, B_q\}$ are input by the user for each radar type; their values are usually known as a result of routine error analyses of a given radar type.

The factor ρ_q in the above equation is related to the spread in effective range resolution due to dispersion. It obviously does not apply to angular measurements; hence, $\rho_u = \rho_v \equiv 1$. For range measurements, it may be shown that $\rho_R \simeq L_D$, where L_D is the dispersive loss factor calculated in subroutine REFIS for the current radar waveform under present conditions.

The outputs of this subroutine are (R_m, u_m, v_m). Note that in Sec. 3, these outputs were denoted in different ways for different event types:

- Search: R_S
- Verification: (R_V, u_V, v_V)
- Track Initiation: (R_1, u_1, v_1) and (R_2, u_2, v_2) as appropriate
- Track: (R_m, u_m, v_m)

Inputs: Z_o, L_D, (R',u',v'), $\{A_R,A_u,A_v,B_R,B_u,B_v\}$, $\{\delta R_b,\delta u_b,\delta v_b\}$

Outputs: (R_m,u_m,v_m)

4.7.2 Subroutine XTHRSHS (Fig. 4.27)

This subroutine determines, in a very simplified manner, whether the output signal-to-noise-plus-clutter ratio (Z_o) computed by subroutine RADMODS does or does not cross a predefined detection threshold. The determination is made simply by computing the output value equivalent to the input threshold $(S/N)_{Ti}$ which is one of the parameters associated with the basic waveform/event dataset, and then comparing Z_o with this value. The output, XTH , is a Hollerith-valued constant, either NO or YES.

Inputs: IFDET, Z_o, $(S/N)_{Ti}$

Outputs: XTH

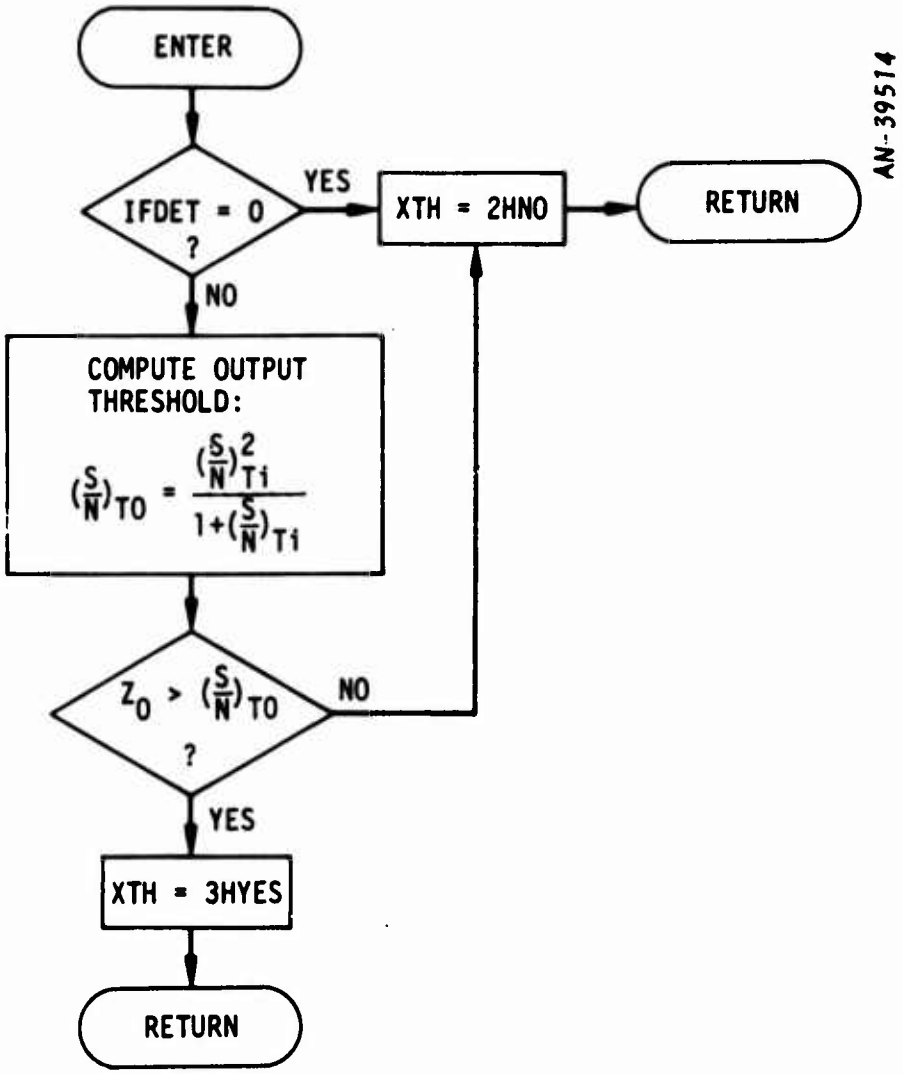


Figure 4.27. Subroutine XTHRSHS

5 PROPAGATION

The following propagation models have been included in ROSCOE:

1. Absorption
2. Fireball Noise
3. Refraction
4. Clutter
5. Low-Altitude Fireball Multipath

5.1 ABSORPTION

Absorptions due to fireballs, betas, delayed gammas and neutrons, and prompt X-rays and neutrons are computed when subroutine ABSORB is called in the program. This routine in turn calls a series of subroutines which perform the individual absorption calculations. The calculations are terminated if at any time the integrated absorption (DBSUM) exceeds a specified threshold (DBMAX).

5.1.1 Assumptions

The attenuation calculation consists of a numerical integration of the non-deviative incremental absorption equation

$$\frac{\partial A}{\partial r} = \frac{7.3 \times 10^{-8} N_e (\nu_{em} + \nu_{ei})}{f^2 + (0.775\nu_{em} + \nu_{ei})^2} \text{ dB/cm}$$

where N_e = electron density, cm^{-3}

f = radar frequency, MHz

ν_{em}, ν_{ei} = average electron-neutral and electron-ion collision frequencies divided by $2\pi \times 10^6$.

The non-deviative assumption, although not strictly valid, is made to reduce computation time. It permits one to decouple the absorption and

refraction calculations so that a relatively fast absorption computation can be made along the path of interest.

The error introduced by this assumption is expected to be small except under severe refractive conditions; under these conditions, the system is probably seriously degraded by refractive errors to the extent that a precision absorption calculation is unwarranted.

5.1.2 Procedure

Subroutine ABSORB computes absorption between two points due to all significant radiation sources. This subroutine calls FBABS to compute fireball and dust absorption, and DELABS to compute absorption due to all other sources.

Fireball absorption is computed by simply integrating the incremental absorption as the radar line-of-sight intersects each fireball region. A more complex integration procedure is used in DELABS. The scheme follows these steps:

1. An array of mandatory points on the path of interest is found where absorption may be significant (e.g., for fireball absorption the point of closest approach to the fireball center is used).
2. The mandatory points are ordered in range from the radar--shortest range first.
3. The integration is started at the first mandatory point and proceeds down in range until one of the following occurs: (a) range = 0; (b) incremental absorption falls below some minimum threshold; (c) incremental absorption is less than 5% of the total integrated absorption.
4. The integration is then continued up in range from the first mandatory point until either the maximum range is reached (e.g., the range to the target) or 3(b) or 3(c) is satisfied.

5. For subsequent mandatory points that have not been included in the above integration, the same procedure is followed, except that when integrating down the path the minimum or limiting range is the last integrated point rather than 0.

In the integration procedure, a variable integration step size is used depending on the magnitude of the current and last computed values of incremental absorption. The increment in slant range DELINC is given by:

$$\text{DELINC} = \text{AMIN1}(\text{AMAX1}(\text{DELINC}' * \text{DBINC1}/\text{DBINC}, \text{DSMIN}), \text{DSMAX})$$

where

DSMIN = .5 * DSNOM

DSMAX = 2 * DSNOM

DSNOM = AMIN1(AMAX1(DSNOM', $.1 \times 10^5$), 5×10^5)

DSNOM' = An internally set value (e.g., .05 × target altitude for fireball absorption)

DBINC1 = Incremental absorption for last calculation (taken as zero initially)

DBINC = Current calculated incremental absorption

DELINC' = DSNOM for first integration step

5.1.3 Flow Charts

Flow charts for the absorption subroutines are presented in Figs. 5.1 to 5.3.

There are a number of flags and variables used in the code to indicate integration direction and integration limits. Some of these are:

S1 = slant range to the ith mandatory point

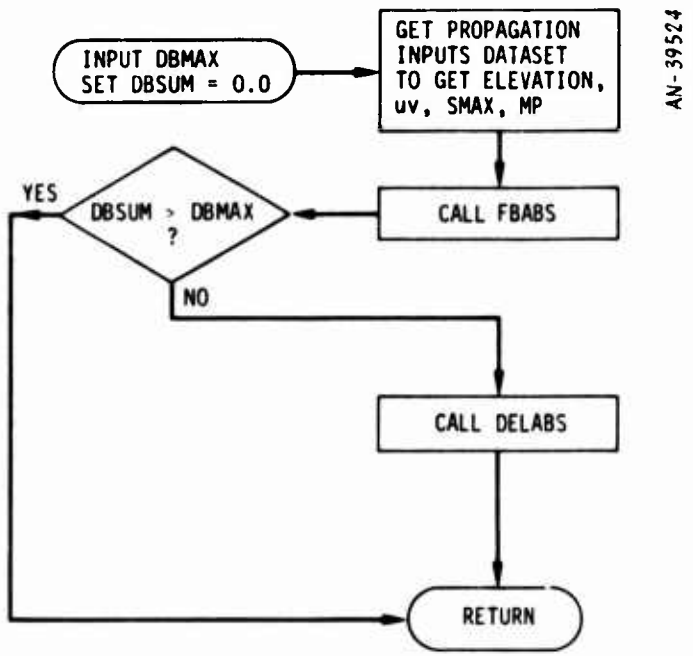


Figure 5.1. Subroutine ABSORB

S2 = maximum slant range already integrated (=0, initially)

k = integration direction (denoted DIRECTN in code) = +1, up path; -1, down path

L = flag to indicate current integration step is starting at point of closest approach (denoted ATPOINT in code) = 1, if a mandatory point; 0, otherwise

k1 = flag to indicate integration has reached lower integration limit (S2) (denoted ATSMIN in code) = 0, if at S2; 1, otherwise

5.1.4 Inputs and Outputs

Subroutine ABSORB is called by subroutine RAD1S of the radar signal processing module. RAD1S feeds ABSORB the radar position, current calculation time, radar frequency, and maximum one-way absorption

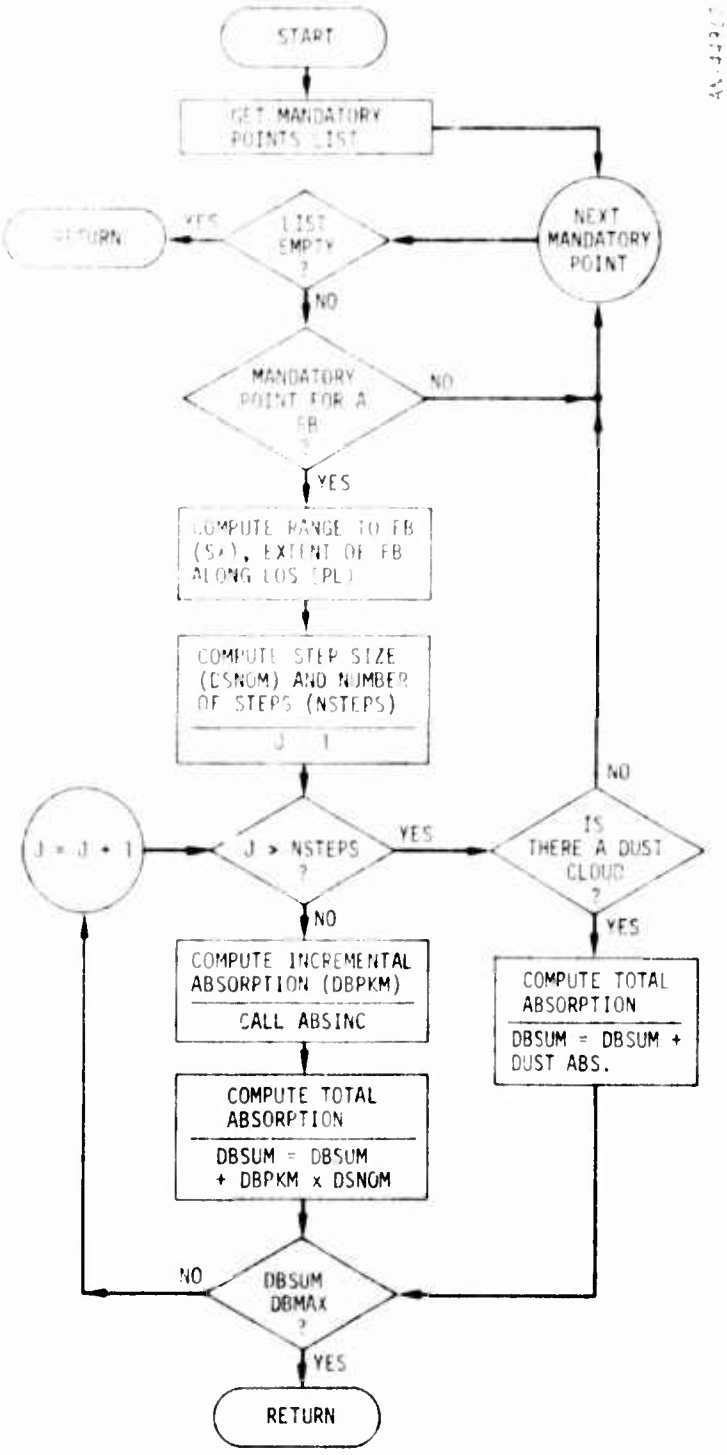


Figure 5.2. Subroutine FBABS

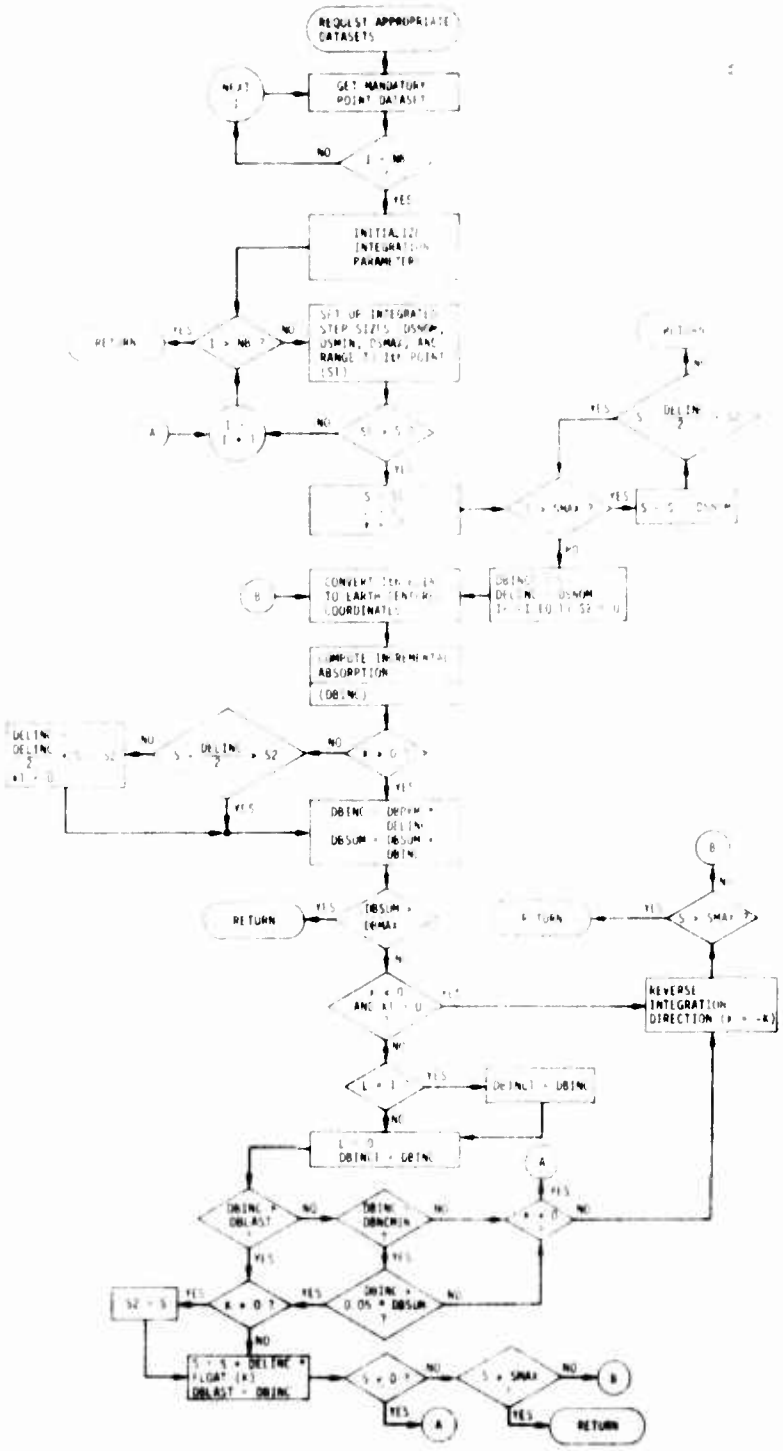


Figure 5.3. Subroutine DELABS

of interest (equal to the total dB required to reduce the signal-to-noise ratio to a value just below the detection threshold). ABSORB returns the total one-way integrated absorption over the current path of interest.

The inputs and outputs of the absorption routines are summarized as follows:

ABSORB Inputs:	DBMAX	Maximum one-way absorption of interest, dB
	TIME	Current time, seconds
	FREQ	Radar frequency, MHz
	R(3)	Radar position, cm
ABSORB Output:	DBSUM	Total integrated absorption, dB
FBABS and DELABS Inputs:	DBMAX	Maximum absorption of interest, dB
	R(3)	Radar position, cm
	UV(3)	Unit vector along radar line-of-sight
	SMAX	Maximum range of interest, cm
	ELEV	Current radar beam elevation, deg
	MP	Pointer to mandatory-points dataset
	TIME	Time, seconds
	FREQ	Radar frequency, MHz
FBABS and DELABS Output:	DBSUM	Total integrated absorption, dB

5.2 FIREBALL NOISE

All fireballs in the main beam of the radar antenna are assumed to be potential noise sources in ROSCOE. Subroutine NOISE computes the effective noise temperature along the radar line-of-sight from each fireball satisfying this condition.

The noise temperature calculation performed in ROSCOE attempts to account for non-uniform temperature and emissivity distributions through the fireball edge by numerically integrating through this region. The form of the expression for the noise temperature contribution from a particular fireball is

$$T_{ef} = 10^{-0.1A_{DB}} \times T_{eff} \times GAIN(\theta,\phi)$$

where

$$T_{eff} = \int_{\ell} T \times e \times 10^{-0.1A_{BBS}} \, d\ell$$

$$GAIN(\theta,\phi) = \iint \frac{G(\theta,\phi)}{4\pi} \sin\phi \, d\phi \, d\theta$$

and A_{DB} = cumulative absorption from outer edge of fireball to radar (or next fireball if there is more than one), dB

T = temperature at point of interest along path ℓ, °K

e = emissivity at point of interest
$= 1 - 10^{-0.1A_{DBINC}}$

A_{BBS} = cumulative absorption from outer edge of fireball to point of interest inside fireball, dB

A_{DBINC} = incremental absorption for point of interest, dB

5.2.1 Assumptions

The following assumptions are made in the NOISE model:

1. The radar beamwidth is narrow and the main contributors to the noise temperature are fireballs in the main beam.

2. Integration over the main beam pattern is used to find the effective contribution to the noise for each fireball, but the noise is propagated along a path coincident with the antenna pointing direction. This simplifies the noise absorption calculation since only a single path has to be considered for all fireballs.

3. A numerical integration is performed through the fireball edge to account for non-uniform temperature and emissivity distributions as discussed above.

5.2.2 Procedure

The noise calculation follows these steps (see Fig. 5.4):

1. An array of mandatory points is found where noise temperature may be significant.

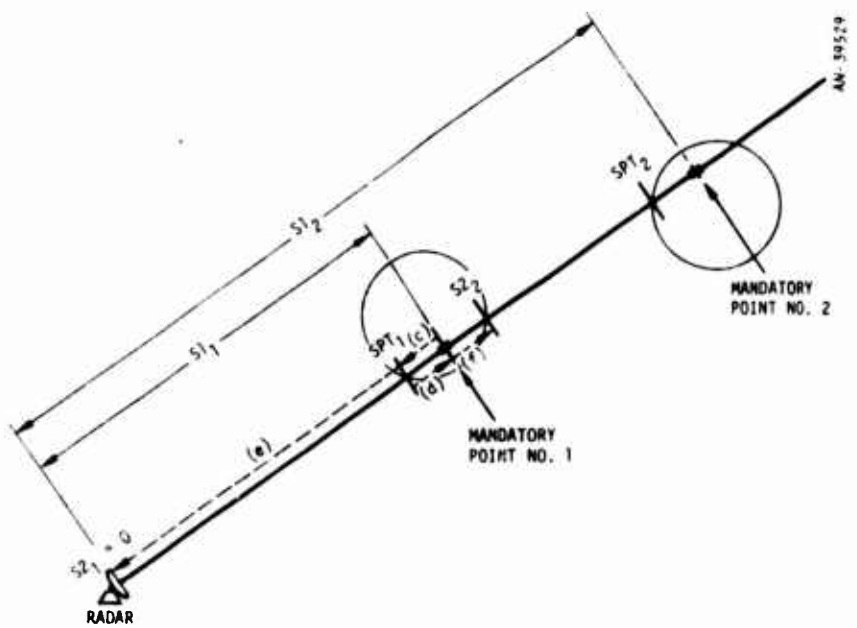


Figure 5.4. Fireball Noise Integration Procedure

2. The points are range ordered--shortest range from the radar first.

3. The calculation is started at the first point and proceeds down in range until the temperature-emissivity product drops below 10% of the system temperature. At each step along the path, the temperature-emissivity product and incremental absorption are stored--step (c).

4. After reaching this point, the calculation reverses direction and begins integrating out in range over the fireball until the starting point is reached (halfway over the fireball intersection path)--step (d).

5. The absorption is then computed from the fireball edge (SPT) to the radar (or to the next fireball down in range (S2))--step (e).

6. The procedure then returns to Mandatory Point No. 1, and continues integrating out over the fireball until the upper edge is reached (i.e., temperature-emissivity again falls below 10% of the system temperature). This point is S2 for the next sequence of calculations--step (f).

7. The procedure is repeated, beginning at the next mandatory point. This is continued until either all mandatory points are covered or the cumulative absorption along the path is larger than 30 dB.

5.2.3 Flow Chart

A flow chart of subroutine NOISE is given in Fig. 5.5. A definition of some of the flags and variables used in the routine follows:

$S1_i$ = slant range to the point of closest approach to the ith fireball (see Fig. 5.4).

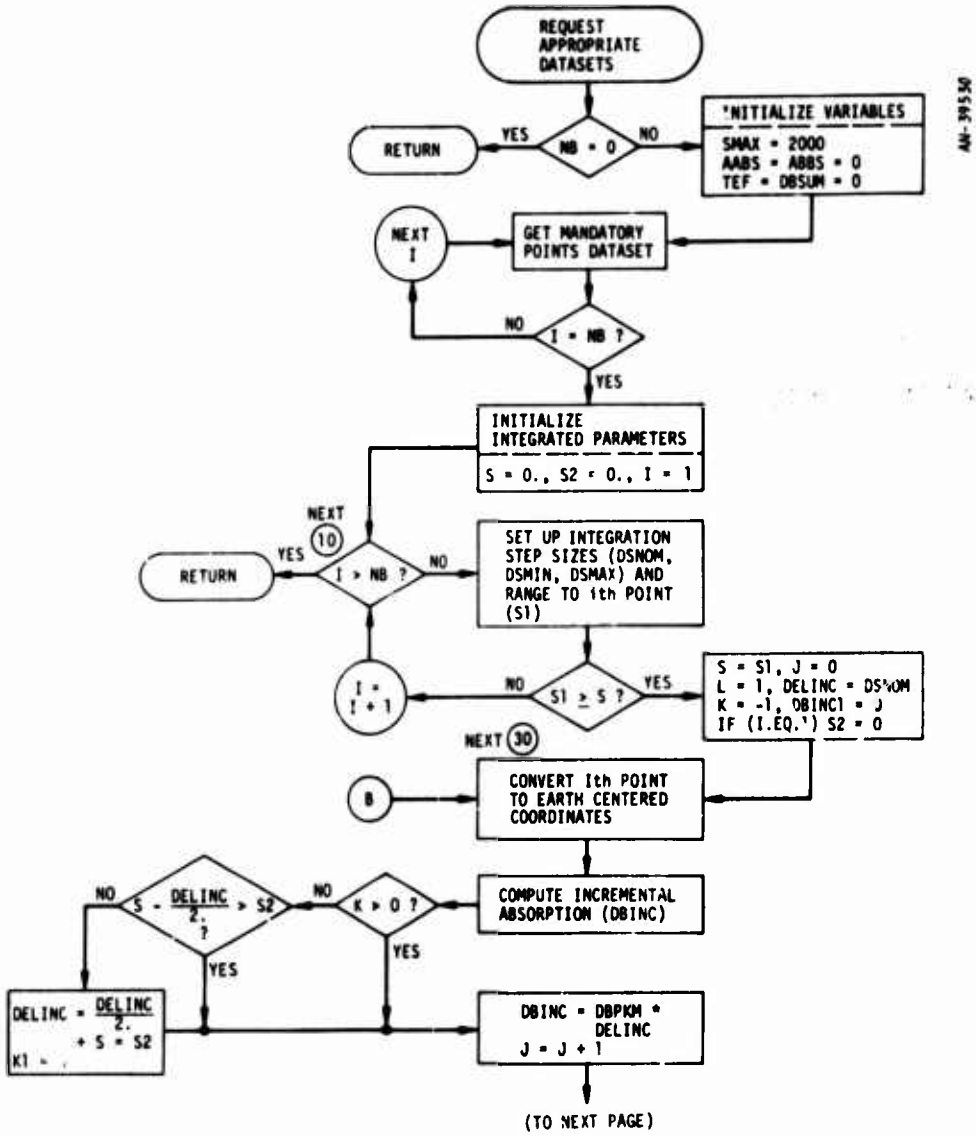


Figure 5.5. Subroutine NOISE

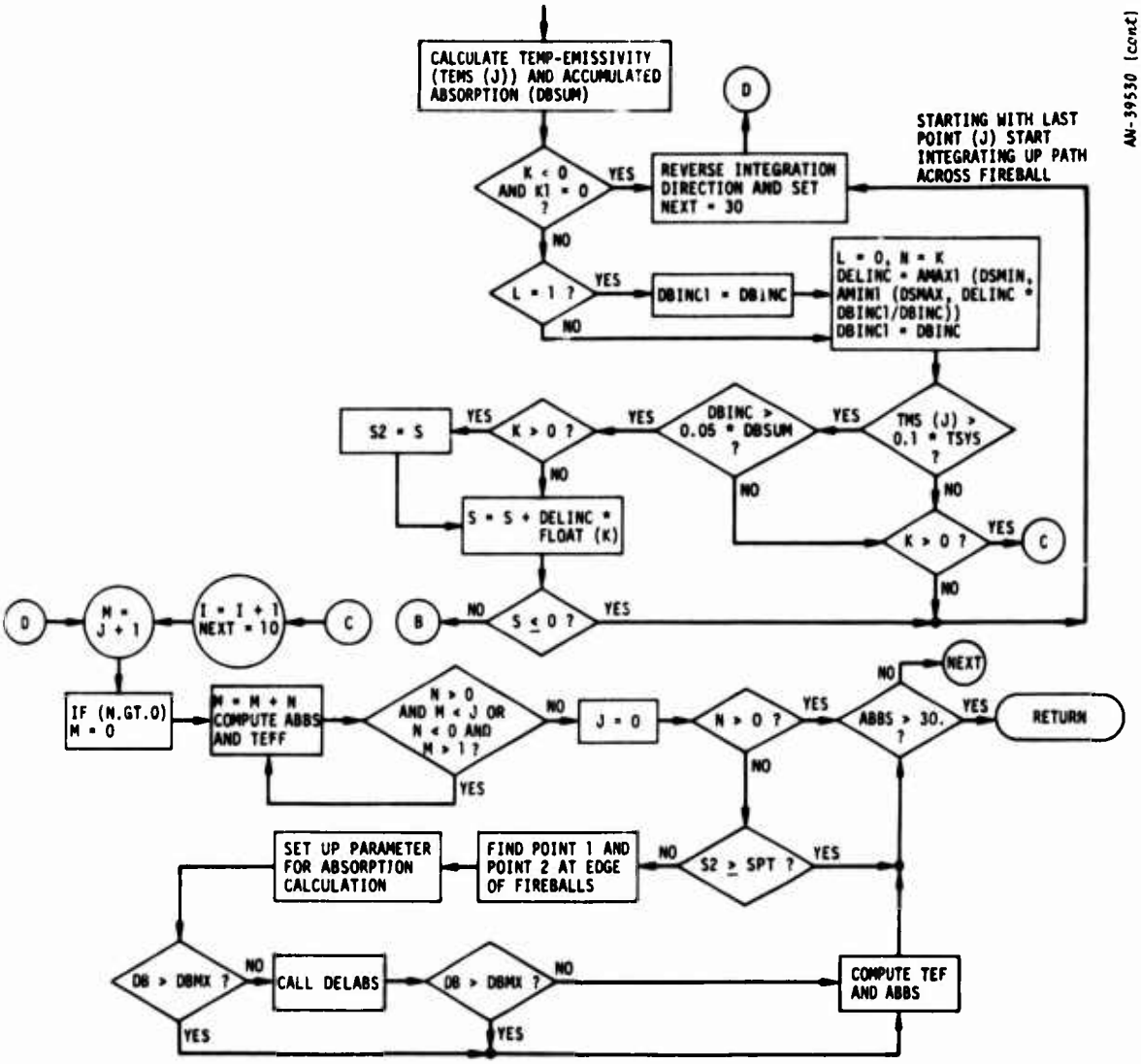


Figure 5.5 (Cont.)

$S2_i$ = range to the upper edge of the (i-1)th fireball already integrated over (see Fig. 5.4).

K = flag indicating the integration direction (denoted DIRECTN in code) = +1, if integrating up the path; -1, if integrating down.

K1 = flag indicating that the lower integration limit (S2) has been reached (denoted ATSMIN in code) = 0, if at S2; 1, otherwise.

L = flag indicating that the integration step is currently at a mandatory point (denoted ATPOINT in code) = 1, if at mandatory point; 0, otherwise.

J = number of integration steps when integrating over fireball.

M = counter to loop over J integration steps.

5.2.4 Inputs and Outputs

The NOISE routine is called by subroutine RAD1S of the radar signal processing module. The inputs and outputs of the NOISE routine are defined as follows:

Inputs:	TSYS	System temperature, °K
	TIME	Current time
	R(3)	Radar position, cm
	FREQ	Radar frequency, MHz
Output:	TEF	Effective noise temperature, °K

5.3 REFRACTION

Roscoe makes the basic assumption that the number of possible propagation paths to the target in the mainbeam is either one (corresponding to the smoothly varying electron density gradient region), or much greater than one (when striations are present). These two cases are discussed separately.

5.3.1 Gross Refraction

In this case, a ray trace is performed, starting at an altitude of 60 km along the radar beam-pointing direction and ending at a point that may be displaced from the target but is at the same range from the radar. The incremental range and angular bias errors at each step are given by:

$$\frac{\partial R}{\partial S} = \frac{1.59 \times 10^9 N_e}{\left(2\pi \times 10^6\right)^2 \left[f^2 + (\nu_{em} + \nu_{ei})^2\right] \mu}$$

$$\frac{\partial \theta_i}{\partial S} = \frac{-1.59 \times 10^9 \nabla N_{ei}}{\left(2\pi \times 10^6\right)^2 \left[f^2 + (\nu_{em} + \nu_{ei})^2\right] \mu}$$

where

μ = real part of the index of refraction

$$= 1 - \left[\frac{3.18 \times 10^9 N_e}{\left(2\pi \times 10^6\right)^2 \left[f^2 + (\nu_{em} + \nu_{ei})^2 \right]} \right]^{1/2}$$

N_e = electron density, cm^{-3}

∇N_{ei} = electron density gradient in the ith direction, cm^{-3}/cm

f = radar frequency, MHz

ν_{em}, ν_{ei} = average electron-neutral and electron-ion collision frequencies divided by $2\pi \times 10^6$

It is assumed that the total angular error obtained from this procedure is not large, so that if the beam is rotated just this amount, the refracted beam will hit the target.*

5.3.2 Refraction due to Striations

For refractive effects due to ionized striations (the latter case mentioned above), the assumption is made that the electron density does not lead to excessive departures of a ray in the beam from its nominal smooth trajectory. With this assumption, it is possible to develop a theory which describes, in statistical terms, the effects of the electron density fluctuations.

*A refinement of this procedure would be to iterate on the beam pointing direction until the refracted beam does hit the target.

The theory presented here, termed the "thin screen approximation," differs from previous models[5,6] in that it systematically develops the "dogleg" range error and phase terms. Because of these differences, a rather complete description of the model is given in Appendix F.

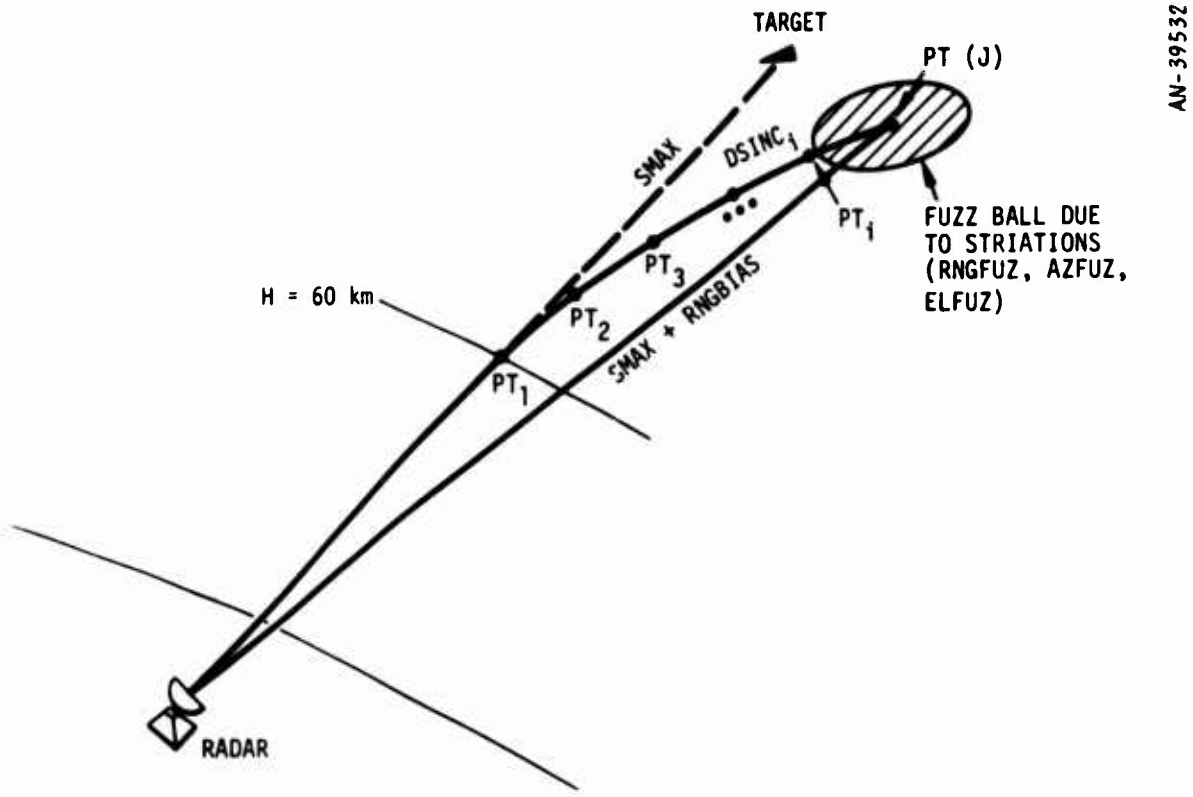
5.3.3 Procedure

Both computations described above are calculated within subroutine REFRCT in ROSCOE. The procedure follows these steps (see Fig. 5.6):

1. The radar beam is assumed to be straight to an altitude of 60 km.
2. At this point, an incremental range step size (DSINC) is selected and the deflection of the ray over this range segment is computed.
3. In addition, random errors due to striations over this range segment are computed and projected to the target range (SMAX).
4. The procedure continues in this way until the range from the radar to the point on the refracted ray (PT_1) is equal to the target range.
5. When this condition is satisfied, the program outputs bias and random errors in range, azimuth, and elevation at the final point, the integrated electron density over the path, and the angle between the line-of-sight and the magnetic field at the point of maximum electron density along the path.

5.3.4 Flow Chart

A flow chart of subroutine REFRCT is presented in Fig. 5.7. This routine requires information from several phenomenology routines and calls other propagation models as shown. A description of some of these subroutines follows.



SMAX = SLANT RANGE TO TARGET

RNGBIAS = TOTAL RANGE BIAS ERROR DUE TO SMOOTHLY VARYING REFRACTION

PT(J) = COORDINATE OF END POINT OF REFRACTED RAY PATH AT RANGE OF TARGET

RNGFUZ, ELFUZ, AZFUZ = RMS UNCERTAINTY IN END POINT

$DSINC_1$ = INCREMENTAL RANGE STEP

= AMAX1 (0.5 * DSNOM, AMIN1 (2. * DSNOM, 0.05/DEFL * DSNOM))

DSNOM = AMIN1 (50., RANGE CORRESPONDING TO 10 km ALTITUDE STEP)

DEFL = AMAX1 (INCREMENTAL DEFLECTION IN AZIMUTH DIRECTION, INCREMENTAL DEFLECTION IN ELEVATION DIRECTION)

Figure 5.6. Ray Trace Procedure

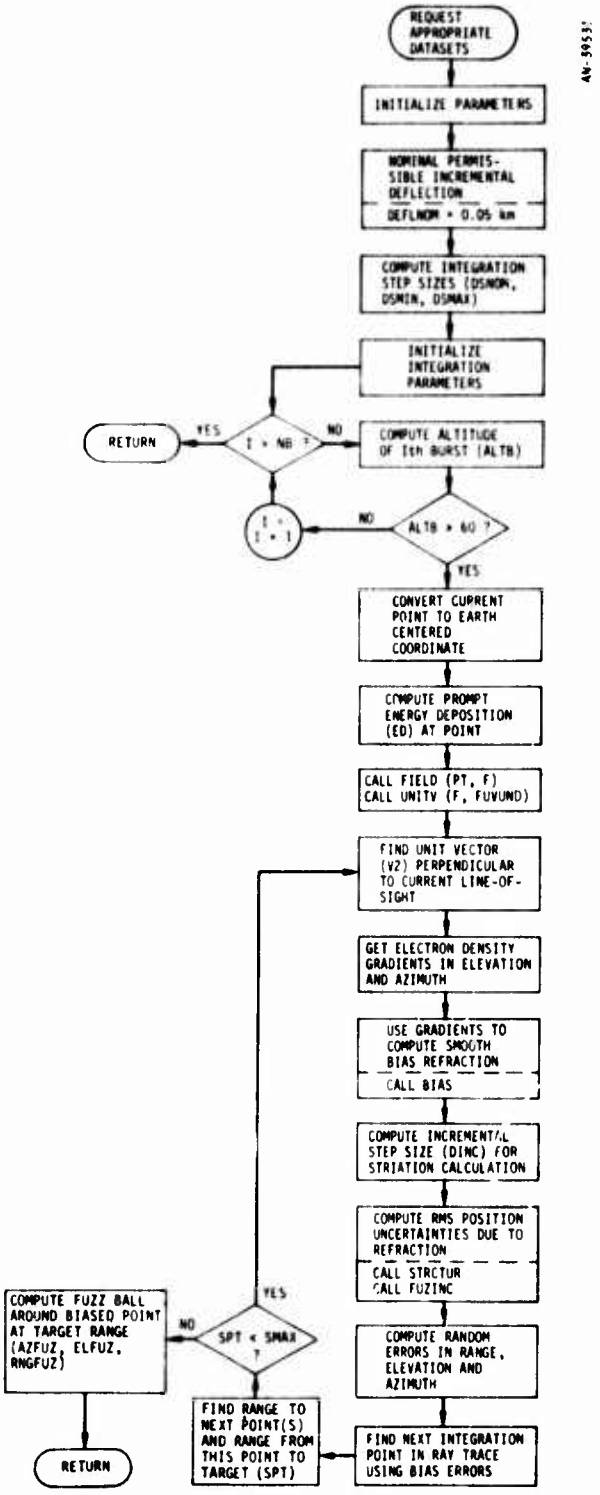


Figure 5.7. Subroutine REFRCT

Phenomenology Subroutines

ELDENS	Provides electron density and temperature at a point due to all sources
GRADNE	Computes electron density gradient at point $\overline{P}$ in a specified direction $\overline{V}$
STRCTUR	Computes structure of a striated region
BFIELD	Returns direction of B-field at a point

Other Propagation-Related Routines

COLLF	Calculates electron-neutral and electron-ion collision frequencies
BIAS	Computes smooth bias refractive errors
FUZINC	Computes RMS position uncertainties due to refraction in striated medium over the path length

5.3.5 Inputs and Outputs

Subroutine REFRCT is called by subroutine REF1S of the radar signal processing module. The inputs and outputs of REFRCT are:

Inputs:	TIME	Engagement time, s
	SMAX	Slant range to target, cm
	R(3)	Radar position, cm
	FREQ	Radar frequency, MHz
Outputs:	$\overline{NP}$	Number of possible paths to target
	XNE	Integrated electron density
	REFRAC(1-3)	Bias errors in R,A,E coordinates
	REFRAC(4-6)	Random errors in R,A,E coordinates
	REFRAC(7)	Angle between LOS and B-field at point of maximum electron density

SIGBAR	Amplitude gain factor for each return
XMU	Angle between major axis of uncertainty ellipse and the u-axis in sine space

The refraction errors are used to compute the apparent target position that the radar "sees." The integrated electron density is used in the estimation of the pulse dispersion, and the angle between the LOS and B-field at the point of maximum electron density is used to determine the Faraday rotation angle, if appropriate.*

5.4 CLUTTER

Radar backscatter or clutter can occur from a variety of regions following a nuclear burst; in general, whenever the beam traverses a region in which the dielectric constant varies. Clutter sources such as dust, fireballs, and field-aligned regions are treated in ROSCOE.

5.4.1 Assumptions

The following assumptions have been made for the clutter model:

1. The absorption of the clutter return from the region surface** is computed along the path between the region's center and the radar. This allows a single path absorption integration to be used for each clutter source.

2. The solid angle subtended by the clutter source is divided into 25 projected area segments for incremental clutter calculations. This makes the clutter integration independent of the geometry of the region.

---

*Faraday rotation is considered when a linearly polarized radar antenna is specified in the input.

**This absorption calculation includes the absorption through the gammas surrounding the fireball, if appropriate.

3. The total clutter return integrated over the beam pattern is multiplied by a factor which represents the fraction of the clutter source in the compressed and uncompressed range cells; that is, a detailed integration is not performed over the range cell.

5.4.2 Procedure

The total clutter power reaching the radar from all clutter sources is computed in subroutine FBCLTR. This routine determines if a clutter source is in the compressed or uncompressed range cell and, if it is, computes the fraction of the cell covered. The clutter power is then integrated over the antenna beam pattern in subroutine CLINT and returned to FBCLTR. Finally, the one-way absorption is computed from the source to the radar and the clutter power is degraded by this factor times the fraction of the range cell subtended by the source.

The clutter integration in CLINT follows these steps:

1. Define the integration step size (incremental area) to be small compared to the extent of the source. (A maximum of 25 steps is allowed.)
2. Direct a vector from the radar to the center of each of these area segments.
3. Compute the clutter power per unit area along the defined vector intersecting the source. Degrade the clutter power by the antenna angular gain.
4. Sweep in azimuth and elevation, integrating over the portions of the source which contribute significantly to the total clutter.

An individual calculation is skipped when (1) the vector does not intersect the geometric region defining the source, or (2) the individual segment contribution is a small fraction of the cumulative clutter power

(this is necessary for fireballs with diffuse edges). The integration is terminated when all 25 steps have been considered.

The incremental clutter is computed in subroutine CLUTINC, which is called by CLINT. It in turn calls routines to get the surface reflection coefficient and applies a doppler loss to the incremental clutter power which is computed by comparing the doppler shift of the clutter source with the doppler shift of the target. The doppler loss is computed from the following equation:

$$FDOP = ABS[(\sin X)/X]^{3.4}$$

where

$$X = \frac{\pi}{B_n} ABS(FD_T - FD_C)$$

B_n = noise bandwidth, Hz

FD_T = expected doppler shift for the target based on past tracking measurements, Hz

FD_C = doppler shift of the clutter source

$$= \frac{2f_o \dot{R}_c}{c}, \text{ Hz}$$

f_o = radar frequency, Hz

c = speed of light, cm/s

$\dot{R}_c$ = velocity of the clutter source toward the radar, cm/s

Thus, the equation for the incremental clutter power per unit area ($\Delta C/\Delta A$) is

$$\frac{\Delta C}{\Delta A} = S \times FDOP \times ALBEDO$$

where S = signal for this computation

ALBEDO = surface reflection coefficient

The total clutter power for this source is then given by:

$$C = \Delta X \times 10^{-\frac{DBSUM}{10}} \int_A \left[\frac{\Delta C}{\Delta A} \times GAIN^2 \right] dA$$

where GAIN = antenna gain in the direction corresponding to the ith area segment

ΔX = fraction of range cell subtended by the source

DBSUM = total absorption from the source to the radar

5.4.3 Flow Charts

Figures 5.8 and 5.9 are flow charts for subroutines FBCLTR and CLINT. Some of the other subroutines and functions called by these routines are listed below.

ABSORB	Computes the absorption from the fireball surface to the radar (see Sec. 5.1)
CLUTINC	Computes the incremental clutter power per unit area for the ith source
EXTENT	Computes the range to the surface of the clutter source and the extent of the source
GOA	Computes the angular gain for an input vector direction off-beam center (see Sec. 4.4.2)
SLDANGL	Computes the solid angle subtended by the clutter source

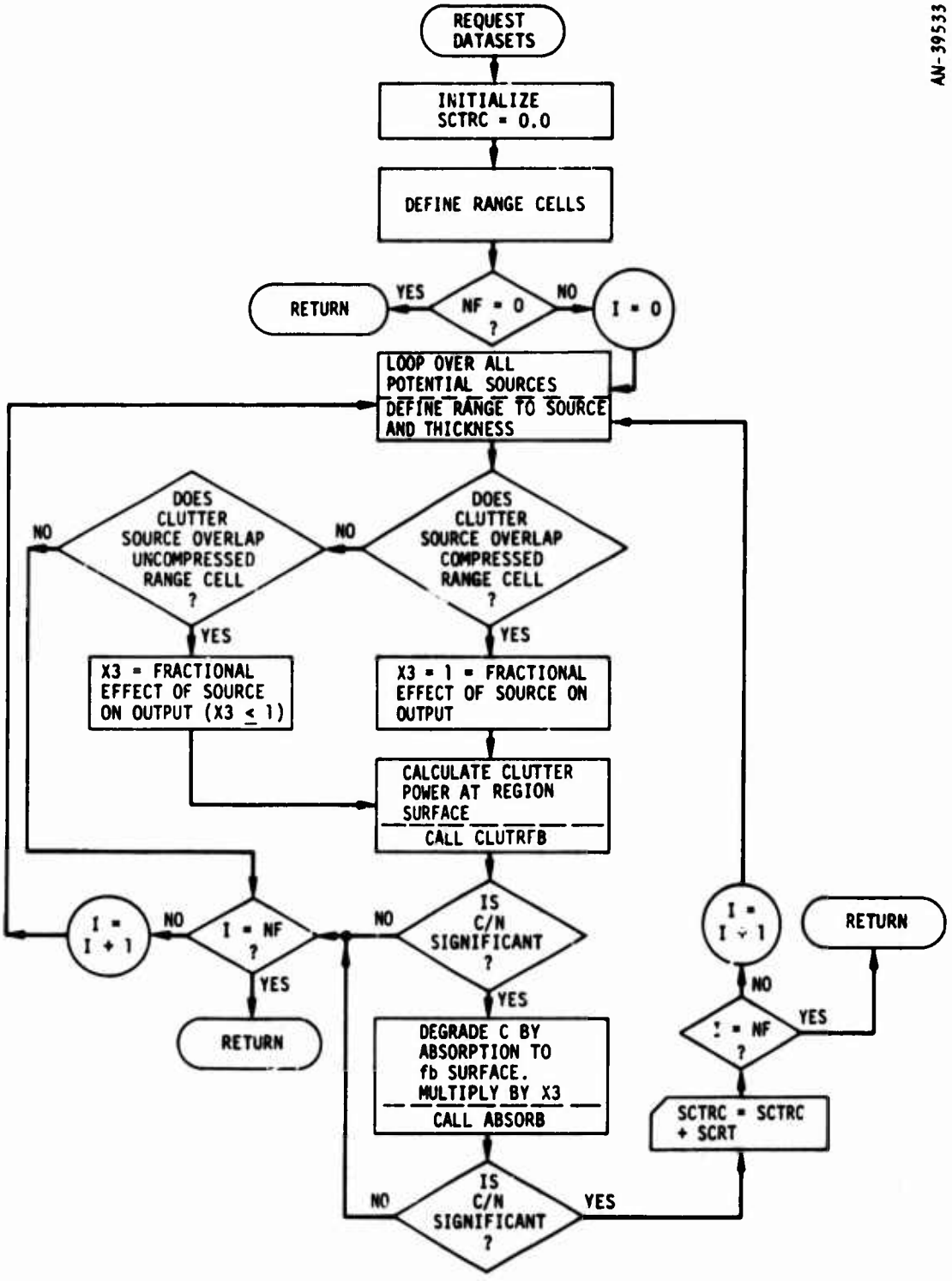


Figure 5.8. Subroutine FBCLTR

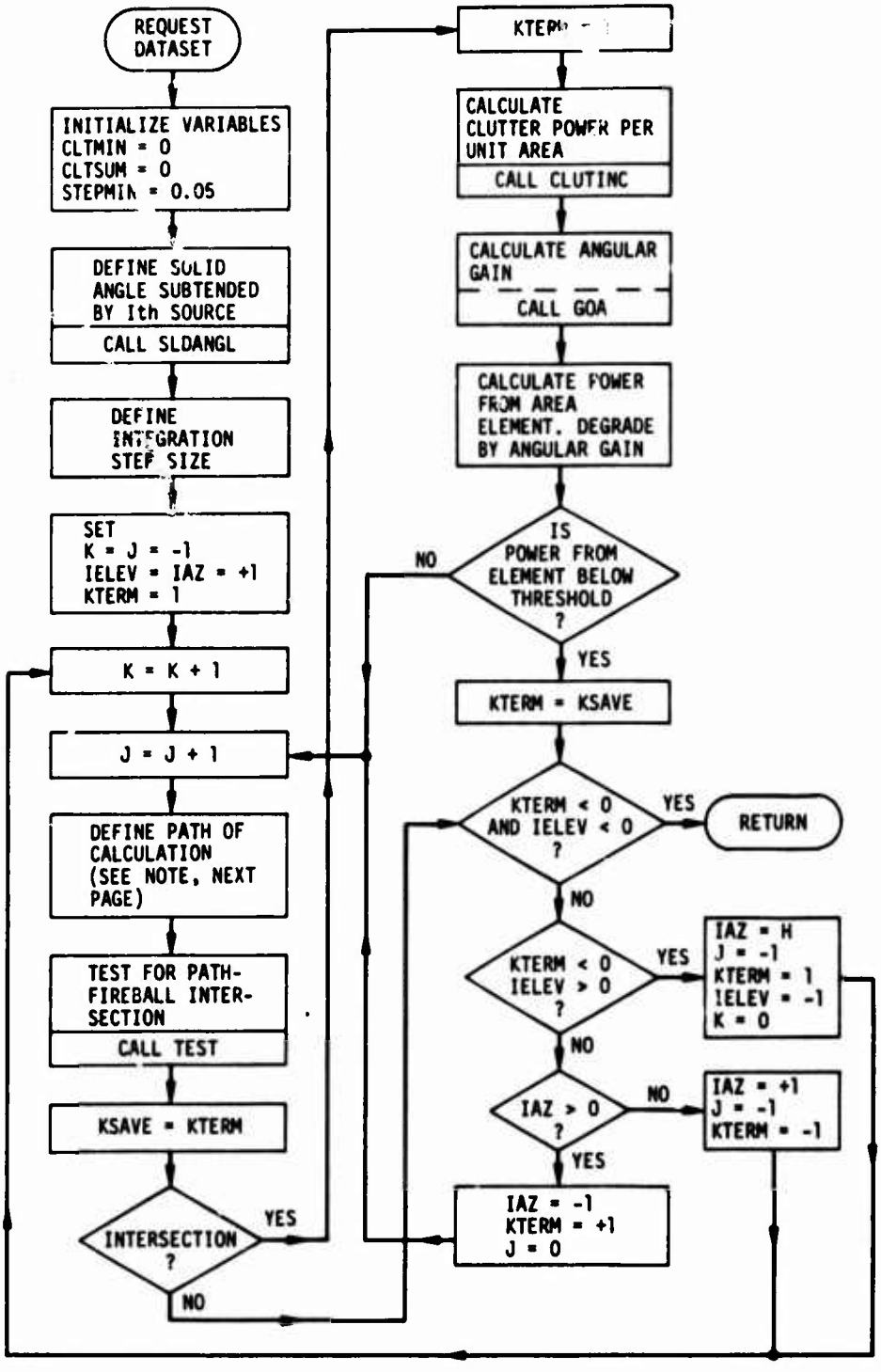


Figure 5.9. Subroutine CLINT

5.4.4 Inputs and Outputs

The inputs and outputs to the three main routines which make up the clutter module (FBCLTR, CLINT, CLUTINC) are listed in Table 5.1.

5.5 LOW-ALTITUDE MULTIPATH

The low-altitude multipath model consists of four subroutines. The main routine (MLTPATH) is called by subroutine REFLST1 (see Sec. 4.4.1). It determines if there are any additional target images caused by low-altitude fireball multipath which might interfere with the primary target signal. The other routines perform the following supporting computations:

BOUNCE--Computes the apparent target image position and signal amplitude for the three additional returns that can occur for each signal

AMPREF--Computes the amplitude reflection coefficient at the designated specular reflection point on the fireball surface

SPCULAR--Computes the specular reflection point on the fireball surface

5.5.1 Assumptions

The following assumptions have been made in the MLTPATH model:

1. Only fireballs below 60 km altitude are considered as possible reflecting surfaces.
2. Multiple reflections are ignored; that is, only a single reflection between any object/fireball pair is considered (but all pairs are considered).
3. Differences in attenuation along the reflected path as compared to the direct path are ignored in computing the received signal.

In addition, some simplifying assumptions are made in AMPREF and SPCULAR:

TABLE 5.1

INPUT/OUTPUT DATA FOR CLUTTER ROUTINES

FBCLTR Inputs (from calling sequence):	RTARG	Range to target, cm
	XT	Total noise power, erg/s
	T	Time of calculation, s
	FREQ	Radar frequency, MHz
	R	Earth-centered coordinates of radar, cm
	BVEC	Boresight vector of radar
FBCLTR Inputs (from datasets):	DELRU	Uncompressed pulse range resolution, cm
	DELRC	Compressed pulse range resolution, cm
	FDBAR	Expected doppler shift, MHz
	SLRED	Range sidelobe reduction factor
	SNMIN	Signal-to-noise threshold
	FBLIS	List of clutter sources
FBCLTR Output:	SCTRC	Total clutter power reaching receiver, erg/s
CLINT Inputs:	DSPFB	Pointer to clutter source of interest
	FPC	Pulse compression ratio
	BN	Noise bandwidth
	R	Radar coordinates, cm
	BVEC	Boresight vector
CLINT Output:	CLTR	Clutter power output (at surface of region), erg/s
CLUTINC Inputs:	DSPFB	Pointer to clutter source of interest
	V	Vector from radar to source point of present calculation
	R	Earth-centered coordinates of radar, cm
	FPC	Pulse compression ratio
	BN	Noise bandwidth
	FREQ	Radar frequency, MHz
CLUTINC Output:	CLTINC	Incremental clutter from ith source, erg/s

1. The amplitude reflection coefficient is computed, using the REFCØ routine, by inputting the proper angle of incidence of the radar beam.

2. The fireball is spherical and the reflection point lies on the surface at the point of closest approach to the direct line-of-sight (see Fig. 5.10).

5.5.2 Discussion

Flow charts are presented for subroutines MLTPATH and BOUNCE in Figs. 5.11 and 5.12. The MLTPATH routine accesses the radar signal processing dataset to get the list of primary targets that have been acquired. It also accesses the propagation inputs dataset to get the mandatory points list, which are the points of closest approach of the fireballs to the current line-of-sight. The routine then loops on each object/point pair for those mandatory points where the associated fireball is below 60 km and calls subroutine BOUNCE.

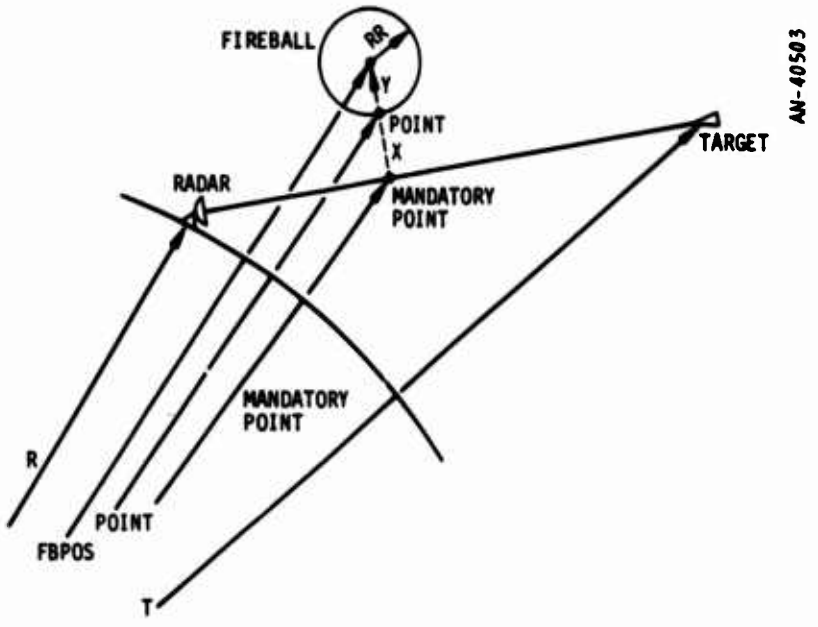


Figure 5.10. Specular Reflection Point

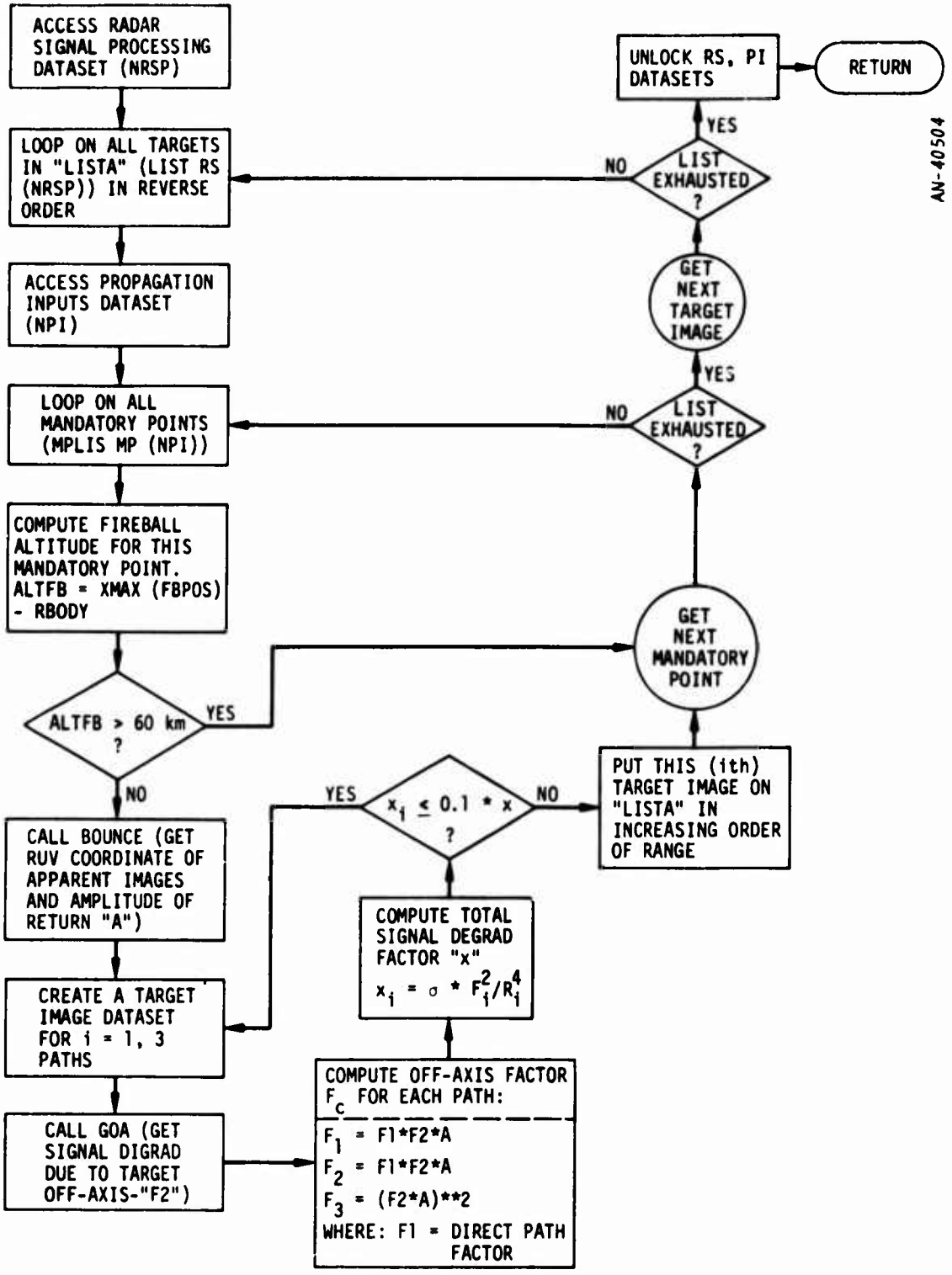


Figure 5.11. Subroutine MLTPATH

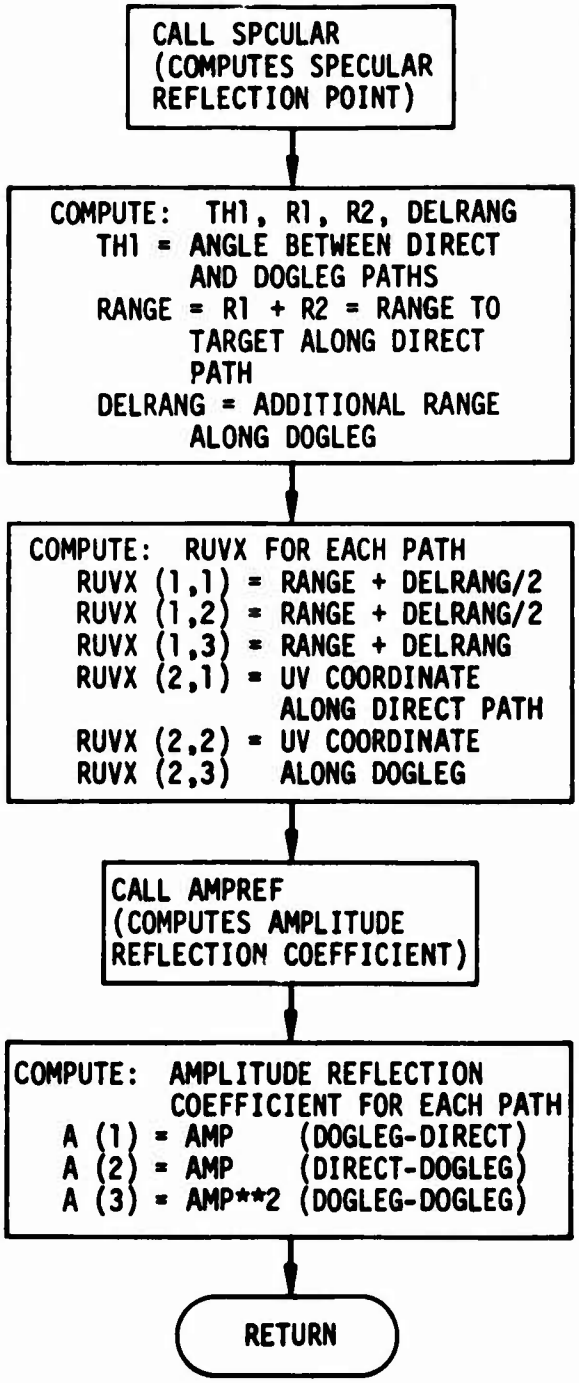


Figure 5.12. Subroutine BOUNCE

BOUNCE constructs the geometry shown in Fig. 5.13 for the object/fireball pair of interest. It first calls subroutine SPCULAR to find the specular reflection point (POINT1 in the figure) and then computes the angle, θ_1 , for the dogleg path and the incremental increase in range, DELRANG, along this path. There are three possible returns which the radar may receive in addition to the direct or primary return. They are:

1. Transmit along the dogleg path, receive along the direct path
2. Transmit along the direct path, receive along the dogleg path
3. Transmit along the dogleg, receive along the dogleg

In the first case, the radar will "see" a target image at a range equal to the actual range plus DELRANG in the same direction as the real target. In the second and third cases, it will "see" the image in the dogleg

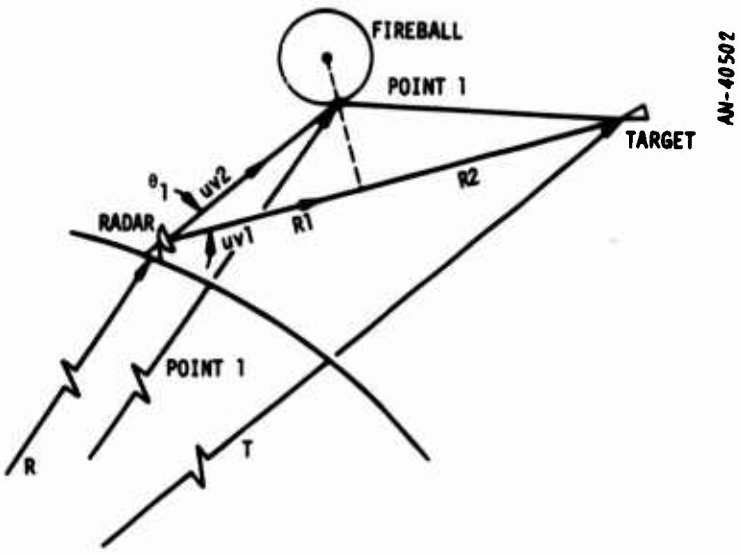


Figure 5.13. Bounce Path Geometry

direction at ranges equal to the range plus DELRANG and the range plus twice DELRANG, respectively. This information is returned to MLTPATH along with the amplitude of each signal relative to the direct-path signal.

In MLTPATH, this information is used to calculate the reduction in signal power. The signal power for these images is reduced because of reflection (signal amplitude) and because the target image is off the main beam axis in either transmit or receive for the first two cases and in both directions for the third case. The resulting signal degradation factor X_i (see Fig. 5.11) for each case is compared to a similar expression for the actual target (direct path). If the ratio of these numbers is greater than -10 dB, the target image dataset associated with this reflected path is added to the target list and may ultimately affect the radar measurement for this look.

5.5.3 Inputs and Outputs

The MLTPATH subroutine is called in REFLST1 with the radar position and boresight vector in the calling sequence. In addition, the radar signal processing and propagation inputs datasets are accessed, as mentioned above. The output of MLTPATH is the creation of new target image datasets which are placed on the target image list LISTA.

Inputs and outputs for the other subroutines are carried through the calling sequences. Definitions for these variables follow:

BOUNCE Inputs:	R(3)	Radar position, earth centered
	UVB(3)	Boresight vector
	T(3)	Assumed target position, earth centered
	DSPMP	Pointer to mandatory point dataset of interest
BOUNCE Outputs:	A(3)	Relative amplitude of return signal for ith path

	RUVX(3,3)	RUV coordinates of apparent target image for ith path
AMPREF Inputs:	DSPMP	Pointer to mandatory point dataset of interest
	R(3)	Radar position, earth centered
	POINT1(3)	Earth-centered coordinates of specular reflection point of fireball surface
AMPREF Output:	AMP	Amplitude reflection coefficient (relative to 1.0 for total reflection)
SPCULAR Inputs:	R(3)	Radar position, earth centered
	T(3)	Assumed target position, earth centered
	DSPMP	Pointer to mandatory point dataset of interest
SPCULAR Output:	POINT(3)	Earth-centered coordinates of specular reflection point of fireball surface

REFERENCES

1. L. R. Ford, Jr. and T. O. Sullivan, An Overview of BAG XIV, A Simulation for Hardsite Defense, General Research Corporation IMR-1484, March 1971.

2. T. Plambeck, The Compleat TRAIDsman, General Research Corporation IM-711/2, September 1969.

3. D. F. Palmer, et al., SETS 1 Software Design Specifications, General Research Corporation CR-4-245, July 1972.

4. R. Manasse, Summary of Maximum Theoretical Accuracy of Radar Measurements, Mitre Technical Series, 2(SR-11), 1 April 1960.

5. J. Ise, R. Rein and D. Sachs, Refraction in a Nuclear Environment, General Research Corporation CR-1-126, DASA 2552, July 1970.

6. R. J. Jordano, RANC Update Preliminary Report: Fireball Striation/Scintillation Models, General Electric Company GE73TMP-30, January 1973.

APPENDIX A

COORDINATE SYSTEMS IN ROSCOE

Three basic coordinate systems are used in the radar modeling for ROSCOE, but for two of these the components are sometimes quoted in rectangular form and sometimes in spherical form, so in essence there are five sets of coordinates:

1. Radar Face Coordinates

   Rectangular: ρ_1,ρ_2,ρ_3

   Spherical (sine-space): R,u,v

2. Radar Site Coordinates

   Rectangular (topocentric): r_1,r_2,r_3

   Spherical (alt-azimuth): R,A,E

3. Geocentric Rectangular Cartesian Coordinates: R_1,R_2,R_3

three components, in each coordinate system, represent the line-of-sight vector, $\vec{R}$, from the radar to the (apparent) target.

The following vector notation is used in this appendix. If $\vec{V}$ represents any vector, then

- V represents the magnitude of the vector $(|\vec{V}|)$
- $\hat{V}$ represents the equivalent unit vector $(\vec{V}/V)$
- V_i represents the ith component of the vector in a given coordinate system (i = 1,2,3)

A.1 RADAR FACE COORDINATES

A.1.1 Rectangular

This is a right-handed orthogonal rectangular Cartesian coordinate system associated with a radar face. Two of its basis vectors lie in the

plane of the radar face, and the third is normal to the face in the boresight (i.e., outward) direction. As long as the radar face is not locally horizontal, the basis vectors, $\hat{\xi}$, $\hat{\eta}$, $\hat{\zeta}$, are defined as follows:

$\hat{\zeta}$ points in the boresight direction

$\hat{\xi}$ is locally horizontal and in the direction determined by $\hat{n} \times \hat{\zeta}$, where $\hat{n}$ is the local unit earth normal at the radar

$\hat{\eta}$ is defined by $\hat{\zeta} \times \hat{\xi}$

If the face is horizontal, this definition becomes meaningless. In this case we make a convenient arbitrary definition:

$\hat{\xi}$ is locally horizontal and points north

$\hat{\eta}$ is locally horizontal and points west

$\hat{\zeta} \equiv \hat{n}$

In this system, the components of the radar pointing vector $\vec{R}$ are

$$\rho_1 = \vec{R} \cdot \hat{\xi}$$
$$\rho_2 = \vec{R} \cdot \hat{\eta}$$
$$\rho_3 = \vec{R} \cdot \hat{\zeta}$$

A.1.2 Spherical

This subsystem is often called "sine-space." The components of the radar pointing vector $\vec{R}$ are

$$R = |\vec{R}|$$
$$u = \rho_1/R \equiv \hat{R} \cdot \hat{\xi}$$
$$v = \rho_2/R \equiv \hat{R} \cdot \hat{\eta}$$

Another quantity of interest in this system is θ, the off-boresight angle, which is given by

$$\theta = \cos^{-1}(\rho_3/R) \equiv \sin^{-1}\sqrt{u^2 + v^2}$$

A.2 RADAR SITE COORDINATES

A.2.1 Rectangular

This is a right-handed orthogonal rectangular Cartesian coordinate system associated with a radar site. Because a site on the surface of the earth is called a "topos," this is sometimes called the topocentric system for short (of course this does not imply that the system is restricted to radars located on the earth's surface). Two of its basis vectors lie in the local horizontal plane, and the third points toward the zenith. These basis vectors, $\hat{x},\hat{y},\hat{z}$, are defined as follows:

$\hat{x}$ is locally horizontal and points east

$\hat{y}$ is locally horizontal and points north

$\hat{z}$ is locally vertical and points up ($\hat{z} \equiv \hat{n}$)

In this system, the components of the radar pointing vector $\hat{R}$ are

$$r_1 = \vec{R} \cdot \hat{x}$$
$$r_2 = \vec{R} \cdot \hat{y}$$
$$r_3 = \vec{R} \cdot \hat{z}$$

A.2.2 Spherical

This subsystem is often called angular-space or the alt-azimuth system.

The components of the radar pointing vector $\vec{R}$ are

$$R = |\vec{R}| \equiv (r_1^2 + r_2^2 + r_3^2)^{1/2}$$

$$A = \tan^{-1}(r_1/r_2)$$

$$E = \sin^{-1}(r_3/R)$$

A.3 GEOCENTRIC RECTANGULAR CARTESIAN COORDINATES

This is a right-handed orthogonal rectangular coordinate system with origin at the center of a spherical earth; the coordinate system rotates with the earth. It has sometimes been referred to simply as "the Cartesian system," but that term is clearly ambiguous; here it is called the G.R.C. system. Its basic vectors, $\hat{X},\hat{Y},\hat{Z}$, are defined as follows:

$\hat{X}$ lies in the equatorial plane and points toward the Greenwich meridian

$\hat{Z}$ lies along the earth's rotational axis and points toward the geographic North Pole

$\hat{Y} = \hat{Z} \times \hat{X}$

In this system, the components of the radar pointing vector $\vec{R}$ are

$$R_1 = \vec{R} \cdot \hat{X}$$

$$R_2 = \vec{R} \cdot \hat{Y}$$

$$R_3 = \vec{R} \cdot \hat{Z}$$

However, these are <u>not</u> the components that are usually desired or meant when one refers to "the GRC coordinates" of a point. What are really desired here are the components of the radius vector from the center of the earth to the (apparent) target point. Let $\vec{W}$ represent this vector. Then

$$\vec{W} = \vec{T} + \vec{K}$$

where $\vec{T}$ is the "topos vector": the radius vector from the center of the

earth to the radar site location. Thus, finally, the desired representation of the target point in the G.R.C. system consists of the components

$$W_1 = \vec{W} \cdot \hat{X} \equiv T_1 + R_1$$
$$W_2 = \vec{W} \cdot \hat{Y} \equiv T_2 + R_2$$
$$W_3 = \vec{W} \cdot \hat{Z} \equiv T_3 + R_3$$

where T_1, T_2, T_3 are the G.R.C. components of $\vec{T}$.

APPENDIX B

DISPERSION AND THE RADAR AMBIGUITY FUNCTION

B.1 UNCHIRPED RECTANGULAR PULSES

In Sec. 4.3.2, a sketch was given of the derivation of the dispersion parameter, A_D, for a simple rectangular narrowband pulse of bandwidth B, and with the implicit assumption that the radar frequency f is always much greater than the plasma frequency everywhere along the propagation path. This derivation, and the results to be discussed below, are based on some previously-performed but as yet unpublished work by one of the authors.*

It is well known that when a radar signal passes through a dispersive medium and then returns to be passed through a receiver filter matched to the originally-transmitted waveform, two major effects are observed:

1. The peak signal-to-noise ratio at the output of the filter is reduced by a factor L_D, the dispersive mismatch loss.
2. Both the received signal and the ambiguity function which is the envelope of the filter output are distorted with respect to the undispersed case, the distortion causing a stretching of the function in the range direction with a consequent loss of resolution.

Both of these effects are modeled in ROSCOE.

For an unchirped rectangular pulse in the absence of dispersion, the matched-filter output is triangular in the range dimension. When dispersion is present, the shape of this cut through the ambiguity function changes, becoming longer, lower, and more rounded. An expression for this output function has been derived. For $A_D << 1$, it remains very similar to the undistorted triangular function, only having its peak amplitude

*H. S. Ostrowsky, Dispersive Effects on Radar Signal Processing (unpublished).

reduced and its width correspondingly expanded so as to satisfy energy conservation. As A_D increases, the output function becomes approximately Gaussian:

$$\chi(R) \approx \frac{1}{\sqrt{L_D}} \exp\left[-\frac{4b^2(R - R_o)^2}{L_D^2 \delta R_o^2}\right]$$

where R_o is the range at the peak of the return and δR_o is the nominal range resolution for the pulse.

By requiring that energy be conserved despite the presence of dispersion, and equating the areas under the distorted and undistorted output functions, the parameter b^2 is found to be given approximately by $b^2 \approx 0.884$.

If the dispersion-distorted range resolution width, $\delta R_o'$, is defined by finding where $|\chi(R)/\chi(R_o)| = 1/2$, it turns out that $\delta R_o'$ is well approximated by

$$\delta R_o' \approx L_D \delta R_o$$

There only remains to find the dispersive loss factor, L_D , in terms of which the ambiguity function (in range) and the range-resolution width are expressed. This quantity is given as a function of A_D by a complicated expression involving Fresnel integrals and trigonometric functions. For reasonably small values of A_D this function was well approximated by a piecewise polynomial fit. For larger values of A_D the function can be expanded into a simple "asymptotic" form. These expressions are used in subroutine DISPERS to compute $L_D(A_D)$.

B.2 CHIRPED PULSES

Many different kinds of chirped signals are used by radar designers. Simple chirped rectangular pulses have such high range sidelobes that they are hardly ever used in practical radar systems. Instead, some form of weighting is employed to reduce these sidelobe levels (at the expense of slightly broadening the peak of the output ambiguity function in range). Rather than complicate the radar model by inserting a selection of possible weighted waveforms, it was decided to make a single reasonable choice to represent the "typical" chirped pulse, defined by its ambiguity function in the range direction.

It was found that a waveform lying between the Hamming and the Hanning weightings (two popular types) could be approximated by a mainlobe of the form

$$\chi(X) = \left[\frac{\sin(\pi X/\delta R_1)}{\pi X/\delta R_1}\right]^P$$

out to its first null (at $|X| = \delta R_1$), and

$$\chi(X) = \text{constant}$$

for $|X| > \delta R_1$. Here,

$$X \equiv R - R_o$$

$$\delta R_1 \equiv 2\delta R_o/F_{pc}$$

The constant range sidelobe level is input by the user, and the best approximation to the Hamming and Hanning waveforms was achieved when $P \approx 1.7$.

This is the ambiguity function (in the range dimension) which is used in the radar model to represent a chirped signal wherever required,

for example in computing the results of dispersion, interfering targets, split-gate range tracking, etc. When dispersion is present, δR_1 is replaced by $\delta R_1' = L_D \delta R_1$ in analogy with the results for the unchirped rectangular pulse. The spreading of the range mainlobe is thus taken into account, and the assumption is made that the dispersive loss continues to be given by $L_D(A_D)$. However, here the parameter A_D is increased by the factor F_{pc} (the pulse compression ratio) compared to its value for the unchirped pulse, to take into account the fact that the effective bandwidth is now $B \simeq F_{pc} B_n$. Since the ambiguity function is normalized to unity at $X = 0$, the sidelobe level is increased by a factor of $\sqrt{L_D}$ to compensate for the actual dispersive loss at the peak; the effective sidelobe level is thus increased by dispersion.

This approach to computing dispersive effects for chirped waveforms is largely heuristic, since the correct expressions have not yet been evaluated in this case. The results appear to be physically reasonable.

B.3 THE DOPPLER DIMENSION

It should be noted that the ambiguity functions discussed above are all in the range dimension only. The cross-range, or doppler, coordinate is taken into account by another heuristic procedure which seems to be reasonable for an unchirped rectangular waveform and which, for chirped waveforms, is based on a substitution which gives correct results in the known case of a simple, frequency-modulated rectangular pulse.

For unchirped pulses, there is of course no range-doppler coupling to produce apparent range errors, but there will be a doppler mismatch loss whenever the doppler frequency (f_d) of the target differs from the frequency ($\bar{f}_d$) for which the doppler filter is set. This mismatch loss is given by

$$L_{Dop} \simeq \left| \frac{\sin X}{X} \right|^{-2}$$

where

$$X \equiv \frac{2\pi}{c} \delta R_o |\bar{f}_d - f_d|$$

Notice that dispersive broadening of the resolution length does not affect the doppler mismatch loss.

For chirped pulses, the doppler mismatch loss is given by a similar expression except that the exponent -2 is replaced by $-2p$, where $p \approx 1.7$ for the approximate "universal chirped waveform" defined above. Actually, however, it is not necessary to compute the mismatch loss explicitly for the chirped waveform, since this is automatically taken care of when the apparent range to the target, R'_a, is calculated with the inclusion of the effect of range-doppler coupling. In the absence of dispersion this is given simply by

$$R'_a = R' - \frac{\delta R_o \delta R_1}{c} (\bar{f}_d - f_d)$$

When dispersion is present ($L_D > 1$), this expression becomes

$$R'_a = R' - L_D \frac{\delta R_o \delta R_1}{c} (\bar{f}_d - f_d)$$

which is justified by a heuristic argument. Notice that by using $L_D = 1$ for the non-dispersive situation, both cases can be represented by the same equation, which may be put into the simpler but equivalent form

$$R'_a = R' - \frac{\delta R_o}{B} (\bar{f}_d - f_d)$$

APPENDIX C

MONOPULSE

The radar monopulse (angle-tracking) system is modeled in ROSCOE by three subroutines, each one pertaining to a different situation.

Since the monopulse subroutines operate prior to the introduction of the radar measurement errors in the code logic, the monopulse output for a single point target lying within the antenna-pattern mainlobe is the target's correct angular position.* This is the output produced by subroutine MNOPLS1 (Sec. 4.6.1). However, a monopulse system has the property that a single point target lying outside the mainlobe is incorrectly assigned to some position within that lobe; this false indication is also appropriately computed by MNOPLS1.

When there are several point targets close enough to interfere, but only a single path between the radar and each target ($N_p = 1$), subroutine MNOPLS is called to compute the monopulse angle indication. This subroutine was qualitatively described in Sec. 4.6.2; details will be given below.

When refraction through a striated region is encountered ($N_p > 1$), subroutine MNOPLSM is called to compute the angle indication. This subroutine was described qualitatively in Sec. 4.6.3; details will be given below.

C.1 GENERAL DEFINITIONS AND PROCEDURES

Assume a circular mainlobe centered on the instantaneous beam axis (u_o, v_o). The 3-dB one-way beamwidth, on boresight, is the input quantity $\delta\theta$. Because the beam size and shape change in angular coordinates as the beam departs from boresight, sine-space coordinates are preferred. The

*That is, correct except for possible errors due to refraction.

half-beamwidth in sine-space is defined to be

$$\delta w = \sin(\delta\theta/2)$$

In the model, the angular sidelobes of the radar beam are represented by an amplitude, S_s, the one-way power level relative to the beam axis at boresight (input by the user), which either remains constant or is tapered, depending on the model type.

For a given (point) target with sine-space coordinates (u,v), the angle off the beam axis in sine-space is given by

$$\xi = [(u - u_o)^2 + (v - v_o)^2]^{1/2}$$

The shape of the one-way mainlobe of the sum beam is represented by the relative gain factor, $F(\xi)$, defined to be unity on the beam axis. In the present circular-beam model, the mainlobe is represented by a $[(\sin x)/x]^2$ function with the argument adjusted so that the 3-dB point comes at the right place. Within most of the mainlobe this is a good approximation to a wide variety of real antenna patterns. Thus the mainlobe relative gain function is

$$F(\xi) = \left[\frac{\sin P\xi}{P\xi}\right]^2 \quad , \quad \text{for} \quad |\xi| \leq \pi/P$$

where $P \equiv 1.392/\delta w$.

C.2 POINT TARGETS

Let R'_o be the indicated range of the target currently being tracked, and let $\delta R'_1$ be the effective half-width of the range resolution cell. MNOPLS considers only targets with ranges R'_i that lie within interfering range of R'_o, i.e., such that $|R'_i - R'_o| \leq \delta R'_1$.

Taking into account doppler mismatch and the waveform ambiguity function, the signal amplitude in the sum channel at R'_o due to target i at

R_i' is proportional to

$$A_i = \frac{F(\xi_i)}{(R_i')^2} \sqrt{\sigma_i}\, \chi_N(R_o' - R_i')$$

where χ_N is the (normalized) waveform ambiguity function and σ_i is the effective cross section of target i.

The signal phase in the sum channel at R_o' due to target i at R_i' is, apart from an additive constant

$$\phi_i = \frac{4\pi}{c} [f_o + (1/2)f_d(i)]R_i'$$

provided that the target lies within the angular mainlobe. For targets in the sidelobes, an extra term of value π should be added whenever the target lies in a "negative" sidelobe. However, the model ignores this extra phase factor on the grounds that such targets are almost always greatly dominated by those in the mainlobe.

The total sum-channel signal voltage (amplitude and phase) at R_o' can be expressed in complex notation as

$$V_s = \sum_i A_i e^{-j\phi_i}$$

In the difference channels, it can be shown that for a typical kind of monopulse circuitry, the amplitude due to the ith target is related to that in the sum channel by an expression

$$A_i(d) = -jA_i(s) \tan \frac{P}{2} (u_i - u_o)$$

for the u-plane difference, with a similar expression for the v-plane. Thus the u-plane difference voltage at R_o' due to all contributing targets can be expressed as

$$V_{du} = -j\sum_i A_i \tan \frac{P}{2}(u_i - u_o)e^{-j\phi_i}$$

and similarly in the v-plane.

The real and imaginary parts of the expressions are the quadrature components of the signals. For the u-plane, these are

$$S_R \equiv \sum_i A_i \cos \phi_i$$

$$S_I \equiv -\sum_i A_i \sin \phi_i$$

$$D_R \equiv -\sum_i A_i \tan \frac{P}{2}(u_i - u_o) \sin \phi_i$$

$$D_I \equiv -\sum_i A_i \tan \frac{P}{2}(u_i - u_o) \cos \phi_i$$

It can be shown that the indicated u-coordinate of the "target" in the beam is given by

$$u' = u_o + \frac{\Theta(u)}{P}$$

where

$$\Theta(u) \equiv \arctan\left\{\frac{2[D_R S_I - D_I S_R]}{S_R^2 + S_I^2 - D_R^2 - D_I^2}\right\}$$

A similar expression applies to v', the indicated v-coordinate.

If there are, in fact, no targets in the beam, the circuitry will act on noise alone to produce a false angular position estimate located somewhere at random within the mainlobe. Subroutine MNOPLS does this when LISTA contains no targets.

If there is only a single point target in LISTA which can contribute to the monopulse determination, the above expressions become so much simplified that a special subroutine, MNOPLS1, was written to deal with this case. If this target point lies within the mainlobe of the beam, then the indicated position is just the true position of the target, i.e., $u' \equiv u$ and $v' \equiv v$. However, if this target point lies outside the mainlobe, the system will produce a false (but determinate, provided that the noise is sufficiently low) angular position indication. This is appropriately treated in subroutine MNOPLS1.

C.3 SPREAD TARGET CLOUDS

For spread target clouds ($N_p > 1$), the monopulse model is contained in subroutine MNOPLSM. Its basic operations are the same as those used in subroutine MNOPLS for point targets, except for obvious required modifications. For example, if any significant portion of a target cloud (not just its center) lies within the range resolution cell, that portion will contribute to the combined interference signal which is processed by the monopulse system. The amplitudes of each "target" signal are computed in a manner similar to that used for point targets, except that the target cloud particle density distribution must be convolved with the waveform ambiguity function to compute the amplitude contribution.

Furthermore, because of the statistical nature of the "targets," the random phase approximation is used in place of adding the in-phase and quadrature components. This changes the final equations somewhat, but in an obvious manner. The relevant equations are:

$$A_i = \frac{F(\bar{\xi}_i)}{(R_i')^2} \sqrt{\sigma_i}\, \chi_N[(R_o' - R_i')/b]$$

where $\bar{\xi}_i$ is the effective off-axis position for the given cloud (as computed by subroutine MULTOAR) and b is a correction factor related to the convolution of cloud statistics with the ambiguity function (see Appendix E).

The phase ϕ_i is now irrelevant. MNOPLSM forms the power sums

$$S \equiv \sum_i A_i^2$$

and

$$D_u \equiv \sum_i A_i^2 \tan^2 \frac{P}{2} (u_i - u_o)$$

with a similar expression for D_v.

Then it turns out that

$$u' = u_o + \frac{\Theta(u)}{P}$$

where

$$\Theta(u) \equiv \arctan\left\{\frac{2\sqrt{D_u \cdot S}}{S - D_u}\right\}$$

and a similar expression for v'.

As before, appropriate actions are taken for targets which are too small or too far out of the mainlobes to contribute anything but essentially random noise.

APPENDIX D

THE SPLIT-GATE RANGE TRACKER

A simple split-gate range tracker is used in ROSCOE, matched to the single approximate pulse-compression waveform. It is, of course, applicable only to those events for which a range gate is defined. There are two subroutines which embody the split-gate tracker: subroutine SPLTGAT for use when $N_p = 1$, and subroutine MLTSPLT for use when $N_p > 1$. These were described in a brief qualitative way in Secs. 4.6.4 and 4.6.5. Further details will be given below.

The basic principle of the split-gate range tracker is to integrate the video signal which is the output of the matched receiver filter over two range regions which together cover the total gate width. If W is the range-gate width, R_c its center range, and $F(R)$ the video output envelope as a function of range, then in effect the system forms the integrals

$$I^- \equiv \int_{R_c - \frac{W}{2}}^{R_c} F(R)\, dR$$

and

$$I^+ \equiv \int_{R_c}^{R_c + \frac{W}{2}} F(R)\, dR$$

Then the sum and difference:

$$T \equiv I^+ + I^-$$

and

$$\epsilon \equiv I^+ - I^-$$

are taken. In the form used in the ROSCOE model, the error signal, Δ,

is related to the ratio ε/T by a known function, by means of which the "true" range of the target relative to R_c is determined.

If there is only a single target in the range gate, and if there is no dispersive distortion of the ambiguity function, then the form of $F(R)$ is simply $|\chi(R - R_t)|$, where R_t is the true range of the target. Assuming the waveform to be known, one can then in principle find R_t as a function of ε/T not only for $\varepsilon/\delta R_1 << 1$, as is the usual approximation, but in fact for all values of ε. It should be noted, however, that the apparent range given by this model will always lie within the range gate, wherever the true target may be; in other words, even if there is no target, or there are many targets, within the range gate, the algorithm will produce a value of R_t lying between $R_c - (W/2)$ and $R_c + (W/2)$. This is a desirable effect for the model, however, since it is equivalent to what a real split-gate tracker would do under the same circumstances. Similarly, if the waveform has been distorted by dispersive effects, the appropriate incorrect range will be found.

In addition to R_t, the split-gate algorithm also outputs a "peak signal power factor," F_M, which is proportional to the amplitude of the target signal in the case of a single target and non-dispersive propagation. In other cases, an appropriate erroneous value of F_M is found.

Assuming, therefore, a single point target located at R_t, and assuming further that the ambiguity function in range has the approximate form

$$\chi(R) = \left| \frac{\sin(\pi R/\delta R_1)}{\pi R/\delta R_1} \right|^{1.7}$$

as discussed in Appendix B, it can easily be shown that

$$\varepsilon = \frac{2}{\pi} \delta R_1 S\left(\frac{\pi \Delta}{\delta R_1}\right) F_M$$

and

$$T = \frac{2}{\pi} \delta R_1 S(\pi) F_M$$

where

$$S(x) \equiv \int_0^x \left(\frac{\sin t}{t}\right)^{1.7} dt$$

Since the function $S(x)$ can be tabulated numerically, its inverse, $x \equiv Q(S)$ can also be tabulated. Then if we define

$$s \equiv S(\pi)\varepsilon/T$$

the error signal, Δ, can be found at once using

$$\Delta = \frac{\delta R_1}{\pi} Q(s)$$

whence the target range is given by

$$R_t = R_c + \Delta$$

The inverse function $Q(s)$ is computed by function QINV (Sec. 4.6.10); it should be noted that

$$\text{sign}(\Delta) = \text{sign}(\varepsilon)$$

and that if $|s| > S(\pi)$ then $|\Delta| \equiv \delta R_1$. The peak power factor, F_M, is given by

$$F_M = T \Big/ \left[\frac{2}{\pi} \delta R_1 S(\pi)\right]$$

The algorithm used in subroutine SPLTGAT for the cases where there is only a single ray path between the radar and each real target ($N_p = 1$) begins by adding the in-phase and quadrature components of the returns from each interfering target at each point within the range-gate, weighted according to the appropriate value of χ at the given point, integrating to find I^+ and I^-, and then proceeding as described above.

The algorithm used in subroutine MLTSPLT for the case of refraction through a striated region ($N_p > 1$) uses the statistical properties of the "spread target cloud" as found by subroutine REFRCT to approximate the video signal envelope which is input to the range-tracker. Because of the statistical nature of this model, in-phase and quadrature components are <u>not</u> computed for each target image; instead, the random-phase approximation is used in combining the contributions of different images to the video signal. See also the discussion in Appendix E.

APPENDIX E

THE SPREAD TARGET CLOUD IN THE RADAR BEAM

When the ray path between a target and the radar intersects a striated debris region, numerous multiple paths are generated ($N_p > 1$), giving rise to the so-called "spread target cloud" of images. In ROSCOE, these images are described statistically, with the parameters

$$\Delta R,\ \Delta u,\ \Delta v,\ \sigma_{\Delta R},\ \sigma_1,\ \sigma_2,\ \mu,\ \bar{\sigma},\ \sigma_\sigma$$

as defined in Appendix G and discussed in Appendix B. For a point target whose true radar coordinates are (R,u,v), the center of its apparent "cloud" distribution is at

$$R_1 = R + \Delta R + I_{pc} \frac{\delta R'_o}{B} (f_d - \bar{f}_d)$$

$$u_1 = u + \Delta u$$

$$v_1 = v + \Delta v$$

where $I_{pc} = 1$ if the radar is operating in a pulse-compression mode, and $I_{pc} = 0$ if not.

This appendix describes the mathematical modeling of the interaction between a spread target cloud so defined and a radar beam.

E.1 INTERACTION WITH THE RADAR ANGULAR BEAM

In sine-space, the cloud can be represented by an error ellipse centered at (u_1,v_1) with semi-major axis of length σ_1 and semi-minor axis of length σ_2 (these define the one-sigma error ellipse). Its orientation is defined by stating the angle, μ, between the major axis of the ellipse and the u-axis of the sine-space coordinate system (to eliminate possible ambiguities, we select $0 \leq \mu < \pi$).

The mainlobe of a circular radar beam centered at (u_o,v_o) can be described by its one-way relative gain factor

$$F(\xi) \equiv \left(\frac{\sin P\xi}{P\xi}\right)^2$$

as described in Appendix C, Sec. C.1. Here we need only recall that P is a known constant and

$$\xi^2 \equiv (u - u_o)^2 + (v - v_o)^2$$

In order to describe the relation of the cloud error ellipse to the radar beam pattern, it is convenient to use a rotated set of sine-space coordinate axes, (u',v'), which have been rotated about (u_o,v_o) through the angle μ, so that u' is parallel to the major axis of the ellipse and v' is parallel to the minor axis. Then the new and old coordinates of any point are related by:

$$u' - u_o = + (u - u_o) \cos \mu + (v - v_o) \sin \mu$$

$$v' - v_o = - (u - u_o) \sin \mu + (v - v_o) \cos \mu$$

Using this rotated coordinate system, the effective one-way relative power received from the cloud can be approximated by

$$F_1 \simeq \iint_{-\infty}^{\infty} F(\xi) \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left[-\frac{(u' - u_1')^2}{2\sigma_1^2}\right] \frac{1}{\sqrt{2\pi}\sigma_2} \exp\left[-\frac{(v' - v_1')^2}{2\sigma_2^2}\right] du'dv'$$

provided that σ_1 and σ_2 are sufficiently small that the integrand has negligible value outside regions where $\sin x \simeq x$.

This integral can be solved by using the exponential approximation to the function $F(\xi)$; i.e.,

$$\frac{\sin x}{x} \approx e^{-Ax^2}$$

where the constant A can be chosen to force agreement at any particular point. If the agreement is to occur at the half-power point, then the constant is $A \approx 0.1788$. Making use of the fact that

$$\xi^2 \equiv (u' - u_o)^2 + (v' - v_o)^2$$

the double integral takes the approximate form

$$F_1 \approx \frac{1}{2\pi\sigma_1\sigma_2} \iint_{-\infty}^{\infty} \exp\left\{ - 2AP^2\left[(u' - u_o)^2 + (v' - v_o)^2\right] - \frac{(u' - u_1')^2}{2\sigma_1^2} - \frac{(v' - v_1')^2}{2\sigma_2^2} \right\} du'dv'$$

This integral can be solved exactly. The result is

$$F_1 \approx \left[(K_1^2 + 1)(K_2^2 + 1)\right]^{-1/2} F(\bar{\xi}) \exp\left\{ - \frac{1}{2}\left[\frac{K_1^4}{\sigma_1^2}(\bar{u}' - u_o)^2 + \frac{K_2^4}{\sigma_2^2}(\bar{v}' - v_o)^2\right]\right\}$$

where

$$K_i^2 \equiv 4AP^2\sigma_i^2$$

$$\bar{u}' \equiv \frac{K_1^2 u_o + u_1'}{K_1^2 + 1}$$

$$\bar{v}' \equiv \frac{K_2^2 v_o + v_1'}{K_2^2 + 1}$$

and

$$\bar{\xi}^2 \equiv (\bar{u}' - u_o)^2 + (\bar{v}' - v_o)^2$$

The point $(\bar{u}',\bar{v}')$ [which can be directly transformed into the unprimed coordinate system as $(\bar{u},\bar{v})$] represents the <u>effective</u> center of the received power from the target cloud as mediated by the antenna beam pattern. In the same way, $\bar{\xi}$ represents the angle off beam axis (in sine-space) of the effective center of the spread target cloud, and F_1 is the effective one-way relative gain factor of the cloud. In this way, we essentially have reduced the cloud to a fictitious point. The quantities $(\bar{R},\bar{u},\bar{v})$, $\bar{\xi}$, and F_1 are computed in subroutine MULTOAR (here $\bar{R} \equiv R_1$).

E.2 INTERACTION WITH THE AMBIGUITY FUNCTION IN RANGE

We have postulated that there are N_P^2 target images in the cloud, with a Gaussian range distribution centered at $r = R'$ and having standard deviation $\sigma_{\Delta R}$ (which in the present subsection will be renamed σ_r for notational simplicity). It is assumed that $\sigma_r << R'$, and of course $N_P^2 >> 1$. The number distribution of the target images in the cloud is

$$N(r) = \frac{N_P^2}{\sqrt{2\pi}\sigma_r} \exp\left[-\frac{(r - R')^2}{2\sigma_r^2}\right]$$

It is further assumed that the mean cross section of the individual target images in the cloud is $\bar{\sigma}$, with standard deviation σ_σ. The total cross section of the entire cloud, if observed as a whole at great distance, is $\sigma_T = N_P^2\bar{\sigma}$.

As before, let $\chi_N(r - r_1)$ be the waveform ambiguity function (normalized to unity at $r = r_1$) giving the output amplitude at r due to a unit target at r_1. Thus the amplitude distribution at a point r due to the target images located around point r_1 is given by

$$P(r) = \sqrt{\bar{\sigma}}\chi_N(r - r_1)N(r_1)$$

The total amplitude at point r is found in principle by integrating $P(r)$ over all values of r_1, taking into account the appropriate phases of each contribution. Because of the statistical nature of the model, however, the random phase approximation may be used; here this means that we add the squared amplitudes, ignoring phase, and take the square root at the end. Thus

$$[A(r)]^2 \approx \bar{\sigma} \int_{-\infty}^{\infty} [\chi_N(r - r_1)]^2 N(r_1)\, dr_1$$

It will be assumed that the ambiguity function in range can always be approximated (where it is of significant magnitude) by an expression of the form

$$\chi_N(x) \approx \exp[-a^2x^2]$$

where a is a known constant, depending on the particular waveform involved.

With the approximation the integral can easily be evaluated to yield

$$A(r) \approx \sqrt{\bar{\sigma}}N_p b^{-1/2} \exp[-a^2(r - r')^2/b^2]$$

where

$$b^2 \equiv 1 + 4a^2\sigma_r^2$$

The peak value of $A(r)$ occurs at $r = R'$ and has the amplitude

$$A_o \approx \sqrt{\bar{\sigma}}N_p b^{-1/2}$$

This may be interpreted to mean that the effective range of the cloud center is at

$$\bar{R} \equiv R'$$

and the effective cross section of the target cloud as mediated by the ambiguity function of the signal waveform is

$$\sigma_E \simeq N_P^2 \bar{\sigma}/b \equiv \bar{\sigma}_T/b$$

Notice that the effective cross section of the target cloud as seen by a reasonable radar waveform is always less than its ideal total cross section.

For the two kinds of waveforms used in ROSCOE, the value of the parameter b will now be calculated.

Simple unchirped rectangular pulse. The ambiguity function (in range) for this waveform is shown in Appendix B to be approximated by

$$\chi_N(x) \simeq \exp[-0.884\ (2x/\delta R_o')^2]$$

whence it is seen that

$$b^2 \simeq 1 + 14.14 \left(\frac{\sigma_r}{\delta R_o'}\right)^2$$

Chirped pulse. The range ambiguity function for the single chirped-pulse approximation used in ROSCOE was defined in Appendix B to be

$$\chi_N \simeq \left[\frac{\sin(\pi x/\delta R_1')}{\pi x/\delta R_1'}\right]^{1.7}$$

Making use of the exponential approxiation to the $\frac{\sin x}{x}$ function, the value of b^2 can be computed to be

$$b^2 \simeq 1 + 12.00\left(\frac{\sigma_r}{\delta R_1'}\right)^2$$

E.3 MULTIPLE CLOUDS IN RANGE AND ANGLE

If there is more than one spread target cloud in the radar range gate, and their combined interference function must be found (as for example in subroutine MLTSPLT, SLNINTM, and MNOPLSM), the following extension of the above theory is used.

Suppose there are n overlapping target cloud distributions, all having the same value of N_p and σ_r but each ahving its own value of R_i' and $\bar{\sigma}_i$. When combining these distributions with respect to the same radar waveform, we may use the random phase approximation as above, but modified so that each individual value of $N_p^2\bar{\sigma}_i$ is replaced by the appropriately weighted value

$$X_i \equiv N_p^2\bar{\sigma}_i F_i^2/(R_i')^4$$

Here the factor $(R_i')^{-4}$ takes into account the different ranges of the cloud centers, and $(F_i)^2$ is the relative gain factor (two-way, this time) to account for the different positions of the clouds with respect to the antenna beam pattern. Replacing the factor $N_p\sqrt{\bar{\sigma}}$ in the equation for $A(r)$ by $\sqrt{X_i}$, and noting that the exponential $\exp[-a^2(r - r')^2/b^2]$ in that equation is merely an approximation to the ambiguity function $\chi_N[(r - R')/b]$, the combined amplitude at point r is given by

$$A(r) \simeq \left\{b^{-1}\sum_{i=1}^{n} X_i\left[\chi_N\left(\frac{r - R_i'}{b}\right)\right]^2\right\}^{1/2}$$

APPENDIX F

STOCHASTIC REFRACTION MODEL

F.1 THIN SCREEN APPROXIMATION

In simulations of radar propagation through a disturbed environment it is desirable to be able to determine the statistical descriptors of the beam as it passes through a medium whose properties vary with position. The medium will generally have a smoothly varying background electron density plus an additional fluctuating density which varies much more rapidly with position. If the fluctuating part of the electron density does not lead to excessive departures of a ray in the beam from its nominal smooth trajectory, it is possible to develop a theory which describes, in statistical terms, the effects of the electron density fluctuations.

The thin screen approximation provides a description of the propagation of electromagnetic radiation through an ionized region of space, provided the following three conditions are met:

1. Ray Optics. The electron density gradients must be sufficiently small that the fractional change in electron density over distances comparable to a wavelength is small. In addition, the attenuation over distances comparable to a wavelength must be small.

2. Straight-Line Approximation. The ray deflections must be sufficiently small that electron densities and density gradients along the true ray path may be estimated by considering a nominal straight-line ray path.

3. Small Electron Densities. The electron densities must be sufficiently small to permit a series expansion in powers of the electron density.

The name "thin screen approximation" is a slight misnomer since the basic assumptions may remain valid even though the path length through the disturbed region is fairly long. However, for a given angular deflection there is a limiting path length beyond which the straight-line approximation breaks down.

We consider a region with a magnetic field in the $\hat{z}$-direction and an electron density $n_e(x,y)$ which is independent of z. In the absence of the electrons, a wave packet of radiation moves in a straight line with the speed of light, and we choose our coordinate system so that the nominal ray lies in the $\hat{x}$, $\hat{z}$-plane, making an angle γ with the z-axis.

The electric field then has the form

$$E = E_o \exp(ik\psi \sin\gamma + ikz\cos\gamma - i\omega t)$$

$$= E_o \exp(i\phi - i\omega t)$$

where ψ is a function only of x and y, satisfying the equation

$$(\nabla\psi)^2 = 1 - \frac{n_e}{n_{cr}\sin^2\gamma} = 1 - \sigma_e$$

according to the ray optics approximation. Here n_{cr} is the critical electron density at the frequency f, and $k = 2\pi/\lambda$. This equation determines the electric field at an arbitrary point in space if the value of ψ is known on the plane $x = x_o$. If the electron densities are sufficiently small, ψ may be expanded in powers of σ_e so that

$$\psi = \psi_o + \psi_1 + \psi_2 + \ldots$$

On the surface $x = x_o$, $\psi(x_o,y) = x_o + \tilde{\psi}(x_o,y)$, where $\tilde{\psi}$ is assumed to be a first-order quantity. We therefore take $\psi_o(x,y) = x$, and this describes the nominal straight-line path.

The the first-order corrections to ψ are determined by

$$2\nabla\psi_o\cdot\nabla\psi_1 = -\sigma_e$$

so that

$$2\frac{\partial\psi_1}{\partial x} = -\frac{1}{2}\sigma_e$$

and

$$\psi_1(x,y) = \tilde{\psi}(x_o,y) - \frac{1}{2}\int_{x_o}^{x}\sigma_e(x',y)\,dx'$$

Similarly, the second-order terms give

$$2\frac{\partial\psi_2}{\partial x} = -(\nabla\psi_1)^2 = -\frac{1}{4}\left(\sigma_e^2\right) - \theta^2$$

where

$$\theta = \frac{\partial\psi_1}{\partial y}$$

is the angle that the projection of the ray path onto the x,y-plane makes with the x-axis (to first order). Thus

$$\psi_2(x,y) = -\frac{1}{8}\int_{x_o}^{x} \sigma_e^2(x',y)\,dx' - \frac{1}{2}\int_{x_o}^{x} \theta^2(x',y)\,dx'$$

On the basis of these approximations to the electric field it is possible to evaluate some physically interesting quantities. Thus the z-component of the velocity of a wave packet is $c \cos\gamma$, and the velocity components in the x,y-plane are given by $c \sin\gamma\,\nabla\psi$. Considering only the motion in the x,y-plane, it is clear that a ray takes a time T to traverse a given path which is given by

$$cT \sin\gamma = \int \frac{d\ell}{|\nabla\psi|} = \int \frac{dx}{|\partial\psi/\partial x|}$$

and associated with this time interval there is an apparent range given by $R = cT$. It is convenient to roughly separate the contributions to the apparent range into a part PL, which is the path length along the ray trajectory, and to associate any additional contributions with the group velocity of the wave packet. Thus

$$R \sin\gamma = cT \sin\gamma = PL + cT_{del} \sin\gamma$$

with

$$PL = \int d\ell = \int \frac{dx}{|\partial\psi/\partial x|}\,|\nabla\psi|$$

and

$$cT_{del} \sin\gamma = \int \frac{d\ell}{|\nabla\psi|} - \int d\ell$$

Since $\partial\psi/\partial x$ is known, and $|\nabla\psi| = (1 - \sigma_e)^{1/2}$, it is a simple matter to expand these expressions in powers of σ_e. In evaluating the

expressions obtained, it must be remembered that y varies with x along a ray trajectory so that

$$\frac{dy}{dx} = \frac{\partial\psi/\partial y}{\partial\psi/\partial x} \approx \theta$$

The results of such an expansion are

$$PL = x - x_o + r_{dl}$$

where the "dogleg" range error is given by

$$r_{dl} = \frac{1}{2} \int_{x_o}^{x} \theta^2(x',y')\, dx'$$

and

$$cT_{del} \sin\gamma = \frac{1}{2} \int_{x_o}^{x} \sigma_e(x',y')\, dx' + \frac{3}{8} \int_{x_o}^{x} \sigma_e^2(x',y')\, dx'$$

so that the apparent range for a ray going from x_o to x is

$$R = x - x_o + \frac{1}{\sin\gamma} r_{dl} + cT_{del}$$

In applications, the second-order contribution to T_{del} is generally negligible compared to the first-order term, and may be ignored. However, because of geometric factors, the dogleg term r_{dl} can be appreciable, and should be retained.

The phase of a ray moving through the medium changes so that

$$\delta\phi = k \sin\gamma\, \delta\psi + k \cos\gamma\, \delta z$$

$$= k \sin\gamma\, |\nabla\psi|\, d\ell + k \cos^2\gamma\, \delta T$$

since in the x,y-plane $\delta r = \nabla\psi\delta\ell/|\nabla\psi|$. It is then easily seen that the phase change in moving from x_o to x is given by

$$\Delta\phi = kR - 2kcT_{del} \sin^2\gamma$$

In deriving the last equation, an approximation has been made in that some liberties were taken with the second-order contributions to T_{del}.

F.2 STOCHASTIC REFRACTION MODEL

(U) In the last section an approximate expression for the electric field was obtained in terms of the electron density distribution. In practice the electron density function is often not known in detail but must be described in statistical terms. We therefore consider a model in which the electron density fluctuations are due to field aligned striations so that

$$\frac{n_e(x,y)}{n_{cr}} = \bar{\nu}(x) + \sum_k \nu_k \exp - \frac{1}{a_k^2}\left[(x - x_k)^2 + (y - y_k)^2\right]$$

where the location of the striation axes (x_k,y_k), the peak electron density of the striation $\nu_k n_{cr}$ and the striation size a_k are not known precisely. Instead we assume (x_k,y_k) are randomly distributed in the x,y-plane with a two-dimensional density function $\rho(x)$, that ν_k and a_k are also randomly distributed, and that the average moments $\left\langle \nu^{n_1} a^{n_2} \right\rangle_k$ are known as functions of x . It is then desired to calculate the expected value of various physically interesting quantities; this is done in the following subsections.

F.2.1 Integrated Electron Density

The quantity

$$d(x,y) = \int_{x_o}^{x} dx' \, \frac{n_e(x',y)}{n_{cr}}$$

is linearly related to the absorption due to electron-neutral collisions and is also useful in that several other quantities are readily calculated once d is known. When a ray passes the kth striation there is a contribution to $d(x,y)$ given by*

$$d_k(y) = \nu_k \exp\left[-(y - y_k)^2/a_k^2\right] \int_{-\infty}^{\infty} \exp\left(-x^2/a_k^2\right) dx$$

$$= \nu_k a_k \pi^{1/2} \exp\left[-(y - y_k)^2/a_k^2\right]$$

and

$$d(x,y) = \int_{x_o}^{x} \bar{\nu}(x') \, dx' + \sum_j d_j$$

where the sum over j extends over all striations in the region $x_o < x_j < x$ and $-\infty < y_j < \infty$.

The average contribution of those striations in the interval from x' to $x' + dx$ can be obtained by replacing the sum over j by an integral over the possible positions of the striation centers so that

*(U) The use of the limits $-\infty$ to $+\infty$ for the x-integration is justified by the assumption that a_k is small compared to other scale sizes of the problem.

$$\left\langle \sum_j d_j(y) \right\rangle \to \rho(x')\, dx' \int_{-\infty}^{+\infty} \left\langle \nu a(\pi)^{1/2} e^{-y^2/a^2} \right\rangle dy$$

where the expectation value under the integral sign is an average over the distribution in ν and a. Thus

$$D(x) = \left\langle d(x,y) \right\rangle = \int_{x_o}^{x} \bar{\nu}(x')\, dx' + \pi \int_{x_o}^{x} \rho(x') \left\langle \nu a^2 \right\rangle dx'$$

F.2.2 Fluctuations in Integrated Electron Density

Consider

$$\left(d(x,y) - \int_{x_o}^{x} \bar{\nu}(x')\, dx' \right)^2 = \sum_i d_i(y) \sum_j d_j(y)$$

As before the average value of this quantity may be obtained by replacing the sums by integrals. However, the terms with $i = j$ must be considered separately since they also give rise to a finite contribution. Thus

$$\sum_i d_i(y) \sum_j d_j(y) = \left(\pi \int_{x_o}^{x} \rho(x') \left\langle \nu a^2 \right\rangle dx' \right)^2$$

$$+ \int_0^x dx'\, \rho(x') \left\langle \int_{-\infty}^{+\infty} \nu^2 a^2 \pi e^{-2y^2/a^2}\, dy \right\rangle$$

where the second term on the right-hand side comes from the terms with $i = j$. Thus the fluctuations in $d(x,y)$ are given by

$$\sigma_D^2(x) = \left\langle \left[d(x,y) - D(x) \right]^2 \right\rangle = 2\left(\frac{\pi}{2}\right)^{3/2} \int_{x_o}^{x} \rho(x') \left\langle \nu^2 a^3 \right\rangle dx'$$

F.2.3 Emergence Angle

Each striation scatters the ray through an angle

$$\theta_k^s = \frac{\nu_k \pi^{1/2} (y - y_k)}{a_k \sin^2 \gamma} \exp\left[- (y - y_k)^2 / a_k^2 \right]$$

so that the ray makes an angle with the $\hat{x}$-direction given by

$$\theta(x) = \sum_k \theta_k^s$$

which is easily seen to have zero expectation value. The square of this angle has expectation value

$$\sigma_\theta^2(x) = \left\langle \theta^2(x) \right\rangle = \left\langle \sum_j \sum_k \theta_j^s \theta_k^s \right\rangle = \left\langle \sum_j (\theta_j^s)^2 \right\rangle$$

and replacing the sum by an integral gives the result

$$\sigma_\theta^2(x) = \int_{x_o}^{x} \frac{d\theta^2}{dx'} dx'$$

with

$$\frac{d\theta^2}{dx} = \left(\frac{\pi}{2}\right)^{3/2} \frac{1}{2 \sin^4 \gamma} \rho(x) \left\langle v^2_a \right\rangle$$

F.2.4 Miss Distance

A ray will suffer a displacement in the y-direction given by

$$\Delta y(x) = \sum_k (x - x_k)\theta_k^s$$

which has zero mean. The expected value of its square is readily found and

$$\sigma^2_{\Delta y}(x) = \left\langle \left[\Delta y(x)\right]^2 \right\rangle = \int_{x_o}^{x} (x - x')^2 \frac{d\theta^2}{dx'} dx'$$

F.2.5 Range Error

A ray in propagating from the kth to the k + 1th striation makes an angle θ_k with the x-axis given by

$$\theta_k = \sum_{j=1}^{k} \theta_j^s$$

and the distance it must travel exceeds the nominal distance $x_{k+1} - x_k$ by an amount $(1/2)(x_{k+1} - x_k)\theta_k^2$. Thus in traversing the medium there is a "dogleg" range error given by

$$r_{dl} = \frac{1}{2} \sum_k (x_{k+1} - x_k)\theta_k^2$$

which has expectation value

$$R_{dl} = \frac{1}{2} \int_{x_o}^{x} \sigma_\theta^2(x')\, dx' = \frac{1}{2} \int_{x_o}^{x} (x - x') \frac{d\theta^2}{dx'}\, dx'$$

The total range error consists of this term plus a time delay term

$$R_e = \frac{1}{\sin \gamma} R_{dl} + cT_{del}$$

where

$$cT_{del} = \frac{1}{2 \sin^3 \gamma} D$$

F.2.6 Range Fluctuations

Consider first

$$r_{dl}^2 = \frac{1}{4} \sum_m (x_{m+1} - x_m)\theta_m^2 \sum_n (x_{n+1} - x_n)\theta_n^2$$

It is easily seen that the terms with $m = n$ give a vanishing contribution to the expectation value of r_{dl}^2 so that it is sufficient to consider $n_1 > m$ and multiply by 2. For $n > m$,

$$\left\langle \theta_n^2 \theta_m^2 \right\rangle = \left\langle \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{\ell=1}^{m} \theta_i^s \theta_j^s \theta_K^s \theta_\ell^s \right\rangle$$

$$= \sigma_\theta^2(x_n)\sigma_\theta^2(x_m) + 2\left[\sigma_\theta^2(x_m)\right]^2 + 2\psi(x_m)$$

where the first term comes from $i = j$, $k = \ell$, and the next term comes from $i = k$, $j = \ell$, and $i = \ell$, $j = k$ contributions, and $i = j = k = \ell$ gives rise to

$$\psi(x) = \frac{1}{2}\left\langle \sum_i (\theta_i^s)^4 \right\rangle = \frac{3}{8}\left(\frac{\pi}{4}\right)^{5/2} \frac{1}{\sin^8 \gamma} \int_{x_o}^{x} \rho(x') \left\langle \nu^4 a \right\rangle dx'$$

so

$$\sigma_{dl}^2(x) = \int_{x_o}^{x} (x - x') \left\{ \left[\sigma_\theta^2(x') \right]^2 + \psi(x') \right\} dx'$$

The cross correlation between the dogleg and line delay terms is given by

$$X_{R_e} = \left\langle ct_{del} r_{dl} \right\rangle - cT_{del} R_{dl}$$

$$= \frac{1}{4 \sin^4 \gamma} \sum_k (x_{k+1} - x_k) \sum_{j=1}^{k} d_j (\theta_j^s)^2$$

$$= \frac{3}{16} \pi^2 \left(\frac{1}{3}\right)^{3/2} \frac{1}{\sin^8 \gamma} \int_{x_o}^{x} (x - x') \rho(x') \left\langle \nu^3 a^2 \right\rangle dx'$$

and

$$\sigma_{R_e}^2 = \frac{1}{4 \sin^6 \gamma} \sigma_D^2 + 2X_{R_e} + \sigma_{dl}^2$$

F.2.7 **Beam Defocussing**

Consider two rays which initially make angles with the $\hat{x}$-axis which differ by $\Delta\theta_o$. After propagating through the medium, two such rays will be separated by a distance

$$\delta y(x) = x\Delta\theta_o + \sum_k (x - x_k) \frac{\partial}{\partial y} \theta_k^s \delta y(x_k)$$

The sum over k has zero expectation so that it is necessary to consider $(\delta y)^2$. (Note that $\delta y(x_k)$ and $\partial\theta_k^s/\partial y$ are statistically independent.) If we let $S(x) = \left\langle \left(\frac{\delta y(x)}{\Delta\theta_o} \right)^2 \right\rangle$ then

$$S(x) = x^2 + \frac{3}{2}\left(\frac{\pi}{2}\right)^{3/2} \frac{1}{\sin^4 \gamma} \int_{x_o}^{x} (x - x')^2 \rho(x') \left\langle \nu^2 a^{-1} \right\rangle S(x')\, dx'$$

However, because of the presence of S on the right-hand side the final result for S involves an expansion containing all powers of the peak electron density. Since in the underlying theory all powers of the peak electron density higher than the second have been ignored, these higher order sums in S are not reliable. Therefore, in applications it is prudent to replace S by its "Born approximation" $S(x) = x^2$ when it occurs under the integral sign.

The gain G due to focussing is then

$$G(x) = [x^2/S(x)]^{1/2}$$

It must be emphasized at this point that the gain (always less than unity) given by this equation is associated with a single ray path connecting the radar to the target. In general, there may be many such

one-way images and the total energy on the target contains a factor $N_i G$. If the radar beamwidth is sufficiently large, it can be argued that $N_i G$ is on the average equal to 1. The energy in the beam sweeps through any region of space on the average N_i times at the range considered. If it is assumed the electric fields associated with different rays are incoherent, then the energies associated with different rays add and energy conservation demands that $N_i G = 1$. The angular separation between rays at the radar associated with successive images can also be estimated. As before, denote the miss distance by Δy. During an angular interval such that $\frac{dy}{d\theta} > 0$ a ray will pass through an interval of angle such that Δy starts at a negative minimum, hits the target, and then rises to a local maximum. If the root mean squared angle traversed in going from a minimum to a maximum is $\Delta\theta_i$ then

$$\Delta\theta_i = 2\left[\frac{\langle(\Delta y)^2\rangle}{\left\langle\left(\frac{dy}{d\theta}\right)^2\right\rangle}\right]^{1/2} = 2\frac{\sigma_{\Delta y}}{S^{1/2}}$$

$$= 2\frac{\sigma_{\Delta y} G}{x}$$

Thus if the antenna beamwidth is γ then the number of images is

$$N_i = \frac{\gamma x}{2\sigma_{\Delta y} G}$$

and the energy incident on the target is multiplied by a factor $(\gamma x)/(2\sigma_{\Delta y})$.

The treatment developed above has some advantages over previous formulations. In the past, expressions for the dogleg range error, its scintillation, and the focussing gain have been obtained by heuristic

arguments. The formulas for this paper have been derived more systematically and to the extent that the underlying thin screen assumptions are valid they are correct. If there are several targets in a radar beam it is possible, using the present formulation, to integrate up to the first target and then simply continue on to the next target. In the older formulations it was necessary to redo the entire calculation for each new target.

APPENDIX G

DEFINITIONS OF SYMBOLS

Underlined terms are <u>lists</u> in the sense explained in Vol. 1. An asterisk (*) denotes an input parameter. A double asterisk (**) denotes a precomputed quantity derived from input parameters stored in some dataset.

$^{**}A_{Dn}$ — Dispersion constant used in subroutine REF1S, $A_{Dn} = 0.0828\ B_n f_o^{-3/2}$

A_D — Dispersion parameter defined in REF1S

$^{*}A_R, A_u, A_v$ — Constants relating to the standard deviations of the radar measurement errors; these give the signal-to-noise-independent term in the respective expressions

A_x — Multiplying factor used in POSSV, POSLIST to include possible interfering targets within a specified angle of the desired target

$^{*}B$ — Effective signal bandwidth

$^{**}B_n$ — Noise bandwidth, $B_n \approx 1/(\text{pulse duration})$

$^{*}B_R, B_u, B_v$ — Constants relating to the standard deviations of the radar measurement errors; these are used to find the signal-to-noise-dependent term in the respective expressions

C — Total received clutter power from fireballs and dust clouds

D_s	Slope of the angle sidelobe taper (not currently used)
*f, f_o	Symbols used interchangeably for the radar transmitted center frequency
F, F_1	Symbols used interchangeably for the one-way relative gain due to off-axis position of the target in the radar beam
*F_{pc}	Pulse compression ratio for the given waveform
$\bar{f}_d$	Expected doppler shift, to which the receiver filter is matched
f_d	Actual doppler shift for a given target
FM, FMAX	Symbols used interchangeably for the amplitude factor from which the output signal power can be computed
*H_{REF}	Altitude below which it is assumed that no refraction takes place (used in REF1S)
H_{RAD}	Current altitude of the radar (used in REF1S)
I_E	Integrated electron density (electrons/cm^2) along the path to the target (computed in REFRCT)
IFAIL	Integer index of success or failure of threshold crossing (see RAD1S, etc.)
IFDET	Integer index output by RADMODS; IFDET = 0 indicates that no detection has occurred

ITO	Serial number in POSLST of the target of interest (computed in RAD1S)
L_D	Dispersive loss factor (computed in DISPERS)
L_F	Faraday rotation loss factor (computed in REF1S)
L_2	Two-way path absorption to the target point (computed in ABSORB)
L_{om}	A maximum absorption parameter (computed in RAD1S)
LISTA,LISTB	Lists of interfering targets (see Sec. 4.4.5)
MLST	Number of targets in LISTA or LISTB (computed in REFLST1 or REFLSTN)
MGRP	List of targets in mutually unresolvable groups (computed in REFLST1)
NLST	Number of targets in list POSLST
NGRPS	Number of close-target groups in LISTA (see REFLST1)
$**N_n$	Nominal system noise power, $N_n = kB_nT_n$
N_p	Number of direct propagation paths between radar and target (computed in REFRCT)
NT_o	Identification number of the target of current interest (used in POSLIST and POSSV)
$**_p$	Beamwidth factor (see GOA, etc.), $p = 1.392/\delta w$

POSLST — Possible-targets list; see Sec. 4.2.3

*R_o — Range at which a 1 m^2 target will just exceed threshold

R_c — Range to the center of the receive range gate

R — Range to a given target point

R_1 — Range to the particular target point defined in RAD1S

$\bar{R}$ — Range to the apparent center of the "spread target cloud" in the striated case (see MULTOAR)

R' — Apparent target range (an ultimate output)

R'_o — Apparent target range when used as an input to the monopulse routines MNOPLS and MNOPLSM

RMAX — Range to the highest output peak in SVPEAKS

R_x — Additive range used in POSLIST to include possible interfering targets within 3000 feet of the range gate

**S_o — Parameter used in computing received signal power, $S_o = (S/N)_{Ti} R_o^4 N_n$

**S_1 — Revised parameter used in computing received signal power; after REF1S it has the value

$$S_1 = S_o \frac{\cos^2 \theta}{L_2 L_D L_F}$$

$^{*}S_s$	One-way angular sidelobe (power) level, relative to unity on the beam axis at boresight (the near-in value for tapered sidelobe model)
$^{*}S_{SL}$	Constant range sidelobe (power) level, relative to unity at the peak of the undispersed ambiguity function, for chirped pulses (used in AMBGN)
$^{*}(S/N)_{Ti}$	Pre-detection signal-to-noise threshold for the given event and waveform
s	Parameter used in split-gate range tracking (see SPLTGAT, MLTSPLT, and QINV)
t	Time of the current event
$^{*}T_e$	Effective receiver temperature
T_o	System reference temperature, $T_o = 290°K$
$^{**}T_n$	Nominal system noise temperature, $T_n = T_o + T_e$
T_x	Total effective outside noise temperature
(u,v)	Sine-space coordinates of a given target point
(u_o,v_o)	Sine-space coordinates of the current radar-beam axis
(u_1,v_1)	Sine-space coordinates of the particular target point defined in RAD1S
$(\bar{u},\bar{v})$	Sine-space coordinates of the apparent center of the "spread target cloud" in the striated case (see MULTOAR)

(u',v')	Apparent target sine-space coordinates (an ultimate output)
W	Current value of the width of the receive gate
X_T	Total noise plus clutter power
Y	Trial signal-to-noise ratio which is compared with L_{om} to see if a threshold crossing is possible, in RAD1S and REF1S
Z_o	Signal-to-noise-plus-clutter ratio at the output of the receiver filter (an ultimate output)
α	Safety factor used in RAD1S (see Sec. 4.3.1)
ΔR	Refractive range bias error (see Appendix B)
$(\Delta A,\Delta E)$	Refractive angular bias errors when $N_p = 1$ (see Appendix B)
$(\Delta u,\Delta v)$	Refractive sine-space bias errors to center of "spread target cloud" when $N_p > 1$ (see Appendix B)
ΔX_T	Equivalent noise increment due to range sidelobe target interference
$^{**}\delta R_o$	Nominal range-resolution width for unchirped pulse, $\delta R_o \simeq c\tau/2$
$^{*}\delta R_b$	Radar bias error in the range coordinate

$^{**}\delta R_1$	Nominal range-resolution width for a chirped pulse, $\delta R_1 \simeq 2\delta R_o / F_{pc}$
$\delta R'_o, \delta R'_1$	Dispersion-distorted range resolution widths, $\delta R'_o \simeq L_D \delta R_o$ and $\delta R'_1 \simeq L_D \delta R_1$
$^{*}\delta u_b, \delta v_b$	Radar bias errors in the u,v coordinates
$^{**}\delta w$	One-way, 3-dB half-beamwidth in sine-space, $\delta w = \sin(\delta\theta/2)$
$^{*}\delta\theta$	One-way, 3-dB angular beamwidth at boresight
θ	Angle off boresight of radar beam axis (computed in RAD1S), $\cos\theta = (1 - u_o^2 - v_o^2)^{1/2}$
μ	Slope-angle in sine-space of the "spread target error ellipse," relative to the u-axis (see Appendix B)
ξ	Angle off beam axis in sine-space for a given target, $\xi = [u - u_o)^2 + (v - v_o)^2]^{1/2}$
$\bar{\xi}$	Angle off beam axis in sine-space of the apparent center of the "spread target cloud" in the striated case (see MULTOAR)
$\sigma_{\Delta R}$	Standard deviation of the refractive range bias error
$(\sigma_{\Delta A}, \sigma_{\Delta E})$	Standard deviations of the refractive angular bias errors when $N_p = 1$ (see Appendix B)
(σ_1, σ_2)	Semi-major and semi-minor axes, respectively, of the sine-space error ellipse in the "spread target" case (see Appendix B)

$\bar{\sigma}$ Mean cross section of the individual targets in the "spread target cloud" for the striated case

σ_σ Standard deviation of the distribution (about $\bar{\sigma}$) of the individual "spread target" cross sections

χ_N Waveform ambiguity function normalized to unity at its peak (see AMBGN). For chirped pulses, its argument already contains an adjustment due to range-doppler coupling (see REFLST1 and MULTOAR)

Ω_F Faraday rotation angle (where applicable; see REF1S)

τ Pulse duration

DISTRIBUTION LIST

DEPARTMENT OF DEFENSE

Director
Command Control Technical Center
ATTN: C312, Ralph Mason

Director
Defense Advanced Rsch Proj Agency
ATTN: Strategic Tech Office

Defense Communication Engineer Center
ATTN: Code R410, James W. McLean

Director
Defense Communications Agency
ATTN: Code 480

Defense Documentation Center
12 cy ATTN: TC

Director
Defense Nuclear Agency
ATTN: DDST
ATTN: STSI, Archives
ATTN: RAAE
3 cy ATTN: STTL, Tech Library

Commander
Field Command, DNA
ATTN: FCPR

Director
Interservice Nuclear Weapons School
ATTN: Document Control

Director
Joint Strat Tgt Planning Staff, JCS
ATTN: JPST, Capt D. Goetz

Chief
Livermore Division, FC, DNA
Lawrence Livermore Laboratory
ATTN: FCPRL

OJCS/J-3
ATTN: J-3 WWMCCS, Mr. Toma

DEPARTMENT OF THE ARMY

Commander/Director
Atmospheric Sciences Laboratory
U.S. Army Electronics Command
ATTN: DRSEL-BL-SY, F. E. Niles

Director
BMD Advanced Tech Ctr
Huntsville Office
ATTN: CRDABH-O, W. Davies
ATTN: ATC-T, Melvin T. Capps

Program Manager
BMD Program Office
ATTN: John Shea
ATTN: Plans Division
ATTN: DACS-BMS, Julian Davidson

DEPARTMENT OF THE ARMY (Continued)

Commander
Harry Diamond Laboratories
ATTN: DRXDO-NP, Francis N. Wimenitz
ATTN: DRXDO-TI, Library

Commander
TRASANA
2 cy ATTN: R. E. Dekinder, Jr.

Director
U.S. Army Ballistic Research Labs
ATTN: Mark D. Kregel
ATTN: Lawrence J. Puckett

Commander
U.S. Army Foreign Science & Tech Ctr
ATTN: P. A. Crowley
ATTN: R. Jones

Commander
U.S. Army Missile Command
ATTN: DRSMI-XS, Chief Scientist

Commander
U.S. Army Nuclear Agency
ATTN: W. J. Berberet, ATCA-NAW

DEPARTMENT OF THE NAVY

Chief of Naval Operations
Navy Department
ATTN: Alexander Brandt
ATTN: Ronald E. Pitkin

Commander
Naval Electronics Laboratory Center
3 cy ATTN: Code 2200, Verne E. Hildebrand

Director
Naval Research Laboratory
ATTN: Code 7701, Jack D. Brown
ATTN: Code 2027, Tech Library
ATTN: Code 7700, Timothy P. Coffey
ATTN: Code 7750, S. Ossakow

Commander
Naval Surface Weapons Center
ATTN: Code WA501, Navy Nuc Prgms Off
ATTN: Code WX21, Tech Library

Director
Strategic Systems Project Office
Navy Department
ATTN: NSP-2722, Marcus Meserole

DEPARTMENT OF THE AIR FORCE

AF Geophysics Laboratory, AFSC
ATTN: OPR, James C. Ulwick
ATTN: OPR, Alva T. Stair
ATTN: OPR, Harold Gardiner

DEPARTMENT OF THE AIR FORCE (Continued)

AF Weapons Laboratory, AFSC
ATTN: DYT, Peter W. Lunn
ATTN: CA, Arthur H. Guenther
ATTN: SUL
ATTN: SAS, John M. Kamm
ATTN: DYT, Capt Mark A. Fry

AFTAC
ATTN: TF, Maj Wiley
ATTN: TN

HQ USAF/RD
ATTN: RDQ

Commander
Rome Air Development Center, AFSC
ATTN: EMTLD Doc Library
ATTN: V. Coyne

SAMSO/MN
ATTN: MNX

SAMSO/SZ
ATTN: SZJ, Maj Lawrence Doan

Commander in Chief
Strategic Air Command
ATTN: XPFS, Maj Brian G. Stephan

ENERGY RESEARCH & DEVELOPMENT ADMINISTRATION

University of California
Lawrence Livermore Laboratory
ATTN: Ralph S. Hager, L-31
ATTN: Donald R. Dunn, L-156

Los Alamos Scientific Laboratory
ATTN: Doc Con for John Zinn
ATTN: Doc Con for Eric Jones

Sandia Laboratories
ATTN: Doc Con for J. C. Eckhardt, Org 1250
ATTN: Doc Con for 3141, Sandia Rpt Coll
ATTN: Doc Con for W. D. Brown
ATTN: Doc Con for Charles Williams

OTHER GOVERNMENT AGENCIES

Department of Commerce
Office of Telecommunications
Institute for Telecom Science
ATTN: William F. Utlaut

DEPARTMENT OF DEFENSE CONTRACTORS

Aerospace Corporation
ATTN: Norman D. Stockwell

Brown Engineering Company, Inc.
ATTN: Joel D. Bigley

Calspan Corporation
ATTN: Romeo A. Deliberis

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

ESL, Inc.
ATTN: R. K. Stevens
ATTN: James Marshall
ATTN: V. L. Mower

General Electric Company
TEMPO-Center for Advanced Studies
ATTN: Tim Stephens
ATTN: Warren S. Knapp
ATTN: DASIAC

General Research Corporation
ATTN: John Ise, Jr.
ATTN: John Boys
ATTN: Joe Gabaring

Johns Hopkins University
Applied Physics Laboratory
ATTN: Document Librarian

Lockheed Missiles & Space Co., Inc.
ATTN: D. R. Churchill

M. I. T. Lincoln Laboratory
ATTN: Lib A-082 for David M. Towle

Martin Marietta Aerospace
Orlando Division
ATTN: Roy W. Heffner

Mission Research Corporation
ATTN: Dave Sowle
ATTN: W. F. Crevier
ATTN: Paul Fisher
ATTN: Roy Hendrick
ATTN: Russell Christian

Physical Dynamics, Inc.
ATTN: Joseph B. Workman

Science Applications, Inc.
ATTN: D. Sachs
ATTN: Raymond C. Lee
ATTN: Curtis A. Smith

Science Applications, Inc.
Huntsville Division
ATTN: Noel R. Byrn
ATTN: Dale Divis

Stanford Research Institute
ATTN: Jacqueline Owen
ATTN: Walter G. Chesnut

Visidyne, Inc.
ATTN: J. W. Carpenter
ATTN: Henry J. Smith
ATTN: Chuck H. Humphrey