UNCLASSIFIED

AD NUMBER
AD902627
LIMITATION CHANGES
TO: **Approved for public release; distribution is unlimited.**
FROM: **Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; JAN 1972. Other requests shall be referred to Air Force Armament Lab., Eglin AFB, FL 32542.**
AUTHORITY
AFATL ltr 18 Nov 1975

THIS PAGE IS UNCLASSIFIED

AFATL-TR-72-16
VOLUME II

Vol I 902532L

BURST HEIGHT DISTRIBUTION COMPUTER MODEL

VOLUME II. ANALYST MANUAL

BOOZ, ALLEN APPLIED RESEARCH, INC.

TECHNICAL REPORT AFATL-TR-72-16, VOLUME II

JANUARY 1972

AIR FORCE ARMAMENT LABORATORY
AIR FORCE SYSTEMS COMMAND • UNITED STATES AIR FORCE
EGLIN AIR FORCE BASE, FLORIDA

(6) **Burst Height Distribution**

Computer Model.

Volume II. Analyst Manual.

(9) Final rept. 18 Jan 71 - 17 Jan 72,

(10) **Donald E. Cudney**

David O. Fraser

(11) Jan 72

(12) 91 p.

(15) F08635-71-C-0093

(16) AF-9134

(17) 913401

(18) AFATL

(19) TR-72-16-Vol-2

Distribution limited to U. S. Government agencies only; this report documents test and evaluation; distribution limitation applied January 1972. Other requests for this document must be referred to the Air Force Armament Laboratory (DLYW), Eglin Air Force Base, Florida 32542.

405 237

1473

FOREWORD

This report documents work accomplished during the period 18 January 1971 through 17 January 1972 by Booz, Allen Applied Research, Inc., P.O. Box 1797, Eglin Air Force Base, Florida, under Contract F08635-71-C-0093 with the U.S. Air Force Armament Laboratory, Eglin Air Force Base, Florida. The program monitors for the Armament Laboratory were Mr. Dan McInnis (DLYW) and Mr. Jerry Bass (DLYW).

The report consists of two volumes. Volume I, the User Manual, contains a detailed description of the input variables, instructions for placing the input variables on punch cards, instructions for arranging the punch cards in proper order, descriptions and definitions of the program output, a description of a sample case which can be used to verify that the program is operating properly, and a checklist of input data required to make up a test case. Volume II, the Analyst Manual, contains the mathematical relationships which were used to develop the model, the assumptions employed in constructing the model, flowcharts depicting the logical structure of the model, a listing of the source deck, and a detailed description of the simulation model based on the coding language employed and including comments identifying algebraic expressions which appear in the mathematical model.

The authors wish to acknowledge the assistance provided by several individuals in developing the methodology and computer program described in this report. Mr. Gary M. Grann of the U.S. Air Force Armament Laboratory was instrumental in developing the initial methodology and computer program. Also, the suggestions and comments provided by the members of the Methodology and Evaluation Working Group of the Degradation Effects Program were extremely helpful in developing the computer program.

This technical report has been reviewed and is approved.

Thomas P. Christie
THOMAS P. CHRISTIE
Chief, Weapon Systems Analysis Division

ABSTRACT

The Burst Height Distribution (BHD) Program described in this report was designed to compute and display burst height distributions for munitions aerially delivered into forest environments. The program uses as input the source and terminal X, Y, and Z coordinates and the average diameters of branches surveyed at actual forested sites, and the munitions travel along straight-line trajectories which are randomly selected. Burst heights are computed for those trajectories which encounter branches large enough to detonate the munition, and after 400 trajectories are examined (100 from each of four azimuth angles), the cumulative burst height distribution for the munition and elevation angle is computed, printed, and optionally punched as output. The computer program was specifically designed for the Control Data Corporation 6600 computer system at Eglin Air Force Base, Florida.

(The reverse of this page is blank)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
A	A	The distance between the branch ends along the transformed Z′ axis.	mm
B	B	The distance between the branch ends along the transformed Y′ axis.	mm
C	C	A value used in computing the distance in the transformed Y′-Z′ plane between the munition trajectory and the branch centerline for Cases 2 and 3 [i.e., $C = -Z'_{1j}(B) - Y'_{1j}(A)$].	mm
C_k	CFREQ(150)	The cumulative frequency for the k^{th} interval of the burst height distribution table.	percent
D^2	D	The square of the distance in the transformed Y′-Z′ plane between the munition trajectory and the branch centerline (see Figures 6, 7, and 8).	mm^2
D_d	BBDIAM	The minimum branch diameter required for detonation by the munition body.	mm
F_k	FREQ(150)	The number of detonations in the k^{th} burst height interval.	none

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
p		The sequential number of the rectangle along the X axis defining the munition trajectory entry point into the top layer of branches.	none
q		The sequential number of the rectangle along the Y axis defining the munition trajectory entry point into the top layer of branches.	none
R_{bj}	AVRAD	The average radius of the j^{th} branch.	mm
R_f	FRAD	The radius of the munition fuze.	mm
R_m	BRAD	The radius of the munition body.	mm
R_{max}		The greater of the distances in the transformed Y'-Z' plane between the munition trajectory and each end of the branch.	mm
R_{min}		The smaller of the distances in the transformed Y'-Z' plane between the munition trajectory and each end of the branch.	mm
R_N		A random number selected from a uniform distribution with $0.0 < R_N < 1.0$.	none

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
R_{Xi}		A random integer between 1 and 10 selected from a uniform distribution.	none
R_{Yi}		A random integer between 1 and 10 selected from a uniform distribution.	none
R_1	$[R(1)]^{\frac{1}{2}}$	The distance in the transformed Y′-Z′ plane between the munition trajectory and one of the ends of the branch (see Figures 7 and 8).	mm
R_2	$[R(2)]^{\frac{1}{2}}$	The distance in the transformed Y′-Z′ plane between the munition trajectory and one of the ends of the branch (see Figures 7 and 8).	mm
S_1	S1	The distance in the transformed Y′-Z′ plane along the centerline of the branch and between one of the ends of the branch and the point at which the normal from the munition trajectory intersects the branch centerline (see Figure 7).	mm

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
S_2	S2	The distance in the transformed Y′-Z′ plane along the centerline of the branch and between one of the ends of the branch and the point at which the normal from the munition trajectory intersects the branch centerline (see Figure 7).	mm
X'_{di}	XDET	The transformed X′ coordinate of the detonation for the i^{th} munition trajectory.	mm
X_{Ei}	XTRAJ(100)	The X coordinate of the i^{th} munition trajectory entry point into the layer of branches.	mm
X'_{Ei}		The transformed X′ coordinate of the i^{th} munition trajectory entry point into the layer of branches.	mm
X_j	BRANCH(6)	The X coordinate of an end of the j^{th} branch.	mm
X'_j	X1(1500) X2(1500)	The transformed X′ coordinate of an end of the j^{th} branch.	mm
X_{max}	XLIMIT	The X coordinate which defines a boundary of the forest environment.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
X'_{min}		The smaller of the transformed X' coordinates of the branch ends.	mm
X_{Oi}	XOUT	The X coordinate of the i^{th} munit on trajectory exit point from the layer of branches.	mm
X_{ri}		The X coordinate of the i^{th} munition trajectory reentry point into the layer of branches.	mm
X'_{ri}		The transformed X' coordinate of the i^{th} munition trajectory reentry point into the layer of branches.	mm
X_{sj}		The X coordinate of an end of the j^{th} branch after rotation through the azimuth angle.	mm
X_t	XTRAN	The distance in the X direction used to translate the rotated branch coordinates into the first octant (see Table I).	mm
Y'_{di}	YDET	The transformed Y' coordinate of the detonation for the i^{th} munition trajectory.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
Y_{Ei}	YTRAJ(100)	The Y coordinate of the i^{th} munition trajectory entry point into the layer of branches.	mm
Y'_{Ei}	YTRAJP	The transformed Y' coordinate of the i^{th} munition trajectory entry point into the layer of branches.	mm
Y_j	BRANCH(6)	The Y coordinate of an end of the j^{th} branch.	mm
Y'_j	Y1(1500) Y2(1500)	The transformed Y' coordinate of an end of the j^{th} branch.	mm
Y_{max}	YLIMIT	The Y coordinate which defines a boundary of the forest environment.	mm
Y_{Oi}		The Y coordinate of the i^{th} munition trajectory exit point from the layer of branches.	mm
Y_{ri}		The Y coordinate of the i^{th} munition trajectory reentry point into the layer of branches.	mm
Y'_{ri}		The transformed Y' coordinate of the i^{th} munition trajectory reentry point into the layer of branches.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
Y_{sj}		The Y coordinate of an end of the j^{th} branch after rotation through the azimuth angle.	mm
Y_t	YTRAN	The distance in the Y direction used to translate the rotated branch coordinates into the first octant (see Table I).	mm
Y'_{Ti}	YTRAJP	The transformed Y' coordinate of the i^{th} munition trajectory.	mm
Y'_{1j}	Y1(1500)	The transformed Y' coordinate of the source end of the j^{th} branch.	mm
Y'_{2j}	Y2(1500)	The transformed Y' coordinate of the terminal end of the j^{th} branch.	mm
Z_b	BOTTOM	The Z coordinate of the bottom of the layer of branches.	mm
Z'_{di}	ZDET	The transformed Z' coordinate of the detonation for the i^{th} munition trajectory.	mm
Z_{Ei}	ZTRAJ(100)	The Z coordinate of the i^{th} munition trajectory entry point into the layer of branches.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
Z'_{Ei}	ZTRAJP	The transformed Z' coordinate of the i^{th} munition trajectory entry point into the layer of branches.	mm
Z_{hi}	ZHGT	The burst height for the i^{th} munition trajectory.	feet
Z_j	BRANCH(6)	The Z coordinate of an end of the j^{th} branch.	mm
Z'_j	Z1(1500) Z2(1500)	The transformed Z' coordinate of an end of the j^{th} branch.	mm
Z_{max}	ZLIMIT	The Z coordinate which defines a boundary of the forest environment.	mm
Z_{Oi}	ZOUT	The Z coordinate of the i^{th} munition trajectory exit point from the layer of branches.	mm
Z_{ri}		The Z coordinate of the i^{th} munition trajectory reentry point into the layer of branches.	mm
Z'_{ri}		The transformed Z' coordinate of the i^{th} munition trajectory reentry point into the layer of branches.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - MATHEMATICAL MODEL (CONCLUDED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN SIMULATION MODEL	DEFINITION	UNITS
Z_{sj}		The Z coordinate of an end of the j^{th} branch after rotation through the azimuth angle.	mm
Z'_{Ti}	ZTRAJP	The transformed Z' coordinate of the i^{th} munition trajectory.	mm
Z'_{1j}	Z1(1500)	The transformed Z' coordinate of the source end of the j^{th} branch.	mm
Z'_{2j}	Z2(1500)	The transformed Z' coordinate of the terminal end of the j^{th} branch.	mm
θ	THETA	The azimuth angle for the munition trajectory paths.	degrees
φ	PHI	The elevation angle for the munition trajectory paths.	degrees

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
A	A	The distance between the branch ends along the transformed Z′ axis.	mm
AVDIAM(1500)		An array containing the average diameters of the branches.	mm
AVRAD	R_{bj}	The average radius of the branch.	mm
AZIMUTH	θ	The azimuth angle for the munition trajectory paths.	radians
B	B	The distance between the branch ends along the transformed Y′ axis.	mm
BBDIAM	D_d	The minimum branch diameter required for detonation by the munition body.	mm
BDIAM	$2R_m$	The diameter of the munition body.	mm
BFDIAM		The minimum branch diameter required for detonation by the munition fuze.	mm
BMT		The difference in height (or the difference in Z coordinates) between the top and the bottom of the layer of branches.	mm
BOTTOM	Z_b	The Z coordinate of the bottom of the layer of branches.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
BRAD	R_m	The radius of the munition body.	mm
BRANCH(6)	X_j Y_j Z_j	An array containing the X, Y, and Z coordinates of the branch ends.	mm
BRDIAM		The average diameter of the branch which caused the detonation.	mm
C	C	A value used in computing the distance in the transformed Y'-Z' plane between the munition trajectory and the branch centerline for Cases 2 and 3.	mm
CFREQ(150)	C_k	The cumulative frequency for each interval of the burst height distribution table.	percent
COSP	$\cos\varphi$	The cosine of the elevation angle.	none
COST	$\cos\theta$	The cosine of the azimuth angle.	none
CTCP	$\cos\theta\cos\varphi$	The cosine of the azimuth angle multiplied by the cosine of the elevation angle.	none

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
CTSP	$\cos\theta\sin\varphi$	The cosine of the azimuth angle multiplied by the sine of the elevation angle.	none
D	D^2	The square of the distance in the transformed Y'-Z' plane between the munition trajectory and the branch centerline.	mm^2
DET(100)		An array containing the burst heights for 100 munition trajectories.	feet
DIAM	$2R_{bj}$	The average diameter of the branch.	mm
DUMMY		The random number generator starting value.	none
D1		An intermediate value used in establishing the burst height intervals.	none
FACTRB	$(R_{bj} + R_m)^2$	The square of the sum of the average radius of the branch and the radius of the munition body.	mm^2
FACTRF	$(R_{bj} + R_f)^2$	The square of the sum of the average radius of the branch and the radius of the munition fuze.	mm^2
FDIAM	$2R_f$	The diameter of the munition fuze.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
FRAD	R_f	The radius of the munition fuze.	mm
FREQ(150)	F_k	An array containing the number of munition detonations for each burst height interval.	none
I		An internal program variable.	none
INC	k	The sequential burst height interval (i.e., 1 identifies ground bursts, 2 identifies the first interval above the ground, etc.).	none
IPRINT		The print option: 0 = print the burst height and branch diameter for each munition detonation and the burst height distribution table. 1 = print the burst height distribution table.	none
IPUNCH		The punch option: 0 = do not punch cards. 1 = punch cards for the burst height intervals, the number of detonations in each interval, and the cumulative frequency for each interval.	none
ITHETA	$\theta/90 + 1$	An integer number which is a function of the azimuth angle.	none

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
IX		The number of munition detonations for the burst height interval.	none
J		An internal program variable.	none
K		An internal program variable.	none
KEEP		A counter for the number of detonations occurring along each munition trajectory in the layer of branches.	none
KEOF		The argument to System Subroutine EOF which exercises control over the program when an end-of-file is read.	none
KEPDET(300)		An array containing the burst heights (or Z coordinates) for detonations occurring along each munition trajectory in the layer of branches.	feet
KEPDIAM(300)		An array containing the branch diameters for detonations occurring along each munition trajectory in the layer of branches.	mm
KTGRD		The number of ground bursts.	none
KTR		The number of air bursts.	none
L		An internal program variable.	none

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
LENGTH	$(A^2 + B^2)^{\frac{1}{2}}$	The length of the branch in the transformed Y′-Z′ plane.	mm
LENSQR	$A^2 + B^2$	The square of the length of the branch in the transformed Y′-Z′ plane.	mm^2
LEVEL		A sequential number which identifies the layers of branch data (i. e., 1 identifies data for the top layer, 2 identifies data for the next layer from the top, etc.).	none
LL		An internal program variable.	none
MIN		A subscript used to identify the branch end which is closest to the munition trajectory [see R(MIN)].	none
NBRAN		The number of branches described in the layer on the branch input data tape.	none
NMUN		The munition identification number.	none
NRNX		An intermediate value used to determine uniform random integers between 1 and 10.	none
NRNXY		An intermediate value used to determine uniform random integers between 1 and 10.	none

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
PHI	φ	The elevation angle for the munition trajectory paths.	degrees
PHISAV		A save location for the elevation angle.	degrees
PHISAVE		A save location for the elevation angle.	degrees
PX		The X′ coordinate of the detonation measured from the ground to the nose of the munition.	mm
R(2)	R^2_1 R^2_2	An array containing the squares of the distances in the transformed Y′-Z′ plane between the munition trajectory and each end of the branch.	mm^2
REPEAT		A flag which when set to 1 indicates that the munition trajectory must be reentered into the layer of branches.	none
R(MIN)	R^2_{min}	The square of the smallest of the distances in the transformed Y′-Z′ plane between the munition trajectory and each end of the branch.	mm^2
RNX	R_{Xi}	A random integer between 1 and 10 selected from a uniform distribution.	none

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
RNY	R_{Y_i}	A random integer between 1 and 10 selected from a uniform distribution.	none
SINP	$\sin\varphi$	The sine of the elevation angle.	none
SINT	$\sin\theta$	The sine of the azimuth angle.	none
SITE		The identifying notation for the forest environment.	none
SPOCP	$\tan\varphi$	The tangent of the elevation angle.	none
SSQ	$R^2_{max} - D^2$	The difference between the square of the largest of the distances in the transformed Y'-Z' plane between the munition trajectory and each end of the branch and the square of the distance in the transformed Y'-Z' plane between the munition trajectory and the branch centerline.	mm^2
SS2	S^2_2	The square of the distance in the transformed Y'-Z' plane along the centerline of the branch and between one of the ends of the branch and the point at which the normal from the munition trajectory intersects the branch centerline.	mm^2

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
STAT(8)		An array containing statistical values returned by System Subroutine BDS.	various
STCP	$\sin\theta\cos\varphi$	The sine of the azimuth angle multiplied by the cosine of the elevation angle.	none
STSP	$\sin\theta\sin\varphi$	The sine of the azimuth angle multiplied by the sine of the elevation angle.	none
S1	S_1	The distance in the transformed Y′-Z′ plane along the centerline of the branch and between one of the ends of the branch and the point at which the normal from the munition trajectory intersects the branch centerline.	mm
S2	S_2	The distance in the transformed Y′-Z′ plane along the centerline of the branch and between one of the ends of the branch and the point at which the normal from the munition trajectory intersects the branch centerline.	mm
THEDET		An intermediate location for storing the burst height of the munition trajectory.	feet
THEDIAM		The branch diameter which caused the munition detonation.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
THETA	θ	The azimuth angle for the munition trajectory paths.	degrees
TOP		The Z coordinate of the top of the layer of branches.	mm
TRASH		A dummy argument for the system uniform random number generator.	none
XCELL		The minimun X coordinate of a rectangle which subdivides the top of the forest environment.	mm
XDATA		The upper boundary of the burst height interval.	feet
XDET	X'_{di}	The transformed X′ coordinate of the detonation for the munition trajectory.	mm
XINT		The burst height interval.	feet
XLIMIT	X_{max}	The X coordinate which defines a boundary of the forest environment.	mm
XLIM0		The X coordinate which defines a boundary of the forest environment divided by 10 (i.e., XLIMIT/10).	mm
XLIM00		The X coordinate which defines a boundary of the forest environment divided by 100 (i.e., XLIMIT/100).	mm

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
XNUM		The total number of munition trajectories evaluated for each elevation angle (always 400).	none
XOUT	X_{Oi}	The X coordinate of the munition trajectory exit point from the layer of branches.	mm
XPLUS		The distance traveled by the munition along the X axis from the entry point into the top of the layer to the exit point through the plane defining the bottom of the layer.	mm
XTRAJ(100)	X_{Ei}	An array containing the X coordinates of the munition trajectory entry points into the layer.	mm
XTRAN	X_t	The distance in the X direction used to translate the rotated branch coordinates into the first octant (see Table I).	mm
XTRANC	$X_t \cos\varphi$	The distance in the X direction used to translate the rotated branch coordinates into the first octant multiplied by the cosine of the elevation angle.	mm

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONTINUED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
XTRANS	$X_t \sin\varphi$	The distance in the X direction used to translate the rotated branch coordinates into the first octant multiplied by the sine of the elevation angle.	mm
XX		The lower boundary of the burst height interval.	feet
X1(1500)	X'_{1j}	An array containing the transformed X' coordinates for the source ends of the branches.	mm
X2(1500)	X'_{2j}	An array containing the transformed X' coordinates for the terminal ends of the branches.	mm
YCELL		The minimum Y coordinate of a rectangle which subdivides the top of the forest environment.	mm
YDET	Y'_{di}	The transformed Y' coordinate of the detonation for the munition trajectory.	mm
YLIMIT	Y_{max}	The Y coordinate which defines a boundary of the forest environment.	mm
YLIM0		The Y coordinate which defines a boundary of the forest environment divided by 10 (i.e., YLIMIT/10).	mm

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
YLIM00		The Y coordinate which defines a boundary of the forest environment divided by 100 (i.e., YLIMIT/100).	mm
YTRAJ(100)	Y_{Ei}	An array containing the Y coordinates of the munition trajectory entry points into the layer of branches.	mm
YTRAJP	Y'_{Ei}	The transformed Y′ coordinate of the munition trajectory entry point into the layer of branches.	mm
YTRAN	Y_t	The distance in the Y direction used to translate the rotated branch coordinates into the first octant (see Table I).	mm
Y1(1500)	Y'_{1j}	An array containing the transformed Y′ coordinates of the source ends of the branches.	mm
Y2(1500)	Y'_{2j}	An array containing the transformed Y′ coordinates of the terminal ends of the branches.	mm
ZDET	Z'_{di}	The transformed Z′ coordinate of the detonation for the munition trajectory.	mm
ZHGT	Z_{hi}	The burst height for the munition trajectory.	feet

LIST OF SYMBOLS AND ABBREVIATIONS - SIMULATION MODEL (CONCLUDED)

SYMBOL OR ABBREVIATION	EQUIVALENT IN MATH MODEL	DEFINITION	UNITS
ZLIMIT	Z_{max}	The Z coordinate which defines a boundary of the forest environment.	mm
ZOUT	Z_{Oi}	The Z coordinate of the munition trajectory exit point from the layer of branches.	mm
ZTRAJ(100)	Z_{Ei}	An array containing the Z coordinates of the munition trajectory entry points into the layer of branches.	mm
ZTRAJP	Z'_{Ei}	The transformed Z′ coordinate of the munition trajectory entry point into the layer of branches.	mm
Z1(1500)	Z'_{1j}	An array containing the transformed Z′ coordinates of the source ends of the branches.	mm
Z2(1500)	Z'_{2j}	An array containing the tranformed Z′ coordinates of the terminal ends of the branches.	mm

SECTION I

INTRODUCTION

The Burst Height Distribution (BHD) Program was designed to compute and display burst height distributions for munitions aerially delivered into forest environments. The computer program was written in FORTRAN IV and is presently operational on the Control Data Corporation 6600 computer system at Eglin Air Force Base, Florida.

The program uses as input the source and terminal X, Y, and Z coordinates and the average diameters of branches surveyed at actual forested sites. Because of the large number of branches included in the characterization of the forest environments, all of the branch coordinates and diameters cannot be read into the computer central processor at one time. An input tape must be prepared with the branch data segmented into layers. The branch data for the top layer are then read into the computer central processor, and the munition trajectories are examined for possible encounters with the branches. For those munition trajectories which pass through the top layer and do not result in a munition detonation, the coordinates of the exit point from the bottom or side of the layer are determined. If the munition trajectory exits the side of the layer, the trajectory is reentered into the layer at the same height. If the trajectory exits the bottom of the layer, the exit coordinates from the top layer are used as entry coordinates for the second layer. This process is repeated until all layers have been examined or until the munition detonates. The procedure permits determination of the munition/branch encounters for all of the munition trajectories by inputting the branch data into the central processor only one time.

The munitions travel along straight-line trajectories which are randomly selected. If an encounter between a branch and the munition body occurs, the program determines if the branch is large enough to cause the munition to detonate. If the branch is large enough, the burst height is calculated. If the branch is too small, the program next determines if the munition fuze encounters the branch. If the fuze encounters the branch, the burst height is calculated, and if the fuze does not encounter the branch, the trajectory is continued. After 100 trajectories are examined from each of four azimuth angles (i.e., 0, 90, 180, and 270 degrees) and from the elevation angle specified as input, the cumulative burst height distribution for the munition is computed, printed, and optionally punched as output.

This volume contains:

- A detailed description of the mathematical model which was used as a basis in developing the program.
- A complete set of flowcharts depicting the logic used in the program.
- A complete FORTRAN IV source listing of the program.
- A detailed description of the simulation coding which was employed in developing the program.
- Listings of the symbols and abbreviations which are used in the mathematical model and the computer program.

Detailed descriptions of the input variables required to properly execute the program, instructions for placing the input variables on punch cards and for arranging the punch cards in proper order, descriptions and definitions of the output available from the program, and a description of a sample case are contained in Volume I, the User Manual, of this report.

SECTION II

MATHEMATICAL MODEL

The mathematical model is divided into five principal parts. The first of these determines the entry coordinates for each of the m .nition trajectories that are to be examined as they pass through the top of the forest environment. The second part translates and rotates the coordinates of the branches so that they are in the first octant and so that the new X′ axis is parallel to the path of the munition trajectory. The third part determines subsequent munition trajectory exit and reentry points as they pass through the layers of the forest environment. The fourth part calculates the distances between each of the munition trajectories and branch centerlines and determines the munition/branch encounters with the maximum height that will detonate a munition traveling along each of the trajectory paths. The fifth part prepares cumulative burst height distribution tables for the 400 munition trajectories delivered from the four azimuth angles (i.e., 100 each from 0, 90, 180, and 270 degrees) and from a specified elevation angle.

DETERMINATION OF INITIAL TRAJECTORY ENTRY COORDINATES INTO TOP LAYER

The source and terminal X, Y, and Z coordinates of the branches in the forest environment are originally specified in the first octant of a cartesian coordinate system, and the limits of the environments are defined by X_{max}, Y_{max}, and Z_{max}, as shown in Figure 1. The branch coordinates and average diameters are contained on magnetic tape, and the tape is subdivided into layers with data for up to 1,500 branches contained in each layer. The entry coordinates for each of the munition trajectories which are to be examined from a given azimuth angle and elevation angle are determined by subdividing the X-Y plane of the forest environment into 100 rectangles and randomly selecting entry coordinates which will pass through each of the rectangles. The expressions used in the model are:

$$X_{Ei} = \frac{(p-1)(X_{max})}{10} + \frac{R_{Xi}(X_{max})}{100} \qquad (1)$$

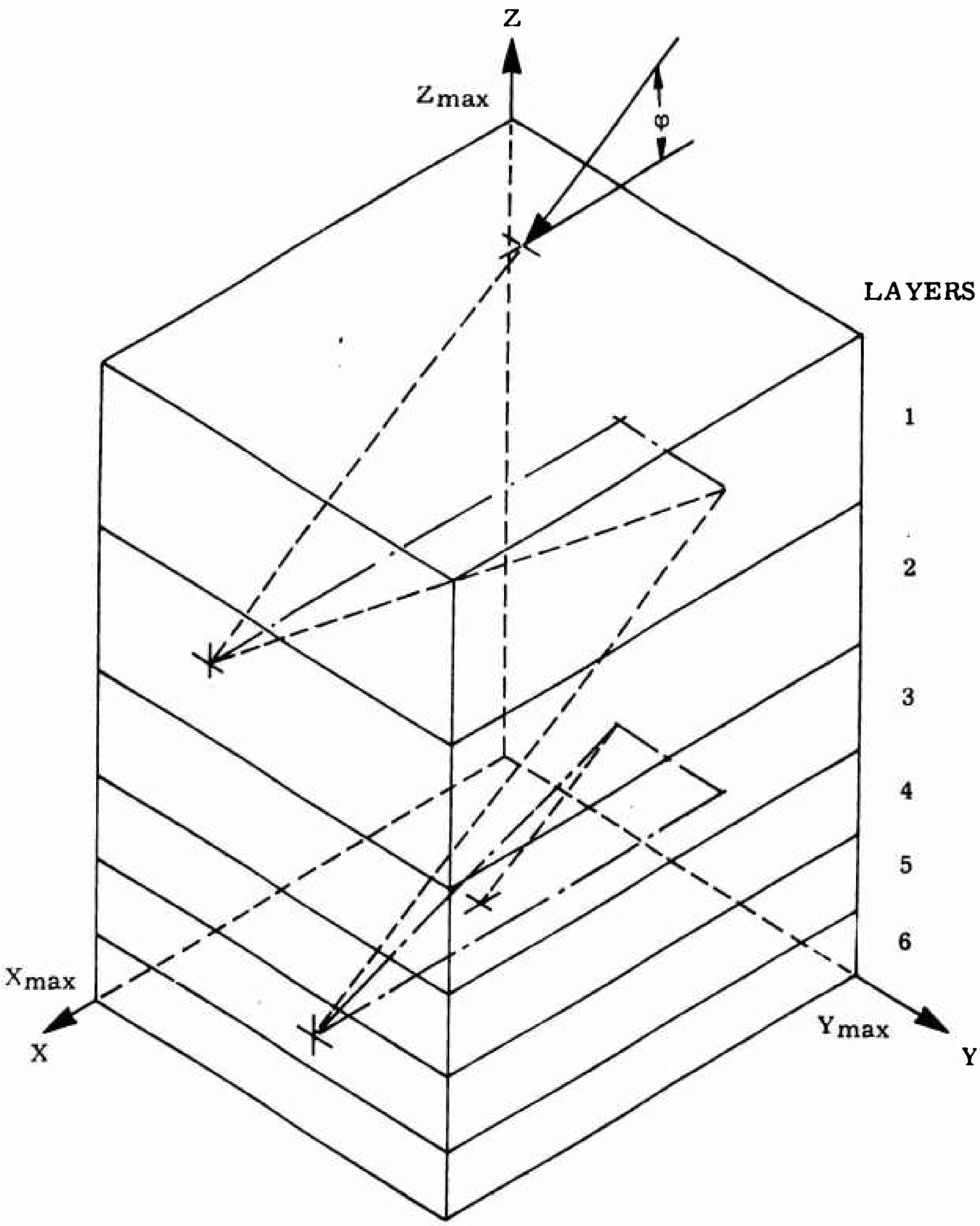

Figure 1. Pictorial Representation of Layers Used in Forest Environment and Trajectory Paths

$$Y_{Ei} = \frac{(q-1)(Y_{max})}{10} + \frac{R_{Yi}(Y_{max})}{100} \quad (2)$$

$$Z_{Ei} = Z_{max} \quad (3)$$

where p is the sequential number of the rectangle along the X axis, q is the sequential number of the rectangle along the Y axis, X_{Ei}, Y_{Ei}, and Z_{Ei} are the X, Y, and Z coordinates of the i^{th} munition trajectory entry point, X_{max}, Y_{max}, and Z_{max} are the X, Y, and Z boundaries of the forest environment (see Figure 1), and R_{Xi} and R_{Yi} are random integers between 1 and 10 selected from a uniform distribution.

TRANSLATION AND ROTATION OF BRANCH COORDINATES

A time saving feature of the BHD Program involves a transformation of the original coordinate axes to permit determination of the munition/branch encounters in two rather than three dimensions. The transformation is made so that the new X' axis is parallel to the path of the munition trajectory, and all subsequent munition/branch encounter calculations are performed using branch coordinates projected onto the new Y'-Z' plane. The equations for the first rotation about the Z axis through the azimuth angle θ and for translation of the coordinates into the first octant are (see Figure 2):

$$X_{sj} = X_j\cos\theta + Y_j\sin\theta + X_t$$

$$Y_{sj} = -X_j\sin\theta + Y_j\cos\theta + Y_t$$

$$Z_{sj} = Z_j$$

where θ is as defined above, X_j, Y_j, and Z_j are the coordinates of the j^{th} point relative to the original X, Y, and Z axes, X_{sj}, Y_{sj}, and Z_{sj} are the new translated and rotated coordinates, and X_t and Y_t are the X and Y values used to translate the new coordinates into the first octant. In matrix form, this first transformation is given by:

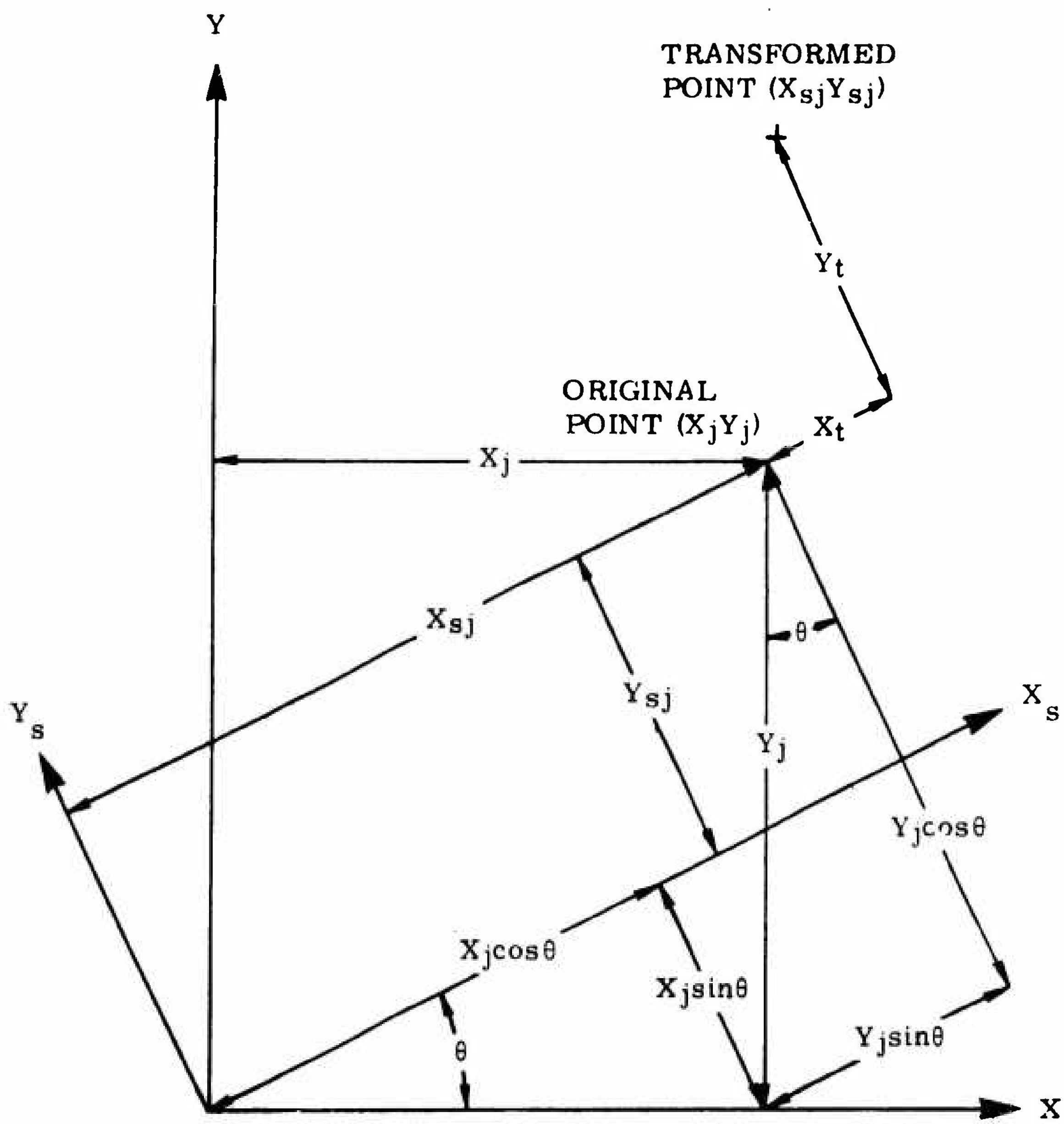

Figure 2. Relationship of Original and Transformed Branch Coordinates After First Rotation in X-Y Plane Through Azimuth Angle θ

$$\begin{bmatrix} X_{sj} \\ Y_{sj} \\ Z_{sj} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_j \\ Y_j \\ Z_j \end{bmatrix} + \begin{bmatrix} X_t \\ Y_t \\ 0 \end{bmatrix}$$

The second rotation is about the newly established Y_s axis through the elevation angle φ, which establishes the X', Y', and Z' coordinate axes. The equations for this second rotation are (see Figure 3):

$$X'_j = X_{sj}\cos\varphi + Z_{sj}\sin\varphi$$

$$Y'_j = Y_{sj}$$

$$Z'_j = -X_{sj}\sin\varphi + Z_{sj}\cos\varphi$$

In matrix form, this second transformation is given by:

$$\begin{bmatrix} X'_j \\ Y'_j \\ Z'_j \end{bmatrix} = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix} \begin{bmatrix} X_{sj} \\ Y_{sj} \\ Z_{sj} \end{bmatrix}$$

The composite equations encompassing both rotations and the translation are expressed in matrix form as:

$$\begin{bmatrix} X'_j \\ Y'_j \\ Z'_j \end{bmatrix} = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_j \\ Y_j \\ Z_j \end{bmatrix}$$

$$+ \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix} \begin{bmatrix} X_t \\ Y_t \\ 0 \end{bmatrix}$$

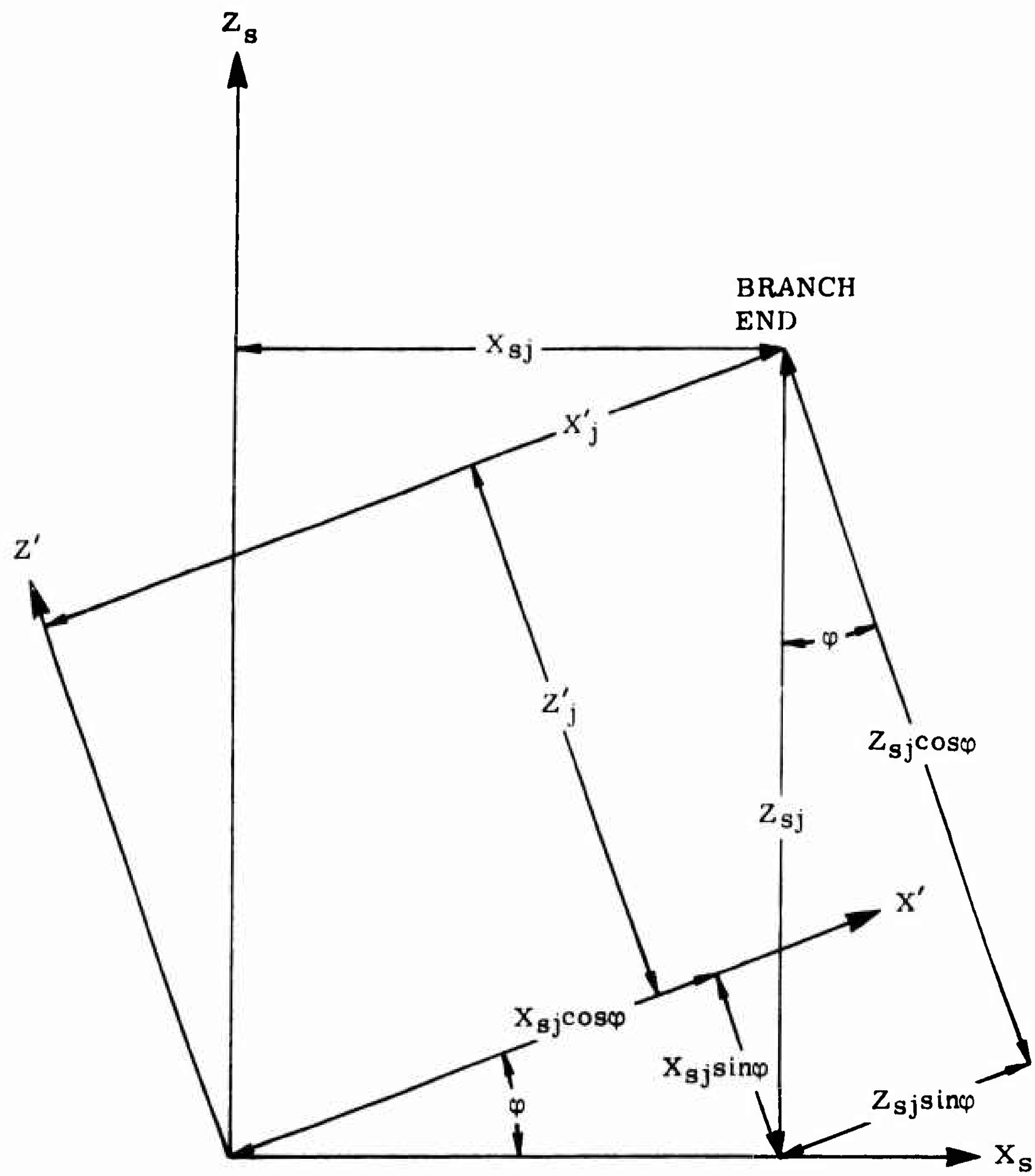

Figure 3. Relationship of Transformed X', Y', and Z' Coordinates to X_{sj}, Y_{sj}, and Z_{sj} Coordinates After Rotation of Axes in X_s-Z_s Plane Through Elevation Angle φ

Multiplying the two square matrices results in the following:

$$\begin{bmatrix} X'_j \\ Y'_j \\ Z'_j \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\varphi & \sin\theta\cos\varphi & \sin\varphi \\ -\sin\theta & \cos\theta & 0 \\ -\cos\theta\sin\varphi & -\sin\theta\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} X_j \\ Y_j \\ Z_j \end{bmatrix} + \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix} \begin{bmatrix} X_t \\ Y_t \\ 0 \end{bmatrix}$$

Thus, the equations expressing the newly established X′, Y′, and Z′ coordinates as functions of the original X, Y, and Z coordinates, the azimuth angle θ, and the elevation angle φ are:

$$X'_j = X_j\cos\theta\cos\varphi + Y_j\sin\theta\cos\varphi + Z_j\sin\varphi + X_t\cos\varphi \quad (4)$$

$$Y'_j = -X_j\sin\theta + Y_j\cos\theta + Y_t \quad (5)$$

$$Z'_j = -X_j\cos\theta\sin\varphi - Y_j\sin\theta\sin\varphi + Z_j\cos\varphi - X_t\sin\varphi \quad (6)$$

where all variables are as defined above. The values of X_t and Y_t used in the program vary with the azimuth angle and are presented in Table I.

TABLE I. TRANSLATION VALUES X_t AND Y_t USED IN BHD PROGRAM

AZIMUTH ANGLE (degrees)	X_t	Y_t
0	0	0
90	0	Y_{max}
180	X_{max}	Y_{max}
270	X_{max}	0

DETERMINATION OF SUBSEQUENT TRAJECTORY EXIT AND REENTRY COORDINATES

The X, Y, and Z coordinates for each of the munition trajectories as they exit the layers of branches through the bottom or side are calculated using the expressions (see Figure 4):

For Munitions Exiting the Layer of Branches at the Bottom:	For Munitions Exiting the Layer of Branches at the Side:	
$X_{Oi} = X_{Ei} + \frac{(Z_{Ei} - Z_b)}{\tan\varphi}$	$X_{Oi} = X_{max}$	(7)
$Y_{Oi} = Y_{Ei}$	$Y_{Oi} = Y_{Ei}$	(8)
$Z_{Oi} = Z_b$	$Z_{Oi} = Z_{Ei} - (X_{max} - X_{Ei})\tan\varphi$	(9)

where X_{max}, X_{Ei}, Y_{Ei}, and Z_{Ei} are as defined above, X_{Oi}, Y_{Oi}, and Z_{Oi} are the coordinates of the munition trajectory as it exits the layer, Z_b is the Z coordinate of the bottom of the layer, and φ is the elevation angle.

If the munition trajectory exits the layer through the side rather than through the bottom, it must be reentered into the layer. The reentry process is repeated until the trajectory exits the bottom of the layer, and the reentry coordinates are determined using the expressions:

$$X_{ri} = 0.0 \quad (10)$$

$$Y_{ri} = R_N(Y_{max}) \quad (11)$$

$$Z_{ri} = Z_{Oi} \quad (12)$$

where Y_{max} and Z_{Oi} are as defined above, X_{ri}, Y_{ri}, and Z_{ri} are the reentry coordinates for the i^{th} munition trajectory, and R_N is a random number selected from a uniform distribution with $0.0 < R_N < 1.0$.

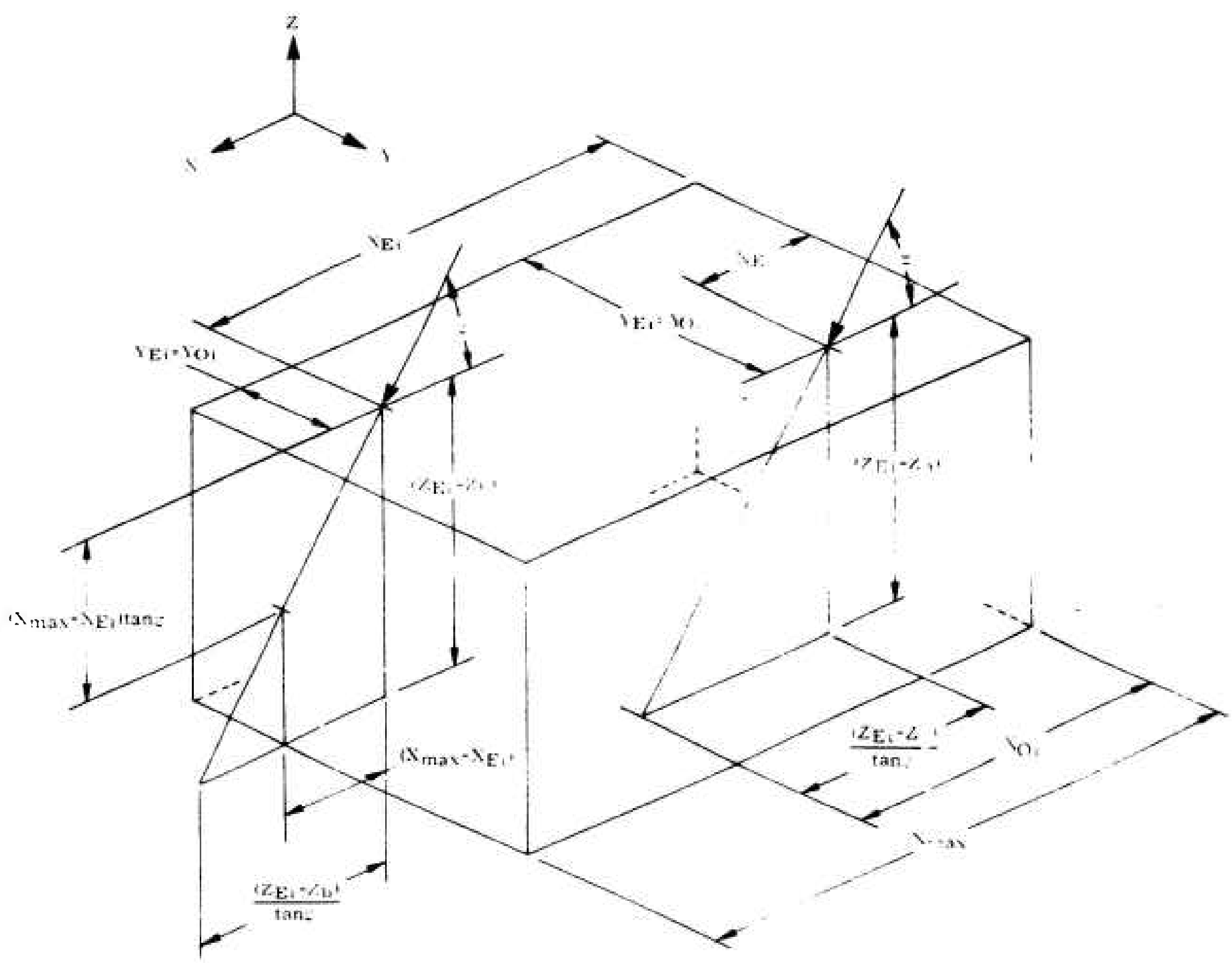

Figure 4. Relationship of Entry and Exit Coordinates as Trajectories Pass Through Layers of Branches

When the munition trajectory exits the bottom of the layer, the reentry coordinates are determined using the expressions:

$$X_{ri} = X_{Oi} \tag{13}$$

$$Y_{ri} = Y_{Ei} \tag{14}$$

$$Z_{ri} = Z_{Oi} \tag{15}$$

where all of the variables are as defined above.

For the top layer of branches, the entry point coordinates of each munition trajectory are rotated through the elevation angle φ and around the Y axis. The expressions used to perform this rotation are (see Fig- [illegible]):

$$X'_{Ei} = X_{Ei}\cos\varphi + Z_{Ei}\sin\varphi \tag{16}$$

$$Y'_{Ei} = Y_{Ei} \tag{17}$$

$$Z'_{Ei} = -X_{Ei}\sin\varphi + Z_{Ei}\cos\varphi \tag{18}$$

where X'_{Ei}, Y'_{Ei}, and Z'_{Ei} are the entry coordinates for the i^{th} munition trajectory after rotation, and all other variables are as defined above. For each succeeding layer of branches and for any munition trajectory which must reenter a layer, the reentry coordinates are rotated using the expressions:

$$X'_{ri} = X_{ri}\cos\varphi + Z_{ri}\sin\varphi \tag{19}$$

$$Y'_{ri} = Y_{ri} \tag{20}$$

$$Z'_{ri} = -X_{ri}\sin\varphi + Z_{ri}\cos\varphi \tag{21}$$

where X'_{ri}, Y'_{ri}, and Z'_{ri} are the reentry coordinates for the i^{th} munition trajectory after rotation, and all other variables are as defined above.

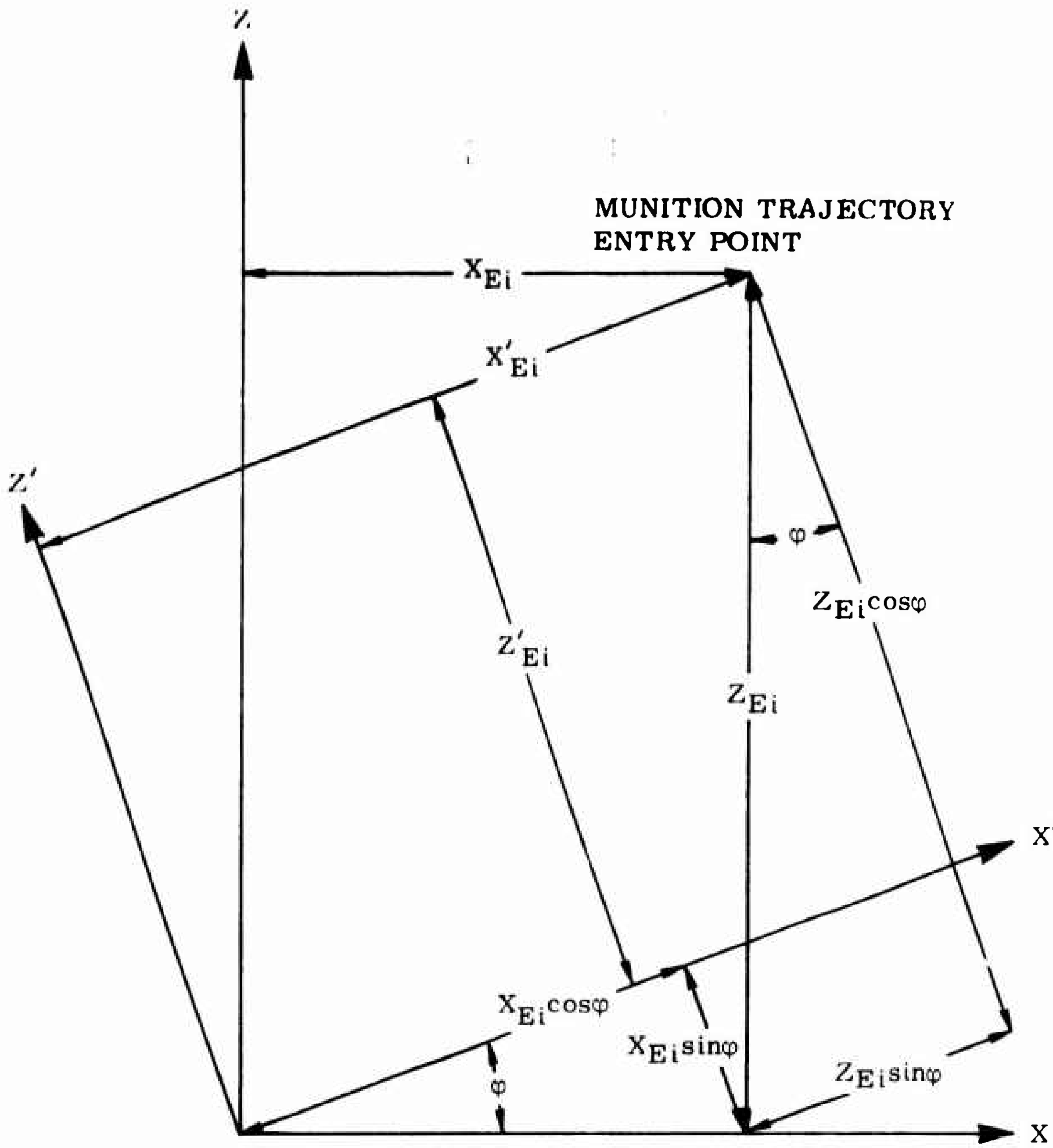

Figure 5. Relationship of Original and Transformed Munition Trajectory Entry Point Coordinates After Rotation in X-Y Plane Through Elevation Angle φ

DETERMINATION OF MUNITION/BRANCH ENCOUNTERS AND BURST HEIGHTS

The munition/branch encounters are evaluated layer by layer until each munition has detonated or has exited the bottom layer of the forest environment without detonating. A layer of branch input data is read into core, and the 400 munition trajectories (100 from each of the four azimuth angles) are evaluated against each of the branches in the layer. The munition/branch encounters are determined by considering one of the following three cases:

Case 1

For this case, the branch is parallel to the munition trajectory (i.e., the source and terminal Y' and Z' coordinates for the branch are identical, as shown in Figure 6). The square of the distance between the munition trajectory and the branch centerline is calculated using the expression:

$$D^2 = (Y'_j - Y'_{Ti})^2 + (Z'_j - Z'_{Ti})^2 \qquad (22)$$

where D is the distance between the munition trajectory and the branch centerline, Y'_j and Z'_j are the translated coordinates of either end of the j^{th} branch, and Y'_{Ti} and Z'_{Ti} are the translated entry or reentry coordinates of the i^{th} munition trajectory. (In this and subsequent munition/branch encounter computations, comparisons are made using squared quantities. This technique has been employed to reduce the computer running time for the program.) For this case, a munition body/branch encounter occurs if:

$$D^2 < (R_{bj} + R_m)^2 \qquad (23)$$

and a munition fuze/branch encounter occurs if:

$$D^2 < (R_{bj} + R_f)^2 \qquad (24)$$

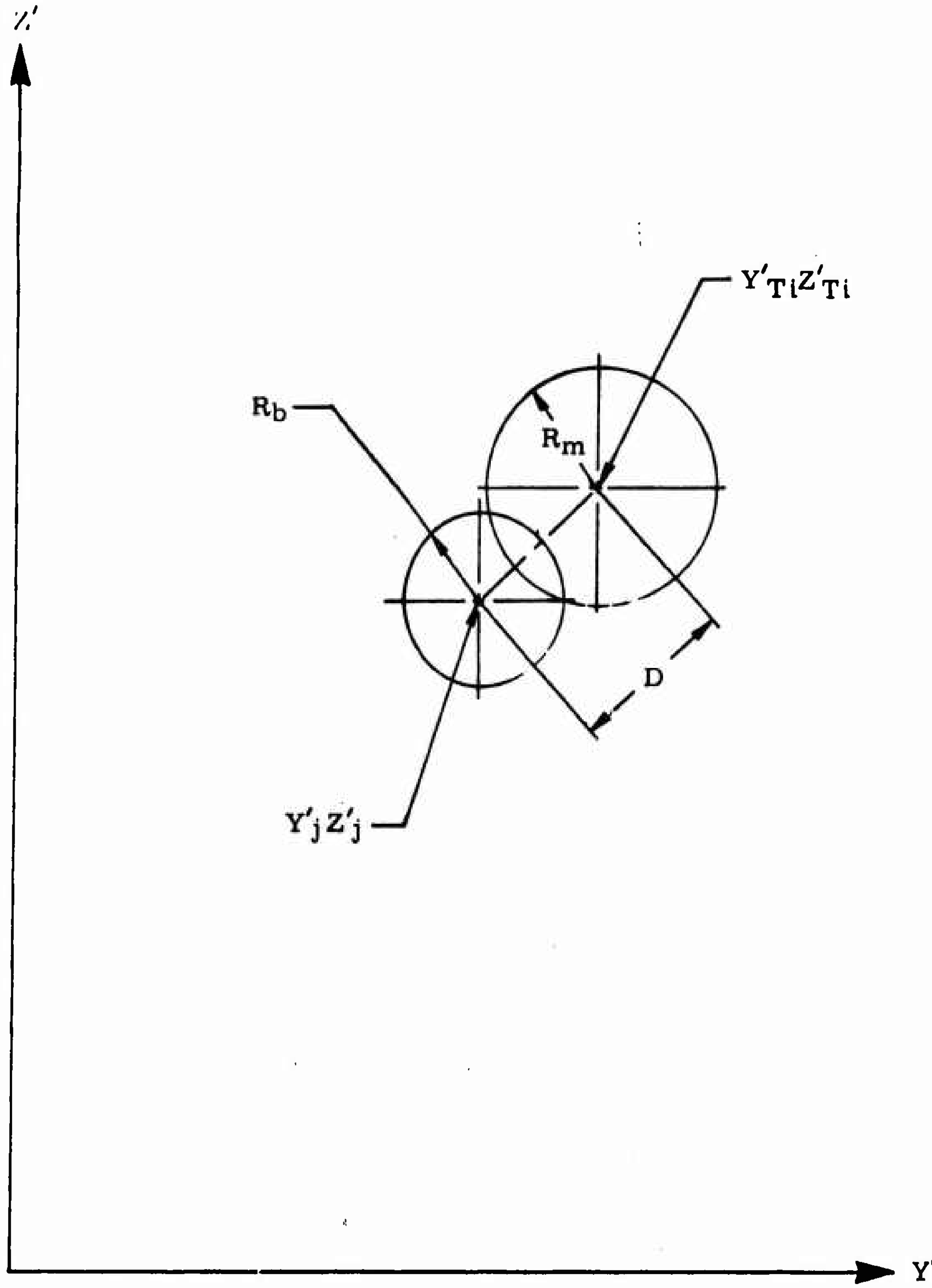

Figure 6. Munition/Branch Encounter - Case 1

where D is as defined above, R_{bj} is the average radius of the j^{th} branch, R_m is the munition body radius, and R_f is the munition fuze radius.

Only those branches with an average radius R_{bj} greater than the minimum branch radius required for detonation by a munition fuze are read in from the branch input tape. Therefore, a detonation caused by a munition fuze/branch encounter will occur if Inequality 24 is satisfied. In addition, a detonation caused by a munition body/branch encounter will occur if Inequality 23 is satisfied and if:

$$2R_{bj} > D_d \qquad (25)$$

where R_{bj} is as defined above, and D_d is the minimum branch diameter required for detonation by a munition body.

If Inequalities 23 and 25 or Inequality 24 are satisfied, a munition detonation is recorded, and the transformed coordinates for the detonation are determined using the expressions:

$$X'_{di} = X'_{min} \qquad (26)$$

$$Y'_{di} = Y'_{Ti} \qquad (27)$$

$$Z'_{di} = Z'_{Ti} \qquad (28)$$

where Y'_{Ti} and Z'_{Ti} are as defined above, X'_{di}, Y'_{di}, and Z'_{di} are the transformed coordinates of the detonation for the i^{th} munition trajectory, and X'_{min} is the smaller of the transformed X' coordinates of the branch ends.

The burst height for the detonation is then computed using the expression:

$$Z_{hi} = 0.003281(X'_{di}\sin\varphi + Z'_{di}\cos\varphi) \qquad (29)$$

where X'_{di}, Z'_{di}, and φ are as defined above, and Z_{hi} is the burst height of the i^{th} munition detonation.

Case 2

For this case, the branch is not parallel to the munition trajectory, and the normal from the munition trajectory to the centerline of the branch intersects the branch between its ends or nodes (see Figure 7). To satisfy the conditions for this case, the following relationship must exist:

$$(R^2_{max} - D^2) < (A^2 + B^2) \tag{30}$$

where R^2_{max} is the greater of R^2_1 and R^2_2:

$$R^2_1 = (Y'_{1j} - Y'_{Ti})^2 + (Z'_{1j} - Z'_{Ti})^2 \tag{31}$$

$$R^2_2 = (Y'_{2j} - Y'_{Ti})^2 + (Z'_{2j} - Z'_{Ti})^2 \tag{32}$$

$$A = Z'_{1j} - Z'_{2j} \tag{33}$$

$$B = Y'_{2j} - Y'_{1j} \tag{34}$$

Y'_{1j} and Z'_{1j} are the transformed Y' and Z' coordinates of one branch end, Y'_{2j} and Z'_{2j} are the transformed Y' and Z' coordinates of the other branch end, D is the distance between the munition trajectory and the branch centerline, D^2 can be determined using the equation for computing the distance between a point and a line, as follows:

$$D^2 = \frac{(AY'_{Ti} + BZ'_{Ti} + C)^2}{A^2 + B^2} \tag{35}$$

where:

$$C = -Z'_{1j}(B) - Y'_{1j}(A) \tag{36}$$

and all other variables are as defined above. If Inequalities 23 and 25 or Inequality 24 are satisfied, a munition detonation is recorded, and the

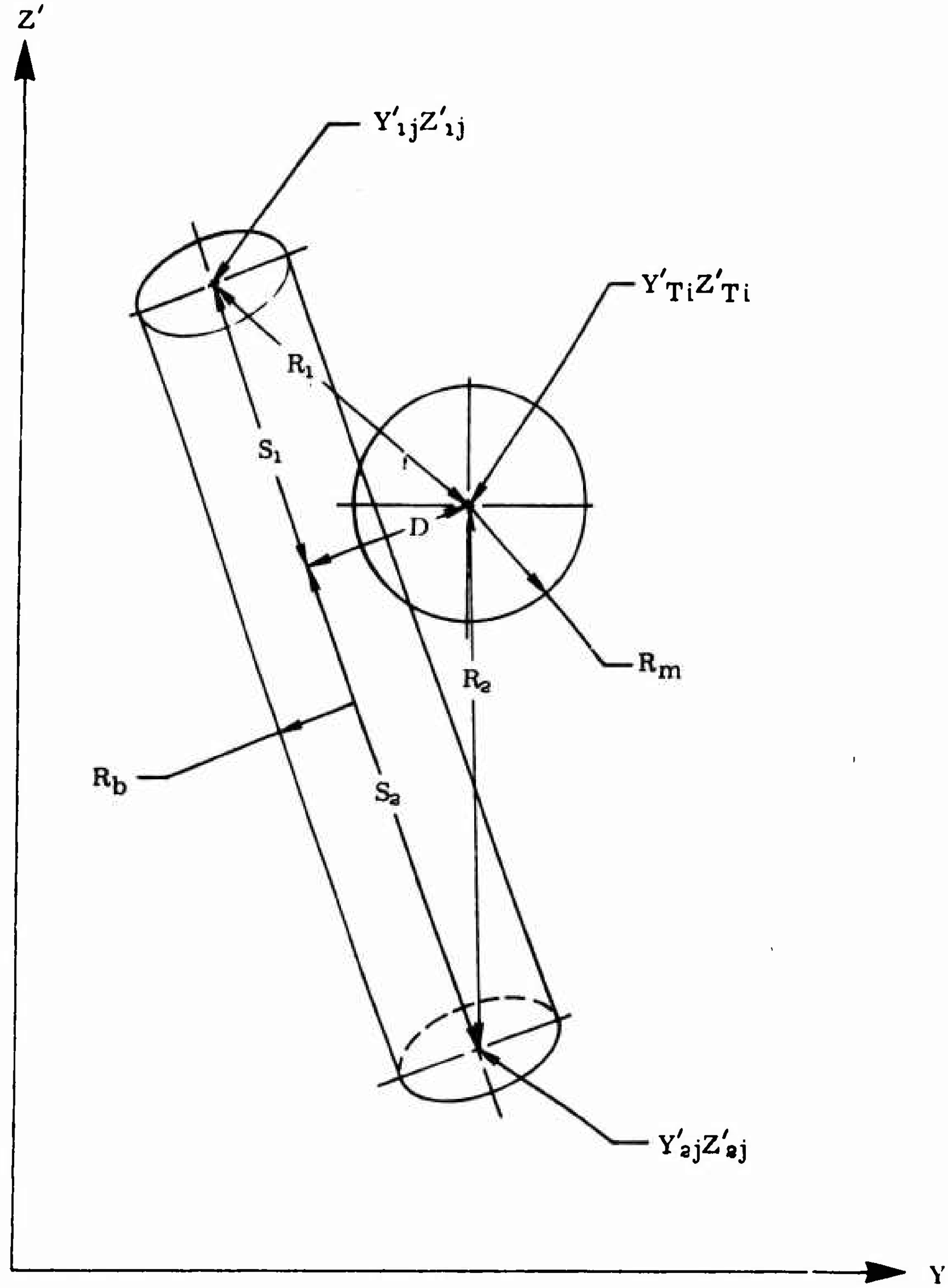

Figure 7. Munition/Branch Encounter - Case 2

transformed coordinates for the detonation are determined using the expressions:

$$X'_{di} = \frac{S_1 X'_{2j} + S_2 X'_{1j}}{S_1 + S_2} \tag{37}$$

$$Y'_{di} = Y'_{Ti} \tag{38}$$

$$Z'_{di} = Z'_{Ti} \tag{39}$$

where:

$$S_1 = \sqrt{A^2 + B^2} - S_2 \tag{40}$$

$$S_2 = \sqrt{R^2_2 - D^2} \tag{41}$$

and all other variables are as defined above. The burst height for the detonation is then computed using Equation 29.

Case 3

For this case, the branch is not parallel to the munition trajectory, and the normal from the trajectory to the centerline of the branch does not intersect the branch between its ends or nodes (see Figure 8). To satisfy the conditions for this case, the following relationship must exist:

$$(R^2_{max} - D^2) \geq (A^2 + B^2) \tag{42}$$

where all variables are as defined above. A munition body/branch encounter occurs if:

$$R^2_{min} < (R_{bj} + R_m)^2 \tag{43}$$

and a munition fuze/branch encounter occurs if:

$$R^2_{min} < (R_{bj} + R_f)^2 \tag{44}$$

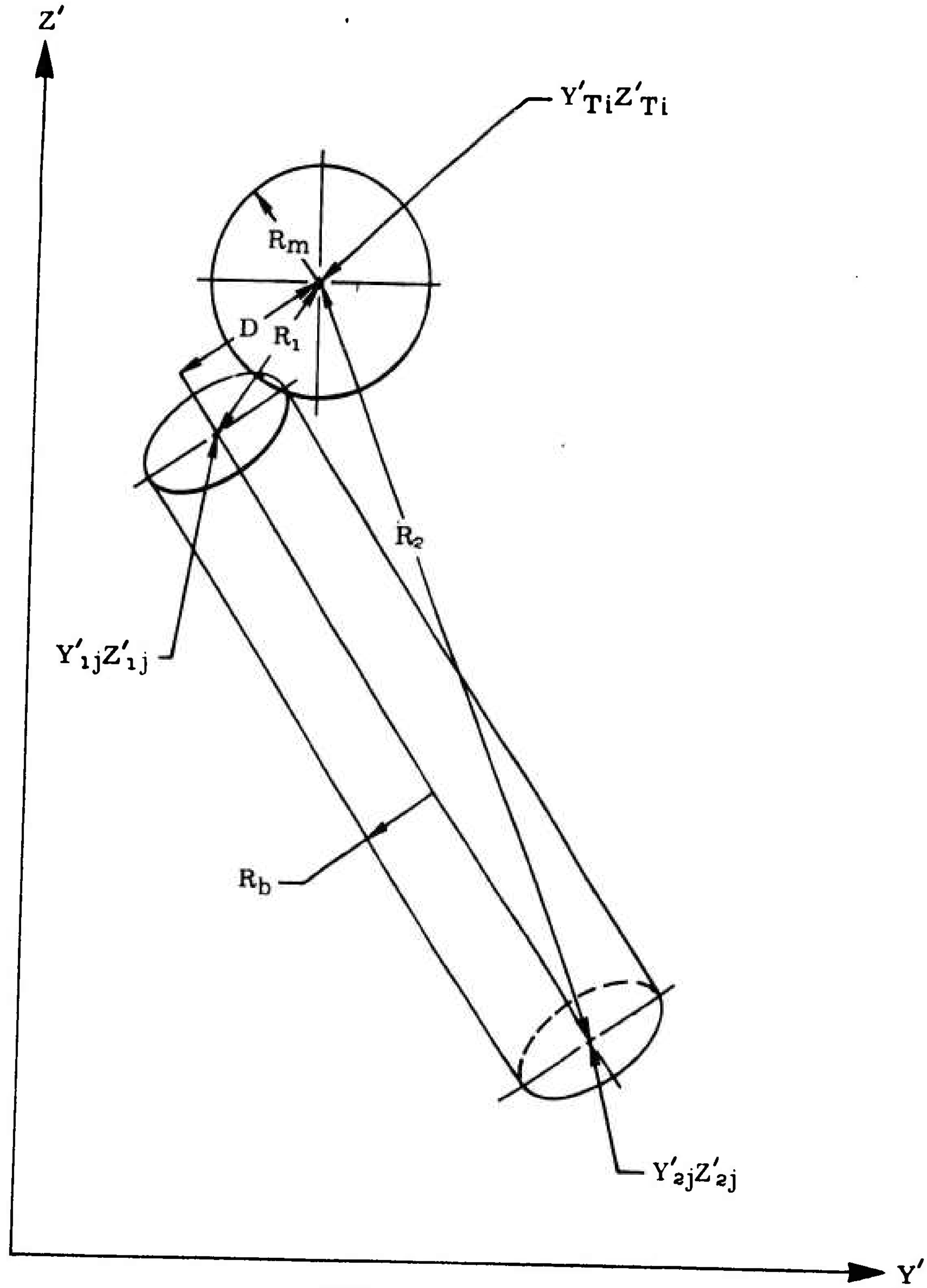

Figure 8. Munition/Branch Encounter - Case 3

where R_{min} is the smaller of R^2_1 and R^2_2, and R_{bj}, R_m, and R_f are as defined above. If Inequalities 25 and 43 or Inequality 44 are satisfied, a munition detonation is recorded, and the transformed coordinates for the detonation are determined using the expressions:

$$X'_{di} = \begin{cases} X'_{1j} & \text{if } R^2_1 < R^2_2 \\ X'_{2j} & \text{if } R^2_2 < R^2_1 \end{cases} \tag{45}$$

$$Y'_{di} = Y'_{Ti} \tag{46}$$

$$Z'_{di} = Z'_{Ti} \tag{47}$$

where all variables are as defined above. The burst height for the detonation is then computed using Equation 29.

COMPUTATION OF CUMULATIVE BURST HEIGHT DISTRIBUTIONS

After the burst heights for the 400 trajectories have been determined, the model groups the values into intervals as specified in the input and determines the number of detonations and the cumulative frequency for each burst height interval. The expression which determines the cumulative frequency is:

$$C_k = C_{k-1} + \frac{100F_k}{400} \tag{48}$$

where C_k is the cumulative frequency (in percent) for the k^{th} interval, and F_k is the number of detonations in the k^{th} interval.

SECTION III

FLOWCHARTS

This section contains the two sets of flowcharts which depict the logical structure of the main program and its subroutine. The flowcharts are based upon the logical intent of the model rather than on displaying the methods used to code the specific program routines. Program MAIN (Figure 9) computes the munition burst heights for each of the 400 munition trajectories, and Subroutine CFRQ (Figure 10) computes, prints, and optionally punches the cumulative burst height distribution for the munition and elevation angle being considered.

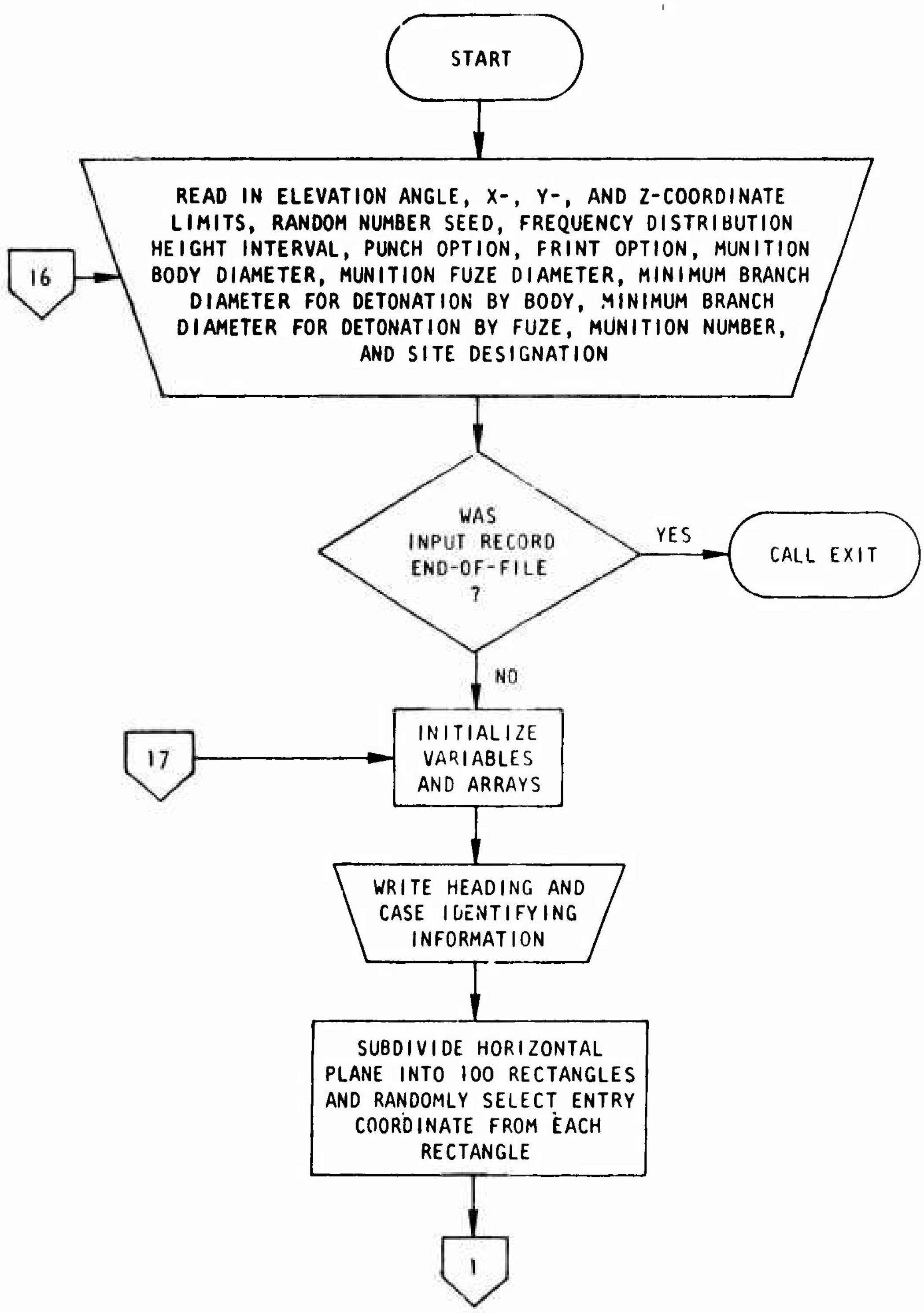

Figure 9. Flowchart, Program MAIN

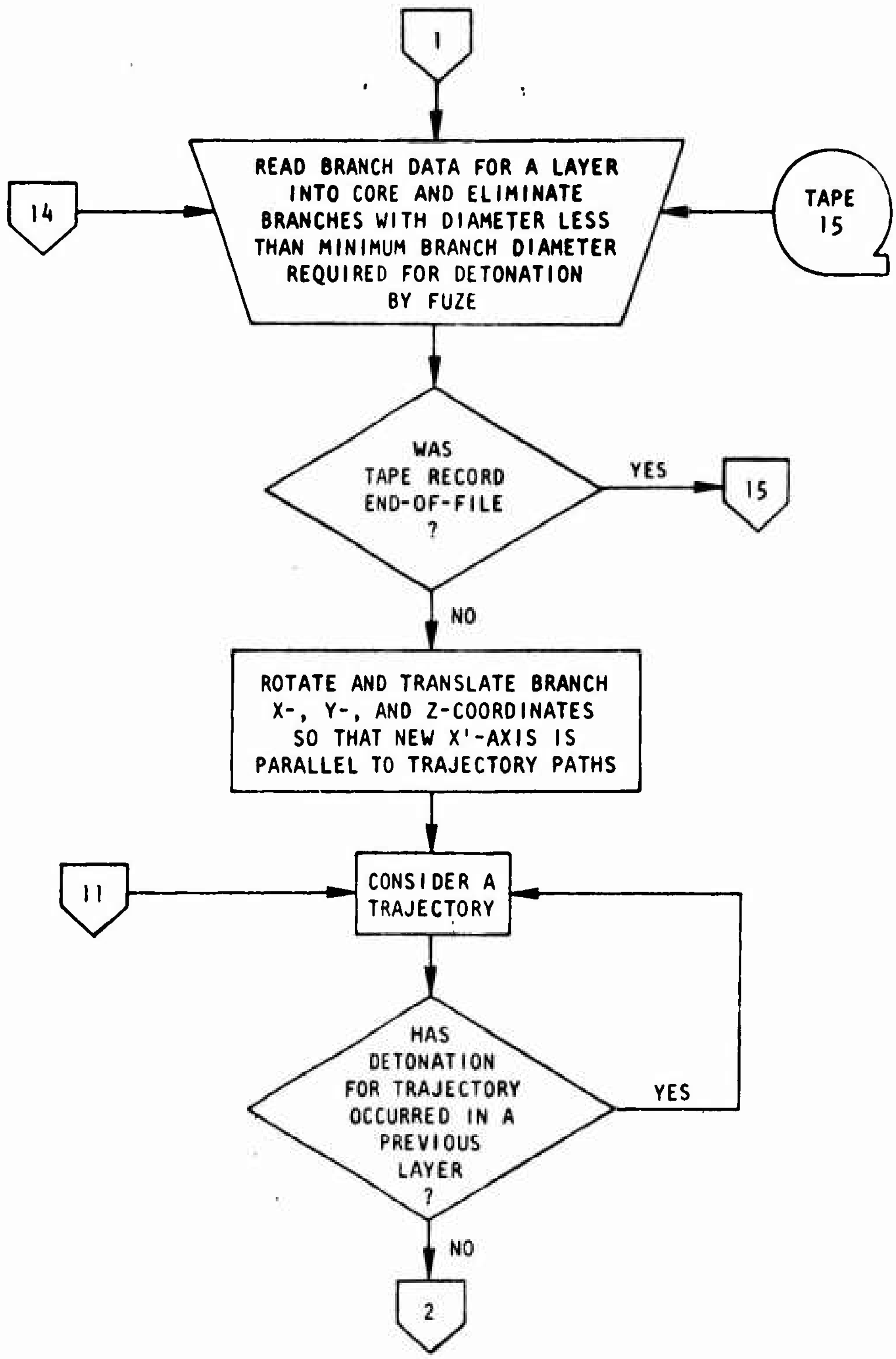

Figure 9. (Continued)

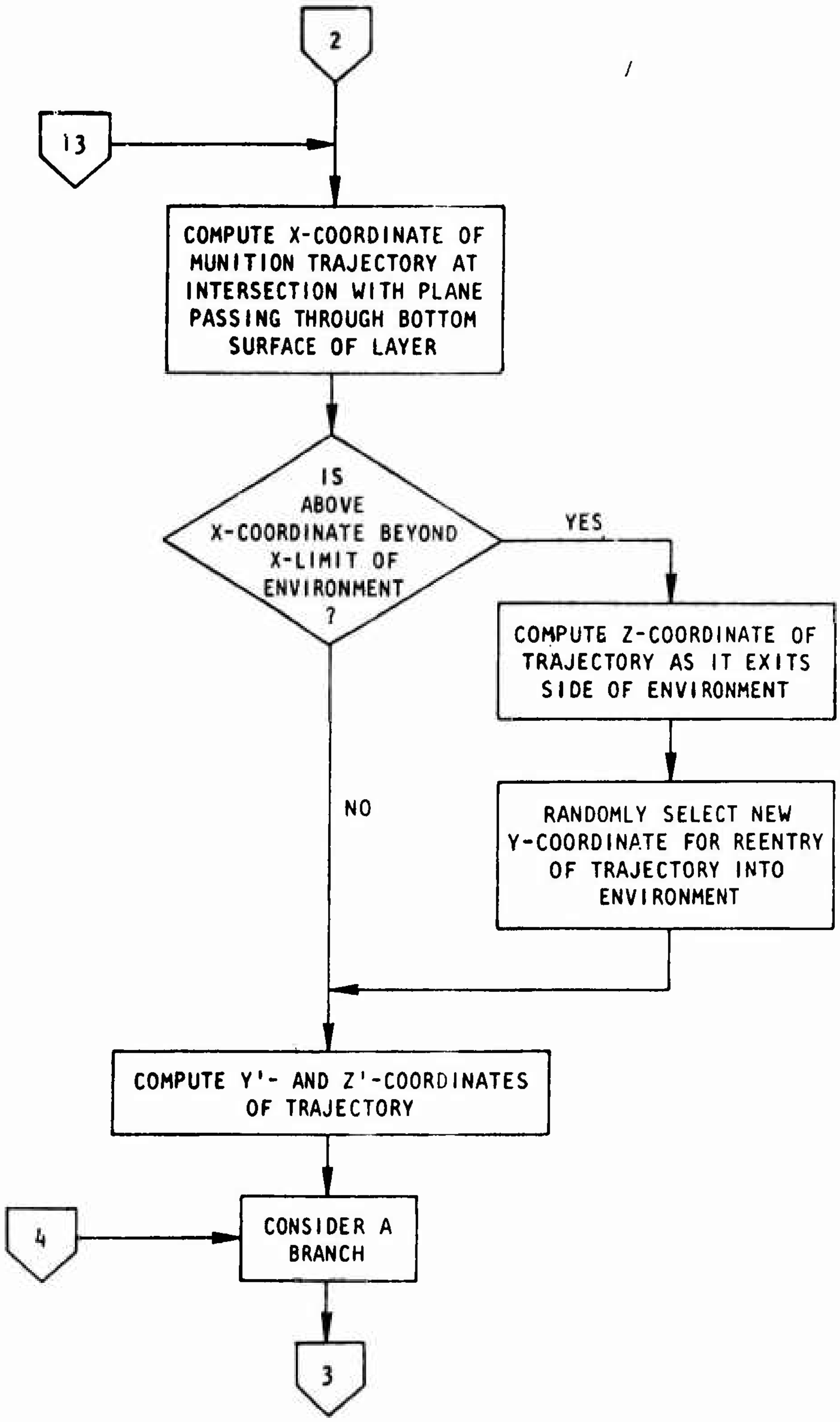

Figure 9. (Continued)

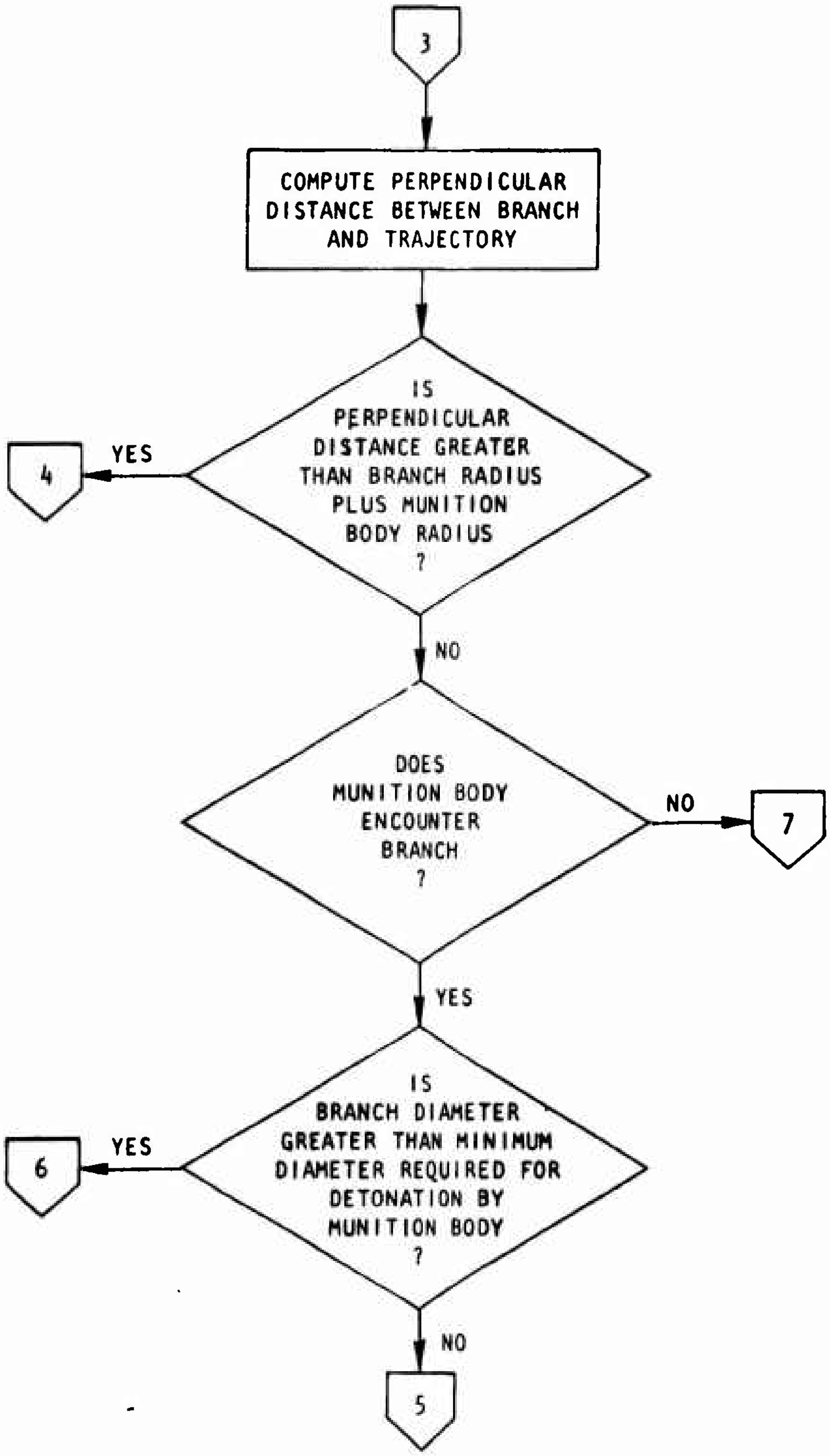

Figure 9. (Continued)

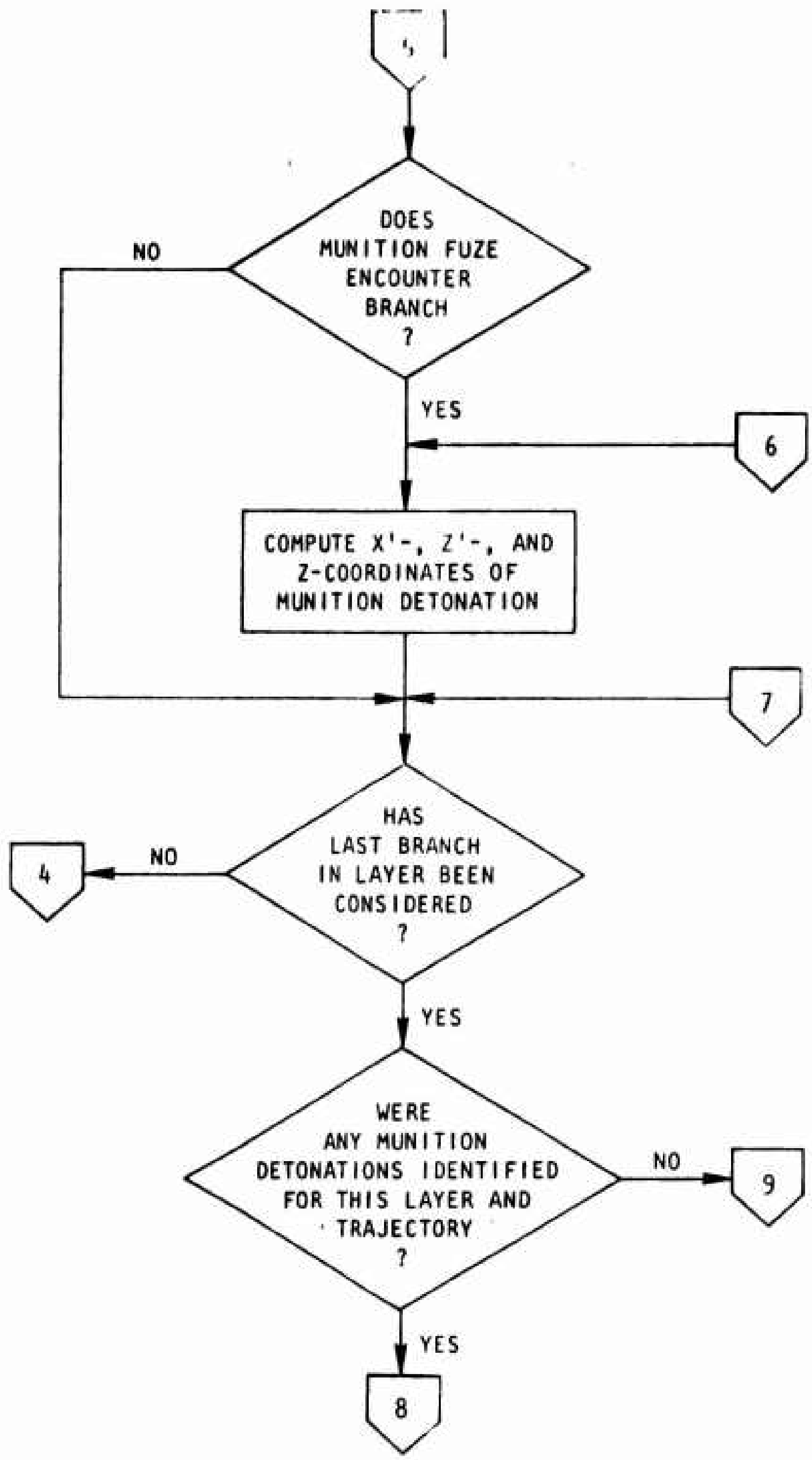

Figure 9. (Continued)

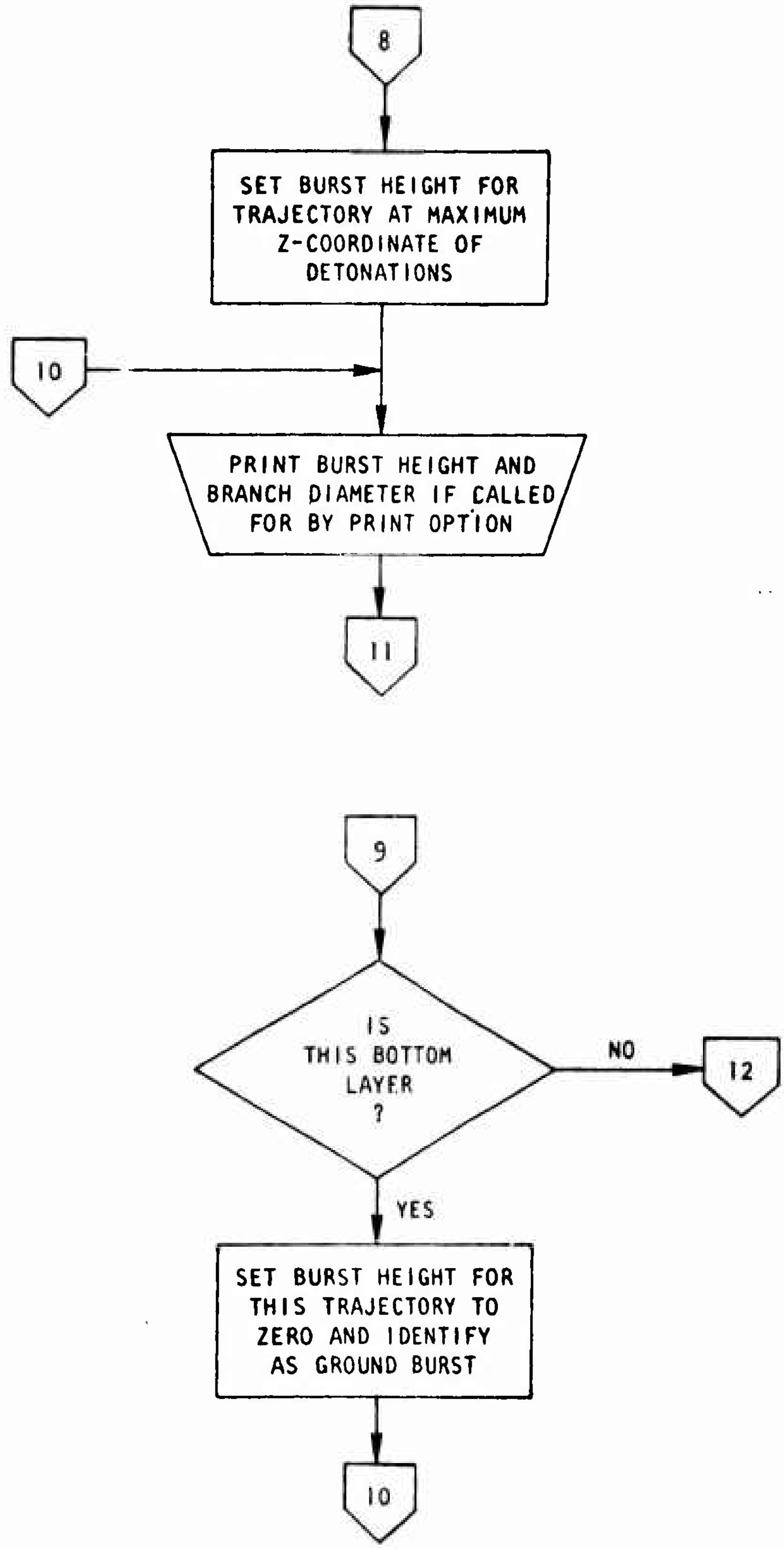

Figure 9. (Continued)

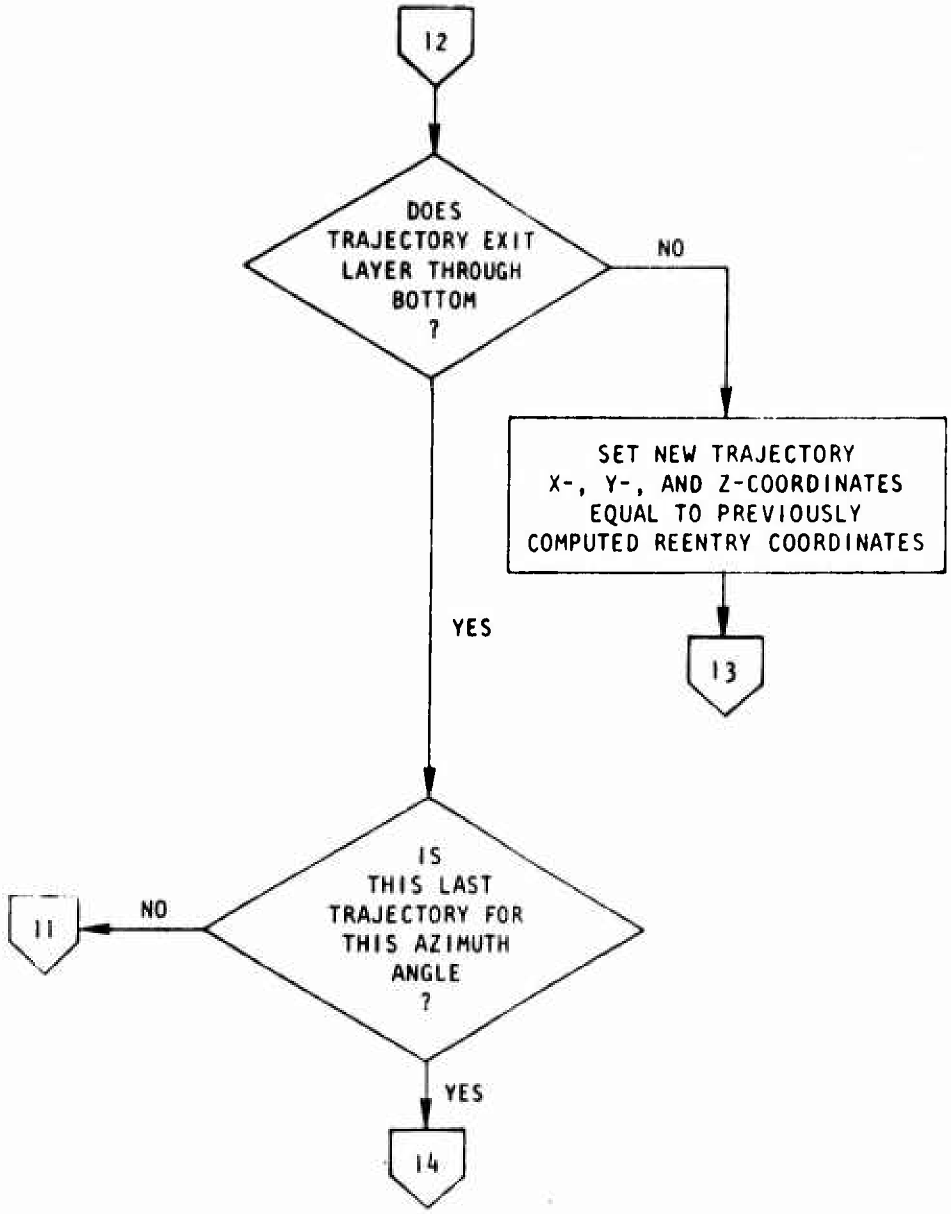

Figure 9. (Continued)

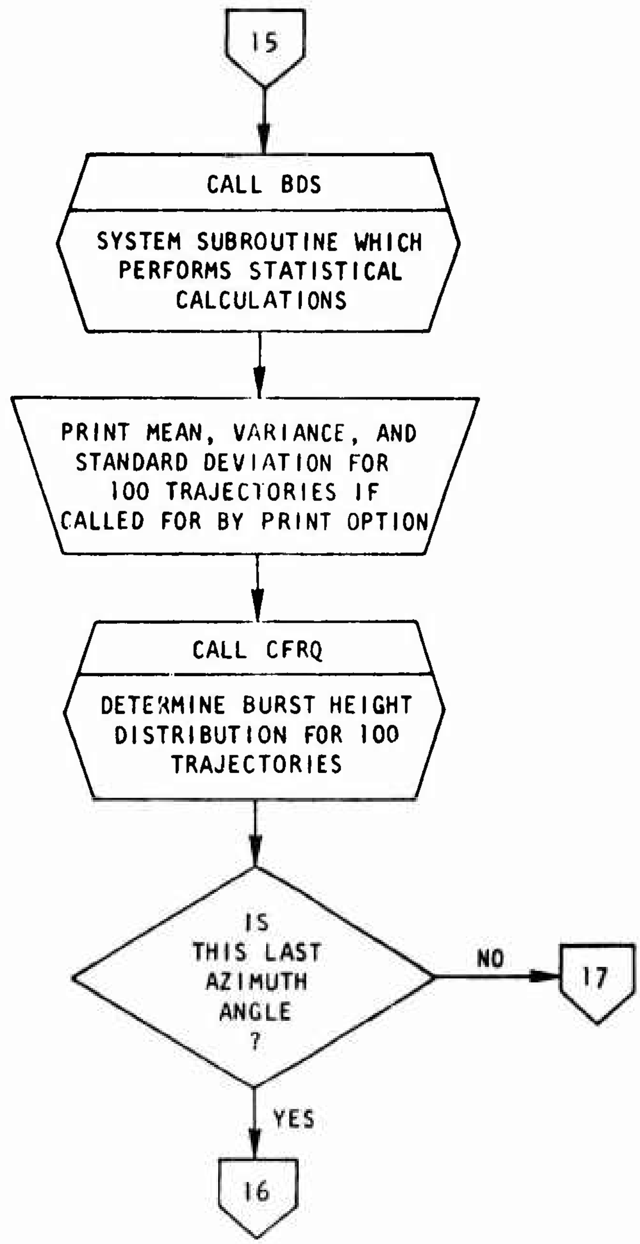

Figure 9. (Concluded)

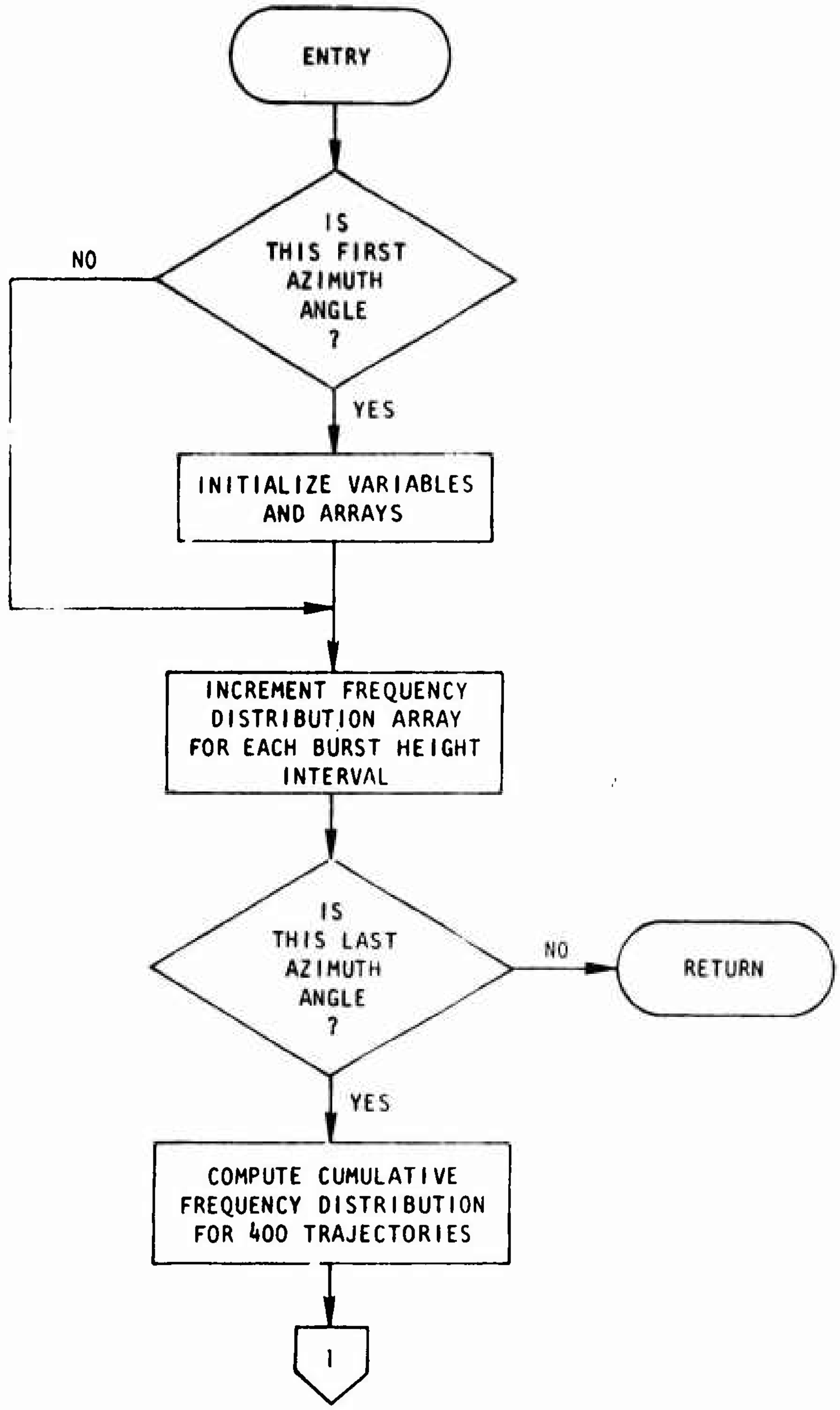

Figure 10. Flowchart, Subroutine CFRQ

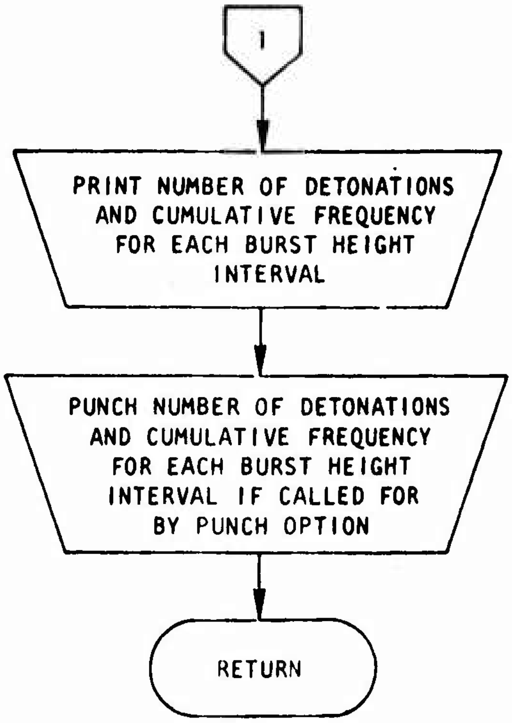

Figure 10. (Concluded)

SECTION IV

SOURCE LISTING

Figure 11 presents a complete source listing of the BHD Program, which was designed for operation on the Control Data Corporation 6600 computer system at Eglin Air Force Base, Florida.

```
      PROGRAM P7821(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE15,TAPE10)
C
C     THIS PROGRAM DETERMINES BURST HEIGHT DISTRIBUTIONS FOR
C     MUNITIONS AERIALLY DELIVERED INTO FOREST ENVIRONMENTS
C
      DIMENSION XTRAJ(100),YTRAJ(100),ZTRAJ(100),BRANCH(6),R(2),X1(1500)
     1,Y1(1500),Z1(1500),X2(1500),Y2(1500),Z2(1500),AVDIAM(1500),DET(100
     2),KEPDET(300),KEPDIAM(300)
      DIMENSION STAT(8)
      COMMON NMUN,SITE,PHISAV,THETA,XINT,IPUNCH,IPRINT,DET
      COMMON X1,Y1,Z1,X2,Y2,Z2,AVDIAM
      INTEGER REPEAT
      REAL LENSQR,LENGTH,KEPDET,KEPDIAM
      CALL EOF(KEOF)
    8 ASSIGN 300 TO KEOF
C
C     READ IN ELEVATION ANGLE, X, Y, AND Z ENVIRONMENT BOUNDARIES,
C     RANDOM NUMBER SEED, BURST HEIGHT INTERVAL, PUNCH OPTION (0=DON'T
C     PUNCH CARDS, 1=PUNCH CARDS), PRINT OPTION (0=PRINT DATA FOR EACH
C     DETONATION AND BURST HEIGHT DISTRIBUTION TABLE, 1=PRINT BURST
C     HEIGHT DISTRIBUTION TABLE), MUNITION BODY DIAMETER, MUNITION
C     FUZE DIAMETER, MINIMUM BRANCH DIAMETER REQUIRED FOR DETONATION
C     BY MUNITION BODY, MINIMUM BRANCH DIAMETER REQUIRED FOR DETONATION
C     BY MUNITION FUZE, MUNITION NUMBER, AND SITE DESIGNATION
C
      READ(5,1000) PHI,XLIMIT,YLIMIT,ZLIMIT,DUMMY,XINT,IPUNCH,IPRINT,
     1BDIAM,FDIAM,BBDIAM,BFDIAM,NMUN,SITE
C
C     PROVIDE TRANSFERS, INITIALIZE VARIABLES AND ARRAYS, AND PERFORM
C     INITIAL CALCULATIONS.
C
      PHISAVE=PHI
      THETA=0.
    6 REWIND 15
      PHI=PHISAVE
      DO 1 I=1,100
    1 DET(I)=0.
      DO 2 I=1,300
      KEPDIAM(I)=0.
    2 KEPDET(I)=0.
      ASSIGN 110 TO KEOF
      KEEP=0
      IF(THETA.NE.0..AND.IPRINT.EQ.1) GO TO 7
      WRITE(6,2000)SITE,NMUN,BDIAM,PHI,FDIAM,THETA,DUMMY,BBDIAM,
     1BFDIAM
```

Figure 11. BHD Program Source Listing

```
      IF(IPRINT.EQ.1) GO TO 7
      WRITE (6,2010)
    7 KTR=0
      PHISAV = PHI
      PHI=-PHI
      XTRAN=0.
      YTRAN=0.
      ITHETA=THETA/90. + 1.
      GO TO(5,3,3,4),ITHETA
    3 IF(ITHETA.EQ.3) XTRAN=XLIMIT
      YTRAN=YLIMIT
      GO TO 5
    4 XTRAN=XLIMIT
    5 FRAD=FDIAM/2.
      BRAD=BDIAM/2.
      AZIMUTH=.017453295*THETA
      PHI=.017453295*PHI
      SINT=SIN(AZIMUTH)
      SINP=SIN(PHI)
      COST=COS(AZIMUTH)
      COSP=COS(PHI)
      CTCP=COST*COSP
      CTSP=COST*SINP
      STSP=SINT*SINP
      STCP=SINT*COSP
      XTRANC=XTRAN*COSP
      XTRANS=XTRAN*SINP
      SPOCP=SINP/COSP
C
C     INITIALIZE RANDOM NUMBER GENERATOR, SUBDIVIDE X-Y PLANE INTO
C     100 RECTANGLES, AND RANDOMLY SELECT TRAJECTORIES PASSING
C     THROUGH EACH RECTANGLE.
C
      XLIM0=XLIMIT/10.
      YLIM0=YLIMIT/10.
      XLIM00=XLIMIT/100.
      YLIM00=YLIMIT/100.
      CALL RANSET(DUMMY)
      DO 10 I=1,10
      XCELL=XLIM0*(I-1)
      K=(I-1)*10
      DO 10 J=1,10
      YCELL=YLIM0*(J-1)
      NRNXY=RANF(TRASH)*100.
      NRNX=NRNXY/10
      RNX=NRNX
```

Figure 11. (Continued)

```
      RNY=NRNXY-(NRNX*10)
      L=K+J
      XTRAJ(L)=XCELL+RNX*XLIM00
      YTRAJ(L)=YCELL+RNY*YLIM00
   10 ZTRAJ(L)=ZLIMIT
C
C     READ BRANCH DATA INTO CORE, ELIMINATE BRANCHES WITH DIAMETER LESS
C     THAN MINIMUM REQUIRED FOR DETONATION BY MUNITION FUZE, AND
C     TRANSFORM COORDINATES OF BRANCH ENDS TO X'-Y'-Z' COORDINATE
C     SYSTEM WITH X' AXIS PARALLEL TO TRAJECTORY PATHS.
C
      TOP=ZLIMIT
   15 READ(15) LEVEL,NBRAN,BOTTOM,BRANCH,DIAM
      K=0
      BMT=BOTTOM-TOP
      DO 30 I=1,NBRAN
      IF(DIAM.LT.BFDIAM)GO TO 20
      K=K + 1
      X1(K)=BRANCH(1)*CTCP+BRANCH(2)*STCP+BRANCH(3)*SINP+XTRANC
      X2(K)=BRANCH(4)*CTCP+BRANCH(5)*STCP+BRANCH(6)*SINP+XTRANC
      Y1(K)=-BRANCH(1)*SINT + BRANCH(2)*COST + YTRAN
      Y2(K)=-BRANCH(4)*SINT + BRANCH(5)*COST + YTRAN
      Z1(K)=-BRANCH(1)*CTSP-BRANCH(2)*STSP+BRANCH(3)*COSP-XTRANS
      Z2(K)=-BRANCH(4)*CTSP-BRANCH(5)*STSP+BRANCH(6)*COSP-XTRANS
      AVDIAM(K)=DIAM
   20 IF(I.EQ.NBRAN)GO TO 30
      READ(15) BRANCH,DIAM
   30 CONTINUE
C
C     ENTER TRAJECTORY LOOP TO DETERMINE MUNITION/BRANCH ENCOUNTERS
C
      DO 100 I=1,100
      IF(XTRAJ(I)-999999.) 135,100
C
C     DETERMINE IF TRAJECTORY EXITS SIDE OR BOTTOM OF LAYER AND
C     DETERMINE COORDINATES FOR REENTRY POINTS
C
  135 XPLUS=BMT/SPOCP
      ZOUT=TOP
   35 XOUT=XTRAJ(I) + XPLUS
      IF(XOUT-XLIMIT) 36,36,136
  136 XOUT=0.
      ZOUT=ZOUT-(XTRAJ(I)-XLIMIT)*SPOCP
      XPLUS=(BOTTOM-ZOUT)/SPOCP
      REPEAT=1
      GO TO 40
```

Figure 11. (Continued)

```
   36 ZOUT=BOTTOM
      REPEAT=0
C
C     ROTATE COORDINATES OF TRAJECTORY ENTRY POINT INTO X'-Y'-Z'
C     COORDINATE SYSTEM
C
   40 YTRAJP=YTRAJ(I)
      ZTRAJP=-XTRAJ(I)*SINP+ZTRAJ(I)*COSP
C
C     ENTER BRANCH LOOP AND COMPUTE FACTORS USED TO
C     DETERMINE IF MUNITION ENCOUNTERS BRANCH
C
      KEEP=0
      DO 70 J=1,K
      AVRAD=AVDIAM(J)/2.
      FACTRF=(AVRAD+FRAD)**2
      FACTRB=(AVRAD+BRAD)**2
      A=Z1(J) - Z2(J)
      B=Y2(J) - Y1(J)
      C=-Z1(J)*B - Y1(J)*A
      LENSQR=A**2 + B**2
      LENGTH=SQRT(LENSQR)
C
C     DETERMINE IF BRANCH IS PARALLEL TO MUNITION TRAJECTORY (CASE 1)
C
      IF(A) 53,141
  141 IF(B) 53,142
  142 D=(Y1(J)-YTRAJP)**2+(Z1(J)-ZTRAJP)**2
C
C     IF D IS GREATER THAN SQUARE OF SUM OF BRANCH RADIUS PLUS
C     MUNITION BODY RADIUS, SKIP TO NEXT BRANCH
C
      IF(D-FACTRB) 143,143,70
C
C     FOR MUNITION BODY/BRANCH ENCOUNTER, DETERMINE IF BRANCH IS
C     LARGE ENOUGH TO DETONATE MUNITION
C
  143 IF(AVDIAM(J)-BBDIAM) 50,44,44
C
C     DETERMINE COORDINATES OF DETONATION
C
   44 BRDIAM=AVDIAM(J)
      YDET=YTRAJP
      ZDET=ZTRAJP
      IF(X1(J)-X2(J)) 145,145,45
  145 XDET=X1(J)
```

Figure 11. (Continued)

```
      GO TO 65
   45 XDET=X2(J)
      GO TO 65
C
C     DETERMINE IF MUNITION FUZE/BRANCH ENCOUNTER OCCURS, AND (IF
C     SO) DETERMINE COORDINATES OF DETONATION
C
   50 IF(D-FACTRF) 44,44,70
C
C     IF D IS GREATER THAN SQUARE OF SUM OF BRANCH RADIUS PLUS
C     MUNITION BODY RADIUS, SKIP TO NEXT BRANCH
C
   53 D=((A*YTRAJP+B*ZTRAJP+C)/LENGTH)**2
      IF(D-FACTRB) 153,153,70
C
C     DETERMINE IF NORMAL FROM TRAJECTORY TO BRANCH CENTERLINE
C     INTERSECTS BRANCH BETWEEN ENDS (CASE 2)
C
  153 R(1)=(Y1(J)-YTRAJP)**2+(Z1(J)-ZTRAJP)**2
      R(2)=(Y2(J)-YTRAJP)**2+(Z2(J)-ZTRAJP)**2
      SSQ=AMAX1(R(1),R(2))-D
      IF(SSQ-LENSQR) 154,57,57
C
C     FOR MUNITION BODY/BRANCH ENCOUNTER, DETERMINE IF BRANCH IS
C     LARGE ENOUGH TO DETONATE MUNITION
C
  154 IF(AVDIAM(J)-BBDIAM) 56,46,46
C
C     DE   MINE COORDINATES OF DETONATION
C
   46 SS2=R(2)-D
      S2=0.
      IF(SS2) 47,47,54
   54 S2=SQRT(SS2)
   47 S1=LENGTH - S2
      PX=(S2*X1(J) + S1*X2(J))/(S1 + S2)
   55 BRDIAM=AVDIAM(J)
      YDET=YTRAJP
      ZDET=ZTRAJP
      XDET=PX
      GO TO 65
C
C     DETERMINE IF MUNITION FUZE/BRANCH ENCOUNTER OCCURS, AND (IF
C     SO) DETERMINE COORDINATES OF DETONATION
C
```

Figure 11. (Continued)

```
 56 IF(D-FACTRF) 46,46,70
C
C     NORMAL FROM TRAJECTORY TO BRANCH CENTERLINE DOES NOT INTERSECT
C     BRANCH BETWEEN ITS ENDS (CASE 3) - DETERMINE IF MUNITION BODY
C     ENCOUNTERS ONE OF BRANCH ENDS
C
 57 MIN=1
    IF(R(2)-R(1)) 164,155,155
164 MIN=2
155 IF(R(MIN)-FACTRB) 156,156,70
C
C     FOR MUNITION BODY/BRANCH ENCOUNTER, DETERMINE IF BRANCH IS
C     LARGE ENOUGH TO DETONATE MUNITION, AND (IF SO) DETERMINE
C     COORDINATES OF DETONATION
C
156 IF(AVDIAM(J)-BBDIAM) 63,159,159
159 GO TO(59,61),MIN
 59 PX=X1(J)
    GO TO 55
 61 PX=X2(J)
    GO TO 55
C
C     DETERMINE IF MUNITION FUZE/BRANCH ENCOUNTER OCCURS, AND (IF
C     SO) DETERMINE COORDINATES OF DETONATION
C
 63 IF(R(MIN)-FACTRF) 159,159,70
C
C     DETERMINE Z COORDINATES FOR DETONATIONS AND SAVE Z COORDINATES
C     AND BRANCH DIAMETERS FOR ALL DETONATIONS UNTIL ALL BRANCHES
C     IN LAYER HAVE BEEN PROCESSED
C
 65 ZHGT=.003281*(XDET*SINP + ZDET*COSP)
    IF(ZHGT.LT.0.)ZHGT=0.
    KEEP=KEEP + 1
    KEPDET(KEEP)=ZHGT
    KEPDIAM(KEEP)=BRDIAM
 70 CONTINUE
C
C     SET BURST HEIGHT AT MAXIMUM Z COORDINATE OF DETONATIONS, INCRE-
C     MENT AIR BURST COUNTER, PRINT BURST HEIGHT AND BRANCH DIAMETER
C     FOR DETONATION (IF REQUIRED), AND FLAG TRAJECTORY SO THAT IT
C     WILL NOT BE CONSIDERED IN SUBSEQUENT LAYERS
C
    IF(KEEP) 170,80
```

Figure 11. (Continued)

```
  170 THEDET=KEPDET(1)
      THEDIAM=KEPDIAM(1)
      DO 71 LL=1,KEEP
      IF(KEPDET(LL)-THEDET) 71,71,171
  171 THEDET=KEPDET(LL)
      THEDIAM=KEPDIAM(LL)
   71 CONTINUE
   75 KTR = KTR + 1
      IF(IPRINT.EQ.0) WRITE(6,1004) THEDET,THEDIAM
      XTRAJ(I)=999999.
   76 DET(I)=THEDET
      GO TO 100
C
C     SET BURST HEIGHT FOR TRAJECTORY TO ZERO IF BOTTOM LAYER HAS BEEN
C     PROCESSED AND IF TRAJECTORY DOES NOT HAVE TO BE REENTERED
C
   80 IF(BOTTOM) 85,180
  180 IF(REPEAT) 85,181
  181 THEDET=0.
      THEDIAM=0.
      XTRAJ(I)=999999.
      GO TO 76
C
C     SET NEW ENTRY OR REENTRY COORDINATES FOR TRAJECTORY, DEFINE TOP
C     OF NEXT LAYER, AND TRANSFER TO READ BRANCH DATA FOR NEXT LAYER
C
   85 XTRAJ(I)=XOUT
      ZTRAJ(I)=ZOUT
      IF(REPEAT) 99,100
   99 YTRAJ(I)=RANF(TRASH)*YLIMIT
      GO TO 35
  100 CONTINUE
      TOP=BOTTOM
      GO TO 15
C
C     AFTER ALL LAYERS HAVE BEEN PROCESSED FOR 100 TRAJECTORIES, CALL
C     SYSTEM SUBROUTINE BDS TO COMPUTE MEAN, VARIANCE, AND STANDARD
C     DEVIATION FOR BURST HEIGHTS, AND PRINT NUMBER OF GROUND BURSTS
C     AND STATISTICAL DATA IF CALLED FOR BY PRINT OPTION
C
  110 DO 200 I=1,100
      IF(DET(I)) 201,200
  200 CONTINUE
      STAT(1)=STAT(5)=STAT(6)=0.
      GO TO 210
  201 CALL BDS(DET,100,STAT)
```

Figure 11. (Continued)

```
  210 KTGRD = 100 - KTR
      IF(IPRINT.EQ.1) GO TO 215
      WRITE (6,1007) KTGRD
      WRITE(6,1006)STAT(1),STAT(5),STAT(6)
  215 DUMMY=DUMMY+1.
C
C     CALL SUBROUTINE CFRQ TO DETERMINE CUMULATIVE BURST HEIGHT
C     DISTRIBUTION AND INCREMENT AZIMUTH ANGLE.  IF LAST AZIMUTH ANGLE
C     HAS NOT BEEN PROCESSED, TRANSFER TO PROCESS TRAJECTORIES FOR
C     NEXT AZIMUTH ANGLE.  OTHERWISE, TRANSFER TO READ IN DATA FOR
C     NEXT CASE OR TERMINATE PROGRAM.
C
      CALL CFRQ
      THETA=THETA+90.
      IF(THETA-360.) 6,8,6
 1000 FORMAT(6F10.0,2I2/4F10.0,10X,I5,A10)
 1001 FORMAT(I4,I6,7F7.0,F6.0)
 1002 FORMAT(6F7.0,F6.0)
 1004 FORMAT(50X,2F11.2)
 1006 FORMAT(1H0,50X,20HMEAN BURST HEIGHT = ,F10.2,6H  FEET/
     *       60X,11HVARIANCE = ,F12.4/
     *       50X,21HSTANDARD DEVIATION = ,F10.2,6H  FEET)
 1007 FORMAT(1H0,50X,15HTHERE WERE ALSO,I4,15H GROUND BURSTS.)
 2000 FORMAT(1H1,60X,4HSITE,A10/1H0,56X,17HMUNITION NUMBER =,I5/1H0,20X,
    124HMUNITION BODY DIAMETER =,F10.2,2X,11HMILLIMETERS,10X,17HELEVATI
    2ON ANGLE =,F10.0,2X,7HDEGREES/1H0,20X,24HMUNITION FUZE DIAMETER =,
    3F10.2,2X,11HMILLIMETERS,10X,17HAZIMUTH ANGLE   =,F10.0,2X,7HDEGREE
    4S/1H0,77X,17HRANDOM NO. SEED =,F12.2/1H0,20X,64HMINIMUM BRANCH DIA
    5METER REQUIRED FOR DETONATION GIVEN BODY HIT =,F10.2,2X,11HMILLIME
    6TERS/1H0,20X,64MINIMUM BRANCH DIAMETER REQUIRED FOR DETONATION GIV
    7EN FUZE HIT =,F10.2,2X,11HMILLIMETERS//)
 2010 FORMAT(1H0,54X,5HBURST,6X,6HBRANCH/55X,6HHEIGHT,4X,8HDIAMETER/
     *       55X,5H(FT.),6X,5H(MM.)/54X,8(1H8),2X,10(1H*)/)
  300 CALL EXIT
      END
```

FIGURE 11. (Continued)

```
      SUBROUTINE CFRQ
C
C     THIS SUBROUTINE DETERMINES CUMULATIVE BURST HEIGHT DISTRIBUTIONS
C
      DIMENSION FREQ(150), CFREQ(150), DET(100)
      COMMON NMUN,SITE,PHISAV,THETA,XINT,IPUNCH,IPRINT,DET
C
C     IF THIS IS FIRST OF FOUR CALLS TO THIS SUBROUTINE, INITIALIZE
C     VARIABLES AND ARRAYS
C
      IF(THETA.NE.0.0) GO TO 200
      DO 50 I=1,150
   50 FREQ(I)=0.0
      XNUM=400.
C
C     DETERMINE INTERVAL ASSOCIATED WITH EACH BURST HEIGHT AND INCREMENT
C     APPROPRIATE ARRAY LOCATION
C
  200 DO 208 I=1,100
      DI=AINT(100.*(DET(I)+.005))/100.
      INC=(DI+XINT-.001)/XINT+1.0
      FREQ(INC)=FREQ(INC)+1.0
  208 CONTINUE
      IF(THETA.NE.270.) RETURN
C
C     IF LAST AZIMUTH ANGLE HAS BEEN PROCESSED, COMPUTE, PRINT, AND PUNCH
C     (IF REQUIRED) HEADER INFORMATION, LIMITS OF BURST HEIGHT INTERVALS,
C     AND CUMULATIVE PERCENTAGES OF DETONATIONS IN INTERVALS
C
  265 CFREQ(1)=FREQ(1)/XNUM*100.0
      XX=.011
      XDATA=0.0
      IX = FREQ(1)
      WRITE(6,324)
      IF(IPRINT.EQ.0) WRITE(6,325)
      WRITE(6,261) IX,CFREQ(1)
      IF(IPUNCH.EQ.0) GO TO 300
      WRITE(10,801) SITE,NMUN,PHISAV
      WRITE(10,802) IX,CFREQ(1)
  300 DO 280 I=2,150
      CFREQ(I)=CFREQ(I-1)+FREQ(I)/XNUM*100.
      XDATA=XDATA+XINT
      IX = FREQ(I)
      WRITE (6,263) XX,XDATA,IX,CFREQ(I)
      IF(IPUNCH.EQ.0) GO TO 310
      WRITE(10,803) XX,XDATA,IX,CFREQ(I)
```

Figure 11. (Continued)

```
 310 [illegible]
     IF(FREQ(I).GE.99.99) GO TO 1000
 280 CONTINUE
1000 RETURN
 261 FORMAT (1H0, 47X, 37HINTERVAL          NO. OF          CUMULATIVE, / 46X,
    *39HFROM      TO      DETONATIONS     PERCENTAGE, / 45X, 6H******,
    *2X, 6H******, 2X, 12H************, 2X, 10H**********, // 45X,
    * 13HGROUND BURSTS,I11,F13.2)
 263 FORMAT(44X,F6.2,F8.2,I11,F13.2)
 324 FORMAT(1H0)
 325 FORMAT(1H1)
 801 FORMAT(4HSITE,A10,5X,15HMUNITION NUMBER,I5,5X,15HELEVATION ANGLE,
    1F7.2,5H DEG.)
 802 FORMAT(20X,I10,F10.2)
 803 FORMAT(2F10.2,I10,F10.2)
     END
```

Figure 11. (Concluded)

SECTION V

SIMULATION MODEL

The BHD Program uses as input from tape the source and terminal X, Y, and Z coordinates and the average diameters of branches surveyed at actual forested sites. It then determines the height (or Z coordinate) for the first branch which is large enough to detonate the munition and which is encountered by randomly selected straight-line trajectories that pass through the environment. The program determines individual burst heights for 100 munition trajectories from each of four azimuth angles (i.e., 0, 90, 180, and 270 degrees) and for any elevation angle specified as input and then prepares a cumulative burst height distribution table for the 400 trajectories.

TABLES AND ARRAYS

No tables are required in the BHD Program, and the arrays are used only for storage of computed data. The DIMENSION and COMMON statements used in Program MAIN are:

```
 DIMENSION XTRAJ(100),YTRAJ(100),ZTRAJ(100),BRANCH(6),R(2),X1(1500)
1,Y1(1500),Z1(1500),X2(1500),Y2(1500),Z2(1500),AVDIAM(1500),DET(100
2),KEPDET(300),KEPDIAM(300)
 DIMENSION STAT(8)
 COMMON NMUN,SITE,PHISAV,THETA,XINT,IPUNCH,IPRINT,DET
 COMMON X1,Y1,Z1,X2,Y2,Z2,AVDIAM
```

and the DIMENSION and COMMON statements used in Subroutine CFRQ are:

```
DIMENSION FREQ(150), CFREQ(150),  DET(100)
COMMON NMUN,SITE,PHISAV,THETA,XINT,IPUNCH,IPRINT,DET
```

The COMMON variables used in Program MAIN and not in Subroutine CFRQ are placed only in Program MAIN to reduce core storage requirements. The system loader entry point is placed at the beginning of BLANK COMMON. BLANK COMMON is larger than the loader, and the core storage requirements are therefore reduced by the size of the loader.

All variables which appear in these DIMENSION and COMMON statements are individually defined in the List of Symbols and Abbreviations (Simulation Model).

DISCUSSION OF SIMULATION

Program MAIN

Program MAIN computes the munition burst heights for each of the munition trajectories being considered. The first executable statements:

```
      CALL EOF(KEOF)
    8 ASSIGN 300 TO KEOF
      READ(5,1000) PHI,XLIMIT,YLIMIT,ZLIMIT,DUMMY,XINT,IPUNCH,IPRINT,
     1BDIAM,FDIAM,BBDIAM,BFDIAM,NMUN,SITE
```

provide a transfer to terminate the program if an end-of-file is read on the input file, and they read the two data cards defining parameters for the case. EOF is a system routine which allows the program to retain control after an end-of-file has been read. The next statements:

```
      PHISAVE=PHI
      THETA=0.
    6 REWIND 15
      PHI=PHISAVE
      DO 1 I=1,100
    1 DET(I)=0.
      DO 2 I=1,300
      KEPDIAM(I)=0.
    2 KEPDET(I)=0.
      ASSIGN 110 TO KEOF
      KEEP=0
```

initialize several program variables and arrays, rewind the binary tape containing the branch input data, and provide a transfer to perform the statistical calculations if an end-of-file is read on the branch input data tape. The next statements:

```
 IF(THETA.NE.0..ANDIPRINT.EQ.1) GO TO 7
 WRITE(6,2000)SITE,NMUN,BDIAM,PHI,FDIAM,THETA,DUMMY,BBDIAM,
1BFDIAM
 IF(IPRINT.EQ.1) GO TO 7
 WRITE (6,2010)
```

write the header information for the case on the output file as called for by the print option. The next statements:

```
7 KTR=0
  PHISAV = PHI
  PHI=-PHI
  XTRAN=0.
  YTRAN=0.
  ITHETA=THETA/90. + 1.
  GO TO(5,3,3,4),ITHETA
3 IF(ITHETA.EQ.3) XTRAN=XLIMIT
  YTRAN=YLIMIT
  GO TO 5
4 XTRAN=XLIMIT
5 FRAD=FDIAM/2.
  BRAD=BDIAM/2.
  AZIMUTH=.017453295*THETA
  PHI=.017453295*PHI
  SINT=SIN(AZIMUTH)
  SINP=SIN(PHI)
  COST=COS(AZIMUTH)
  COSP=COS(PHI)
  CTCP=COST*COSP
  CTSP=COST*SINP
  STSP=SINT*SINP
  STCP=SINT*COSP
  XTRANC=XTRAN*COSP
  XTRANS=XTRAN*SINP
  SPOCP=SINP/COSP
```

perform initial calculations which are used later in the program. These calculations include changing the sign of the elevation angle, determining the translation values required to place all transformed branch coordinates into the first octant, computing the munition fuze and body radii, and computing trignometric functions of the azimuth and elevation angles. The next statements:

```
   XLIMO=XLIMIT/10.
   YLIMO=YLIMIT/10.
   XLIMOO=XLIMIT/100.
   YLIMOO=YLIMIT/100.
   CALL RANSET(DUMMY)
   DO 10 I=1,10
   XCELL=XLIMO*(I-1)
   K=(I-1)*10
   DO 10 J=1,10
   YCELL=YLIMO*(J-1)
   NRNXY=RANF(TRASH)*100.
   NRNX=NRNXY/10
   RNX=NRNX
   RNY=NRNXY-(NRNX*10)
   L=K+J
   XTRAJ(L)=XCELL+RNX*XLIMOO
   YTRAJ(L)=YCELL+RNY*YLIMOO
10 ZTRAJ(L)=ZLIMIT
```

initialize the system uniform random number generator, subdivide the X-Y plane into 100 rectangles, and randomly select munition trajectory X and Y entry coordinates which pass through each of the rectangles (Equations 1, 2, and 3). The sequential number of the layer of branch input data, the number of branches in the layer, the Z coordinate at the bottom of the layer, and the X, Y, and Z coordinates and average diameter of the first branch are read in by the statements:

```
   TOP=ZLIMIT
15 READ(15) LEVEL,NBRAN,BOTTOM,BRANCH,DIAM
   K=0
   BMT=BOTTOM-TOP
```

and the DO LOOP:

```
      DO 30 I=1,NBRAN
      IF(DIAM.LT.BFDIAM)GO TO 20
      K=K + 1
      X1(K)=BRANCH(1)*CTCP+BRANCH(2)*STCP+BRANCH(3)*SINP+XTRANC
      X2(K)=BRANCH(4)*CTCP+BRANCH(5)*STCP+BRANCH(6)*SINP+XTRANC
      Y1(K)=-BRANCH(1)*SINT + BRANCH(2)*COST + YTRAN
      Y2(K)=-BRANCH(4)*SINT + BRANCH(5)*COST + YTRAN
      Z1(K)=-BRANCH(1)*CTSP-BRANCH(2)*STSP+BRANCH(3)*COSP-XTRANS
      Z2(K)=-BRANCH(4)*CTSP-BRANCH(5)*STSP+BRANCH(6)*COSP-XTRANS
      AVDIAM(K)=DIAM
   20 IF(I.EQ.NBRAN)GO TO 30
      READ(15) BRANCH,DIAM
   30 CONTINUE
```

reads in the source and terminal X, Y, and Z coordinates of the remaining branches, eliminates those branches with a diameter less than the minimum branch diameter that will detonate the munition if encountered by the munition fuze, and transforms the X, Y, and Z coordinates of the branch source and terminal ends into a new X'-Y'-Z' coordinate system with the X' axis parallel to the paths of the munition trajectories (Equations 4, 5, and 6). The branch coordinates are rotated through the azimuth angle about the Z axis, translated into the first octant, and rotated through the elevation angle about the new Y' axis.

The trajectory loop is entered to determine the munition/branch encounters, and the statements:

```
      DO 100 I=1,100
      IF(XTRAJ(I)-999999.) 135,100
```

cause the program to proceed to the next munition trajectory if a flag indicates that a detonation has been identified for the munition trajectory in a previous layer. The next statements:

```
135 XPLUS=BMT/SPOCP
    ZOUT=TOP
 35 XOUT=XTRAJ(I) + XPLUS
    IF(XOUT-XLIMIT) 36,36,136
```

compute the X coordinate of the point at which the munition trajectory intersects the plane defining the bottom of the layer (Equation 7). If the munition trajectory exits the layer on the side rather than on the bottom, the reentry coordinates for the munition trajectory are computed by the statements (Equations 9 and 10):

```
136 XOUT=0.
    ZOUT=ZOUT-(XTRAJ(I)-XLIMIT)*SPOCP
    XPLUS=(BOTTOM-ZOUT)/SPOCP
    REPEAT=1
    GO TO 40
 36 ZOUT=BOTTOM
    REPEAT=0
```

If the munition trajectory exits through the bottom of the layer, the reentry coordinates for the next layer are computed (Equations 13, 14, and 15). The reentry coordinates are then transformed so that the munition trajectory paths are parallel to the X′ axis (Equations 17 and 18) by the statements:

```
40 YTRAJP=YTRAJ(I)
   ZTRAJP=-XTRAJ(I)*SINP+ZTRAJ(I)*COSP
```

The branch loop is then entered, and the statements:

```
KEEP=0
DO 70 J=1,K
AVRAD=AVDIAM(J)/2.
FACTRF=(AVRAD+FRAD)**2
FACTRB=(AVRAD+BRAD)**2
```

```
A=Z1(J) - Z2(J)
B=Y2(J) - Y1(J)
C=-Z1(J)**B - Y1(J)**A
LENSQR=A**2 + B**2
LENGTH=SQRT(LENSQR)
```

compute several factors which are used later in the program to determine if the munition body or fuze encounters a branch (Equations 33 through 36). The next statements:

```
    IF(A) 53,141
141 IF(B) 53,142
```

check to determine if both A and B are zero (i.e., the branch centerline is parallel to the munition trajectory). If so, the requirements for Case 1 are satisfied, and the square of the distance in the transformed Y'-Z' plane between the munition trajectory and the branch centerline (D) is computed by the statement (Equation 22):

```
142 D=(Y1(J)-YTRAJP)**2+(Z1(J)-ZTRAJP)**2
```

The value of D is then compared to the square of the sum of the munition body radius and the branch radius (FACTRB) by the statement (Inequality 23):

```
IF(D-FACTRB) 143,143,70
```

If D is larger than FACTRB, the munition body does not encounter the branch, and the program proceeds to the next trajectory/branch combination. If D is smaller than FACTRB, the munition body encounters the branch, and the statement (Inequality 25):

```
143 IF(AVDIAM(J)-BBDIAM) 50,44,44
```

determines if the branch is large enough for detonation by the munition body. If the branch diameter is larger than the minimum diameter required for detonation by the munition body, the statements (Equations 26, 27, and 28):

```
 44 BRDIAM=AVDIAM(J)
    YDET=YTRAJP
    ZDET=ZTRAJP
    IF(X1(J)-X2(J)) 145,145,45
145 XDET=X1(J)
    GO TO 65
 45 XDET=X2(J)
    GO TO 65
```

compute the transformed X′, Y′, and Z′ coordinates of the munition detonation. If the branch diameter is smaller than the minimum diameter required for detonation by the munition body, the statement (Inequality 24):

```
50 IF(D-FACTRF) 44,44,70
```

determines whether the munition fuze encounters the branch. If the munition fuze encounters the branch, the transformed X′, Y′, and Z′ coordinates for the detonation are computed as described earlier. If the munition fuze does not encounter the branch, the program proceeds to the next trajectory/branch combination.

If the values of A and B are not zero, the branch centerline is not parallel to the munition trajectory, and the statements:

```
53 D=((A*YTRAJP+B*ZTRAJP+C)/LENGTH)**2
   IF(D-FACTRB) 153,153,70
```

compute the square of the normal in the transformed Y'-Z' plane from the munition trajectory to the branch centerline (Equation 35) and compare this value (D) to the square of the sum of the munition body radius and the branch radius (FACTRB). If D is larger than FACTRB (Inequality 23), the munition body does not encounter the branch, and the program proceeds to the next trajectory/branch combination. If D is smaller than FACTRB, the statements (Equations 31 and 32 and Inequality 30):

```
153 R(1)=(Y1(J)-YTRAJP)**2+(Z1(J)-ZTRAJP)**2
    R(2)=(Y2(J)-YTRAJP)**2+(Z2(J)-ZTRAJP)**2
    SSQ=AMAX1(R(1),R(2))-D
    IF(SSQ-LENSQR) 154,57,57
```

determine whether the normal in the transformed Y'-Z' plane from the munition trajectory to the branch centerline intersects the branch centerline between the branch ends. If so, the requirements for Case 2 are satisfied, and the statement (Inequality 25):

```
154 IF(AVDIAM(J)-BBDIAM) 56,46,46
```

determines if the branch is large enough for detonation by the munition body. If the branch diameter is larger than the minimum diameter required for detonation by the munition body, the statements (Equations 37 through 41):

```
46 SS2=R(2)-D
   S2=0.
   IF(SS2) 47,47,54
54 S2=SQRT(SS2)
47 S1=LENGTH - S2
   PX=(S2*X1(J) + S1*X2(J))/(S1 + S2)
55 BRDIAM=AVDIAM(J)
   YDET=YTRAJP
   ZDET=ZTRAJP
   XDET=PX
   GO TO 65
```

compute the transformed X', Y', and Z' coordinates of the munition detonation. If the branch diameter is smaller than the minimum diameter required for detonation by the munition body, the statement (Inequality 24):

```
56 IF(D-FACTRF) 46,46,70
```

determines whether the munition fuze encounters the branch. If the munition fuze encounters the branch, the transformed X', Y', and Z' coordinates are computed as described earlier. If the munition fuze does not encounter the branch, the program proceeds to the next trajectory/branch combination.

If the normal in the transformed Y'-Z' plane from the munition trajectory to the branch centerline does not intersect the branch centerline between the branch ends, the requirements for Case 3 are satisfied, and the statements (Inequality 43):

```
 57 MIN=1
    IF(R(2)-R(1)) 164,155,155
164 MIN=2
155 IF(R(MIN)-FACTRB) 156,156,70
```

determine if the munition body encounters either of the branch ends. If the munition body does not encounter either of the branch ends, the program proceeds to the next trajectory/branch combination. If the munition body encounters a branch end, the statement (Inequality 25):

```
156 IF(AVDIAM(J)-BBDIAM) 63,159,159
```

determines if the branch is large enough for detonation by the munition body. If the branch diameter is larger than the minimum diameter required for detonation by the munition body, the statements:

```
159 GO TO(59,61),MIN
 59 PX=X1(J)
    GO TO 55
 61 PX=X2(J)
    GO TO 55
```

select the transformed X′ coordinate and compute the Y′ and Z′ coordinates of the munition detonation (Equations 45, 46, and 47). If the branch diameter is smaller than the minimum diameter required for detonation by the munition body, the statement (Inequality 44):

```
63 IF(R(MIN)-FACTRF) 159,159,70
```

determines whether the munition fuze encounters the branch. If the munition fuze encounters the branch, the transformed X′, Y′, and Z′ coordinates for the detonation are computed as described earlier. If the munition fuze does not encounter the branch, the program proceeds to the next trajectory/branch combination.

In the last statements of the branch loop:

```
65 ZHGT=.003281*(XDET*SINP + ZDET*COSP)
   IF(ZHGT.LT.0.)ZHGT=0.
   KEEP=KEEP + 1
   KEPDET(KEEP)=ZHGT
   KEPDIAM(KEEP)=BRDIAM
70 CONTINUE
```

the Z coordinate for the detonation is computed (Equation 29), and the Z coordinates and corresponding branch diameters are saved for all detonations until all branches in the layer have been processed. If any detonations occur in the layer, the statements:

```
      IF(KEEP) 170,80
  170 THEDET=KEPDET(1)
      THEDIAM=KEPDIAM(1)
      DO 71 LL=1,KEEP
      IF(KEPDET(LL)-THEDET) 71,71,171
  171 THEDET=KEPDET(LL)
      THEDIAM=KEPDIAM(LL)
   71 CONTINUE
```

set the burst height for the trajectory at the maximum Z coordinate of the detonations. The next statements:

```
   75 KTR = KTR + 1
      IF(IPRINT.EQ.0) WRITE(6,1004) THEDET,THEDIAM
      XTRAJ(I)=999999.
   76 DET(I)=THEDET
      GO TO 100
```

increment the counter for the number of air bursts, print the burst height and the branch diameter which caused the detonation if called for by the print option, and flag the munition trajectory so that it will not be considered in the subsequent layers of branches. If no detonations occur for the munition trajectory in the layer of branches, the statements:

```
   80 IF(BOTTOM) 85,180
  180 IF(REPEAT) 85,181
  181 THEDET=0.
      THEDIAM=0.
      XTRAJ(I)=999999.
      GO TO 76
```

set the burst height for the munition trajectory to zero (i.e., a ground burst) if the bottom layer of branches has just been processed and if the munition trajectory does not have to be reentered into the layer. Otherwise, the statements:

```
 85 XTRAJ(I)=XOUT
    ZTRAJ(I)=ZOUT
    IF(REPEAT) 99,100
 99 YTRAJ(I)=RANF(TRASH)*YLIMIT
    GO TO 35
100 CONTINUE
```

set the entry or reentry coordinates to the previously computed values. These will be either the entry coordinates for the next layer or, if the munition trajectory did not exit the layer through the bottom, the reentry coordinates for the current layer. After the 100 munition trajectories have been processed for the layer of branches, the statements:

```
TOP=BOTTOM
GO TO 15
```

define the top of the next layer and provide a transfer to read the branch data for the next layer. After all layers have been processed for the 100 munition trajectories, the statements:

```
110 DO 200 I=1,100
    IF(DET(I)) 201,200
200 CONTINUE
    STAT(1)=STAT(5)=STAT(6)=0.
    GO TO 210
201 CALL BDS(DET,100,STAT)
```

call System Subroutine BDS to compute the mean, variance, and standard deviation for the burst heights of the 100 munition trajectories if any air bursts have occurred. The next statements:

```
210 KTGRD = 100 - KTR
    IF(IPRINT.EQ.1) GO TO 215
    WRITE (6,1007) KTGRD
    WRITE(6,1006)STAT(1),STAT(5),STAT(6)
215 DUMMY=DUMMY+1.
```

print the number of ground bursts and the statistical parameters if called for by the print option. Finally, the statements:

```
CALL CFRQ
THETA=THETA+90.
IF(THETA-360.) 6,8,6
```

call Subroutine CFRQ to determine the cumulative distribution of the burst heights and increment the azimuth angle. If all azimuth angles have not been processed, a transfer is made to process 100 munition trajectories for the next azimuth angle. Otherwise, a transfer is made to read in data for the next case. The statements:

```
300 CALL EXIT
    END
```

terminate the program.

Subroutine CFRQ

The purpose of this subroutine is to compute, print, and optionally punch the cumulative burst height distribution for the munition and elevation angle being considered. The first executable statements:

```
    IF(THETA.NE.0.0) GO TO 200
    DO 50 I=1,150
 50 FREQ(I)=0.0
    XNUM=400.
```

initialize the cumulative burst height distribution array and set the total number of data points if this call to the subroutine is for the first azimuth angle (i.e., 0 degrees). The DO LOOP:

```
200 DO 208 I=1,100
    D1=AINT(100.*(DET(I)+.005))/100.
    INC=(D1+XINT-.001)/XINT+1.0
    FREQ(INC)=FREQ(INC)+1.0
208 CONTINUE
```

determines the interval associated with each burst height and increments the appropriate array location. If the 100 munition trajectories for the last azimuth angle (i.e., 270 degrees) are being processed, the statements:

```
    IF(THETA.NE.270.) RETURN
265 CFREQ(1)=FREQ(1)/XNUM*100.0
    XX=.011
    XDATA=0.0
    IX = FREQ(1)
    WRITE(6,324)
    IF(IPRINT.EQ.0) WRITE(6,325)
    WRITE(6,261) IX,CFREQ(1)
    IF(IPUNCH.EQ.0) GO TO 300
    WRITE(10,801) SITE,NMUN,PHISAV
    WRITE(10,802) IX,CFREQ(1)
```

print the header information and compute and print the limits of the first burst height interval, the number of detonations in the interval, and the cumulative percentage of detonations in the interval (Equation 48). These values are also punched (or written on Logical File 10) if called for by the punch option. Finally, the statements:

```
300 DO 280 I=2,150
    CFREQ(I)=CFREQ(I-1)+FREQ(I)/XNUM*100.
    XDATA=XDATA+XINT
    IX = FREQ(I)
```

```
      WRITE (6,263) XX, XDATA,IX,CFREQ(I)
      IF(IPUNCH.EQ.0) GO TO 310
      WRITE(10,803) XX,XDATA,IX,CFREQ(I)
  310 XX=XX+XINT
      IF(CFREQ(I).GE.99.99) GO TO 1000
  280 CONTINUE
 1000 RETURN
      END
```

compute, print, and optionally punch the remaining burst height intervals, numbers of detonations, and cumulative percentages of detonations and stop when the cumulative burst height value reaches 100 percent.

(The reverse of this page is blank)

INITIAL DISTRIBUTION

USAMSAA (AMXRD-AT/MR. BUTLER)	1
USAMSAA (AMXRD-XSR/MR. EISSNER)	1
USAMSAA (AMXRD-AT/MR. GROVES)	1
USAMSAA (AMXRD-AT/MR. MYERS)	1
USAT&EC (R&D METHOD DIV/MR. WISE)	1
US ARMY ENG WTRWAY EXPER STA (MR. WEST)	1
NWL (CODE KRT-2/MR. CLODIUS)	1
NOL (SYS EVAL DIV (WO)-NS/MR. LUTTRELL)	1
PICATINNY ARS (SMUPA-DW6/MR. GAYDOS)	1
PICATINNY ARS (SMUPA-SM/MR. BEVELOCK)	1
PICATINNY ARS (SMUPA-DF1/MR. HAUPTMAN)	1
PICATINNY ARS (SMUPA-VC5/MR. KOVACS)	1
PICATINNY ARS (SMUPA-DW6/MR. MATZKOWITZ)	1
PICATINNY ARS (SMUPA-DF1/MR. SCHULMAN)	1
FRANKFORD ARS (MR. SIKRA)	1
ADTC (DGWT/CPT GRAZIADEI)	1
AFSC (SCSMM)	1
AFSC (SDW)	1
AFATL (DLRD/MR. COLLINS)	1
AFATL (DL)	1
AFATL (DLGP)	1
AFATL (DLOSL)	2
AFATL (DLYW)	10
CORNELL AERO LAB (MR. EUSANIO)	1
CORNELL AERO LAB (MR. KELLER)	1
CORNELL AERO LAB (MR. WOZER)	1
HARRY DIAM LABS (AMXDO-SA/MR. ROTKIN)	1
OKLAHOMA STATE UNIVERSITY	1
BOOZ, ALLEN APPLIED RESEARCH, INC.	6
DDC	2
ASD (ENYS)	1
AUL (AUL-LSE-70-239)	1
HQ PACAF (IGY)	1

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Booz, Allen Applied Research, Inc. P.O. Box 1797 Eglin Air Force Base, Florida 32542	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

BURST HEIGHT DISTRIBUTION COMPUTER MODEL, VOLUME II. ANALYST MANUAL

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final Report - January 1971 through January 1972

5. AUTHOR(S) (First name, middle initial, last name)

Donald E. Cudney
David O. Fraser

6. REPORT DATE	7a. TOTAL NO OF PAGES	7b. NO OF REFS
January 1972	91	---

8a. CONTRACT OR GRANT NO F08635-71-C-0093	9a. ORIGINATOR'S REPORT NUMBER(S) ----
b. PROJECT NO 9134	
c. Task No. 01	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d. Work Unit No. 001	AFATL-TR-72-16, Volume II

10. DISTRIBUTION STATEMENT

Distribution limited to U. S. Government agencies only; this report documents test and evaluation; distribution limitation applied January 1972. Other requests for this document must be referred to the Air Force Armament Laboratory (DLYW), Eglin Air Force Base, Florida 32542.

11. SUPPLEMENTARY NOTES Available in DDC	12. SPONSORING MILITARY ACTIVITY Air Force Armament Laboratory Air Force Systems Command Eglin Air Force Base, Florida 32542

13. ABSTRACT

The Burst Height Distribution (BHD) Program described in this report was designed to compute and display burst height distributions for munitions aerially delivered into forest environments. The program uses as input the source and terminal X, Y, and Z coordinates and the average diameters of branches surveyed at actual forested sites, and the munitions travel along straight-line trajectories which are randomly selected. Burst heights are computed for those trajectories which encounter branches large enough to detonate the munition, and after 400 trajectories are examined (100 from each of four azimuth angles), the cumulative burst height distribution for the munition and elevation angle is computed, printed, and optionally punched as output. The computer program was specifically designed for the Control Data Corporation 6600 computer system at Eglin Air Force Base, Florida.

DD FORM 1 NOV 65 1473

UNCLASSIFIED
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Mathematics and statistics						
Computer programming						
Monte Carlo modeling						
Weapons effects						
Aerially delivered munitions						