UNCLASSIFIED

AD NUMBER

AD892420

LIMITATION CHANGES

TO:
Approved for public release; distribution is unlimited.

FROM:
Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; DEC 1971. Other requests shall be referred to Air Force Armament Laboratory, DLYD, Eglin AFB, FL 32542.

AUTHORITY

afatl per dtic form 55

THIS PAGE IS UNCLASSIFIED
WEAPON OPTIMIZATION TECHNIQUES

MARTIN MARIETTA CORPORATION

TECHNICAL REPORT AFATL-TR-71-170

DECEMBER 1971

Distribution limited to U. S. Government agencies only; this report documents test and evaluation of military weapons; distribution limitation applied December 1971. Other requests for this document must be referred to the Air Force Armament Laboratory (DLYD), Eglin Air Force Base, Florida 32542.

AIR FORCE ARMAMENT LABORATORY

AIR FORCE SYSTEMS COMMAND • UNITED STATES AIR FORCE

EGLIN AIR FORCE BASE, FLORIDA
Weapon Optimization Techniques

William R. Day
George W. Brooks
William S. Strickland

Distribution limited to U. S. Government agencies only; this report documents test and evaluation of military weapons; distribution limitation applied December 1971. Other requests for this document must be referred to the Air Force Armament Laboratory (DLYD), Eglin Air Force Base, Florida 32542.
FOREWORD

This report summarizes the investigations conducted by the Martin Marietta Corporation, Orlando, Florida, during the period August 1968 through September 1969 under Contract F08635-68-C-0010. This contract was administered by the Air Force Armament Laboratory, Air Force Systems Command, Eglin Air Force Base, Florida, and was funded under project 2543. Air Force Armament Laboratory program monitors were Mr. Massey B. Valentine and Mr. John W. Barter III (DLYD). Contractor report number is OR 10,247.

The Martin Marietta Corporation task leaders for this program were Mr. E. L. Scherich, Mr. T. Kitchin, and Mr. W. R. Day. Other contributing Martin Marietta personnel included Mr. G. W. Brooks, Mr. W. S. Strickland, Mr. D. W. Coleman, and Mrs. T. Linthicum. This report was prepared by Messrs. W. R. Day, G. W. Brooks, and W. S. Strickland under the direction of Mr. W. R. Porter, Chief of the Analysis Section and Mr. C. A Borcher, Manager of the Ordnance Department.

Acknowledgement is made to Mr. M. Valentine, Mr. J. Barter, Capt. J. Duffy, and Lt. P. Smith of the Air Force Armament Laboratory for their valuable technical assistance and guidance in the performance of this program.

This technical report has been reviewed and is approved.

Thomas P. Christie

THOMAS P. CHRISTIE
Director, Weapon Systems Analysis Division
ABSTRACT

This investigation established the techniques to evaluate the lethality of blast/fragmentation, continuous rod, and submissile warheads when deployed against airborne targets. A computer program was developed which is capable of optimizing warhead design parameters and terminal encounter conditions or of parametrically examining the effect of design and encounter variables on lethality against any air target. A number of mathematical models to permit evaluation of linear shaped charge warheads were also developed. The formulations of required computer routines can be completed and integrated into the air optimization computer program during future contract efforts. In addition, a second computer program was developed to compute the lethal area of flechette-projecting warheads deployed against standing, prone or foxholed troop concentrations. The program simulates the actual flechette trajectories to properly account for the effects of gravity, and establishes the lethal area by integrating kill probability over the impact pattern area.

Sections I through IV contain the air encounter optimization computer program. Sections V through VIII contain the documentation for the flechette lethal area program, and modifications to a previously developed surface targets weapon optimization lethal area program.

Distribution limited to U. S. Government agencies only; this report documents test and evaluation of military weapons; distribution limitation applied December 1971. Other requests for this document must be referred to the Air Force Armament Laboratory (DLYD), Eglin Air Force Base, Florida 32542.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Air-to-Air Weapon Optimization Computer Program</td>
<td>1</td>
</tr>
<tr>
<td>II</td>
<td>Air Optimization Computer Program Capabilities</td>
<td>4</td>
</tr>
<tr>
<td>III</td>
<td>Encounter Conditions and Methodology</td>
<td>15</td>
</tr>
<tr>
<td>IV</td>
<td>Conclusions and Recommendations</td>
<td>21</td>
</tr>
<tr>
<td>V</td>
<td>Flechette Lethal Area and Lethal Area Optimization Programs</td>
<td>22</td>
</tr>
<tr>
<td>VI</td>
<td>Flechette Lethal Area Program</td>
<td>25</td>
</tr>
<tr>
<td>VII</td>
<td>Lethal Area Optimization Program</td>
<td>30</td>
</tr>
<tr>
<td>VIII</td>
<td>Conclusions and Recommendations</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td><strong>Appendixes</strong></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>Utilization Guide</td>
<td>35</td>
</tr>
<tr>
<td>II</td>
<td>Derivation of Basic Methodology</td>
<td>60</td>
</tr>
<tr>
<td>III</td>
<td>Flow Diagram Air-to-Air Weapon Optimization Computer Program M8054</td>
<td>108</td>
</tr>
<tr>
<td>IV</td>
<td>Formulation of Linear Shaped Charge Munition for Effectiveness Evaluation</td>
<td>113</td>
</tr>
<tr>
<td>V</td>
<td>Geometry and Flow Diagram</td>
<td>121</td>
</tr>
<tr>
<td>VI</td>
<td>Runge Kutta Solutions of Equations of Motion and Techniques for</td>
<td>126</td>
</tr>
<tr>
<td></td>
<td>Determining Polygon Parameters</td>
<td></td>
</tr>
<tr>
<td>VII</td>
<td>Grid Point Test</td>
<td>129</td>
</tr>
<tr>
<td>VIII</td>
<td>Utilization Report - Flechette Computer Program</td>
<td>130</td>
</tr>
<tr>
<td>IX</td>
<td>Utilization Report - Fragmentation/Blast Warhead Optimization Program</td>
<td>139</td>
</tr>
<tr>
<td>References</td>
<td></td>
<td>149</td>
</tr>
<tr>
<td>Figure</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1</td>
<td>Program Flow Plan</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Basic Geometry of Each Encounter.</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Chemical Fuzing Concept.</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>Spherical Fuzing Concept.</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>Double Cone Fuzing Concept.</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>Fragmentation Warhead.</td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td>Continuous Rod Warhead.</td>
<td>9</td>
</tr>
<tr>
<td>8</td>
<td>Submissile Warhead.</td>
<td>9</td>
</tr>
<tr>
<td>9</td>
<td>Target Vulnerability Characteristics.</td>
<td>19</td>
</tr>
<tr>
<td>10</td>
<td>Air-to-Air Weapon Optimization Program Flow Plan.</td>
<td>23</td>
</tr>
<tr>
<td>11</td>
<td>Sample Warhead Geometry</td>
<td>25</td>
</tr>
<tr>
<td>12</td>
<td>Trajectory Geometry</td>
<td>27</td>
</tr>
<tr>
<td>13</td>
<td>Sample Impact Pattern Geometry.</td>
<td>28</td>
</tr>
<tr>
<td>I-1</td>
<td>Basic Geometry of Each Encounter</td>
<td>36</td>
</tr>
<tr>
<td>I-2</td>
<td>Target Subrange Geometry</td>
<td>51</td>
</tr>
<tr>
<td>I-3</td>
<td>Geometry of Fragment Vulnerability Data Options</td>
<td>52</td>
</tr>
<tr>
<td>II-1</td>
<td>Basic Geometry of Each Encounter</td>
<td>61</td>
</tr>
<tr>
<td>II-2</td>
<td>Typical Case of Fuzing</td>
<td>67</td>
</tr>
<tr>
<td>II-3</td>
<td>Illustrative Case When Lobe 4 Intersects the Position of the Jammer</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>Specified by the Vector Jammer</td>
<td></td>
</tr>
<tr>
<td>II-4</td>
<td>Definition of Input Variables Describing a Warhead with Three Angular</td>
<td>71</td>
</tr>
<tr>
<td></td>
<td>Fragment Spray Zones and Two Fragment Velocity Groups</td>
<td></td>
</tr>
<tr>
<td>II-5</td>
<td>Geometry Used in Subroutine &amp;D</td>
<td>72</td>
</tr>
<tr>
<td>II-6</td>
<td>Geometry When Blast Wave Reaches a Blast Center</td>
<td>75</td>
</tr>
<tr>
<td>II-7</td>
<td>Geometry of Direct Hit</td>
<td>78</td>
</tr>
<tr>
<td>II-8</td>
<td>Geometry of Fragmentation When VMBAR &gt; VFBAR.</td>
<td>81</td>
</tr>
<tr>
<td>II-9</td>
<td>Geometry of Fragmentation When VMBAR &lt; VFBAR.</td>
<td>84</td>
</tr>
<tr>
<td>II-10</td>
<td>Shielding Geometry</td>
<td>87</td>
</tr>
<tr>
<td>II-11</td>
<td>Computation of RHOD</td>
<td>90</td>
</tr>
<tr>
<td>II-12</td>
<td>Geometry of Fragment Vulnerability Data Options</td>
<td>97</td>
</tr>
<tr>
<td>II-13</td>
<td>Vector Diagram for Continuous Rod Routine</td>
<td>102</td>
</tr>
<tr>
<td>III-1</td>
<td>General Flow Diagram for Air-to-Air Weapon Optimization Program</td>
<td>109</td>
</tr>
<tr>
<td>IV-1</td>
<td>Fragment Spray Representation</td>
<td>116</td>
</tr>
<tr>
<td>IV-2</td>
<td>Computation of RHOD</td>
<td>118</td>
</tr>
</tbody>
</table>
LIST OF FIGURES Concluded

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>V-1</td>
<td>Sample Warhead Geometry</td>
<td>122</td>
</tr>
<tr>
<td>V-2</td>
<td>Sample Impact Pattern Geometry</td>
<td>123</td>
</tr>
<tr>
<td>V-3</td>
<td>General Flow Diagram for Flechette Lethal Area Program</td>
<td>124</td>
</tr>
<tr>
<td>VII-1</td>
<td>Grid Test Geometry</td>
<td>129</td>
</tr>
</tbody>
</table>

LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Combinations of Weapon Parameter Values</td>
<td>20</td>
</tr>
<tr>
<td>I-I</td>
<td>Optimization or Parametric Data</td>
<td>39</td>
</tr>
<tr>
<td>I-II</td>
<td>Variables for Optimization or Parameterization</td>
<td>40</td>
</tr>
<tr>
<td>I-III</td>
<td>Warhead Data</td>
<td>41</td>
</tr>
<tr>
<td>I-IV</td>
<td>Intercept Conditions</td>
<td>44</td>
</tr>
<tr>
<td>I-V</td>
<td>Fuzing</td>
<td>45</td>
</tr>
<tr>
<td>I-VI</td>
<td>Target</td>
<td>47</td>
</tr>
<tr>
<td>I-VII</td>
<td>Continuous Rod Target Data</td>
<td>53</td>
</tr>
<tr>
<td>I-VIII</td>
<td>Bomblet Data</td>
<td>55</td>
</tr>
<tr>
<td>I-IX</td>
<td>Miscellaneous</td>
<td>56</td>
</tr>
<tr>
<td>I-X</td>
<td>Error Diagnostics</td>
<td>57</td>
</tr>
<tr>
<td>VIII-I</td>
<td>Output Parameters</td>
<td>132</td>
</tr>
<tr>
<td>VIII-II</td>
<td>Flechette Lethal Area Program</td>
<td>133</td>
</tr>
<tr>
<td>VIII-III</td>
<td>Multiple Runs (Clobber Data)</td>
<td>137</td>
</tr>
<tr>
<td>VIII-IV</td>
<td>MACH Storage</td>
<td>138</td>
</tr>
<tr>
<td>IX-I</td>
<td>Optimization or Parametric Data</td>
<td>139</td>
</tr>
<tr>
<td>IX-II</td>
<td>Variable Names</td>
<td>141</td>
</tr>
<tr>
<td>IX-III</td>
<td>Warheads with Constant Charge to Mass Ratio</td>
<td>141</td>
</tr>
<tr>
<td>IX-IV</td>
<td>Warheads with Varying Charge to Mass Ratio-Optimization or Parametric Runs for Spherical Fragment Mass Only</td>
<td>143</td>
</tr>
<tr>
<td>IX-V</td>
<td>Target Data</td>
<td>144</td>
</tr>
<tr>
<td>IX-VI</td>
<td>Multiple Runs</td>
<td>147</td>
</tr>
</tbody>
</table>

vii
(The reverse of this page is blank.)
SECTION I
AIR-TO-AIR WEAPON OPTIMIZATION COMPUTER PROGRAM

This report summarizes a 12 month research program to develop an Air-to-Air Weapon Optimization Computer Program capable of optimizing warhead design parameters and terminal target-interceptor encounter parameters to maximize lethality of various warhead kill mechanisms against tactical air targets.

1. PROGRAM OBJECTIVES

Overall objective of this program was to perform necessary studies and analyses to develop mathematical models and techniques for simulating air-to-air weapons and optimizing the characteristics of these weapons to obtain maximum systems effectiveness, and then to develop computer programs capable of meeting these objectives. Specific tasks of the program were to:

1. Investigate the feasibility of optimizing and evaluating fragmentation, blast, cluster, and continuous rod warheads against tactical air targets;

2. Develop air target models which provide accurate target representation consistent with the warhead types to be evaluated;

3. Formulate fuzing models which simulate hollow cone, proximity, double cone, and smart fuzes, and which are an integral part of the computer program;

4. Provide for the evaluation of relevant system errors such as fuzing errors, velocity errors, and attack direction errors;

5. Combine the above objectives in simulating a warhead/target encounter, including the interceptor flight from immediately following release and separation to detonation, as well as the end game kinematics;

6. Investigate the possibility of adding a coverage model to one of the computer programs previously developed;

7. Maintain the lethal area and kill probability optimization programs developed under present and previous effects;

8. Develop a flechette model for inclusion in the previously developed lethal area program, giving proper treatment to gravity effects on the flight of flechettes launched from high altitudes.

Computer programs developed during this study must be compatible with the CDC 6600 equipment located at Eglin Air Force Base, Florida.

2. PROGRAM APPROACH AND RESULTS

a. Approach

The program approach to satisfy the objectives listed above consisted of three major tasks. The first was to develop the methodology required for preparing the Air-to-Air Weapon Optimization Computer Program. This phase is discussed in Sections I through IV, and the general outline of the program for developing the Air-to-Air Weapon Optimization Computer Program is presented in Figure I. The efforts expended for developing the area coverage program and the flechette program are discussed in Sections V through VIII.
Formulation of Blast Frag W/H Model

Formulation of Target and Encounter Geometry Models

Coding of Mathematical Model

Program Checkout & Verification

Air-to-Air Weapon Optimization Techniques Computer Program

Formulation of Cluster W/H Model

Formulation of Target and Encounter Geometry Models

Coding of Mathematical Model

Program Checkout & Verification

Formulation of Continuous Rod W/H Model

Formulation of Target and Encounter Geometry Models

Coding of Mathematical Model

Program Checkout & Verification

Adaptation of Programs to Eglin AFB CDC 6600

Formulation of Math Model for Evaluation of Linear Shaped Charge W/H

Delivery of Report & Recommendations

Final Report & Recommendations

Figure 1. Program Flow Plan
Liaison was maintained with the Air Force Armament Laboratory (AFATL) and other government agencies, and extensive literature surveys were conducted to ensure that state of the art methodology was incorporated into the program. Since previous work indicated that the method of steepest ascent was a satisfactory technique for determining optimal parameter values, this method was used in developing the Air-to-Air Weapon Optimization Computer Program (References 1 and 2). Other techniques which were investigated include linear programming, accelerated gradient, and differential programming. In addition, finite step parameterization techniques are used for the parameter sensitivity studies. These procedures were followed because they make the program flexible, accurate, and efficient.

b. Results

The major accomplishment of this phase of the contract was development of the Air-to-Air Weapon Optimization Computer Program with the capability to evaluate not only blast/fragmentation warheads, but continuous rod and submunition warheads as well. In addition, some basic formulations were completed toward increasing program capabilities to include linear shaped charge kill mechanisms. The developed program provides the following advantages:

1. The model is highly efficient for optimization or computation of warhead lethality with minimum computer time;
2. State-of-the-art technology is employed in the weapon evaluation model;
3. Mathematical definition of warhead performance data and target vulnerability data is compatible with current government agency weapon and target descriptions;
4. The program is constructed in modular form to permit future modifications with minimal effort.

3. RECOMMENDATIONS

As a result of this program, it is recommended that follow-on weapon optimization technique programs be conducted to:

1. Complete formulation of the mathematical model for evaluating linear shaped charge warheads and add this capability to the Air-to-Air Weapon Optimization Computer Program;
2. Develop a composite air-to-surface computer program which combines the missile launching and flight trajectory simulation program with an end game effectiveness evaluation program so that the output from the trajectory routines will provide input for lethality evaluation;
3. Formulate and code a computer program for the lethality evaluation of aerially emplaced mines against mobile personnel and materiel targets;
4. Formulate and develop computer routines for evaluation of variable charge-to-metal ratio warheads (catastrophic structural kill criteria);
5. Develop a composite aerial dogfight-to-end game lethality evaluation computer program to provide flight simulations from target acquisition to kill probability computation;
6. Develop computer routines to predict dispersion patterns of lifting body flechettes and submunitions for incorporation into present effectiveness programs.

4. REPORT ORGANIZATION

The remainder of this report discusses in detail the capabilities, methodology, and input format used in the program. Section II describes the capabilities of the Air-to-Air Weapon Optimization Computer Program developed under this contract, and also discusses means for describing the various warheads, targets, and fuzes which can be evaluated with this program. Section III describes encounter conditions and basic program methodology. Appendices I through IV contain the utilization guide, derivations of basic methodology, program general flow diagrams, and formulation of linear shaped charge munition for effectiveness evaluation as applicable to Sections II through IV.
SECTION II

AIR OPTIMIZATION COMPUTER PROGRAM CAPABILITIES

This section outlines major features of the Air-to-Air Weapon Optimization Computer Program, and presents detailed discussion of warhead, target, and fuze characteristics.

1. PROGRAM APPLICATION

This program utilizes a Monte Carlo random number technique to select various encounter geometry and fuzing parameters from specified distributions of each parameter. In so doing, the program simulates a large number of random warhead/target encounters in calculating either an average or single-shot kill probability (\( P_{SSK} \)).

The program can be used in either an optimization or parametric mode of operation. For the optimization mode, warhead design and encounter parameters are varied for each \( P_{SSK} \) calculation by a method of steepest ascent. Thus, the combination of parameter values which maximizes kill probability is readily determined. In the parametric mode, the parameters are varied at specified intervals until the \( P_{SSK} \) for all desired combinations of parameter values has been computed.

2. BASIC PROGRAM LOGIC

For each encounter which is evaluated, the program must consider a large number of variables, the more significant of which are sequentially described below.

a. Target-Detonation Point Geometry

(1) Encounter Geometry

The basic encounter geometry is illustrated by Figure 2, consisting of three left handed coordinate systems. The first is the reference coordinate system fixed to the target with unprimed \( X, Y, \) and \( Z \) axes. The \( Y \) axis is directed along the target velocity vector (\( V_{T BAR} \)), the \( Z \) axis is up and the \( X \) axis is out the left side of the target.

The second is the interceptor missile coordinate system with \( X^1, Y^1, \) and \( Z^1 \) axes. The \( Y^1 \) axis is directed opposite to the direction of the missile velocity (\( V_{M BAR} \)), the \( Z^1 \) axis is \( Y \) with respect to the missile and the \( X^1 \) axis is out the right side of the missile. The third is the relative coordinate system with \( X'' \), \( Y'' \), and \( Z'' \) axes. The direction of the \( Y'' \) axis is defined by the vector difference of \( V_{M BAR} \) and \( V_{T BAR} \). CPSI and GAMMA are the azimuth and elevation approach angles between the target (reference) coordinate system and the missile coordinate system. \( \Psi \) and \( \Lambda \) are functions of CPSI and GAMMA and are the azimuth and elevations between the reference coordinate and the relative coordinate system. The origin of the relative coordinate system is also fixed to the target and thus \( X'' \) and \( Z'' \) axes are defined by rotation of the unprimed coordinate system through PSI and LAMBDA.

Azimuth and elevation encounter aspect angles are selected at random from a table of distribution values in which up to 36 zones and the probability of the encounter aspect occurring within each zone are defined. Within each zone, probability density is assumed to be random-uniform, i.e., all angles are equally likely.
(2) Target Glitter Points

To simulate what the homing radar sees, the target is represented by a set of up to 20 reflection points (homing glitter points). Up to 10 of these glitter points may be visible for any given encounter aspect zone. Tabular values for up to 10 glitter points may be input for each of the 36 aspect zones. To compute the radar target centroid, the coordinates of the visible homing glitter points are averaged.

(3) Aim Point

The aim point to which the guidance seeker homes may be specified in three ways:

1. Input values $X$, $Y$, $Z$ in the unprimed (target) coordinate system;
2. Input values $X''$, $Z''$ in the primed (relative) coordinate system;
3. Computed radar centroid of glitter points.

(4) Guidance Error

Three options are available for determining the guidance error:

1. Discrete miss distances in the $X''$ and $Z''$ directions may be specified as constant for all encounters;
2. A standard deviation (sigma) may be specified and the miss distances chosen by Monte Carlo random selection;
3. A fixed miss distance $R = \sqrt{X''^2 + Z''^2}$.

(5) Fuzing

As with homing glitter points, up to 20 fuzing glitter points may be specified with up to 10 visible from any given aspect angle. Fuzing occurs when a fuzing point passes through the appropriate fuzing surface. Four options are available for defining the fuzing surfaces:
1 Single cone fuze is activated when the target glitter point passes through the surface of a cone with the axis of symmetry along the longitudinal axis of the interceptor and the apex at the guidance seeker, Figure 3;

2 Spherical or proximity fuze is activated when the target glitter point passes through the surface of a sphere with center at the guidance seeker, Figure 4;

3 Double cone fuze computes the time between target penetration of two conical surfaces as in 1 above. Fuze is activated at the second cone provided the time between cones is between specified minimum and maximum fuze gate times, Figure 5;

4 Fuze jammer is activated when target glitter point passes through any of a family of conical surfaces representing radar side lobes provided certain signal strength and range criteria are met. Combinations of and variations to these basic fuzing routines may be made by input variables and minimal re-programming effort.

(6) Delay Time

Detonation of the warhead occurs at a given delay time after fuzing. Three options are available to specify the delay time:

1 A fixed mean delay time and a standard deviation may be specified with random selection for each encounter;

2 A variable delay time may be specified in tabular form as a function of relative velocity;

3 An optimum delay time may be computed for each encounter which causes fragments from a given polar zone to hit the target aim point.

At this point the warhead coordinate system is rotated to account for pitch and yaw standard deviation, and the detonation point location is corrected to account for the distance between the seeker and the warhead centroid.

Figure 3. Conical Fuzing Concept
Figure 4. Spherical Fuzing Concept

Figure 5. Double Cone Fuzing Concept
b. Kill Probability Computation

The kill probability is computed by iteration routines, as follows.

1. Blast Computations

Blast ellipsoids about the target are defined which determine the maximum radius at which blast overpressure at the target surface will be sufficient for a kill. If warhead detonation occurs within the blast ellipsoid, the $P_K$ is 1.0, otherwise the $P_K$ is 0. In computing blast kill, a routine is included to properly account for the finite velocity of the blast wave.

2. Direct Hit Computations

Direct hit ellipsoids are defined to model physical size and shape of the target aircraft. If the warhead trajectory passes through any portion of a direct hit ellipsoid prior to detonation, kill probability due to direct hit is a prespecified input value less than or equal to 1.0.

3. Fragmentation Computations - Figure 6

The dynamic density, striking angle, and velocity are computed for fragments striking vulnerable target components and vulnerable area is determined from tabular input data as a function of fragment mass, velocity, and striking angle. Shielding ellipsoids are input and are considered to be impenetrable to fragments. Kill probability is computed by the formula

$$P_K = 1 - \exp(-\alpha A_v)$$

where:

- $\alpha$ is dynamic fragment density,
- $A_v$ is vulnerable area.

Figure 6. Fragmentation Warhead
(4) Continuous Rod Computations - Figure 7

Calculations are made to determine each spar i.e., load carrying structural member of the fuselage, empennage, or wing, that is hit by a continuous rod. Dynamic rod velocity and striking angle are compared to input rod rejection criteria to determine if a spar is cut. Each spar cut contributes a percentage toward a kill. If spar cut values add up to 1.0, the target \( P_K \) is 1.0. If total spar cut values are less than 1.0, target \( P_K \) may be zero or the sum of spar cut values.

![Figure 7. Continuous Rod Warhead](image)

(5) Bomblet or Submissile Computations - Figure 8

Bomblets are ejected perpendicular to missile flight path and the equations of motion for the actual trajectories are integrated for an input bomblet delay time. Effects of gravity and air drag are considered to establish a plane surface where all of the bomblets are fuzed. Individual bomblet detonation points are computed by a random selection of delay times. Blast, direct hit, and fragmentation kill are evaluated for each bomblet.

![Figure 8. Submissile Warhead](image)
Total kill probability for all kill mechanisms is computed to complete one sample. This entire process is repeated many times with a different set of random variables for each sample, and an average value for all samples is computed to establish single-shot kill probability.

In these computations, a minimum of 50 samples should be employed to attain comparative data. One hundred or more samples are recommended for a valid statistical sample of warhead lethality.

3. WARHEAD TYPES

This paragraph describes the three warhead types which can be evaluated with the Air-to-Air Weapon Optimization Computer Program. Characteristics required to define the particular munition models, i.e., fragmentation/blast, continuous rod, and submunition warheads, are also presented.

a. Fragmentation/Blast

Any fragmentation/blast warhead which has an axis of symmetry coincident with its line of flight can be accurately defined using the mathematical techniques incorporated in the Air-to-Air Weapon Optimization Computer Program. Fragmentation characteristics are defined in up to 36 fragment polar zones, each of which may have up to 10 classes of fragments. The following data must be specified to define each fragment class:

1. Fragment initial velocity;
2. Fragment mass;
3. Fragment average presented area or K factor, where K is defined as (presented area divided by the 2/3 power of the mass);
4. Number of fragments;
5. Fragment shape or drag coefficient versus Mach number curve.

For more precise representation of fragment velocities, a provision is incorporated into the program to allow for a standard deviation of static fragment velocities. Also, to more accurately describe the warhead performance characteristics, provision for missile structural shielding of the warhead is incorporated into the program. This shielding allows the user to simulate wings, radar, and other components, which obstruct fragmentation effects of the warhead.

The blast characteristics for each warhead are defined in terms of blast centers associated with one or more blast ellipsoids about each of up to 10 vulnerable target areas. For each sample, detonation points of the warhead are checked to determine if they fall within any ellipsoid at the time the blast wave reaches the blast center. If so, the target is considered killed; if not, there is no target damage due to blast.

b. Continuous Rod

The capability to evaluate a continuous rod warhead is included in the Air-to-Air Weapon Optimization Computer Program. This portion may be used to evaluate continuous rod effects either alone or in conjunction with fragmentation and blast characteristics.

The following data define a continuous rod warhead:

1. Number of continuous rods in the warhead;
2. Initial flyoff angle for rods, measured from missile centerline;
3. Rod density;
4. Rod cross sectional width;
5. Maximum rod radius;
6. Initial number of breaks in the rods;
7. Number of breaks in the rod at maximum rod diameter;
The maximum number of diameters of the continuous rods;
Initial static rod flyoff velocity.

The above data along with system and fuze data which are described in paragraphs 4 and 5 below can be used in conjunction with the definition of the target encountered and its vulnerability data to evaluate and optimize lethality of the weapon system under consideration.

c. Cluster (Fragmentation/Blast Submissile)

The cluster warhead evaluation routine is capable of evaluating any fragmentation and/or blast munition that has at least one axis of symmetry. The formulation enables accurate simulation of an air encounter between a bomblet pattern and an airborne target. A trajectory integration subroutine is included to simulate the actual dispersion of bomblets according to bomblet ejection velocity and missile velocity at the time of deployment. Bomblet trajectories are computed, including gravity effects, by numerically integrating the equations of motion. The program calculates the size of the pattern, location of pattern, and bomblet velocity at detonation. Fragmentation and blast effects from the bomblet are computed as for unitary warheads which were previously discussed in paragraph 3a above.

With the cluster warhead model, it is possible to optimize or parameterize the range from interceptor to target for a proximity fuze, in order to establish the desired release point of bomblets. This portion of the program can also be used to optimize or parameterize the bomblet detonation delay time, bomblet ejection velocity, fuzing cone half angle, radius of spherical proximity fuze, and fragmentation characteristics of the bomblets.

4. TARGET REPRESENTATION

In formulating an effectiveness computer program, proper representation of targets is a vital consideration in obtaining accurate results and minimizing computer time. The vulnerability data used to define targets are a function of the type warhead to be evaluated. This paragraph discusses target descriptions used in conjunction with fragmentation/blast, continuous rod, and cluster warheads.

a. Vulnerability to Fragmentation

The probability of achieving a conditional fragmentation kill (probability of killing the target given a fragment hit) is a function of target characteristics, fragment striking velocity, fragment mass, and the azimuth and elevation strike angles.

The target may be defined using any of three different types of vulnerability data:

1. Symmetrical (missile) target vulnerability data;
2. Box type target vulnerability data;
3. Nonsymmetrical (BRL type) target vulnerability data.

When symmetrical target vulnerability data are used, one vulnerable point is used to define the target. For this format, vulnerable areas are input for varying aspect angles, assuming symmetry in the roll plane. Up to 10 striking velocities, 19 aspect angles, and 5 fragment masses may be used.

Box type target vulnerability data may have a different vulnerable area for each of the six sides of a box about each vulnerable target point. In this format, up to 10 striking velocities, 5 fragment masses, and 10 vulnerable points may be utilized to define the target.

If nonsymmetrical (BRL type) vulnerability data are used, vulnerable areas are input which are a function of azimuth and elevation polar angles. Up to ten vulnerable points can be employed. In order to make this type target definition as flexible as possible, the parameters used in the target definition are not individually limited in number. Rather it is required that the product of the number of discrete values for both polar angles, striking velocity and fragment mass, plus the sum of the total number of discrete values for the seven parameters, does not exceed 1000.
If shielding is present in the target, provision is made to define a set of shielding ellipsoids, which in effect prevent fragments from the warhead from striking a vulnerable target point if the ellipsoid lies between the detonation point and vulnerable point.

b. Vulnerability to Blast and Direct Hit

Target vulnerability to blast effects is defined in terms of blast centers associated with one or more blast ellipsoids about each of up to 10 vulnerable areas of the target. The distance from the ellipsoid surfaces to the target represents the maximum distance at which a given warhead can achieve a blast kill against the target. For each encounter analyzed in the program, the detonation point of the warhead is checked to determine if it falls within the bounds of any of the ellipsoids at the time the blast wave reaches the blast center. If so, the target is assumed to be killed; otherwise, there is no target damage due to blast.

An additional set of ellipsoids is used to approximate the physical shape and dimensions of the target. These ellipsoids are employed to determine direct hit kills. For a given weapons/system/target encounter if the detonation point of the warhead is within any of the direct hit ellipsoids or the trajectory of the missile passes through any of the ellipsoids prior to detonation, the kill probability due to direct hit is a prespecified input value, less than or equal to 1.0.

c. Vulnerability to Continuous Rods

In evaluating continuous rod warheads, the target is represented by a series of line segments called spars, which when broken contribute toward achieving a structural kill of the target. A maximum of 50 spars may be used to describe the target. The capability to input up to five sets of vulnerability data for each target spar is provided in the program. This capability allows the user to optimize diameter of a continuous rod, since rod diameter directly affects vulnerability of target spars.

Each spar is defined by the following:

1. Direction cosine of the spar with respect to the X axis;
2. Direction cosine of the spar with respect to the Y axis;
3. Direction cosine of the spar with respect to the Z axis;
4. Spar length;
5. The rod rejection criteria number (Appendix I);
6. X coordinate of spar origin with respect to target CG;
7. Y coordinate of spar origin with respect to target CG;
8. Z coordinate of spar origin with respect to target CG;
9. X direction cosine of the normal to the target surface at the spar;
10. Y direction cosine of the normal to the target surface at the spar;
11. Z direction cosine of the normal to the target surface at the spar;
12. Spar contribution toward a kill, if hit by rod of given diameter.

Effects of target shielding are considered in the continuous rod warhead evaluation, by inputing up to 20 ellipsoid shaped shields. The shielding ellipsoids are defined using:

1. X, Y, and Z coordinates of ellipsoid centroids;
2. Length of ellipsoid axes with respect to X, Y, and Z axes.

When the striking point of the rod on the first spar is obtained, all shielding surfaces are checked to determine if the rod passed through any shield before impacting the spar. It is assumed that if the rod intersects any shielding surface prior to striking the target, it will not kill the target.
If no shield is encountered, the velocity of the rod where it impacts the spar and the angle the strike velocity vector makes with the spar are computed. The strike angle and velocity are then compared with the strike angle and velocity required to kill the spar. If the spar is killed, its contribution toward killing the target is recorded and the program continues to evaluate effects at the next spar. If the strike velocity or angle are not sufficient to kill the target, a ricochet is recorded and the program proceeds to the next spar. The target is assumed killed when the statistical sum of all individual spar contributions toward a kill exceeds an input value specified for this purpose. The program is capable of evaluating rods both before and after breakup occurs. At a radius that is a prespecified number of times larger than the breakup radius, the rod fragments are assumed to be nonlethal.

5. FUZE CHARACTERISTICS

The Air-to-Air Weapon Optimization Computer Program user may simulate various types of fuze concepts. Included at the present time is the capability of evaluating four concepts as discussed below (refer to Appendix III for detailed explanation). Three methods for calculating delay times, that is, the elapsed time between the time of fuzing and time of warhead detonation, are included. These methods where applicable are discussed in detail. In no case can the delay time be negative.

The four fuzes simulated by the program all have the following basic characteristics:

1. A radar fuze cone, symmetric about the body axis of the missile, which is capable of detecting a return from any of several fuzing glitter points (specified by input) on the target if such a point enters the radar beam within a certain maximum range (range to cutoff distance);

2. Mathematically, different glitter points can enter the radar beam at different times; however, fuzing is said to occur at the time when the first (in time) return from a glitter point is detected;

3. The program is capable of simulating the effect of a fuze jammer carried by the target; such a fuze jammer can cause fuzing to occur at a time which is different from the time fuzing would have occurred in the absence of the jammer; the jammer can cause fuzing to occur on fuzing radar beam side lobes or to occur prematurely (at long range) on the main lobe. If the target is carrying a fuze jammer, the computations required for determining whether or not the jammer causes fuzing are mathematically the same as for normal fuze glitter points. If a jammer is used, actual fuzing will occur because of the jammer or because of a fuzing glitter point, whichever occurs first in time.

The four types of fuzes are discussed in the following paragraphs.

a. Hollow Cone Fuze (See Figure 3, page 6)

Warhead fuzing occurs when the target glitter points pass through the conical surface of a radar fuze cone whose apex is located at the radar antenna and whose axis of symmetry extends forward along the longitudinal axis of the missile. The fuze cone angle \( \alpha \) is measured from the axis of symmetry at the cone apex. A cutoff range measured from the cone apex is also defined beyond which a penetration of the conical surface will not fuse the warhead. The warhead detonates at some predetermined time delay after penetration of the conical surface within the range-to-cutoff distance. A table of delay times versus estimated closing velocities may be input and the program picks the specified fuse delay time for the estimated closing velocity and detonates the warhead at that point.

b. Proximity Fuze (See Figure 4, page 7)

Warhead fuzing occurs when the target glitter points pass through the spherical surface of a radar fuze cone. This spherical surface area is defined by a specific cone angle and spherical radius (range-to-cutoff distance) input, and is symmetric about the longitudinal axis of the missile. When the target glitter points pass through the surface of the sphere, fuzing occurs and the warhead detonates at some predetermined time delay thereafter. As previously noted, a table of delay times versus estimated closing velocities may be input and the program will pick the specified delay time.
c. Double Cone Fuze (See Figure 5, page 9)

Detonation point of the warhead is determined by calculating the elapsed time between target penetration of the first and second fuze cones. Input values for this routine specify two angles measured from the longitudinal axis of the missile, a minimum and maximum fuze actuation time, and the range-to-cutoff distance as measured from the apex of the cone, located at the radar antenna. The conical surface of each cone is defined as in (a.) above. The cutoff range defines the distances beyond which the penetration of either cone will not be detected. If a target glitter point passes within the cutoff range, a timing circuit measures the elapsed time between target penetration of the first and second fuze cones. If the elapsed time is greater than the specified minimum and less than the specified maximum fuze actuation gate times, then the warhead is detonated at a predetermined (fixed) delay time after passing the second surface.

d. Smart Fuze

The smart fuze is activated when the radar fuze cone detects a target glitter point passing through the conical surface of the radar cone. The geometric representation of the conical surface is the same as that given for the hollow fuse cone, (paragraph 5a above). When the smart fuze receives the return from the target glitter point, an attempt is made to compute and use an optimum delay time. For this fuzing option, coordinates of the target radar centroid are used as the aim point. The warhead is moved along in small delay time steps and iterative attempts are made at each step to determine if fragments leaving the warhead at some angle \( \theta \) with initial static velocity \( V_p \) will hit the coordinates \( X, Y, Z \) of the target radar centroid. As soon as the optimum delay time is found the search terminates and the warhead is detonated.
SECTION III
ENCOUNTER CONDITIONS AND METHODOLOGY

This section presents a discussion of warhead/target encounter parameters, and the basic methodology used in formulating the Air-to-Air Weapon Optimization Computer Program.

1. ENCOUNTER CONDITIONS

In defining warhead/target encounters which can be evaluated using the Air-to-Air Weapon Optimization Computer Program, it is necessary to specify terminal delivery conditions, warhead, target, and fuze characteristics, number of Monte Carlo interceptions which are to be evaluated, approach angles, guidance errors, aim points, and certain atmospheric data.

A Monte Carlo random number selection process is used in specifying the azimuth and elevation approach angles. A series of angle increments, both elevation and azimuth, are chosen and a probability assigned to each increment. Thus the probability of a given encounter being in each particular static angle increment defines the attack conditions for each sample.

Miss distances may be computed in three ways. The XZ plane of the primed (relative) coordinate system is the plane of intercept, and one method of computing miss distances varies the X and Z values independently, while a second method varies these values dependently. A third method uses a fixed radial miss distance, rather than different values in the X and Z directions.

Intercept parameters which may be evaluated in the parametric or optimization mode are:

1. Standard deviation of guidance miss distance in X direction;
2. Standard deviation of guidance miss distance in Z direction;
3. Guidance miss distance in X-Z plane;
4. Static azimuth approach angle;
5. Static elevation approach angle;
6. Mean azimuth angle of attack;
7. Mean elevation angle of attack;
8. Speed of interceptor;
9. Speed of target.

In addition, several target and warhead characteristics available in the parametric or optimization mode effect the intercept calculations; these parameters are listed in Table I-II, Appendix I.

2. METHODOLOGY

This section presents a discussion of the basic methodology used in formulating the Air-to-Air Weapon Optimization Computer Program. Detailed derivations of expressions used in the analysis are contained in Appendix II, and general Flow diagrams are in Appendix III.
a. Method of Steepest Ascent

(1) Basis

A mathematical technique is required to determine the values of weapon parameters which maximize the lethal area of a given munition. The best known method for determining the maximum value of a function is to differentiate the function, set the derivative equal to zero, and solve to maximize parameter values. The function defining average kill probability (Equation (1)), however, is too complicated to differentiate and solve in closed form:

\[
P_K = \iiint P(x,y,z) f(x,y,z) \, dx \, dy \, dz
\]

where:

\[P(x,y,z)\] is the kill probability for a warhead detonating at \((x,y,z)\),

\[f(x,y,z)\] is the probability density function for the warhead detonating at \((x,y,z)\).

Therefore, a numerical technique is required which involves the evaluation of the function at discrete points in a manner so as to converge to the optimal parameter values as efficiently as possible. A number of techniques are available for this purpose (Reference 3), but the method of steepest ascent, as defined below, was selected as the most promising method for optimization problems when a relatively large number (three or more) of independent parameters are considered.

(2) Mathematical Formulation

The method of steepest ascent, as employed herein, involves repeated applications of the gradient vector of the function defining the munition kill probability until the computations converge to the optimal parameter values. Assuming that kill probability is a function of independent weapon parameters such as fragment mass, size, velocity, height of burst, pattern dimensions, etc., it is defined as follows:

\[
P_K = P_K(x_1, x_2, \ldots, x_n)
\]

where:

\[P_K = \text{kill probability}\]

\[x_i, i = 1, \ldots, n\] are independent weapon parameters.

The gradient vector, \(\text{Grad } P_K\), of Equation (2) is defined by following \(n\) dimensional vector:

\[
\begin{bmatrix}
\frac{\partial P_K}{\partial x_1} \\
\frac{\partial P_K}{\partial x_2} \\
\vdots \\
\frac{\partial P_K}{\partial x_n}
\end{bmatrix}
\]

(3)
Equation (3) may be approximated by

$$\text{Grad}_K P$$

(4)

where $\Delta P_K$ denotes the change in the munition kill probability for an incremental change in $x$ ($\Delta x$). At this point, $\text{Grad}_K P$ is modified in order to increase the rate of convergence of the method of steepest ascent to the optimal parameter values. The modification solves a dimensional problem by defining $\Delta x_i$, $i = 1, \ldots, n$ to be "unit increments" in each variable. Hence, $\Delta x_i = 1$ in the denominator of Equation (4) for $i = 1, \ldots, n$ may be feet, feet per second, grains, degrees, or without dimensions. Equation (4) is now redefined by

$$\text{Grad}_K P = \begin{pmatrix}
\frac{\Delta P}{\Delta x_1} \\
\frac{\Delta P}{\Delta x_2} \\
\vdots \\
\frac{\Delta P}{\Delta x_n}
\end{pmatrix}$$

(5)

where $\Delta P_{K_i}$, $i = 1, \ldots, n$ denotes the change in the munition kill probability with respect to $x_i$, $i = 1, \ldots, n$.

In this application of the method of steepest ascent, it is assumed that repeated application of Equation (5), accompanied by redefining the increments of the independent variables, will eventually converge to the maximum value of Equation (2). The entire optimization procedure is discussed below including initialization, the basic cycle involving applications of gradient $P_K$ and the convergence criteria.

(a) Initialization

If the average kill probability of a particular munition is to be maximized with respect to the $n$ variables, $x_i$, $i = 1, \ldots, n$, the upper and lower limits for each variable must be designated and defined by

$$x_{\text{max}}^{(i)}, x_{\text{min}}^{(i)}, i = 1, \ldots, n$$

(6)

where:

- $x_{\text{max}}^{(i)} = \text{the maximum of the } x_i$
- $x_{\text{min}}^{(i)} = \text{the minimum of the } x_i$

From Equation (6), the increment for each variable is defined, in a manner to efficiently converge to the optimal values, as follows:
\[ \Delta x_i = \left( x_{\text{max}}(i) - x_{\text{min}}(i) \right) / 10 \text{ for } i = 1, \ldots, n \]  

(7)

where:

- \( \Delta x_i \) is the incremental value of the \( i \)th variable
- 10 is a constant selected to enhance efficient convergence.

Initial estimates of the optimal values of each parameter must also be designated and are denoted by the case for \( m = 1 \), as follows:

\[ x_i(m), i = 1, \ldots, n \]  

(8)

where \( x_i(m) \) is the value of the \( i \)th variable at the \( m \)th cycle. The kill probability for \( x_i(1), i = 1, \ldots, n \) is computed and defined by the \( m = 1 \) case of

\[ P_K = P_K \left[ x_1(m), x_2(m), \ldots, x_n(m) \right]. \]  

(9)

(b) Applications of Gradient \( P_K \)

The series of operations through which the procedure cycles, until convergence is initiated by computing the components of the vector defined in Equation (5) with the use of first order forward differences, is as follows:

\[ \Delta P_{K_i} = P_K \left[ x_1(m), \ldots, x_i(m) + \Delta x_i(m), \ldots, x_n(m) \right] \]  

\[ - P_K \left[ x_1(m), \ldots, x_n(m) \right] \text{ for } i = 1, \ldots, n. \]  

(10)

The gradient of \( P_K \) at the point \( x_i(m), i = 1, \ldots, n \) as approximated by Equation (5) has now been determined and can easily be manipulated to compute the direction cosines. The direction cosine in each dimension is defined by

\[ \gamma_i(m) = \Delta P_{K_i} \left( \sum_{j=1}^{n} \Delta P_{K_j} \right)^{1/2}, i = 1, \ldots, n. \]  

(11)

A step is now taken in the direction of this unit vector to a new pivotal point. The coordinates of this new point are defined by

\[ x_i(m+1) = x_i(m) + 3 \Delta x_i(m), i = 1, \ldots, n. \]  

(12)

The step size, \( 3(\Delta x_i) \), in each dimension is used to increase the convergence rate of the optimization process.

Before recycling for the next series of calculations, the direction cosines of each variable are investigated to determine whether that variable crossed the maximum point. It is assumed that if \( x_i(m)/x_i(m-1) < 0 \), then the \( i \)th variable has "peaked out." This being the case, \( \Delta x_i \) is set equal to \( \Delta x_i / 2 \) to enable rapid and accurate convergence to the optimal value of \( x_i \). This adjustment of the parameter increments and the computation of

\[ P_K \left[ x_1(m+1), x_2(m+1), \ldots, x_n(m+1) \right] \]  

ends the \( m \)th cycle.

(c) Check for Convergence

After the first three cycles two checks for convergence of the optimization procedure are performed. The first criterion is that every variable must have at least one change in sign of its direction cosines. This requires, in effect, that the value of kill probability "peak out" at least once for each variable during the optimization procedure. (Upper and lower limits for each parameter to be optimized must be established. Therefore, the value of kill probability is assumed to "peak out" when either of these limits is reached.)
The second and most important criteria for convergence is that the weapon's kill probability for three consecutive cycles will not vary more than two given percentages as defined below:

\[
\frac{P_K(m) - P_K(m-1)}{P_K(m-1)} < CC1
\]

\[
\frac{P_K(m+1) - P_K(m)}{P_K(m)} < CC2
\]

where CC1 and CC2 are constants used to control the accuracy of the optimal parameter values, e.g., 0.04 (4 percent) and 0.02 (2 percent) are typical values for these constants.

(3) Weapon Optimization Capability

(a) General

The discussion of optimization, thus far, has been abstract, i.e., unrelated to a weapon design or an effectiveness problem. The method of steepest ascent is capable of optimizing desired weapon parameters when used in conjunction with a warhead lethality evaluation program. Within this overall study, techniques were developed to enable the optimization of 24 parameters which define warhead, target, and encounter characteristics. Detailed discussions of these capabilities are contained in the applicable portion of Sections III through VI of this volume, and a list of variables which can be optimized or parameterized is contained in Table I-II of Appendix I.

(b) Summary

The method of steepest ascent has a minor limitation peculiar to all numerical techniques: the characteristics of a function are known only at points where it has been evaluated. Therefore, if an optimization computer run determines the maximum average kill probability of a given warhead, this maximum might possibly be only a local maximum value and not the absolute maximum. These cases will be rare and can be anticipated by briefly investigating warhead and target input data. For example, suppose the kill probability for a particular target for a given fragment mass and velocity showed the trends presented in Figure 9. If the fuze delay time of a warhead was being optimized, the kill probability could have two local maxima, only one of which was truly optimal. This example is illustrative of the logic required to determine when such cases exist. An additional feature of the program which provides further protection against optimizing at a local maximum is the parametric capability discussed in the following paragraph.

![Figure 9. Target Vulnerability Characteristics](image-url)
b. Parametric Mode Capability

An additional feature incorporated into the weapon optimization program, which provides for the parameterization of selected variables instead of determining their optimal values, is included in the model. The same system and warhead parameters that the program optimizes can be compared at given increments between limits designated by the user.

This capability was formulated so that all combinations of the values to be evaluated for each parameter can be determined. For instance, if it is desired to vary missile velocity from 1000 to 2000 feet per second in increments of 500 feet per second, the miss distance 0 to 20 feet in increments of 10 feet, and fragment mass from 60 grains to 120 grains in increments of 60 grains, then a total of 18 munition kill probabilities would be calculated for the parameters given in Table I.

As described above, the ability to easily compute kill probability over a spectrum of designated values is provided in the program. This capability enhances the value of the computer program for two reasons:

1. The sensitivity of kill probability to given parameters may be investigated;
2. The presence of more than one relative maximum point may be determined.

<table>
<thead>
<tr>
<th>Missile Velocity (ft/s)</th>
<th>Miss Distance (ft)</th>
<th>Fragment Mass (gr)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,000</td>
<td>0</td>
<td>60</td>
</tr>
<tr>
<td>1,000</td>
<td>0</td>
<td>120</td>
</tr>
<tr>
<td>1,000</td>
<td>10</td>
<td>60</td>
</tr>
<tr>
<td>1,000</td>
<td>10</td>
<td>120</td>
</tr>
<tr>
<td>1,000</td>
<td>20</td>
<td>60</td>
</tr>
<tr>
<td>1,000</td>
<td>20</td>
<td>120</td>
</tr>
<tr>
<td>1,500</td>
<td>0</td>
<td>60</td>
</tr>
<tr>
<td>1,500</td>
<td>10</td>
<td>120</td>
</tr>
<tr>
<td>1,500</td>
<td>10</td>
<td>60</td>
</tr>
<tr>
<td>1,500</td>
<td>20</td>
<td>120</td>
</tr>
<tr>
<td>1,500</td>
<td>20</td>
<td>60</td>
</tr>
<tr>
<td>2,000</td>
<td>0</td>
<td>60</td>
</tr>
<tr>
<td>2,000</td>
<td>0</td>
<td>120</td>
</tr>
<tr>
<td>2,000</td>
<td>10</td>
<td>60</td>
</tr>
<tr>
<td>2,000</td>
<td>10</td>
<td>120</td>
</tr>
<tr>
<td>2,000</td>
<td>20</td>
<td>60</td>
</tr>
</tbody>
</table>
SECTION IV
CONCLUSIONS AND RECOMMENDATIONS

1. CONCLUSIONS

An Air-to-Air Weapon Optimization Computer Program was developed which can optimize and evaluate fragmentation, blast, cluster, and continuous rod warheads against tactical air targets. Models were developed which accurately represent the targets, consistent with the warhead types to be evaluated. The warhead and target models are used in combination with fuse models and system errors to accurately and realistically simulate an air encounter.

The program developed utilizes a Monte Carlo random number selection routine to select various encounter geometry and fuzing parameters from specified distributions of each parameter. Thus, the program is able to accurately simulate a large number of random warhead/target encounters to determine an average or single-shot kill probability ($P_{SSK}$).

The program uses the method of steepest ascent optimization technique, supplemented with the use of a parametric mode for parameter sensitivity studies. Other techniques which were investigated include linear programming, accelerated gradient, and differential programming. In addition, the program incorporates the following features:

1. High efficiency for optimization or computation of warhead lethality with minimum computer time;
2. State-of-the-art technology employed in the weapon evaluation model;
3. Mathematical definition of warhead performance data and target vulnerability data compatible with current government agency weapon and target descriptions;
4. Program constructed in modular form to permit future modifications with minimal effort.

2. RECOMMENDATIONS

Based on current armament requirements and the experience and knowledge acquired during the conduct of this program, it is recommended that the following items be investigated to further develop the capability to optimize and evaluate nonnuclear weapon systems:

1. Complete the formulations described in Appendix IV of the mathematical model for evaluation of linear shaped charge warheads and add the capability for evaluating this type munition to the Air-to-Air Weapon Optimization Computer Program;
2. Formulate and code a computer program for the lethality evaluation of aerially emplaced mines against mobile personnel and materiel targets;
3. Formulate and develop computer routines for evaluation of variable charge-to-metal ratio warheads (catastrophic structural kill criteria);
4. Develop a composite aerial dogfight-to-endgame lethality evaluation computer program to provide flight simulations from target acquisition to kill probability computations.
SECTION V

FLECHETTE LETHAL AREA AND LETHAL AREA OPTIMIZATION PROGRAMS

1. PROGRAM OBJECTIVES

The overall objective of this portion of the research program was to develop computerized mathematical models for evaluating the lethal area of munitions against personnel and area targets. Specific tasks of the program were to:

1. Construct a computer program for computing the lethal area of flechette dispersing warheads whose projectile flight paths are affected by gravity;
2. Incorporate into the Lethal Area Optimization Program (previously developed under Contract AF08(635)-5871) methodology for establishing fractional kill probabilities of bomblet munitions against area targets.

2. PROGRAM APPROACH AND RESULTS

The general outline of the program is presented in Figure 10. As illustrated, methodology was developed to construct a flechette lethal area model; and, in a parallel effort techniques were formulated to compute fractional kill probabilities against area targets. The approach and results for each of these areas are discussed below.

a. Flechette Lethal Area Program

(1) Approach

Review of present technology determined that no current lethality program was applicable to flechette warheads employed in such a way that gravity effects were of significance. Therefore, an entirely new program was undertaken to develop mathematical derivations for evaluating the lethal area of flechette warheads against personnel targets. Extensive liaison and literature surveys were conducted to ensure that the methodology developed could be incorporated into an accurate and practical program.

(2) Results

A program was assembled to compute the lethal area of warheads projecting flechettes, especially from high altitudes. The principle feature is that flechette trajectories are actually simulated by the equations of motion to properly account for the effects of gravity. The warhead attack is defined by a height and angle of flechette release. In this program, flechette characteristics may be described by up to three classes, each containing unique considerations of:

1. Mass;
2. Initial velocity;
3. Beam spray angle, distribution, and number of flechettes;
4. Reference diameter;
5. $C_D$ versus Mach number data.

Munition lethal area is established by integrating kill probability over the impact pattern area, for standing or prone troops or troops crouching in foxholes.
Air-to-Air Weapon Optimization Techniques

Modify and Develop Lethal Area and Kill Probability Weapon Optimization Computer Programs

Development of Methodology for Establishing Fractional Kill Probabilities

Development of Methodology for Flechette Lethal Area Program

Coding of Mathematical Models

Coding of Mathematical Models

Program Checkout and Verification

Program Checkout and Verification

Adaptation of Programs to Eglin AFB CDC 6600

Delivery of Programs to Eglin AFB

Final Report and Recommendations

Figure 10. Air-to-Air Weapon Optimization Program Flow Plan
b. Fractional Kill Against Area Targets

(1) Approach

Development of methodology for fractional kill probabilities against area targets was accomplished through the formulation of mathematical equations relating composite target conditional kill probabilities and fractional target coverage. The methodology resulting from this investigation was incorporated into the Lethal Area Optimization Program.

(2) Results

This study resulted in a modification to the Lethal Area Optimization Program expanding its capabilities to include the evaluation of fractional kill against area targets. The area target may consist of one to five elements, unrevetted and uniformly randomly distributed in the target area. One payload of bomblets is delivered to the target with a specified error distribution. The derived methodology uses a pattern optimization technique which maximizes lethal areas to establish the optimal pattern size and the kill probability achieved by this pattern. This additional capability increases the flexibility of the Lethal Area Program and provides an expedient technique for optimizing patterns.

3. RECOMMENDATIONS

As a result of this program, it is recommended that follow-on weapon optimization technique programs be conducted to:

1. Expand and modify the Flechette Lethal Area Program to include:
   a. Evaluation of lifting body flechettes;
   b. Materiel type targets;
   c. Optimization procedures for warhead and flechette release parameters;

2. Investigate the feasibility of modeling flight paths of bomblets (with and without magnus lift) and incorporating this model into the fractional kill routine of the Lethal Area Program for the purpose of optimizing bomblet release parameters.

4. REPORT ORGANIZATION

The remainder of this report discusses in detail the input formats, capabilities, and methodology of the two programs discussed above. Section VI explains the warhead and target representation, terminal delivery conditions, and methodology contained in the Flechette Lethal Area Program. Section VII contains the methodology and formulation of fractional kill probabilities against area targets, and an explanation of the modifications incorporating this model into the Lethal Area Program. Section VIII of this report contains conclusions and recommendations. Appendixes V through IX contain details of geometry and flow diagrams, Runge-Kutta solutions of equations of motions and techniques for determining polygon parameters, grid point test, and utilization reports on the Flechette Computer Program and the Lethal Area Optimization Program.
SECTION VI
FLECHETTE LETHAL AREA PROGRAM

This section presents a detailed discussion of the input parameters, capabilities, and methodology contained in the Flechette Lethal Area Program.

1. WARHEAD REPRESENTATION - Figure 11

The warhead is assumed to release flechettes in a conical pattern with prescribed angular limits defined by a beam spray half angle measured off the longitudinal axis of the warhead. The flechettes can be distributed within this conical pattern in a uniform or normal manner, determined by an indicator in the input data. Beam spray half angles are limited to 30 degrees or less for warhead dive angles less than 30 degrees. In no case can the beam spray half angle exceed 90 degrees (this is a 3 sigma value for normal distributions). Only one-half of the ejection pattern is evaluated as the warhead is assumed to be symmetric about a vertical plane containing the longitudinal axis. The conical pattern defined at release is subdivided into increments along the beam spray half angle, measured off the longitudinal axis of the warhead, and further divided into radial increments measured around and perpendicular to this axis (Appendix V, Figure V-1). These angular increments of the flechette pattern are then projected into the ground plane by trajectory simulations. The program is capable of simulating a warhead containing up to three distinct classes of flechettes. Each class of flechettes may have a unique:

1. Flechette mass;
2. Initial flechette velocity;
3. Beam spray angle and angular increments;
4. Distribution and number of flechettes;
5. Flechette reference diameter;
6. $C_d$ versus Mach number curve.

Figure 11. Sample Warhead Geometry
In addition, up to five sets of flechette reference diameters and/or $C_D$ versus Mach number curves may be used as a function of real time of flight for each class of flechettes. This capability enables the simulation of unstable flechette flight.

2. TARGET REPRESENTATION

The Flechette Lethal Area Program was compiled to evaluate the lethal area of flechette projecting warheads against personnel targets. Munition lethal area is established by integrating kill probability against either standing or prone troops or troops crouching in foxholes within the impact pattern area. Vulnerability of personnel targets is defined in terms of target presented area and a conditional kill probability established for a hit on the target. Presented area of the target is a function of the orientation of the flechette at impact; i.e., the impact angle. Conditional kill probability is established through a casualty criterion equation which utilizes flechette mass, striking velocity, and three constants defining the kill criterion in accordance with "Casualty Criterion for Wounding Soldiers" BRL TN1486 (Reference 4).

3. TERMINAL DELIVERY CONDITIONS

Terminal delivery conditions of the missile are significant in that a damage assessment point in the lethal area grid must be determined by integrating the equations of motion from specified initial conditions. Velocity and direction of a given flechette at release predetermines the point of impact in the ground plane. The warhead, as discussed above, is subdivided into a grid defining orientation of a flechette dispersion pattern, where each point in this pattern is integrated through time to establish an impact pattern in the ground plane. Release geometry at launch is defined relative to the missile coordinate system, and may include up to three distinct classes of flechettes.

Initial velocity, $V_M$, imposed on flechettes of a given class is determined relative to the ground plane. This is the total velocity of the flechette and represents the vector sum of the missile velocity and any ejection velocity associated with the dispersion mechanism. At the time of release stability of the flechette becomes a problem that is not readily defined. However, the tumbling of an unstable flechette may be simulated by adjusting a portion of the drag curve where instability would occur. Since the reference diameter is directly proportional to the drag it may also be altered to account for instability. Up to five sets of flechette diameters and/or $C_D$ versus Mach number curves may be input for each class. These data are a function of time of flight and enable the simulation of tumbling, oscillating, and stable motion. A smaller time increment of integration may be employed in the trajectory computation for a specified period of time following a change from one set of drag data to another. This is to ensure accuracy in the iterative solution of the equations of motion.

Height of release, HOB, is defined by input data, and is common for all classes of flechettes. This program was especially designed to cope with high altitude releases, as the gravity effects are more influential for this terminal release condition.

The warhead terminal attack angle, ALPHA, (measured positively downward from the horizontal to the nose of the missile) is input to relate the flechette flight direction to the ground. This is necessary as the dispersion pattern is defined relative to the missile axis, and some relationship must be established between its orientation and the impact plane. Gravitational forces are assumed to be normal to the ground plane.

4. METHODOLOGY

The methodology contained in the Flechette Lethal Area Program was formulated from basic mathematical and physical equations. It may be subdivided into three orderly sections discussed in detail below.

a. Warhead Methodology

The simulation of a flechette dispersing warhead was formulated by geometrically establishing an ejection grid, relative to the warhead or missile axis as discussed in
paragraph 1 above. Each point in this pattern is associated with the directional vector of a flechette at the time of release. The vector is assigned an initial velocity obtained from the input data, and is transformed into its three components along the coordinate axes of the ground reference system. These three velocity components given by $V_X, V_Y, V_Z$, are then used in solving the equations of motion for the trajectory routine.

b. Trajectory Analysis - Figure 12

All effectiveness methodology currently available contains the assumption that fragments fly straight line trajectories from warhead to the target. This assumption is valid if velocity is high, distance to the target is low, and hence time of flight is small. However, flechettes launched at high altitudes have a relatively long flight time; thus gravity significantly affects the trajectories of the flechettes. To accurately account for these effects the equations of motion must be integrated from specified initial conditions (velocity and direction) at launch, through time and space, to impact. The lethal area integration grid must be established by initial conditions of projectiles at the warhead point of detonation. The equations of motion defining the flight of each flechette are derived from Newton's second law where the drag force is given by:

$$F_D = \frac{1}{2} C_D \cdot \rho \cdot AV^2 = -M \frac{dv}{dt}.$$  

Expanding this formula into component forms relative to the ground reference system we can generally describe the equations of motion for a given flechette by:

$$X = -\left(\frac{1}{2} C_D \cdot \rho \cdot A/M\right) V_X V_x - g$$  \hspace{1cm} (13)

$$Y = -\left(\frac{1}{2} C_D \cdot \rho \cdot A/M\right) V_Y V_y$$  \hspace{1cm} (14)

$$Z = -\left(\frac{1}{2} C_D \cdot \rho \cdot A/M\right) V_Z V_z - g$$  \hspace{1cm} (15)
where:
\[ C_D \] is the drag coefficient
\[ \rho \] is the density of air
\( A \) is the presented area of the flechette
\( V_x, V_y, V_z \) are the velocity components of the flechette velocity vector \( V \)
\( M \) is the flechette mass
\( g \) is the acceleration of gravity.

Integration of Equations (13), (14), and (15) cannot be performed in closed form and requires a finite difference iterative procedure. The numerical solution of these equations is performed in the program by the method of Runge and Kutta (Appendix VI). From the solution of these basic equations of motion, the striking velocity, impact angle, and the impact coordinates (relative to the point of release) of the flechettes in the ground plane can be determined. The impact pattern (Figure 13) is determined by looping the above procedure for each point of release in the ejection grid. The polygons formed in the ground pattern by the impact coordinates are sequentially numbered as shown in Figure 13.

![Figure 13. Sample Impact Pattern Geometry](image-url)

### c. Lethal Area Methodology

As discussed in paragraph 1 above, methodology was developed to evaluate munitions projecting up to three classes of flechettes. This makes it possible for the simulation to contain one, two, or three impact patterns which may or may not overlap each other.

For the lethal area computations, the total impact pattern of all classes of flechettes is divided into two parts:

1. Areas in the ground plane which contain flechettes from only one class;
2. Areas in the ground plane which contain flechettes from two or more classes.

Lethal area computations are accomplished through a numerical integration of kill probabilities over all area. These kill probabilities can be easily calculated for areas in the ground pattern which contain flechettes from only one class; however, those areas in which the patterns overlap present a more complex solution. The program initially assumes that no overlap occurs, and the area of each polygon (N) in all class patterns is individually calculated. The impact angle and terminal velocity of flechettes at each polygon vertex was calculated in the trajectory portion of the program. An average or mean value of velocity, and impact angle is calculated for each polygon using these data (Appendix VI). In addition, density of the flechettes is calculated by taking the total number of flechettes in the original polygon (determined at release) and dividing by the polygon area at impact. Kill probability can now be calculated by:

\[
PK(N) = 1 - e^{-\frac{A \cdot V \cdot C}{P \cdot C}}
\]  

(16)
where:

- \( \rho \) = flechette density in polygon \( N \)
- \( A \) = presented area of target
- \( P_c \) = conditional kill probability defined by Reference 5
- \( f(M, \text{VAVE}) \)
- \( M \) = flechette mass
- \( \text{VAVE} \) = average flechette impact velocity in polygon \( N \).

A preliminary munition lethal area is calculated by summing the product of kill probability and area for all polygons.

\[
AL = \sum_{i=1}^{K} P_{c}(i) \cdot A(i)
\]  

(17)

where \( K \) is the total number of polygons in all classes.

At this point in the program a check is made to determine if the ground patterns from the various groups overlap. If only one class of flechettes is being run, or if no overlap occurs, one half the munition lethal area is given by Equation (17) above. If two or more classes have flechettes falling in a common area, the overlap condition exists. Limits are then determined to localize the areas of overlap, and a rectangular grid of points is superimposed over this area. Each grid point is associated with a calculated or prespecified area, \( A \), depending on whether a total number of grid points or a \( \Delta A \) is input respectively. A test is made on each grid point to determine what polygons it lies in for all classes (Appendix VII), i.e., a grid point may lie in polygon two, class one, and polygon twelve, class two. The loss of lethal area, \( \Delta AL \), due to overlap is calculated for the incremental area associated with this grid point by:

\[
\Delta AL = \Delta A \sum_{i=1}^{J} P_{c}(i) - \left[ 1 \right] - \left[ 1 - P_{c}(i) \right]
\]

(18)

where:

- \( \Delta A \) is grid incremental area, \( \Delta A \)
- \( P_{c}(i) \) is kill probability in polygon \( N \) of class \( I \)
- \( J \) is the total number of polygons which contain the grid point.

The lethal area for cases of overlap is then given by:

\[
\text{LETHAL AREA} = (AL - \sum_{i=1}^{G} \Delta AL) \cdot 2
\]

(19)

where \( G \) is the number of grid points having overlap.

The factor of two must be included to account for the fact that only one half the impact pattern is evaluated due to symmetry.

5. SUMMARY OF RESULTS

The Flechette Lethal Area Program has been summarized in a general flow diagram presented in Figure V-3, Appendix V. In essence this program represents a unique technique for evaluating the lethal area of flechette warheads against personnel targets. The program utilizes inputs of flechette orientation, velocity, mass, drag data, and number of flechettes, for up to three classes, to calculate impact patterns and munition lethal area. A special plot subroutine is available to provide a print out of the impact pattern. Options are available for varying the reference diameter (used to obtain flechette presented area) and \( C_d \) (drag coefficient) curves. This capability enables the user to generally simulate projectile flight instability. A detailed utilization report is presented in Appendix VIII for reference.
SECTION VII
LETHAL AREA OPTIMIZATION PROGRAM

1. AREA TARGETS

The Lethal Area Weapon Optimization Computer Program developed under contract AF08(635)-5871 has been modified to incorporate the capability to evaluate or optimize cluster munitions against large area targets. This program is now capable of establishing the following data:

1. Expected fraction of elements killed within limits of an area target by one weapon payload of bomblets;
2. Pattern size which maximizes fractional kill probability in 1 above.

Target input parameters include the following:

1. Target half width, half length, and type (rectangular or elliptical);
2. Target lethal area (generated within the program);
3. Target element weighting factors.

Item 3 above enables evaluation and optimization of fractional kill probability against a spectrum of up to five different target types. The weapon is defined by the following parameters:

1. Number of bomblets;
2. Total delivery errors (sigma) in the cross range and range directions;
3. Pattern half length, half width, and type (rectangular or elliptical).

Techniques employed for establishing fractional kill probability of a bomblet payload against one target require the computation of conditional kill probability for target elements in the bomblet impact pattern area, and the expected fraction of target area covered by a bomblet pattern.

Conditional kill probability of a cluster weapon against an area target can be determined using Equations (20) and (21) below, given the following assumptions:

1. Bomblets are uniformly randomly distributed in the impact pattern;
2. Impact pattern is large with respect to the bomblet radius of effectiveness;
3. Target elements are unrevetted and uniformly randomly distributed in the target area;
4. One payload of bomblets is delivered to the target with a specified error distribution.

\[
P_C = \sum_{i=1}^{N_t} W_i P_{C_i} \quad (20)
\]
where:

\[ P_c \] is composite target conditional kill probability

\[ N_T \] is number of types of target elements

\[ W_i \] is weighting factor of the \( i \)th element type (0 < \( W_i \) ≤ 1)

\[ P_{k|} \] is conditional kill probability of the \( i \)th element type

computed by Equation (21) below:

\[ P_{k|} = 1 - \left( 1 - \frac{A_{L_i}}{A_p} \right)^{N_B} \]  \hspace{1cm} (21)

where:

\[ N_B \] is number of bomblets in a payload

\[ A_{L_i} \] is lethal area of each bomblet for the \( i \)th element type

\[ A_p \] is bomblet pattern area.

At this point, the expected fraction of target area covered by one bomblet pattern is determined. Equation (22) below is numerically integrated to obtain this value:

\[ f_c = \frac{1}{A_T} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi} \sigma_x \sigma_y} \exp \left[ -\frac{1}{2} \left( \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right) \right] \, dx \, dy \]  \hspace{1cm} (22)

where:

\[ f_c \] is expected fraction of the target area covered

\[ A_T \] is target area

\[ A(x,y) \] is target area covered given a pattern center at \((x,y)\)

\[ \sigma_x, \sigma_y \] are the standard deviations of the total delivery errors.

The expected fraction of target elements killed by one payload, fractional kill probability, is given by:

\[ F = P_c \cdot f_c \]  \hspace{1cm} (23)

These procedures have been compiled into the lethal area optimization computer program and checked for computational accuracy.

2. EXPANSION OF WARHEAD CAPABILITIES

An additional modification made to the Lethal Area Program expands its warhead capabilities to include the evaluation of warheads emitting a plane of fragments. Methodology involved in evaluating a warhead of this type assumes this plane of fragments is oriented perpendicular to the longitudinal axis of the warhead. Total number of fragments in the plane is provided by input data. The detonation point of the warhead locates the plane in space, at which time the program determines whether or not the plane intersects the target vulnerable area. If an intersection occurs kill probability is calculated using standard formulation for fragmentation kill. This modification increases capability of the Lethal Area program enabling the evaluation of conceptual designs utilizing a plane of high velocity fragments.
To calculate the fragment density from a plane of fragments requires a departure from normal procedures as follows. 1) The requirement that the vulnerable point be contained within the fragment beam spray cannot be used. Instead the average presented area of the target is defined from the physical dimensions and used to compute the interaction of the fragment plane with the target. 2) The total number of fragments striking the target presented area is determined by calculating the arc length of the interaction area and dividing by 2π. 3) The fragment density is calculated by dividing the total number of fragments by the average presented area of the target. 4) The kill probability is computed by using the fragment density above and the target vulnerable area (as opposed to presented area) in the formula

\[ P_K = 1 - e^{-\bar{A}_v} \]

The formulation for average presented area \( A_p \) is as follows.

Assume the target is a box with length, width and height of \( L, W, \) and \( H \). Azimuth and elevation angles are \( \theta \) and \( \alpha \) respectively. The area of the top is given by \( L \cdot W \). The presented area of the roof is not a function of azimuth but varies only with elevation such that

\[ A_p(\text{roof}) = \frac{4}{\pi} \sqrt{L \cdot W} \sin \alpha \, d\theta = L \cdot W \sin \alpha \]

The total area of the sides is given by \( 2L \cdot H + 2W \cdot H \) since from any azimuth only two adjacent sides can be seen the average presented area as a function of azimuth is given by

\[ A_p(\text{sides}) = \frac{4}{\pi} \sqrt{L \cdot H \cos \alpha \cos \beta + WH \cos \alpha \sin \beta} \, d\phi = \frac{2H(L + W)}{\cos \alpha} \]

Therefore the average presented area (integrated over azimuth) as a function of elevation is given by

\[ A_p(T) = L \cdot W \sin \alpha + \frac{2H(L + W)}{\cos \alpha} \]

by a series of simultaneous solutions for a line and a plane and another series of checks to determine if these solutions lie on the target surfaces, it is determined whether or not the plane of fragments intersects the target and if so the points \( (x_1, y_1, z_1) \) and \( (x_2, y_2, z_2) \) are defined which are the limits of the line across the target cut by the fragment plane. The above two points plus the warhead detonation point form a triangle in the plane of the fragments, and the interior angle \( \beta \) at the detonation point is determined by the law of cosines.

If the total number of fragments in the plane is \( N_F \), then the number which hit the target \( (N_H) \) is \( \frac{N_F}{2\pi} \) and the average density over the target \( (c_v) \) is \( \frac{N_H}{A_p} \).

The kill probability for the fragment plane is then

\[ P_K = 1 - e^{-\bar{A}_v} \]

where \( A_v \) is the tabular vulnerable area defined in the usual manner as a function of fragment mass, velocity and elevation strike angle.
3. SUMMARY OF RESULTS

Modifications incorporated into the Lethal Area Optimization Program enable it to optimize pattern size of a cluster munition against area targets; and, to evaluate the kill probabilities of warheads emitting a high velocity plane of fragments. The methodology in paragraph 1 above utilizes maximum lethal areas to establish optimum pattern sizes. Planer warhead methodology makes possible the optimization of fragment size and velocities emitted in a plane perpendicular to the warhead axis. It is felt these additions offer rapid and accurate techniques for weapon optimization calculations. A complete, updated utilization guide appears in Appendix IX.
SECTION VIII

CONCLUSIONS AND RECOMMENDATIONS

1. CONCLUSIONS

A computer program was developed to compute the lethal areas of flechette warheads against personnel targets. (The program is capable of simulating the flight trajectory of selected individual flechettes by accounting for the effects of gravity in computing their flight paths.) The methodology developed in this program is unique and fundamental in lethal area formulations. The basic theories involved in the ejection geometry, equations of motion, and numerical integrations over projectile impact patterns may be used to develop specialized models for advanced system concepts in this area.

In addition, modifications to the Lethal Area Optimization Program have expanded its capabilities to include optimization of cluster munition patterns against area targets. The programs developed under this contract were designed:

1. To generate accurate data with minimal computer time;
2. To be compatible with state of the art target vulnerability data and warhead fragmentation characteristics;
3. In modular form permitting modification with minimal effort;
4. Using state of the art technology employed in weapon evaluations.

2. RECOMMENDATIONS

Based upon experience acquired during the development of these programs, the following recommendations are made to further the capabilities of effectiveness technology in the nonnuclear field. It is recommended that additional studies be conducted to include:

1. The development of a flechette lethal area program to evaluate the effects of a variety of flechette sizes against materiel targets;
2. An evaluation of lifting body flechettes against personnel and materiel targets;
3. Optimization procedures for flechette warhead parameters;
4. The development of a composite air-to-surface computer program that would combine missile release and flight conditions with bomblet flight trajectory simulations to compute bomblet impact patterns on the ground plane. The resulting impact patterns can then be incorporated into the fractional kill routine of the lethality evaluation program to optimize missile release parameters.
APPENDIX I

UTILIZATION GUIDE

1. TITLE

Air-to-Air Weapon Optimization Computer Program.

2. Program Description

This program utilizes a Monte Carlo routine to simulate warhead-target encounters, and calculates an average single-shot kill probability \((P_{SSK})\) for a given set of input data. The program can be utilized in the parametric or optimization mode, varying warhead design and encounter parameters. The optimization mode uses the method of steepest ascent to select the optimum variable specified. In the parametric mode, the parameters are varied at specified intervals until the \(P_{SSK}\) for all desired combinations of parameter values have been computed. Various options are available for simulating guidance errors, fuzing, detonation delay time, blast and fragment damage, direct hits, and warhead types. The program is capable of evaluating fragmentation, continuous rod, and sub-missile warheads.

3. TIMING

Timing is highly dependent on the type run being made. A typical run to optimize delay time and fuze angle for a continuous rod warhead against a MIG 17 required 400 seconds of CDC 6400 computer time.

4. MULTIPLE RUN

Runs may be stacked by placing a *RUN card (*in column 1, RUN in columns 7-9) after last data card and then repeating the cards A-H. Cards D-H may be omitted if card A (GTITLE) contains the following message in columns 1-16, "USE PREVIOUS W/H." Columns 17-72 may contain other information which will be printed with output. Only the data in locations 1-5200 which will change need be input ending with a 22222 card. Missile, box, or BRL type vulnerability data may not be changed in a stacked run.

5. INPUT DATA FORMAT

All integers must be right adjusted in their field. Table I-I contains optimization or parametric data, Table I-II lists the variables which may be optimized or parameterized, Table I-III lists warhead data, Table I-IV contains intercept conditions, Table I-V lists fuzing data, and Table I-VI contains general target data. Specific continuous rod target data are contained in Table I-VII, and bomblet data in Table I-VIII. Miscellaneous input data are listed in Table I-IX. Table I-X is a listing of error codes. The input format for Tables I-I through I-III (cards A through G) is self-explanatory.

Each of the data cards in Tables I-IV through I-X has the format given below.

<table>
<thead>
<tr>
<th>Card Column</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>Normally, these columns contain the memory location (right adjusted integer) of the data punched in columns 11-20 of the same card. Table I-IV gives the memory location of every input variable. However, these columns may be blank if the memory location of the data in columns 11-20 is implied by a previous card. This is explained below.</td>
</tr>
</tbody>
</table>
NOTES:
1) "P" is origin of interceptor coordinate system ($X_1$, $Y_1$, $Z_1$).
2) "O" is origin of target and relative coordinate systems ($X$, $Y$, $Z$), ($X''$, $Y''$, $Z''$) respectively.
3) $V_{PBA}$ is velocity vector of interceptor
    $V_{TBA}$ is velocity vector of target
    $V_{BPA}$ is relative velocity vector between the target and interceptor, and is the positive "Y" direction in relative double prime system.

Figure I-1. Basic Geometry of Each Encounter
Card Columns | Contents
--- | ---
6-9 | The total number (a right-adjusted integer) of following input values to be placed in sequential memory locations beginning with the memory location given in columns 1-5. If number in columns 6-9 is <6 the input values will all be punched on this same card. If >6, additional cards are used (with columns 1-9 blank) until all values have been specified. The number in columns 6-9 must always be \(<1000\).

10 | Not used.

11-20 | Data value (floating point) to be placed in the memory location given in columns 1-5. If columns 1-9 are blank, data will be placed in the memory location immediately following the location of the 6th data value on the previous card.

21-30 | Data value (floating point) to be stored in the location immediately following the location of the value in columns 11-20 of this card.

61-70 | Data value (floating point) to be stored in the location immediately following the location of the value in columns 51-60 of this card.

71-80 | Available for alphanumeric identification.

Note that all data values (columns 11-70) must contain a decimal point even though some of these values might more naturally be integers. The first data value on each card must always be in columns 11-20.

When making stacked runs, each run will use the same data as the previous run except for those values which are changed by new data cards. Once a non-zero value has been entered it remains unchanged until changed by a subsequent data card.

a. Printout of the Input data

For each run, the contents of the title card and all data cards is printed out exactly as it appears on the cards. This is followed by a printout (with headings) of all data that is actually used for the run, including data values retained from the previous run. In order to shorten this printout, some variables which will not be used for the run are omitted. For example, if the target has no vulnerable points, fragment vulnerability data is not printed out, since it is not used at all.

b. Explanation of Table I-IV

All variables are stored in "natural order." This is best explained by an example. Table I-IV indicates that NH is a three dimensional array of numbers 10 x 6 x 6, or a total of 360 storage locations. Table I-IV specifies that the NH array begins in memory location 41. The elements of NH are stored in the following manner ("natural order"):

<table>
<thead>
<tr>
<th>Memory Location</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>41</td>
<td>NH(1,1,1)</td>
</tr>
<tr>
<td>42</td>
<td>NH(2,1,1)</td>
</tr>
<tr>
<td>43</td>
<td>NH(3,1,1)</td>
</tr>
<tr>
<td>44</td>
<td>NH(4,1,1)</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>40 + i</td>
<td>NH(i,1,1)</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>50</td>
<td>NH(10,1,1)</td>
</tr>
<tr>
<td>51</td>
<td>NH(1,2,1)</td>
</tr>
<tr>
<td>52</td>
<td>NH(2,2,1)</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

37
<table>
<thead>
<tr>
<th>Memory Location</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>NH(10,2,1)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>NH(10,6,1)</td>
</tr>
<tr>
<td>101</td>
<td>NH(1,1,2)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>400</td>
<td>NH(10,6,6)</td>
</tr>
</tbody>
</table>
### TABLE I-I. OPTIMIZATION OR PARAMETRIC DATA

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Name</th>
<th>Limits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-72</td>
<td>Alphanum</td>
<td>GTITLE</td>
<td></td>
<td>General title of run</td>
</tr>
<tr>
<td></td>
<td>1-5</td>
<td>Integer</td>
<td>NVOPT</td>
<td>$0 \leq 30$</td>
<td>Number of variables to be optimized</td>
</tr>
</tbody>
</table>
|      | 6-10    | Integer | NPAR     | $0 \leq 8$ | Number of variables in parametric mode. If NVOPT ≠ 0, NPAR = 0 and vice versa. Optimization procedure will not converge until the following two conditions are satisfied:  
  \[
  \frac{PKSS(N-1) - PKSS(N-2)}{PKSS(N-2)} < CC1 \\
  \frac{PKSS(N) - PKSS(N-1)}{PKSS(N-1)} < CC2
  \]

  where $PKSS$ is probability of a single shot kill, $N$ is the optimization trial number. See above

|      | 11-20   | Decimal | CC1      |        | Intermediate print controls. If IPR(1) = 0, no print of option 1. If IPR(1) ≠ 0, print as follows: |
|      | 21-30   | Decimal | CC2      |        | See above                                                                 |
|      | 41      | Integer | IPR(1)   | 0.1    | Basic geometry of each encounter                                           |
|      | 42      | Integer | IPR(2)   | 0.1    | Miss distance information                                                  |
|      | 43      | Integer | IPR(3)   | 0.1    | Fuzing information                                                         |
|      | 44      | Integer | IPR(4)   | 0.1    | Position at detonation                                                     |
|      | 45      | Integer | IPR(5)   | 0.1    | Blast effect                                                              |
|      | 46      | Integer | IPR(6)   | 0.1    | Direct hit computation                                                     |
|      | 47      | Integer | IPR(7)   | 0.1    | Intermediate print from subroutine FIMPAC                                  |
|      | 48      | Integer | IPR(8)   | 0.1    | Intermediate print from subroutine DELTAT                                   |
|      | 49      | Integer | IPR(9)   | 0.1    | Intermediate print from subroutine FIT                                     |
|      | 50      | Integer | IPR(10)  | 0.1    | Shielding computations                                                     |
|      | 51      | Integer | IPR(11)  | 0.1    | Vulnerable area computations                                               |
|      | 52      | Integer | IPR(12)  | 0.1    | Complete statistics on probability of killing various vulnerable components of each group |
|      | 53      | Integer | IPR(13)  | 0.1    | Prints out key parameters for each encounter, plus distribution tables of striking velocity, fragment density, vulnerable area, and number of hits |
|      | 54      | Integer | IPR(14)  | 0.1    | Intermediate print from subroutine ODT                                      |
|      | 55      | Integer | IPR(15)  | 0.1    | Prints message and other data is blast iteration does not converge         |
|      | 56      | Integer | IPR(16)  | 0.1    | Non-symmetrical warhead computations                                        |
|      | 57      | Integer | IPR(17)  | 0.1    | Intermediate print from continuous rod subroutine CNTROD                   |
|      | 58      | Integer | IPR(18)  | 0.1    | Not used                                                                  |
|      | 59      | Integer | IPR(19)  | 0.1    | Not used                                                                  |
|      | 60      | Integer | IPR(20)  | 0-9    | If IPR(20) > 0, all possible intermediate print will be printed out for all Monte Carlo encounters number IPR(20) or greater. Should be used with caution! |

Note: Card "C" is repeated for each variable being optimized or parameterized. See Table I-II.

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Name</th>
<th>Limits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-6</td>
<td>Alphanum</td>
<td>NAME(I)</td>
<td></td>
<td>Name of variable to be optimized or parameterized. See Table I-II.</td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>XMIN(I)</td>
<td></td>
<td>Minimum value of variable</td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>Decimal</td>
<td>XMAX(I)</td>
<td></td>
<td>Maximum value of variable</td>
</tr>
<tr>
<td></td>
<td>31-40</td>
<td>Decimal</td>
<td>X(I)</td>
<td></td>
<td>First estimate of variable in Optimization Mode: Delta X in parametric mode.</td>
</tr>
</tbody>
</table>

Note: Card "C" is repeated for each variable being optimized or parameterized.
<table>
<thead>
<tr>
<th>Columns</th>
<th>Name</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>TDX</td>
<td>Sec</td>
<td>Nominal delay time</td>
</tr>
<tr>
<td>1-6</td>
<td>SMISSX*</td>
<td>Ft</td>
<td>Standard deviation of guidance miss in X&quot; direction</td>
</tr>
<tr>
<td>1-6</td>
<td>SMISSZ</td>
<td>Ft</td>
<td>Standard deviation of guidance miss in Z&quot; direction</td>
</tr>
<tr>
<td>1-5</td>
<td>RMISS</td>
<td>Ft</td>
<td>Guidance miss distance in X&quot;, Z&quot; plane</td>
</tr>
<tr>
<td>1-5</td>
<td>RCOFF</td>
<td>Ft</td>
<td>Range cutoff on fuzing</td>
</tr>
<tr>
<td>1-5</td>
<td>ZONEL</td>
<td>Deg</td>
<td>Lower limit of first fragment spray zone in warhead</td>
</tr>
<tr>
<td>1-5</td>
<td>ZONEU</td>
<td>Deg</td>
<td>Upper limit of last fragment spray zone in warhead</td>
</tr>
<tr>
<td>1-5</td>
<td>V(I)</td>
<td>Ft/sec</td>
<td>Static fragment velocity, group I, I = 1, 10</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(I)</td>
<td>Grains</td>
<td>Fragment mass, group I, I = 1, 10</td>
</tr>
<tr>
<td>1-6</td>
<td>FRG(I)</td>
<td>Number of fragments in group I, I = 1, 10</td>
<td></td>
</tr>
<tr>
<td>1-2</td>
<td>AZ</td>
<td>Deg</td>
<td>Static azimuth approach angle</td>
</tr>
<tr>
<td>1-4</td>
<td>ELEV</td>
<td>Deg</td>
<td>Static elevation approach angle</td>
</tr>
<tr>
<td>1-4</td>
<td>AAAM</td>
<td>Deg</td>
<td>Mean azimuth angle of attack</td>
</tr>
<tr>
<td>1-4</td>
<td>EAAM</td>
<td>Deg</td>
<td>Mean elevation angle of attack</td>
</tr>
<tr>
<td>1-5</td>
<td>VMBAR</td>
<td>Ft/sec</td>
<td>Speed of interceptor</td>
</tr>
<tr>
<td>1-5</td>
<td>VTBAR</td>
<td>Ft/sec</td>
<td>Speed of target</td>
</tr>
<tr>
<td>1-6</td>
<td>BLTVEL</td>
<td>Ft/sec</td>
<td>Ejection velocity of bomblets</td>
</tr>
<tr>
<td>1-3</td>
<td>NSA</td>
<td>Deg</td>
<td>See Table I-V, location 1281</td>
</tr>
<tr>
<td>1-4</td>
<td>XMTB</td>
<td>Deg</td>
<td>See Table I-V, location 1271</td>
</tr>
<tr>
<td>1-6</td>
<td>AIM(I)</td>
<td>Ft</td>
<td>X coord of target aim point</td>
</tr>
<tr>
<td>1-6</td>
<td>AIM(2)</td>
<td>Ft</td>
<td>Y coord of target aim point</td>
</tr>
<tr>
<td>1-6</td>
<td>AIM(3)</td>
<td>Ft</td>
<td>Z coord of target aim point</td>
</tr>
<tr>
<td>1-6</td>
<td>BLTDT</td>
<td>Sec</td>
<td>Bomblet delay time</td>
</tr>
<tr>
<td>1-6</td>
<td>DMD(I)</td>
<td>In</td>
<td>Diameter of rod in group I, I = 1, 5</td>
</tr>
</tbody>
</table>

*If SMISSX and SMISSZ are to be varied independently, INDMSD (location 964) is set to zero (0). If SMISSX and SMISSZ are equal, INDMSD is set to one (1). See Figure I-1 for intercept geometry.

NOTE: All names in Table I-II must begin in Column 1 and must be in the exact format as indicated.
<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>1-72</td>
<td>Alphanumeric</td>
<td>TITLE</td>
<td></td>
<td></td>
<td>71</td>
<td>Warhead title</td>
</tr>
<tr>
<td>E</td>
<td>1-5</td>
<td>Integer</td>
<td>NOBSTR</td>
<td>0 ≤ 10</td>
<td></td>
<td>71</td>
<td>Number of obstructions on missile which may shield fragment spray</td>
</tr>
<tr>
<td>6-7</td>
<td>Integer</td>
<td>KRODS*</td>
<td>0 ≤ 5</td>
<td></td>
<td></td>
<td>46</td>
<td>Number of groups of continuous rods in the warhead</td>
</tr>
<tr>
<td>8</td>
<td>Logical</td>
<td>VETO*</td>
<td>Blank or T</td>
<td></td>
<td></td>
<td></td>
<td>&quot;T&quot; indicates the Monte Carlo routine accepts a sample P_k of 0 or 1, depending on whether the sum of P_k values from the spar contribution toward kill is ≤ CRASH or ≥ CRASH. ** Blank indicates the M.C. routine accepts the numerical value of this sum as the P_k for a given sample.</td>
</tr>
<tr>
<td>9-10</td>
<td>Integer</td>
<td>NPOLAR</td>
<td>0 ≤ 36</td>
<td>(See Note 4)</td>
<td></td>
<td>58</td>
<td>Number of polar zones required to describe the fragmentation data for warhead</td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>LL</td>
<td>lb</td>
<td></td>
<td></td>
<td>45</td>
<td>Explosive weight in equivalent pounds of TNT</td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>CD</td>
<td>Ft/sec</td>
<td></td>
<td></td>
<td>47</td>
<td>Coefficient of drag of fragments. (Must be input even if drag curve is also specified, INDCD)</td>
</tr>
<tr>
<td>31-40</td>
<td>Decimal</td>
<td>VMBAR</td>
<td>Ft/sec</td>
<td></td>
<td></td>
<td>38</td>
<td>Velocity of interceptor</td>
</tr>
<tr>
<td>41-50</td>
<td>Decimal</td>
<td>W</td>
<td>Ft</td>
<td></td>
<td></td>
<td></td>
<td>Separation between warhead and seeker measured along body axis of warhead. W is positive when warhead is behind seeker.</td>
</tr>
<tr>
<td>51-60</td>
<td>Decimal</td>
<td>SVF</td>
<td>Ft/sec</td>
<td></td>
<td></td>
<td>71</td>
<td>Standard deviation on static fragment velocity</td>
</tr>
<tr>
<td>61-70</td>
<td>Decimal</td>
<td>SIGOSC</td>
<td>Deg</td>
<td></td>
<td></td>
<td></td>
<td>Standard deviation of oscillation angle of obstruction</td>
</tr>
<tr>
<td>E</td>
<td>71</td>
<td>Integer</td>
<td>IFRAG</td>
<td>0 or 1</td>
<td></td>
<td></td>
<td>Fragment density indicator. If IFRAG = 0, the fragment density, FRAG(I,J), is input in fragments/steradian. If IFRAG = 1, the fragment density is input as the total number of fragments and the program computes fragments/steradian</td>
</tr>
<tr>
<td>72</td>
<td>Integer</td>
<td>INDCD*</td>
<td>0 ≤ 3</td>
<td></td>
<td></td>
<td></td>
<td>Indicates which drag curve is used to determine fragment impact velocity. 0 = constant C_p; 1 = sphere; 2 = cube; 3 = random</td>
</tr>
<tr>
<td>F</td>
<td>1-5</td>
<td>Integer</td>
<td>NGPS(I)</td>
<td>1 ≤ 10</td>
<td></td>
<td></td>
<td>Number of classes of fragmentation data in polar zone I. All zones must have same number of classes.</td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>THSU(I)</td>
<td>0 ≤ 180 Deg</td>
<td></td>
<td></td>
<td>59</td>
<td>Upper angle defining polar zone I for warhead</td>
</tr>
<tr>
<td>Card</td>
<td>Columns</td>
<td>Mode</td>
<td>Name</td>
<td>Limits</td>
<td>Units</td>
<td>Error Code</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>------------</td>
<td>--------</td>
<td>-------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>TSHL(I)</td>
<td>0-180 Deg</td>
<td>Deg</td>
<td>59</td>
<td>Lower angle defining polar zone I for warhead</td>
<td></td>
</tr>
<tr>
<td>G1</td>
<td>1-10</td>
<td>Decimal</td>
<td>FM(I,J)</td>
<td>Grains</td>
<td></td>
<td>Fragment mass for polar zone I and class J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>VF(I,J)</td>
<td>0-180 Ft/sec</td>
<td></td>
<td></td>
<td>Initial velocity for polar zone I and class J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>FPA(I,J)</td>
<td>0-180 In²</td>
<td></td>
<td></td>
<td>Fragment presented area for polar zone I and class J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31-40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Blank</td>
<td></td>
</tr>
<tr>
<td>41-50</td>
<td>Decimal</td>
<td>FRAG(I,J)</td>
<td>0-180 Decimal</td>
<td></td>
<td></td>
<td>Fragment density or number of fragments for polar zone I and class J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51-60</td>
<td>Decimal</td>
<td>D1(J)**</td>
<td>0-180 Inches</td>
<td></td>
<td></td>
<td>Width of rectangular fragments in class J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>61-70</td>
<td>Decimal</td>
<td>D2(J)**</td>
<td>0-180 Inches</td>
<td></td>
<td></td>
<td>Height of rectangular fragments in class J</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G2</td>
<td>1-10</td>
<td>Decimal</td>
<td>VO(I)</td>
<td>Ft/sec</td>
<td></td>
<td>Initial rod velocity of Group I</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>AR(I)</td>
<td>0-180 Degree</td>
<td></td>
<td></td>
<td>Initial flyoff angle for rods in group I - measured from the missile</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>centerline</td>
<td></td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>PROD(I)</td>
<td>0-180 Lb/ft³</td>
<td></td>
<td></td>
<td>Density of rod material in group I</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31-40</td>
<td>Decimal</td>
<td>DRODX(I)</td>
<td>0-180 Feet</td>
<td></td>
<td></td>
<td>Rod Diameter</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41-50</td>
<td>Decimal</td>
<td>RMAX(I)</td>
<td>0-180 Feet</td>
<td></td>
<td></td>
<td>Maximum rod radius in group I at which BREAK2 occurs</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51-55</td>
<td>Decimal</td>
<td>BREAK1(I)</td>
<td>0-180 Integer</td>
<td></td>
<td></td>
<td>Initial number of breaks in rod in group I</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56-60</td>
<td>Decimal</td>
<td>BREAK2(I)</td>
<td>0-180 Integer</td>
<td></td>
<td></td>
<td>Number of breaks in rod that occur at RMAX(I) in group I</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>61-65</td>
<td>Decimal</td>
<td>DIAS(I)**</td>
<td>0-180 Integer</td>
<td></td>
<td></td>
<td>Integral number of diameters at which the continuous rod evaluation is</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>terminated</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>66-70</td>
<td>Integer</td>
<td>NRODS(I)</td>
<td>0-360 Degree</td>
<td></td>
<td></td>
<td>Number of rods in group I</td>
<td></td>
</tr>
<tr>
<td>1-10</td>
<td>Decimal</td>
<td>OSMN(I)</td>
<td>0-360 Degree</td>
<td></td>
<td></td>
<td>Mean oscillation angle of obstruction measured from the negative Z₁ axis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>in the roll plane of the interceptor coordinate system</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11-25</td>
<td>Decimal</td>
<td>ASHD(I)</td>
<td>0-360 Degree</td>
<td></td>
<td></td>
<td>Arc shielded by obstruction I (in roll plane)</td>
<td></td>
</tr>
</tbody>
</table>

42
### TABLE I-III---CONCLUDED

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>THET1(I)</td>
<td>0 ≤ 180</td>
<td>Degree</td>
<td>71</td>
<td>Lower polar zone angle that bounds obstruction I in vertical plane (measured off longitudinal axis of interceptor)</td>
<td></td>
</tr>
<tr>
<td>31-40</td>
<td>Decimal</td>
<td>THET2(I)</td>
<td>0 ≤ 180</td>
<td>Degree</td>
<td>71</td>
<td>Upper polar zone angle that bounds obstruction I in vertical plane (measured off longitudinal axis of interceptor)</td>
<td></td>
</tr>
</tbody>
</table>

*For continuous rod warhead only.

**See location 1717 in continuous rod target data.

#If INDCD > 0, then VSOUND (location 1074) must be input.

*Not required if fragment mass is not being optimized or parameterized.

*To be used if rod is discontinuous at warhead initiation.

One diameter equal 2 * RMAX(I).

Notes:
(1) Card F and G1 are repeated for each polar zone.
(2) Card G1 is repeated NGPS(I) times.
(3) Card G2 is repeated KRODS times.
(4) NPOLAR must = 0 when KRODS > 0, and cards F and G1 are omitted.
(5) Card H is repeated NOBSTR times.
<table>
<thead>
<tr>
<th>Memory Location</th>
<th>Variable Name</th>
<th>Limits</th>
<th>Error Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NIT</td>
<td>&gt;0</td>
<td>69</td>
<td>Maximum number of Monte Carlo interceptions</td>
</tr>
<tr>
<td>976</td>
<td>NMIN</td>
<td>&gt;0</td>
<td>69</td>
<td>Minimum number of Monte Carlo interceptions</td>
</tr>
<tr>
<td>978</td>
<td>NGRID</td>
<td>0 &lt; 20</td>
<td>72</td>
<td>Size of square matrix of W/H location points at burst</td>
</tr>
<tr>
<td>1649</td>
<td>XINTEV</td>
<td>0–1</td>
<td>1</td>
<td>Maximum acceptable standard deviation of sample</td>
</tr>
<tr>
<td>2</td>
<td>NFAAC</td>
<td>2–18</td>
<td></td>
<td>Number of points in azimuth approach angle table PAZ(I) versus AZ(I). (See location 1001 and 1019).</td>
</tr>
<tr>
<td>3</td>
<td>SPEAC</td>
<td></td>
<td></td>
<td>Number of points in elevation approach angle table PELEV(I) versus ELEV(I). (See locations 1037 and 1055).</td>
</tr>
<tr>
<td>4</td>
<td>INDTC</td>
<td>-1,0,1</td>
<td></td>
<td>Indicates how to compute the target aim point. If INDTC = 0, uses PCPX and PCPZ. If INDTC = 1, uses the average of the homing radar reflection points. If INDTC = -1, uses AIM(1), AIM(2), and AIM(3).</td>
</tr>
<tr>
<td>797</td>
<td>INDMD</td>
<td>0,1</td>
<td></td>
<td>Indicates how to compute miss distance with respect to the aim point. If INDMD = 0, uses SMISSX and SMISSZ. If INDMD = 1, uses RMISS.</td>
</tr>
<tr>
<td>964</td>
<td>INDMSD</td>
<td>0,1</td>
<td></td>
<td>Indicates how SMISSX and SMISSZ are varied in parametric and optimization mode. If INDMSD = 0, SMISSX and SMISSZ are varied independently. If INDMSD = 1, SMISSX and SMISSZ are equal.</td>
</tr>
<tr>
<td>1474</td>
<td>AAAM</td>
<td>0–360</td>
<td>Degree</td>
<td>Mean azimuth angle of attack. (Yaw)</td>
</tr>
<tr>
<td>1076</td>
<td>AOAa</td>
<td>0–180</td>
<td>Degree</td>
<td>The one sigma value on azimuth angle AAAM.</td>
</tr>
<tr>
<td>1475</td>
<td>SAAAM</td>
<td>0–360</td>
<td>Degree</td>
<td>Mean elevation angle of attack. (Pitch)</td>
</tr>
<tr>
<td>1077</td>
<td>ECAa</td>
<td>0–180</td>
<td>Degree</td>
<td>The one sigma value on elevation angle of attack</td>
</tr>
<tr>
<td>1076</td>
<td>PCPX</td>
<td>Feet</td>
<td></td>
<td>The X&quot; coordinate of the aim point. Used only if INDTC = 0</td>
</tr>
<tr>
<td>1079</td>
<td>PCPZ</td>
<td>Feet</td>
<td></td>
<td>The Z&quot; coordinate of the aim point. Used only if INDTC = -1.</td>
</tr>
<tr>
<td>1646</td>
<td>AIM(3)</td>
<td>Feet</td>
<td></td>
<td>The X, Y, Z coordinates of the target aim point. Used only when INDTC = -1.</td>
</tr>
<tr>
<td>1472</td>
<td>KNISSM</td>
<td>Feet</td>
<td></td>
<td>Mean guidance miss in X&quot; direction. Used only when INDMD = 0.</td>
</tr>
<tr>
<td>1280</td>
<td>SMISSX</td>
<td>Feet</td>
<td></td>
<td>Standard deviation of guidance miss in X&quot; direction. Used only when INDMD = 0.</td>
</tr>
<tr>
<td>1473</td>
<td>ZMISSM</td>
<td>Feet</td>
<td></td>
<td>Mean guidance miss in Z&quot; direction. Used only when INDMD = 0.</td>
</tr>
<tr>
<td>1081</td>
<td>SMISSZ</td>
<td>Feet</td>
<td></td>
<td>Standard deviation of guidance miss in Z&quot; direction. Used only when INDMD = 0.</td>
</tr>
<tr>
<td>1082</td>
<td>RMISS</td>
<td>Feet</td>
<td></td>
<td>Guidance miss distance in X&quot;, Z&quot; plane. Used only when INDMD = 1.</td>
</tr>
<tr>
<td>1578</td>
<td>XPP</td>
<td>0–1</td>
<td>Atmospheres</td>
<td>Ratio of ambient air pressure at intercept altitude to pressure at sea level</td>
</tr>
<tr>
<td>1579</td>
<td>RHO</td>
<td>0</td>
<td>lb/ft³</td>
<td>Air density at intercept altitude</td>
</tr>
<tr>
<td>1074</td>
<td>VSOUND</td>
<td>0</td>
<td>ft/sec</td>
<td>Velocity of sound at intercept altitude</td>
</tr>
<tr>
<td>1201</td>
<td>PAZ(18)</td>
<td>0–1</td>
<td></td>
<td>PAZ(18) is the probability that the static azimuth approach angle is between AZ(I-1) and AZ(I). AZ(0) is assumed to be zero.</td>
</tr>
<tr>
<td>1019</td>
<td>AZ(18)</td>
<td>0–180</td>
<td>Degree</td>
<td>33 See PAZ(18) above</td>
</tr>
<tr>
<td>1037</td>
<td>PELEV(18)</td>
<td>0–1</td>
<td></td>
<td>PELEV(18) is the probability that the static elevation approach angle is between ELEV(I-1) and ELEV(I). ELEV(0) is assumed to be -90 degrees</td>
</tr>
<tr>
<td>1555</td>
<td>ELEV(18)</td>
<td>-90 to +90</td>
<td>Degree</td>
<td>35 See PELEV(18) above</td>
</tr>
</tbody>
</table>

*Normal distribution function used.
### TABLE I-V. FUZING

<table>
<thead>
<tr>
<th>Location</th>
<th>Variable Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>799</td>
<td>INDDT</td>
<td>0,1,-1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>977</td>
<td>ISSFUZ*</td>
<td>0,1</td>
<td></td>
<td></td>
<td>1 indicates spherical fuze; 0 indicates a single fuze cone</td>
</tr>
<tr>
<td>1249</td>
<td>TDX</td>
<td>&gt;0</td>
<td>seconds</td>
<td>41</td>
<td>Nominal fuze delay time. Used only when INDDT = 0</td>
</tr>
<tr>
<td>1250</td>
<td>SFMF</td>
<td></td>
<td>seconds</td>
<td></td>
<td>Standard deviation on TDX. Used only when INDDT = 1</td>
</tr>
<tr>
<td>1508</td>
<td>VRBART(I)</td>
<td>ft/sec</td>
<td></td>
<td>55</td>
<td>VRBART(I) versus TDT(I) is a table of fuze delay time versus estimated closing velocity. Used only when INDDT = 1</td>
</tr>
<tr>
<td>1518</td>
<td>TDT(I)</td>
<td>seconds</td>
<td></td>
<td></td>
<td>See VRBART(I) above</td>
</tr>
<tr>
<td>1075</td>
<td>SIGMAR</td>
<td>ft/sec</td>
<td></td>
<td></td>
<td>The one sigma variation on VRBART, the measured relative speed of warhead and target. Used only when INDDT = 1</td>
</tr>
<tr>
<td>804</td>
<td>NVRBER</td>
<td>2-10</td>
<td></td>
<td>12</td>
<td>Number of entries in table VRBART(I) vs TDT(I). Used only when INDDT = 1</td>
</tr>
<tr>
<td>1248</td>
<td>SDE</td>
<td>&gt;0</td>
<td>seconds</td>
<td>68</td>
<td>Time to wait after fuizing before beginning search for delay time. Used only if INDDT = 1</td>
</tr>
<tr>
<td>1645</td>
<td>DTINC</td>
<td>&gt;0</td>
<td>seconds</td>
<td>65</td>
<td>Time increment used in procedure to select the delay time when INDDT = -1</td>
</tr>
<tr>
<td>975</td>
<td>MNTODT</td>
<td>&gt;0</td>
<td></td>
<td>67</td>
<td>The maximum number of time increments allowed in selecting the delay time when INDDT = -1</td>
</tr>
<tr>
<td>1281</td>
<td>HSA</td>
<td>0-180</td>
<td>degree</td>
<td>66</td>
<td>The angle in degrees between missile body axis and the fragments which are supposed to hit the target at the aim point specified by INDTC. Used only when INDDT = -1</td>
</tr>
<tr>
<td>1251</td>
<td>VRTB(I)</td>
<td>ft/sec</td>
<td></td>
<td>49</td>
<td>VRTB(I) is the closing velocity which causes the fuze cone angle to be OMTB(I) with standard deviation SIGTB(I).</td>
</tr>
<tr>
<td>1261</td>
<td>SIGTB(I)</td>
<td>degree</td>
<td></td>
<td></td>
<td>See VRTB(I) above</td>
</tr>
<tr>
<td>1271</td>
<td>SMTB(I)</td>
<td>0-180</td>
<td>degree</td>
<td>50</td>
<td>See VRTB(I) above</td>
</tr>
<tr>
<td>800</td>
<td>VNR</td>
<td>2-10</td>
<td></td>
<td>8</td>
<td>Number of entries in table VRTB(I) versus SIGTB(I) and SMTB(I).</td>
</tr>
<tr>
<td>1083</td>
<td>RCOFF</td>
<td>&gt;0</td>
<td>feet</td>
<td>39</td>
<td>If ISSFUZ = 0, RCOFF is the range cutoff on fuizing, i.e., if a fuze point enters the fuze cone at a range greater than RCOFF, no fuizing on that particular fuze point occurs; If ISSFUZ = 1, RCOFF is the functional fuze distance</td>
</tr>
<tr>
<td>Location</td>
<td>Variable Name</td>
<td>Limits</td>
<td>Units</td>
<td>Error Code</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>---------------</td>
<td>--------</td>
<td>------------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>1507</td>
<td>DD</td>
<td>20</td>
<td>seconds</td>
<td>43</td>
<td>Step size in delay time distribution table printout. Used only when INDDT = -1</td>
</tr>
<tr>
<td>798</td>
<td>NLOBES</td>
<td>0-7</td>
<td></td>
<td>7</td>
<td>The number of lobes (including center lobe) of the fuze cone receiver antenna. This number is used only if the target is using fuze jamming equipment</td>
</tr>
<tr>
<td>1224</td>
<td>JAMMER(3)</td>
<td></td>
<td>feet</td>
<td></td>
<td>JAMMER(1), JAMMER(2) and JAMMER(3) are the X, Y, Z coordinates of the fuze jamming jammer being used by the target. Used only if NLOBES &gt; 0.</td>
</tr>
<tr>
<td>1227</td>
<td>ANGJAM(7)</td>
<td>0-180</td>
<td>degree</td>
<td></td>
<td>ANGJAM(1) is the angle (measured from the main fuze cone lobe) of the ith lobe of the fuze cone receiving antenna. Normally one value of ANGJAM should be zero to account for the main lobe. Used only if NLOBES &gt; 0.</td>
</tr>
<tr>
<td>1234</td>
<td>FACTOR(7)</td>
<td></td>
<td></td>
<td></td>
<td>A number associated with the jamming power received by the fuze cone receiver. Used only if NLOBES &gt; 0.</td>
</tr>
<tr>
<td>1241</td>
<td>RECSEN(7)</td>
<td></td>
<td></td>
<td></td>
<td>A number associated with the fuze cone receiver sensitivity. Used only when NLOBES &gt; 0</td>
</tr>
<tr>
<td>989</td>
<td>I2FC</td>
<td>0.1</td>
<td>seconds</td>
<td>49</td>
<td>1 indicates two fuze cones are used</td>
</tr>
<tr>
<td>1650</td>
<td>FTMIN</td>
<td></td>
<td>seconds</td>
<td></td>
<td>Minimum fuze time when using I2FC indicator</td>
</tr>
<tr>
<td>1651</td>
<td>FTMAX</td>
<td></td>
<td>seconds</td>
<td></td>
<td>Maximum fuze time when using I2FC indicator</td>
</tr>
<tr>
<td>1652</td>
<td>OMGMIN</td>
<td>0-180</td>
<td>degree</td>
<td></td>
<td>Minimum fuze cone half angle when using I2FC indicator</td>
</tr>
<tr>
<td>1653</td>
<td>OMGMAX</td>
<td>0-180</td>
<td>degree</td>
<td></td>
<td>Maximum fuze cone half angle when using I2FC indicator</td>
</tr>
</tbody>
</table>

*Spherical fuze functions when the distance from the warhead to target is equal to RCOFF (See location 1081).

*A third option is available for special conical type fuzes; see location 989.

FACTOR(1) = \( \frac{Pj \cdot Gj \cdot A}{Bj \cdot R} \)

where:
- \( Pj \) = jamming power - watts
- \( Gj \) = jammer antenna gain
- \( A = \) area of receiver antenna aperture - \( \text{ft}^2 \)
- \( Bj \) = bandwidth of jammer - \( \text{mc} \)
- \( R \) = slant range between jammer and receiver - ft.

RECSEN(1) = \( K \cdot T \cdot NF \cdot (S/N) \cdot G_r \cdot L_r \)

where:
- \( K = \) Boltzman's constant = \( 1.38 \times 10^{-23} \) joules/°K
- \( T = \) °Kelvin
- \( NF = \) noise factor
- \( S/N = \) signal to noise factor of the receiver
- \( G_r = \) Receiver antenna gain for either main or side lobes
- \( L_r = \) Receiver losses.

Note: If the time required for the target to pass between the two fuze points (one on each cone) does not fall between FTMIN and FTMAX, no fuze occurs.
<table>
<thead>
<tr>
<th>Location</th>
<th>Variable Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1073</td>
<td>VTBAR</td>
<td>&gt;0</td>
<td>Ft/sec</td>
<td>37</td>
<td>Speed of the target</td>
</tr>
<tr>
<td>801</td>
<td>NEDH</td>
<td>0-10</td>
<td>Feet</td>
<td>9</td>
<td>Number of ellipsoids which describe the physical shape and dimensions of the target, i.e., &quot;direct hit&quot; ellipsoids</td>
</tr>
<tr>
<td>1412</td>
<td>EDH (6,10)</td>
<td></td>
<td>Feet</td>
<td>53</td>
<td>EDH(1,I), EDH(2,I) and EDH(3,I) are the X, Y, Z coordinates of the Ith direct hit ellipsoids. EDH(4,I), EDH(5,I) and EDH(6,I) are the three semi-axes of the Ith ellipsoid.</td>
</tr>
<tr>
<td>805</td>
<td>NCB</td>
<td>0-10</td>
<td></td>
<td>13</td>
<td>Number of blast centers associated with the blast kill ellipsoids.</td>
</tr>
<tr>
<td>1282</td>
<td>PCB (4,10)</td>
<td></td>
<td>Feet</td>
<td>51</td>
<td>PCB(1,I), PCB(2,I), and PCB(3,I) are the X, Y, Z coordinates of the Ith blast center. PCB(4,I) is the number of blast ellipsoids associated with the Ith blast center.</td>
</tr>
<tr>
<td>1322</td>
<td>EDB (6,15)</td>
<td></td>
<td>Feet</td>
<td>52</td>
<td>EDB(1,I), EDB(2,I) and EDB(3,I) are the X, Y, Z coordinates of the Ith blast ellipsoid. EDB(4,I), EDB(5,I) and EDB(6,I) are the three semi-axes of the Ith blast ellipsoid.</td>
</tr>
<tr>
<td>806</td>
<td>NPBT</td>
<td>2-25</td>
<td></td>
<td>14</td>
<td>Number of entries in TAB12Y(I) versus TAB12X(I). Used only if NCB &gt; 0.</td>
</tr>
<tr>
<td>1553</td>
<td>TAB12Y (25)</td>
<td></td>
<td>Milliseconds</td>
<td>56</td>
<td>TAB12Y(I) is the time required for the blast wave to travel a distance TAB12X(I) if the ambient pressure is one atmosphere and weight of explosive is one lb TNT. Used only if NCB &gt; 0.</td>
</tr>
<tr>
<td>1528</td>
<td>TAB12X (25)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>See TAB12Y(25) above.</td>
</tr>
<tr>
<td>807</td>
<td>NSHE</td>
<td>0-10</td>
<td></td>
<td>15</td>
<td>Number of shielding ellipsoids.</td>
</tr>
<tr>
<td>1580</td>
<td>SHIELD (6,10)</td>
<td></td>
<td>Feet</td>
<td>57</td>
<td>SHIELD(1,I), SHIELD(2,I) and SHIELD(3,I) are the X, Y, Z coordinates of the center of the Ith shielding ellipsoid. SHIELD(4,I), SHIELD(5,I) and SHIELD(6,I) are the semi-axes of the Ith shielding ellipsoid. Used only when NSHE &gt; 0.</td>
</tr>
<tr>
<td>1084</td>
<td>H(3,20)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>H(1,I), H(2,I) and H(3,I) are the X, Y, Z coordinates of the Ith homing radar reflection point. Used only when INDTC = 1.</td>
</tr>
<tr>
<td>5</td>
<td>NHP(6,6)</td>
<td>0-10</td>
<td></td>
<td>3</td>
<td>NHP(I,J) is the number of homing radar reflection points visible in the Ith PSI sub-range and the Jth LAMBDA sub-range. Used only when INDTC = 1.</td>
</tr>
<tr>
<td>41</td>
<td>NH (10,6,6)</td>
<td>1-20</td>
<td></td>
<td>4</td>
<td>NH(K,I,J) is the serial number of the Kth homing radar reflection point visible in the Ith PSI range and the Jth LAMBDA range, i.e., if NH(2,5,4) = 3, then a homing radar reflection point with coordinates H(X,3), H(Y,3) and H(Z,3) is the second of NHP(5,4) such points visible for 120 deg ≤ PSI ≤ 150 deg and 0 ≤ LAMBDA ≤ 30 deg. Used only when INDTC = 1.</td>
</tr>
<tr>
<td>Location</td>
<td>Variable</td>
<td>Limits</td>
<td>Units</td>
<td>Definition</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>--------</td>
<td>-------</td>
<td>------------</td>
<td></td>
</tr>
<tr>
<td>1164</td>
<td>F(1,20)</td>
<td>Feet</td>
<td></td>
<td>F(1,I), F(2,I) and F(3,I) are the X, Y, Z coordinates of the Ith fuze glitter point.</td>
<td></td>
</tr>
<tr>
<td>1144</td>
<td>PGP(20)</td>
<td></td>
<td></td>
<td>PGP(I) is the probability that the Ith fuze point will cause fuzing, assuming the fuze point enters the fuze cone within a range RCOFF feet. PGP(I) is independent of approach angle.</td>
<td></td>
</tr>
<tr>
<td>401</td>
<td>NNF(6,6)</td>
<td>0-10</td>
<td></td>
<td>NNF(I,J) is the number of fuze points visible in the Ith PSI range and the Jth LAMBDA range.</td>
<td></td>
</tr>
<tr>
<td>437</td>
<td>SF</td>
<td>1-20</td>
<td></td>
<td>NF(K,I,J) is the serial number of the Kth fuze point visible in the Ith PSI range and the Jth LAMBDA range. This is similar to NH(K,I,J).</td>
<td></td>
</tr>
<tr>
<td>802</td>
<td>NFW(6,6)</td>
<td>0-10</td>
<td></td>
<td>Number of points on the target which are vulnerable to fragments.</td>
<td></td>
</tr>
<tr>
<td>1476</td>
<td>PV(3,10)</td>
<td>Feet</td>
<td></td>
<td>PV(1,I), PV(2,I) and PV(3,I) are the X, Y, Z coordinates of the Ith vulnerable area. Used only when NPV ( \neq 0 ).</td>
<td></td>
</tr>
<tr>
<td>842</td>
<td>NGROUP</td>
<td>1-10</td>
<td></td>
<td>Number of vulnerable point groups.</td>
<td></td>
</tr>
<tr>
<td>843</td>
<td>NCOM(10)</td>
<td>0</td>
<td></td>
<td>NCOM(I) is the number of vulnerable points in the Ith vulnerable point group.</td>
<td></td>
</tr>
<tr>
<td>851</td>
<td>NPVCOM(10,10)</td>
<td></td>
<td></td>
<td>NPVCOM(I,J) is the serial number of the Ith vulnerable point in the Jth group.</td>
<td></td>
</tr>
<tr>
<td>953</td>
<td>KILL(10)</td>
<td>0</td>
<td></td>
<td>KILL(I) is the number of vulnerable points in the Ith group which must be killed to kill the target.</td>
<td></td>
</tr>
<tr>
<td>808</td>
<td>IAV</td>
<td></td>
<td></td>
<td>Indicates the type of fragment vulnerability data to use. If IAV ( &lt; 0 ), program uses symmetrical target (missile) vulnerability data. If IAV ( = 0 ), program uses box type target vulnerability data. If IAV ( &gt; 0 ), program uses nonsymmetrical target vulnerability data.</td>
<td></td>
</tr>
<tr>
<td>965</td>
<td>NFH(10)</td>
<td>1-5</td>
<td></td>
<td>Number of fragment masses used to describe the vulnerability of target point I.</td>
<td></td>
</tr>
<tr>
<td>1751</td>
<td>FMAS(5,10)</td>
<td>Grains</td>
<td></td>
<td>As ascending ordered table of fragment masses used in the vulnerability data for point I.</td>
<td></td>
</tr>
<tr>
<td>840</td>
<td>NPEVLT(5,10)</td>
<td>2-10</td>
<td></td>
<td>Number of values of strike velocity, VSBART(I), used in target vulnerability table. Used only when IAV ( \leq 0 ).</td>
<td></td>
</tr>
<tr>
<td>1741</td>
<td>VSBART(10)</td>
<td>Ft/sec</td>
<td></td>
<td>VSBART(I) is the Ith value of fragment strike velocity used to describe the target vulnerability data. Used only when IAV ( \leq 0 ).</td>
<td></td>
</tr>
<tr>
<td>841</td>
<td>NFANG(19)</td>
<td>2-19</td>
<td></td>
<td>Number of values of polar angle, AVANGT(J), used in target vulnerability table. Used only when IAV ( &lt; 0 ).</td>
<td></td>
</tr>
<tr>
<td>1801</td>
<td>AVANGT(19)</td>
<td>Degree</td>
<td></td>
<td>AVANGT(J) is the Jth value of polar angle used to describe the target vulnerability data. Used only when IAV ( &lt; 0 ).</td>
<td></td>
</tr>
<tr>
<td>Location</td>
<td>Variable Name</td>
<td>Limits</td>
<td>Units</td>
<td>Error Code</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>---------------</td>
<td>--------</td>
<td>-------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>2001</td>
<td>AVTM</td>
<td>&gt;0</td>
<td>Sq ft</td>
<td>24</td>
<td>AVTM(I,J,K) is the vulnerable area of the target when a fragment of mass FMAS(K,I) with strike velocity VSBART(I) strikes the target at an angle of AVANGT(J) degrees. Used only when IAV &lt; 0.</td>
</tr>
<tr>
<td>1801</td>
<td>AVRT</td>
<td>&gt;0</td>
<td>Sq ft</td>
<td>27</td>
<td>AVRT(I,K,N) is the vulnerable area on the rear side of the Nth vulnerable point PV(N) to a fragment of mass FMAS(K,I) with strike velocity VSBART(I). Used only when IAV = 0.</td>
</tr>
<tr>
<td>2401</td>
<td>AVFT</td>
<td>&gt;</td>
<td>Sq ft</td>
<td>27</td>
<td>AVFT(I,K,N) is the vulnerable area on the front side of the Nth vulnerable point PV(N) to a fragment of mass FMAS(K,I) with strike velocity VSBART(I). Used only when IAV = 0.</td>
</tr>
<tr>
<td>2901</td>
<td>AVBT</td>
<td>&gt;0</td>
<td>Sq ft</td>
<td>27</td>
<td>AVBT(I,K,N) is the vulnerable area on the bottom side of the Nth vulnerable point PV(N) to a fragment of mass FMAS(K,I) with strike velocity VSBART(I). Used only when IAV = 0.</td>
</tr>
<tr>
<td>3401</td>
<td>AVTT</td>
<td>&gt;0</td>
<td>Sq ft</td>
<td>27</td>
<td>AVTT(I,K,N) is the vulnerable area on the top side of the Nth vulnerable point PV(N) to a fragment of mass FMAS(K,I) with strike velocity VSBART(I). Used only when IAV = 0.</td>
</tr>
<tr>
<td>3901</td>
<td>AVRST</td>
<td>&gt;0</td>
<td>Sq ft</td>
<td>27</td>
<td>AVRST(I,K,N) is the vulnerable area on the right side of the Nth vulnerable point PV(N) to a fragment of mass FMAS(K,I) with strike velocity VSBART(I). Used only when IAV = 0.</td>
</tr>
<tr>
<td>4401</td>
<td>AVLST</td>
<td>&gt;0</td>
<td>Sq ft</td>
<td>27</td>
<td>AVLST(I,K,N) is the vulnerable area on the left side of the Nth vulnerable point PV(N) to a fragment of mass FMAS(K,I) with strike velocity VSBART(I). Used only when IAV = 0.</td>
</tr>
<tr>
<td>810</td>
<td>NPA(I)</td>
<td>2-20</td>
<td></td>
<td>16</td>
<td>NPA(I) is the number of polar angles POLAT(I,N) used to describe the target vulnerability data. Used only when IAV &gt; 0.</td>
</tr>
<tr>
<td>1801</td>
<td>POLAT</td>
<td>Degree</td>
<td></td>
<td>17</td>
<td>POLAT(I,N) is the Ith polar angle A used to describe the vulnerability of the Nth vulnerable point. Used only when IAV &gt; 0.</td>
</tr>
<tr>
<td>820</td>
<td>NPB(I)</td>
<td>2-20</td>
<td></td>
<td>16</td>
<td>NPB(I) is the number of polar angles POLBT(I,N) used to describe the target vulnerability data. Used only when IAV &gt; 0.</td>
</tr>
<tr>
<td>2001</td>
<td>POLBT</td>
<td>Degree</td>
<td></td>
<td>18</td>
<td>POLBT(I,N) is the Ith polar angle B used to describe the vulnerability of the Nth vulnerable point. Used only when IAV &gt; 0.</td>
</tr>
<tr>
<td>830</td>
<td>NDV(I)</td>
<td>2-20</td>
<td></td>
<td>16</td>
<td>NDV(I) is the number of values of strike velocity DVTAB(I,N) used to describe the target vulnerability data. Used only when IAV &gt; 0.</td>
</tr>
<tr>
<td>2201</td>
<td>DVTAB</td>
<td>Ft/sec</td>
<td></td>
<td>19</td>
<td>DVTAB(I,N) is the Ith strike velocity used to describe the vulnerability of the Nth vulnerable point. Used only when IAV &gt; 0.</td>
</tr>
</tbody>
</table>
### TABLE I-VI---CONCLUDED

<table>
<thead>
<tr>
<th>Location</th>
<th>Variable Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>2401</td>
<td>AVTAB</td>
<td>&gt;0</td>
<td></td>
<td>21</td>
<td>AVTAB(I,N) is the value of vulnerable area of the Nth vulnerability points corresponding to particular values of POLAT, POLBT, and DVTAB. Used only when IAV &gt; 0.</td>
</tr>
<tr>
<td>809</td>
<td>IAVOP</td>
<td></td>
<td></td>
<td></td>
<td>Indicator (0 or non zero). Indicates the way fragment strike angles are measured when IAV &gt; 0. (See Figure 1-4).</td>
</tr>
<tr>
<td>1506</td>
<td>Q</td>
<td>&gt;0</td>
<td>Seconds</td>
<td>42</td>
<td>A time in seconds used in the process of determining if and when fragments strike a vulnerable area. Q is the maximum time after warhead detonation that a fragment can strike a vulnerable point, i.e., any fragment striking a vulnerable point more than Q seconds after detonation are ignored.</td>
</tr>
<tr>
<td>1640</td>
<td>TTOL</td>
<td>&gt;10^-10</td>
<td>Seconds</td>
<td>60</td>
<td>Relative error criterion used in the iteration procedure for determining fragment hits. If a vulnerable point and a fragment are at a particular point in space at a time whose relative difference is less than TTOL, the vulnerable point is considered to have been hit by the fragment.</td>
</tr>
<tr>
<td>1641</td>
<td>YTOL</td>
<td>&lt;0.1</td>
<td>Feet</td>
<td>61</td>
<td>The distance tolerance used in the iteration procedure for determining fragment hits on a vulnerable point. If a fragment passes within YTOL feet of a vulnerable point, the vulnerable point is considered to have been hit by the fragment.</td>
</tr>
<tr>
<td>1720</td>
<td>PKHIT</td>
<td>&lt;1</td>
<td></td>
<td></td>
<td>Probability of kill given a direct hit.</td>
</tr>
</tbody>
</table>

*If IAV < 0, NPV ≤ 1.*

*Measured off target (missile center axis).*

*If DT/TFRAG ≤ TTOL the vulnerable point is considered to have been hit by the fragment.*

DT = Time difference between arrival of vulnerable point and fragment.

TFRAG = Total iteration time from detonation.

**Notes:**

1. Reflection points are sequentially assigned serial numbers from 1 to 20.
2. PSI sub-range (1) limits are 0 to 30 degrees.
   LAMBDA sub-range (1) limits are -90 to -60 degrees. The entire range of PSI and LAMBDA values is divided into similar 30 degree increments, forming 36 zones. See Figure 1-2, page 51.
3. PSI ranges from 0-180 degrees and LAMBDA goes from -90 to +90 degrees.
4. The two options for input of POLAT and POLBT are determined by IAVOP (See Figure 1-3, page 52).
5. The product of NFM(I), NPA(I), NPB(I), and HDV(J) plus their sum cannot be greater than 1000.
Notes:
1) \( Z(l,J) \) is Defined by PSI(I), LAMBDA(J)
2) PSI(I) Runs from 0 to 30 Degrees
   LAMBDA(1) Runs from -90 to -60 Degrees
   LAMBDA(2) Runs from -60 to -30 Degrees
   ...
3) The Natural Order of Zones, \( Z(I,J) \), as Read into the Computer is \( Z(1,1), Z(2,1), Z(3,1) \ldots \)
   \( Z(6,1), Z(1,2), Z(2,2) \ldots \)

Figure 1-2. Target Subrange Geometry
DVTAB = Strike Velocity of Fragments

Geometry When IAV > 0 and IAVAOP = 0

\[ 0 \leq \text{POLAT} < 2\pi \quad -\frac{\pi}{2} \leq \text{POLBT} < \frac{\pi}{2} \]

Geometry When IAV > 0 and IAVAOP ≠ 0

\[ 0 \leq \text{POLAT} < \pi \quad 0 \leq \text{POLBT} < 2\pi \]

Figure I-3. Geometry of Fragment Vulnerability Data Options
<table>
<thead>
<tr>
<th>Location</th>
<th>Variable Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>983</td>
<td>NSPRDM</td>
<td>1-5</td>
<td></td>
<td>74</td>
<td>Number of rod diameters in spar data</td>
</tr>
<tr>
<td>1741</td>
<td>SPARM(5)</td>
<td></td>
<td></td>
<td>75</td>
<td>Diameter of rods corresponding to spar kill (SPAR(10,1) through SPAR(16,1))</td>
</tr>
<tr>
<td>984</td>
<td>ISPAR</td>
<td>1-50</td>
<td></td>
<td></td>
<td>Number of SPARS or lines required to describe the target</td>
</tr>
<tr>
<td>985</td>
<td>LREJCT</td>
<td>1-20</td>
<td></td>
<td>76</td>
<td>Total number of rod rejection equations used in target description</td>
</tr>
<tr>
<td>2401</td>
<td>REJECT(1,1)</td>
<td></td>
<td></td>
<td></td>
<td>Empirical constant for reject velocity equation</td>
</tr>
<tr>
<td></td>
<td>(See Note 1)</td>
<td></td>
<td></td>
<td></td>
<td>Minimum velocity at all strike angles which will result in the target being killed</td>
</tr>
<tr>
<td></td>
<td>REJECT(2,1)</td>
<td></td>
<td></td>
<td></td>
<td>Empirical constant for reject velocity equation</td>
</tr>
<tr>
<td></td>
<td>REJECT(3,1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4401</td>
<td>SPAR(16,50)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SPAR(1,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Direction cosine of the SPAR with respect to the X axis</td>
</tr>
<tr>
<td></td>
<td>SPAR(2,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Direction cosine of the SPAR with respect to the Y axis</td>
</tr>
<tr>
<td></td>
<td>SPAR(3,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Direction cosine of the SPAR with respect to the Z axis</td>
</tr>
<tr>
<td></td>
<td>SPAR(4,1)*</td>
<td></td>
<td>Feet</td>
<td></td>
<td>The length of the SPAR</td>
</tr>
<tr>
<td></td>
<td>SPAR(5,1)+</td>
<td></td>
<td>Feet</td>
<td></td>
<td>The rod rejection equation number used with SPAR(1)</td>
</tr>
<tr>
<td></td>
<td>SPAR(6,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>X coordinate of SPAR origin with respect to target c.g.</td>
</tr>
<tr>
<td></td>
<td>SPAR(7,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Y coordinate of SPAR origin with respect to target c.g.</td>
</tr>
<tr>
<td></td>
<td>SPAR(8,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Z coordinate of SPAR origin with respect to target c.g.</td>
</tr>
<tr>
<td></td>
<td>SPAR(9,1)</td>
<td></td>
<td></td>
<td></td>
<td>X direction cosine of the normal to the target surface at the SPAR</td>
</tr>
<tr>
<td></td>
<td>SPAR(10,1)</td>
<td></td>
<td></td>
<td></td>
<td>Y direction cosine of the normal to the target surface at the SPAR</td>
</tr>
<tr>
<td></td>
<td>SPAR(11,1)</td>
<td></td>
<td></td>
<td></td>
<td>Z direction cosine of the normal to the target surface at the SPAR</td>
</tr>
<tr>
<td></td>
<td>SPAR(12,1)</td>
<td></td>
<td></td>
<td></td>
<td>SPAR contribution toward a kill, if hit by rod diameter SPARDM(1)</td>
</tr>
<tr>
<td></td>
<td>SPAR(13,1)</td>
<td></td>
<td></td>
<td></td>
<td>SPAR contribution toward a kill, if hit by rod diameter SPARDM(2)</td>
</tr>
<tr>
<td>Location</td>
<td>Variable Name</td>
<td>Limits</td>
<td>Units</td>
<td>Error Code</td>
<td>Definition</td>
</tr>
<tr>
<td>----------</td>
<td>---------------</td>
<td>--------</td>
<td>-------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td></td>
<td>SPAR (14,1)</td>
<td></td>
<td></td>
<td></td>
<td>Spar contribution toward a kill, if hit by rod diameter SPARDM(3)</td>
</tr>
<tr>
<td></td>
<td>SPAR (15,1)</td>
<td></td>
<td></td>
<td></td>
<td>Spar contribution toward a kill, if hit by rod diameter SPARDM(4)</td>
</tr>
<tr>
<td></td>
<td>SPAR (16,1)</td>
<td></td>
<td></td>
<td></td>
<td>Spar contribution toward a kill, if hit by rod diameter SPARDM(5)</td>
</tr>
<tr>
<td>096</td>
<td>NUMBER</td>
<td>0-20</td>
<td></td>
<td>76</td>
<td>Number of shields</td>
</tr>
<tr>
<td>3401</td>
<td>SHD(6,20)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>X coordinate of the center of the Ith shielding ellipsoid</td>
</tr>
<tr>
<td></td>
<td>SHD(1,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Y coordinate of the center of the Ith shielding ellipsoid</td>
</tr>
<tr>
<td></td>
<td>SHD(2,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Z coordinate of the center of the Ith shielding ellipsoid</td>
</tr>
<tr>
<td></td>
<td>SHD(3,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Length of semi-axis of the Ith shielding ellipsoid in X direction</td>
</tr>
<tr>
<td></td>
<td>SHD(4,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Length of semi-axis of the Ith shielding ellipsoid in Y direction</td>
</tr>
<tr>
<td>3401</td>
<td>SHD(5,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Length of semi-axis of the Ith shielding ellipsoid in Z direction</td>
</tr>
<tr>
<td></td>
<td>SHD(6,1)</td>
<td></td>
<td>Feet</td>
<td></td>
<td>Probability of kill value required for a target kill to occur</td>
</tr>
<tr>
<td>1717</td>
<td>CRASH</td>
<td>&gt;0</td>
<td>Feet</td>
<td></td>
<td>Average spar radius</td>
</tr>
<tr>
<td>1718</td>
<td>RADTG</td>
<td></td>
<td>Feet</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*If SPAR(4,I) is input as a negative value the target is considered symmetric and a mirror image of this SPAR will be created on the other side of the Y, Z plane. |
*Indicator which selects the proper reject equation (J) to use in evaluating damage against SPAR(I). |

**Note:** 
(1) REJECT(I,J), where J = 1 to 20, defines a reject velocity equation used in the SPAR kill probability analysis. These equations are sequentially numbered automatically.  

\[
V_{REJECT(J)} = \frac{REJECT(2,J)}{REJECT(1,J)} - \cos STA
\]

Where STA is the striking angle of the rod.
<table>
<thead>
<tr>
<th>Location</th>
<th>Variable Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>979</td>
<td>NBLTS</td>
<td></td>
<td></td>
<td></td>
<td>Number of bomblets</td>
</tr>
<tr>
<td>980</td>
<td>IDIST</td>
<td>0,1</td>
<td></td>
<td></td>
<td>Distribution of bomblets in pattern. 0 = normal, 1 = uniform</td>
</tr>
<tr>
<td>981</td>
<td>IBPIT</td>
<td>0,1</td>
<td></td>
<td></td>
<td>Distribution about mean pitch angle of bomblets. 0 = normal, 1 = uniform</td>
</tr>
<tr>
<td>982</td>
<td>IBYAW</td>
<td>0,1</td>
<td></td>
<td></td>
<td>Distribution about mean yaw angle of bomblets. 0 = normal, 1 = uniform</td>
</tr>
<tr>
<td>988</td>
<td>IBDT</td>
<td>0,1</td>
<td></td>
<td></td>
<td>Distribution about mean bomblet delay time. 0 = normal, 1 = uniform</td>
</tr>
<tr>
<td>1654</td>
<td>DIVE</td>
<td>Degree</td>
<td></td>
<td></td>
<td>Elevation angle of target relative to the ground</td>
</tr>
<tr>
<td>1655</td>
<td>BLTVEL</td>
<td></td>
<td>Ft/sec</td>
<td></td>
<td>Ejection velocity of bomblets</td>
</tr>
<tr>
<td>1656</td>
<td>BLTDIA</td>
<td>0</td>
<td>Inches</td>
<td></td>
<td>Bomblet diameter</td>
</tr>
<tr>
<td>1657</td>
<td>BLTWT</td>
<td>0</td>
<td>Lbs</td>
<td></td>
<td>Bomblet weight</td>
</tr>
<tr>
<td>1658</td>
<td>BMNPIT*</td>
<td>Degree</td>
<td></td>
<td></td>
<td>Mean pitch angle of bomblets</td>
</tr>
<tr>
<td>1659</td>
<td>BSGPIT</td>
<td>Degree</td>
<td></td>
<td></td>
<td>Standard deviation of mean pitch angle of bomblets.</td>
</tr>
<tr>
<td>1660</td>
<td>BNBYAW*</td>
<td>Degree</td>
<td></td>
<td></td>
<td>Mean yaw angle of bomblets.</td>
</tr>
<tr>
<td>1661</td>
<td>BSQYAW</td>
<td>Degree</td>
<td></td>
<td></td>
<td>Standard deviation of mean yaw angle of bomblets.</td>
</tr>
<tr>
<td>1662</td>
<td>BLTDT</td>
<td>0</td>
<td>Seconds</td>
<td></td>
<td>Mean bomblet delay time.</td>
</tr>
<tr>
<td>1663</td>
<td>BSGDT</td>
<td>0</td>
<td>Seconds</td>
<td></td>
<td>Standard deviation of mean bomblet delay time.</td>
</tr>
<tr>
<td>1664</td>
<td>DTU</td>
<td>&gt;0</td>
<td>Seconds</td>
<td>78</td>
<td>Time increment used in bomblet trajectory iterations for all Mach numbers greater than MACHX.</td>
</tr>
<tr>
<td>1665</td>
<td>DTL</td>
<td>&gt;0</td>
<td>Seconds</td>
<td>78</td>
<td>Time increment used in bomblet trajectory iterations for all Mach numbers less than MACHX.</td>
</tr>
<tr>
<td>1666</td>
<td>MACHX</td>
<td></td>
<td></td>
<td></td>
<td>Mach number at which the iterative time increment is changed from DTU to DTL.</td>
</tr>
<tr>
<td>987</td>
<td>NBMACH</td>
<td>2-25</td>
<td></td>
<td>77</td>
<td>Number of points in bomblet drag curve.</td>
</tr>
<tr>
<td>1667</td>
<td>BMACH (25)</td>
<td></td>
<td></td>
<td>77</td>
<td>Mach numbers for bomblet drag curve.</td>
</tr>
<tr>
<td>1692</td>
<td>BLTCD (25)</td>
<td>&gt;0</td>
<td></td>
<td>77</td>
<td>C_D values of bomblet drag curve.</td>
</tr>
</tbody>
</table>

♦ Measured relative to interceptor velocity vector.
<table>
<thead>
<tr>
<th>Location</th>
<th>Variable Name</th>
<th>Limits</th>
<th>Units</th>
<th>Error Code</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>963</td>
<td>NRANDM</td>
<td>&gt;0</td>
<td></td>
<td>32</td>
<td>An arbitrary five digit number used to initialize the random number generator.</td>
</tr>
<tr>
<td>1719</td>
<td>RHOSTL</td>
<td>&gt;0</td>
<td>Lb/in³</td>
<td>1</td>
<td>Density of fragment material. Needed only when optimizing or parameterizing fragment mass.</td>
</tr>
<tr>
<td>1642</td>
<td>$SSIZE1$</td>
<td>&gt;0</td>
<td>Ft/sec</td>
<td>62</td>
<td>Step size in strike velocity for strike velocity distribution table printout.</td>
</tr>
<tr>
<td>1643</td>
<td>$SSIZE2$</td>
<td>&gt;0</td>
<td></td>
<td>63</td>
<td>Step size in fragment density at vulnerable point for fragment density distribution table printout.</td>
</tr>
<tr>
<td>1644</td>
<td>$SSIZE3$</td>
<td>&gt;0</td>
<td>Sq ft</td>
<td>64</td>
<td>Step size in vulnerable area for vulnerable area distribution table printout.</td>
</tr>
<tr>
<td>22222</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Last data card</td>
</tr>
</tbody>
</table>

Multiple runs: Runs may be stacked by placing a *RUN card (*in column 1, RUN in columns 7-9) after last data card and then repeating the cards A-H. Cards D-H may be omitted if card A (GTITLE) contains the following message in columns 1-16 "USE PREVIOUS W/H". Columns 17-72 may contain other information which will be printed with output. Only the data in locations 1-5200 which will need change need be input ending with a 22222 card. Missile, Box, or BRL type vulnerability data may not be changed in a stacked run.
<table>
<thead>
<tr>
<th>Error Code</th>
<th>Condition Causing Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NPAAC &gt; 18 OR NPAAC &lt; 2.</td>
</tr>
<tr>
<td>2</td>
<td>NPEAC &gt; 18 OR NPEAC &lt; 2.</td>
</tr>
<tr>
<td>3</td>
<td>NHP(I,J) &gt; 10 or NHP(I,J) &lt; 0 for some I and J, I = 1,2,...,6 and J = 1,2,...,6</td>
</tr>
<tr>
<td>4</td>
<td>NH(K,I,J) &lt; 20 and NHP(I,J) = KSTOP &gt; 0 for some I, J, and K where I = 1,2,...,6; J = 1,2,...,6; K = KSTOP</td>
</tr>
<tr>
<td>5</td>
<td>NH(K,I,J) = NH(KCK,1,J) for some K and some KCK where K = 1,2,...,KSTOP-1 and KCK = K+1, K+2,...,KSTOP.</td>
</tr>
<tr>
<td>6</td>
<td>NF(I,J) &gt; 20 or NF(I,J) &gt; 0 for some I, J where I = 1,2,...,6 and J = 1,2,...,6.</td>
</tr>
<tr>
<td>7</td>
<td>NHP(I,J) &gt; 0 OR NHP(I,J) &gt; 20 and NHP(I,J) = KSTOP &gt; 0 for some I, J, K where I = 1,2,...,6; J = 1,2,...,6; K = 1,2,...,KSTOP.</td>
</tr>
<tr>
<td>8</td>
<td>NLOBES &lt; 0 or NLOBES &gt; 7.</td>
</tr>
<tr>
<td>9</td>
<td>NVR &lt; 2 OR NVR &gt; 10.</td>
</tr>
<tr>
<td>10</td>
<td>NEDH &lt; 0 OR NEDH &gt; 10.</td>
</tr>
<tr>
<td>11</td>
<td>NVR &gt; 0 and NVR &gt; 1.</td>
</tr>
<tr>
<td>12</td>
<td>RHSTL &lt; 0 if optimizing or parameterizing fragment mass.</td>
</tr>
<tr>
<td>13</td>
<td>NVRBER &lt; 2 OR NVRBER &gt; 10 AND INDDT &gt; 0.</td>
</tr>
<tr>
<td>14</td>
<td>NCB &lt; 0 OR NCB &gt; 10.</td>
</tr>
<tr>
<td>15</td>
<td>NPBT &lt; 2 OR NPBT &gt; 25 and NCB &gt; 0.</td>
</tr>
<tr>
<td>16</td>
<td>NPA(I)*NPD(I)*NDV(I)*NFH(I) &gt; 1000.</td>
</tr>
<tr>
<td>17</td>
<td>POLAT(J-1,I) &gt; POLAT(J,I) for some J = 2,3,...,NPV and I = 1,2,...,NPV when NPV &gt; 0 and IAV &gt; 0.</td>
</tr>
<tr>
<td>18</td>
<td>POLBT(J-1,I) &gt; POLBT(J,I) for some J = 2,3,...,NPB(I) and I = 1,2,...,NPB(I) when NPV &gt; 0 and IAV &gt; 0.</td>
</tr>
<tr>
<td>19</td>
<td>DVTAB(J-1,I) &gt; DVTAB(J,I) for some J = 2,3,...,NDV(I) and I = 1,2,...,NDV(I) when NPV &gt; 0 and IAV &gt; 0.</td>
</tr>
<tr>
<td>20</td>
<td>FMAS(J,I) &lt; 0 for some J = 1,2,...,NPV and I = 1,2,...,NPV when NPV &gt; 0 and IAV &gt; 0.</td>
</tr>
<tr>
<td>21</td>
<td>AVTABS(J,I) &lt; 0 for some J = 1,2,...,NPV and I = 1,2,...,NPV when NPV &gt; 0 and IAV &gt; 0.</td>
</tr>
<tr>
<td>22</td>
<td>NPVELT &lt; 2 or NPVELT &gt; 10 and NPV &gt; 0 and IAV &lt; 0.</td>
</tr>
<tr>
<td>23</td>
<td>NPANG &lt; 2 or NPANG &gt; 19 and IAV &lt; 0.</td>
</tr>
<tr>
<td>24</td>
<td>AVTMV(J,K) &lt; 0 for some J = 1,2,...,NPANG and some K = 1,2,...,NPANG and IAV &lt; 0.</td>
</tr>
<tr>
<td>25</td>
<td>VSBART(J-1,I) &gt; VSBART(J,I) for some J = 2,3,...,NPV and IAV &lt; 0.</td>
</tr>
<tr>
<td>26</td>
<td>AVANGT(J-1,I) &gt; AVANGT(J,I) for some J = 2,3,...,NPANG and IAV &lt; 0.</td>
</tr>
<tr>
<td>27</td>
<td>AVRT(J,1) &lt; 0 OR AVRT(J,2) &lt; 0 OR AVRT(J,3) &lt; 0 OR AVRT(J,4) &lt; 0 OR AVRT(J,5) &lt; 0 OR AVRT(J,6) &lt; 0 for some J = 1,2,...,NPV and IAV = 0.</td>
</tr>
<tr>
<td>28</td>
<td>NPV &gt; 0 AND NGROUP &lt; 1 or NGROUP &gt; 10.</td>
</tr>
<tr>
<td>29</td>
<td>NCOM(I) = 1 for some I = 1,2,...,NGROUP.</td>
</tr>
</tbody>
</table>
TABLE I-X---CONTINUED

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Condition Causing Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>For some I = 1,2,..., NPV</td>
</tr>
<tr>
<td></td>
<td>I ≠ NPVCOM(J,K) for all</td>
</tr>
<tr>
<td></td>
<td>K = 1,2,...,NGROUP and J=1,2,...,NCOM(K)</td>
</tr>
<tr>
<td></td>
<td>In other words, there is some vulnerable point which is not in any vulnerability group.</td>
</tr>
<tr>
<td>31</td>
<td>KILL(K) &lt; 1 OR KILL(K) &gt; NCOM(K) for some K = 1,2,...,NGROUP and NPV &gt; 0.</td>
</tr>
<tr>
<td>32</td>
<td>NRANDM ≤ 0.</td>
</tr>
<tr>
<td>33</td>
<td>AZ(J-1) &gt; AZ(J) for some J = 2,3,...,NPAC</td>
</tr>
<tr>
<td>34</td>
<td>FAZ(J) &lt; 0 OR FAZ(J) &gt; 1 for some j = 1,2,...,NPEAC</td>
</tr>
<tr>
<td>35</td>
<td>ELEV(J-1) &gt; ELEV(J) for some J = 2,3,...,NPEAC</td>
</tr>
<tr>
<td>36</td>
<td>PELEV(J) &lt; 0 OR PELEV(J) &gt; 1 for some J = 1,2,...,NPEAC.</td>
</tr>
<tr>
<td>37</td>
<td>VTBAR &lt; 0.</td>
</tr>
<tr>
<td>38</td>
<td>VMBAR &lt; 0.</td>
</tr>
<tr>
<td>39</td>
<td>PVOFF &lt; 0.</td>
</tr>
<tr>
<td>40</td>
<td>FGUP(I) &lt; 0 for some I = 1,2,...,20</td>
</tr>
<tr>
<td>41</td>
<td>TDX &lt; 0.</td>
</tr>
<tr>
<td>42</td>
<td>Q &lt; 0.</td>
</tr>
<tr>
<td>43</td>
<td>DD &lt; 0.</td>
</tr>
<tr>
<td>44</td>
<td>XPP &lt; 0.</td>
</tr>
<tr>
<td>45</td>
<td>LL &lt; 0.</td>
</tr>
<tr>
<td>46</td>
<td>NPV &gt; 0 when KRODS &gt; 0.</td>
</tr>
<tr>
<td>47</td>
<td>CD &lt; 0.</td>
</tr>
<tr>
<td>48</td>
<td>RHO &lt; 0.</td>
</tr>
<tr>
<td>49</td>
<td>VMBART(I-1) &gt; VMBART(I) for some I = 2,3,...,NVR.</td>
</tr>
<tr>
<td>50</td>
<td>OMTB(I) &lt; 0 OR OMTB(I) &gt; 180 for some I = 1,2,...,NVR.</td>
</tr>
<tr>
<td>51</td>
<td>PCB(4,I) &lt; 0 for some I = 1,2,...,NEDB</td>
</tr>
<tr>
<td></td>
<td>OR</td>
</tr>
<tr>
<td></td>
<td>NEDB</td>
</tr>
<tr>
<td></td>
<td>I = 1</td>
</tr>
<tr>
<td></td>
<td>PCB(4,I) &gt; 15</td>
</tr>
<tr>
<td></td>
<td>EDB(J,I) &lt; 0 for some J = 4,5,6 and some I = 1,2,...,NEDB</td>
</tr>
<tr>
<td></td>
<td>Σ PCB(4,K)</td>
</tr>
<tr>
<td></td>
<td>K = 1</td>
</tr>
<tr>
<td>53</td>
<td>EDH(J,I) &lt; 0 for some J = 4,5,6 and some I = 1,2,...,NEDM.</td>
</tr>
<tr>
<td>54</td>
<td>VF(I) &lt; 0.</td>
</tr>
<tr>
<td>55</td>
<td>VRBART(I-1) &gt; VRBART(I) for some I = 2,3,...,NVRBER and INDDT &gt; 0.</td>
</tr>
<tr>
<td>56</td>
<td>TAB12X(I-1) &gt; TAB12X(I) for some I = 2,3,...,NPBT and NCB &gt; 0.</td>
</tr>
<tr>
<td>57</td>
<td>SHEILD(I,J) ≤ 0 for some I = 4,5,6 and some J = 1,2,...,NPAC and NSH &lt; 0.</td>
</tr>
<tr>
<td>58</td>
<td>NPOLAR &lt; 0 or NPOLAR &gt; 36.</td>
</tr>
<tr>
<td>59</td>
<td>THSL(I) &lt; 0 OR THSL(I) &gt; THSU(I) for some I = 2,3,...,NPOLAR OR THSU(I) &gt; 180 for some I = 2,3,...,NPOLAR.</td>
</tr>
<tr>
<td>60</td>
<td>TTOL &lt; 10^-10 and NPV &gt; 0.</td>
</tr>
<tr>
<td>61</td>
<td>YTOL &lt; 0.1 and NPV &gt; 0.</td>
</tr>
<tr>
<td>62</td>
<td>SSIZE1 ≤ 0 and NPV &gt; 0.</td>
</tr>
<tr>
<td>63</td>
<td>SSIZE2 ≤ 0 and NPV &gt; 0.</td>
</tr>
<tr>
<td>64</td>
<td>SSIZE3 ≤ 0 and NPV &gt; 0.</td>
</tr>
<tr>
<td>65</td>
<td>DTINC ≤ 0 and INDDT &lt; 0.</td>
</tr>
<tr>
<td>66</td>
<td>HSA &lt; 0 or HSA &gt; 180 and INDDT &lt; 0.</td>
</tr>
<tr>
<td>67</td>
<td>MXTODT &lt; 0.</td>
</tr>
<tr>
<td>68</td>
<td>SDE &lt; 0.</td>
</tr>
<tr>
<td>69</td>
<td>NMIN &gt; NIT</td>
</tr>
<tr>
<td>70</td>
<td>THET1(I) &gt; THET2(I) for some I = 1,2,...,NOBSTR or NOBSTR &gt; 10.</td>
</tr>
<tr>
<td>71</td>
<td>NGRID &gt; 20.</td>
</tr>
</tbody>
</table>
| 72         | NFM(I) ≤ 0 or NFM(I) > 5 for some I = 1,2,...,NPF.
<table>
<thead>
<tr>
<th>Error Code</th>
<th>Condition Causing Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>73</td>
<td>NSPRDM &lt; 0 or NSPRDM &gt; 5 when KRODS &gt; 0.</td>
</tr>
<tr>
<td>74</td>
<td>ISPAR &lt; 0 or ISPAR &gt; 50 when KRODS &gt; 0.</td>
</tr>
<tr>
<td>75</td>
<td>LREJCT &lt; 0 or LREJCT &gt; 20 or NUMBER &gt; 20 when KRODS &gt; 0.</td>
</tr>
<tr>
<td>76</td>
<td>NBMACH &lt; 2 or NBMACH &gt; 25 when NBLTS &gt; 0 or BMACH(N-1) &gt; BMACH(N) for some N = 2, NBMACH.</td>
</tr>
<tr>
<td>77</td>
<td>DTU &lt; 0 or DTL &lt; 0 when NBLTS &gt; 0.</td>
</tr>
</tbody>
</table>
APPENDIX II
DERIVATION OF BASIC METHODOLOGY

1. BASIC GEOMETRY OF EACH ENCOUNTER

Figure II-1 illustrates the geometry of a typical encounter.

The reference coordinate system (unprimed X, Y, Z coordinate system, Figure II-1) is a left-handed cartesian system with origin at a point on the target to which the positions of spatial characteristics of the target can be conveniently referred. The Y axis is directed along the target velocity vector. The Z axis is up and the X axis is out the left side of the target.

The approach angles are the azimuth (CPSI) and elevation (GAMMA) angles of the interceptor (negative) velocity vector in the X, Y, Z system. Computation of approach angles for each engagement is by random selection from an input distribution on the approach angles (see inputs PAZ and PELEV).

To define a continuous distribution for either approach angle, the distribution is first approximated by subdividing the range of the distribution into as many as 18 parts such that the distribution may be considered to be uniform over each subrange. The angles defining the boundaries of the subranges are input in ascending order, starting with the azimuth angle nearest zero and the elevation angle nearest -90 degrees. Associated with each subrange is a probability that an approach angle will occur in that range. The sum of these probabilities over the total range must be unity. These probabilities are entered in the same order as the corresponding subranges. The first azimuth subrange extends from zero degrees to the first boundary angle. The first elevation subrange extends from -90 degrees to the first boundary angle. The first elevation subrange extends from -90 degrees to the first boundary angle.

To define deterministic (fixed) approach angles, the distributions for these angles must be defined in such a way that the second subrange has zero width and unity probability. This zero width subrange must be located at the desired angle. To do this, set the first two subrange boundary angles equal to the desired angle. Set the first subrange probability to zero and the second one to unity. Only two subranges are used.

The double primed coordinate system (X'', Y'', Z'') is used later for calculations of homing radar and guidance errors. Such target position errors are relative to the interceptor, so it is natural to align one axis with the relative velocity vector. These errors are also influenced by the attitude of the target as seen by the interceptor. The X'' and Z'' axes are oriented to facilitate treatment of these attitude effects. The mathematical relationship between these coordinate systems is derived in paragraph c below.

After forming the matrix for the coordinate transformation just described, this section computes the following:

1. Direction cosines (ETA(1), ETA(2), ETA(3)) of a negative unit interceptor velocity vector, with respect to the reference coordinate system;
2. Interceptor angles of attack in azimuth (AAA) and elevation (AAE). Positive values of AAA and AAE correspond to yaw left and pitch down, respectively; specific values of AAA and AAE are selected at random from Gaussian distributions having standard deviations of AOAA and EOAA, respectively, as specified by the input data;

60
Direction cosines \((\mathbf{E}(1), \mathbf{E}(2), \mathbf{E}(3))\) of a unit vector along the negative interceptor center line; this vector is coincident with the negative unit interceptor velocity vector when the azimuth and elevation angles of attack are zero.

The origin of the interceptor coordinate system is at the center of the homing radar. To visualize the orientation of this system, assume that the interceptor is an aircraft. The interceptor X axis is directed toward the right wing, the Y axis is directed toward the tail, and the Z axis is directed toward the top. When all azimuth and elevation angles are zero (both approach angles and attack angles), then the orientation of the interceptor coordinate system coincides with the reference system. In general, when the angles of attack are zero, the X axis (wing axis) is horizontal, that is, parallel to the reference XY plane. The interceptor nose is pointed in the direction of travel. When an azimuth angle of attack is introduced, the interceptor is rotated about its own Z axis. Thus, its X axis is no longer necessarily horizontal. When an elevation angle of attack is added, the interceptor is rotated about its own X axis.

The following paragraphs explain or derive some of the mathematical relationships used.

a. Expression for \(\psi\)

Let \(\hat{u}\) be a vector along \(\hat{i}\). Let \(\hat{u}'\) be \(\hat{u}\) rotated through angle \(\psi\). Let \(\hat{u}''\) be \(\hat{u}'\) rotated through angle \(\gamma\). Then

\[
\hat{u}' = u \cos \psi \hat{i} + u \sin \psi \hat{j} \\
\hat{u}'' = \cos \gamma u \sin \psi \hat{i} + \cos \gamma u \cos \psi \hat{j} + u \sin \gamma \hat{k} \\
\hat{u}'' \cdot \hat{i} = |\hat{u}''|^2 [\hat{i} \cos \psi = u'' \cos \psi] \\
\hat{u}'' \cdot \hat{j} = u \cos \gamma \cos \psi \\
\hat{u}'' \cos \psi = u \cos \gamma \cos \psi.
\]

Since \(u'' = u\), \(\psi = \arccos (\cos \gamma \cos \psi)\).

b. Expression for \(\lambda\)

Let \(x\) be the angle between -\(\overrightarrow{VRBA}\) and \(\hat{i}\). Then, by analogy to paragraph a above:

\[
\cos x = \cos \psi \cos \lambda
\]
so \( \Psi = \arccos \left( \frac{\cos x}{\cos \Lambda} \right) \)

Also \( \sin x = \frac{\sin (t-\Xi)}{V_R} \)

so \( x = \arcsin \left( \frac{V_M \sin \Xi}{V_R} \right) \)

Therefore \( \Psi = \arccos \left( \cos \left( \arcsin \left( \frac{V_M \sin \Xi}{V_R} \right) \right) \right) / \cos \Lambda \).

c. Derivative of Rotation Matrices

We first rotate through the angle \( \Psi \) about the Z axis:

\[
\begin{align*}
&x' = x \cos \Psi + y' \sin \Psi \\
y' = -x' \sin \Psi + y' \cos \Psi \\
z' = z'.
\end{align*}
\]

Thus the matrix to go from primed to unprimed system is:

\[
P_1 = \begin{pmatrix}
\cos \Psi & \sin \Psi & 0 \\
-sin \Psi & \cos \Psi & 0 \\
0 & 0 & 1
\end{pmatrix}.
\]

Since \( P_1 \) is orthonormal, \( P_1^{-1} = P_1 \) (transpose of \( P_1 \)).

We next rotate through \( \Lambda \) about the \( x' \) axis:
By inspection
\[ x' = x'' \]
\[ y' = y'' \cos \lambda - z'' \sin \lambda \]
\[ z' = z'' \sin \lambda + x'' \cos \lambda. \]

The matrix to go from the double primed to the unprimed system is therefore:
\[
P_2 = \begin{pmatrix}
1 & 0 & 0 \\
0 & \cos \lambda & -\sin \lambda \\
0 & \sin \lambda & \cos \lambda
\end{pmatrix}
\]
\[
P_2 = \begin{pmatrix}
\cos \psi & \sin \psi & \cos \lambda & -\sin \psi & \sin \lambda \\
-\sin \psi & \cos \psi & \cos \lambda & -\sin \psi & \sin \lambda \\
0 & \sin \lambda & \cos \lambda & \sin \psi & \cos \psi
\end{pmatrix}
\]

The program variables PRIME(I, J), I = 1, 2, 3; J = 1, 2, 3 are related to \( P_2 \) by
\[
P_2 = (\text{PRIME}(I, J))'.
\]

d. Expression for Vector \( \text{ETE}(I), I = 1, 2, 3 \)

The vector \( \text{ETA}(I), I = 1, 2, 3 \) (components in unprimed coordinate system) is a unit vector in the direction opposite to the interceptor missile's velocity vector \( \text{VMVAR} \). \( \text{ETE}(I), I = 1, 2, 3 \) is a unit vector opposite to the missile's body axis. To obtain \( \text{ETE} \) from \( \text{ETA} \) we rotate \( \text{ETA} \) through an azimuth angle of attack (AAA) and then through an elevation angle of attack (AAE) as shown in Figure II-1.

\( \text{ETE}(I), I = 1, 2, 3 \) are to be components in the unprimed coordinate system. We first obtain the components of \( \text{ETE} \) in the \( x_1, y_1, z_1 \) coordinate system. This is accomplished by using a matrix analogous to \( P_2 \) in paragraph above.

\[
\text{ETE} \quad \text{(in} \ x_1, y_1, z_1 \text{system)} = \begin{pmatrix}
\cos \text{AAA} & \sin \text{AAA} \cos \text{AAE} & -\sin \text{AAA} \sin \text{AAE} \\
-\sin \text{AAA} & \cos \text{AAA} \cos \text{AAE} & -\cos \text{AAA} \sin \text{AAE} \\
0 & \sin \text{AAE} & \cos \text{AAE}
\end{pmatrix}
\]

The \( x_1, y_1, z_1 \) coordinate system is related to the \( x, y, z \) coordinate system by a matrix analogous to \( P_2 \).

So

\[
\text{ETE} \quad \text{(in} \ x, y, z \text{system)} = \begin{pmatrix}
\cos \text{CPSI} & \sin \text{CPSI} \cos \gamma & -\sin \text{CPSI} \sin \gamma \\
-\sin \text{CPSI} & \cos \text{CPSI} \cos \gamma & -\cos \text{CPSI} \sin \gamma \\
0 & \sin \gamma & \cos \gamma
\end{pmatrix},
\]

or in the \( (x, y, z) \) system

\[
\text{ETE} \ (1) = \cos \text{CPSI} \sin \gamma \cos \gamma + \sin \text{CPSI} \cos \gamma \cos \gamma \cos \gamma + \sin \text{CPSI} \sin \gamma \cos \gamma \\
-\sin \text{CPSI} \sin \gamma \cos \gamma + \cos \text{CPSI} \cos \gamma \cos \gamma \cos \gamma - \cos \text{CPSI} \sin \gamma \cos \gamma \\
\sin \gamma \cos \gamma \cos \gamma + \cos \gamma \cos \gamma \cos \gamma + \sin \gamma \cos \gamma \cos \gamma.
\]
2. RADAR TARGET CENTROID

There are three ways to compute the radar centroid of the target (see input variable INDTC).

a. INDTC > 0

To simulate what the homing radar sees, the target is represented by a set of up to 20 reflection points (homing glitter points). The coordinates of these points are specified by input data. The points are automatically assigned serial numbers from 1 to 20. A second input data table is used to indicate the serial numbers of up to 10 of these points which are actually visible to the radar for the PSI and LAMBDA angles determined in Section 3. The first set of serial numbers applies for PSI values from 0 to 30 degrees and LAMBDA values from -90 to -60 degrees. The entire range of PSI and LAMBDA values is divided into similar 30 degree sub-ranges, requiring a total of 36 sets of serial numbers to complete the table.

To compute the radar target centroid (PCP(I), I = 1, 2, 3) in the double primed coordinate system, the coordinates of the homing glitter points are transformed from the reference coordinate system into the double primed system and averaged.

b. INDTC = 0.

The X", Z" coordinates of the radar target centroid are PCPX and PCPZ (input numbers). The Y" coordinate is zero. This method of specifying the radar centroid is not the same as when INDTC < 0 discussed below.

c. INDTC < 0.

The target radar centroid is placed at x, y, z coordinates AIM(I), AIM(2), AIM(3) in the unprimed coordinate system.

3. GUIDANCE ERROR

The aim point position vector is defined to be the sum of the radar centroid position vector (just discussed in (4) above) and the guidance error vector. There are two ways to compute the guidance error vector (depending on input variable INDMD).

a. INDMD = 0

The guidance error in the X" direction is a sample from a Gaussian distribution with sigma SMISSX (input). The guidance error in the Z" direction is a sample from a gaussian distribution with sigma SMISSZ (input). The Y" component is zero.

b. INDMD ≠ 0

The guidance error is taken at random on a circle of radius RMISS (input). The circle lies in a plane parallel to the X", Z" plane, so the Y" component of guidance error is zero.

After determining the coordinates of the aim point in the double-primed coordinate system (PTP(I), I = 1, 2, 3) the corresponding coordinates in the reference coordinate system (PT(I), I = 1, 2, 3) are determined using the rotation matrix.

4. FUZING

The fuze simulated by the program has the following characteristics:

1. A radar fuse cone, symmetric about the body axis of the missile, (with cone angle ACTOMG ≠ π/2) is capable of detecting a return from any of several fuzing glitter points (specified by input) on the target if such a point enters the radar beam within a certain minimum range RCOFF (input);

2. Mathematically, different glitter points can enter the radar beam at different times; however, fuzing is said to occur at the time when the first (in time) return from a glitter point is detected;
The program is capable of simulating the effect of a fuze jammer carried by the target; such a fuze jammer can cause fuzing to occur at a time which is different from the time fuzing would have occurred in the absence of a jammer; the jammer can cause fuzing to occur on fuzing radar beam side lobes or to occur prematurely (at long range) on the main lobe.

Figure II-2 illustrates the geometry of fuzing on a fuzing glitter point. Figure II-3 illustrates and describes fuzing caused by a jammer.

a. Subroutine FUZE

An explanation of the equations in Subroutine FUZE is in order (Figure II-2).

\[ \mathbf{A} = A_1 \mathbf{i} + A_2 \mathbf{j} + A_3 \mathbf{k} \]

where \( \mathbf{i}, \mathbf{j}, \mathbf{k} \) refer to \((w'', y'', z'')\) system.

\( \mathbf{ETE} \) is a unit vector in the negative body axis direction. \( \mathbf{ETE}(1), \mathbf{ETE}(2), \mathbf{ETE}(3) \) are the components of \( \mathbf{ETE} \) in the unprimed coordinate system. For convenience, let \( V_1, V_2, V_3 \) be the component of \( \mathbf{ETE} \) in the double primed system. Let \( V_\mathbf{C} \) be the vector from the apex of the fuze cone to the point on the cone where the cone intersects the glitter point. By inspection of Figure II-2:

\[ V_\mathbf{C} = A_1 \mathbf{i} + (A_2 + \Delta y) \mathbf{j} + A_3 \mathbf{k}. \]

Now

\[ \mathbf{V}_\mathbf{C} \cdot \mathbf{ETE} = -V_\mathbf{C} \cos \alpha_{\mathbf{ACTOMG}}. \]

Substituting the expressions for \( V_\mathbf{C} \) and \( \mathbf{ETE} \) and squaring both sides gives

\[ (V_1 A_1 + V_2 (A_2 + \Delta y) + V_3 A_3)^2 = (A_1^2 + (A_2 + \Delta y)^2 + A_3^2) \cos^2 \alpha_{\mathbf{ACTOMG}}. \]

After expanding, regrouping terms, and dividing through by \( \cos^2 \alpha_{\mathbf{ACTOMG}} \) we obtain:

\[ 2 \cos^2 \alpha_{\mathbf{ACTOMG}} \Delta y^2 + 2 \left( \frac{V_1}{\cos^2 \alpha_{\mathbf{ACTOMG}}} \frac{V_2}{\cos^2 \alpha_{\mathbf{ACTOMG}}} \right) \Delta y + \left( \frac{(V_1 A_1 + V_2 A_2 + V_3 A_3)^2}{\cos^2 \alpha_{\mathbf{ACTOMG}}} - |A|^2 \right) = 0. \]

For convenience we let

\[ B_1 = A_2, \] \[ B_2 = V_2, \] \[ B_3 = |A|^2, \] \[ B_4 = V_1 A_1 + V_2 A_2 + V_3 A_3 = \mathbf{A} \cdot \mathbf{ETE} = A_1 \mathbf{ETE}(1) + A_2 \mathbf{ETE}(2) + A_3 \mathbf{ETE}(3). \]

Substituting we have:

\[ \left( \frac{B_2}{\cos^2 \alpha_{\mathbf{ACTOMG}}} \right)^2 \Delta y^2 + 2 \left( \frac{B_2 B_4}{\cos^2 \alpha_{\mathbf{ACTOMG}}} - B_3 \right) \Delta y + \left( \frac{B_4}{\cos^2 \alpha_{\mathbf{ACTOMG}}} \right)^2 - B_3 = 0. \]

This equation will have real roots \( \text{TPLUS} \) and \( \text{TMINUS} \), if and only if the "mathematical" fuze cone intersects the fuze glitter point, i.e., the existence of \( \text{TPLUS} \) and \( \text{TMINUS} \) implies that

1. The actual fuze cone intersects the glitter point twice, or
2. The actual fuze cone intersects the glitter point once, or
3. The actual fuze cone does not intersect the glitter point at all.
The problem now is to determine whether the case is 1, 2, or 3 and which root(s) (if any) correspond to cases of actual intersection. The two roots can be analyzed independently. The following discussion applies to either root. Suppose \( \mathbf{V} \) is the vector from the apex of the cone to the point where the glitter point "mathematically" intersects the cone. We let:

\[
\mathbf{Q} = \frac{\mathbf{E}}{\cos \mathbf{ACTOMG}}\mathbf{V} = |\mathbf{V}| \cos \varphi
\]

where \( \varphi \) is the angle between \( \mathbf{V} \) and \( \mathbf{ETE} \). Obviously, \( \varphi \) is \( \mathbf{ACTOMG} \) or \( \pi - \mathbf{ACTOMG} \). If \( \varphi = \mathbf{ACTOMG} \), the "mathematical" intersection is fictitious. Actual intersection occurs if, and only if, \( \varphi = \pi - \mathbf{ACTOMG} \). Now let:

\[
\mathbf{V} C = \frac{\mathbf{Q}}{\cos \mathbf{ACTOMG}}
\]

(The symbol, \( \mathbf{V} C \) does not mean \( |\mathbf{V}| \)).

Combining the two expressions gives

\[
\mathbf{V} C = \frac{|\mathbf{V}| \cos (\pi - \mathbf{ACTOMG})}{\cos \mathbf{ACTOMG}} = -|\mathbf{V}|
\]

i.e., if, and only if \( |\mathbf{V}| < 0 \).

The exceptional case of \( \mathbf{V} C = 0 \) occurs if the apex of the cone actually hits the glitter point. In this case we assume that intersection occurred. Thus, we say that intersection actually occurred if, and only if \( \mathbf{V} C \leq 0 \).

b. Intersection

Actual intersection, as determined in paragraph A above, can occur for several glitter points. Fuzing will occur only on the first (in time) intersecting glitter point which has a \( \mathbf{V} C \) such that \( |\mathbf{V} C| < \text{RCOFF} \) (an input). Which intersection occurred first depends on only one number, the \( Y' \) component of the \( \mathbf{P} \) vector associated with each intersection. The intersection which occurred first is the one whose \( Y' \) component of \( \mathbf{P} \) is greatest.

c. Fuze Jammer Computations

If the target is carrying a fuze jammer the computations required for determining whether or not the jammer causes fuzing are mathematically the same as for normal fuze glitter points except that:

1. The jammer can cause fuzing on fuze cones other than the main fuze cone, i.e., on side lobes of the main cone. If the jammer intersects one of these cones, fuzing can occur if the range is such that the signal received by the fuze cone receiver (which depends on jammer transmitting power) is greater than the threshold sensitivity of the receiver. Mathematically, in order for the jammer to cause fuzing it must be true that:

\[
\frac{\text{FACTOR}(I)}{\mathbf{V} C + 2} > \text{RECSEN}(I)
\]

where

\[
\text{FACTOR}(I) = \frac{P \cdot G \cdot A}{B \cdot 4}, \quad \text{RECSEN}(I) = K \cdot T \cdot N F \cdot (S/N) \cdot G \cdot L
\]

where
\[ P_j = \text{jamming power - watts} \]
\[ G_j = \text{jammer antenna gain} \]
\[ A = \text{area of receiver antenna apperature (ft)}^2 \]
\[ B_j = \text{bandwidth of jammer - mc} \]
\[ K = \text{Boltzman's Constant} = 1.38 \times 10^{-23} \text{joules/K} \]
\[ T = \text{Kelvin} \]
\[ NF = \text{noise factor} \]
\[ S/N = \text{signal to noise factor of the receiver} \]
\[ G_r = \text{receiver antenna gain for either main or side lobes} \]
\[ L_r = \text{receiver losses}. \]
\[ Vc^2 = \text{the range squared from the cone apex to the point where the } i^{th} \text{ lobe intersects the jammer position.} \]

The cone angles for the cones which are susceptible to jamming are measured with respect to the main fuze cone whose cone angle is \( \text{ACTOMG} \). The input quantities \( \text{ANGJAM}(i), i = 1, 2, \ldots, \text{NLOBES} \) are the angles of the side lobes with respect to the main fuze cone, i.e., the cone angle of the \( i^{th} \) lobe measured with respect to the body axis of the missile is \( \text{ACTOMG} + \text{ANGJAM}(i) \). Normally the main fuse cone is susceptible to jamming, in which case one value of \( \text{ANGJAM} \) should be zero.

Finally, if a jammer is used, actual fuzing will occur because of the jammer or because of a fusing glitter point, which ever occurs first in time.

**5. DELAY TIME, ASSOCIATED ERRORS AND DETERMINATION OF BURST POINT**

By delay time we mean the elapsed time between the time of fuzing and time of warhead detonation. There are three optional (controlled by input variable \( \text{INDDT} \)) methods for determining delay time. In no case can delay time be negative. \( \text{PMFP} \) is the location of the fuze cone apex when fuzing occurred. \( \text{PPP} \) is the position of the fuze cone apex when burst occurs. However, the warhead is located \( W \) feet (input) behind the cone apex. Therefore, \( \text{PW} \), as computed, is the position of the warhead at time of burst.

The three options for computing delay time are as follows:

1. If \( \text{INDDT} = 0 \), the delay time is a sample from a gaussian distribution with mean \( \text{TDX} \) (input) and sigma \( \text{SPMF} \) (input).
2. If \( \text{INDDT} > 0 \), the delay time is a function of estimated relative velocity \( \text{VRBER} \).
   The nominal delay time is determined by linearly interpolating in an input table of nominal delay time versus estimated relative velocity. Suppose the interpolated value is \( \text{TD} \). Then actual delay time is a sample from a gaussian distribution with mean \( \text{TD} \) and sigma \( \text{SPMF} \) (input).
3. If \( \text{INDDT} < 0 \), an attempt is made to compute and use a delay time \( \text{TD} \) which is an optimum delay time in the following sense: the coordinates of the target radar centroid are used as the aim point. (Regardless of how computed, the centroid is at \( \text{CEN}(1), \text{CEN}(2), \text{CEN}(3) \). (See paragraph 4.) An angle \( \text{HSA} \) is also input.
   The program, specifically Subroutine \( \text{DRT} \), attempts to find a delay time \( \text{TD} \) which will be such that fragments leaving the warhead at angle \( \text{HSA} \) with initial static velocity \( \text{VF}(1) \) will hit the point \( \text{CEN}(1), \text{CEN}(2), \text{CEN}(3) \). The angle \( \text{HSA} \) is measured the same way as the values of \( \text{THETA} \) are measured in Figure II-4.

In order to profitably use option 3 one must understand Subroutine \( \text{DRT} \) (Figure II-5).

The procedure will not work well if the interceptor missile is approaching the target from the sides, top, or bottom, i.e., in order to work well, we should have:

- \( -30^\circ \leq \text{GAMMA} \leq 30^\circ \) and \( 0 \leq \text{CPSI} \leq 30^\circ \), or
- \( -30^\circ \leq \text{GAMMA} \leq 30^\circ \) and \( 150^\circ \leq \text{CPSI} \leq 180^\circ \).

70
Figure II-4. Definition of Input Variables Describing
A Warhead with Three Angular Fragment Spray Zones
and Two Fragment Velocity Groups

- Fragment Spray in Symmetric About Body Axis
- Limits of Fragment Spray
- Zone 1
- Zone 2
- Zone 3
- $\theta(1)$
- $\theta(2)$
- $\theta(3)$
- $\theta(4)$
- $\text{VF}(1)$
- $\text{VF}(2)$

- $\text{NVEL} = 2$
- $\text{NTHET} = 4$
- These are $\text{FRAG}(1,J)$ (I=1,2,3) (J=1,2)
- Fragments with Initial Static Velocity $\text{VF}(J)$ in Zone 1
Figure II-5. Geometry Used in Subroutine &DT

Position Vector of AIM Point at Time of Fuzing (CEN(1), CEN(2), CEN(3))
The method used in Subroutine 0DT is quite crude. The warhead is first moved to where it will be SDE seconds (input) after fuzing occurred. Then the warhead is moved along in small time steps of DTINC seconds (input). At each step we compute the point, or points, on the line traversed by the aim point which are hit by fragments leaving the warhead with a static angle of HSA. Knowing the time when such fragments reach the point, we compute the position, at that time, of the aim point. As the warhead is stepped along, we watch the sign of the algebraic difference between the y position of the aim point and the point y hit by fragments. If the sign at a particular step is different from the sign on the preceeding step, the desired TD occurred somewhere between the two steps. A linear interpolation is made to compute the value of TD. The maximum number of time steps is MNT0DT (input). Obviously, the input numbers SDE, DTINC, and MNT0DT are critical. The desired value of TD will not be found unless it lies between SDE and SDE + MNT0DT * DTINC. If no value of TD is found, no detonation occurs and the program proceeds to the next Monte Carlo encounter. Note also that as soon as a value of DT is found the search terminates. All other possible solutions are ignored.

We now derive the equations used to determine the point, or points, on the aim point trajectory line which would be hit be the appropriate fragments if the warhead detonated at a particular step in the search for the desired TD. See Figure II-5. P is the position vector of the warhead at this step. The two vectors V are the two directions in which fragments must travel in order to hit the line traversed by the aim point. The vectors E are unit vectors perpendicular to P - ETE. We have:

\[ \mathbf{V} = \mathbf{VMBAR} + (\mathbf{VF}(1)) \cos \text{HSA} \mathbf{(-ETE)(1)} + (\mathbf{VF}(1)) \sin \text{HSA} \mathbf{ETE} \]  
\[ \mathbf{E} \cdot \mathbf{ETE} = 0 \]  
\[ |\mathbf{E}| = 1 \]  
\[ \mathbf{P} - \mathbf{V} = \mathbf{CEN}(1) \mathbf{i} + y \mathbf{j} + \mathbf{CEN}(3) \mathbf{k} \]

Where t is a parameter and y is the y coordinate of the point where Vt intersects the trajectory line of the aim point. Only cases where t > 0 are of interest. The above equations (a total of 8 equations) can be solved for the eight unknowns V, E, y, t. For convenience, let:

\[ \begin{align*}
AA &= (\mathbf{CEN}(3) - \mathbf{P}(3)) \cdot \mathbf{VF}(1) \cdot \sin \text{HSA} \\
BB &= (\mathbf{CEN}(1) - \mathbf{P}(1)) \cdot \mathbf{VF}(1) \cdot \sin \text{HSA} \\
CC &= (\mathbf{CEN}(1) - \mathbf{P}(1)) \cdot ((\mathbf{VMBAR} \cdot (-\mathbf{ETA}(3)) + \mathbf{VF}(1) \cdot \cos \text{HSA} \cdot (-\mathbf{ETE}(3))) \\
&\quad - (\mathbf{CEN}(3) - \mathbf{P}(3)) \cdot ((\mathbf{VMBAR} \cdot (-\mathbf{ETA}(1)) + \mathbf{VF}(1) \cdot \cos \text{HSA} \cdot (-\mathbf{ETE}(1))))
\end{align*} \]

Two of the equations implied by (II-4) above are:

\[ t \cdot \mathbf{V}(1) = \mathbf{CEN}(1) - \mathbf{P}(1) \]  
\[ t \cdot \mathbf{V}(3) = \mathbf{CEN}(3) - \mathbf{P}(3). \]

Eliminating t we have:

\[ \mathbf{V}(1) \cdot (\mathbf{CEN}(3) - \mathbf{P}(3)) = \mathbf{V}(3) \cdot (\mathbf{CEN}(1) - \mathbf{P}(1)). \]  
\[ (II-5) \]

Using two components of (II-1) we have:

\[ AA \cdot E(1) + BB \cdot E(3) = CC. \]  
\[ (II-6) \]

From (2) and (3):

\[ E(1) \cdot \mathbf{ETE}(1) + E(2) \cdot \mathbf{ETE}(2) + E(3) \cdot \mathbf{ETE}(3) = 0 \]  
\[ E(1)^2 + E(2)^2 + E(3)^2 = 1 \]  
\[ (II-7) \]

If \( \mathbf{ETE}(2) \neq 0 \) and \( BB \neq 0 \), the solution of Equations II-6, II-7, and II-8 for \( E(1), E(2), \) and \( E(3) \) is straight forward. In case there is no real solution, the program ignores this step, moves the warhead along another step and tries again. In general there are two distinct solutions because (II-8) is quadratic.
If \( ETE(2) = 0 \), the solution for both sets of roots is straightforward unless \( ETE(1) \times BB = ETE(3) \times AA \). If \( ETE(1) \times BB = ETE(3) \times AA \) there are either an infinite number of solutions or none at all. In either case, this step is ignored and is treated as if there were no solution.

If \( BB = 0 \), the solution is straightforward unless \( AA \neq 0 \). But, \( AA = BB = 0 \) if, and only if, \( CEN(1) - P(1) = CEN(3) - P(3) = 0 \). This means that the warhead is positioned exactly on the trajectory line. In this case we proceed as if there are no real roots.

Now, assume we have found two real solutions. Let \( E(1), E(2), E(3) \) be either solution. Using equation (1) we compute \( V \). If \( V(1) \neq 0 \), \( t = (CEN(1) - P(1))/V(1) \). If \( V(1) = 0 \) and \( V(3) \neq 0 \), \( t = (CEN(3) - P(3))/V(3) \). If \( V(1) = V(3) = 0 \), something is wrong and program stops. If \( t < 0 \), problem has no solution of interest. Assuming \( t > 0 \), we can compute \( y = t \times V(2) + P(2) \) which is the \( y \) coordinate where these fragments hit the aim point trajectory line. If \( y < CEN(2) \) this solution is not of interest. Assuming \( y > CEN(2) \), we can compute the time required for fragments to reach \( y \) since we know \( V \). Thus we can fine:

\[
y_T = y
\]

where \( y_T \) is the \( y \) coordinate of the aim point when the fragments reached its trajectory line.

To facilitate the use of this option a distribution of those values of \( TD \) which were found is printed out.

6. BLAST PHENOMENA

This section determines whether or not the target is killed by blast effects.

There are \( NCB \) (input) points on the target called "blast centers". Associated with each blast center are one or more "blast ellipsoids". The \( x, y, z \) coordinates of the blast centers are specified by the input array, \( PCB(I, J) \) where \( I = 1, 2, 3 \) and \( J = 1, 2, ..., NCB \). The number of blast ellipsoids associated with the \( J \)th blast center (with coordinates \( PCB(I, J), I = 1, 2, 3 \)) is \( PCB(4, J) \). The characteristics of the blast ellipsoids are specified by the input array \( EDB(I, J) \) where \( I = 1, 2, ..., 6 \) and \( J = 1, 2, ..., NCB \).

\[
I = 1, 2, 3 \quad \text{are the } x, y, z \text{ coordinates of the } J\text{th blast ellipsoid. } I = 4, 5, 6 \text{ are the three axes of the } J\text{th blast ellipsoid, } EDB(4, J) \text{ being the } x \text{ axis of the ellipsoid, etc. In the } EDB \text{ array, the first } PCB(4, 1) \text{ ellipsoids correspond to the first blast center. The next } PCB(4, 2) \text{ ellipsoids correspond to the second blast center, etc. Thus there must be a total of:}
\]

\[
\sum_{J=1}^{NCB} PCB(4, J) \text{ ellipsoids specified.}
\]

Qualitatively speaking, the target will be killed by blast if the warhead detonation point is within a blast ellipsoid associated with a blast center at the time the blast wave (due to warhead's detonation) reaches the blast center.

We will first discuss the problem of determining the time at which the blast wave reaches a blast center. Figure II-6 illustrates the situation at the time the blast wave reaches a blast center.

\[
PCB = \text{position of blast center with respect to origin at time of detonation.}
\]
\[
PW = \text{position of warhead burst.}
\]
\[
T = \text{time when blast wave hits blast center. At instant of detonation, } T = 0.
\]
\[
f = \text{distance travelled by blast wave in time } T.
\]

We have

\[
f = fV - PCB - (0, V_{TBAR} \times T, 0).
\]
Let SR denote $|\vec{P}|$ and $\vec{Q} = \vec{P} - \vec{F}_b$. Then

$$SR = (Q^2(1) + (Q(2) - VT\bar{B}AR*T)^2 + Q^2(3))^{1/2}.$$ 

The input table of TAB12Y(I) versus TAB12X(I), $I = 1, 2, \ldots$ MBT is used to compute the time required for the blast wave to travel a distance SR. TAB12Y(I) is the time required (in milliseconds) for the blast wave to travel a distance TAB12X(I) if ambient pressure is 1 atmosphere and weight of explosive is 1 lb. A scale factor $C_{12A} = (X_{PP}/LL)^{1/3}$ is used for other conditions. The table is entered with the value $SR*C_{12A}$. Linear interpolation gives a value of AB. The time (in seconds) required to travel a distance SR is then $AB/(C_{12A}*1000)$.

In order to find the time (if any) when the blast wave strikes a blast center one of two different iteration schemes is used. In order to choose the appropriate iteration scheme we compute the time DTG required for the blast wave to reach the trajectory line of the blast center. The point where the blast wave first reaches the trajectory line is $(PCB(1, I), PW(2), PCB(3, I))$. Then, if the blast center has not reached the point $(PCB(1, I), PW(2), PCB(3, I))$ in DTG seconds the blast wave must hit it at some time and iteration scheme (a) in paragraph a below is used. Otherwise, iteration scheme (b) in paragraph b below is used.

a. Iteration Scheme (a)

In this case there must be a solution. We know an interval on the trajectory line which must contain the point where the blast wave hits the blast center. A straight forward iteration process involving linear interpolation is used to find the solution. In case you're wondering, iteration scheme (b) will not always work for this situation.

b. Iteration Scheme (b)

Let $f$ represent the functional relationship specified by the table of TAB12Y(I) versus TAB12X(I). Then we can write:

$$T = \frac{f(SR*C_{12A})}{1000*C_{12A}} = \frac{f(C_{12A}*(Q^2(1) + (Q(2) - VT\bar{B}AR*T)^2 + Q^2(3))^{1/2})}{1000*C_{12A}}.$$ 

If this equation has a solution $T \geq 0$, the blast wave strikes the blast center at time $T$. The program attempts to find the solution to this equation by a simple iteration process. A first estimate to the solution is $t_0 = $ time required for blast center to reach point $(PCB(1, I), PW(2), PCB(3, I))$. The general iteration equation is:

$$T_{m+1} = \frac{f(C_{12A}*(Q^2(1) + (Q(2) - VT\bar{B}AR*T_m)^2 + Q^2(3))^{1/2})}{1000*C_{12A}}.$$ 

If the process converges, then $\lim_{m \to \infty} |T_{m+1} - T_m| = 0$.

In the program, the iteration process is terminated when:

$$|T_{m+1} - T_m| < \text{RELR}\text{R} (\text{RELR} = 0.01).$$ 

It is also assumed that unless this test is satisfied in 12 iterations, the process will never converge.

The question naturally arises, under what conditions can convergence be guaranteed? It can be shown that, if the recursion equation is $T_{m+1} = f(T_m)$, then convergence will occur only if $|df/dT| < 1$ for values of $T$ in the neighborhood of the solution. In our problem:

$$\frac{df}{dT} = \frac{f(C_{12A}*(Q^2(1) + (Q(2) - VT\bar{B}AR*T)^2 + Q^2(3))^{1/2})}{1000*C_{12A}}.$$ 

76
Where \( f' \) denotes the slope of the curve specified by the table TAB12Y(I) versus TAB12X(I), i.e., \( f' \) depends only on the nature of the input table. Performing the indicated differentiation we have:

\[
\frac{df}{dt} = \frac{C12A*VTBAR*(VTBAR*T - W(2))}{C12A*1000 (Q^2(1) + Q^2(2) - VTBAR*T^2 + Q^2(3))^{1/2}}
\]

Thus, to guarantee convergence we must have:

\[
|f'| < 1000. \frac{(Q^2(1) + Q^2(2) - VTBAR*T^2 + Q^2(3))^{1/2}}{VTBAR*(VTBAR*T - Q(2))}
\]

or

\[
|f'| < 1000. \frac{Q^2(1) + Q^2(3)}{VTBAR*(VTBAR*T - Q(2))^{1/2}} + 1
\]

or, in terms of original variable \( f \)

\[
|f'| < 1000. \frac{P^2(1) + P^2(3)}{P^2(2)} + 1.
\]

In the program, if convergence does not occur, a message is printed out along with appropriate numbers which should aid in determining what is causing divergence.

Now, assume that convergence occurred and that we have the values of time, etc. corresponding to when the blast wave reached the blast center. The target will be killed by blast if the burst point is within any ellipsoid associated with the blast center. Then the \( J \)th ellipsoid satisfies the following equation in the \( x, y, z \) system:

\[
\frac{(x - EDB(1, J))^2}{EDB(4, J)^2} + \frac{(y - EDB(2, J))^2}{EDB(5, J)^2} + \frac{(z - EDB(3, J))^2}{EDB(6, J)^2} = 1.
\]

The point specified by PDSB will be within (or on) this ellipsoid if and only if

\[
\frac{(PDSB(1) - EDB(1, J))^2}{EDB(4, J)^2} + \frac{(PDSB(2) - EDB(2, J))^2}{EDB(5, J)^2} + \frac{(PDSB(3) - EDB(3, J))^2}{EDB(6, J)^2} \leq 1.
\]

7. DIRECT HIT

See Figure II-7.

It is possible that the target is hit directly by the interceptor prior to warhead detonation, in which case the target is killed. To determine whether or not a direct hit occurs, the physical dimensions of the target are represented by ellipsoids whose size and location (with respect to the \( x, y, z \) coordinate system) are specified by the input array EDH(I, J), \( I = 1, 2, \ldots, 6; J = 1, 2, \ldots, NEDH \).

A direct hit ellipsoid has the form (dropping second subscript hereafter):

\[
\frac{(x - EDH(1))^2}{EDH(4)^2} + \frac{(y - EDH(2))^2}{EDH(5)^2} + \frac{(z - EDH(3))^2}{EDH(6)^2} = 1.
\]

in the \( x, y, z \) coordinate system. Figure II-7 shows an \( x_T, y_T, z_T \) coordinate system. The \( x_T, y_T, z_T \) system is the \( x'', y'', z'' \) coordinate system translated to the point defined by the vector \( PT \). Using the rotation matrix derived in paragraph 1 one can show that the relationship between \( x, y, z \) and \( x_T, y_T, z_T \) is:

\[
\begin{pmatrix}
  x \\
  y \\
  z
\end{pmatrix} =
\begin{pmatrix}
  x_T \\
  y_T \\
  z_T
\end{pmatrix} + \begin{pmatrix}
  PT(1) \\
  PT(2) \\
  PT(3)
\end{pmatrix}
\]

The interceptor is moving along the \( y_T \) axis. If the direct hit ellipsoid is pierced by the \( y_T \) axis it is possible (depending on where detonation occurred) that the ellipsoid was hit prior to detonation. To find the points on the \( y_T \) axis which lie on the direct hit ellipsoid (if they exist) proceed as follows:
Figure II-7. Geometry of Direct Hit
In the equation for the ellipsoid given above substitute for \( x, y, z \) in terms of \( x_T, y_T, z_T \) and \( P_T \). This gives the equation of the ellipsoid in the \( x_T, y_T, z_T \) system.

To find points on the \( y_T \) axis which lie on the ellipsoid, set \( x_T = z_T = 0 \).

For convenience, let:

\[
\begin{align*}
E(K) &= P_T(K) - EDH(K) \\
R(K) &= EDH(K + 3)**2 \\
\text{for } K = 1, 2, 3.
\end{align*}
\]

Also, let \( \eta = y_T \). The resulting equation, after appropriately rearranging terms is:

\[
\frac{(P_{T1})^2}{R(1)} + \frac{(P_{T2})^2}{R(2)} + \frac{(P_{T3})^2}{R(3)} = \frac{(E(1))^2}{R(1)} + \frac{(E(2))^2}{R(2)} + \frac{(E(3))^2}{R(3)} - 1.0. 
\]

The program solves for the two roots of this equation. If they are complex, then the \( y_T \) axis does not intersect the ellipsoid. If real, the two roots are denoted by \( IMINUS \) and \( IPLUS \). The resulting equation, after appropriately rearranging terms is:

\[
\frac{(P_{T1})^2}{R(1)} + \frac{(P_{T2})^2}{R(2)} + \frac{(P_{T3})^2}{R(3)} = \frac{(E(1))^2}{R(1)} + \frac{(E(2))^2}{R(2)} + \frac{(E(3))^2}{R(3)} - 1.0. 
\]

The resulting equation, after appropriately rearranging terms is:

\[
\frac{(P_{T1})^2}{R(1)} + \frac{(P_{T2})^2}{R(2)} + \frac{(P_{T3})^2}{R(3)} = \frac{(E(1))^2}{R(1)} + \frac{(E(2))^2}{R(2)} + \frac{(E(3))^2}{R(3)} - 1.0. 
\]

Thus, if \( YWP \leq IPLUS \) or \( YWP \geq IMINUS \) a direct hit occurred.

9. **FRAGMENTATION**

There are certain specified components on the target which are vulnerable to fragments. The approximate centers (called vulnerable points) of these vulnerable components are specified by giving \( x, y, z \) coordinates of each point. These coordinates are the input array \( PV(I, J) \), \( I = 1, 2, 3; J = 1, 2, \ldots, NPV \) where \( NPV \) is the total number of vulnerable components. In order to determine whether a particular vulnerable component is killed by fragments several steps are required:

1. Determine how many (if any) times the vulnerable point corresponding to the vulnerable component is struck by a spray of fragments, assuming no shielding of the component. The performance of this step is the sole concern of this section.

2. Of the hits determined in 1., were any nullified by the presence of shielding? This step is discussed later in paragraph 11.

3. For each actual hit (by a spray of fragments) on a vulnerable point determine the probability that the component was killed by that particular hit. This step is discussed in paragraph 12.

The following input variables are important to this section:

- \( NPV \) = number of vulnerable components or points.
- \( NVEL \) = number of different static fragment velocities.
- \( NTHET \) = the number of angle values used to specify the different fragment spray zones of the warhead.
- \( VF(I), (I = 1, 2, \ldots, NVEL) \) = the \( NVEL \) nominal static fragment velocities.
- \( SVF \) = standard deviation on nominal static fragment velocity.
- \( PV(I, J), (I = 1, 2, 3; J = 1, 2, \ldots, NPV) \) = coordinates of the \( NPV \) vulnerable points.
- \( SF \) = single fragment weight in ounces.
- \( THETA(I), I = 1, 2, NTHET \) = the angles which define the static fragment spray zones. Zone 1 lies between \( THETA(I) \) and \( THETA(I + 1) \). \( THETA \)'s are measured from the body axis of the interceptor.
FRAG(I,J),(I-1,2,...,NTHET-1),(J-1,2,...,NVEL) = number of fragments with velocity
VF(J) in the fragment spray zone lying between THETA(I) and THETA(I+1).

See Figure II-4 for an illustration of the usage of some of these variables.

Now, consider the problem of determining whether or not a fragment with nominal initial static velocity of VF(K) can possibly hit the NTHET vulnerable point whose coordinates are PV(I,N), I=1,2,3. If atmospheric drag were ignored the problem can be solved in closed form for all possible situations. However, when drag is considered, some iterative technique must be used.

Until further notice, assume isotropic warhead. Figure II-8 illustrates the geometry involved for the case when the actual initial static velocity VFBAR (a sample from a Gaussian distribution with mean VF(K) and sigma SF) is less than the interceptor velocity VMBAR. (The cases when VFBAR > VMBAR and VFBAR = VMBAR will be discussed later). Fragments will hit the vulnerable point trajectory only between points A and B. We now discuss the procedure for finding points A and B.

The set of all lines passing through the burst point and tangent to the sphere form a cone: The cosine of the angle between any line of the set and the vector VMBAR is:

\[ \cos \theta = \frac{\sqrt{VMBAR^2 - VFBAR^2}}{VMBAR^2} \]

or

\[ \cos \theta = \frac{\sqrt{1 - VFBAR^2/VMBAR^2}}. \]

Let CK = 1 - VFBAR^2/VMBAR^2. Let \( \hat{z} \) be a vector from the burst point to any point (x, y, z) on the cone. Then:

\[ \hat{z} = (x - PW(1), y - PW(2), z - PW(3)). \]

The vector (-ETA) is a unit vector in the direction of VMBAR. Thus:

\[ - \hat{z} \cdot ETA = \hat{z} \cdot CK. \]

Let UVL = -ETA(1), UVM = -ETA(2), UVN = -ETA(3).

Then

\[ UVL \cdot CK + UVL \cdot PW(1) + UVM \cdot PW(2) + UVN \cdot PW(3) = \sqrt{(x - PW(1))^2 + (y - PW(2))^2 + (z - PW(3))^2}. \]

is the equation of the cone. To find where the cone intersects the trajectory of the vulnerable point let x = PV(1,N), z = PV(3,N). Then:

\[ UVL \cdot (PV(1,N) - PW(1)) + UVM \cdot (y - PW(2)) + UVN \cdot (z - PW(3)) = \sqrt{(PV(1,N) - PW(1))^2 + (y - PW(2))^2 + (z - PW(3))^2}. \]

defines the points A and B (if they exist). For convenience, let:

\[ X\check{N}E = PV(1,N) - PW(1) \]
\[ Z\check{N}E = PV(3,N) - PW(3) \]
\[ E = UVL \cdot X\check{N}E + UVN \cdot Z\check{N}E \]
\[ DEN = UVM^2 - CK. \]

Then the equation becomes:

\[ UVM \cdot (y - PW(2)) + E = \sqrt{(y - PW(2))^2 + X\check{N}E^2 + Z\check{N}E^2}. \]

80
Points A, B, and C form a plane containing the vulnerable point trajectory. Any fragment initially lying in this plane will eventually intersect the trajectory line of the vulnerable point.

- **Sphere of radius** \( \text{VFBA} \) with center at end of \( \text{VMBAR} \) vector
- **Line tangent to sphere and intersecting vulnerable point trajectory at point A.**
- **Initial velocity vector of a typical fragment travelling in plane formed by A, B, C.**
- **Line tangent to sphere and intersecting vulnerable point trajectory at Point B.**
- **Line along which vulnerable point moves with velocity \( \text{VFBA} \) (trajectory line).**
- **Position vector of vulnerable point at time of warhead burst.**
- **Projections in X, Y plane.**

**Figure II-8. Geometry of Fragmentation when VMBAR > VFBAR**
Squaring both sides and regrouping terms gives \((UVM^2 - CK) (y - PW(2))^2 + 2*UVM*E*(y - PW(2)) + E^2 - CK*(XNE^2 + ZNE^2) = 0\).

Let \(RAD = CK*(E^2 + DEN*(XNE^2 + ZNE^2))\).

Then the above equation has real roots if and only if \(RAD > 0\). (If \(RAD < 0\), cone does not intersect trajectory line of vulnerable point). If \(DEN \neq 0\) and \(RAD > 0\), the two roots are:

\[
\frac{-UVM*E + RAD}{DEN} + PW(2)
\]

\[
\frac{-UVM*E - RAD}{DEN} + PW(2).
\]

(We will consider the case \(DEN = 0\) later). Let \(ROOT(1) > ROOT(2)\) where \(ROOT(1)\) and \(ROOT(2)\) are the two roots. Unfortunately, one or both of the roots obtained may not have physical significance since the cone "mathematically" has two parts.

Consider a plane passing through the burst point and perpendicular to the vector \(VMBAR\). This plane intersects the vulnerable point trajectory at the point.

\[y_p = \frac{-E}{UVM} + PW(2).\]

Further, it can be shown that

1. If \(UVM < 0\), no fragments can hit a point \(y\) on the trajectory line unless \(y < y_p\).
2. If \(UVM > 0\), we must have \(y > y_p\).
3. If \(UVM = 0\), \(y_p\) does not exist, so further analysis is required.

The preceding analysis justifies the following conclusions:

1. If \(UVM < 0\) and \(y_p\) is not between \(ROOT(1)\) and \(ROOT(2)\), then a point \(y\) can be hit if, and only if \(ROOT(1) < y < ROOT(2)\) and \(y < y_p\).
2. If \(UVM < 0\) and \(y_p\) is between \(ROOT(1)\) and \(ROOT(2)\), then a point \(y\) can be hit if, and only if, \(y < e^{\text{ROOT}(1)}\).
3. If \(UVM > 0\) and \(y_p\) is not between \(ROOT(1)\) and \(ROOT(2)\) then a point \(y\) can be hit if, and only if, \(y < e^{\text{ROOT}(2)}\) and \(y > y_p\).
4. If \(UVM = 0\) and \(y_p\) is between \(ROOT(1)\) and \(ROOT(2)\) a point \(y\) can be hit if, and only if, \(y > e^{\text{ROOT}(2)}\).
5. If \(UVM = 0\), the vector \(VMBAR\) is perpendicular to the trajectory line. Consider the quantity \(D\text{otted}_{\text{P}} = (PW - PV).(-\text{ETA})\).
   - If \(D\text{otted}_{\text{P}} > 0\), it is impossible for fragments to intersect any point on the trajectory line.
   - If \(D\text{otted}_{\text{P}} < 0\), only the part of the trajectory line between \(ROOT(1)\) and \(ROOT(2)\) can be hit.
   - If \(D\text{otted}_{\text{P}} = 0\), then detonation occurred exactly on the trajectory line.
   - In this case, the vulnerable point is hit by fragments if, and only if, \(PW = PV\).

Now, consider the case when \(DEN = 0\), that is

\[UVM^2 = CK\]

or

\[UVM = \sqrt{1 - (VFBAR^2/VMBAR^2)}\].

It is easy to show that this is true if, and only if, one line (of the set of lines making up the cone) is parallel to the trajectory line. With \(DEN = 0\), the original quadratic equation in \((y = PW(2))\) becomes a linear equation whose only root is (assuming that \(UVM*E \neq 0\)).
Therefore

1. If $UVM < 0$, fragments can intersect the trajectory line at a point $y$ if, and only if, $y \leq \text{ROOT}(1)$ and $\text{ROOT}(1) \leq y_p$.

2. If $UVM > 0$, fragments can intersect the trajectory line at a point $y$ if, and only if, $y \geq \text{ROOT}(1)$ and $\text{ROOT}(1) \geq y_p$.

3. If $UVM = 0$, $\text{ROOT}(1)$ is not defined.

We must now consider the case when $DEN = 0$ and $UVM*E = 0$. In this case the original quadratic equation reduces to:

$$E^2 - CK*(XN^2 + ZN^2) = 0.$$
Sphere of radius $V_{FBA R}$ with center at end of $V_{FBA R}$ vector

$V_{FBA R}$ is the initial velocity of a fragment which hits Point Y

Point Y

Line along which vulnerable point moves with velocity $V_{FBA R}$ (trajectory line)

Position vector of vulnerable point at time of warhead burst

Projections in X, Y Plane

Figure II-9. Geometry of Fragmentation when $V_{MBA R} < V_{FBA R}$
We now have defined a region on the trajectory line, any point of which can definitely be hit by one or more fragments at one or more different times. Suppose that \( Y \) is the y coordinate of any point on the trajectory line between \( Y_{\text{MAX}} \) and \( Y_{\text{MIN}} \). (See Figure VIII). Then, to hit the point, a fragment must travel (in a negative sense) a distance \( |\vec{P}| \) along a vector \( \vec{P} \) where:

\[
\begin{align*}
P(1) &= P_W(1) - PV(1,N) \\
P(2) &= P_W(2) - Y \\
P(3) &= P_W(3) - PV(3,N).
\end{align*}
\]

The initial velocity of the fragment \( V_{\text{NBAR}} \) is:

\[
V_{\text{NBAR}} = V_{\text{BAR}} \cdot \cos \chi + J_{\text{vFBAR}}^2 - V_{\text{BAR}}^2 \cdot \sin^2 \chi
\]

where \( \cos \chi = \cos (\chi) \).

Obviously:

\[
\cos \chi = \frac{(\eta(1) \cdot P(1) + \eta(2) \cdot P(2) + \eta(3) \cdot P(3))}{A}
\]

where \( A = |\vec{P}| \). The following comments are in order:

1. If \( V_{\text{FBAR}}^2 - V_{\text{BAR}}^2 \cdot \sin^2 (\chi) < 0 \) there is a mistake somewhere. By definition, \( Y \) is a point which can be hit.

2. The \( \pm \) sign in the expression is significant. If \( V_{\text{NBAR}} \) is positive using both signs, there is a certainty that the point \( Y \) can be hit by two different fragments at two different times. If \( V_{\text{NBAR}} > 0 \) for both signs, all of the remaining steps in this entire paragraph must be gone through using each value of \( V_{\text{NBAR}} \) separately. Henceforth, we merely assume that \( V_{\text{NBAR}} > 0 \).

We are now in a position to compute the time \( T_{\text{FRAG}} \) at which a fragment with initial velocity \( V_{\text{NBAR}} \) would reach point \( Y \).

\[
T_{\text{FRAG}} = \frac{A}{C_{\text{11A}}}e^{\frac{-1}{V_{\text{BAR}}}}
\]

where

\[
C_{\text{11A}} = \frac{M_F/M}{S_F R_{\text{H0}}}
\]

where \( M_F, S_F, \) and \( R_{\text{H0}} \) are all input numbers. The vulnerable point will arrive at the point \( Y \) at a time \( T_{\text{VP}} \):

\[
T_{\text{VP}} = \frac{(Y - PV(2,N))}{V_{\text{BAR}}}
\]

Thus, a fragment with initial velocity \( V_{\text{NBAR}} \) will hit the vulnerable point if, and only if, \( T_{\text{FRAG}} - T_{\text{VP}} = 0 \). Thus, we will know where and when a fragment can hit the vulnerable point if we can find a point \( Y \) such that \( T_{\text{FRAG}} - T_{\text{VP}} = 0 \).

For any point \( Y \), let:

\[
D_T = T_{\text{FRAG}} - T_{\text{VP}}.
\]

\( D_T \) is the difference in time of arrival at point \( Y \) of a fragment and the vulnerable point, i.e., \( D_T \) is a function of \( Y \). One could theoretically draw a graph of \( D_T \) versus \( Y \) for \( Y_{\text{MIN}} \leq Y \leq Y_{\text{MAX}} \).
The curve can intersect the Y axis once, twice, or not at all. It cannot intersect the axis more than twice. We attempt to find the root(s) by an iterative process. The process uses a set of three equally spaced values of Y denoted by \( Y_1, Y_2, Y_3 \) (\( DY = Y_2 - Y_1 = Y_3 - Y_2 \)) where \( Y_1 < Y_2 < Y_3 \) and their corresponding values of DT, DT1, DT2, DT3.

The process begins with

\[
Y_1 = Y_{\text{MIN}}, \quad Y_2 = \frac{Y_{\text{MIN}} + Y_{\text{MAX}}}{2}, \quad Y_3 = Y_{\text{MAX}}. \tag{1}
\]

The resulting set of three DT values on the curve are fit to a parabola of the form \( DT = ay^2 + by + c \) where \( a, b, c \) are constants. The roots of this equation are found. There are several possible results:

1. **Complex Roots** - In this case the value of Y for which DT is a minimum is found, denoted by \( Y_{\text{EXT}} \). If \( Y_1 < Y_{\text{EXT}} < Y_3 \), the process is repeated using \( Y_{\text{EXT}} \) as the \( Y_2 \) of an appropriate set of Y's. If \( Y_{\text{EXT}} > Y_3 \) or \( Y_{\text{EXT}} < Y_1 \) we assume there is no root and the process terminates.

2. **Distinct Real Roots** - If neither root is between \( Y_1 \) and \( Y_3 \), the search is terminated. If only one root \( R_1 \) is between \( Y_1 \) and \( Y_3 \), the process is continued with \( Y_3 = R_1 \) and \( DT = \text{(old value of DT)}/(1.2) \). If both roots are between \( Y_{\text{MIN}} \) and \( Y_{\text{MAX}} \) an attempt is made to separate the roots. If it proves impossible to separate them in 10 tries, the search is terminated.

3. **In Case \( a = 0 \)** - There Is Only One Root - It is handled similar to case 2 above. Of course this whole process has to be terminated at some arbitrary step because it will normally never arrive precisely at a root. Therefore, the process is terminated if either of the following conditions are met at any step in the procedure:
   - More than 20 complete iterations have already been performed;
   - \( DY <YT_0L \) (an input number);
   - \(|DT|/TFRAG < TTOL \) (an input number);

It is possible for a particular vulnerable point to be hit by two distinct sprays of fragments. The times (TFRAG) when the fragments hit are different, as well as the striking geometry, fragment density, etc. Since there are three possible initial static fragment velocities, it is possible for a given vulnerable point to be hit a total of six times. The way these multiple hits are handled will be discussed in paragraph 12.

Now, assume that we have found a particular time and place where the vulnerable point is struck by a spray of fragments from an isotropic warhead. Since the warhead is not isotropic, we must determine if the fragment could have come from the non-isotropic warhead. It did if, and only if, the angle \( DUM \) between \( V_{\text{F BAR}} \) and \( -ETE \) is between \( \Theta_1 \) and \( \Theta_{\text{NTHET}} \). Now:

\[
DUM = \arccos \left( \frac{(-ETE) \cdot V_{\text{F BAR}}}{|V_{\text{F BAR}}|} \right)
\]

But

\[
V_{\text{F BAR}} = V_{\text{N BAR}} - V_{\text{M BAR}} = V_{\text{N BAR}} + V_{\text{M BAR}} \tag{2}
\]

So

\[
V_{\text{N BAR}} = - V_{\text{N BAR}} \tag{3}
\]

and

\[
DUM = \arccos \left( \frac{\Theta_1 (V_{\text{N BAR}} \cdot \frac{\overline{P}}{|P|} - V_{\text{M BAR}} (\Theta_1))}{|V_{\text{F BAR}}|} \right)
\]

\[
\arccos \left( \frac{\Theta_1 (V_{\text{N BAR}} \cdot \frac{\overline{P}}{|P|} - V_{\text{M BAR}}) \Theta_1}{|V_{\text{F BAR}}|} \right).
\]

If \( \Theta_1 \leq DUM \leq \Theta_{\text{NTHET}} \) we determine the angular zone which contains \( DUM \).
9. SHIELDING

Suppose now that the procedure discussed in paragraph 8 has indicated that a spray of fragments has hit a vulnerable point. We must now determine whether shielding of the vulnerable point prevented it from being hit. The question we ask is, was the shield between the burst point and the vulnerable point at time TFRAG? This is not the same as asking, did the shield prevent the vulnerable point from being hit? However, we assume that the former question is adequate for our purposes.

The NSHE shielding ellipsoids are specified by the input array SHIELD(I,J), I = 1, 2, ..., 6; J = 1, 2, ..., NSHE in a manner similar to blast ellipsoids. See Figure 11-10. The equation of a shielding ellipsoid at time = TFRAG is (dropping subscript J):

\[
\frac{(x - \text{SHIELD}(1))^2}{\text{SHIELD}(4)^2} + \frac{(y - \text{SHIELD}(2) - \text{VTRAC} \cdot \text{TFRAG})^2}{\text{SHIELD}(5)^2} + \frac{(z - \text{SHIELD}(3))^2}{\text{SHIELD}(6)^2} = 1.
\]

Let PUV be a unit vector along the line traveled by the fragments. PUV points in the direction opposite to fragment travel. The x, y, z coordinates of any point on the fragment travel line can be expressed as:

\[
x = \text{PV}(1,N) + T \cdot \text{PUV}(1)
\]
\[
y = \text{PV}(2,N) + T \cdot \text{PUV}(2) + \text{VTRAC} \cdot \text{VTBAR}
\]
\[
z = \text{PV}(3,N) + T \cdot \text{PUV}(3)
\]

where T is a parameter. (Actually |T| is the distance from the vulnerable point to the point in question at time = TFRAG). We wish to find the values of T which correspond to points which lie on the shielding ellipsoid. Thus, solve the equation:

Figure 11-10. Shielding Geometry
\[
\left( PV(1,N) + T \cdot PUV(1) - SHIELD(1) \right)^2 +
\left( SHIELD(4) \right)^2
\]
\[
\left( PV(2,N) + T \cdot PUV(2) + TFRAG \cdot VTBAR - SHIELD(2) - VTBAR \cdot TFRAG \right)^2 +
\left( SHIELD(5) \right)^2
\]
\[
\left( PV(3,N) + T \cdot PUV(3) - SHIELD(3) \right)^2 = 1
\]
\[
\left( SHIELD(6) \right)^2
\]
for T. For convenience, let
\[
E(I) = \frac{1}{\left( SHIELD(I+3) \right)^2}, \quad I = 1, 2, 3
\]
\[
R(I) = PV(I,N) - SHIELD(I).
\]

Thus:
\[
E(1) \left( 1 + T \cdot PUV(1) \right)^2 + E(2) \left( (R(2) + T \cdot PUV(2)) \right)^2 + E(3) \left( (R(3) + T \cdot PUV(3)) \right)^2 = 1
\]
or
\[
T^2 \left[ E(1) \cdot PUV(1)^2 + E(2) \cdot PUV(2)^2 + E(3) \cdot PUV(3)^2 \right]
\]
\[
+ \left( 2 \cdot T \left[ E(1) RV(1) + E(2) RV(2) \cdot PUV(2) + E(3) RV(3) \cdot PUV(3) \right]
\]
\[
\left[ E(1) RV(1)^2 + E(2) RV(2)^2 + E(3) RV(3)^2 - 1 \right] = 0.
\]

If the two roots are complex, the shield is not intersected by the fragment travel line. If both roots are real consider each root \( \text{ROOT} \) separately as follows:

1. If \( \text{ROOT} < 0 \), shield intersected fragment travel line on "other side" of vulnerable point;
2. If \( 0 < \text{ROOT} < |P| \) (where \( P \) was defined in Section 10), the shield was between the burst point at time \( TFRAG \);
3. If \( \text{ROOT} > |P| \), shield intersects fragment travel line on "other side" of burst point.

By definition, a shielding ellipsoid is never penetrated by fragments. Thus, if either root satisfies (2) the spray of fragments under consideration do not hit or damage the vulnerable point. Of course, this does not rule out the possibility of hitting the vulnerable point with some other spray of fragments.

10. COMPUTATION OF PROBABILITY THAT A PARTICULAR VULNERABLE COMPONENT IS KILLED WHEN STRUCK BY FRAGMENTS

If a spray of fragments hits a vulnerable point, the probability that the corresponding component is killed is \( P = 1 - \exp(-RH0D \cdot AV) \) where \( RH0D \) is the density of fragments and \( AV \) is the "vulnerable area" of the component. This expression for \( P \) established as described below.

It is assumed that the probability of \( n \) hits is controlled by the Poisson distribution:
\[
P_n = \frac{m^n}{n!} \exp(-m)
\]
where \( m \) is the average number of hits expected and in this case is simply the product of the fragment density \( RH0D \) and the target area \( A \). Thus the probability of at least one hit is
\[
P_1 = 1 - \exp(-A \cdot RH0D).
\]

However, one or more hits do not necessarily kill the component. To account for this, \( A \) is modified. The \( A \) term is replaced by \( AV = p \cdot A \) where \( p \) is the probability that the component is killed, given that it is hit by one or more fragments. The computation of \( RH0D \) and \( AV \) are complicated. We discuss each separately.
a. Computation of \( RH_0 \)

(This method is taken from the report ATL-TDR-64-48 on the Northrop Nortronics Air-to-Air Warhead Effectiveness Program). The equation for fragment density on the surface of a spherical segment may be computed by dividing the number of fragments in the segment by the area of the segment, thus:

\[
RH_0 = \frac{FNF}{2\pi \cdot PMAG^2 \cdot (\cos \varphi_1 - \cos \varphi_2)} \quad \text{Fragments/Steradian}
\]

where \( FNF \) is the number of fragments contained between \( \varphi_1 \) and \( \varphi_2 \),
\( PMAG \) is the distance traveled by the fragments,
\( \varphi_1 \) and \( \varphi_2 \) the angles which bound the fragment zone. \( \varphi_1 \) and \( \varphi_2 \) are measured from the missile body axis vector. (See Figure II-11.)

The equation for \( RH_0 \) is valid if, and only if:

1. There is no angle of attack, i.e., the missile velocity vector coincides with the missile axis vector;
2. The warhead has no velocity at the time of detonation;
3. There is no drag

In reality, neither 1, 2, nor 3 is true. However, we assume that angle of attack is zero, and all derivations which follow are only approximate. We do attempt to correct for 2 and 3. Because 2 and 3 are not true the spherical wave is distorted and more fragments strike the vulnerable area than are contained in an equal area on the surface of the segment. the radius of the spherical section is no longer \( PMAG \), but is now \( RC \), the radius of curvature, and the density must be computed in the direction of \( RC \), normal to the wave. Thus, the fragment density striking the vulnerable area can be computed by dividing \( RH_0 \) by the cosine of the angle between the strike vector \( PMAG \) and the radius of curvature \( RC \), and replacing \( PMAG \) with \( RC \). Thus we must find the angle \( \beta \) and \( RC \). Solution is as follows.

The diagrams in Figure II-11 lie in the plane determined by the missile velocity vector and the fragment trajectory vector. For convenience, we choose an \( x \) and \( y \) coordinate system. \( x \) points along the missile velocity vector and \( y \) is perpendicular to \( x \) as in Figure II-11(a):

\[
RH_0 = \frac{FNF}{2\pi \cdot (RC)^2 \cdot (\cos \varphi_1 - \cos \varphi_2)}
\]

\[
PMAG = C11A \ln \left( 1 + \frac{VNBAR}{C11A} \cdot TFRAG \right)
\]

\[
VNBAR = VNBAR \cdot \cos \varphi + SIGH \cdot \sqrt{\varphi^2 - VNBAR^2 \cdot \sin^2 \varphi}
\]

Consider the function

\[
F = PMAG - C11A \ln \left( 1 + \frac{VNBAR}{C11A} \cdot TFRAG \right) = 0.
\]

The direction of the normal can be computed from gradient of \( F \).

\[
\text{Grad } F = \left( \frac{\partial F}{\partial x}, \frac{\partial F}{\partial y} \right) = \sqrt{\left( \frac{\partial F}{\partial x} \right)^2 + \left( \frac{\partial F}{\partial y} \right)^2}
\]

\[
\left( \frac{\partial F}{\partial x}, \frac{\partial F}{\partial y} \right) = \left( 3PMAG \cdot \frac{\partial \varphi}{\partial x}, 3PMAG \cdot \frac{\partial \varphi}{\partial y} \right)
\]

89
\[ \begin{align*}
\frac{\partial F}{\partial PMAG} &= 1 \\
\frac{\partial F}{\partial PMAG} &= - \frac{TFRAG}{CIL} \cdot \frac{\partial VMBAR}{\partial TFRAG + 1} \\
\frac{\partial VMBAR}{\partial TFRAG} &= -VMBAR \cdot \sin\theta + \frac{\text{SIGN} \left( \frac{1}{2} \right) \frac{VMBAR^2}{\sqrt{V^2 - VMBAR^2 \sin^2\theta}}}{\sqrt{V^2 - VMBAR^2 \sin^2\theta}} \\
&= \frac{-VMBAR \cdot \sin\theta - \text{SIGN} \cdot VMBAR^2 \cdot \sin\theta \cdot \cos\theta}{\sqrt{V^2 - VMBAR^2 \sin^2\theta}} \\
&= \frac{VMBAR \cdot \sin\theta - VMBAR^2 \cdot \sin\theta \cdot \cos\theta - VMBAR^2 \cdot \sin\theta \cdot \cos\theta}{VMBAR \cdot VMBAR^2 \cdot \sin\theta} \\
&= \frac{VMBAR \cdot \sin\theta \cdot VMBAR + VMBAR^2 \cdot \sin\theta \cdot \cos\theta - VMBAR^2 \cdot \sin\theta \cdot \cos\theta}{VMBAR \cdot VMBAR^2 \cdot \sin\theta} \\
&= \frac{VMBAR \cdot \sin\theta \cdot VMBAR + VMBAR^2 \cdot \sin\theta \cdot \cos\theta - VMBAR^2 \cdot \sin\theta \cdot \cos\theta}{VMBAR \cdot VMBAR^2 \cdot \sin\theta}
\end{align*} \]

\[ \begin{align*}
\frac{\partial F}{\partial PMAG} &= \frac{\partial CIL}{\partial TFRAG} \\
V &= \sqrt{x^2 + y^2} \\
\frac{\partial V}{\partial x} &= \frac{-y}{\sqrt{x^2 + y^2}} = -\sin\theta \\
\frac{\partial V}{\partial y} &= \frac{x}{\sqrt{x^2 + y^2}} = \cos\theta \\
\frac{\partial V}{\partial x} &= \frac{x}{\sqrt{x^2 + y^2}} = \cos\theta \\
\frac{\partial V}{\partial y} &= \frac{y}{\sqrt{x^2 + y^2}} = \sin\theta
\end{align*} \]
Hence:

\[
\begin{pmatrix}
\frac{\partial F}{\partial x} \\
\frac{\partial F}{\partial y}
\end{pmatrix} =
\begin{pmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{pmatrix}
\begin{pmatrix}
1 \\
c\sin \theta
\end{pmatrix}
\]

\[
\frac{\partial F}{\partial x} = \cos \theta - \frac{c}{\text{PMAG}} \sin^2 \theta
\]

\[
\frac{\partial F}{\partial y} = \sin \theta + \frac{c}{\text{PMAG}} \sin \theta \cos \theta
\]

\[
\left(\frac{\partial F}{\partial x}\right)^2 = \cos^2 \theta - 2\frac{c}{\text{PMAG}} \sin^2 \theta \cos \theta + \frac{c^2}{\text{PMAG}^2} \sin^4 \theta
\]

\[
\left(\frac{\partial F}{\partial y}\right)^2 = \sin^2 \theta + 2\frac{c}{\text{PMAG}} \sin^2 \theta \cos \theta + \frac{c^2}{\text{PMAG}^2} \sin^2 \theta \cos^2 \theta
\]

\[
\left(\frac{\partial F}{\partial x}\right)^2 + \left(\frac{\partial F}{\partial y}\right)^2 = 1 + \frac{c^2}{\text{PMAG}^2} \sin^2 \theta
\]

Let:

\[
D = \sqrt{\left(\frac{\partial F}{\partial x}\right)^2 + \left(\frac{\partial F}{\partial y}\right)^2} = \sqrt{1 + \frac{c^2}{\text{PMAG}^2} \sin^2 \theta}
\]

Then

\[
\text{Grad } F = \frac{1}{D} \left[ (\cos \theta - \frac{c}{\text{PMAG}} \sin^2 \theta) \hat{i} + (\sin \theta + \frac{c}{\text{PMAG}} \sin \theta \cos \theta) \hat{j} \right]
\]

A unit vector in direction of PMAG is \([\cos \theta \hat{i} + \sin \theta \hat{j}]\).

Thus:

\[
\cos \theta = \text{Grad } F \cdot (\cos \theta \hat{i} + \sin \theta \hat{j})
\]

\[
= \frac{1}{D} \left[ \cos^2 \theta - \frac{c}{\text{PMAG}} \sin^2 \theta \cos \theta + \sin^2 \theta + \frac{c}{\text{PMAG}} \sin^2 \theta \cos \theta \right] = \frac{1}{D}
\]

The length of RC is computed as follows:

\[
\sin \phi = \frac{\frac{3F}{D}}{\frac{\partial F}{\partial y}}
\]

\[
\text{RC} = \frac{\text{PMAG} \cdot \sin \phi}{\sin \phi} = \frac{D \cdot \text{PMAG} \cdot \sin \phi}{\frac{3F}{\partial y}}
\]

\[
= \frac{D \cdot \text{PMAG}}{1 + \frac{c}{\text{PMAG}} \cos \theta}
\]

92
Hence:

\[ RH_01 = \frac{FHF \cdot D}{2\pi R_1^2 (\cos \theta_1 - \cos \theta_2)} \]

since \( \cos \beta = \frac{1}{D} \).

The equation for \( RH_01 \) is valid only if the static angles \( \theta_1 \) and \( \theta_2 \) are the same as the dynamic angles which limit the sector. The remaining problem is to determine how to modify \( RH_01 \) when the dynamic angles are different from the static angles. See Figure II-11 (b) and (c).

Let \( RH_0D \) be the corrected value of fragment density. Let \( FRAG \) be the total number of fragments in the spray zone defined by \( \theta_1 \) and \( \theta_2 \). Then:

\[ FRAG = 2\pi R_1^2 \int_{\theta_1}^{\theta_2} \sin \alpha S \cdot RH_01 \cdot d\alpha \]

\[ = 2\pi R_1^2 \int_{\alpha_1}^{\alpha_2} \sin \alpha D \cdot RH_0D \cdot d\alpha \]

Taking differential and dividing through by \( 2\pi R_1^2 \) gives:

\[ RH_0D = \frac{\sin \alpha S \cdot \alpha_1}{\sin \alpha D \cdot \alpha_1} \cdot RH_01 \]

From Figure II-11(b): \( \sin \alpha = \frac{R_1}{PMAG} \cdot \sin \alpha D \).

From Figure II-11(c): \( \sin \alpha = \frac{VFBAR}{VBAR} \cdot \sin \alpha S \).

Hence:

\[ \frac{\sin \alpha S}{\sin \alpha D} = \frac{R_1}{PMAG} \cdot \frac{VFBAR}{VBAR} \]

From Figure II-11(b): \( \alpha = \beta + \delta \).

From Figure II-11(c): \( \alpha = \beta + \alpha \).

Hence:

\[ \frac{d\alpha}{d\beta} + \frac{d\alpha}{d\delta} = \frac{1 + \frac{d\alpha}{d\beta}}{\frac{d\alpha}{d\delta}} \]

From Figure II-11(b):

\[ R_1^2 = PMAG^2 + DIST^2 - 2 \cdot PMAG \cdot DIST \cdot \cos \delta \]

\[ DIST^2 = PMAG^2 + R_1^2 - 2 \cdot PMAG \cdot R_1 \cdot \cos \delta \]

From Figure II-11(c):

\[ VBAR^2 = VBAR^2 + VBAR^2 - 2 \cdot VBAR \cdot VBAR \cdot \cos \delta \]

93
\[ V_{MBAR}^2 = V_{NBAR}^2 + V_{FBAR}^2 - 2 \cdot V_{NBAR} \cdot V_{FBAR} \cdot \cos \alpha. \]

Note that DIST, \( RC \), \( V_{FBAR} \) and \( V_{MBAR} \) are taken as constants.

Thus:

\[
\frac{\cos \alpha}{2} = \frac{V_{NBAR}^2 + V_{FBAR}^2 - V_{MBAR}^2}{2 \cdot V_{NBAR} \cdot V_{FBAR}}.
\]

\[ DIST = \left( PMAG^2 + RC^2 - 2 \cdot PMAG \cdot RC \cdot \cos \theta \right)^{1/2}. \]

Taking derivative with respect to \( \theta \):

\[
0 = 2 \frac{V_{NBAR}}{3 \theta} - 2 \frac{\cos \alpha \cdot V_{FBAR}}{3 \theta} + 2 \frac{V_{NBAR} \cdot V_{FBAR} \cdot \sin \alpha}{3 \theta}.
\]

Since:

\[
\frac{V_{NBAR}}{3 \theta} = \frac{V_{NBAR} \cdot \sin \theta \cdot V_{FBAR}}{V_{NBAR} - V_{MBAR} \cdot \cos \theta},
\]

\[
0 = -V_{NBAR} \cdot \sin \theta \cdot V_{FBAR} \quad \frac{V_{NBAR} - V_{FBAR}^2 \cdot V_{FBAR}^2 - V_{MBAR}^2}{2V_{NBAR} \cdot V_{FBAR}} \cdot V_{FBAR}^2\]

\[
+ V_{NBAR} \cdot V_{FBAR} \cdot \sin \alpha \cdot \frac{3 \alpha}{3 \theta}.
\]

\[
\frac{3 \alpha}{3 \theta} = \frac{V_{MBAR} \cdot \sin \theta \cdot V_{FBAR}^2 - V_{MBAR} \cdot \cos \theta \cdot \frac{2V_{NBAR} \cdot V_{FBAR}^2 - V_{MBAR} \cdot V_{FBAR}}{\sin \alpha \cdot 2V_{NBAR} \cdot V_{FBAR}}}{V_{NBAR} - V_{MBAR} \cdot \cos \theta}.
\]

But, from Figure II-ll(c):

\[ V_{FBAR} = \frac{\sin \theta \cdot V_{NBAR}}{\sin \alpha}. \]

So:

\[
\frac{3 \alpha}{3 \theta} = \frac{V_{NBAR}^2 - V_{FBAR}^2 + V_{MBAR}^2}{2V_{NBAR} \cdot \left( V_{NBAR} - V_{MBAR} \cdot \cos \theta \right)}
\]

\[ = \frac{V_{NBAR}^2 + V_{FBAR}^2 - 2V_{FBAR}^2 - V_{MBAR}^2 + 2V_{MBAR}^2}{2V_{NBAR} \cdot V_{FBAR} \cdot \left( V_{NBAR} - V_{MBAR} \cdot \cos \theta \right)} \cdot \frac{V_{FBAR}}{V_{NBAR} - V_{MBAR} \cdot \cos \theta} - \frac{\cos \alpha \cdot V_{MBAR}^2 - V_{FBAR}^2}{V_{NBAR} \cdot V_{FBAR}}.
\]

But:

\[ 2 \frac{V_{FBAR} \cos \alpha}{V_{NBAR}} = \frac{V_{NBAR}^2 + V_{FBAR}^2 - V_{MBAR}^2}{V_{NBAR}}. \]

So:

\[ \frac{V_{NBAR}^2 - V_{FBAR}^2}{V_{NBAR}} = 2 \frac{V_{FBAR} \cos \alpha + V_{NBAR}}{V_{NBAR}}. \]

So:

\[ \frac{3 \alpha}{3 \theta} = \frac{V_{NBAR} - V_{FBAR} \cos \theta}{\left( V_{NBAR} - V_{MBAR} \cdot \cos \theta \right)}.
\]

Taking derivative of \( DIST^2 \) with respect to \( \theta \) gives:
Taking derivative of $RC^2$ with respect to $\theta$ gives:

$$0 = 2 \frac{dPMAG}{d\theta} - 2 \frac{dPMAG}{d\theta} \cdot \text{DIST} \cos \theta + 2 \frac{dPMAG}{d\theta} \cdot \text{DIST} \cdot \sin \theta$$

so

$$\frac{dPMAG}{d\theta} = \frac{PMAG \cdot \text{DIST} \cdot \sin \theta}{PMAG \cdot \text{DIST} \cdot \cos \theta}$$

Hence:

$$\frac{d\theta}{d\theta} = \frac{PMAG \cdot \text{DIST} \cdot \sin \theta}{PMAG \cdot \text{DIST} \cdot \cos \theta} (PMAG - RC \cos \theta)$$

Since:

$$\frac{\text{DIST}}{\sin \theta} = \frac{RC}{\sin \theta}$$

$$\frac{d\theta}{d\theta} = \frac{PMAG - RC \cdot \cos \theta}{PMAG - \text{DIST} \cdot \cos \theta}$$

Taking all these expressions together, we have:

$$\text{RHFD} = \frac{\text{PMF.D}}{2m (\cos \theta_1 - \cos \theta_2)} \cdot \frac{\text{RC VNBAR}}{\text{PMAG VFBAR}} \cdot \frac{1}{\frac{\text{VTBAR} \cdot \text{VFBAR} \cos \phi}{\text{VNBAR} \cdot \text{VFBAR} \cos \phi} + \frac{1}{\frac{\text{PMAG} - \text{RC} \cdot \cos \phi}{\text{PMAG} - \text{DIST} \cdot \cos \phi}}}$$

RHFD is the desired fragment density.

b. Computation of AV

There are three different ways to specify and compute AV. The input variable IAV determines which method to use for any particular run.

1) Missile Type Fragment Vulnerability Data (IAV < 0)

If the target is a missile or a warhead, its vulnerability to fragments can usually be described best by this type data. For this type target AV is a function of the fragment striking velocity $DVIMP$ and one strike angle $AVANG$. (See Figure II-12.) In this case there can be only one vulnerable point:

$$\text{VNIMP}$$ is computed as:

$$\text{VNIMP} = \text{VNBAR}/(1. + \text{VNBAR} \cdot \text{TFRAG}/\text{IAV})$$

The unit vector $\text{PUV}$ is in the direction of $-\text{VNIMP}$. Since $\text{VTBAR}$ has a component only in the y direction we have:

$$\text{DVIMP} = \begin{pmatrix}
-\text{VNIMP} \cdot \text{PUV}(1) \\
-\text{VNIMP} \cdot \text{PUV}(2) - \text{VTBAR} \\
-\text{VNIMP} \cdot \text{PUV}(3)
\end{pmatrix}$$
We let:

\[
\begin{align*}
\text{DELX} &= V\text{NIMP} \times P\text{UV}(1) \\
\text{DELY} &= V\text{NIMP} \times P\text{UV}(2) + V\text{TBAR} \\
\text{DELZ} &= V\text{NIMP} \times P\text{UV}(3)
\end{align*}
\]

Thus \(\text{AVANG} = \arccos\left(-\frac{\text{DVIMP}(2)}{\text{DVIMP}}\right)\)

\(= \arccos\left(\frac{\text{DELY}}{\text{DVIMP}}\right)\).

The values AVANG and DVIMP are used as entry points for linear interpolation for AV in the table AVTMV(I,J) versus VSBART(I) and AVANGT(J) where \(I = 1,2,\ldots,\text{NPVELT}\) and \(J = 1,2,\ldots,\text{NPANG}\).

\(2\) Box Type Data (IAV = 0)

This type data implies that the vulnerable components vulnerability can be specified using the six sides of a "box" around the vulnerable component. (See Figure II-12.) For each component there are six separate tables, one for each side. Each table is a function only of strike velocity. The values of vulnerable area in the table for a particular side represent the vulnerable area of the component when the strike velocity DVIMP is into and perpendicular to the particular side. If the angle \(\beta\) between the side and DVIMP is other than 90 degrees, the vulnerable area from that side is taken to be sin \(\beta\) times the value of vulnerable area in the table. The total vulnerable area is the sum of the vulnerable areas of the three sides which are hit. (No more than three sides can be hit):

\[
\begin{align*}
\text{DELY} &= 0 \text{ hits front} \\
&= 0 \text{ hits neither front or rear} \\
&< 0 \text{ hits rear} \\
\text{DELZ} &= 0 \text{ hits top} \\
&= 0 \text{ hits neither top or bottom} \\
&< 0 \text{ hits bottom} \\
\text{DELX} &= 0 \text{ hits neither left or right sides} \\
&< 0 \text{ hits right side}
\end{align*}
\]

Using DVIMP to interpolate linearly in the appropriate tables one obtains AV1, AV2, AV3 which correspond to front or rear, top or bottom, left or right.

Thus, the total vulnerable area AV is:

\[
\text{AV} = \frac{(AV1 \times \text{ABS}(\text{DELY}) + AV2 \times \text{ABS}(\text{DELZ}) + AV3 \times \text{ABS}(\text{DELX}))}{\text{DVIMP}}
\]

\(3\) BRL 1732 Type Data (IAV >0)

Tables of vulnerable area in this case are a function of a strike angle (called polar angle A), a strike angle (called polar angle B) and the strike velocity DVIMP. There is a distinct table for each vulnerable component. The data for the \(N\)th component is:

\[
\begin{align*}
\text{AVTAB}(I,N) \text{ where } I = 1,2,\ldots,\text{NPA}(N) \times \text{NPB}(N) \\
\text{POLAT}(I,N) \text{ where } I = 1,2,\ldots,\text{NPA}(N) \\
\text{POLBT}(I,N) \text{ where } I = 1,2,\ldots,\text{NPB}(N) \\
\text{DVTAB}(I,N) \text{ where } I = 1,2,\ldots,\text{NDV}(N)
\end{align*}
\]

POLAT(I,N) is an array of values of polar angle A. POLBT(I,N) is an array of values of polar angle B. DVTAB(I,N) is an array of values of strike velocity. The table AVTAB(I,N) contains values of vulnerable area. The entries AVTAB(1,N), AVTAB(2,N), ..., AVTAB(NPA(N),N) are the values of vulnerable area corresponding to a polar angle B of POLBT(1,N), strike velocity DVTAB(1,N), and polar angle A values of POLAT(1,N), POLAT(2,N), ..., POLAT(NPA(N),N). The entries AVTAB(NPA(N) + 1, N), ..., AVTAB(2 * NPA(N),N) correspond to a polar angle B of POLBT(1,N), strike velocity DVTAB(1,N), and polar angle A values POLAT(1,N), ..., POLAT(NPA(N),N). In like manner, AVTAB(2 * NPA(N) + 1, N), ..., AVTAB(3 * NPA(N),N) correspond to POLBT(1,N), DVTAB(1,N) and POLAT(1,N), ..., POLAT(NPA(N),N).
Figure II-12. Geometry of Fragment Vulnerability Data Options
continues until \( \text{AVTAB}(NPA(N) \times (NPB(N) - 1) + 1, N) \), ..., \( \text{AVTAB}(NPA(N) \times NPB(N), N) \) correspond to \( \text{POLBT}(NPB(N), N) \), \( \text{DVTAB}(1, N) \) and \( \text{POLAT}(1, N) \), ..., \( \text{POLAT}(NPA(N), N) \). Then, starting with \( \text{AVTAB}(NPA(N) \times (NPB(N) + 1), N) \) all of the above is repeated for \( \text{DVTAB}(2, N) \), \( \text{DVTAB}(3, N) \) and eventually for \( \text{DVTAB}(NDV(N), N) \). The last value of \( \text{AVTAB}(1, N) \) is \( \text{AVTAB}(NPA(N) \times NPB(N), N) \).

In case the above explanation is not clear the following example is given.

Suppose:

\[ \text{NPA}(N) = 2 \]
\[ \text{NPB}(N) = 3 \]
\[ \text{NDV}(N) = 4. \]

Then the table of \( \text{AVTAB}(1, N) \) would be set up as follows:

<table>
<thead>
<tr>
<th>POLBT(1,N)</th>
<th>AVTAB(1,N)</th>
<th>AVTAB(2,N)</th>
<th>AVTAB(3,N)</th>
<th>AVTAB(4,N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>POLBT(2,N)</td>
<td>AVTAB(5,N)</td>
<td>AVTAB(6,N)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>POLBT(3,N)</td>
<td>AVTAB(7,N)</td>
<td>AVTAB(8,N)</td>
<td>AVTAB(9,N)</td>
<td>AVTAB(10,N)</td>
</tr>
<tr>
<td></td>
<td>AVTAB(11,N)</td>
<td>AVTAB(12,N)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>POLBT(4,N)</td>
<td>AVTAB(13,N)</td>
<td>AVTAB(14,N)</td>
<td>AVTAB(15,N)</td>
<td>AVTAB(16,N)</td>
</tr>
<tr>
<td></td>
<td>AVTAB(17,N)</td>
<td>AVTAB(18,N)</td>
<td>AVTAB(19,N)</td>
<td>AVTAB(20,N)</td>
</tr>
<tr>
<td></td>
<td>AVTAB(21,N)</td>
<td>AVTAB(22,N)</td>
<td>AVTAB(23,N)</td>
<td>AVTAB(24,N)</td>
</tr>
</tbody>
</table>

The way polar angles A and B are measured depends on the value of the input variable \( \text{IAVA0P} \) (Figure 11-12).

(a) When \( \text{IAVA0P} = 0 \)

\( \text{POLARA} = \text{ATANQ}(\text{DELX}, \text{DELY}) \times \text{DEG} \) where \( \text{ATANQ} \) chooses the angle between 0 and 2\( \pi \).
\( \text{POLARB} = \text{ASIN}(\text{DELZ}/\text{DVIMP}) \times \text{DEG} \) and \( \text{ASIN} \) is normal principal value of arcsine function.

Therefore, input data should be such that

- \( \text{POLAT}(1, N) = 0. \)
- \( \text{POLAT}(NPA(N), N) = 360. \)
- \( \text{POLBT}(1, N) = -90. \)
- \( \text{POLBT}(NPB(N), N) = 90. \)

(b) When \( \text{IAVA0P} \neq 0 \)

\( \text{POLARA} = \text{ACOS}(\text{DELY}/\text{DVIMP}) \times \text{DEG} \) and
\( 0 < \text{POLARA} < 180. \)
\( \text{POLARB} = \text{ATANQ}(\text{DELZ}, \text{DELX}) \times \text{DEG} \) and
\( 0 < \text{POLARB} < 360. \)

Input data should be such that

- \( \text{POLAT}(1, N) = 0. \)
- \( \text{POLAT}(NPA(N), N) = 180. \)
- \( \text{POLBT}(1, N) = 0. \)
- \( \text{POLBT}(NPB(N), N) = 360. \)

Fragmentation characteristics of submunitions are evaluated in a manner identical to that defined for unitary fragmentation munitions.
11. CALCULATION OF PROBABILITY THAT TARGET IS KILLED BY FRAGMENTS IN A PARTICULAR ENcounter

Paragraph 10 discussed the computation of probability of killing a particular vulnerable component by a particular hit by fragments. Now, a particular component can be hit by as many as six different groups of fragments. Suppose it is hit by NAHITS groups. Let $P_i$ = probability that component was killed by hit $i$. Then total probability that component was killed by the NAHITS hits is:

$$1 - \prod_{i=1}^{NAHITS} (1 - P_i)$$

We now wish to compute the total probability that the target was killed by fragments. Since there are usually more than one vulnerable component, there must be some means for specifying the components, or the minimum number of vulnerable components which must be killed in order to say that the target is killed. The following input variables are used for this purpose:

- $NGROUP$ = number of groups of vulnerable components. The target is killed by fragments if any group is killed.
- $NCOM(I)$, $(I = 1, 2, ..., NGROUP)$; $NCOM(I)$ is the number of vulnerable components in the $I$th group.
- $NPVCOM(I,J)$, $(J = 1, 2, ..., NGROUP)$, $(I = 1, 2, ..., NCOM(J))$; $NPVCOM(I,J)$ is the serial number of the $I$th vulnerable component in group $J$. For example, if $NPVCOM(2,5)$ = 4, this means that the second vulnerable component in the 5th group is vulnerable component number 4.
- $KILL(I)$, $I = 1, 2, ..., NGROUP$ = the minimum number of vulnerable points in the $I$th group which must be killed in order to kill the target.

For completeness and logical consistency, this input data must satisfy certain conditions:

1. Each vulnerable component should be a member of one and only one group:

$$\sum_{I=1}^{NGROUP} NCOM(I) = NPV$$

and for any $I = 1, 2, ..., NPV$, it should be true that $I = NPVCOM(J,K)$ for one $K$ and one $J$.

2. $KILL(K) \leq NCOM(K)$ for all $K$. Let $PKF(I)$ = the probability that the $I$th component is killed. Consider the $J$th group of components. Let:

$$P(I) = PKF(NPVCOM(I,J)),$$

$$Q(I) = 1 - P(I),$$

$(I = 1, 2, ..., NCOM(J)).$

Consider the polynomial in $x$:

$$NCOM(J) \prod_{I=1}^{NCOM(J)} (P(I) \times Q(I)) = \sum_{i=0}^{NCOM(J)} a_i x^i.$$

We state, without proof, that $a_i$ $(i = 1, 2, ..., NCOM(J))$ is the probability that exactly $i$ components in the $J$th group are killed. Thus, the probability $PQ(J)$ that at least $KILL(J)$ components are killed is:

$$PQ(J) = \sum_{i=KILL(J)}^{NCOM(J)} a_i.$$

There are $NGROUP$ groups. The probability $PKFG$ that the target is killed by fragments is:

$$PKFG = 1 - \prod_{J=1}^{NGROUP} (1 - PQ(J)).$$
12. CALCULATION OF PROBABILITY THAT TARGET IS KILLED BY CONTINUOUS ROD*

The continuous rod warhead destroys by cutting the load-carrying skin structure or spars of the fuselage, empennage, or wing. The action of this warhead can be described as the expansion of a hinged steel ring caused by the detonation of a high explosive. The severance of a sufficient number of spars causes catastrophic structural failure. The contribution of each spar cut toward a kill is added, and the total of these contributions called Hit, is then checked against the total damage requirement for catastrophic kill of the target (Crash). The spars are examined in order until Hit exceeds Crash or until the kill contribution from all spars has been evaluated.

The vector analysis and formulations of subroutine CNTROD are as follows:

Let:
\[ A \] = Missile velocity vector
\[ B \] = Miss distance vector
\[ C \] = Vector perpendicular to plane of \( A \) and \( B \).
\[ D \] = Vector perpendicular to plane of \( A \) and \( C \).
\[ \varphi \] = Angle of attack (function of pitch and yaw).

The missile velocity vector \( A \), the missile centerline vector, the origin of the coordinate system, and the miss distance vector \( B \) lie in the same plane. The missile centerline vector is rotated through an angle \( \varphi \) from the missile velocity vector toward the origin. \( F_1, G_1, H_1 \), are the direction cosines of the missile velocity vector \( VM \), and \( X_M, Y_M, Z_M \) are the direction cosines of the miss distance vector \( B \) in the unprimed coordinate system:

\[
\begin{align*}
F_1 &= \cos \Gamma \cdot \sin \Psi \cdot \sin \Theta \\
G_1 &= \cos \Gamma \cdot \cos \Psi \cdot \sin \Theta \\
H_1 &= \sin \Gamma \\
X_M &= \cos \Psi \cdot \sin \Lambda \cdot X_M \\
Y_M &= \sin \Psi \cdot \cos \Lambda \cdot X_M + \cos \Xi \cdot Z_M \\
Z_M &= -\sin \Psi \cdot \sin \Lambda \cdot X_M + \sin \Theta \cdot Z_M \\
B &= X_M \hat{i} + Y_M \hat{j} + Z_M \hat{k} \\
C &= A \times B \\
D &= A \times C \\
MCL &= \text{Missile centerline vector} = A \cos \varphi - B \sin \varphi \\
MCL &= F_1 \hat{i} + G_1 \hat{j} + H_1 \hat{k} \\
C &= \begin{vmatrix} i & j & k \\ F_1 & E_1 & H_1 \\ X_M & Y_M & Z_M \end{vmatrix} \\
C_1 &= G_1 \cdot Z_M - H_1 \cdot X_M \\
C_2 &= H_1 \cdot X_M - F_1 \cdot Z_M \\
C_3 &= F_1 \cdot Y_M - G_1 \cdot X_M \\
C &= C_1 i + C_2 j + C_3 k \\
D &= \begin{vmatrix} i & j & k \\ F_1 & G_1 & H_1 \\ C_1 & C_2 & C_3 \end{vmatrix}
\]

*The Basic Methodology for the Continuous Rod Portion of this program was extracted from a program developed by Northrup Nortronics for the Air Force under contract AF 08(635)3549.*
\[
U = G_1 \cdot C_3 - H_1 \cdot C_2 \\
V = H_1 \cdot C_1 - F_1 \cdot C_3 \\
W = F_1 \cdot C_2 - G_1 \cdot C_1 \\
\vec{D} = U\hat{i} + V\hat{j} + W\hat{k} \\
F = F_1 \cdot \cos \theta - U \cdot \sin \theta \\
G = G_1 \cdot \cos \theta - V \cdot \sin \theta \\
H = H_1 \cdot \cos \theta - W \cdot \sin \theta .
\]

The direction cosines \(F, G,\) and \(H\) must now be normalized:

\[
QD = \sqrt{F^2 + G^2 + H^2} \\
F = F / QD \\
G = G / QD \\
H = H / QD
\]

The next step is to find the location of the missile at the instant of fuzing. With reference to Figure 11-13, vector analysis yields the fuzing equation:

\[
P_H \cdot P_F - P_{N_E} = |P_H| P_F \cdot |P_{N_E}| \cos \frac{\gamma}{2}
\]

where:

\(P_G\) is the missile cg,
\(P_H\) is the fuse cone apex,
\(P_W\) is the warhead cg location at fuzing,
\(P_F\) is a fuzing or glitter point on the target,
\(P_M\) is the miss distance vector location,
\(P_{N_E}\) is a unit vector along the missile centerline,
\(F, G, H\) are the direction cosines of the missile centerline,
\(T\) is the maximum fuzing time.

Since \(P_G\) and \(P_M\) lie on \(V_R\):

\[
X_G = X_M - V_X \cdot T \\
Y_G = Y_M - V_Y \cdot T \\
Z_G = Z_M - V_Z \cdot T
\]

\(P_G, P_W, P_H\) all lie on the missile centerline, therefore:

\[
X_H = X_G + F \cdot DF \\
Y_H = Y_G + G \cdot DF \\
Z_H = Z_G + H \cdot DF \\
X_{W'} = X_G + F \cdot DW \\
Y_{W'} = Y_G + G \cdot DW \\
Z_{W'} = Z_G + H \cdot DW
\]

*Denotes multiplication.

In the basic coordinate system, the fuzing equation can be written as:

\[
(X_F - X_H) \cdot F + (Y_F - Y_H) \cdot G + (Z_F - Z_H) \cdot H = \cos \frac{\gamma}{2} \left[ (X_F - X_H)^2 + (Y_F - Y_H)^2 + (Z_F - Z_H)^2 \right]^{\frac{1}{2}}.
\]
Figure 11-13. Vector Diagram for Continuous Rod Routine

There are now 10 equations and 10 unknowns; i.e., the coordinates of \( P_W, P_H, P_G \), and values of \( T \). Solving for \( T \):

\[
T = \frac{A_B - (P_V X + Q_V Y + R_V Z) \cos^2 (\gamma/2)}{A^2 - V_R^2 \cos^2 (\gamma/2)}
\]

*(See footnote)*

\[
\frac{[A_B - (P_V X + Q_V Y + R_V Z) \cos^2 (\gamma/2)]^2 - [A^2 - V_R^2 \cos^2 (\gamma/2)]}{A^2 - V_R^2 \cos^2 (\gamma/2)}
\]

\[
\left[ (B^2 - \cos^2 (\gamma/2) (P^2 + Q^2 + R^2)) \right]^{1/2}
\]

where

\[
\begin{align*}
P &= X_M - X_F + DF*F \\
Q &= Y_M - Y_F + DF*G \\
R &= Z_M - X_F + DF*H \\
A &= F^*V_X + Q^*V_Y + R^*V_Z \\
B &= F^*P + G^*Q + H^*R.
\end{align*}
\]

* \( \gamma/2 \geq 90^\circ \), use minus sign; \( \gamma/2 = 90^\circ \), both roots are equal; \( \gamma/2 < 90^\circ \), use plus sign.

102
The possible values of T after determining the value of \( \frac{Y}{2} \), and applying the footnote criterion, is a plus root, a minus root, or zero. If \( \frac{Y}{2} \) < 90° and the root is minus, no fuzing occurs. If \( \frac{Y}{2} \) > 90° and the root is plus, no fuzing occurs. This equation is computed for each target glitter point. The glitter point which yields the maximum root is chosen to be the correct fuzing point.

Having selected a tentative fuzing time (T), it is necessary to determine if the maximum fuzing range criterion is met. This is done with the following equation:

\[
RM - CSC \left( \frac{Y}{2} \right) \left\lbrack \left( P - V \frac{Y}{2} T \right)^2 + \left( Q - V \frac{Y}{2} T \right)^2 + \left( R - V \frac{Z}{2} T \right)^2 \right\rbrack^{\frac{1}{2}} > 1.0
\]

If the value of the left-hand side of this equation is less than 1.0, the glitter point which gave the next largest plus root (for \( \frac{Y}{2} \) < 90°) is used and checked. If the glitter point selected for \( \frac{Y}{2} > 90° \) causes the left-hand side of the equation to be less than 1.0, no fuzing occurs.

For \( \frac{Y}{2} < 90° \), if all glitter points will not satisfy the equation, it is necessary to go back to the previous equation and use the minus sign on the radical. If any additional positive roots occur, no fuzing occurs, as is the situation when positive roots occur and do not satisfy the equation.

The warhead cg location at fuzing corrected for distance between warhead cg and seeker is:

\[
\begin{align*}
X'_{W} &= X_{W} - V_{X} \cdot T \\
Y'_{W} &= Y_{W} - V_{Y} \cdot T \\
Z'_{W} &= Z_{W} - V_{Z} \cdot T
\end{align*}
\]

To correct for time delay, TD, the warhead location at detonation becomes:

\[
\begin{align*}
X_{W} &= X'_{W} + V_{X} \cdot TD \\
Y_{W} &= Y'_{W} + V_{Y} \cdot TD \\
Z_{W} &= Z'_{W} + V_{Z} \cdot TD
\end{align*}
\]

An iterative solution is used to determine if, when and where the continuous rod strikes a particular spar. The following basic assumptions are used in the solution.

1. The warhead is detonated at coordinates \((X_W, Y_W, Z_W)\) relative to the target (unprimed) coordinate system;
2. The warhead is traveling in the direction of missile velocity vector, with direction cosines \(F_1 = VMX/VM, G_1 = VMY/VM, H_1 = VMZ/VM\);
3. The target is aimed along the Y axis and traveling with a constant velocity, \(VT\);
4. The target is represented by a series of lines called spars, and each spar is defined by the following variables:
   - \(B_1\) = Direction cosine of the spar with respect to the X axis
   - \(B_2\) = Direction cosine of the spar with respect to the Y axis
   - \(B_3\) = Direction cosine of the spar with respect to the Z axis
   - \(B_4\) = The length of the spar in feet
   - \(B_5\) = The rod rejection criteria number
   - \(B_6\) = X coordinate of spar origin with respect to target cg
   - \(B_7\) = Y coordinate of spar origin with respect to target cg
   - \(B_8\) = Z coordinate of spar origin with respect to target cg
B9 = X direction cosine of the normal to the target surface at the spar
B10= Y direction cosine of the normal to the target surface at the spar
B11= Z direction cosine of the normal to the target surface at the spar
B12 thru B16 = Spar contribution toward a kill if hit by rod diameter (SPARDM)

VW = rod flyoff velocity at detonation relative to missile;
V00 = rod initial velocity relative to atmosphere:

\[ \text{V} = \sqrt{\text{VW}^2 + \text{V}^2} \]

7 The rod flies off at an angle \( \alpha \) to the missile velocity:
\[ \sin \alpha = \frac{\text{VW}}{\text{V00}}, \cos \alpha = \frac{\text{V0}}{\text{V00}}, \tan \alpha = \frac{\text{Vgl}}{\text{VM}}; \]

The velocity of the rod at any time is:
\[ \text{V} = \frac{\text{VW}/\alpha^\text{S}}{\text{S} - \frac{1}{3} \log (1 + \alpha \text{V00}/\text{T})} \]

where \( \text{S} \) - slant distance on surface of cone generalized by expanding rod.
\( \alpha \) - ballistic slowdown factor
\( \alpha \) = density of air
\( \text{S} \) = 6.6 density of the rod
\( \text{V} \) = rod thickness, inches

9 The distance from the spar origin (B6, B7, B8) to the strike point, measured along the spar, is S1, feet;

10 The coordinates of the strike point are (Xl, Yl, Zl) relative to the target cg at the time of detonation.

Derivation is accomplished by computing the strike point from the spar origin (T held constant):

\[
\begin{pmatrix}
X1 \\
Y1 \\
Z1
\end{pmatrix} = \begin{pmatrix}
X6 \\
Y6 \\
Z6
\end{pmatrix} + T \begin{pmatrix}
0 \\
V0 \\
0
\end{pmatrix}.
\]

The distance from the warhead location to the strike point is:

\[ S = \sqrt{(X1 - XW)^2 + (Y1 - YM)^2 + (Z1 - ZW)^2}. \]

The direction cosines of the rod path:

\[ \frac{X1 - XW}{S}, \frac{Y1 - YM}{S}, \frac{Z1 - ZW}{S}. \]

The cosine of the angle between the missile velocity and the strike vector, \( S \), is \( \cos \alpha \), and is equal to the dot product of these two vectors:

\[ \cos \alpha = F1 \cdot \left( \frac{X1 - XW}{S} \right) + G1 \cdot \left( \frac{Y1 - YM}{S} \right) + H1 \cdot \left( \frac{Z1 - ZW}{S} \right). \]

An iterative solution is obtained by substituting for X1, Y1, and Z1 in the above equation and solving for S1. The distance, S, is then computed, and finally a new value of T obtained from the following equation:

\[ T = \frac{\alpha S}{\text{V00}}. \]

The value of T just computed is compared with the starting T, and a value of delta T computed:
\[\Delta T = T - T_0\]
\[\Delta S = (VT) \left| \Delta T \right|\].

The iteration is assumed to have converged when \(\Delta S\), the distance moved by the target between successive iterations, is less than 1 foot. When \(\Delta S\) is greater than 1 foot, the process is started over using the new value of \(T\). (For the purpose of starting the iteration, \(T\) is taken to be zero initially.)

Once the iteration is complete, the value of \(S_1\) is compared with the length of the spar. If \(0.0 < S_1 < B4\), a definite strike on this particular spar has been obtained.

The angle between the normal to the surface at the strike point and the incoming strike vector is now obtained by the equation:

\[
\sin \theta = \frac{X_1 - X_M}{S} + \frac{Y_1 - Y_M}{S} + \frac{Z_1 - Z_M}{S}.
\]

The cosine of the angle between the strike vector and the surface is now:

\[
\cos \theta = \sqrt{1 - \sin^2 \theta}.
\]

Having computed the cosine of the angle between the incoming vector and the plane tangent to the surface at the strike point, \(\cos \theta\), a value of velocity necessary for the penetration at this strike angle is now obtained from the file of rejection criteria, the number of which is stored in location \(B5\). Up to 20 sets of these rejection criteria may be input. Each set consists of a minimum velocity and 2 empirical constants to define rod penetration of spars as a function of striking velocity and striking angle according to the following equation for reject velocity:

\[
V_{REJECT} = REJECT(2, J) + \frac{REJECT(1, J)}{\cos \theta} - \sin \theta.
\]

where the index \(J\) refers to the \(J\)th set of rejection constants and:

- \(REJECT(2)\) = Minimum velocity to cut the spar at any striking angle
- \(\cos \theta\) = Cosine of the strike angle
- and \(REJECT(1)\) and \(3\) are empirical constants.

The impact velocity as computed from the slowdown equation is

\[
V_{IMPAC} = \frac{V_0}{e^{\alpha S}}.
\]

If \(V_{IMPAC}\) exceeds \(V_{REJECT}\), then the strike vector is checked as in the fragmentation subroutine to see if it passed through a shielding ellipsoid before reaching the strike point. If no shielding occurred then the contribution of this spar cut toward a kill is determined. This contribution is determined from spar input values \(B12\) through \(B16\) as a function of spar diameter \(SPARDM\). The contribution is then added to the total contribution from all strikes from this particular warhead detonation. The total of these contributions, called Hit, is then checked against the total damage requirement for a kill of this particular target, called Crash. If Hit does not exceed Crash, then the next spar is examined for a possible strike, and the process repeated until a kill is obtained, or all parts of the airplane have been examined.

If the distance \((S)\) from the warhead location to the strike point is greater than \(RMAX\) (the maximum rod radius), then the rod will break up before impacting the target. The number of breaks in the rod at \(RMAX\) is called \(BREAK2\). \(BREAK1\) may also be defined as the initial number of breaks in the rod. \(RMAX\), \(BREAK1\), and \(BREAK2\) are input values.

In the event of rod breakup, the probability of a spar being cut if it is in the plane of the expanding rod is less than 1.0 even though the velocity and striking angle are within limits. Thus the probability of cutting the spar is defined by an empirical equation as follows
\[
    P_c = \frac{R_{\text{MAX}}}{S} \left[ \frac{-P \cdot RS}{e^{R_{\text{MAX}}}} + \frac{\text{BREAK} \cdot P \cdot RS}{e^{R_{\text{MAX}} \cdot \text{BREAK}}} - \frac{1}{e^{\text{BREAK} \cdot \text{BREAK}}} - 1.0 \right]
\]

where

\( P \) = Spar contribution toward a kill if hit by the rod

\( RS \) = Radius of individual spar.

The first term \( \frac{R_{\text{MAX}}}{S} \) reflects the ratio of rod radius to impact radius. The remaining values are corrections for overlap requirements and reduced kill capability for a broken rod.

13. SUMMARY OF ALL ENCOUNTERS

For each encounter the target is subject to the effects of blast, direct hit, and fragments. For a particular encounter we set:

\[ PKBL = 0, \text{ if target was not killed by blast} \]
\[ PKBL = 1, \text{ if target was killed by blast} \]
\[ PKDH = 0, \text{ if target did not receive a direct hit} \]
\[ PKDH = 1, \text{ if target did receive a direct hit.} \]

\( PKFG \) is computed as in Section 13.

All computations pertaining to each phenomenon are made regardless of the others, i.e., an encounter could result in \( PKBL = 1, PKDH = 1 \) and \( PKFG = 0.92 \).

Let \( PKBL_i, PKDH_i, PKFG_i \) be the results of the \( i \)th encounter. Let \( P_{Q_i}(J), (J = 1, 2, \ldots, \text{NGROUP}) \) be the probability of killing the \( J \)th group by fragments on the \( i \)th encounter. (The \( P_{Q_i}(J)'s \) are discussed in Section 13). Then the average probability of single shot kill (for the \( N_{\text{IT}} \) encounters) is computed as:

\[
    PKSS = \frac{\sum_{i=1}^{N_{\text{IT}}} \max (PKBL_i, PKDH_i, PKFG_i, PKROD)}{N_{\text{IT}}}.
\]

The average probability of kill is

for blast:

\[
    SPKBL = \frac{\sum_{i=1}^{N_{\text{IT}}} PKBL_i}{N_{\text{IT}}}.
\]

for direct hit:

\[
    SPKDH = \frac{\sum_{i=1}^{N_{\text{IT}}} PKDH_i}{N_{\text{IT}}}.
\]

for fragments:

\[
    SPKFG = \frac{\sum_{i=1}^{N_{\text{IT}}} PKFG_i}{N_{\text{IT}}}.
\]
for continuous rods:

\[ SPKROD = \frac{\sum_{i=1}^{NIT} PKROD_i}{NIT}, \]

We also compute the average probability of killing the Jth group with fragments:

\[ \frac{\sum_{i=1}^{NIT} PQ_i(J)}{NIT}. \]
This appendix contains the general flow diagram for the Air-to-Air Weapon Optimization Program M3054.
Main Program

Read in Optimization or Parameterization Data

Call CLOBIN Read in W/R, Intercept, Fuzing, & Target Data

Begin Main Program Loop

Compute Approach Angles

Form Rotation Matrix in Double Prime System

Compute Aim Point Centroid

Unprimed (Target) Coordinate System

Computed Radar Centroid of Glitter Points

Primed (Relative) Coordinate System

Figure III-1. General Flow Diagram for Air-to-Air Weapon Optimization Program
Figure III-1. General Flow Diagram for Air-to-Air Weapon Optimization Program (Continued)
Figure III-1. General Flow Diagram for Air-to-Air Weapon Optimization Program (Continued)
Figure III-1. General Flow Diagram for Air-to-Air Weapon Optimization Program (Concluded)
APPENDIX IV

FORMULATION OF LINEAR SHAPED CHARGE
MUNITION FOR EFFECTIVENESS EVALUATION

In defining linear shaped charge warheads for effectiveness computations, it is necessary to evaluate three kill mechanisms: blast, jet-slug, and fragmentation effects. The current formulation of these parameters is discussed in this appendix.

1. BLAST EFFECTS

Blast from linear shaped charge warheads is capable of inflicting "K" and "A" kills on airborne targets by severe structural damage. The airborne target structure will be represented by a series of ellipsoids with specified axis lengths. For example, an aircraft might have an ellipsoid to represent the physical dimensions for each of the fuselage, wings, vertical and horizontal stabilizers.

The warhead is assumed to project two blast waves of varying geometry as follows:

1. An ellipsoid, with center at the warhead centroid and specified axis lengths, which, in turn, defines a volume in which the target is assumed killed if any of its structures enter; this volume is defined as a function of time from detonation thereby simulating the finite propagation velocity of a blast wave;

2. A series of "spikes" simulating each linear shaped charge jet and projecting beyond the ellipsoid described in 1 above; the target is assumed killed if any of its structures intersect one of these jets at a time compatible with the blast propagation velocity; the jets are defined by a lethal distance, two angles measured from the forward and aft ends of the warheads, and an angular increment measured around the warhead centerline. (Warhead length is considered rather than assuming a point source for the kill mechanism.)

The combination of 1 and 2 above enables the accurate simulation of both conventional blast kill mechanisms and "spikes" or jets characteristic of linear shaped charge warheads. This combination of effects is discussed below.

2. BLAST ENHANCEMENT FOR LINEAR SHAPED CHARGE WARHEADS

Blast kill for conventional warheads is presently modeled by specifying a number of points on the target called "blast centers." Associated with each blast center are one or more "blast ellipsoids". The target is assumed to be killed by blast if the warhead detonation point is within a blast ellipsoid associated with a blast center at the time the blast wave (due to the warhead detonation) reaches the blast center. An iteration scheme is used to determine the time that the blast wave strikes a blast center and to compute the relative geometry of the blast ellipsoid and the projection detonation point. A tabular input of time T required for the blast wave to travel a distance X is used in the iteration scheme to determine if the blast wave reaches the blast center. Linear interpolation is used to compute the elapsed time involved, which can be related to blast overpressure or impulse in defining the blast ellipsoids for a given warhead with an equivalent bare charge weight of explosive.

113
A linear shaped charge warhead will have blast kill potential equivalent to a conventional warhead except in the jet-slug zones, where the blast kill will be enhanced by increased overpressure and consequently higher blast kill capability. To model the shaped charge blast kill, the blast ellipsoids will be defined and checked initially for conventional blast kill. If the target is not killed by conventional blast, a second set of larger blast ellipsoids will be checked to determine if the target is within lethal range of the enhanced blast. The final check will be to determine if the blast center is contained within the warhead jet-slug zones.

3. JET/SLUG EFFECTS

A linear shaped charge warhead projects slugs or rods within the jet-slug zone oriented parallel to the warhead centerline. The capability of representing up to three of these slugs for each jet will be provided. The slugs will be described by mass, cross section, ballistic data, initial velocity and flyoff direction. The slugs will be projected in the angular segments previously described for representing the jets in Section 1 above.

Target vulnerability to slug impact will be defined by a series of ellipsoids representing structural components. Encounter geometry, slug velocity, drag, and direction will be used to establish the event of a slug-target intercept with the target ellipsoids. Each ellipsoid will have a conditional kill probability assigned for slug impacts. The probabilities will be combined statistically, assuming independent events, for more than one impact.

4. FRAGMENTATION EFFECTS

In addition to slugs, fragments with a spectrum of masses, initial velocities, and drag characteristics are projected from each linear shaped charge liner. The capability for representing up to 10 classes of fragments will be provided. The fragment spray will be limited by two angles measured from the forward and aft ends of the warhead, and an angular increment about the warhead centerline.

Target vulnerability will be represented by a vulnerable area centered about one or more points. If a vulnerable point is established to be in one of the fragment sprays, then kill probability will be computed by

\[ P = 1 - \exp \left( - \sum_{i=1}^{N} \frac{C_i}{AV_i} \right) \]

where:
- \( C_i \) is impacting fragment density of the ith class
- \( AV_i \) is target vulnerable area of the ith class
- \( N \) is the number of fragment classes striking the target.

The computer program formulation will include the capability to input tabular vulnerable area as a function of fragment mass, velocity, and three types of angular data:

1. Box type data in which six sets of vulnerable areas are input for each side and end of the target;
2. BRL type data in which vulnerable areas are input for a series of elevation and azimuth impact angles;
3. Single point type data in which vulnerable areas are input as a function of polar impact angle.

In the methodology being evolved, the basic geometry for a specific weapon-target encounter, including aim point, guidance error, fusing point, and detonation point, is established identically for linear shaped charge warheads as for conventional fragmentation munitions. However, techniques for computing fragmentation kill probability for conventional warheads, assuming a point source of fragments, are not adequate for evaluating linear shaped charge warheads because of the narrow polar beam spray zone (on the order of 5 degrees) which includes all fragments projected by the warhead. For example, the width of a 5 degree beam spray at a distance of 50 feet from the warhead is < 5 feet.
thus a warhead length of 1 foot would constitute an error of 20 percent in the actual beam spray width if a point source of fragments was assumed. The following paragraphs present techniques for correcting beam spray zone width to account for warhead length.

The fragmentation subroutine for the current air-optimization computer program starts to compute fragment kill with a target description and a warhead detonation point \((X_D, Y_D, Z_D)\). The point of fragment-target intercept, \(X_s (i, j), Y_s (i, j), Z_s (i, j)\), is found by an iterative process which determines the points where fragments projected from the warhead with velocity \(i\) hit the target vulnerable component \(j\). For an intercept, the polar angle \(\theta_p\) from which the fragments of the warhead were projected to strike target component is computed. This angle is then compared to the forward and aft beam spray angles \(\theta_1\) and \(\theta_2\) to determine if the solution is valid \((\theta_1 < \theta_p < \theta_2)\). For a linear shaped charge, the values of \(\theta_1\) and \(\theta_2\) must be modified to account for warhead length.

Figure IV-1(a) is a diagram of the current fragmentation model for a cylindrical warhead showing the warhead detonation point \((X_D, Y_D, Z_D)\), the points where the fragments strike the target vulnerable point \((X_s, Y_s, Z_s)\), and the forward and aft polar angles \(\theta_1\) and \(\theta_2\). With this model it is obvious that the vulnerable point is outside the mathematically defined beam spray zone \((\theta_p < \theta_1)\) and the solution would be rejected. Figure IV-1(b) shows how the target vulnerable point can be included in the beam spray and establishes the geometry for formulating equations to correctly model the fragment-target intercept.

The formulation is as follows:

The distance of fragment flight \(D_F\) is given by

\[
D_F = \sqrt{(X_s - X_D)^2 + (Y_s - Y_D)^2 + (Z_s - Z_D)^2}.
\]

The perpendicular distance from the warhead centerline to the target vulnerable point \(L_p\) is given by

\[
L_p = D_F \cdot \sin \theta_p.
\]

The width of the forward beam spray at the perpendicular distance \(L_p\) is defined by

\[
S_1 = \frac{L_p}{\tan \theta_1}.
\]

The width of the aft beam spray at the perpendicular distance \(L_p\) is defined by

\[
S_2 = \frac{L_p}{\tan \theta_2}.
\]

Adding half the warhead length, \(W_L\), to each value gives

\[
S_1' = \frac{L_p}{\tan \theta_1} + \frac{W_L}{2}, \quad S_2' = \frac{L_p}{\tan \theta_2} + \frac{W_L}{2}.
\]

The corrected angles \(\theta_1'\) and \(\theta_2'\) are then given by

\[
\theta_1' = \tan^{-1}\frac{S_1'}{L_p}, \quad \text{and} \quad \theta_2' = \tan^{-1}\frac{S_2'}{L_p}.
\]

If \(\theta_p\) is between the corrected values \(\theta_1'\) and \(\theta_2'\) \((\theta_1' < \theta_p < \theta_2')\) the vulnerable point will be in the fragment beam spray.
Figure IV-1 (a). Conventional

Figure IV-1 (b). Corrected

Figure IV-1. Fragment Spray Representation
The density of fragments which strike the target must also be established to compute kill probability. Hence, techniques employed for point-source fragmentation cannot be directly applied for evaluating a linear shaped charge warhead. The equation which determines the dynamic striking fragment density for conventional fragmentation warheads can be employed with two changes:

1. Use the modified polar angles, \( \phi_1' \) and \( \phi_2' \), to account for warhead length;
2. Use a factor of \( 2\pi/\Delta \gamma \), where \( \Delta \gamma \) is the roll angular increment containing the fragments projected from a liner, to reflect the fact that a linear shaped charge warhead is not isotropic.

The required equation for establishing the dynamic impacting fragment density for linear shaped charge warheads becomes:

\[
\text{RHOD} = \frac{\text{FNP} \cdot D}{2\pi (\text{RC}^2 \cos \phi_1' - \cos \phi_2')} \times \frac{\text{RC} \cdot \text{VNBAR} \cdot \Delta \gamma}{\text{PMAG} \cdot \text{VFBAR}},
\]

where:

- \( \text{FNP} \) is the number of fragments projected from a linear shaped charge liner
- \( \phi_1' \) and \( \phi_2' \) are the modified fragment polar angles
- \( \text{VNBAR} \) is the warhead velocity
- \( \text{VFBAR} \) is the fragment initial static velocity
- \( \text{VNBAR} \) is the fragment initial dynamic velocity
- \( \text{PMAG} \) is the distance traveled by the fragments
- \( \text{RC} \) is the radius of curvature of the sphere containing the fragments at target impact.

\( \phi, \phi, \psi, \text{DIST} \) are presented in Figure IV-2(a), 2(b), and 2(c).

\( \Delta \gamma = \text{seconds} (\ddot{\gamma}) \).

Kill probabilities due to fragmentation are determined as currently computed in the Air-to-Air Weapon Optimization Computer Program, accounting for drag, impacting density, mass, velocity, and angle, and the dynamic missile velocity. However, since a continuous distribution of fragments are not projected about the roll axis of the missile as for a conventional fragmentation warhead, the roll angle formed by the missile centerline and a line from the warhead to the target vulnerable point, \( \gamma \), discussed above, must be established. This angle is in turn compared with roll angles limiting the fragmentation from the jets to determine whether the fragments actually encounter the target. The derivation of equations for establishing this angle is presented below.

The relative positions of fragmentation spikes of the warhead can be most easily expressed in terms of a coordinate system defined in the following manner. In Figure IV-1 the vector \( \mathbf{L} \) represents the longitudinal axis of the missile, plane number 1 is a plane perpendicular to \( \mathbf{L} \), and plane number 2 is a vertical plane passing through \( \mathbf{L} \). Let \( \mathbf{L} \) represent the \( X \)-axis of the new reference system, \( X'' \), and let the intersection of planes 1 and 2 represent the \( Z \)-axis of the new reference system, \( Z'' \). The new \( Y \)-axis is \( Y'' \), defined as being perpendicular to both \( X'' \) and \( Z'' \). For clarity, the axes \( Z'' \) and \( X'' \) can be thought of as lying in the plane of the paper and \( Y'' \) as going into the paper, perpendicular to both \( X'' \) and \( Z'' \).

Let \( \phi \) be the angle between plane number 2 and the \( X \)-axis and \( \psi \) the angle between the \( XY \) plane and \( \mathbf{L} \). The vector \( \mathbf{V}_{\phi} \) is the component of \( \mathbf{V}_{\phi} \) defining the flyoff direction of fragments which strike the target in plane number 1 and make an angle \( \alpha \) with \( Z'' \).
Figure IV-2 (a)

Figure IV-2 (b)

Figure IV-2 (c)

Figure IV-2. Computation of RHOD
Thus, a comparison must be made to determine whether the angle $\gamma$ lies between two angles in the number 1 plane defining the limitation of a linear shaped charge jet in the roll axis.

The angle $\gamma$ is determined in the following manner.

Let the $X$, $Y$, and $Z$ components of $V_{FS}$ be defined as $V_{FSX}$, $V_{FSY}$, and $V_{FSZ}$. From Figure IV-1, it is apparent that the $XY$ plane can be rotated about the $Z$-axis through an angle $\xi$ so that $Y$ is parallel to $Y''$. Then the $X$, $Y$, and $Z$ components of $V_{FS}$ in this "intermediate" coordinate system (denoted by a prime) are

$$V'_{FSX} = V_{FSX} \cos \xi - V_{FSY} \sin \xi \quad (1)$$

$$V'_{FSY} = V_{FSX} \sin \xi + V_{FSY} \cos \xi \quad (2)$$

$$V'_{FSZ} = V_{FSZ} \quad (3)$$

Now the $X'' Z''$ plane can be rotated about the $Y''$ axis through an angle $\eta$ so that $X''$ is parallel to $X'$ and $Z''$ is parallel to $Z'$. Then the $X, Y, Z$ components of $V_{FS}$ in the double primed system are

$$V''_{FSX} = V'_{FSX} \cos \eta - V'_{FSY} \sin \eta \quad (4)$$

$$V''_{FSY} = V'_{FSX} \sin \eta + V'_{FSY} \cos \eta \quad (5)$$

$$V''_{FSZ} = V'_{FSZ} \quad (6)$$

Substituting equations (1-3) into equations (4-6), gives

$$\dot{V}_{FS} = \dot{V}'_{FS} = (V_{FSX} \cos \xi \cos \eta - V_{FSY} \sin \xi \cos \eta - V_{FSZ} \sin \eta)$$

$$+ j'' (V_{FSX} \sin \xi + V_{FSY} \cos \xi)$$

$$+ k'' (V_{FSX} \cos \xi \sin \eta + V_{FSY} \sin \xi \sin \eta + V_{FSZ} \cos \eta) \quad (7)$$

which is the vector equation for $V_{FS}$ in the double primed reference system as defined previously. The angle $\gamma$ can be evaluated by

$$\tan = \frac{V'_{FSY}}{V'_{FSZ}} \quad (8)$$

$$\tan = \frac{V_{FSX} \sin \xi + V_{FSY} \cos \xi}{V_{FSX} \cos \xi \sin \eta - V_{FSY} \sin \xi \sin \eta + V_{FSZ} \cos \eta} \quad (9)$$

The vector equation of $L$ in terms of the unprimed coordinate system is defined by the missile velocity, pitch, and yaw. The angle $\epsilon$ is determined from

$$\tan \epsilon = -\frac{\dot{V}_y}{\dot{V}_x}$$

and the angle $\mu$ is determined from

$$\tan \mu = \frac{L_2}{(L_x^2 + L_y^2)^{1/2}}$$
where \( L_x, L_y, \) and \( L_z \) are the X, Y, and Z components of \( \hat{L} \) in the unprimed coordinate system.

This formulation is a significant portion of the capability required to establish fragmentation kill probability from linear shaped charge warheads.

E. TOTAL KILL PROBABILITY

The total kill probability resulting from blast \( (P_B) \), slugs \( (P_S) \), and fragmentation \( (P_p) \) will be formulated as follows:

\[
P = 1 - (1 - P_B)(1 - P_S)(1 - P_p)
\]  

(2)
APPENDIX V

GEOMETRY AND FLOW DIAGRAM

This appendix contains a descriptive sample of warhead geometry, a descriptive sample of impact pattern geometry, and a general flow diagram for the Flechette Lethal Area Program.
Figure V.2. Sample Impact Pattern Geometry
CALL INPUT
READ Input Data

Define Flechette Ejection Grid for CLASS (I)

Do For All (I) where I = 1 to N Groups

Loop Complete?

CALL TRAJ

Solve Trajectory Equations of Each Point in Grid
Determine Impact Pattern for CLASS (I)

Do for All (I) where I = 1 to N Groups

Loop Complete?

YES

NO

YES

Calculate Area of Each Polygon (N) in Impact Pattern (I)

Determine Average Flechette Velocity and Impact Angle for Flechettes in Polygon (N), CLASS (I)

Figure V-3. General Flow Diagram for Flechette Lethal Area Program
Calculate Kill Probability of Flechettes in Polygon (N), CLASS (I)

Loop Complete?

YES

Integrate Areas of No Overlap
\[ AL = \sum PK(I,N) \times AREA(I,N) \]

Are there Areas of Overlap?

NO

YES

CALL OVRLAP

Determine Areas of Overlap

Calculate Loss of Lethal Area Due to Overlap for Each Grid Point.
\[ AL = AL - \sum DAL \]
where DAL is the Lethal Area Lost due to Overlap at Grid Point (XG,YG)

Lethal Area = 2 * AL

If IPR1 = 1, CALL PLOT

EXIT

Figure V-3. (Concluded)
APPENDIX VI

RUNGE KUTTA SOLUTION OF EQUATIONS OF MOTION
AND TECHNIQUES FOR DETERMINING POLYGON PARAMETERS

1. RUNGE KUTTA SOLUTION OF EQUATIONS OF MOTION

The following formulation demonstrates the basic techniques used to solve the equations of motion describing the flechette flight path.

Equations of motion:

\[ \dot{x} = - \frac{C_D \rho d^2}{2 M \cdot 4} (V) (V_x) \]  \hspace{1cm} (VI-1)

\[ \dot{y} = - \frac{C_D \rho d^2}{2 M \cdot 4} (V) (V_y) \]  \hspace{1cm} (VI-2)

\[ \dot{z} = - \frac{C_D \rho d^2}{2 M \cdot 4} (V) (V_z) - g \]  \hspace{1cm} (VI-3)

where:

\( \frac{d^2}{4} \) is the presented area of the projectile; \( D \) is the aerodynamic reference number

\( C_D \) is drag coefficient

\( \rho \) is density of air

\( M \) is mass of projectile

\( V \) is projectile total velocity

\( V_x, V_y, V_z \) are velocity components of \( V \)

\( g \) is the acceleration of gravity.

The general formula of Runge and Kutta for a third order approximation solution is given by:

\[ \Delta y = f(x,y) = \frac{1}{4} (\Delta' y + \Delta'' y + \Delta''' y) \]  \hspace{1cm} (Reference 5)

where:

\[ \Delta' y = f(x,y) \Delta h \]

\[ \Delta'' y = f(x + \frac{1}{3} \Delta h, y + \frac{1}{3} \Delta' y) \Delta h \]

\[ \Delta''' y = f(x + \frac{2}{3} \Delta h, y + \frac{2}{3} \Delta'' y) \Delta h \]

The solution that follows demonstrates the techniques used in the program to solve equations (VI-1), (VI-2), (VI-3). Only the formulation involving equation (1) will be presented here as all three equations use similar methodology in their solutions.
From Equation (VI-4) let:

(a) \( \delta y = \Delta VX \) (for time increment \( \Delta t \))

(b) \( f(x, y) = -\frac{C_0 \, \rho \, n \, d^2}{2 \cdot \eta \cdot 4} \cdot (V) \cdot (VX) = (X1) \cdot (V) \cdot (VX) \)

where \( V \), \( VX \) are evaluated at time \( t \)

(c) \( h = DT \) = time increment, \( \Delta t \)

(d) \( \Delta'y = (X1) \cdot (V) \cdot (VX) \cdot (DT) = C40 \)

(e) \( \Delta''y = (X1) \cdot (V1) \cdot (VX + 1/3 \, C40) \cdot (DT) = C41 \)

(f) \( \Delta'''y = (X1) \cdot (V2) \cdot (VX + 2/3 \, C41) \cdot (DT) = C42 \)

where:

\( V1 \) and \( V2 \) are new total velocities calculated from the velocity components evaluated at the 1/3 and 2/3 positions of the formulation; i.e.:

\[
V1 = \sqrt{(VX + 1/3 \, C40)^2 + (VY + 1/3 \, C50)^2 + (VZ + 1/3 \, C60)^2}
\]

where:

\( C50 = (X1) \cdot (V) \cdot (VY) \cdot (DT) \)

\( C60 = (X1) \cdot (V) \cdot (VZ) \cdot (DT) - G^*DT \).

Substituting (a) thru (f) into equation (VI-4) we have:

\[
\begin{align*}
\text{(5)} & \quad \Delta VX = 1/4 \left[ C40 + 3 \, (C42) \right] \\
\text{(6)} & \quad VX = VX_0 + \Delta VX
\end{align*}
\]

where:

\( VX \) is the velocity component in the X direction evaluated at time \( t + \Delta t \)

\( VX_0 \) is the velocity component at time \( t \)

\( \Delta VX \) is the change in velocity in the X direction for the time increment \( \Delta t \).

This procedure is paralleled by similar procedures to generate the velocity components in the Y and Z directions. Thus, at some time, \( t \), \( VX \), \( VY \), and \( VX \) are defined in space. The iterative procedure is now repeated for another time increment \( \Delta t \), using the answers from the previous iteration as initial values. This procedure generates the trajectory of the projectile ultimately defining impact parameters.

2. TECHNIQUES FOR DETERMINING POLYGON PARAMETERS

The following mathematical techniques are used to calculate the area, average flechette velocity, and average flechette impact angle associated with each polygon formed by the impact pattern.

a. Calculation of Polygon Areas

The impact pattern is generated by a trajectory subroutine, whose output consists of impact coordinates, terminal velocities, and impact angles of flechettes released in ejection grid. The impact pattern geometry (Figure V-2, Appendix V) forms numerous adjacent polygons whose vertices are defined by the impact coordinates discussed above. As Figure V-2 demonstrates, there are two basic types of polygons created, triangles and quadrilaterals. In the process of determining the lethal area of the munition the area of each of these polygons must be evaluated. From the coordinates of the vertices of any triangle the area may be calculated by:

\[
\text{AREA} = \frac{1}{2} \left| (X1 \cdot Y2 + X2 \cdot Y3 + X3 \cdot Y1 - Y1 \cdot X2 - Y2 \cdot X3 - Y3 \cdot X1) \right|
\]

where:

\((X1, Y1), (X2, Y2), (X3, Y3)\) are the coordinates of the vertices respectively.
From this basic equation the areas of all triangular and quadrilateral areas of the impact pattern are calculated. Quadrilateral areas are determined by dividing the figure into two triangular areas and applying the above equation. The area given by Equation (VI-7) is positive or negative according as the vertices are numbered counterclockwise or clockwise; therefore, proper ordering of points is necessary unless the absolute value of the quantity is calculated.

b. Average Flechette Velocity in Polygon (N)

The average terminal velocity of flechettes in polygon (N) is calculated by numerically averaging the velocities at each vertex of the figure. For a triangular area this is approximated by merely adding the three velocities and dividing by three. For a quadrilateral figure the technique is altered somewhat to account for the weighting of a distance vertex. The average velocity is determined by:

\[ V_{AVE} = \frac{A(124) (V_1 + V_2 + V_4) + A(134) (V_1 + V_3 + V_4) + A(123) (V_1 + V_2 + V_3) + A(234) (V_2 + V_3 + V_4)}{3 \times \left[ A(124) + A(123) + A(234) + A(134) \right]} \]  

(VI-8)

where:

1, 2, 3, and 4 are numbers defining the four vertices of the quadrilateral

\( V_1 \) is the flechette impact velocity at vertex 1

\( V_2 \) is the flechette impact velocity at vertex 2

\( V_3 \) is the flechette impact velocity at vertex 3

\( V_4 \) is the flechette impact velocity at vertex 4

\( A(124) \) is the area of a triangle formed by vertices 1, 2 and 4

\( A(134) \) is the area of a triangle formed by vertices 1, 3 and 4.

A. Average Flechette Impact Angle in Polygon (N)

The techniques used in paragraph b above are applicable for calculating the average impact angle, BETAVE, in polygon (N). Equation (VI-8) is used to average the impact angles at each vertex of a quadrilateral area, by substituting the impact angles \( \beta_1, \beta_2, \beta_3, \beta_4 \), for the vertex velocities.
APPENDIX VII
GRID POINT TEST

In the process of calculating the lethal area of impact pattern areas where overlap occurs, a test is made on each grid point \((X_G, Y_G)\), to determine what polygons contain this point. This test is made by mathematically checking the location of the point \((X_G, Y_G)\) relative to each polygon in all classes. The technique used calculates the areas formed by two vertices of a given polygon and the grid point \((X_G, Y_G)\). The sum of the triangular areas formed by this point and each of the polygon sides (a side being defined by any two vertices) must equal the area of the polygon for the point to lie inside its boundaries (see Figure VII-1). Each grid point is checked against all polygons in all classes, thus determining what incremental areas, \(\Delta A\), associated with each grid point contain flechettes of more than one class.

\[
\begin{align*}
A1 &= \text{area of } \Delta(G,1,2) \\
A2 &= \text{area of } \Delta(G,2,3) \\
A3 &= \text{area of } \Delta(G,3,4) \\
A4 &= \text{area of } \Delta(G,4,1)
\end{align*}
\]

Figure VII-1. Grid Test Geometry

The test area is determined and is given by:

\[
\text{TAREA} = A1 + A2 + A3 + A4 \quad (VII-1)
\]

If \(\text{TAREA} = \text{AREA (area of polygon (N))}\) the point lies inside polygon (N) (Figure VII-1a)).

If \(\text{TAREA} > \text{AREA}\) the point is outside polygon (N) (Figure VII-1b)).

This technique is applicable to both triangular and quadrilateral polygons. Once the loop is completed for a given grid point the loss of lethal area due to overlap is determined by Equation (6) Section VI, paragraph 4. This equation uses the polygons in each class that grid point \((X_G, Y_G)\) is common to, and calculates the loss of lethal area due to overlap for this grid point.
APPENDIX VIII

UTILIZATION REPORT - FLECHETTE COMPUTER PROGRAM

1. PROGRAM DESCRIPTION

This program was assembled to compute the lethal area of warheads projecting flechettes, especially from high altitudes. The principle unique feature is that flechette trajectories are actually simulated by the equations of motion to properly account for gravity effects.

The warhead attack is defined by a height and angle of flechette release. Flechette characteristics may be described by up to three classes each containing a unique:

1. Mass;
2. Initial velocity;
3. Beam spray angle, distribution, and number of flechettes;
4. Reference diameter;
5. \( C_D \) versus Mach number data.

In addition, up to five sets of flechette reference diameters and/or \( C_D \) versus Mach number curves may be used as a function of real time of flight for each class of flechettes. This capability enables the pseudo-simulation of unstable flechette flight initially and becoming stable with time. Provisions are in the input format to allow the use of a smaller than usual value for the iterative time increment at the time of changing ballistic constants to maintain accuracy.

The warhead is assumed to release flechettes in a conical pattern with prescribed angular limits and uniform or normal distributions. (Beam spray half angles are limited to 30 degrees or less for warhead dive angles less than 30 degrees. In no case can the beam spray half angle exceed 90 degrees (this is a 3 sigma value for normal distribution).) The program divides this pattern into a number of angular increments from the warhead longitudinal axis and around the warhead axis. Data generated show 5 and 9 of these angular increments from and about the warhead axis, respectively, to be sufficient to attain four place accuracy for uniform distribution of flechettes; 10 and 18 of these angular increments are required to attain three place accuracy for normally distributed flechette densities.

These angular increments of the flechette pattern are then projected into the ground plane by trajectory simulations.

Munition lethal area is established by integrating kill probability against either standing, prone, or troops in foxholes over the impact pattern area. In areas of overlapping classes of flechettes, a special integration technique employing grid points representing increments of area is employed to properly combine kill probabilities contributed by each class.

Complete input data is output for checking and documentation purposes. In addition, flechette impact pattern data and sketches are output as well as munition lethal area. A list of output parameters and their definitions appears in Table VIII-1 of this Appendix.

2. TIMING

A typical run consisting of input data for three classes of flechettes, and utilizing angular increments of 5 and 9, (from and about the warhead axis respectively) takes less than 30 seconds to run on the CDC-6400 computer.
3. INPUT DATA FORMAT

All integers must be right adjusted in their field. Table VIII-II contains all input data pertinent to the warhead, encounter geometry and target, plus various print options available for output.

4. MULTIPLE RUNS

Runs may be stacked by placing a *RUN card after the last data card and then clobbering the previous input data per instructions found in Table VIII-III. Storage order of MACH elements are given in Table VIII-IV.
### TABLE VIII-1. OUTPUT PARAMETERS

1. I designates class number.

2. J designates THETA (beam spray half angle measured from the warhead axis) increment number (Figure V-2, Appendix V).

3. K designates PHI (radial angle measured about the warhead axis) increment number (Figure V-2, Appendix V). (J,K) defines a point formed by the intersection of these two increments.

4. $V_x$ is the X component of the terminal velocity vector of the flechette at point (J,K).

5. $V_y$ is the Y component of the flechette terminal velocity vector at (J,K).

6. $V_z$ is the Z component of the flechette terminal velocity vector at (J,K).

7. X and Y are the coordinates of the flechette at point (J,K) in the ground pattern. The point of release is (0,0,HOE).

8. V is the terminal velocity vector of flechette impacting at (J,K).

9. BETA is the terminal elevation angle of the flechette at point (J,K), measured positively downward off the horizontal.

10. N designates the triangle or quadrilateral identification number.

11. VAVE is the average flechette terminal velocity in polygon N.

12. BETAVE is the average BETA in polygon N.

13. $P_c$ is the conditional kill probability in polygon N.

14. $\rho$ is the flechette density (number of flechettes per square foot) in polygon N.

15. APTGT is the average target presented area in polygon N.

16. $P_K$ is the kill probability in polygon N given by $1 - \exp(-\rho \cdot APTGT \cdot P_c)$.

17. AREA is the area of polygon N.

18. DAL is the contribution to total munition lethal area of polygon N. $DAL = P_K (I,N) \times AREA (I,N)$.

19. AL is the cumulative lethal areas of polygons, reduced to account for overlap.

20. IX, IY indicate the dimensioning of the grid points in the overlap area. IX is the column number and IY is the row number measured in the positive directions of a right handed coordinate system. (From left to right and from bottom to top respectively.)

21. $X_G$ and $Y_G$ are the X and Y coordinates of a grid point (IX, IY).

22. SUM is the sum of the $P_K$ values of all polygons in which grid point (IX, IY) is common. $(P_{K1} + P_{K2} + \ldots + P_{Kn})$ where IX, IY lies in polygons 1, 2 ... n.

23. PROD is the product of one minus $P_K$ of all polygons in which grid point IX, IY is common. $(1 - P_{K1}) (1 - P_{K2}) \ldots (1 - P_{Kn})$ where IX, IY lies in polygons 1, 2 ... n.

24. LETHAL AREA = 2-AL is the munition lethal area.
<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1-72</td>
<td>ALPHN</td>
<td>ITITLE</td>
<td>-</td>
<td>General title of run.</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>INTEG</td>
<td>IPR1</td>
<td>0 or 1</td>
<td>1 indicates plot of ground pattern will be made for each flechette class.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>INTEG</td>
<td>IPR2</td>
<td>0 or 1</td>
<td>1 indicates impact coordinates, velocity, and angle for flechette impact points will be printed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>INTEG</td>
<td>IPR3</td>
<td>-</td>
<td>Not used.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>INTEG</td>
<td>IPR4</td>
<td>0 or 1</td>
<td>1 indicates average velocity, angle, conditional kill probability, flechette density, target presented area, kill probability, polygon area, and its contribution to total munition lethal area will be printed for each polygon.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>INTEG</td>
<td>IPR5</td>
<td>0 or 1</td>
<td>1 indicates (X_G) and (Y_G) coordinates of the overlap grid points, identification number for polygons in which (X_G) and (Y_G) are contained, class of flechettes impacting within each polygon, area of each polygon, and kill probability data will be printed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>INTEG</td>
<td>NGPS</td>
<td>1 (\leq) 3</td>
<td>Number of classes of flechettes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>DEC</td>
<td>ALPHA</td>
<td></td>
<td>Warhead terminal attack angle (measured positively downward from the horizontal to the nose of the missile).</td>
<td>Degrees</td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>DEC</td>
<td>HOB</td>
<td></td>
<td>Height of flechette release.</td>
<td>Feet</td>
</tr>
<tr>
<td></td>
<td>31-35</td>
<td>INTEG</td>
<td>NGRID*</td>
<td>0 (\leq) 99999</td>
<td>Number of grid points in area of overlap between flechette classes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>36-40</td>
<td>INTEG</td>
<td>MAXGR*</td>
<td>0 (\leq) 99999</td>
<td>Maximum number of grid points if GRAREA is used (use only when NGRID = 0).</td>
<td>Feet²</td>
</tr>
<tr>
<td></td>
<td>41-50</td>
<td>DEC</td>
<td>GRAREA*</td>
<td></td>
<td>Differential area about grid point (use only when NGRID = 0).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(REPEAT CARDS C-F NGPS TIMES)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>1-10</td>
<td>DEC</td>
<td>FLMAS(I)</td>
<td>-</td>
<td>Flechette mass for Class I</td>
<td>Grains</td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>DEC</td>
<td>VF(I)</td>
<td>-</td>
<td>Initial velocity of flechettes for Class I</td>
<td>Feet/Second</td>
</tr>
</tbody>
</table>

*For overlap calculations two separate methods of inputing the number of grid points have been provided. The first is to input NGRID directly which represents the total number of grid points in the flechette class overlap area. These grid points are distributed uniformly in this area. A second method is to set NGRID = 0, and input GRAREA, which is a differential area about each grid point. The program then determines the total number of grid points to be used and compares this value to MAXGR. The smaller of these two numbers is then used as NGRID for overlap computations. MAXGR is a cut-off value to provide an upper limit for NGRID.
<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>21-30</td>
<td>DEC</td>
<td>SIGMA(I)</td>
<td>-</td>
<td>The beam spray half angle measured off the longitudinal axis of the warhead for a uniform distribution of flechettes, or one third of the total beam spray half angle for a normal distribution of flechettes (Figure V-1, Appendix V).</td>
<td>Degrees</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 indicates flechette density is normally distributed about the warhead longitudinal axis. 2 indicates flechette density is uniformly distributed about the warhead longitudinal axis.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>35</td>
<td>INTEG</td>
<td>IDIST(I)</td>
<td>1 or 2</td>
<td>The number of increments in the beam spray angle (Figure V-1, Appendix V).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>39-40</td>
<td>INTEG</td>
<td>NTHETA(I)</td>
<td>1 ≤ 40</td>
<td>Number of radial increments (Figure V-1) measured around and perpendicular to the warhead longitudinal axis.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>44-45</td>
<td>INTEG</td>
<td>NPHI(I)</td>
<td>1 ≤ 36</td>
<td>IF INTD(I) = 1, the number of curves used to define the flechette drag coefficient versus Mach number function and reference diameter at various flight times for Class I will be input. If INTD(I) = 0, the number of reference diameters used to define ballistic characteristics at various flight times for flechettes of Class I will be input.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>INTEG</td>
<td>NTC(I)</td>
<td>1 ≤ 5</td>
<td>0 indicates flechette reference diameter (DM(I,L)) will be varied, but the same drag curve will be used; 1 indicates more than one drag curve will be used (if desired DM(I,L) can be also varied).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>55</td>
<td>INTEG</td>
<td>INTD(I)</td>
<td>0 or 1</td>
<td>Number of flechettes in Class(I).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>61-70</td>
<td>INTEG</td>
<td>NFLTS(I)</td>
<td>-</td>
<td>0 indicates flechette reference diameter (DM(I,L)) will be varied, but the same drag curve will be used; 1 indicates more than one drag curve will be used (if desired DM(I,L) can be also varied).</td>
<td></td>
</tr>
</tbody>
</table>

(IF INTD(I) = 0 INPUT CARDS (D-F) AND REPEAT D (NTC(I) - 1) TIMES, IF INTC(I) = 1 REPEAT CARDS D-F NTC(I) TIMES)

<table>
<thead>
<tr>
<th>D</th>
<th>1-10</th>
<th>DEC</th>
<th>DM(I,L)</th>
<th>-</th>
<th>Flechette reference diameter (for ballistic computations).</th>
<th>Inches</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>11-20</td>
<td>DEC</td>
<td>DTU(I,L)</td>
<td>-</td>
<td>AT upper - Time increment used in trajectory iterations for all Mach numbers greater than MACHX(I,L).</td>
<td>Seconds</td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>DEC</td>
<td>MACHX(I,L)</td>
<td>-</td>
<td>Mach number at which the iterative time increment is changed from DTU(I,L) to DTL(I,L).</td>
<td></td>
</tr>
<tr>
<td>Card</td>
<td>Columns</td>
<td>Mode</td>
<td>Quantity</td>
<td>Limits</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>----------------</td>
<td>--------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>D</td>
<td>31-40</td>
<td>DEC</td>
<td>DTL(I,L)</td>
<td>ΔT lower - Time increment used in trajectory iterations for all Mach numbers less than MACHX(I,L).</td>
<td>Seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>41-50</td>
<td>DEC</td>
<td>TMIN(I,L)</td>
<td>Lower time limit on drag curve (I,L) measured in real time during flechette flight.</td>
<td>Seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>51-60</td>
<td>DEC</td>
<td>TMAX(I,L)</td>
<td>Upper time limit on drag curve (I,L) measured in real time during flechette flight.</td>
<td>Seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>61-65</td>
<td>DEC</td>
<td>DTSP(I,L)</td>
<td>Optional ΔT that may be used when INDTD(I) ≠ 0.</td>
<td>Seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>66-70</td>
<td>DEC</td>
<td>TMAXSP(I,L)</td>
<td>Cut off time for DTSP(I,L) (Input only when DTSP(I,L) &gt; 0).</td>
<td>Seconds</td>
<td></td>
</tr>
<tr>
<td></td>
<td>71-72</td>
<td>INTEG</td>
<td>NPT(I,L)</td>
<td>Number of points in drag coefficient curve for Class I, time span L.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(If INDTD(I) = 0, cards E and F are input only for L = 1)

| E    | 1-10   | DEC  | MACH(I,L,1)    | Mach number data points on drag coefficient versus Mach number curve for Class I, time span L. | |
|      | 11-20  | DEC  | MACH(I,L,2)    | |
|      |       |      |                | (repeat card E until NPT(I,L) have been input) | |
|      | 61-70  | DEC  | MACH(I,L,7)    | |

| F    | 1-10   | DEC  | CD(I,L,1)      | Coefficient of drag for corresponding MACH(I,L,1) above | |
|      | 11-20  | DEC  | CD(I,L,2)      | |
|      |       |      |                | (repeat card F until NPT(I,L) have been input) | |
|      | 61-70  | DEC  | CD(I,L,7)      | |

| G    | 5      | INTEG| ITROOP         | 1 = Prone Troops. 2 = Standing Troops. 3 = Troops in foxholes. | Feet |
|      | 11-20  | DEC  | 2T             | Height of target centroid. | |

*DTSP(I,L) is an optional time increment that is used in lieu of DTU(I,L) or DTL(I,L) for flechette flight time from TMAX(I,L-1) to TMAXSP(I,L) following change of drag curves. TMAXSP(I,L) assumes some value greater than TMAX(I,L-1) but less than TMAX(I,L), i.e., TMAXSP(I,L) limits the trajectory real time in which DTSP(I,L) is used instead of DTU(I,L) or DTL(I,L). This optional ΔT has been included to provide a finer integration at the critical time in the trajectory simulation where transition of Cdp curves is made. This input may be used only when more than one drag curve and/or flechette reference diameter DN(I,L) has been input for a given class. It may be input only on card D for L > 1. DTSP(I,L) is used only for the time specified, therefore DTU(I,L) and DTL(I,L) may not be omitted.
<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>21-30</td>
<td>DEC</td>
<td>A</td>
<td></td>
<td>Constants defining casualty criterion in accordance with &quot;Casualty Criterion for Wounding Soldiers&quot; BRL TN1486.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>31-40</td>
<td>DEC</td>
<td>B</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>41-50</td>
<td>DEC</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card</td>
<td>Columns</td>
<td>Mode</td>
<td>Quantity</td>
<td>Limits</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>--------</td>
<td>----------</td>
<td>---------</td>
<td>--------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>1-7-9</td>
<td>Alphanum</td>
<td>*</td>
<td></td>
<td>Multiple run card</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>1-72</td>
<td>Alphanum</td>
<td>ITITLE</td>
<td></td>
<td>Title card</td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>1-5</td>
<td>Integer</td>
<td>IADD</td>
<td>1 &lt; 900</td>
<td>Clobber data table address of first word on card.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7-9</td>
<td>Integer</td>
<td>N</td>
<td></td>
<td>The total number of following input values to be placed in sequential memory locations beginning with the address given in columns 1-5.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>DCLOB(1)</td>
<td></td>
<td>Data word</td>
<td></td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>61-70</td>
<td>Decimal</td>
<td>DCLOB(6)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X*</td>
<td>1-10</td>
<td></td>
<td></td>
<td></td>
<td>Blank</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>DCLOB(7)</td>
<td></td>
<td>Data word</td>
<td></td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>Decimal</td>
<td>DCLOB(8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>61-70</td>
<td>Decimal</td>
<td>DCLOB(12)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>1-5</td>
<td>Integer</td>
<td>22222</td>
<td></td>
<td>End of data</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
<th>Parameter</th>
<th>Parameter</th>
<th>Address</th>
<th>Parameter</th>
<th>Address</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IPR1</td>
<td>12-14</td>
<td>NTHTHA(3)</td>
<td>50</td>
<td>ALPHA</td>
<td>122-136</td>
</tr>
<tr>
<td>2</td>
<td>IPR2</td>
<td>15-17</td>
<td>NPHI(3)</td>
<td>51</td>
<td>HOB</td>
<td>137-151</td>
</tr>
<tr>
<td>3</td>
<td>IPR3</td>
<td>18-20</td>
<td>NTC(3)</td>
<td>52</td>
<td>GRAREA</td>
<td>155-166</td>
</tr>
<tr>
<td>4</td>
<td>IPR4</td>
<td>21-23</td>
<td>NFLS(3)</td>
<td>53-55</td>
<td>FLMAS(3)</td>
<td>167-181</td>
</tr>
<tr>
<td>5</td>
<td>IPR5</td>
<td>24-38</td>
<td>NFT(3,5)</td>
<td>56-58</td>
<td>VF(3)</td>
<td>182-556</td>
</tr>
<tr>
<td>6</td>
<td>NGRS</td>
<td>39</td>
<td>ITROOP</td>
<td>59-61</td>
<td>SIGMA(3)</td>
<td>557-931</td>
</tr>
<tr>
<td>7</td>
<td>NGRID</td>
<td>40-49</td>
<td>IEKROOP(10)</td>
<td>62-76</td>
<td>DM(3,5)</td>
<td>932</td>
</tr>
<tr>
<td>8</td>
<td>MAXGR</td>
<td></td>
<td></td>
<td>77-91</td>
<td>DTU(3,5)</td>
<td>933</td>
</tr>
<tr>
<td>9-11</td>
<td>IDIST(3)</td>
<td></td>
<td></td>
<td>92-106</td>
<td>MACHX(3,5)</td>
<td>934</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>107-121</td>
<td>DTL(3,5)</td>
<td>935</td>
</tr>
</tbody>
</table>

*If $N \leq 6$, omit K cards.

Use as many J cards as necessary.

Cards H through L may be repeated any number of times.

For run number $n$ ($n > 1$) it is necessary to read in the values of only those input variables whose values are to be different from what they were for run $n-1$.

Note: All variables are stored in "natural order". For example Table VII-II indicates that MACH is a three dimensional array of numbers $25 \times 5 \times 3$, or a total of 375 storage locations. Table VII-II also specifies that the MACH array begins in address 182. Then elements of MACH are stored in the following manner (innermost subscript varies most rapidly).
<table>
<thead>
<tr>
<th>Address</th>
<th>Contents</th>
<th>Address</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>182</td>
<td>MACH(1,1,1)</td>
<td>282</td>
<td>MACH(1,5,1)</td>
</tr>
<tr>
<td>183</td>
<td>MACH(2,1,1)</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>184</td>
<td>MACH(3,1,1)</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>306</td>
<td>MACH(25,5,1)</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>307</td>
<td>MACH(1,1,2)</td>
</tr>
<tr>
<td>181+i</td>
<td>MACH(1,1,1)</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>331</td>
<td>MACH(25,1,2)</td>
</tr>
<tr>
<td>206</td>
<td>MACH(25,1,1)</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>207</td>
<td>MACH(1,2,1)</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>208</td>
<td>MACH(2,2,1)</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>556</td>
<td>MACH(25,5,3)</td>
</tr>
<tr>
<td>231</td>
<td>MACH(25,2,1)</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>
APPENDIX IX
UTILIZATION REPORT - FRAGMENTATION/BLAST WARHEAD
OPTIMIZATION PROGRAM

1. PROGRAM DESCRIPTION

This program was assembled to enable the optimization of fragmentation, blast, and fragmentation/blast area weapons against a spectrum of target types (material and personnel) numbering from one to five. The program will now handle warheads containing a plane of fragments as well as the regular fragmentation type warheads. The warhead lethal area is maximized as a function of weapon parameters such as height of burst, terminal velocity and attack angle, fragment mass or masses and fragment distributions. A parametric mode is available for use when optimization procedures are undesirable. Additional capability has been included to evaluate fractional kill probabilities of cluster munitions against area targets. This portion of the program uses maximum lethal areas to optimize pattern size and kill probability associated with fractional coverage simulations.

2. TIMING

A typical run optimizing 2 variables against one target using every radius at grid intervals of 5 degrees and utilizing the option of calculating optimum pattern sizes for fractional coverage, takes about 65 seconds on the CDC 6600.

3. INPUT DATA FORMAT

All integers must be right adjusted in their fields. Table IX-I contains optimization or parametric data, Table IX-II lists variable names, Table IX-III is used for warheads with constant charge to mass ratio, Table IX-IV is used for warheads with varying charge to mass ratio, and Table IX-V contains target data.

4. MULTIPLE RUNS

Runs may be stacked by placing a *RUN card after the last data card, and then loading new input data per instructions found in Table IX-VI.

### TABLE IX-I. OPTIMIZATION OR PARAMETRIC DATA

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quality</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1-72</td>
<td>Alphanum</td>
<td>GTITLE</td>
<td>0 &lt; 15</td>
<td>General title.</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>1-5</td>
<td>Integer</td>
<td>NVOPT</td>
<td>0 &lt; 8</td>
<td>Number of variables to be optimized.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>Integer</td>
<td>NPAR</td>
<td>0 &lt; 8</td>
<td>Number of variables in parametric mode: If NVOPT &gt; 0, NPAR = 0 and vice-versa.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>CC1</td>
<td></td>
<td>Optimization procedure will not converge until the following two conditions are satisfied:</td>
<td></td>
</tr>
</tbody>
</table>
|      | 21-30   | Decimal | CC2     |        | \[
|      |         |         |         | \[A_L(N-1) - A_L(N-2) < CC1 \] and \[
|      |         |         |         | \[A_L(N) - A_L(N-1) < CC2 \] \] |       |

where $A_L$ is lethal area; $N$ is the optimization trial number.
### Table IX—I—Concluded

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quality</th>
<th>Limits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td>Integer</td>
<td>IPR6</td>
<td>0 or 1</td>
<td>1</td>
<td>Indicates intermediate print in optimization portion of program.</td>
</tr>
<tr>
<td>40</td>
<td>Integer</td>
<td>IOPPAT</td>
<td>0 or 1</td>
<td>1</td>
<td>= optimize pattern size. If in parametric mode, IEACH = 1 indicates pattern size will be optimized for each set of conditions. If IEACH = 0, pattern size will be optimized for the largest value of ( N ) (ap) where ( N ) is the number of target elements.</td>
</tr>
<tr>
<td>45</td>
<td>Integer</td>
<td>IEACH</td>
<td>0 or 1</td>
<td>1</td>
<td>= optimize pattern size. If in parametric mode, IEACH = 1 indicates pattern size will be optimized for each set of conditions. If IEACH = 0, pattern size will be optimized for the largest value of ( N ) (ap) where ( N ) is the number of target elements.</td>
</tr>
</tbody>
</table>

**C** 1-6 Alphanumeric NAME(I) Name of variable to be optimized or parameterized (see Table II).

11-20 Decimal XMIN(I) Variable minimum.

21-30 Decimal XMAX(I) Variable maximum.

31-40 Decimal X(I) First estimate in optimization mode; Delta X in parametric mode.

| D | Integer | IPRT1 | 1 or 0 | =1, Prints static, dynamic and fragmentation warhead data. |
| 1 | Integer | IPRT2 | 1 or 0 | =1, Prints effective and lethal area data. |
| 2 | Integer | IPRT3 | 1 or 0 | =1, Print PK matrix. |
| 3 | Integer | IPRT4 | 1 or 0 | =1, Prints target data input. |
| 4 | Integer | IPRT5 | 1 or 0 | =0, Special print used for program checkout only. Any or all of the above indicators may be set to 0 and the specified print-out will be omitted. The program contains a table of radii for the target damage assessment grid. INDXR = 1, indicates every radius in table is to be used. 2 indicates every other radius; 3 indicates every third radius, etc. |
| 5 | Integer | INDXR | 1 < 9 | |

21-30 Decimal DGRID 5 ≤ 180 Angular increment for target damage assessment grid. Should divide into 180 evenly.

31-40 Decimal ALPHA 0 ≤ 90 Warhead terminal attack angle. Degrees.

41-50 Decimal HOB 0 ≤ 90 Warhead height of burst. Ft.

51-60 Decimal VM 0 ≤ 90 Warhead terminal velocity. Ft./Sec.

61-70 Decimal RGMAX* Maximum effective radius to be evaluated. If RGMAX = 0, grid radius will be determined by a calculated effective range of warhead. Ft.

*Measured from the point of warhead burst to target centroid.

Card C is repeated for each variable being optimized or parameterized. Col 1-5 are print indicators.

NOTE - If optimizing ALPHA, HOB, or VM the above values should correspond with first estimate on Card C.
**TABLE IX-II. VARIABLE NAMES**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Name</th>
<th>Description</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>HOB</td>
<td>Warhead height of burst</td>
<td>Ft.</td>
</tr>
<tr>
<td>1-2</td>
<td>VM</td>
<td>Terminal warhead velocity</td>
<td>Ft./Sec.</td>
</tr>
<tr>
<td>1-5</td>
<td>ALPHA</td>
<td>Terminal warhead attack angle</td>
<td>Degrees</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(1)</td>
<td>Fragment mass, Group 1</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(2)</td>
<td>Fragment mass, Group 2</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(3)</td>
<td>Fragment mass, Group 3</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(4)</td>
<td>Fragment mass, Group 4</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(5)</td>
<td>Fragment mass, Group 5</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(6)</td>
<td>Fragment mass, Group 6</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(7)</td>
<td>Fragment mass, Group 7</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(8)</td>
<td>Fragment mass, Group 8</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(9)</td>
<td>Fragment mass, Group 9</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>FM(10)</td>
<td>Fragment mass, Group 10</td>
<td>Grains</td>
</tr>
<tr>
<td>1-5</td>
<td>ZONEL</td>
<td>Lower limit of first zone in warhead</td>
<td>Degrees</td>
</tr>
<tr>
<td>1-5</td>
<td>ZONEL</td>
<td>Upper limit of last zone in warhead</td>
<td>Degrees</td>
</tr>
<tr>
<td>1-3</td>
<td>FM1</td>
<td>Fragment mass for plane of fragments</td>
<td>Grains</td>
</tr>
</tbody>
</table>

Names must begin in column 1 and be in the exact format given.

**NOTE** - Use either Table IX-III or IX-IV as appropriate.

**TABLE IX-III. WARHEADS WITH CONSTANT CHARGE TO MASS RATIO**

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td>1-72</td>
<td>Alphanumeric</td>
<td>WTITLE</td>
<td></td>
<td>Warhead title.</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>1-5</td>
<td>Integer</td>
<td>NPOLAR*</td>
<td>1 ≤ 36</td>
<td>Number of polar zones required to describe the fragmentation data for warhead.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>Integer</td>
<td>NPOLAR*</td>
<td></td>
<td>Blank.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>PBEXP**</td>
<td></td>
<td>The bare charge of TNT equivalent explosive in the warhead.</td>
<td>Lbs.</td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>Decimal</td>
<td>RHOMTL</td>
<td></td>
<td>Density of fragment material. May leave blank if using steel fragments.</td>
<td>lb/in³</td>
</tr>
<tr>
<td></td>
<td>35</td>
<td>Integer</td>
<td>MINI</td>
<td>0 or 1</td>
<td>l indicates warhead contains a plane of fragments. Blank.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>36-60</td>
<td>Integer</td>
<td>FACTK+</td>
<td></td>
<td>K Factor. FPA = (FACTK) (FM)²/3.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>61-70</td>
<td>Integer</td>
<td>IFRAG</td>
<td>0 or 1</td>
<td>Fragment density indicator. If IFRAG = 0 the fragment density, rho(I,J), is input in fragments/steradian; if IFRAG = 1, the fragment density is input as the total number of fragments and the program computes fragments/steradian.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>71</td>
<td>Integer</td>
<td>INDCD</td>
<td>1 ≤ 9</td>
<td>Drag indicator: 1 = sphere; 2 = cube; 3 = random. If INDCD &gt; 3 then a special drag curve must be input using card II.</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>1-5</td>
<td>Integer</td>
<td>NGPS(I)</td>
<td>1 ≤ 10</td>
<td>Number of classes of fragmentation data in polar zone I. All zones must have same number of classes.</td>
<td></td>
</tr>
<tr>
<td>Card</td>
<td>Columns</td>
<td>Mode</td>
<td>Quantity</td>
<td>Limits</td>
<td>Description</td>
<td>Units</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>----------</td>
<td>--------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>THSU(I)</td>
<td>0 &lt; 180</td>
<td>Upper angle defining polar zone I for warhead.</td>
<td>Degree</td>
<td></td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>THSL(I)</td>
<td>0 &lt; 180</td>
<td>Lower angle defining polar zone I for warhead.</td>
<td>Degree</td>
<td></td>
</tr>
<tr>
<td>H 1-10</td>
<td>Decimal</td>
<td>FM(I,J)</td>
<td>0 &lt; 180</td>
<td>Fragment mass for polar zone I, and class J.</td>
<td>Grain</td>
<td></td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>PV(I,J)</td>
<td>Initial velocity for polar zone I and class J.</td>
<td>Ft./Sec.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>FPA(I,J)</td>
<td>Fragment presented area for polar zone I and class J.</td>
<td>Ft.²</td>
<td></td>
<td></td>
</tr>
<tr>
<td>41-50</td>
<td>Decimal</td>
<td>RHO(I,J)</td>
<td>Fragment density or number of fragments for polar zone I and class J.</td>
<td>Grain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>51-60</td>
<td>Decimal</td>
<td>Dl(J)±</td>
<td>Width of rectangular fragments in class J.</td>
<td>Inches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>61-70</td>
<td>Decimal</td>
<td>D2(J)±</td>
<td>Height of rectangular fragments in class J.</td>
<td>Inches</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**NOTE** - Card H is repeated NGPS(I) times.

**NOTE** - Cards G and H are repeated for each polar zone.

**NOTE** - Omit H' if MIN I = 0.

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>H' 1-10</td>
<td>Decimal</td>
<td>FMI</td>
<td>Fragment mass for MINI plane of fragments.</td>
<td>Grain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>FV1</td>
<td>Initial velocity for MINI plane of fragments.</td>
<td>Ft./Sec.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>FPA1</td>
<td>Fragment presented area for MINI plane of fragments.</td>
<td>Ft.²</td>
<td></td>
<td></td>
</tr>
<tr>
<td>41-50</td>
<td>Decimal</td>
<td>RHO1</td>
<td>Number of fragments in MINI plane of fragments.</td>
<td>Grain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>51-60</td>
<td>Decimal</td>
<td>DDL±</td>
<td>Width of rectangular fragments in MINI plane of fragments.</td>
<td>Inches</td>
<td></td>
<td></td>
</tr>
<tr>
<td>61-70</td>
<td>Decimal</td>
<td>D2(J)±</td>
<td>Height of rectangular fragments in MINI plane of fragments.</td>
<td>Inches</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

If INDIC < 3, omit cards II (Special drag table)

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>II 1-10</td>
<td>Decimal</td>
<td>CD(1)</td>
<td>Drag coefficient at velocity = 0 ft./sec.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>CD(2)</td>
<td>Drag coefficient at velocity = 447 ft./sec.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. .</td>
<td>. .</td>
<td>. .</td>
<td>. .</td>
<td>. .</td>
<td>. .</td>
<td></td>
</tr>
<tr>
<td>61-70</td>
<td>Decimal</td>
<td>CD(7)</td>
<td>Drag coefficient at velocity = 1005 ft./sec.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* If warhead contains only a plane of fragments and no fragments in polar zones, then NPOLAR may be set equal to 0 and cards G and H omitted.

** Not required if using blast radii, i.e., IRAD = 1 in target deck.

† Leave blank if fragment presented area is used.

‡ Not required if fragment mass is not being optimized or parameterized.

Card II is repeated until 25 drag coefficients have been input (7 per card).

The corresponding velocities must be: 0, 447, 558, 670, 782, 893, 1005, 1116, 1228, 1340, 1451, 1563, 1786, 1898, 2010, 2233, 2679, 3014, 3350, 3908, 4679, 5024, 5582, 6475, 10048.
<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td>1-72</td>
<td>Alphanum</td>
<td>WITLE</td>
<td></td>
<td>Warhead title.</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>5</td>
<td>Integer</td>
<td>IWHDT</td>
<td>1 or 2</td>
<td>Warhead type: 1 = sphere or tabular data: 2 = cylinder.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>Integer</td>
<td>NPOLAR</td>
<td>1 ≤ 36</td>
<td>Number of polar zones required to describe the fragmentation data for warhead.</td>
<td></td>
</tr>
<tr>
<td>11-20</td>
<td>Decimal</td>
<td>PBEXP*</td>
<td></td>
<td></td>
<td>The bare charge of TNT equivalent explosive in the warhead</td>
<td>Lbs.</td>
</tr>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>RHOMTL</td>
<td></td>
<td></td>
<td>Density of fragment material. Leave blank if using steel fragments.</td>
<td>Lb/in³</td>
</tr>
<tr>
<td>35</td>
<td>Integer</td>
<td>MINI</td>
<td></td>
<td></td>
<td>1 indicates warhead contains a plane of fragments.</td>
<td></td>
</tr>
<tr>
<td>41-50</td>
<td>Decimal</td>
<td>WLTH</td>
<td></td>
<td></td>
<td>Length of warhead.</td>
<td>Inches</td>
</tr>
<tr>
<td>51-60</td>
<td>Decimal</td>
<td>MDIAM</td>
<td></td>
<td></td>
<td>Diameter of warhead.</td>
<td>Inches</td>
</tr>
<tr>
<td>61-70</td>
<td>Decimal</td>
<td>FACTK**</td>
<td></td>
<td></td>
<td>Fragment density indicator</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>Integer</td>
<td>IFRAG</td>
<td>0 or 1</td>
<td></td>
<td>If IFRAG = 0 the fragment density, RHO (I,J), is input in fragments/steradian; if IFRAG = 1, the fragment density is input as the total number of fragments and the program computes fragment/steradian.</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>Integer</td>
<td>INDCD</td>
<td>1 &lt; 9</td>
<td></td>
<td>Drag indicator: 1 = sphere; 2 = cube; 3 = random. If INDCD &gt; 3 then a special drag curve must be input using cards II.</td>
<td></td>
</tr>
</tbody>
</table>

**NOTE** - Card G is repeated NPOLAR times.

| G    | 1-5    | Integer | NGPS(I) | 1 ≤ 10 | Number of classes of fragmentation data in polar zone I. All zones must have same number of classes. |       |
|      | 11-20  | Decimal | THSU(I) | 0 ≤ 180| Upper angle defining polar zone I for warhead. | Degree |
|      | 21-30  | Decimal | THSL(I) | 0 ≤ 180| Lower angle defining polar zone I for warhead. | Degree |
| H1   | 5      | Integer | IFORM   | 0 or 1 | 0 = tabular warhead data, 1 = Gurney formula. |       |
|      | 6-10   | Integer | MMAS+   | 2 ≤ 14 | Number of masses in table. |       |
|      | 11-20  | Decimal | ENERGY+ |        | Energy constant. | Ft./Sec. |
|      | 21-30  | Decimal | PACKEFP |        | Packing efficiency, e_p. | Lbs./in.³ |
|      | 31-40  | Decimal | RHOEXTP |        | Density of matrix material, p_m. | Lbs./in.³ |
|      | 41-50  | Decimal | RHOEXP+ |        | Density of explosive p_e. | Lbs./in.³ |

**NOTE** - Cards H2 through H4 not required if IFORM = 1.

| H2   | 1-10   | Decimal | SFMAS(I) |        | First mass in table. | Grains |
|      | 1-10   | Decimal | SVEL(1)  |        | Spherical fragment velocity in 1st class. | Ft./Sec. |
|      | 11-20  | Decimal | SVEL(2)  |        | Spherical fragment velocity in 2nd class. | Ft./Sec. |
**TABLE IX-IV—CONCLUDED**

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>21-30</td>
<td>Decimal</td>
<td>SVEL(3)</td>
<td></td>
<td></td>
<td>Spherical fragment velocity in 3rd class.</td>
<td>Ft./Sec.</td>
</tr>
<tr>
<td>61-70</td>
<td>Decimal</td>
<td>SVEL(7)</td>
<td></td>
<td></td>
<td>Spherical fragment velocity in 7th class.</td>
<td>Ft./Sec.</td>
</tr>
<tr>
<td>H4</td>
<td>1-10</td>
<td>SRHO(1)</td>
<td></td>
<td></td>
<td>Number of spherical fragments in 1st class.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>SRHO(2)</td>
<td></td>
<td></td>
<td>Number of spherical fragments in 2nd class.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>61-70</td>
<td>SRHO(7)</td>
<td></td>
<td></td>
<td>Number of spherical fragments in 7th class.</td>
<td></td>
</tr>
</tbody>
</table>

Repeat cards H3 and H4 for each polar zone.

Repeat cards H2 through H4 NMAS times.

If IFORM = 1, the warhead is assumed to be a simple (one polar zone, one fragment layer) sphere or cylinder. If IFORM = 0, all fragments of all layers are assumed to be the same mass and (NMAS) (NPOLAR) (NGPS) ≤ 90.

If INDCD < 3, omit card II (Special drag table).

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>1-10</td>
<td>CD(1)</td>
<td></td>
<td></td>
<td>Drag coefficient at velocity = 0 Ft./Sec.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>CD(2)</td>
<td></td>
<td></td>
<td>Drag coefficient at velocity = 447 Ft./Sec.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>61-70</td>
<td>CD(7)</td>
<td></td>
<td></td>
<td>Drag coefficient at velocity = 1005 Ft./Sec.</td>
<td></td>
</tr>
</tbody>
</table>

* Not required if using blast radii, i.e., IRAD = 1 in target data.
** Leave blank if fragment presented area is used.
† Not required if IFORM = 0.
‡ Not required if IFORM = 1.
Card II is repeated until 25 drag coefficients have been input (7 per card).
The corresponding velocities must be: 0, 447, 558, 670, 782, 893, 1005, 1116, 1228, 1340, 1451, 1563, 1786, 1898, 2010, 2233, 2679, 3014, 3350, 3908, 4679, 5024, 5582, 6475, 10048.

**TABLE IX-V. TARGET DATA**

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>5</td>
<td>Integer</td>
<td>NELMTS</td>
<td>1 &lt; 5</td>
<td>Number of elements.</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Integer</td>
<td>NESH</td>
<td>1 &lt; 5</td>
<td>Number of material target elements which are same height. These should be grouped together as the first targets to be input.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Integer</td>
<td>LTYPE(1)</td>
<td>0 or 1</td>
<td>Element No. 1: 0 = material, 1 = personnel.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

144
<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>J</td>
<td>1-72</td>
<td>Alphanumeric</td>
<td>ETITLE(I)</td>
<td></td>
<td>Element title.</td>
<td></td>
</tr>
</tbody>
</table>

**NOTE - If the target is personnel, skip to Card R.**

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>1-5</td>
<td>Integer</td>
<td>NPMS(I)</td>
<td>1 ≤ 12</td>
<td>Number of fragment masses used to describe the vulnerability of target element I.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>Integer</td>
<td>NVEL(I)</td>
<td>1 ≤ 12</td>
<td>Number of fragment velocities used to describe the vulnerability of target element I.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-15</td>
<td>Integer</td>
<td>NELVS(I)</td>
<td>1 ≤ 6</td>
<td>Number of elevation angles used to describe the vulnerability of target element I.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>16-26</td>
<td>Integer</td>
<td>TLGTH(I)</td>
<td></td>
<td>Length of element (I).</td>
<td>Ft</td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>Decimal</td>
<td>TWDTH(I)</td>
<td></td>
<td>Width of element (I).</td>
<td>Ft</td>
</tr>
<tr>
<td></td>
<td>31-40</td>
<td>Decimal</td>
<td>THGT(I)</td>
<td></td>
<td>Height of element (I).</td>
<td>Ft</td>
</tr>
<tr>
<td></td>
<td>41-50</td>
<td>Decimal</td>
<td>ZT(I)</td>
<td></td>
<td>Height of target element centroid.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>51-60</td>
<td>Decimal</td>
<td>IKT(I)</td>
<td>0, 1, 2</td>
<td>Number of fragment masses used to describe the vulnerability of target element X.</td>
<td></td>
</tr>
</tbody>
</table>

If IKT(I) = 0, the target vulnerability data as a function of ELVS elevation angle are used. If IKT(I) = 1, the target vulnerability data for the upper hemisphere are used. If IKT(I) = 2, this option is incorporated into the program to evaluate vulnerability for future elements. Not needed at this time.

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>1-5</td>
<td>Decimal</td>
<td>AV(IT,JT,1)</td>
<td></td>
<td>Vulnerability area for fragment mass IT:</td>
<td>Ft.²</td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>Decimal</td>
<td>AV(IT,JT,2)</td>
<td></td>
<td>Fragment velocity, JT: and fragment striking angle, KT.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11-15</td>
<td>Decimal</td>
<td>AV(IT,JT,3)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card</td>
<td>Columns</td>
<td>Mode</td>
<td>Quantity</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card M</td>
<td>61-65</td>
<td>Decimal</td>
<td>AV(IT,JT, 13)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Card N</td>
<td>66-70</td>
<td>Decimal</td>
<td>AV(IT,JT, 13)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N</td>
<td>1-5</td>
<td>Decimal</td>
<td>FMAS(I,1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-10</td>
<td>Decimal</td>
<td>FMAS(I,2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>11-15</td>
<td>Decimal</td>
<td>FMAS(I,3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>O</td>
<td>1-5</td>
<td>Decimal</td>
<td>VEL(I,1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-10</td>
<td>Decimal</td>
<td>VEL(I,2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>11-15</td>
<td>Decimal</td>
<td>VEL(I,3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>1-5</td>
<td>Decimal</td>
<td>ELV(I,1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-10</td>
<td>Decimal</td>
<td>ELV(I,2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>11-15</td>
<td>Decimal</td>
<td>ELV(I,3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>1-5</td>
<td>Decimal</td>
<td>VEETBL(I,IT,1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6-10</td>
<td>Decimal</td>
<td>VEETBL(I,IT,2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>26-30</td>
<td>Decimal</td>
<td>VEETBL(I,IT,KT)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Card M is repeated NFMS(I), NVEL(I) times.
Card Q is repeated NFMS(I) times.
Cards J through Q are repeated for each material target.

If target is personnel, omit cards K through Q and use the following cards in their place.

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>1-5</td>
<td>Integer</td>
<td>ITROOP(I)</td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>ST(I)</td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>Decimal</td>
<td>BLSTLI(I)</td>
</tr>
<tr>
<td></td>
<td>31-40</td>
<td>Decimal</td>
<td>BLSTLII(I)</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>Integer</td>
<td>IRAD(I)</td>
</tr>
</tbody>
</table>

Personnel target type:
1 = prone, 2 = standing, 3 = foxhole.
Weight of target.
See card L.
See card L.
See card L.

KT = 13 if IKT = 1.
KT = 14 if IKT = 2.
An ascending ordered table of fragment masses used in the vulnerability data for element I.
An ascending ordered table of fragment velocities used in the vulnerability data for element I.
An ascending ordered table of fragment striking angles used in the vulnerability data for element I.
Minimum lethal fragment velocity for target element I, fragment mass (IT), fragment striking angle (KT).

Ft/s, Ft./Sec., Degrees
### TABLE IX-V---CONCLUDED

<table>
<thead>
<tr>
<th>Card</th>
<th>Columns</th>
<th>Mode</th>
<th>Quantity</th>
<th>Limits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>1-10</td>
<td>Decimal</td>
<td>A</td>
<td></td>
<td>Constants defining casualty criterion in accordance with &quot;Casualty Criterion for Wounding Soldiers,&quot; BRL TN 1486.</td>
</tr>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>B</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>21-30</td>
<td>Decimal</td>
<td>C</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Repeat cards J, R and S for each personnel target.

If IOPPAT = 0, omit cards T thru V

<table>
<thead>
<tr>
<th>T</th>
<th>1-10</th>
<th>Decimal</th>
<th>TARHL</th>
<th></th>
<th>Target half length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>TARHW</td>
<td></td>
<td>Target half width</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>Integer</td>
<td>IDTAR</td>
<td>0 or 1</td>
<td>0 indicates elliptical target shape; 1 indicates a rectangular target shape.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>1-10</th>
<th>Decimal</th>
<th>PATHL</th>
<th></th>
<th>Weapon pattern half length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>PATHW</td>
<td></td>
<td>Weapon pattern half width</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>Integer</td>
<td>IDPAT</td>
<td>0 or 1</td>
<td>0 indicates elliptical pattern shape; 1 indicates rectangular pattern shape.</td>
</tr>
</tbody>
</table>

| 31-40 | Decimal | SIGL* |        | Standard deviation on pattern center in range direction                  |
| 41-50 | Decimal | SIGW* |        | Standard deviation on pattern center in cross range direction            |
| 51-60 | Decimal | DL**  |        | Delta on pattern size                                                    |
| 61-70 | Integer | NBLTS |        | Number of bomblets                                                       |

<table>
<thead>
<tr>
<th>V</th>
<th>1-10</th>
<th>Decimal</th>
<th>FP(1)</th>
<th>0 ≤ 1.0</th>
<th>Weighting factors for target element I.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>11-20</td>
<td>Decimal</td>
<td>FP(2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>41-50</td>
<td>Decimal</td>
<td>FP(5)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Normal distribution used.
**Optimization procedure increments the pattern length and width equally.

### TABLE IX-VI. MULTIPLE RUNS

**Multiple Runs**: Runs may be stacked by placing a *RUN card (**in column 1, RUN in columns 7 - 9) after last data card and then repeating the cards in Table I. The GTITLE card is used to indicate if the warhead data and/or the target data are to be changed.

**GTITLE Card**:

```
1  16  20  34  35  72
```

**NEW WARHEAD DATA**

**NEW TARGET DATA**

If NEW WARHEAD DATA is in GTITLE card columns 1-16, then Table IX-III or IX-IV cards must be input. If NEW TARGET DATA is in columns 20-34, then the entire Table IX-IV card deck must be input. Columns 35-72 may contain any other information to be printed at top of page.

This may be done any number of times, thus enabling the user to optimize several warheads against various targets.

**Description of Output**: All output is labeled and self-explanatory.

**Machine Requirements**: Fortran IV monitor and 2 binary scratch tapes.
REFERENCES


5. Kunz, K. S., Numerical Analysis, 1957
WEAPON OPTIMIZATION TECHNIQUES

This investigation established the techniques to evaluate the lethality of blast/fragmentation, continuous rod, and submissile warheads when deployed against airborne targets. A computer program was developed which is capable of optimizing warhead design parameters and terminal encounter conditions or of parametrically examining the effect on lethality of design and encounter variables against any air target. In addition, a second computer program was developed to compute the lethal area of flechette-projecting warheads deployed against standing, prone or foxholed troop concentrations. The program simulates the actual flechette trajectories to properly account for the effects of gravity, and establishes the lethal area by integrating kill probability over the impact pattern area. Sections I through IV contain the air encounter optimization computer program. Sections V through VIII contain the documentation for the flechette lethal area program, and modifications to a previously developed surface targets weapon optimization lethal area program.
<table>
<thead>
<tr>
<th>KEY WORDS</th>
<th>LINK A</th>
<th>LINK B</th>
<th>LINK C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ROLE</td>
<td>WT</td>
<td>ROLE</td>
</tr>
<tr>
<td>Weapon optimization techniques</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Computer program, weapon optimization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Air-to-air weapon optimization techniques</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Submunition warhead</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Warhead design parameters</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Terminal encounter conditions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flechette - projecting warheads</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>