JSTPS-TR-76-2

A COMPUTER ROUTINE FOR PROBABILITY OF DAMAGE CALCULATIONS

PDCALC

TECHNICAL REPORT NO. JSTPS-TR-76-2

Robert H. Miller

Lieutenant Colonel, USA

JOINT STRATEGIC TARGET PLANNING STAFF
OFFUTT AIR FORCE BASE
NEBRASKA
68113

APRIL 1976

INCORPORATES CHANGE 2

ABSTRACT

The purpose of this report is to provide guidance to users of the JSTPS computer routine for nuclear weapon damage assessments and to detail the mathematical theory associated with probability of damage using the methodology of the Cumulative Log Normal Distance Damage Function. The JSTPS computer routine is a flexible assemblage of FORTRAN IV subroutines suitable for interaction software applications as well as batch process programs. The FORTRAN coding was accomplished by Lt Fred E. Keller, ACS/Data Systems, Headquarters Strategic Air Command. The routine is dynamically maintained by the JSTPS to insure timely response to changes in vulnerability data and methodology. Correspondence pertaining to this memorandum should be addressed to:

Director
Strategic Target Planning
ATTN: JLTW
Offutt AFB, NE 68113

TABLE OF CONTENTS

I. INTRODUCTION

The JSTPS standardizes use of the Cumulative Log Normal (CLN) distance damage function commensurate with the existing publications and methodology of the Defense Intelligence Agency (DIA). Prior to introduction of the CLN function, the Hasting's approximation for a Circular Coverage Function (CCF) was in use; however, the CCF was unable to satisfy certain boundary conditions in probability theory. In special application programs and in certain mathematical situations, the CCF is still used as a first approximation to a solution. Examples are in optimization of DGZ locations and in offset distance computations. Therefore, the JSTPS computer routine includes both distance-damage function methods with the capability to permit the main calling program to direct computation by either method. Only the CLN function is detailed in this document.

The following probability of damage mathematical operations are internally performed in the JSTPS routine called SUBROUTINE PDCALC:

1. Vulnerability Number (VNTK) Decoding.
2. Vulnerability Number (VN) Adjustment for K-Factor.
3. Weapon Radius Computation.

4. Probability of Damage Calculation.

   a. Point Targets.

   b. Circular Normal Area Targets.

   c. Equivalent Target Area.

   d. Fatalities/Casualties.

5. DGZ-Target Offset Distance.

The data in PDCALC have been extracted from the DIA "Physical Vulnerability Handbook - Nuclear Weapons" and the methodology from "Mathematical Background and Programming Aids for the Physical Vulnerability System for Nuclear Weapons" also published by DIA.

II. MATHEMATICAL FORMULATION

A. Cumulative Log Normal Damage Function.

Probability of damage using the cumulative log normal methodology is given by:

$P_d(d) = \int \int$ [illegible] $\times\, e^{-\frac{(r^2+d^2-2rd\cos\theta)}{2\sigma^2}} \times P(r)\, r\, dr\, d\theta$

$$PD = \int_{r=0}^{r=\infty} f(r)\, dr$$

$$f(r) = P_d(r) \cdot r e^{-\frac{(r^2+x^2)}{2\sigma^2}} \cdot \frac{1}{2\pi\sigma^2} \int_0^{2\pi} e^{\frac{rx\cos\theta}{\sigma^2}}\, d\theta$$

$$P_d(r) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} \cdot e^{-y^2/2}\, dy$$

The latter, $P_d(r)$, is the cumulative log normal distance-damage function. It is, in fact, merely a Gaussian distribution where the variable (y) is a dummy random variable for integration. The factor (z) gives the upper limit for the integration as a function of weapon radius (w), target damage sigma (σ_d) and distance (r) from a DGZ (or AGZ with CEP = 0) to the target, point or area.

$$z = \frac{1}{\beta} \ln\left(\frac{\alpha}{r}\right)$$

$$\beta = \sqrt{-\ln(1-\sigma_d^2)} \qquad \alpha = w(1-\sigma_d^2)$$

In f(r), the terms other than P_d (r) represent (1) in the case of a point target, the circular normal distribution of impact points about a DGZ or (2) in the case of area targets, the joint circular normal distribution of the impact points about a DGZ along with the circular normal distribution of target elements about a target center.

Before evaluating any probability of damage, all variables are standardized (normalized).

For the circular normal distribution of impact points, the mathematical expression is (See Figure 1):

$$\int_0^{2\pi}\int_0^{\rho} \frac{1}{2\pi\sigma_w^2} e^{-\frac{\rho^2}{2\sigma_w^2}} \rho\, d\rho\, d\phi$$

Where .6551 σ_w is the standard deviation of this distribution.
By definition of CEP, it follows that:

$$\int_0^{2\pi}\int_0^{CEP} \frac{1}{2\pi\sigma_w^2} e^{-\frac{\rho^2}{2\sigma_w^2}} \rho\, d\rho\, d\phi = \frac{1}{2}$$
$$(\rho = CEP)$$

which gives:

$$\sigma_w = \frac{CEP}{\sqrt{2 \ln 2}} = \frac{CEP}{1.1774}$$

For the circular normal distribution of target elements, the mathematical expression is similar except now the parameter, σ_t, is for the distribution of target elements:

$$\frac{1}{2\pi\sigma_t^2} e^{-\frac{r^2}{2\sigma_t^2}} r\,dr\,d\theta$$

By definition of P-95 or R-95:

$$\int_0^{2\pi}\int_0^{R\text{-}95} \frac{1}{2\pi\sigma_t^2} e^{-\frac{r^2}{2\sigma_t^2}} r\,dr\,d\theta = 0.95$$

$$\sigma_t = \frac{R\text{-}95}{\sqrt{2\ln 20}} = \frac{R\text{-}95}{2.4477}$$

The origin of our coordinate system is always selected at the target coordinates (See Figure 2) in which case the polar variables are (r,θ). The circular normal distribution of impact points is easily written in terms of r and θ from the Law of Cosines and from realizing the $(\rho\ d\rho\ d\phi)$ is an infinitesimal area element, dA, equivalent to $(r\ dr\ d\phi)$ in magnitude.

It is easy to show that:

$$\int_0^{2\pi}\int_0^{\rho}\frac{1}{2\pi\sigma_w^2}e^{-\frac{\rho^2}{2\sigma_w^2}}\rho\, d\rho\, d\phi =$$

$$\int_0^{2\pi}\int_0^{r}\frac{1}{2\pi\sigma_w^2}e^{-\frac{(r^2+x^2-2rx\cos\theta)}{2\sigma_w^2}}r\, dr\, d\theta$$

$$\rho^2 = r^2 + x^2 - 2rx\cos\theta$$

The interpretation is that when evaluated either of the above expressions gives the probability of a weapon impacting in <u>any</u> infinitesimal area element, dA, located <u>between</u> the DGZ <u>and</u> a distance of p, or, between the DGZ and a distance of r from the target.

FIGURE 1 - CIRCULAR NORMAL DISTRIBUTION OF IMPACT POINTS

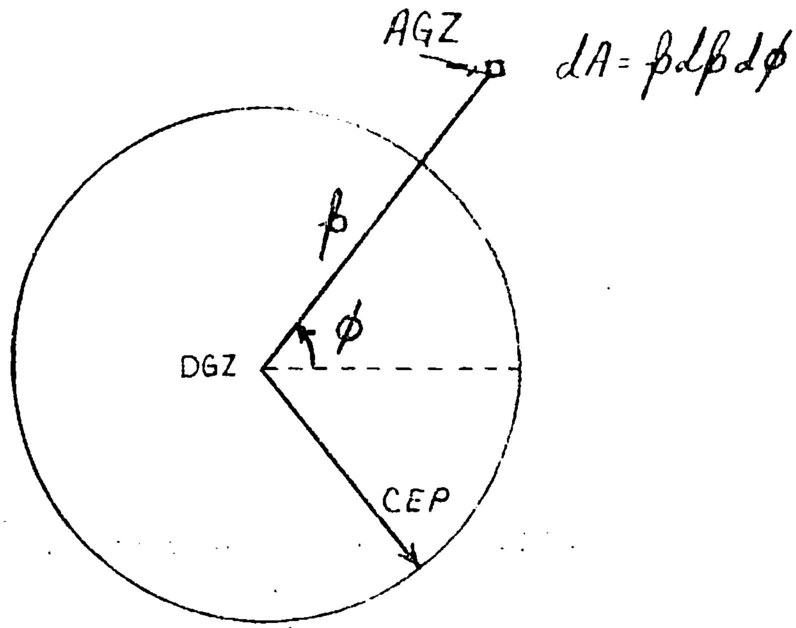
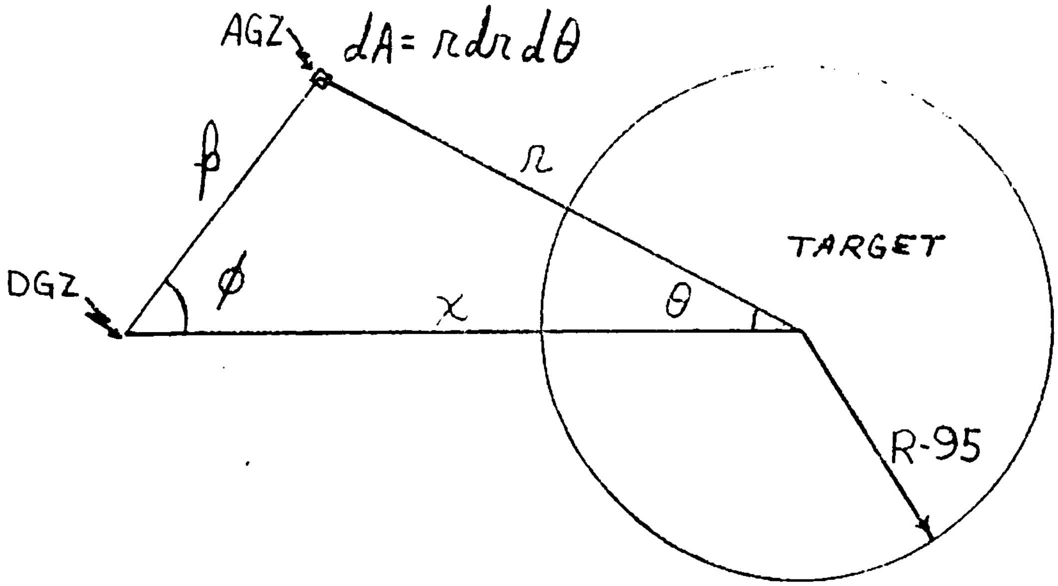


FIGURE 2 - DGZ-TARGET COORDINATE SYSTEM



Variables have not yet been standardized although now the coordinate system has been established. The parameter, σ, of f(r) is the factor involved. Therefore, for point targets:

$$\sigma = \sigma_w$$

For area targets, since both the weapon delivery error and the target distribution are circular normal, so is their joint distribution. Furthermore, since their standard deviations may be assumed independent, a useful covariance relationship is used:

$$\sigma^2 = \sigma_w^2 + \sigma_t^2$$

$$\sigma = \frac{\sqrt{(CEP)^2 + 0.231\,(R\text{-}95)^2}}{1.1774}$$

The numerator has been called an "adjusted CEP", (CEP_a).

Standardization of variables then occurs with division by :

$$R = \left(\frac{r}{\sigma}\right)$$

$$X = \left(\frac{x}{\sigma}\right)$$

$$W = \left(\frac{w}{\sigma}\right)$$

Henceforth the capitalized variables represent standardized ones as in the above equa~~t~~lities.

From the circular normal distribution of impact points it is easy to show that probability of having a weapon land further than 4σ from the DGZ is $\leq .00005$. Therefore, it should be obvious that:

$\sigma = \frac{CEP}{1.177}$

$$\int_0^\infty f(r)\,dr \simeq \int_A^B f(R)\,dR$$

since

$$f(R) \simeq 0 \qquad (A > R > B)$$

One limit for B is:

$$B = X + 4 \qquad X = \left(\frac{x}{\sigma}\right)$$

The limit for A, since A cannot be negative, is the larger of:

$$A = 0$$

$$A = X - 4$$

Sometimes, however, the weapon radius is sufficiently small so that $P_d(r)$ goes to zero before $R = X + 4$. In fact when:

$$r \geq 1.06 W e^{2.86\,\sigma_d} \quad ; \quad P_d(r) \leq 0.0005$$

so the limit for B is the smaller of:

$$B = X + 4$$

$$B = 1.06 W e^{2.86\,\sigma_d}$$

The mathematical solution to

$$\int_0^B f(R)\,dR$$

is a Gauss-Legendre 10-point quadrature formula:

$$\frac{B-A}{2} \sum_{i=1}^{i=10} k_i \, f'(R_i) \, P_d(R_i)$$

where

$$R_i = 0.5 \left[(B-A) q_i + B + A \right]$$

and where the q_i symmetry coefficients and k_i weighting factors are defined below:

GAUSS-LEGENDRE QUADRATURE

i	q_i	k_i
1	.9739065285	.066671344
2	.8650633667	.1494513492
3	.6794095683	.2190863625
4	.4333953941	.2692667193
5	.1488743390	.2955242247
6	-.1488743390	.2955242247
7	-.4333953941	.2692667193
8	-.6794095683	.2190863625
9	-.8650633667	.1494513492
10	-.9739065285	.066671344

The quadrature formula serves to locate ten AGZs about the DGZ so that each of the ten fall within the boundary conditions for probability of damage greater than zero. These AGZs are located at a distance, R_i, <u>from</u> <u>the</u> <u>target</u> in standardized dimensional units. The very last term in f(r):

$$\frac{1}{2\pi\sigma^2}\int_0^{2\pi} e^{\frac{r x \cos\theta}{\sigma^2}}\, d\theta$$

is a modified Bessel function.

After determining the value of σ to standardize variables and evaluating the appropriate limits for A and B, the quadrature formula is applied over the index from i = 1 to i = 10 evaluating completely $f(R_i)$ before applying the k_i weighting factor.

For each R_i, the Z_i is evaluated as:

$$Z_i = \frac{1}{\beta} \ln\left(\frac{W e^{-\beta^2}}{R_i}\right)$$

and $P_d(R_i)$ is easily calculated from:

$$P_d(R_i) = \frac{1}{2} + \frac{1}{2} \cdot \frac{|Z_i|}{Z_i}\, \mathrm{erf}\left(\frac{|Z_i|}{\sqrt{2}}\right)$$

Here, erf, is the error function which is given by:

$$\mathrm{erf}(C) = 1 - \frac{1}{(1 + a_1 C + a_2 C^2 + a_3 C^3 + a_4 C^4 + a_5 C^5 + a_6 C^6)^{16}}$$

$a_1 = 0.07052308$ $\quad$ $a_4 = 0.00015201$

$a_2 = 0.04228201$ $\quad$ $a_5 = 0.00027657$

$a_3 = 0.0092\cdot053$ $\quad$ $a_6 = 0.00004306$

However, certain additional conditions apply:

$$\text{If } Z_i > 3.87 \qquad P_d(R_i) = 1.00$$

$$\text{If } Z_i < -3.87 \qquad P_d(R_i) = 0.00$$

$$\text{If } |Z_i| < 5\times10^{-6} \qquad P_d(R_i) = 0.50$$

The next step is to evaluate the remaining part of f(r):

$$f'(r_i) = r_i\, e^{-\frac{1}{2\sigma^2}(r_i^2+x^2)} \cdot \frac{1}{2\pi\sigma^2}\int_0^{2\pi} e^{\frac{r_i x\cos\theta}{\sigma^2}}\, d\theta$$

which is now, in standardized units:

$$f'(R_i) = R_i\, e^{-\frac{1}{2}(R_i^2+X^2)} \cdot \frac{1}{2\pi}\int_0^{2\pi} e^{R_i X\cos\theta}\, d\theta$$

A new standardized variable is defined:

$$H_i = XR_i$$

If $X = 0$, obviously all $H_i = 0$ and the previous expression reduces to:

$$f'(R_i) = R_i\, e^{-\frac{R_i^2}{2}}$$

If $H_i > 3.75$, then G is set equal to $\frac{3.75}{H_i}$ and the solution is:

$$f'(R_i) = \frac{R_i\, e^{-\frac{(X-R_i)^2}{2}} f_1(G)}{\sqrt{H_i}}$$

where

$$f_1(G) = b_0 + b_1 G + b_2 G^2 + b_3 G^3 + b_4 G^4 + \cdots + b_8 G^8$$

$b_0 = +0.39894228$

$b_1 = +0.01328592$

$b_2 = +0.00225319$

$b_3 = -0.00157565$

$b_4 = +0.00916281$

$b_5 = -0.02057706$

$b_6 = +0.02635537$

$b_7 = -0.01647633$

$b_8 = +0.00392377$

This function, $f_1(G)$, is equal to:

$$\frac{\sqrt{XR_i}}{2\pi X R_i} \int_0^{2\pi} e^{XR_i \cos\theta}\, d\theta$$

If $H_i \leq 3.75$, then G is set equal to

$$G = \left(\frac{H_i}{3.75}\right)^2$$

and the solution is:

$$R_i \, e^{-\frac{1}{2}(R_i^2 + X^2)} f_2(G)$$

where:

$$f_2(G) = 1 + c_1 G + c_2 G^2 + c_3 G^3 + c_4 G^4 \cdots + c_6 G^6$$

$c_1 = 3.5156229$ $c_4 = 0.2659732$

$c_2 = 3.0899424$ $c_5 = 0.0360768$

$c_3 = 1.2067492$ $c_6 = 0.0045813$

Here, the function, $f_2(G)$ is:

$$\frac{1}{2\pi}\int_0^{2\pi} e^{X R_i \cos\theta} \, d\theta$$

As noted previously, the above equations are solved for each of the ten values for R_i by calculating $f'(R_i)$. Each $f'(R_i)$ is weighted by the Gauss-Legendre quadrature weighting coefficients, k_i, multiplied by $P_d(R_i)$ and summed. That sum is then multiplied by the factor, $\frac{(B - A)}{2}$, to give the <u>probability on the average</u>, of causing <u>at least</u> the level of damage <u>specified</u> by the target vulnerability number (VNTK).

IMPORTANT!

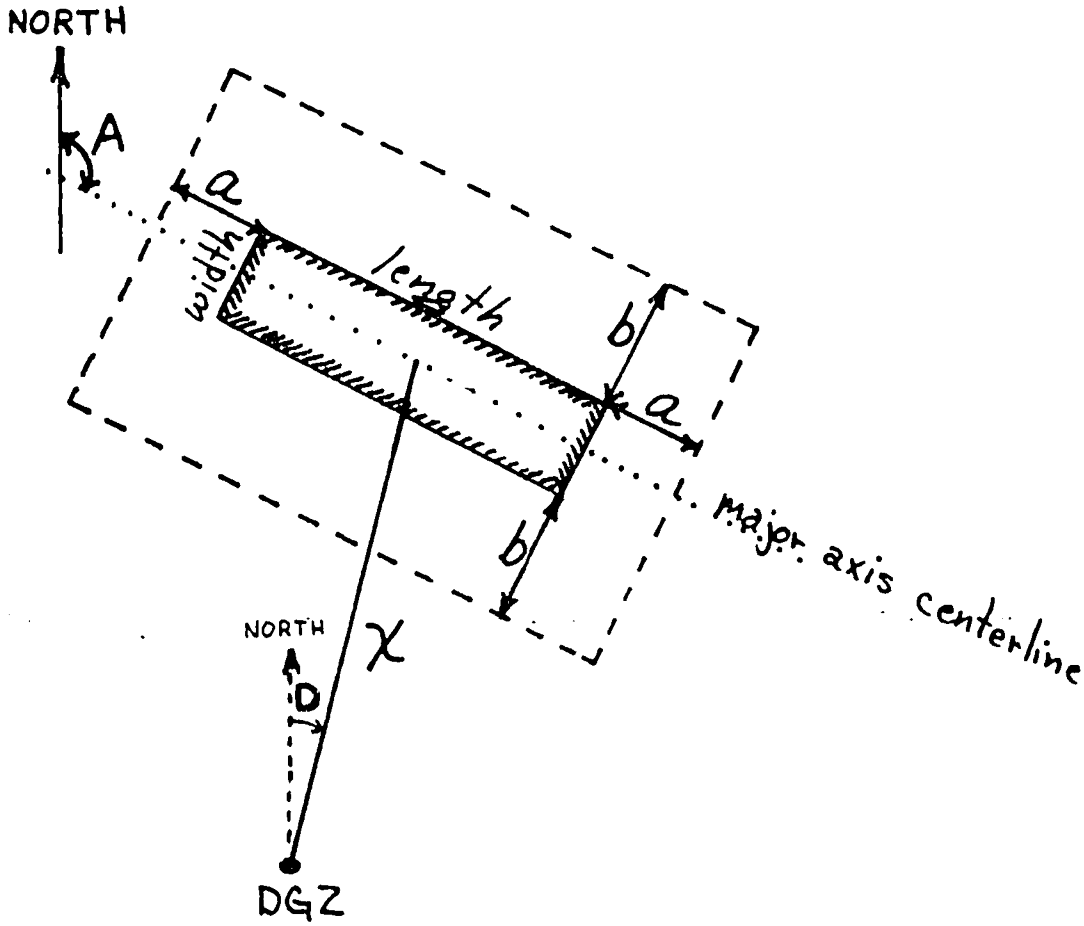
THIS PAGE INTENTIONALLY LEFT BLANK

B. Equivalent Target Area.

Certain special types of targets are treated by the equivalent target area (ETA) approximation. Bridges, dams, locks, etc. are examples of such targets which, in general, are rectangular in shape and have large length to width ratios. The damage objective is considered satisfied on any part of the ETA target. The approximation then equates the probability of placing a weapon in the target area to the probability of achieving the desired level of damage to the target.

The target area is made "equivalent" to the weapon considered and the desired damage by adjusting the target's actual length and width parameters, coded in its VNTK, with weapon radius factors. Because some targets vary in effects sensitivity along the length axis compared to the width axis, the adjustment to the target area is not always the same to each axis. The new equivalent target area is defined by adding to the width a distance equal to two width weapon radii and to the length a distance of two length weapon radii. Figure 3 illustrates the construction of the ETA. The standard deviations for the target axes are given by:

FIGURE 3. EQUIVALENT TARGET AREA CONSTRUCTION



$\measuredangle$ A : Orientation Angle (ORIEN) (for $\chi > 0$)

$\measuredangle$ D : DGZ Azimuth (AZMTH) (for $\chi > 0$)

a = length weapon radius (WR_ℓ)

b = width weapon radius (WR_w)

$$\sigma_w = \frac{\sqrt{(CEP)^2 + (1.1774\, \sigma_d\, WR_w)^2}}{1.1774}$$

$$\sigma_\ell = \frac{\sqrt{(CEP)^2 + (1.1774\, \sigma_d\, WR_\ell)^2}}{1.1774}$$

When the DGZ is not located at the target center, it is necessary to know the azimuth of the line from the DGZ to target center and the orientation azimuth for the target in order to fix a rectangular coordinate system commensurate with the problem. Otherwise, the principles remain the same.

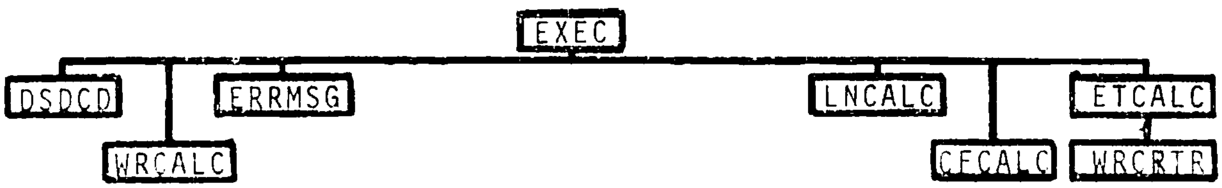
THIS PAGE INTENTIONALLY LEFT BLANK

III. JSTPS SUBROUTINE PDCALC

A. General.

The figure below outlines the basic structure of PDCALC with its internal subroutines.

PDCALC



Each subroutine performs specific operations as later described.

B. EXEC - Input/Output.

The EXEC subroutine accepts input data through a call statement, directs computational operations by interpreting the value in an input flag field (IFLG) and returns output data to the main calling program. The call statement is:

VN T ~K=10 kT ft R95 (n.mi)

CALL PDCALC (IV, JT, KF, YLD, HOB1, R95, CEP, D, WR, POD, IFLG, AZMTH).

The arguments in the call statement are:

IV	VN portion of VNTK for target. Integer format. (INPUT)
JT	T portion of VNTK for target. Alpha/integer format. (INPUT)
KF	K portion of VNTK for target. Integer format. (INPUT)
YLD	Weapon yield in kilotons. Floating point. (INPUT)

HOB1 Weapon height of burst in feet. Floating point. (INPUT).

R95 Radius in nautical miles for circular normal area target. Floating point. (INPUT)

(or):

Orientation angle in degrees for ETA target when D≠0. (Carried as R95 in NTB). Floating point. (INPUT).

CEP Weapon circular error probable in feet. Floating point. (INPUT).

target offset distance

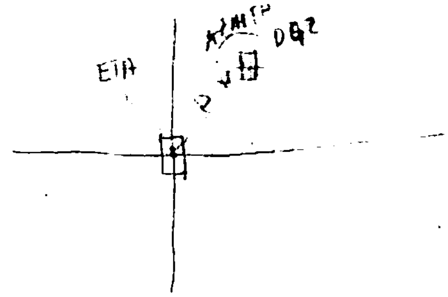
D Distance in nautical miles from DGZ to target. Floating point. (INPUT) (OUTPUT for IFLG + 5 or 6).

WR Weapon effects ? radius in feet. Floating point. (OUTPUT) (INPUT for IFLG = 10).

POD Probability of Damage. Floating point. (OUTPUT) (INPUT for IFLG = 5 or 6).

IFLG Integer format flag which directs PDCALC to perform specified functions. (Refer to Appendix 1 for complete description). (INPUT).

AZMTH Azimuth in degrees from DGZ to ETA target when D≠0. Floating point. (INPUT).



C. DSDCD - Damage Sigma.

This subroutine decodes the input VNTK and assigns the target damage sigma (σ_d). Specific definitions of the T field in VNTK related to damage sigma values are at reference 3, page 7, while specific ETA codings in terms of VNTK are at pages 254, 292, and 321. Integer values of 1 and 2 in the T field are interpreted as "P" and "Q," respectively; a 3 is interpreted as "X" for personnel vulnerability. With IFLG = 8 or 9, any damage sigma greater than zero and less than one can be input through the POD field for special excursions.

D. WRCALC - Weapon Effect Radius.

The weapon Effect radius is computed in this subroutine based on the weapon input data and the target VNTK. The adjusted vulnerability number is also computed here. The applicable overpressure effect, dynamic pressure effect, personnel fatality, or personnel incapacitating casualty is determined through the decoded VNTK. For personnel vulnerability, the K field in VNTK matches the personnel environment in reference 2 as follows:

K	REF 2 FIGURE	K	REF 2 FIGURE
1	III-1	6	III-6
2	III-2	7	III-14
3	III-3	8	III-15
4	III-4	9	III-16
5	III-5		

Recent changes to personnel vulnerability have been incorporated. (Reference 4)

An automatic feature of this subroutine does not permit attempted evaluations for heights of burst beyond the so-called "knee" of the weapon radius curves. An error message will be returned in these cases.

E. <u>CFCALC - Circular Coverage Function</u>

This routine computes probability of damage in the methodology of the Hastings approximation to the circular coverage function only if so directed by input IFLG = 11. Additionally, it is automatically used when computing an offset distance based on an input POD of less than 0.50 (IFLG = 8 or 9). In the latter case, it is recommended that POD be reevaluated by LNCALC using the computed offset distance and IFLG = 2 in the interest of accuracy and consistency.

F. <u>LNCALC - Cumulative Log Normal Function</u>.

As noted earlier, the JSTPS standardizes reporting POD based on the CLN methodology. The general procedure has already been briefly described.

G. <u>ETCALC - Equivalent Target Area</u>.

Equivalent Target Area problems are evaluated in this routine. The only options permitted with ETA targets are IFLG = 1 or 2. Data arrays store applicable target parameters tabulated in reference 3.

H. WRCRTR - Crater Radius.

This routine is called by ETCALC for a weapon crater radius when such is demanded due to the nature of a target.

I. ERRMSG - Error Messages.

Error messages are printed by this routine. Sources of errors may be inadvertently erroneous input data or, for example, too high a height of burst for a given target. Details are listed in Appendix 2.

Error messages 2 and 10 (Appendix 2) can be suppressed, if necessary, by adding 100 to the input IFLG value with each call by the main program.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. COMMENTS.

A. PROBABILITY OF DAMAGE INTERPRETATION.

Probability of damage relates to at least the level of damage specified by a vulnerability number. The level of damage, for example, might be moderate structural, severe structural, or fatality. If the vulnerability number corresponds to moderate structural damage, then the numerical value for probability of damage incorporates all levels of damage at and higher than moderate. The term (1-PD) is not a probability of survival; it is the probability of achieving all levels of damage less than that specified. With a moderate structural damage VNTK (1-PD) is the probability for light damage including survival or escaping all damage. Similarly, when treating personnel vulnerability, the probability of causing at least incapacitating casualties is a numerical figure accounting for these casualties and also fatalities. To separately identify casualty and fatality probabilities requires two computations for each "target" - one for at least fatalities (e.g., VNTK = 99X1) and one for at least casualties (99X2). The difference in the two probability values is for casualties.

Probability of damage numerical values reflect averages. It should not be interpreted for the prediction of the outcome of any single nuclear explosion. The numerical result represents on a percentage basis the number of times that the specified outcome should, on the average, occur if the event were identically repeated a number of times. In terms of specified damage levels, probability of damage theory recognizes only two possible outcomes: (1) achieving the specified level of damage or higher, or, (2) achieving less than the specified level of damage. A target in its entirety, point or area, is either damaged at least as specified or damaged at any level lower than specified. This important fact is frequently overlooked for area type targets containing some number of indistinguishable elements. Probability of damage value can be used effectively as relative indicators. For example, the location of a DGZ can be determined for optimum damage results or yields and heights of burst can be juggled for the same purpose. Comparisons between nuclear plans can be made with probability of damage values but only with caution. The assumptions incorporated into the theory must be borne in mind.

B. UNCERTAINTIES.

Aside from inaccuracies related to target locations, distribution and composition of the elements comprising area targets etc., some statements can be made regarding accuracy bands from measured empirical data.

(1) Warhead yields: Techniques used in test programs to certify the test device yield are accurate to

no better than a certain classified percentage figure. That figure varies with the fission - fusion yield ratio, however.

(2). Warhead Age: Weapons may be boosted or unboosted. Yield of boosted weapons vary with the age variations of limited life component boosting material.

(3). Pressure - Range Data: Overpressure range data is reported reliable to ±20%; dynamic pressure data to ±40%.

(4). Damage Sigma: Values of 0.20 are consistently used at the present time for P targets where the actual value varies from 0.10 to 0.30. For Q targets, 0.30 is used where it varies from 0.20 to 0.40.

(5). K-Factor: The K factor in VNTK is rounded to the nearest integer and can give a relative weapon radius error as high as ±20%.

(6). Vulnerability Number: The VN portion of VNTK is rounded to integer values. This gives a ±5% relative error in weapon radius.

(7). Height of Burst: Errors in height of burst can cause considerable variance in weapon radius. It is for this reason alone that attempts to optimize effects at or near the "optimum" weapon radius, or "knee" of weapon radii curves should be avoided.

The relative error in weapon radius from combined independent error sources can be considerable.
A conservative estimate places a lower limit on weapon radius error at ±10%. The corresponding probability of damage variance is calculable through these limits and is also dependent on specific targeting cases.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX 1

IFLG INPUT DESCRIPTION

IFLG VALUE	SEE NOTE	
1		Compute LNCALC POD (Maximum = .990)
2	(1)	Compute LNCALC POD (Maximum = .999)
3, 4		Compute weapon radius.
5, 6	(2)	Compute offset distance.
7	(3)	Compute fatality and casualty effect.
8		Compute LNCALC POD (Special case).
9		Compute LNCALC POD (Special case).
10		Compute LNCALC POD (Special case).
11		Compute CFCALC POD (Special case).
12	(4)	Compute floor space POD and fatality.

VALUE	INPUT (FIELD)	OUTPUT (FIELD)
1	D	POD, WR
2	D	POD, WR
3, 4	D	WR
5, 6	POD	D, WR
7	D	Fatality (as POD) Casuality (as WR)
8	Damage Sigma (thru POD)	POD
9	Damage Sigma (thru POD) and WR	POD
10	WR	POD
11	D	POD, WR
12	D	Floor Space POD (as POD) Fatality (as WR)

(1) Normal IFLG value for structural damage.
(2) Not allowed for ETA targets.
(3) Allowed only for JT = X or 3.
(4) Normal IFLG value for personnel effects based on National Target Base (NTB) P-95 VNTK as P or Q type target. Not allowed for JT = X or 3.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX 2

ERROR MESSAGES

IERR VALUE	MESSAGE	SOURCE
1	YOU CANNOT ACHIEVE DESIRED POD WITH THIS WEAPON	CFCALC LNCALC
2	VN (IV) IS TOO LARGE FOR AVAILABLE DATA CURVES	WRCALC
3	SHOB > 900 FEET	WRCALC
4	THE ONLY OPTIONS AVAILABLE W/ETA TARGETS ARE IFLG = 1 OR 2	EXEC
5	JT MUST CONTAIN AN X OR 3 WHEN USING IFLG = 7	EXEC
6	CHECK VALUE IN KF. IT MUST BE LESS THAN 10.	DSDCD
7	CHECK VALUE IN KF. IT MUST BE > OR = ZERO.	DSDCD
8	INVALID VALUE IN T PORTION OF VNTK FOR ETA-TYPE TARGETS (JT)	ETCALC
9	INVALID VALUE IN T PORTION OF VNTK (JT)	DSDCD
10	VN IS TOO LARGE FOR SHOB > OR = 900 FEET	WRCALC
11	FOR IFLG = 12 YOU MUST INPUT A P OR A Q TYPE TARGET	EXEC
12	FOR JT = Z, A CONTACT BURST IS REQUIRED	DSDCD

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX 3

FORTRAN LISTING

SUBROUTINE PDCALC

A sequence number for each card in the PDCALC deck appears as an 8-digit number in the FORTRAN listing. The 8-digit field corresponds to card columns 73 through 80.

The sequence numbers provide an adequate means for collation. These numbers also provide a convenient basis for referencing any communication between users and the JSTPS.

THIS PAGE INTENTIONALLY LEFT BLANK

AVN = Adjusted VN

DGZ = Target Offset Distance

ETA = Equivalent Target Area

VNTK = Vulnerability [illegible] for Decoding

CCF = Circular Coverage Function

CLN = Cumulative log normal

w = weapon radius

σ_d = target damage sigma

x = distance from a DGZ (or AGZ with CEP = 0)

PD CALC ≡ PROB. DAMAGE CALC.

```
C*********************************************************************
C
      SUBROUTINE PDCALC(IV,JT,KF,YLD,HOB1,R95,CEP,D,WR,POD,IFLG,AZMTH)
C
C*********************************************************************
C
C
C  PDCALC IS A SUBROUTINE IN FORTRAN 4 WHICH CALCULATES THE AVERAGE
C         PROBABILITY OF ACHIEVING AT LEAST THE LEVEL OF DAMAGE,
C         SPECIFIED BY A VULNERABILITY NUMBER, TO AN INSTALLATION
C         USING A SPECIFIC WEAPON TARGETTED AGAINST AN EXPLICIT
C         DGZ LOCATION. OPTIONAL CALCULATIONS ARE AVAILABLE THROUGH
C         IFLG CONTROL.
C
C
C
      CONTINUE
C
C    THE SUBROUTINE ARGUMENTS HAVE THE FOLLOWING MEANINGS:
C
C          IV = AN INTEGER NUMBER DESCRIBING TARGET HARDNESS OR
C               TARGET DIMENSIONS (ETA). INDICATES VULNERABILITY
C               NUMBER (VN OF VNTK).
      CONTINUE
C          JT = 'T' PORTION OF VNTK. CAN BE 1, 2, OR 3 IN ADDITION
C               TO ALPHABETICS DEFINED IN TDI HANDBOOK.
C          KF = 'K' PORTION OF VNTK WHICH IS AN INTEGER NUMBER FROM
C               0 TO 9. FOR P AND Q TYPE TARGETS THIS DENOTES TARGET
C               RESPONSE TO SHOCK DURATION. FOR POPULATION EFFECTS IT
C               DENOTES THE DOMINANT STRUCTURE IN THE AREA.
      CONTINUE
C         YLD = YIELD OF WEAPON IN KILOTONS
C
C        HOB1 = ACTUAL HEIGHT OF BURST OF THE WEAPON
C
C         R95 = RADIUS IN NAUTICAL MILES OF A CIRCLE ENCOMPASSING
C               95 PER CENT OF THE CIRCULAR NORMAL TARGET AREA
C               FOR ETA TARGETS, R95*10 = ORIENTATION OF THE TARGET
C               IN DEGREES.
      CONTINUE
C         CEP = CIRCULAR ERROR PROBABLE OF THE SPECIFIED WEAPON SYSTEM
C
C           D = DISTANCE IN NAUTICAL MILES FROM DGZ TO TARGET.
C
C          WR = WEAPON RADIUS IN FEET.
C
C         POD = PROBABILITY OF ACHIEVING THE SPECIFIED LEVEL OF DAMAGE
C               AGAINST THE GIVEN TARGET WITH THE GIVEN WEAPON.
C
      CONTINUE
C        IFLG = THERE ARE DIFFERENT RESULTS THAT PDCALC CAN PRODUCE.
```

```
C               THE OUTPUT CREATED IS CONTROLLED BY GIVING IFLG THE
C               FOLLOWING VALUES:
      CONTINUE
C               1 = PRODUCE POD UP TO VALUE OF .990. D MUST BE INPUT
C
C               2 = PRODUCE POD UP TO VALUE OF .999. D MUST BE INPUT
C
C               3=
C                   PRODUCE WEAPON RADIUS.
C               4 =
      CONTINUE
C               5 =
C                   PRODUCE D, THE MAXIMUM DISTANCE AT WHICH A GIVEN POD
C                   CAN BE ACHIEVED.  POD MUST BE INPUT.
C               6 =
C
C               7 = PRODUCE FATALITY POD AND CASUALTY POD.
C                   THESE VALUES ARE RETURNED IN POD AND WR
C                   VARIABLES, RESPECTIVELY. D MUST BE INPUT.
      CONTINUE
C               8 = DAMAGE SIGMA IS INPUT THROUGH POD VARIABLE. POD IS
C                   OUTPUT.  (D IS INPUT)
C
C               9 = DAMAGE SIGMA AND WEAPON RADIUS ARE INPUT. POD IS
C                   OUTPUT.  (D IS INPUT)
C
C              10 = WR INPUT. POD IS OUTPUT. (D IS INPUT)
C
C              11 = CALCULATE POD UP TO .999 USING CCF FUNCTION INSTEAD
C                   OF THE CLN FUNCTION FOR IFLG=1,2.
      CONTINUE
C              12 = PRODUCE POD AND FATALITY POD. FATALITY POD IS
C                   RETURNED IN THE WR VARIABLE. D MUST BE INPUT.
      CONTINUE
C       AZMTH = AZIMUTH IN DEGREES FROM DGZ TO TARGET.
C
C            *** OPTION ***
C     IF IFLG > 100 THEN ERROR MESSAGES 2 AND 10 ARE SUPPRESSED.
C     100 IS SUBTRACTED FROM IFLG IN THIS CASE. PROCESSING THEN CONTINUES
C     AS BEFORE.   IFLG IS NOT RETURNED TO ITS ORIGINAL VALUE.  THIS
C     MEANS THAT IFLG MUST BE RESET BEFORE EACH CALL TO PDCALC  IF YOU
C     WANT TO USE THIS OPTION.
C
C         IERR IS A FLAG FOR FINDING PROBLEMS IN THE INPUT DATA IF
C              THEY EXIST.
C
C
C               FIND DAMAGE SIGMA---- DSDCD RETURNS JJT=1 FOR P,=2 FOR
C                                     Q, =3 FOR PPLN, =4,5,6 FOR ETA
      RADIUS = 10000000.
      ANGL1 = .0
      ANGL2 = 360.
```

```
      IFGFLG = 0
      IF  (IFLG.GT.100)  IFGFLG = 1
      IF  (IFLG.GT.100)  IFLG = IFLG - 100
      LF = KF
          LP = KF / 2
          IF(((2*LP.EQ.LF).AND.(IFLG.EQ.7)).AND.(KF.GT.0)) LF = LF - 1
          IFLH = IFLG
          IF (IFLG.GE.11) IFLH = 2
      IERR = 0
C***********************************************************************
      CALL DSDCD (JT, LF, YLD, HOB1, DSIG, JJT, POD, IFLH, IERR)
C***********************************************************************
      IF (IERR .NE.0) GO TO 990
C             IF IFLG = 8,9, DSDCD INSERTS POD'S VALUE INTO DSIG.
C
C       IF ETA (JJT=4) COMPUTE POD FOR IT, OTHERWISE SKIP THIS PART.
C
      IF (JJT.LT.4) GO TO 100
C
C             CHECK IFLG  FOR VALID VALUE TO USE WITH ETA
      IF (IFLG.LE.2) GO TO 50
C
      IERR = 4
      GO TO 990
C
 50   CONTINUE
C
C  IF D > 0 & YOU HAVE A LOCK OR DAM GO TO 992 & SET POD = 0.0
C
      IF  ((D.GT..001) .AND. (JJT.NE.4)) GO TO 992
C
C***********************************************************************
      CALL  ETCALC (IV,JT,KF,YLD,CEP,HOB1,R95,AZMTH,D,POD,IERR)
C***********************************************************************
      IF (IERR .NE.0) GO TO 990
      RETURN
C
 100  CONTINUE
C
C            CHECK FOR WR INPUT. IF NOT AN INPUT, CALL WRCALC.
C
C***********************************************************************
      IF (IFLH.LT.9) CALL WRCALC(YLD,HOB1,IV,JJT,LF,DSIG,WR,IERR)
C***********************************************************************
C
      IF (IERR.NE.0) GO TO 990
C            CHECK TO SEE IF WR IS ONLY DESIRED OUTPUT. IF SO, RETURN
C
      IF((IFLG.NE.3) .AND. (IFLG.NE.4)) GO TO 110
C
      RETURN
C
```

```
 110   CONTINUE
      IF (IFLG.EQ.7) IFLH = 2
C
C        CALL THE APPROPRIATE DAMAGE ROUTINE, DEPENDING ON DSIG VALUE.
C                 IF IFLG = 11 DO THE OPPOSITE
C
C*********************************************************************
      IF ((((IFLG.EQ.5) .OR. (IFLG.EQ.6)) .AND. (POD.GT..50))  .OR.
     A   (((IFLG.NE.5) .AND. (IFLG.NE.6)) .AND. (IFLG.NE.11)))
     B    CALL LNCALC (CEP, DSIG, WR, R95, POD, D, IFLH, RADIUS,
     A                 ANGL1, ANGL2, IERR)
      IF ((((IFLG.EQ.5) .OR. (IFLG.EQ.6)) .AND. (POD.LE..50))  .OR.
     A   (IFLG.EQ.11))
     B   CALL CFCALC (CEP, DSIG, WR, R95, POD, D, IFLH, IERR)
C*********************************************************************
      IF (IERR.NE.0) GO TO 990
C
C             IF YOU WANT NOTHING ELSE, RETURN CONTROL TO CALLING PROG.
C
      IF (IFLG.EQ.7) GO TO 140
      IF (IFLG.EQ.12)GO TO 150
 139  CONTINUE
      RETURN
C
 140  CONTINUE
C
C          MAKE SURE THAT FOR IFLG=7, YOU HAVE 99X TARGET.
C
      IF (KF.EQ.9) GO TO 139
      IF (JJT.EQ.3) GO TO 160
        IERR = 5
        GO TO 990
C
C          MAKE SURE THAT FOR IFLG=12, YOU HAVE P OR Q TARGET.
C
 150  LF = 0
      LT = 3
      IF (JJT.EQ.2) LF =2
      IF (JJT.LT.3) GO TO 165
      IERR = 11
      GO TO990
C
 160  CONTINUE
C
C               THIS IS THE 2ND POD CALCULATION AREA
C
      LT = 3
 165  P1 = POD
      LF = LF + 1
C
C*********************************************************************
      CALL DSOCD (LT, LF, YLD, HOB1, DSIG, JJT, POD, IFLH, IERR)
```

```
C*********************************************************************
      IF (IERR.NE.0) GO TO 990
C
C*********************************************************************
      CALL WRCALC(YLD, HOB1, IV, JJT, LF, DSIG, WR, IERR)
C*********************************************************************
C
      IF (IERR.NE.0) GO TO 990
C
C*********************************************************************
      CALL LNCALC (CEP, DSIG, WR, R95, POD, D, IFLH, RADIUS,
     A                    ANGL1, ANGL2, IERR)
C*********************************************************************
C
      IF (IERR.NE.0) GO TO 990
C
      P2 = POD
          WR = POD
      POD = P1
      RETURN
C
C
 990  CONTINUE
      IF ((IFGFLG.EQ.1) .AND. ((IERR.EQ.2) .OR. (IERR.EQ.10)))GO TO 991
      CALL ERRMSG (IERR,IV,JT,KF,YLD,CEP,HOB1,R95,D,WR,POD,IFLG)
      RETURN
C
 991  D = .0
      WR = .0
 992  POD = 0.0
      RETURN
      END
C*********************************************************************
C
      SUBROUTINE DSDCD (JT, KF, YLD, HOB1, DSIG, JJT, POD, IFLG, IERR)
C
C*********************************************************************
C
      DIMENSION TYD(4,2,9), SIGMA(5,2,9), NBRYDS(2,9)
      DIMENSION HT(23),IT(23)
      DATA  NP, NQ, NX /'P','Q','X'/
      DATA HT, IT / 11*.0, .1,.3,.4,.5,.1,.2,.4,.5, 4*.0,
      DATA HT, IT / 11*.0, .1,.3,.4,.5,.1,.2,.4,.5, 3*.0, .3,
     A       'A','B','C','D','E','F','G','H','I','J','K','L','M','N',
     B       'O','R','S','T','U','V','W','Y','Z'/
C
C         ARRAY TYD CONTAINS THE YIELD VALUES AT WHICH DAMAGE SIGMA
C               CHANGES FOR POPULATION TYPE TARGETS.
C
C         ARRAY SIGMA CONTAINS THE DAMAGE SIGMA VALUES FOR POPULATION
C               TYPE TARGETS.
C
```

DSDCD = DAMAGE SIGMA DE CODE DATA

*   REPLACED*
*   REPLACEMENT*

```
      CONTINUE
      DATA  TYD
     A/40.,200.,.0,.0,         10.,40.,.0,.0,
     B 8.,40.,.0,.0,           4.,8.,100.,.0,
     C 80.,.0,.0,.0,           30.,.0,.0,.0,
     D 4.,30.,70.,.0,          4.,8.,100.,.0,
     E 35.,250.,.0,.0,         20.,120.,.0,.0,
     F 3.,70.,.0,.0,           1.,10.,.0,.0,
     G 10.,40.,180.,.0,        5.0,10.,40.,500.,
     H 4.,13.,70.,.0,          2.,7.,25.,150.,
     I 6.,40.,.0,.0,           2.4,10.,150.,.0/
C
      DATA SIGMA
     A/.2,.3,.4,.0,.0,        .2,.3,.4,.0,.0,
     B .3,.4,.5,.0,.0,        .3,.4,.5,.5,.0,
     C .3,.3,.0,.0,.0,        .3,.3,.0,.0,.0,
     D .3,.3,.4,.4,.0,        .3,.3,.4,.4,.0,
     E .5,.4,.4,.0,.0,        .5,.4,.4,.0,.0,
     F .5,.4,.4,.0,.0,        .5,.4,.4,.0,.0,
     G .2,.3,.4,.5,.0,        .2,.3,.4,.5,.5,
     H .2,.3,.4,.5,.0,        .2,.3,.4,.5,.5,
     I .3,.4,.5,.0,.0,        .3,.4,.5,.5,.0/
      CONTINUE
      DATA NBRYDS
     A/3,3, 3,4, 2,2, 4,4, 3,3, 3,3, 4,5, 4,5, 3,4/
C
C          DSDCD IS A SUBROUTINE WHICH DECODE THE T FIELD (JT) TO
C                DETERMINE THE APPROPRIATE DAMAGE SIGMA FOR USE
C                BY WRCALC AND THE POD SUBROUTINES.
C
      CONTINUE
C
C                JT = T PORTION OF 'VNTK'.  THIS MAY BE EITHER AN
C                     INTEGER (1,2,OR 3) OR AN ALPHABETIC
C                     CHARACTER (A - Z).  CURRENTLY, 1'='P, 2'='Q,
C                     AND 3'='X.
C                     THE EXPLANATION OF THE ALPHABETICS WILL            *         REPLACED*
C                    THE EXPLANATION OF THE ALPHABETICS IS               *      REPLACEMENT*
C                     BE INSERTED WHEN IT IS AVAILABLE.                  *         REPLACED*
C                    DOCUMENTED IN THE TDI HANDBOOK (DIA)                *      REPLACEMENT*
      CONTINUE
C                KF = K PORTION OF 'VNTK'. IT HOLDS VALUES RANGING
C                     FROM 0 TO 9. (INPUT)
C
C               YLD = WEAPON YLD IN KILOTONS. (INPUT)
C
C              HOB1 = HEIGHT OF BURST IN FEET. (INPUT)
      CONTINUE
C              DSIG = DAMAGE SIGMA FOR THE TARGET. (OUTPUT)
C
C               JJT = 1 FOR P TYPE TARGETS
C                   = 2 FOR Q TYPE TARGETS
```

```
C                        = 3 FOR POPULATION TYPE TARGETS (99X).
C                        = 4 FOR BRIDGES
C                        = 5 FOR CANAL LOCKS
C                        = 6 FOR DAMS
      CONTINUE
C                  IERR = 0 FOR NO INPUT DISCREPANCY
C                        = 6 FOR KF>9
C
C                        = 7 FOR KF < 0
C                      = 12  FOR JT = 'Z' AND HEIGHT OF BURST > 0          *          INSERTED*
      CONTINUE
      DSIG = .0
C                    CHECK FOR INVALID KF
      IF (KF.GT.9) GO TO 40
      IF (KF.LT.0) GO TO 60
C
C
      IF (JT.LT.1) GO TO 100
C
C            HERE WE KNOW THAT JT IS A NUMBER EQUAL TO 1,2,OR 3.
C
      JJT = JT
      IF ((IFLG.EQ.8).OR.(IFLG.EQ.9)) GO TO 50
C
C       GO TO THE APPROPRIATE DAMAGE SIGMA ROUTINE
C
      IF (JJT-2) 10,20,30
C                  P-TYPE TARGET
 10   DSIG = .2
      GO TO 50
C                  Q-TYPE TARGET
 20   DSIG = .3
      GO TO 50
C                  POP-TARGET
  29  JJT = 3
 30   IHB = 1
      IF (HOB1 .GT. .001)
     A   IHB = 2
C     ENDIF
      JKF = KF / 2
      JKF = JKF * 2
C
      NLO = NBRYDS(1,KF)
      NHI = NBRYDS(IHB,KF)
      NYDL = NLO - 1
      NYDH = NHI - 1
      DO 31   JS=1, NYDL
         IF (YLD .LT. TYD(JS,1,KF))
     A         NLO = NLO - 1
C        ENDIF
C     ENDDO
31           CONTINUE
```

```
      DO 32   JS=1, NYDH
         IF (YLD .LT. TYD(JS,IHB,KF))
     A       NHI = NHI - 1
C        ENDIF
C     ENDDO
32              CONTINUE
      DSIG = AMAX1(SIGMA(NLO,1,KF), SIGMA(NHI,IHB,KF))
      IF (KF .EQ. JKF)
     A   DSIG = AMIN1(SIGMA(NLO,1,KF), SIGMA(NHI,IHB,KF))
C     ENDIF
C
      GO TO 50
C
 40   IERR = 6
 50   CONTINUE
      IF ((IFLG.EQ.8).OR.(IFLG.EQ.9)) DSIG = POD
      RETURN
  60  IERR = 7
      RETURN
 100  DSIG=.0
      JJT = 0
C
      IF (JT.EQ.NP) DSIG = .2
      IF (JT.EQ.NP) JJT = 1
      IF (JT.EQ.NQ) DSIG = .3
      IF (JT.EQ.NQ) JJT = 2
      IF (JT.EQ.NX) GO TO 29
      IF (JT .EQ. IT(23)) GO TO 170                          *          INSERTED*
      IF (DSIG.LE.0) GO TO 110
      GO TO 50
 110    IF (JT.LE.IT(7)) GO TO 180
      IF((JT.GE.IT(8)) .AND. (JT.LE.IT(15))) GO TO 150
      IF((JT.LT.IT(16)).OR.(JT.GT.IT(23))) GO TO 200
      JJT = 2
C
      DO 120 J=16,23
C
      IF (JT.EQ.IT(J)) DSIG = HT(J)
C
 120  CONTINUE
C
      IF (DSIG.LE.0.) GO TO 200
      GO TO 50
C
 150  CONTINUE
      JJT = 1
C
      DO 160  J=8, 15
C
      IF (JT.EQ.IT(J)) DSIG = HT(J)
C
 160  CONTINUE
```

```
C
      IF (DSIG.LE..0) GO TO 200
      GO TO 50
C
 170  CONTINUE                                                              *   INSERTED*
C              JT = 'Z' IS A P-TARGET, DSIG OF .3, AND REQUIRES HOB = 0     *   INSERTED*
      JJT = 1                                                               *   INSERTED*
      DSIG = .3                                                             *   INSERTED*
      IF (HOB1 .LT. .99) GO TO 50                                           *   INSERTED*
      IERR = 12                                                             *   INSERTED*
      RETURN                                                                *   INSERTED*
C                                                                           *   INSERTED*
 180  CONTINUE
C
      IF (JT.EQ.IT(1)) JJT = 4
      IF (JT.EQ.IT(2)) JJT = 4
      IF (JT.EQ.IT(3)) JJT = 4
      IF (JT.EQ.IT(4)) JJT = 6
      IF (JT.EQ.IT(5)) JJT = 5
      IF (JJT.EQ.0) GO TO 200
      RETURN
C
 200  IERR = 9
      RETURN
      END
C**********************************************************************
C
      SUBROUTINE WRCALC (YLD, HOB1, IV, JJT, KF, DSIG, WR, IERR)
C
C**********************************************************************
C
C             WRCALC IS THE SUBROUTINE WHICH CALCULATES WEAPON RADIUS
C
      DIMENSION WP(8,2,11),WQ(8,10), CTB(5,2,9), XTB(5,2,9), YDTB(4,2,9)
     A         ,TYD3(9), TVNP(2,11), TVNQ(10), SHOBM(9), NBRYDS(2,9)
      DIMENSION WP1(88), WP2(88), WQ1(72), WQ2(8)
      EQUIVALENCE  (WP(1), WP1(1)),  (WP(89), WP2(1)),
     A             (WQ(1), WQ1(1)),  (WQ(73), WQ2(1))
      DATA  WP1 /
C
C        ARRAY WP CONTAINS THE VALUES FOR THE 7TH ORDER POLYNOMIAL
C            APPROXIMATION FOR WR COMPUTATIONS FOR P-TYPE TARGETS
C
C                       SHOB = 0,    AVN.LE.7.5
     A 8.206936, -.09866222,-.004270532, .00044673610, 4*.0,
C
C                       SHOB = 0     AVN.LE.1000
     B 8.263243, -.12109524, .001274266, -.000009206549, 4*.0,
C
C                       SHOB = 100, AVN.LE.7.5
     C 8.29123, -.1132939, .0003119908, 5*0.0,
C
```

WRCALC = WEAPON EFFECT RADIUS CALC.

```
C                         SHOB = 100, AVN.LE.51
     C 8.29959, -.1104334, -.00048494085, .0000658301,-.00000091680378,
     4 3*.0,
C
C                         SHOB = 200, AVN.LE.41
     E 8.395223,-.14717856, .01274489,-.002063277, .0001667591,
     5 -.00000689342, .0000001423714,-.0000000011675015,
C
C                         SHOB = 200, AVN.LE.41 (THIS MUST BE REPEATED)
     E 8.395223,-.14717856, .01274489,-.002063277, .0001667591,
     5 -.00000689342, .0000001423714,-.0000000011675015,
C
C                         SHOB = 300, AVN.LE.34
     F 8.41958,-.09982782,-.0041872797, .0005449084,-.00003758352,
     6 .000001400969,-.000000020170989, .0,
C
C                         SHOB = 300, AVN.LE.34
     F 8.41958,-.09982782,-.0041872797, .0005449084,-.00003758352,
     6 .000001400969,-.000000020170989, .0,
C
C                         SHOB = 400, AVN.LE.30
     G 8.499489,-.1096521,-.003444575, .0007261706,-.0000710905,
     7 .000003319013,-.00000005668505, .0,
C
C                         SHOB = 400, AVN.LE.30    (REPEAT)
     G 8.499489,-.1096521,-.003444575, .0007261706,-.0000710905,
     7 .000003319013,-.00000005668505, .0,
C
C                         SHOB = 500  AVN.LE.27
     H 8.525985,-.06312055,-.02562219, .005426447,-.0005926339,
     8  .00003485504,-.000001022865, .0000000114432/
      DATA  WP2     /
C
C                         SHOB = 500, AVN.LE.27    (REPEAT)
     H 8.525985,-.06312055,-.02562219, .005426447,-.0005926339,
     8  .00003485504,-.000001022865, .0000000114432,
C
C                         SHOB = 600, AVN.LE.25
     I 8.586222, -.1002711,-.009917176, .00260232,-.0003602822,
     9  .00002802515,-.000001082636, .00000001541557,
C
C                         SHOB = 600, AVN.LE.25    (REPEAT)
     I 8.586222, -.1002711,-.009917176, .00260232,-.0003602822,
     9  .00002802515,-.000001082636, .00000001541557,
C
C                         SHOB = 700, AVN.LE.22
     J 8.655962,-.1367989, .01426281,-.004092999, .0005028125,
     1 -.00002571224, .0000004379003,.0,
C
C                         SHOB = 700, AVN.LE.22   (REPEAT)
     J 8.655962,-.1367989, .01426281,-.004092999, .0005028125,
     1 -.00002571224, .0000004379003,.0,
```

```
C
C                      SHOB = 800, AVN.LE.7.5
     K 8.681285,-.1143286,-.001788869, .0001595909, 4*.0,
C
C                      SHOB = 800, AVN.LE.21
     L12.51342,-1.516344, .1769944,-.008900835, .0001400736, 3*.0,
C
C                      SHOB = 900, AVN.LE.7.5
     M 8.719654,-.1215853, .001203604,-.0001386328,4*.0,
C
C                      SHOB = 900, AVN.LE.20
     N13.47289,-1.971983, .2547267,-.014325115, .0002640371, 3*.0,
C
C                      SHOB = 120
     O 8.277177,-.099615839,-.0015938039, .00007625197, .000003217474,
     2 -.00000024832771, .000000005639562,-.0000000000462407 6,
     O 8.277177,-.099615839,-.0015938039, .00007625197, .000003217474,
     2 -.00000024832771, .000000005639562,-.00000000004624076/
C
C
C
C
      DATA WQ1    /
C
C       ARRAY WQ CONTAINS THE VALUES FOR THE 7TH ORDER POLYNOMIAL
C            APPROXIMATION FOR WR CALCULATIONS FOR Q TYPE TARGETS.
C
C                      SHOB =    0, AVN.LE.35
     A 8.315159,-.106087, .0005224,-.000313, .000032265,-.00000123227,
     1  .0000000196707,-.00000000105880,
C
C                      SHOB = 100, AVN.LE.35
     B 8.376082,-.104295,-.0012014,-.0000391136, .0000128757,
     2 -.000000497579, .00000000577257, .0,
C
C                      SHOB = 200, AVN.LE.35
     C 8.42024,-.109473, .001462288,-.0005969792, .00006697002,
     3 -.000003014946, .00000006188228,-.0000000004866633,
C
C                      SHOB = 300, AVN.LE.35
     D 8.485315,-.103139,-.0034114, .0003087,-.0000107267, .00000315662
     4,-.00000000556646, .0,
C
C                      SHOB = 400, AVN.LE.31
     E 8.576,-.103989,-.0065788, .0012382,-.0001333, .00000801387,
     5 -.000000234684, .00000000251295,
C
C                      SHOB = 500, AVN.LE.28
     F 8.6435,-.1110564,-.0041904, .0006644,-.0000776848, .00000598695,
     6 -.000000227079, .00000000300626,
C
C                      SHOB = 600, AVN.LE.26
```

```
     G 8.686697,-.116482, .0003634,-.0006169, .0000857541,-.0000487263,
     7  .0000000566402, .0,
C
C                          SHOB = 700, AVN.LE.25
     H 8.72745,-.117550, .0023483,-.0013054, .0001909,-.00001152,
     8  .0000000283079,-.0000000244704,
C
C                          SHOB = 800, AVN.LE.23
     I 8.736328,-.1151805, .0021175,-.0015218,.0002654,-.000019675,
     9  .000000618015,-.0000000720562/
       DATA WO2     /
C
C                          SHOB = 900, AVN.LE.22
     J 8.793042,-.1154885, .0001871,-.0011008, .0002357,-.0000201562,
     K  .00000069752,-.00000008874866/
C
C
C
C
       DATA TVNP    /
C
C         ARRAY TVNP CONTAINS YIELD LIMITS FOR P-TYPE TARGETS
C
     A  7.5,1000., 7.5,51., 2*41., 2*34., 2*30., 2*27., 2*25., 2*22.,
     B  7.5,21., 7.5,19., 2*1000./
C
C
       DATA TVNQ    /
C
C         ARRAY TVNQ CONTAINS YIELD LIMITS FOR Q-TYPE TARGETS
C
     A 4*35., 31., 28., 26., 25., 23., 22./
C
C  DIMENSION SUBSCRIPTS RELATIONSHIPS GIVEN FOR A REPRESENTATIVE GROUP
C    ARRAYS: W0(I,JT) I GOES FROM 1 TO 10, DEPENDING ON THE SCALED HGT
C    OF BURST.
C   CTB(N,L,K)  XTB(N,L,K)  YDTB(JJ,L,K)  TYD3(K)  SIGD(N,K)
C        K=K OF 99XK  MAX OF 9
C        L IS FOR DIFFERENT SHOB'S
C        N IS FOR ROWS OF TABLE (FOR DIFFERENT YIELD RANGES). MAX OF 3.
C        JJ IS FOR THE YIELD RANGES GIVEN IN THE TABLE. MAX OF 2 VALUES.
       DATA  NBRYDS  / 3,3, 3,4, 2,2, 4,4, 3,3, 3,3, 4,5, 4,5, 3,4 /
C  TABLE NBRYDS GIVES # OF DIFF YLD RANGES FOR DIFF K AND SHOB
C
       DATA CTB
C  TABLE CTB(N,L,K) GIVES THE VALUE OF C FOR THE WR EQUATION
C    WR=C*YLD**X  FOR 99X TARGETS
C              TABLE III-1  K=1
     A/2420.,1570.,1375.,.0,.0,   2650.,1650.,1680.,.0,.0,
C              TABLE III-2   K=2
     B 2975.,2360.,2525.,.0,.0,   3260.,2540.,2900.,4190.,.0,
C              TABLE III-3   K=3
```

```
     C 2675.,1150.,.0,.0,.0,       2950.,1375.,.0,.0,.0,
C             TABLE III-4   K=4
     D 3135.,2500.,2500.,2430.,.0,3450.,2660.,2715.,3750.,.0,
C             TABLE III-5   K=5
     E 2900.,2285.,1395.,.0,.0,    3170.,2075.,1350.,.0,.0,
C             TABLE III-6   K=6
     F 3350.,3100.,2100.,.0,.0,    3790.,3475.,3025.,.0,.0,
C             TABLE III-14   K=7
     G 2660.,2150.,1650.,1855.,.0,2910.,2300.,1930.,2010.,2620.,
C             TABLE III-15   K=8
     H 3090.,2750.,2200.,2200.,.0,3325.,2710.,2070.,2070.,3550.,
C             TABLE III-16   K=9
     I 3625.,2570.,3025.,.0,.0,    4100.,3200.,3200.,4965.,.0/
C
      DATA XTB
C  TABLE XTB(N,L,K) GIVES VALUES OF X FOR WR EQUATION
C     WR=C*YLD**X    FOR 99XK TARGETS
C             TABLE III-1   K=1
    A/.21,.34,.37,.0,.0,      .18,.39,.39,.0,.0,
C             TABLE III-2   D=2
    B .26,.38,.37,.0,.0,      .32,.50,.45,.37,.0,
C             TABLE III-O   D=3
    C .17,.36,.0,.0,.0,        .15,.38,.0,.0,.0,
C             TABLE III-4   K=4
    D .20,.34,.34,.36,.0,     .24,.43,.44,.37,.0,
C             TABLE III-5   D=5
    E .18,.24,.33,.0,.0,      .16,.28,.37,.0,.0,
C             TABLE III-6   D=6
    F .22,.26,.35,.0,.0,      .21,.29,.36,.0,.0,
C             TABLE III-14   K=7
    G .21,.31,.38,.36,.0,     .17,.34,.42,.41,.37,
C             TABLE III-15   D=8
    H .20,.28,.38,.39,.0,     .17,.39,.50,.50,.39,
C             TABLE III-16   D=9
    I .22,.40,.27,.0,.0,      .20,.46,.46,.37,.0/
C
      DATA YOTB
C  TABLE YOTB(JJ,L,K) CONTAINS YIELD RANGES USED TO DETERMINE WHERE IN
C    THE TABLE III'S TO LOOK FOR C,X FOR 99XK TARGETS.
C             TABLE III-1   D=1
    A/40.,200.,.0,.0,         10.,40.,.0,.0,
C             TABLE III-2   K=2
    B 8.,40.,.0,.0,           4.,8.,100.,.0,
C             TABLE III-3   K=3
    C 80.,.0,.0,.0,           30.,.0,.0,.0,
C             TABLE III-4   K=4
    D 4.,30.,70.,.0,          4.,8.,100.,.0,
C             TABLE III-5   K=5
    E 35.,250.,.0,.0,         20.,120.,.0,.0,
C             TABLE III-6   K=6
    F 3.,70.,.0,.0,           1.,10.,.0,.0,
C             TABLE III-14   K=7
```

```
     G 10.,40.,180.,.0,        5.0,10.,40.,500.,
C              TABLE III-15   K=8
     H 4.,13.,70.,.0,          2.,7.,25.,150.,
C              TABLE III-16   K=9
     I 8.,40.,.0,.0,           2.4,10.,150.,.0/
C
      DATA TYD3
C  TABLE TYD3(K) CONTAINS THE MINIMUM VALUE YIELD NEEDED FOR TABLE III.
     A/18.,1.2,5.,.5,50.,12.,8.5,.65,1.0/
C
      DATA SHOBM
C  TABLE SHOBM(K) CONTAINS MAX SHOB FOR WHICH EQUATIONS ARE KNOWN FOR WR
     A/800.,1000.,700.,1000.,700.,900.,900.,900.,900./
      CALL ERRSET (208,256,-1,1,1,1)
      K = KF
       JT = JJT
      VN = IV
       FK = KF
      DS2 = 1. / (1.-DSIG**2)
      IERK = 0
      YLDCU = YLD**.33333333
      ISWR = 0
      SHOB = HOBI / YLDCU
C
C         IF 99X, GO TO POPULATION WR ROUTINE. OTHERWISE, CALCULATE
C                 THE ADJUSTED VN FOR PA AND Q TYPE TARGERS.
C
      GO TO (6,7,45), JJT
C
C
C  R2 IS VN ADJUSTMENT NUMBER WHICH WILL BE IMPROVED ON BY AN
C    ALGORTTHM
6     R2=2.0
      EE = 0.50
      GO TO 10
7     R2=3.0
      EE=.33333333
C  ALGORITHM FOR IMPROVING R2
10    R1=1-(FK/10)+(((FK/10)*(20/YLD)**.33333333)*(R2**EE))
      ABDIF=R1-R2
      R2 = R1
      ABDIF = ABS(ABDIF)
      IF (ABDIF.LT..001) GO TO 15
      GO TO 10
15    CONTINUE
      IF (JT.EQ.1) GO TO 25
C  FIND WEAPON RADIUS FOR Q AND P TYPE TARGETS
C  AVN IS ADJUSTED VN
      AVN=VN+2.742*ALOG(R2)
      GO TO 26
25    AVN = VN + 5.485 * ALOG(R2)
26    CONTINUE
```

```
C
      GO TO 200
C
C             COMPUTE WR FOR 99X TARGETS
C
C  COMPUTE WR FOR 99X TARGETS
C
45    CONTINUE
      IF (SHOB.GT.SHOBM(K)) IERK = 3
      NLO = NBRYDS(1,K)
      NHI = NBRYDS(2,K)
      NYDL = NLO - 1
      NYDH = NHI - 1
      DO 50   N=1, NYDL
         IF (YLD .LT. YDTB(N,1,K))
     A      NLO = NLO - 1
C        ENDIF
C     ENDDO
50                  CONTINUE
      DO  55   N=1, NYDH
         IF (YLD .LT. YDTB(N,2,K))
     A      NHI = NHI - 1
C        ENDIF
C     ENDDO
55                 CONTINUE
C
      WRL = CTB(NLO,1,K) * (YLD ** XTB(NLO,1,K))
      WRH = CTB(NHI,2,K) * (YLD ** XTB(NHI,2,K))
      WR  = WRL + ((SHOB/SHOBM(K)) * (WRH-WRL))
 200   CONTINUE
      IF ((JJT.EQ.3) .AND. (IERK.EQ.3))WR=WRH-7.5*(SHOB-SHOBM(K))*YLDCU
      IF (JJT .EQ.3)  GO TO 400
      IF (SHOB.LE.900.) GO TO 210
C            SET ERROR MESSAGE IF SHOB > 900 FEET
C
         IERR = 3
         SHOB = 900.
      RETURN
  210 CONTINUE
C
C             CHECK FOR P OR Q
C
      IF (JJT.EQ.1) GO TO 240
C
C          COMPUTE WR FOR Q TYPE TARGETS
C
      AX = 1.10
C             COMPUTE SUBSCRIPTS FOR ENTERING COEFFICIENT TABLE
      SIL = SHOB / 100. + 1.0001
      IL = SIL
      IH = IL + 1
      IF (IH.GE.11) IH = 10
```

```
C          CHECK VN LIMIT AND SET IERR IF OUT OF BOUNDS
      IF (AVN.GT.TVNQ(IL)) IERR = 2
      IF (IERR.NE.0) GO TO 400
C
C            COMPUTE SWR BRACKETS
C
      SWRL = WQ(1,IL)
      SWRH = WQ(1,IH)
C
      DO 220 KK=2, 8
C
         SWRL = SWRL + WQ(KK,IL)*AVN**(KK-1)
         SWRH = SWRH + WQ(KK,IH)*AVN**(KK-1)
C
 220  CONTINUE
C
      GO TO 300
C
 240  CONTINUE
C
C       COMPUTE WR FOR P TYPE TARGETS
C
      AX = 1.04
C              CHECK FOR SHOB = 120 FEET
      IF ((SHOB.GT.119.).AND.(SHOB.LT.121)) GO TO 260
C
C          COMPUTE SUBSCRIPTS FOR ENTERING COEFFICIENT TABLE
      SIL = SHOB/100. + 1.0001
      IL = SIL
      IH = IL + 1
      IF (IH.GT.10) IH = 10
      JVN = 1
      JV1 = 1
C            CHECK LIMITS ON AVN. IF OUT OF BOUNDS, SET IERR.
      IF (AVN.LE.TVNP(JVN,IL))GO TO 250
      JVN = 2
      IF (AVN.LE.TVNP(JVN,IL))GO TO 250
      IERR = 2
      IF (IL.EQ.10) IERR = 10
      IF (IL.EQ.10) GO TO 400
      RETURN
C
 250  CONTINUE
C
      IF (AVN.GT.TVNP(JV1,IH))JV1 = 2
      GO TO 270
C
C                  SET SUBSCRIPTS FOR SHOB = 120
 260  IH = 11
      IL = 11
      JVN = 1
      JV1 = 1
```

```
C
 270  CONTINUE
C
C          COMPUTE SWR BRACKETS
C
      SWRL = WP(1,JVN,IL)
      SWRH = WP(1,JV1,IH)
C
      DO 280 KK=2, 8
C
         SWRL = SWRL + WP(KK,JVN,IL)*AVN**(KK-1)
         SWRH = SWRH + WP(KK,JV1,IH)*AVN**(KK-1)
C
 280  CONTINUE
C
 300  CONTINUE
C
      SWRL = EXP(SWRL)
      SWRH = EXP(SWRH)
      SIL = IL
      SLI = (SIL - 1.0001) * 100.
C
      WR =  (SWRL + ((SHOB-SLI)/100.)*(SWRH-SWRL)) * YLDCU * DS2 / AX
400   CONTINUE
      IF (WR.LE..0) WR = .0
      RETURN
      END
C*********************************************************************
C
      SUBROUTINE CFCALC (CEP, DSIG, WR, R95, POD, D, IFLG, IERR)
C
C*********************************************************************
C
C              CFCALC CALCULATES DAMAGE AND OFFSET DISTANCE WHEN DSIG IS
C                     LESS THAN OR EQUAL TO .3
      DIMENSION A(251),  B(251),  C(251),
     A           A1(179), B1(159), C1(119),
     B           A2( 72), B2( 92), C2(119),
     C                             C3( 13)
C
      EQUIVALENCE  (A(1),A1(1)), (A(180),A2(1)), (B(1),B1(1)),
     A             (B(160),B2(1)), (C(1),C1(1)), (C(120),C2(1)),
     B             (C(239),C3(1))
C
C
      DATA A1 /.12536,.12535,.12531,.12525,.12516,.12505,.12491,.12475,
     A .12456,.12435,.12411,.12385,.12356,.12325,.12292,.12256,.12218,
     B .12177,.12134,.12089,.12042,.11992,.11941,.11887,.11831,.11773,
     C .11713,.11651,.11586,.11520,.11452,.11382,.11310,.11235,.11159,
     D .11081,.11000,.10918,.10834,.10748,.10660,.10569,.10477,.10383,
     E .10287,.10190,.10090,.09988,.09886,.09782,.09677,.09570,.09462,
     F .09353,.09244,.09134,.09023,.08911,.08798,.08684,.08569,.08453,
```

CFCALC ≡ CIRCULAR COVERAGE FUNCTION CALCULATION

```
G .08336,.08219,.08101,.07983,.07864,.07747,.07629,.07512,.07397,  00087100
H .07281,.07164,.07046,.06928,.06808,.06688,.06569,.06451,.06334,  00087200
I .06217,.06101,.05984,.05866,.05750,.05632,.05515,.05400,.05286,  00087300
J .05174,.05063,.04953,.04843,.04734,.04624,.04515,.04407,.04301,  00087400
K .04197,.04095,.03996,.03898,.03800,.03702,.03604,.03507,.03411,  00087500
L .03317,.03226,.03137,.03051,.02966,.02882,.02798,.02715,.02632,  00087600
M .02551,.02472,.02395,.02322,.02250,.02180,.02111,.02042,.01974,  00087700
N .01906,.01840,.01776,.01715,.01656,.01600,.01545,.01491,.01437,  00087800
O .01383,.01330,.01278,.01228,.01180,.01133,.01089,.01046,.01004,  00087900
P .00963,.00922,.00883,.00845,.00808,.00773,.00739,.00707,.00676,  00088000
Q .00645,.00615,.00586,.00557,.00529,.00502,.00477,.00453,.00430,  00088100
R .00408,.00386,.00365,.00345,.00325,.00306,.00287,.00270,.00254,  00088200
S .00238,.00223,.00208,.00194,.00180,.00166,.00153,.00140,.00128/  00088300
 DATA A2 /.00117,.00106,.00096,.00086,.00076,.00066,.00057,.00048, 00088400
A .00040,.00032,.00025,.00018,.00012,.00006,.0,-.00006,-.00011,    00088500
B -.00016,-.00020,-.00025,-.00029,-.00032,-.00035,-.00038,-.00041, 00088600
C -.00043,-.00046,-.00048,-.00050,-.00052,-.00052,-.00053,-.00054, 00088700
D -.00055,-.00056,-.00056,-.00056,8* -.00056 ,-.00056,-.00056,     00088800
E -.00054,-.00053,-.00052,-.00051,-.00050,-.00049,-.00048,-.00047, 00088900
F -.00046,-.00045,-.00044,-.00042,-.00041,-.00040,-.00038,-.00037, 00089000
G -.00035,-.00033,-.00032,-.00030,-.00028,-.00026,-.00024,-.00022, 00089100
H -.00020/                                                         00089200
 DATA B1 /.007321,.007283,.007244,.007204,.007163,.007120,.007072  00089300
A,.007026,.006980,.006934,.006890,.006850,.006808,.006766,.006724, 00089400
B .006680,.006636,.006592,.006546,.006498,.006450,.006399,.006347, 00089500
C .006295,.006243,.006190,.006137,.006083,.006029,.005975,.005920, 00089600
D .005866,.005812,.005756,.005698,.005640,.005579,.005518,.005456, 00089700
E .005393,.005330,.005262,.005195,.005129,.005065,.005002,.004940, 00089800
F .004880,.004821,.004764,.004708,.004650,.004595,.004544,.004496, 00089900
G .004452,.004408,.004362,.004313,.004262,.004208,.004155,.004105, 00090000
H .004058,.004015,.003975,.003936,.003895,.003852,.003807,.003760, 00090100
I .003715,.003675,.003640,.003610,.003586,.003564,.003540,.003515, 00090200
J .003488,.003460,.003433,.003410,.003391,.003376,.003364,.003353, 00090300
K .003340,.003325,.003307,.003288,.003268,.003251,.003235,.003222, 00090400
L .003210,.003199,.003186,.003171,.003156,.003138,.003120,.003102, 00090500
M .003085,.003069,.003053,.003037,.003020,.003001,.002982,.002962, 00090600
N .002941,.002921,.002900,.002880,.002860,.002839,.002817,.002794, 00090700
O .002770,.002745,.002719,.002692,.002666,.002639,.002611,.002583, 00090800
P .002555,.002526,.002496,.002466,.002436,.002405,.002375,.002345, 00090900
Q .002315,.002285,.002254,.002222,.002190,.002158,.002125,.002093, 00091000
R .002060,.002028,.001995,.001962,.001929,.001896,.001863,.001829, 00091100
S .001795,.001762,.001729,.001698,.001666,.001635,.001603,.001572/ 00091200
 DATA B2 /.001540,.001508,.001476,.001445,.001415,.001385,.001355, 00091300
A .001326,.001297,.001268,.001239,.001210,.001182,.001154,.001127, 00091400
B .001100,.001075,.001049,.001024,.001000,.000975,.000951,.000927, 00091500
C .000904,.000881,.000859,.000838,.000817,.000796,.000775,.000753, 00091600
D .000732,.000711,.000691,.000671,.000653,.000635,.000618,.000600, 00091700
E .000581,.000563,.000544,.000525,.000507,.000490,.000474,.000459, 00091800
F .000444,.000428,.000412,.000396,.000379,.000362,.000347,.000332, 00091900
G .000319,.000306,.000294,.000281,.000268,.000255,.000241,.000228, 00092000
H .000216,.000205,.000195,.000187,.000178,.000170,.000160,.000149, 00092100
I .000138,.000127,.000117,.000108,.000101,.000095,.000089,.000082, 00092200
```

```
     J .000075,.000066,.000056,.000050,.000042,.000037,.000030,.000025,
     K .000017,.000010,.000004,.000001,.0/
      DATA C1 /.00047810,.00047810,.00047808,.00047806,.00047804,
     A .00047800,.00047797,.00047794,.00047788,.00047775,.00047750,
     B .00047688,.00047612,.00047522,.00047418,.00047300,.00047135,
     C .00046973,.00046813,.00046655,.00046500,.00046333,.00046139,
     D .00045919,.00045673,.00045400,.00045035,.00044513,.00043833,
     E .00042995,.00042000,.00040983,.00040079,.00039289,.00038613,
     F .00038050,.00037508,.00036894,.00036208,.00035450,.00034620,
     G .00033790,.00033034,.00032350,.00031738,.00031200,.00030670,
     H .00030085,.00029444,.00028748,.00027996,.00027256,.00026594,
     I .00026011,.00025507,.00025081,.00024665,.00024190,.00023656,
     J .00023062,.00022410,.00021774,.00021228,.00020774,.00020412,
     K .00020140,.00019876,.00019534,.00019116,.00018622,.00018050,
     L .00017480,.00016990,.00016580,.00016250,.00016000,.00015751,
     M .00015425,.00015021,.00014539,.00013980,.00013422,.00012946,
     N .00012550,.00012234,.00012000,.00011770,.00011469,.00011095,
     O .00010651,.00010134,.00009627,.00009211,.00008887,.00008653,
     P .00008510,.00008377,.00008174,.00007900,.00007555,.00007140,
     Q .00006731,.00006407,.00006166,.00006009,.00005937,.00005873,
     R .00005741,.00005542,.00005276,.00004942,.00004616,.00004375,
     S .00004218,.00004145,.00004157,.00004176,.00004126,.00004006/
      DATA C2 /.00003817,.00003560,.00003310,.00003143,.00003062,
     A .00003064,.00003150,.00003241,.00003256,.00003196,.00003060,
     B .00002849,.00002804,.00002758,.00002714,.00002675,.00002634,
     C .00002599,.00002567,.00002538,.00002511,.00002485,.00002464,
     D .00002447,.00002434,.00002423,.00002414,.00002406,.00002400,
     E .00002395,.00002391,.00002388,.00002386,.00002385,.00002383,
     F .00002382,.00002382,.00002382,.00002382,.00002383,.00002384,
     G .00002386,.00002388,.00002389,.00002390,.00002390,.00002389,
     H .00002388,.00002387,.00002385,.00002383,.00002381,.00002377,
     I .00002373,.00002369,.00002364,.00002360,.00002353,.00002346,
     J .00002338,.00002330,.00002322,.00002313,.00002304,.00002295,
     K .00002286,.00002276,.00002266,.00002256,.00002245,.00002234,
     L .00002223,.00002232,.00002241,.00002150,.00002160,.00002170,
     M .00002160,.00002150,.00002139,.00002128,.00002117,.00002108,
     N .00002098,.00002087,.00002076,.00002065,.00002054,.00002043,
     O .00002032,.00002021,.00002010,.00001998,.00001986,.00001974,
     P .00001962,.00001950,.00001938,.00001926,.00001913,.00001900,
     Q .00001887,.00001872,.00001856,.00001840,.00001824,.00001808,
     R .00001792,.00001776,.00001761,.00001746,.00001731,.00001715,
     S .00001699,.00001683,.00001668,.00001652,.00001637,.00001622/
      DATA C3 /.00001606,.00001591,.00001576,.00001561,.00001545,
     A .00001529,.00001511,.00001497,.00001483,.00001469,.00001454,
     B .00001438,.00001421/
C
      CALL ERRSET(207,256,-1,1,1,206)
      CALL ERRSET(209,260,-1,1,1,208)
      CALL ERRSET (253,256,-1,1,1,252)
      CALL ERRSET (208,256,-1,1,1,1)
C
      KKK = 1
```

```
      IF ((IFLG.EQ.5) .OR. (IFLG.EQ.6)) KKK = 3
      RR5 = (6076.1155) * R95 / SQRT(2.*ALOG(20.))
      DENOM =  .7213475 * CEP**2  +  ((DSIG*WR)**2)  +  RR5**2
      VV = SQRT(DENOM)
      IF (WR.LE..001) VV = 1000.
      BR = WR / VV
      SR = D * 6076.1155 / VV
      HK = (BR * 50.) + 1.
      KH = HK
      HH = KH
      KK = KH + 1
      BB = (HK - HH)
C                 CALCULATE PD FOR D = 0
      PV = 1. - EXP(-(BR**2/2.))
C
C
C  INTERPOLATE BETWEEN MEMBERS IN CIRCULAR COVERAGE FUNCTION TABLE AND
C      HASTINGS' APPROXIMATION TABLES.
C
      AV=BB*(A(KK)-A(KH))+A(KH)
      BV=BB*(B(KK)-B(KH))+B(KH)
      CV=BB*(C(KK)-C(KH))+C(KH)
C
C  IF DISTANCE OUTPUT (D) IS DESIRED GO TO STATEMENT NUMBER 400.
C
      IF (KKK.EQ.3) GO TO 400
C
C  COMPUTE POD USING HASTINGS' APPROXIMATION.
C
      POV=PV/(1.+(AV*SR*SR)+BV*(SR**4)+CV*(SR**6))**4
      POD = POV
      IF ((POV.GT..99) .AND. (IFLG.EQ.1)) POD = .99
      IF ((IFLG.EQ.2) .AND.(POV.GT..999)) POD = .999
C
C  RETURN CONTROL TO CALLING PROGRAM.
C
      RETURN
 400    CONTINUE
          IF (PV.LT.POD) GO TO 195
      IF (PV.LE.POD) GO TO 901
      OK=(((9.*AV*BV)/CV)-(2.*(BV**3))/(CV**2)+27.*(((PV/POD)
     1     **.25)-1.))/(54.*CV)
      YK=((BV**2/CV)-(3.*AV))/(9.*CV)
C
37    CN=(OK**2)-(YK**3)
      IF (CN) 50,40,40
40    CN=CN**.5
      CNPOK = ABS(CN+OK)
      CNMOK = ABS(CN-OK)
      CRA=((CNPOK/(CN+OK))*CNPOK**.33333333)+((CNMOK)/(OK-CN) *
     A        CNMOK**.33333333)
      GO TO 410
```

```
50    CN=OK/(YK**1.5)
      IF (CN.GT.1.) IERR=8
      IF (CN.GT.1.0) GO TO 900
      BK=ARCOS(CN)
      AK=BK/3.0
      CN=COS(AK)
      CRA=2.0*(YK**.5)*CN
C
C  COMPUTE D (OFFSET DISTANCE).
C
410   D=(((CRA-(BV/(3.*CV)))*(VV**2))**.5)/6076.1155
      JSIG=0
C
C  RETURN CONTROL TO CALLING PROGRAM.
C
      RETURN
C
 900  CONTINUE
 195  IERR = 1
 901  D = .0
      RETURN
      END
C*****************************************************************
C
      SUBROUTINE LNCALC (CEP, DSIG, WR, R95, POD, D, IFLG, RADIUS,
     A                   ANGL1, ANGL2, IERR)
C
C*****************************************************************
C
C       SUBROUTINE LNCALC IS A SUBROUTINE USED TO CALCULATE POD AND
C             OFFSET DISTANCE USING THE LOG NORMAL PROBABILITY FCTN
      DIMENSION  COEM(6,5), COEV(6,5),PJ(3,5)
      DIMENSION W(10), ZP(10)
      F(U) = (1. + U*(.278393 + U*(.230389 + U*(.000972 + U*.078108))))
     A       **4
      G(U)=1.-1./F(U)
      P(R)= 0.5 +     SIGHN(ZX) / 2.0 *G( ABS( ZX )/1.414214)
      P1(R)= R* EXP(-.5*(R**2 + X**2))
      P2(R) = R*EXP(-.5*(X-R)**2)
      T(R)=R*X/3.75
      P2B(R) = 1./ SQRT(R*X) * (((((((.00392377/T(R) -.01647633)/T(R) +
     A         .02635537)/T(R) - .02057706)/T(R) + .00916281)/T(R) -
     B         .00157565)/T(R) + .00225319)/T(R) + .01328592)/T(R) +
     C         .39894228)
      P2A(R) = 1. + T(R)*T(R) *(3.5156229 + T(R) *T(R) *(3.0899424 +
     A              T(R)*T(R)*(1.2067492 + T(R)*T(R)*(0.2659732 +
     B              T(R)*T(R)*(0.0360768 + T(R)*T(R)*0.0045813)))))
      FR(R)=P(R)*P1(R)*P2A(R)
      FS(R) = P(R)*P2(R)*P2B(R)
      FSS(R) = P2(R)*P2B(R)
      FRR(R)=P1(R)*P2A(R)
      DATA COEM /
```

LNCALC ≡ CUM. LOG. NORMAL FUNCTION CALC.

```
C  COEM(X,IZ) IS TABLE USED TO COMPUTE THE SLOPE,M,OF ASSYMPTOTES IN D
C    CALCULATION SECTION. IZ/10=SIGD    THIS IS TABLE IV
     A 1.1200156,-.733645,3.14852,-7.68466,8.60303,-3.57196,
     B1.371312,-3.040636,12.6217,-27.77372,28.63859,-11.1665,
     C 1.70948,-6.337744,26.4362,-56.40256,55.64071,-20.54056,
     D 2.119486,-11.33779,47.46499,-97.99416,93.57898,-33.48051,
     E 2.700537,-20.52581,91.33112,-189.66498,179.74574,-63.36967/
C
      DATA COEV
C  COEV(X,IZ) IS TABLE USED TO COMPUTE THE VALUE V,D/CLAP AT WRNM=5.
C    THIS IS TABLE V.
     A/7.4698077,-20.56971,90.3795,-206.5916,218.1826,-86.43575,
     B 8.4127425,-28.25754,119.86435,-266.1521,274.2819,-106.30246,
     C 9.7803031,-44.7466,193.84425,-423.3461,427.3663,-161.90186,
     D10.4684977,-45.97732,177.55602,-368.4701,363.1512,-136.27004,
     E13.8920708,-102.93643,458.59119,-968.1678,935.9434,-337.67352/
C
      DATA PJ
C  TABLE PJ(X,IZ) GIVES VALUES USED TO FIND WR/CLAP
     A/6*0.0,
     B 5.75569,-15.3945,10.9252,
     C 9.05179,-23.3401,15.8942,
     D 9.4156,-24.3744,16.8657/
C
      DATA  W / .0666713443, .1494513492, .2190863625, .2692667193,
     W           .2955242247/,
     Z     ZP / .9739065285, .8650633667, .6794095683, .4333953941,
     P           .1488743390/
C
      CALL ERRSET (208,256,-1,1,1,1)
      CALL ERRSET(209, 256, -1, 1, 1, 1)
C
      RADIAL = RADIUS
C
      IF (RADIUS .LE. 0.0)
     A    RADIAL = 10000000.
C     ENDIF
C
      ANGLFR = ABS (ANGL2-ANGL1) / 360.
C
      IF (ANGLFR .GT. 1.0)
     A    ANGLFR = 1.0
C     ENDIF
C
      KKK = 1
      IF ((IFLG.EQ.5) .OR. (IFLG.EQ.6)) KKK = 3
      IF (KKK.EQ.3)  D = 0.0
      D = D * 6076.1155
      IJOB=0
      ITCH=0
      FK=KF
      RR5 = 6076.1155 * R95
```

```
      ADCEP = SQRT(CEP**2 + .231 * RR5**2)
      IF (WR.LE..001) GO TO 68
 201  CONTINUE
C
C  COMPUTE BETA-FACTOR USED IN COMPUTING Z, THE UPPER LIMIT OF THE
C    INTEGRAL. ALSO COMPUTE 'ADJUSTED CEP', ADCEP, USE IT TO NORMALIZE
C    D AND WR.
C
46    BETA = (-ALOG(1-DSIG**2))**.50
      EX = EXP(-(BETA)**2)
      IF (ADCEP.GT.0.00) GO TO 70
C
C  COMPUTE POD WHEN CEP = R95 = 0
C
C  IF  D ALSO EQUALS 0 SET POD = .999
C    OTHERWISE, I    COMPUTE POD. THIS IS DIFFERENT THAN THE GENERAL
C    CASE AS D AND WR CANNOT BE NORMALIZED.
      IF (D.EQ.0.0) GO TO 66
C  COMPUTE Z
      Z = (1/BETA) * ALOG((WR*EX)/D)
C
C  IF Z > 3.87 POD =.999, IF Z IS CLOSE TO 0, POD =.50
C    IF Z <-3.87 POD IS 0 FOR ALL PRACTICAL PURPOSES.
C
      IF (Z.GT.3.87) GO TO 66
      ZAB = ABS(Z)
      IF (ZAB.LT.0.0000005) GO TO 67
      IF (Z.LT.-3.87) GO TO 68
C  POD EQUALS .5 + .5 * (ABS(Z)/Z) * ERF(1)
      C = (ABS(Z)) / (2**.5)
      ERFU = 1 - 1/((1 + .0705230784*C + .0422820123*C**2 +
     A .0092705272*C**3 + .0001520143*C**4 + .0002765672*C**5 +
     B .0000430638*C**6)**16)
      POV = .5 + .5*ABS(Z)/Z * ERFU
      GO TO 150
  66  POV = .999
      GO TO 150
  67  POV = .500
      GO TO 150
  68  POV = 0.00
      GO TO 150
70    CONTINUE
C
C  NORMALIZE WR AND D.
C  X IS THE SYMBOL USED FOR NORMALIZED D
C
      RADN = 1.1774 * RADIAL / ADCEP
      WRN = 1.1774 * WR / ADCEP
        X = 1.1774 *  D / ADCEP
      L = 0
C
C  FSUM WILL SUM TERMS OF GAUSSIAN QUADRATURE
```

```
C
71    CONTINUE
      FSUM = 0.0
      BMINSA = .0
C  IF DN-4 < 0 BEGIN INTEGRATION WITH RADIUS OF ZERO, OTHERWISE AT DN-4.
C  SET INTEGRATION INTERVAL.
      XBB = 1.06 * WRN * EXP (2.86 * DSIG)
      XB = X + 4.0
      N1 = 5
      IF (XBB .LT. XB)  XB = XBB
      IF (RADN.LT.XB)
     A     XB = RADN
C     ENDIF
      IF (X -4.0) 75,75,80
 75   XA = 0.0
      BPLUSA = XB
      BMINSA = XB
      GO TO 85
 80   XA = X - 4.0
      BPLUSA = XA + XB
      BMINSA = XB - XA
      IF (BMINSA) 870, 870, 85
85    CONTINUE
C
C  COMPUTE POD THROUCH LOOP 120
C
C                    BEGINNING OF LOOP
C
      DO 120 N=1,N1
      R1 =(-BMINSA * ZP(N) + BPLUSA) / 2.0
      R2 = (BMINSA * ZP(N) + BPLUSA) / 2.0
C  COMPUTE Z'S, UPPER LIMITS OF INTEGRALS
      Z1 = (1/BETA) * (ALOG(WRN*EX/R1))
      Z2 = (1/BETA) * (ALOG(WRN*EX/R2))
C  CHECK MAGNITUDE OF R1 TO SEE IF R1 > 3.75
      ZX=Z1
      IF (T(R1).GT.1.0) GO TO 88
C  CHECK LIMITS ON Z1
      IF (Z1.GT.3.87) GO TO 86
      IF (Z1.LT.-3.87) GO TO 870
      FSUM = FSUM + (W(N) * FR(R1))
      GO TO 89
86    FSUM = FSUM + (W(N) * FRR(R1))
C  CHECK MAGNITUDE OF R2
89    ZX=Z2
      IF (T(R2).GT.1.0) GO TO 95
C  CHECK LIMITS ON Z2
      IF (Z2.GT.3.87) GO TO 96
      IF (Z2.LT.-3.87) GO TO 97
      FSUM = FSUM + (W(N) * FR(R2))
      GO TO 97
96    FSUM = FSUM + (W(N) * FRR(R2))
```

```
      GO TO 97
C  DO THE ABOVE FOR R1 > 3.75
88    IF (Z1.GT.3.87) GO TO 90
      IF (Z1.LT.-3.87) GO TO 870
      FSUM = FSUM + (W(N) * FS(R1))
      GO TO 89
90    FSUM = FSUM + (W(N) * FSS(R1))
      GO TO 89
C DO THE ABOVE FOR R2 < 3.75
95    IF (Z2.GT.3.87) GO TO 100
      IF (Z2.LT.-3.87) GO TO 97
      FSUM = FSUM + (W(N) * FS(R2))
      GO TO 97
100   FSUM = FSUM + (W(N) * FSS(R2))
C  INCREMENT R1 AND R2
97    CONTINUE
120   CONTINUE
C
C                    END OF LOOP
C
870   CONTINUE
C
C
      POV = FSUM * BMINSA/2.0 * ANGLFR
150   CONTINUE
C
C  WE NOW HAVE A GOOD POD
C
C  WHERE DO WE GO FROM HERE?                          FOR D GO TO 600.
      IF (KKK.EQ.1) GO TO 160
      IF (KKK.EQ.3) GO TO 600
160   POD = POV
      D = D / 6076.1155
      IF (IFLG.EQ.2) GO TO 390
      IF (POV.GT..99) POD = .99
      RETURN
390   CONTINUE
      IF (POV.GT..999) POD = .999
      RETURN
C
600   CONTINUE
C
C  THIS IS WHERE COMPUTATION OF D, OFFSET DISTANCE, OCCURS IF IT IS
C    DESIRED.  THIS COMPUTES THE MAX DISTANCE AT WHICH A GIVEN
C    MINIMUM POD CAN BE OBTAINED.
C
C  SINCE IN THIS CASE POV WAS COMPUTED WITH D =0, IF DESIRED POD > POV,
C    POD IS UNATTAINABLE.
      IF (ITCH.GT.0) GO TO 620
      IF (POV.LT.POD) GO TO 690
C  CLIP IS ADJUSTED CEP FOR THIS PORTION OF THE PROGRAM, WRNM IS
C    NORMALIZED WR
```

```
      CLIP = SQRT(CEP**2 + .231*RR5**2)
C
C             SEE IF ADJUSTED CEP = 0. IF SO, MAKE A DIFFERENT 1ST GUESS
      IF (CLIP.EQ.0) GO TO 615
C
      WRNM = 1.1774 * WR / CLIP
      ADSG = DSIG * 10 + .0001
      IZ = ADSG
      BADM = COEM(1,IZ) + COEM(2,IZ)*POD + COEM(3,IZ)*POD**2 +
     1 COEM(4,IZ)*POD**3 + COEM(5,IZ)*POD**4 + COEM(6,IZ)*POD**5
C  BADM IS THE SLOPE OF ASSYMPTOTE OF HYPERBOLA FOR D VS WR CURVES.
      BADV = COEV(1,IZ) + COEV(2,IZ)*POD + COEV(3,IZ)*POD**2 +
     1 COEV(4,IZ)*POD**3 + COEV(5,IZ)*POD**4 + COEV(6,IZ)*POD**5
C  BADV IS THE D/CLIP VALUE AT WR/CLIP = 5.
C  GIN IS THE WR/CEP INTERCEPT APPROXIMATION
      GIN = 5.- BADV/BADM
      IF ((POD.LE..70).OR.(DSIG.LE..2001)) GO TO 605
C  APWRC IS THE APPROXIMATE WR/CLIP.
      APWRC = PJ(1,IZ)+ PJ(2,IZ)*POD + PJ(3,IZ)*POD**2
      APWRC = EXP(APWRC)
      GO TO 610
605   APWRC = (1./SQRT(1.386295*((-1.)/(2.*ALOG(1.-POD)))-DSIG**2))
610   CONTINUE
C
C  THIS NEXT FORMULA COMPUTES THE FIRST GUESS AT MAX D,DM.
C
      DMSQ = BADM**2*(WRNM-GIN)**2 - (BADM*(APWRC-GIN))**2
        DM = CLIP * .1
      IF  (DMSQ.GT..0) DM = CLIP * SQRT(DMSQ) / 1.1774
      ITCH = 1
      D = DM
      GO TO 201
C
 615  ITCH = 1
      DM = (1.5 - POD) * WR
      D = DM
      GO TO 201
620   PDA = ABS(POD-POV)
      IF (PDA.LT..001)  GO TO 666
      IF (ITCH.GT.1) GO TO 625
      D1 = DM
      D = D1 - (POD-POV)*D1
      PC = POV
      ITCH = 2
      GO TO 201
 625  IF (POV.LT.PC) GO TO 627
      DM = D - (POD-POV) / (PC-POV) * (D-D1)
626   I = 3
      PC = POV
      D1 = D
      D = DM
      IF (DM.LT.0.0) GO TO 690
```

```
      GO TO 201
627   DM = D1 + (D-D1)*(PC-POD)/(PC-POV)
      GO TO 626
666   D = D / 6076.1155
C
C  HERE IS WHERE CONTROL IS RETURNED TO MAIN PROGRAM FROM OFFSET
C    DISTANCE COMPUTATION.
C
      RETURN
690   CONTINUE
      IERR = 1
      D    = 0.0
      RETURN
      END
      REAL FUNCTION  SIGHN (ZX)
      SIGHN = 1.0
      IF (ZX.LT.0.0)  SIGHN = -1.0
      RETURN
      END
C*****************************************************************************
C
      SUBROUTINE ETCALC (IV,JT,KF,YLD,CEP,HOB1,ORIEN,AZMTH,DI,POD,IERR)
C
C*****************************************************************************
C
C       ETCALC CALCULATES POD FOR EQUIVALENT TARGET AREA TYPE TARGETS.
C            THESE TGTS INCLUDE BRIDGES, CANAL LOCKS, AND DAMS.
C
      DIMENSION  INW(3,10,5),  CRW(  10,5),  DSWV(  10,5),   VNW(10,5),
     A           INL(6,10,5),  CRL(2,10,5),  DSLV(2,10,5),   VNL(10,5)
C
C           *****   FUNCTIONS   *****
      DD(B,C) = ABS(B) / (SQ2*C)
C
      ER(B,C) = 1. + DD(B,C)*(W1+DD(B,C)*(W2+DD(B,C)*(W3+DD(B,C)*(W4+
     A  DD(B,C)*(W5+DD(B,C)*W6)))))
C
      ERFP(B,C) = (1. - (1./ER(B,C))**16) * ABS(B)/(2.*B)
C
C                **  POD FUNCTION  **
      P(B,C,D,E,F,G,H,A) = (ERFP(D,E) - ERFP(B,C)) *
     A                     (ERFP(H,A) - ERFP(F,G))
C
C                  **  DELIVERY SIGMA FUNCTION  **
      ACEP(A,B) = SQRT(CEP**2 + (1.1774*A*B)**2)/ 1.1774
C
C
      DATA IA,IB,IC,ID,IE / 'A', 'B', 'C', 'D', 'E'/
C
      DATA  INW     /
C
C     INW(I,J,L) CONTAINS VNTK VALUES TARGET WIDTHS IF THEY EXIST.
```

ETCALC ≡ EQUIV. TARGET AREA CALCULATION

```
C       I=1 IS VN, I=2 IS T, I=3 IS K. J=KF+1. K=1,2 IS FOR BRIDGES,
C        K=3 IS FOR CANALS, K=4 IS FOR DAMS.
C
C                     BRIDGES
     A  0,0,0,  0,0,0,  0,0,0, 31,1,0, 25,2,6, 20,2,6, 18,2,6, 25,2,8,
     B 15,2,9, 16,2,8, 0,0,0, 18,2,9, 17,2,9, 16,2,8, 15,2,9, 17,2,8,
     C14,2,9, 16,2,9, 16,2,9, 0,0,0,
     D18,2,9, 17,2,9, 16,2,8, 15,2,9, 16,2,9, 17,2,8, 17,2,8, 9*0,
C                       DAMS (UPSTREAM VNTK)
     E 41,1,0, 38,1,0, 38,1,0, 42,1,0, 39,1,0, 39,1,0, 39,1,0, 35,1,0,
     F 35,1,0, 0,0,0,
C                       CANALS
     G 30 * 0 /
C
      DATA CRW          /
C
C    CRW(J,L) CONTAINS CRATER RADIUS FACTOR FOR WIDTH TGTS IF IT EXISTS.
C
C                       BRIDGES
     A 1.5, 2.0, 1.5, 27*.0,
C                        DAMS (UPSTREAM CRF)
     C  9*.0, 1.0,
C                       CANALS
     D 1.0, 1.5, 1.0, 1.5, 1.0, 1.5, 4*.0/
C
      DATA  INL          /
C
C    INL(I.J.L) CONTAINS LENGTH VNTK FOR ETA TGT FOR BOTH FRONT AND BACK.
C             SUBSCRIPTS HAVE MEANINGS SIMILAR TO INW.
C
C                        BRIDGES
     A 18*0, 38,1,0,0,0,0,   29,2,6,0,0,0, 23,2,6,0,0,0, 21,2,6,0,0,0,
     B 29,2,8,0,0,0, 18,2,9,0,0,0, 22,2,8, 9*0, 22,2,9,0,0,0,20,2,9,
     2 0,0,0, 19,2,8,
     C 0,0,0, 21,2,7,0,0,0, 23,2,8,0,0,0, 23,2,7,0,0,0, 25,2,8,0,0,0,
     D 25,2,8, 9*0,
     E 22,2,8,3*0, 22,2,8,3*0, 22,2,8,3*0, 23,2,7,3*0,
     3 25,2,8,3*0, 23,2,7,3*0, 25,2,8, 21*0,
C                         DAMS (DOWNSTREAM VNTK)
     F 60 * 0,
C                       CANALS
     G 12*0, 31,1,4*0, 31,1,4*0, 31,1,0,31,1,0, 31,1,0,31,1,25*0/
C
      DATA   CRL          /
C
C    CRL(I,J,L) CONTAINS FRONT AND REAR CRF'S FOR ETA TGTS
C
C                         BRIDGES
     A 1.25,0., 1.5,.0, 1.25,.0, 34*.0,
     B 20*.0,
C                         DAMS (DOWNSTREAM CRF)
     C .5,.0, .5, .0, .5,.0, .5,.0, .5,.0,.5,.0, .5,.0, .5,.0, .5,.0,
```

```
     3 1.5, .0,
C                    CANALS
     D 2*1.0, 2*1.5, .0,1.0, .0,1.5, 12*.0/
C
      DATA    DSWV            /
C
C    DSWV(J,K) CONTAINS WIDTH DAMAGE SIGMAS
C              BRIDGES
     A 3*.3, .2, 6*.3,.0, 8*.3, .0,
     B 7*.3, 3*.0,
C                     DAMS (UPSTREAM DSIG)
     C 9*.2, .3,
C                    CANALS
     D 6*.3, 4*.0/
C
      DATA   DSLV                 /
C
C    DSLV(I,J,L) CONTAINS LENGTH DAMAGE SIGMAS AND DOWNSTREAM DSIG'S
C                     BRIDGES
     A .3,.0,.3,.0,.3,.0, .2,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,
     1 .0,.0, .3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0, .0,.0,
     B .3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,7*.0,
C                      DAMS  (W/DOWNSTREAM DSIG'S)
     C .0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,.0,.3,
C                  CANALS
     D 4*.3, .2,.3, .2,.3, 4*.2, 8*.0/
C
      DATA   VNW         /
C
C    VNW(J,L) CONTAINS WIDTH DIMENSIONS.
C
C                      BRIDGES
     A 5., 15., 25., 35., 45., 55., 65., 75., 85., 90.,
     A 5., 15., 25., 35., 45., 55., 65., 75., 85., 90.,
     A 5., 15., 25., 35., 45., 55., 65., 75., 85., 90.,
C                    DAMS
     B 5., 15., 22., 27., 40., 63., 88., 125., 200., 250.,
C                    CANALS
     C 20., 50., 70., 90., 110., 130., 150., 170., 190., 200./
C
      DATA     VNL           /
C
C    VNL(J.L) CONTAINS LENGTH DIMENSIONS.
C
C                    BRIDGES
     A 50.,150.,400.,800.,1200.,1600.,2000.,2400.,2800.,3000.,
     A 50.,150.,400.,800.,1200.,1600.,2000.,2400.,2800.,3000.,
     A 50.,150.,400.,800.,1200.,1600.,2000.,2400.,2800.,3000.,
C                    DAMS
     B 250.,750.,1500.,2500.,3500.,4500.,7500.,12500.,20000.,25750.,
C                    CANALS
     C  50.,150., 400., 800.,1200.,1600.,2000., 2400., 2800., 3000./
```

```
C
      DATA W1, W2, W3, W4, W5, W6   /
C
C  WI'S ARE THE CONSTANTS FOR THE ERROR FUNCTION QPPROXIMATION
     A .0705230784,.0422820123,.009270572,.0001520143,.0002765672,
     B .0000430638/
C
C               SET CONSTANTS AND INITIALIZE VARIABLES.
      SQ2 = SQRT (2.)
      IGV = IV/10
      IGN = IV - (IGV*10)
      WRL1 = .0
      WRL2 = .0
      WRW1 = .0
      KK = KF +1
      JTS = 0
C          CHECK DIMENSION SUBSCRIPTS
      IF (IGN.EQ.0)IGN = 10
      IF (IGV .EQ. 0) IGV =10
C                       DECODE JT
      IF (JT.EQ.IA) GO TO 102
      IF (JT.EQ.IB) GO TO 101
      IF (JT.EQ.IC) GO TO 100
      IF (JT.EQ.ID) GO TO 400
      IF (JT.EQ.IE) GO TO 300
C            IF STILL  HERE,  SET IERR
      IERR = 8
      RETURN
C
C               ****  BRIDGE SECTION  ****
C
C                      SET JTS TO 1 OR 2 OR 3 FOR BRIDGES
 100  JTS = 1
 101  JTS = JTS + 1
 102  JTS = JTS + 1
C
C                 DETERMINE WEAPON RADII
C
C                SEE IF CRATER OR NON-CRATER
      IF (CRL(1,KK,JTS).GT.0) CALL WRCRTR(YLD,CRL(1,KK,JTS),WRL1,JTS,KF)
      IF (INL(2,KK,JTS).GT.0) CALL WRCALC (YLD,HOB1,INL(1,KK,JTS),
     A          INL(2,KK,JTS), INL(3,KK,JTS),DSLV(1,KK,JTS),WRL1,IERR)
C
      IF (CRW(KK,JTS).GT.0) CALL WRCRTR (YLD,CRW(KK,JTS),WRW1,JTS,KF)
      IF (INW(2,KK,JTS).GT.0) CALL WRCALC(YLD,HOB1,INW(1,KK,JTS),
     A        INW(2,KK,JTS),INW(3,KK,JTS),DSWV(  KK,JTS),WRW1,IERR)
C
C
C             DETERMINE X AND Y OFFSET DISTANCES
C                 ORIEN IS TARGET ORIENTATION
C                 AZMTH IS AZIMUTH FROM DGZ TO TARGET
C                 XO IS THE EAST-WEST COMPONENT
```

```
C                 YO IS THE NORTH-SOUTH COMPONENT
      DDUM = DI * 6076.1155
      ANGLE = (AZMTH - ORIEN * 10.) / 57.295779
      XO = DDUM * SIN(ANGLE)
      YO = DDUM * COS(ANGLE)
C
C                   COMPUTE BOUNDARIES
C
      W = VNW(IGN,JTS)
      SL= VNL(IGV,JTS)
C
      A = -W/2. - WRW1 + XO
      B =  W/2. + WRW1 + XO
      C = -SL/2. - WRL1 + YO
      D =  SL/2. + WRL1 + YO
C
C                COMPUTE DELIVERY SIGMAS
C
      AA = ACEP(WRW1, DSWV(KK,JTS))
      AB = AA
      AC = ACEP(WRL1, DSLV(1,KK,JTS))
      AD = AC
C
C         **   COMPUTE POD   **
C
      POD = P(A,AA,B,AB,C,AC,D,AD)
C
      RETURN
C
C           ****   CANAL SECTION  ****
C
 300  JTS = 5
C
C               IF AIR-BURST SET POD TO ZERO
      IF (HOB1 .GT. .001) GO TO 500
C
C              DETERMINE WEAPON RADII
C         SEE IF CRATER OR NOT AND COMPUTE WR'S ACCORDINGLY
C
      IF (CRL(1,KK,JTS).GT.0) CALL WRCRTR(YLD,CRL(1,KK,JTS),WRL1,JTS,KF)
      IF (INL(2,KK,JTS).GT.0) CALL WRCALC (YLD,HOB1,INL(1,KK,JTS),
     A          INL(2,KK,JTS), INL(3,KK,JTS),DSLV(1,KK,JTS),WRL1,IERR)
      IF (CRL(2,KK,JTS).GT.0) CALL WRCRTR(YLD,CRL(2,KK,JTS),WRL2,JTS,KF)
      IF (INL(5,KK,JTS).GT.0) CALL WRCALC (YLD,HOB1,INL(4,KK,JTS),
     A          INL(5,KK,JTS), INL(6,KK,JTS),DSLV(1,KK,JTS),WRL2,IERR)
      IF (CRW(KK,JTS).GT.0) CALL WRCRTR (YLD,CRW(KK,JTS),WRW1,JTS,KF)
      IF (INW(2,KK,JTS).GT.0) CALL WRCALC(YLD,HOB1,INW(1,KK,JTS),
     A        INW(2,KK,JTS),INW(3,KK,JTS),DSWV(  KK,JTS),WRW1,IERR)
C
C               COMPUTE BOUNDARIES
      W = VNW (IGN,JTS)
      SL= VNL (IGV,JTS)
```

```
C
      A = -W/2. - WRW1
      B = -A
      C = -SL/2. - WRL1
      D = SL/2.  + WRL2
C                   COMPUTE DELIVERY  SIGMAS
      AA = ACEP(WRW1, DSWV(KK,JTS))
      AB = AA
      AC = ACEP (WRL1,DSLV(1,KK,JTS))
      AD = ACEP(WRL2,DSLV(2,KK,JTS))
C
C                *  COMPUTE POD   *
      POD = P(A,AA,B,AB,C,AC,D,AD)
C
      RETURN
C
C               ****   DAM SECTION   ****
C
C                      SET JTS = 4 FOR DAMS
 400  JTS = 4
C
C                 IF AIR-BURST SET POD TO ZERO
      IF (HOB1 .GT. .001) GO TO 500
C
C                 DETERMINE WEAPON RADII
C
      IF (CRL(1,KK,JTS).GT.0) CALL WRCRTR(YLD,CRL(1,KK,JTS),WRL1,JTS,KF)
      IF (INL(2,KK,JTS).GT.0) CALL WRCALC (YLD,HOB1,INL(1,KK,JTS),
     A          INL(2,KK,JTS), INL(3,KK,JTS),DSLV(1,KK,JTS),WRL1,IERR)
      IF (CRW(KK,JTS).GT.0) CALL WRCRTR (YLD,CRW(KK,JTS),WRW1,JTS,KF)
      IF (INW(2,KK,JTS).GT.0) CALL WRCALC(YLD,HOB1,INW(1,KK,JTS),
     A        INW(2,KK,JTS),INW(3,KK,JTS),DSWV(  KK,JTS),WRW1,IERR)
C
C           COMPUTE BOUNDARIES
      W = VNW (IGN,JTS)
      SL= VNL (IGV,JTS)
      C = -SL/2.
      D = -C
      IF (KF.EQ.9) GO TO 410
      A = -WRL1 + .10
      B =  WRW1 + .10
      GO TO 420
C
410                     CONTINUE
      A = -WRL1 + W/2.
      B =  WRW1 - W/2.
C
 420                  CONTINUE
C                                 COMPUTE DELIVERY SIGMAS
      AA = ACEP (WRL1,DSLV(2,KK,JTS))
      AB = ACEP (WRW1,DSWV(  KK,JTS))
      AC = ACEP (SL/2.,DSLV(1,KK,JTS))
```

```
      AD = AC
C
C              *  COMPUTE POD  *
C
      POD = P(A,AA,B,AB,C,AC,D,AD)
C
      RETURN
C
 500  POD = .0
      RETURN
      END
C***********************************************************************
C
      SUBROUTINE WRCRTR (YLD,CRF,WR,JTS,KF)
C
C***********************************************************************
C
      WR = 1.1 * CRF * 61. * YLD**.30
      RETURN
      END
C*********************************************************************** 
C
      SUBROUTINE ERRMSG(IERR,IV,JT,KF,YLD,CEP,HOB1,R95,D,WR,POD,IFLG)
C
C*********************************************************************** 
      DIMENSION  JJ(3)
      DATA  JJ / '1', '2', '3'/
      IT = JT
      IF (((JT.EQ.1) .OR. (JT.EQ.2)) .OR. (JT.EQ.3)) IT = JJ(JT)
      WRITE (6,5) IERR,IV,IT,KF,YLD,HOB1,R95,CEP,D,WR,POD,IFLG
    5 FORMAT (' ','YOU HAVE INPUT ERROR NO. ',I2,' YOUR INPUTS ARE AS FO
     ALLOWS:',/,' ',I4, A4, I4,
     6  F10.1, 2F10.2, 2F8.2, F10.0, F5.2, I2)
C
      D = .0
      WR = .0
      POD = .0
C
      GO TO (10,20,30,40,50,60,70,80,90,100,110), IERR                *         REPLACED*
      GO TO (10,20,30,40,50,60,70,80,90,100,110,120), IERR            *      REPLACEMENT*
C
 10   WRITE (6,11)
 11   FORMAT ('  YOU CANNOT ACHIEVE DESIRED POD WITH THIS WEAPON')
      RETURN
C
 20   WRITE (6,21)
 21   FORMAT(' VN (IV) IS TOO LARGE TO USE IN AVAILABLE DATA CURVES')
      RETURN
C
 30   WRITE (6,31)
 31   FORMAT ('  SHOB > 900 FT')
      RETURN
```

WRCRTR ≡ WEAPON EFFECTS RADIUS FOR CRATERING

```
C                                                                       00170200
 40   WRITE (6,41)                                                      00170300
 41   FORMAT (' THE ONLY OPTIONS AVAILABLE W/ ETA TGTS ARE IFLG=1,2.....00170400
     A. YOUR IFLG CONTAINS SOME OTHER VALUE.')                          00170500
      RETURN                                                            00170600
C                                                                       00170700
 50   WRITE (6,51)                                                      00170800
  51  FORMAT (' JT MUST CONTAIN AN X OR 3 WHEN USING IFLG = 7')         00170900
      RETURN                                                            00171000
C                                                                       00171100
 60   WRITE (6,61)                                                      00171200
 61   FORMAT(' CHECK VALUE IN KF. IT MUST BE LESS THAN 10')             00171300
      RETURN                                                            00171400
C                                                                       00171500
 70   WRITE (6,71)                                                      00171600
 71   FORMAT(' CHECK VALUE IN KF. IT MUST BE > OR = ZERO')              00171700
      RETURN                                                            00171800
C                                                                       00171900
 80   WRITE (6,81)                                                      00172000
81    FORMAT ('  INVALID VALUE IN  T  PORTION OF VNTK FOR ETA-TYPE TGTS 00172100
     A(JT)')                                                            00172200
       RETURN                                                           00172300
C                                                                       00172400
 90   WRITE (6,91)                                                      00172500
 91   FORMAT ('  INVALID VALUE IN  T  PORTION OF VNTK  (JT)')           00172600
       RETURN                                                           00172700
C                                                                       00172800
 100  WRITE (6,101)                                                     00172900
 101  FORMAT ('  VN IS TOO LARGE FOR SHOB > OR = 900 ')                 00173000
      RETURN                                                            00173100
C                                                                       00173200
 110  WRITE (6,111)                                                     00173300
 111  FORMAT(' FOR IFLG = 12, YOU MUST INPUT A P OR Q TYPE TARGET')     00173400
      RETURN                                                            00173500
C                                                                       00173600
 120  WRITE (6,121)                                                     00173650    *     INSERTED*
121   FORMAT (' FOR JT =Z, A CONTACT BURST IS REQUIRED ')               00173655    *     INSERTED*
      RETURN                                                            00173660    *     INSERTED*
      END                                                               00173700
```

ADDRESSEE	NUMBER OF COPIES
Organization of the Joint Chiefs of Staff SAGA/SFD Washington D.C. 20301	2
Department of the Army Office of the Chief of Research, Development, and Acquisition DAMA-CSM Washington D.C. 20301	1
Department of the Navy Office of the Chief of Naval Operations Strategic Policy and Nuclear Systems Washington D.C. 20350	1
Command and Control Technical Center Computer Service Directorate Washington D.C. 20305	2
Defense Nuclear Agency STVL Washington D.C. 20301	2
Defense Intelligence Agency DIR-4 Washington D.C. 20301	2
Lawrence Livermore Laboratory Wallace F. Dale (L-93) P.O. Box 808 Livermore, California 94550	1
Joint Stragetic Target Planning Staff	
CINCLANT Representative	2
CINCPAC Representative	2
CINCEUR Representative	2
JLP	10
JLT	15
JLE	2
JPS	2
JPT	2
JPP	2
Headquarters, Strategic Air Command Offutt AFB, Nebraska 68113	
IN	2
AD	5
TOTAL	54

FOREWORD

This report contains no classified information.

This Technical Report was written by:

ROBERT H. MILLER
Lt Colonel, USA
Operations Staff Officer
Weapons Effects Section

Reviewed by:

ROBERT S. KUBBY
Colonel, USA
Chief, Targets Branch

Approved for publication:

RICHARD J. PETERSON
Colonel, USAF
Deputy Chief, NSTL Division

JSTPS-TR-76-2

PDCALC

A COMPUTER ROUTINE FOR PROBABILITY OF DAMAGE CALCULATIONS

TECHNICAL REPORT NO. JSTPS-TR-76-2
JOINT STRATEGIC TARGET PLANNING STAFF
OFFUTT AIR FORCE BASE
NEBRASKA
APRIL 1976

INCORPORATES CHANGE 2

TELEDYNE
BROWN ENGINEERING

MEMORANDUM

TO: DISTRIBUTION

DATE: 17 January 1978

FROM: W. F. Dale

SUBJECT: On Blast Kill Probability and the VN System

The attached notes are intended to give a general understanding of the subject to those who are unfamiliar with this commonly used treatment of target hardness and kill probability.

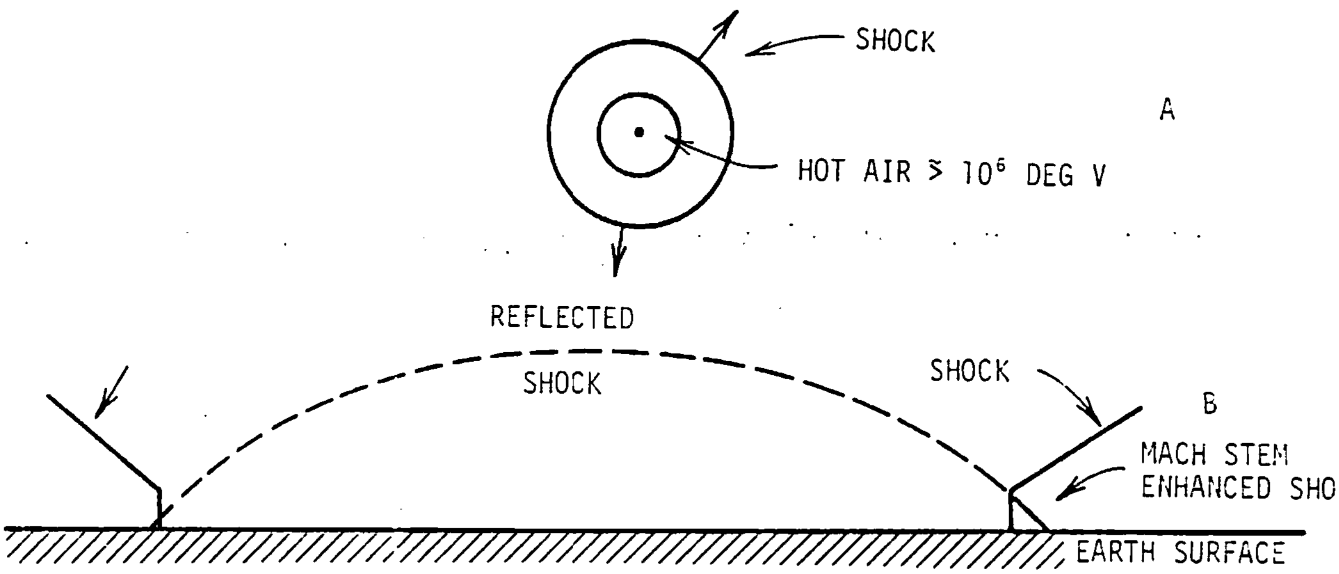
No claim is made for exactness. The References will give depth when they are obtained. Naturally, I would be happy to discuss the subject in broader detail with anyone who may care to do so.

W. F. Dale

W. F. Dale

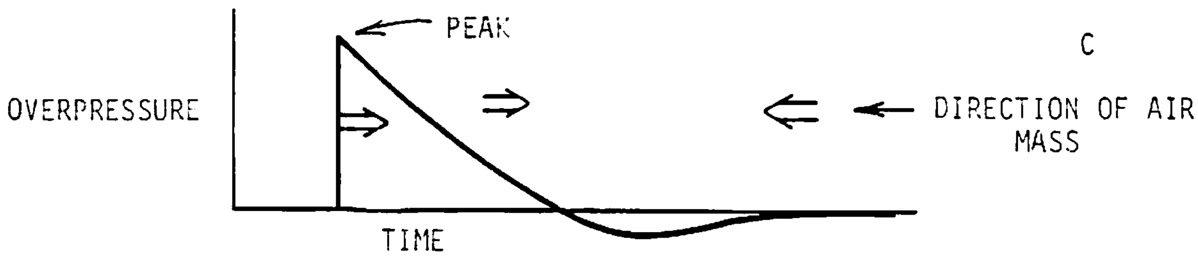
ON BLAST KILL PROBABILITY AND THE VN SYSTEM

Consider an above surface detonation of a nuclear warhead. Nuclear explosions generate large quantities of radiation in the UV and soft x-ray region. This radiation is absorbed by the air close to the burst, heats it to a very high temperature, and this spherical ball (or fireball) of hot air generates a spherical shock wave. The strength of the shock is proportional to the warhead yield. This spherical shock will have high temperature for a period of time and be seen as a fireball. It will cool as it expands, cease radiating, and the inner fireball will be seen. Both fireballs contribute to the thermal flux. The intensity, total flux, and time duration are yield dependent.



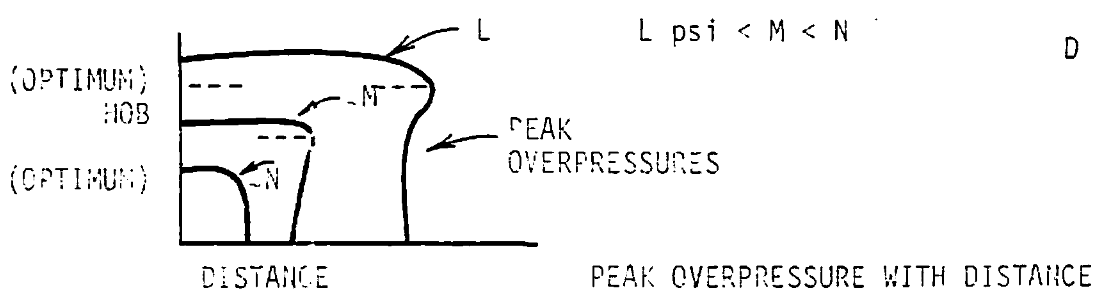
Schematically, A and B show shock generation and ground reflection for a warhead burst above the surface but below a height of burst where the shock does not reach the surface. A surface, or near surface, burst would not have a reflected shock — only a semi-spherical shock. The shock strength is asymmetrically reduced as depth of burst increases until containment of the effects is reached. The enhanced shock at the surface from reflection gives us the optimum height of burst whereby a given overpressure is felt at maximum surface distance.

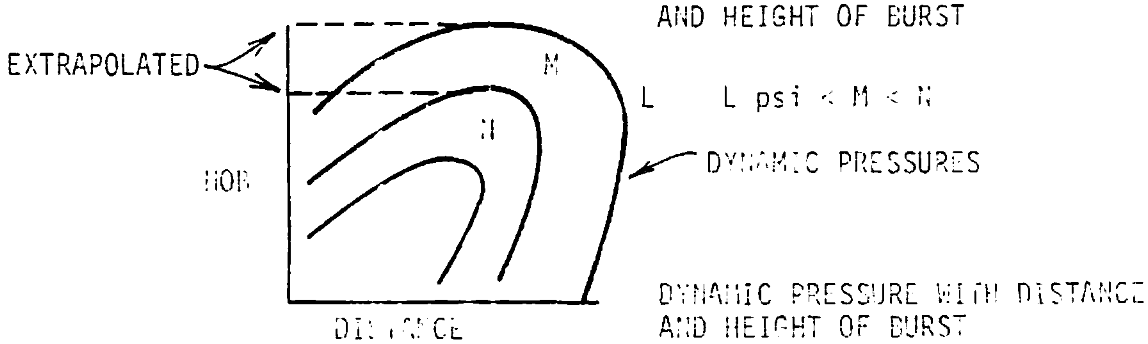
Two pressures are considered for blast damage — peak overpressure and dynamic pressure. The peak overpressure results from the shock front and is shown schematically with time as:



Dynamic pressure results from the kinetic energy of the mass of air moving past the observation point. Note that the mass of air moves away from the burst for a period of time and finally reverses direction back to the burst — nature abhors a vacuum. Obviously, both peak overpressure and dynamic pressure exist at the same observation point.

Schematically, with distance, we find for yield (Y):





We speak of the horizontal component of the dynamic pressure. Close in, and directly below, no horizontal component exists. Mathematically, one could get no calculated damage below a burst; thus a hole in a doughnut. The curves are usually extrapolated over as shown. It is important to note that the data is for an ideal surface. Cities do have buildings, forests do have trees, surface "pop corning" from thermal can occur, dust is raised, etc.

Peak overpressure results in an instantaneous static load which decays with time. Dynamic pressure results in a similar loading with time due to the mass of air in and behind the shock. Dimension and structure of a target determine response to shock waves.

To facilitate communication and common understanding of target response or damage, the physical vulnerability coding system was derived. Some structures fail primarily from peak overpressure — others from dynamic pressure. We call these P-type and Q-type, respectively. Each target type has been given a vulnerability number in most data bases. They are of the form NPK (as 12P5) and NQK (as 12Q8), respectively, for P-type and Q-type targets.

These vulnerability numbers are tied to peak and dynamic overpressures as:

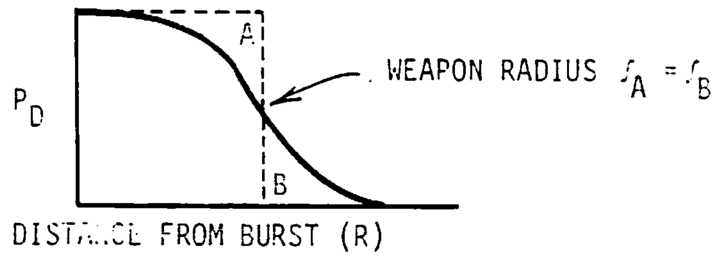
For P-type: $N = C_1 \; LOG_{10}P_{50} + C_2$ (1)

For Q-type: $N = C_3 \; LOG_{10}Q_{50} + C_4$ (2)

where P is the peak overpressure that will give a probability of damage of 50% from a 20 KT warhead; similarly for Q. Other probability (95%, 20%, etc.) can be used but 50% or .5 is normally taken when given a target susceptibility overpressure or VN. More on this later. Care should be taken when discussing damage overpressures as you may not be talking about the same damage probability.

It is most important to note that it is difficult, if not impossible, to determine a pressure that will cause a desired damage to all targets, or, more generally, all types of a target class; e.g., catalytic towers of oil refineries. Required pressures have been derived from observation of the destruction at Hiroshima and Nagasaki, test data at Rocky Flats, and/or structural analysis. We generally speak of moderate structural damage, MSD (moderate damage to building and contents, say, and "short" repair/utilization time) and severe structural damage, SSD (severe structural damage to buildings and contents — long, if at all, repair time). The probability consideration takes into account that all structures of a type are not the same, building orientation and/or shielding from other structures is not determined, etc.

Probability of damage is a function of distance from the burst ($P, Q \propto \frac{1}{R^3}$) and is shown, schematically, as:



The weapon radius is defined as that distance (or radius) within which there are as many targets undamaged to the desired degree as there are targets past that distance (or radius) which are damaged to the desired degree.

The damage function is cumulative log normal as:

$$PD = \int_{R=0}^{\infty} f(R)\,dR = \int \text{distribution fn} \cdot \text{damage fn}$$

$$f(R) = P_d(R) \cdot \frac{R \exp}{2\pi\sigma_w^2} \left[-(R^2 + X^2)/2\sigma_w\right] \int_0^{2\pi} e^{RX \cos\theta/\sigma_w^2}\, d\theta \qquad (3)$$

$$P_d(R) = \int_{-\infty}^{Z} \frac{1}{\sqrt{2\pi}}\; e^{-Y^2/2}\, dY$$

$P_d(R)$ is the cumulative log normal distance-damage function. It is a Gaussian distribution where the variable (Y) is a dummy. Z gives the upper limit as a function of weapon radius (WR), target damage sigma (σ_D) and distance (R) from a DGZ (or AGZ with CEP = 0) (desired ground zero and actual ground zero, respectively):

$$Z = \frac{1}{\beta} \ln \left(\frac{\alpha}{R}\right) \qquad (4)$$

$$\beta = \sqrt{-\ln(1 - \sigma_d^2)}$$

$$\alpha = WR(1 - \sigma_d^2)$$

For the circular normal distribution of impact points:

$$\sigma_w = CEP/1.1774 \tag{5}$$

It is also worthwhile noting that the probability of having a weapon land further than $4\sigma_w$ from the DGZ is $\leq$ .00005.

It is important to understand that P_n is the probability of achieving at least the required damage. A P_D of .8 (say) does not mean that 80% of the target is damaged to the desired extent!

Given a weapon radius (WR) — a function of yield, VN and height of burst — impact and damage sigma (σ_w and σ_D) or (CEP and σ_D) one can integrate for the probability of achieving at least a given level of damage at a (ground) distance from a DGZ (or AGZ if CEP is zero). The integration has, from necessity, been performed on digital computers. P_D has been plotted onto graphs and circular probability hand calculators that we have seen and used.

If we look at the shocks from nuclear bursts, we find that shock strengths and shock times of duration at any point away from the burst are proportional to the yield. Many targets are damaged to the same extent from different dynamic or peak overpressures if warhead yields are different — i.e., if the time that the target "sees" the shock is different. For example, a target may respond differently to 10 psi seen form 20KT compared to the response for 10 psi from 10 MT. Also, if 10 psi will cause SSD from 20 KT, SSD may result from only 4 psi from 10 MT as the time duration of the shock loading is significantly different. Damage vulnerability numbers account for this yield and time dependence.

Rigorously: We speak of the VN number, or the physical vulnerability number for, at least, a required target damage. This VN is of the form VNTK:

where

$5 < VN \lesssim 60$	- basic VN
$T \equiv$ either P or Q	
$0 \leq K \leq 9$	- showing yield dependence.

	NO YIELD DEPENDENCE	YIELD DEPENDENT
Examples:	12 PO	11Q6
	44 PO	18Q7
	22 PO	37P6

The basic VN is proportional to the Log of the pressure required for damage by equations (1) or (2). K is termed the "K-factor" and is used to <u>adjust the basic VN</u> for the yield dependence of the time duration of the shock. If K = 0, no yield dependence is observed. If K = 9, highest yield dependence is observed; e.g. 12P0 and 12P9.

A yield of 20 KT has been chosen as the reference yield for the theory. Below 20 KT a positive adjustment is made to the basic VN; above 20 KT a negative adjustment is made. Physically, this is sensible as shown by the example:

VN	K-FACTOR	DAMAGE
18Q7	$K \equiv 7$	SSD

Yield:	1 KT	20 KT	100 KT	1 MT	10 MT
Adjustment:	+ 3	0	-1.2	-2.2	-2.8
Adjusted VN:	21 Q	18 Q	16.8 Q	15.8 Q	15.2 Q
Dynamic Pressure:	42 psi	28 psi	13 psi	9 psi	7 psi

These adjusted VN show the importance of the K-factor. If, as assumed, 28 psi from 20 KT will result in PD = .5 of SSD, 42 psi is required from 1 KT, but only 7 psi from 10 MT. Distance-dynamic over-pressure curves (sketch D) would indicate large differences in distances from AGZ at which SSD would occur to 18Q7 type targets for this large yield spectrum as:

$$\frac{D_1}{D_2} = \left(\frac{Y_1}{Y_2}\right)^{\frac{1}{3}} \qquad (6)$$

Distances scale as yield $^{1/3}$. For example, if a certain overpressure from yield Y_1 is at D_1, the same overpressure would be observed at D_2 from yield Y_2. Therefore, in the VN example, a tremendous difference exists in the distances from AGZ at which 18Q7 damage occurs as one goes from 1 KT to 10 MT — not only from the above equation but from the adjustment to the VN for the time dependent structural response. Those interested in a rigorous treatment of the mathematics and VN numbers are referred to:

Mathematical Background and Programming Aids for the Physical Vulnerability System for Nuclear Weapons, DI-550-27-54, DIA (1974) unclassified.

Physical Vulnerability Handbook - Nuclear Weapons, AP--50-1-21 INT, DIA, (1969) RPT: Confidential.

Effects Manual, EM1 (vols 1 and 2), DIA rpt SRD.

Derivation of the K-Factor in the Physical Vulnerability System, DIA rpt. Confidential.

ACCURACY:

1. Overpressure range data is imported reliable to $\pm$ 20%, dynamic pressure data to $\pm$ 40%.

2. Values of 0.2 for σ_D are consistently used at the present time for P targets where the actual value varies from 0.1 to 0.3. For Q targets, 0.3 is used where it varies from 0.2 to 0.4.

3. The K-factor in VNTK is rounded to the nearest integer and can give a relative weapon radius error as high as $\pm$ 20%.

4. The VN portion of VNTK is rounded to integer values which gives a $\pm$ 5% relation error in weapon radius.

5. Conservative estimate on error in the weapon radius is $\pm$ 10% from all sources.

SOURCE: PDCALC - A Computer Routine for Probability of Damage Calculations, JSTPS-TR-76-2, Joint Strategic Target Planning Staff, Offutt Air Force Base, Nebraska, Apr 76, UNCL.

SINGLE SHOT KILL PROBABILITY (P_{kss}) AS A FUNCTION OF σ_y/σ_x AND LETHAL RADIUS (R_L)/CEP

σ_y/σ_x

R_L/CEP	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
	SINGLE SHOT KILL PROBABILITIES EXPRESSED AS PERCENT (%)									
0.1	2.21	1.26	0.96	0.90	0.87	0.85	0.84	0.83	0.83	0.83
0.2	7.54	4.69	3.64	3.20	2.98	2.87	2.81	2.77	2.76	2.75
0.3	13.82	9.78	7.84	6.97	6.56	6.31	6.17	6.10	6.06	6.05
0.4	19.85	15.81	13.18	11.91	11.26	10.89	10.69	10.57	10.52	10.50
0.5	25.43	22.16	19.27	17.72	16.89	16.43	16.16	16.09	15.93	15.91
0.6	30.91	28.40	25.70	24.09	23.19	22.67	22.36	22.19	22.11	22.08
0.7	35.91	34.34	32.17	30.72	29.67	29.37	29.07	28.90	28.82	28.80
0.8	40.81	39.91	38.46	37.37	36.70	36.30	36.06	35.92	35.85	35.83
0.9	45.51	45.12	44.42	43.85	43.47	43.24	43.10	43.01	42.97	42.96
1.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
1.1	54.27	54.58	55.16	55.71	56.16	56.43	56.61	56.71	56.76	56.77
1.2	58.33	58.87	59.93	61.00	61.67	62.43	62.79	63.00	63.11	63.14
1.3	62.16	62.89	64.30	65.62	67.07	67.92	63.45	68.79	68.96	69.01
1.4	65.76	66.63	68.31	70.18	71.75	72.85	73.54	74.01	74.23	74.29
1.5	69.13	70.11	71.98	74.10	75.93	77.23	78.10	78.62	78.95	78.98
1.6	72.27	73.33	75.31	77.60	79.60	81.06	82.04	82.64	82.95	83.04
1.7	75.19	76.30	78.34	80.72	82.82	84.36	85.42	86.07	86.40	86.50
1.8	77.88	79.02	81.09	83.48	85.61	87.19	88.28	88.96	89.31	89.41
1.9	80.36	81.50	83.55	85.92	88.02	89.58	90.61	91.35	91.70	91.81
2.0	82.62	83.75	85.77	83.06	90.08	91.59	92.30	93.30	93.64	93.74
2.1	84.66	85.79	87.73	89.92	91.83	93.26	94.25	94.87	95.19	95.30
2.2	86.55	87.63	89.48	91.53	93.31	94.63	95.54	96.11	96.39	96.51
2.3	88.24	89.27	91.03	92.93	94.55	95.75	96.56	97.09	97.33	97.44
2.4	89.76	90.73	92.36	94.13	95.59	96.66	97.37	97.83	98.07	98.15
2.5	91.12	92.04	93.54	95.15	96.45	97.38	98.04	98.35	98.55	98.62
2.6	92.32	93.18	94.56	96.02	97.17	98.00	98.46	98.81	99.03	99.08
2.7	93.39	94.18	95.44	96.73	97.74	98.40	98.89	99.19	99.32	99.36
2.8	94.33	95.06	96.20	97.33	98.21	98.79	99.21	99.45	99.53	99.56
2.9	95.16	95.82	96.85	97.85	98.55	99.09	99.42	99.60	99.68	99.71
3.0	95.89	96.46	97.38	98.24	98.91	99.29	99.58	99.71	99.78	99.80
3.1	96.50	97.06	97.85	98.56	99.18	99.48	99.70	99.80	99.86	99.87
3.2	97.06	97.51	98.21	98.89	99.22	99.61	99.79	99.87	99.91	99.92
3.3	97.50	97.96	98.52	99.15	99.52	99.72	99.85	99.91	99.94	99.95
3.4	97.94	98.26	98.82	99.33	99.63	99.79	99.90	99.94	99.96	99.97
3.5	98.23	98.54	99.08	99.48	99.72	99.85	99.93	99.96	99.97	99.98
3.6	98.50	98.83	99.27	99.60	99.83	99.89	99.95	99.97	99.98	
3.7	98.78	99.08	99.42	99.69	99.85	99.92	99.97	99.98		
3.8	99.04	99.24	99.54	99.76	99.89	99.95	99.98			
3.9	99.21	99.39	99.63	99.82	99.92	99.97				
4.0	99.36	99.51	99.71	99.86	99.94	99.98				
4.1	99.48	99.61	99.78	99.90	99.95					
4.2	99.58	99.69	99.83	99.93	99.97					
4.3	99.66	99.75	99.87	99.94	99.98					
4.4	99.72	99.80	99.90	99.96						
4.5	99.78	99.84	99.92	99.97						
4.6	99.83	99.88	99.94	99.98						
4.7	99.86	99.91	99.96							
4.8	99.89	99.93	99.97							
4.9	99.92	99.94	99.98							
5.0	99.93	99.96								
5.1	99.95	99.97								
5.2	99.96	99.97								
5.3	99.97	99.98								
5.4	99.98									
5.5	99.96									

DISTRIBUTION:

J. Benson
R. Bidlack
W. Dale
F. Deis
J. Dobkins
E. Donovan
R. Graves
E. Kells
S. Kuca
W. Langley
F. Leopard
G. Ludick
A. Moon
C. Ostner
L. Weiner
F. Wilson
R. Wunderman

REFERENCES

1. Mathematical Background and Programming Aids for the Physical Vulnerability System for Nuclear Weapons, DI-550-27-54, DIA, (1974).

2. Physical Vulnerability Handbook - Nuclear Weapons, AP-550-1-2INT, DIA, (1969).

3. Target Data Inventory Handbook, DDI-2600-815-76, DIA (1976).

4. Personnel Fatality/Casualty Predictions, DIA Memorandum C-4042, (1975).

5. DIA Message, "Proposed VN Coding for Crater-Related Damage Mechanisms, (1976).