MATHEMATICAL FORMULATION

OF THE

ARSENAL EXCHANGE MODEL

(REVISION 7)

JUNE 1973

MARTIN MARIETTA CORPORATION
DENVER DIVISION
DENVER, COLORADO 80201

MATHEMATICAL FORMULATION

OF THE

ARSENAL EXCHANGE MODEL

(REVISION 7)

JUNE 1973

BY:

J. F. BOZOVICH
P. L. GROOVER
P. C. RYAN
J. A. BATTILEGA

APPROVED: Frederic D. Selbie Jr
FREDERIC D. SELBIE, JR.
DEPUTY DIRECTOR
DEFENSE SYSTEMS

MARTIN MARIETTA CORPORATION
DENVER DIVISION
DENVER, COLORADO 80201

FOREWORD

This document contains a description of a computerized strategic war game as developed by funding provided by the Martin Marietta Corporation and the United States Government. The document represents work performed during various contracts since 1966. This continuously updated compilation of work done under several contracts is maintained as new work is performed because it provides the most convenient "one-source" document for all the various users of AEM.

During the previous twelve months work has been conducted in several areas relating to this document. Fundamental modifications accomp;ished this year in AEM are described in Chapters IV-L, M, T and U.

TABLE OF CONTENTS

PAGE

I. INTRODUCTION

During the past several years strategic systems analysts at the Martin Marietta Corporation have been developing a family of system evaluation methodologies and applying those methodologies to a wide variety of problems. These developments have resulted in the arsenal exchange model to be described in this paper. The key feature of that model, namely the capability to analyze various forms of weapon exchanges between two opponents, is a product of the developed realization that adequate analyses must include the two opponents - two arsenals fact of life.

For example, comparison of the relative strengths of two countries must be based on the total resources of each and, of equal importance, on their objectives in using those resources. The outcome of an exchange is a strong function of which side strikes first and which targets he attacks. On the other hand, the outcome is also a function of the total target complex to be attacked and the capability of existing systems to attain all desired objectives on those targets. Thus, any comparison of opponents depends upon a complex interaction between existing arsenal characteristics, each opponent's objectives and potential target characteristics.

Development of a computerized war game model can take a wide variety of forms. The model described in this report is of the expected value type. This approach was chosen because it was felt that a simulation model would be self-defeating in the sense that computation time might preclude analysis of a variety of scenarios, arsenals, target characteristics and objectives. Of equal importance was the feeling that the outcome of an exchange is most dependent upon strategies, objectives, and the nature of all resources engaging in the duel. The nature of simulations can often make analysis of such factors difficult to implement.

Such considerations led to a model which uses descriptions of the systems and information possessed by two opponents, their objectives in the exchange and the industrial resources each is attempting to protect. With these inputs the model conducts the exchanges and provides resource management in such a way that each opponent maximizes the level of his objectives he attains as constrained by his resources and the behavior of his opponent. The model is distinguished by its ability to:

1) Analyze different types and levels of exchanges.
2) Accept a variety of forms of strike objectives.
3) Include in the analysis impacts of uncertainties in each side's knowledge about his opponent.
4) Output measures of the utility of all systems taking part in the exchanges.

To aid in the readers' understanding of the model and its capabilities, the presentation is in four distinct phases. Presented first is a section describing the general concepts involved in the model. This acquaints the reader with several key thoughts which are important to understanding of the model. This section is followed by a hypothetical, but fairly representative, set of examples of different types of analyses and results which can be obtained. Next, several mathematical problems which turned out to be the key hurdles to be overcome in implementing the basic concepts are discussed. Finally, there is a section describing the method of actually using and interpreting the model. This section describes the complete set of input parameters and the method for manipulating those parameters to obtain any given type of result.

II. THE MODEL

A. BASIC ASSUMPTIONS

Two primary assumptions inherent in the logic of the model are that

1) A real exchange can be approximated as a sequence of strikes alternating between the opponents, and that

2) All targets can be grouped into force, other military or value categories. (For convenience of presentation all non-force, military targets will be considered as belonging to the value target list.)

It is recognized that in a real exchange one side could be striking at his opponent while his opponent was striking at him. As the model was in its early stages of development, however, it was felt that a model utilizing a definite sequence of strikes would be of considerable aid in analyzing arsenal exchanges and possibly could be expanded later to allow for such overlapping strikes.

The major problems in a sequential strike model revolve around the choice of appropriate targets for the weapons allocated in any given strike. This problem of force management, or the classical weapon-to-target allocation problem, can be solved mathematically if it is possible to place relative values on all targets. Unfortunately the value of an ICBM site relative to a city is not at all obvious. Our approach to solving the force target vs. value target values made the second major assumption necessary.

In the division of all targets into two categories the classical definitions are utilized. A force target is defined to be a point, or area which possesses a force which can retaliate against you if it survived. A value target is one which represents in some measurable way a portion of the industrial capacity, or non-retaliatory military capacity of your opponent. These definitions allow a potential target to be viewed as one which represents an immediate threat to yourself or one which represents a longer term, potential threat.

Given this target classification system it is possible to proceed to a system of relative values for all targets, given that certain forms of strike objectives can be stated. For example, an acceptable objective is one expressed in the form of a relative desirability of minimizing damage to yourself (damage limitation) compared to achieving damage against your opponents' industrial and non-retaliatory military capability (assured destruction).

B. A BASIC SCENARIO

To clarify these concepts of sequential strikes, strike objectives and the two special classes of targets, consider a simple exchange which is a massive first strike with retaliation. In this exchange, RED prepares the maximum possible portion of his force for a strike on BLUE. After the strike, BLUE takes whatever survivors he has left and retaliates with them. (In reality several strikes of diminishing size can occur but for our purposes it will be best to consider only the first strike by each side.)

Such an idealized exchange can be viewed in diagram form as represented in Figure 1. In that figure, the circles represent the two target categories for each side, F for force and V for value targets. The arrows represent a flow of weapons to the opponents' targets.

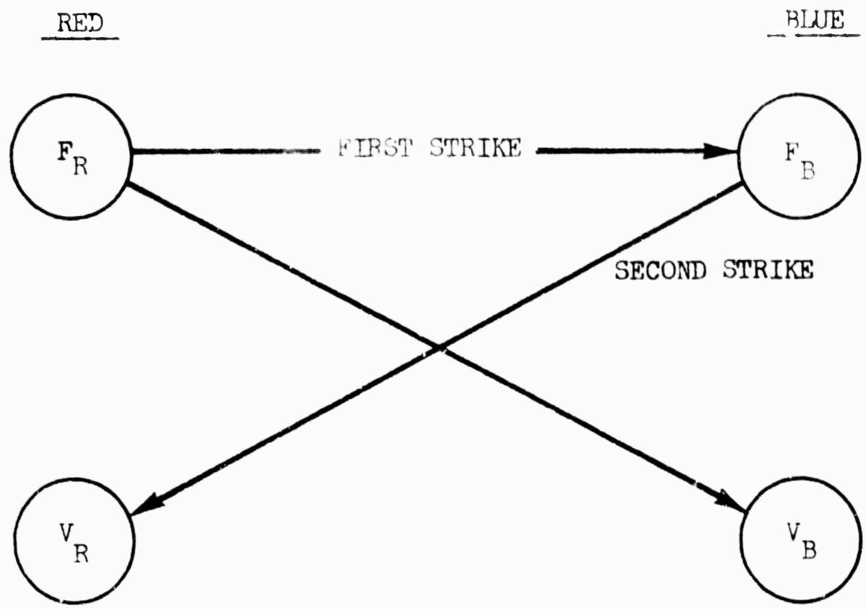


FIGURE 1: MASSIVE FIRST STRIKE WITH RETALIATION

The final result of such a two-move exchange is some damage done to the value targets, V_R and V_B, of each side. The damage done to the forces, F_B of BLUE are reflected in a reduction in the damage which could have occurred to V_R.

In setting up the first strike RED would have to choose between attacking force and value targets by stating his relative preference for damage limitation and assured destruction. RED must then utilize that preference in combination with an analysis of the individual targets of BLUE to allocate each of his weapons to some target.

In RED's weapon-to-target allocation he would have to consider the fact that not all targets are identical. They vary in their vulnerabilities, the value targets have variable industrial capacities and the force targets have variable retaliation potentialities. In addition, RED will have a mixed arsenal with weapons of varying capabilities.

The necessary ingredients for this allocation are a description of the capability of every weapon, e.g. CEP, reliability, yield, and the vulnerability of every target, e.g. area, hardness, defense level. In addition, the relative industrial capacity, or value, of every value target must be specified. Normally these values are based on relative populations, manufacturing value added to the economy by that city, or some similar measure.

A basic methodology, which is described in Section IV, was developed for this weapon-to-target allocation problem for the general mixed arsenal to mixed target system case. The methodology is based on a paper by Everett[1] in which a generalized Lagrange multiplier technique is described. The resultant program can optimally allocate up to 25 types of weapons against up to 50 classes of targets in 10 to 60 seconds on a third generation computer. In this program all weapons of identical characteristics are grouped into types, e.g. all Minuteman II would be a type, and targets of identical, or very similar, characteristics are grouped into classes, e.g. all cities of 50,000 to 100,000 might be a class. This grouping causes very little loss in accuracy and considerably shortens computation time.

A natural by-product of the Lagrange multiplier technique is an output of the marginal utility, or effectiveness of every system taking part in the exchange. For example, if RED has 5 types of weapons, the Lagrange multipliers indicate how much the damage levels could be changed by RED if he possessed one more of any one of the types.

---

(1) Everett, H., "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources." <u>Operations Research</u>. Vol. II, pp 399-417, 1963.

C. WEAPON RELATIVE VALUES

Before RED can allocate his weapons, it is also necessary for him to determine relative values for BLUE's forces. Those values must be chosen so that the weapon allocation by RED will achieve the maximum measure of the game to RED under the assumption that BLUE utilizes all of his surviving forces to maximize damage to RED's value targets.

To understand how strike objectives can lead to relative values for force targets consider Figure 2.

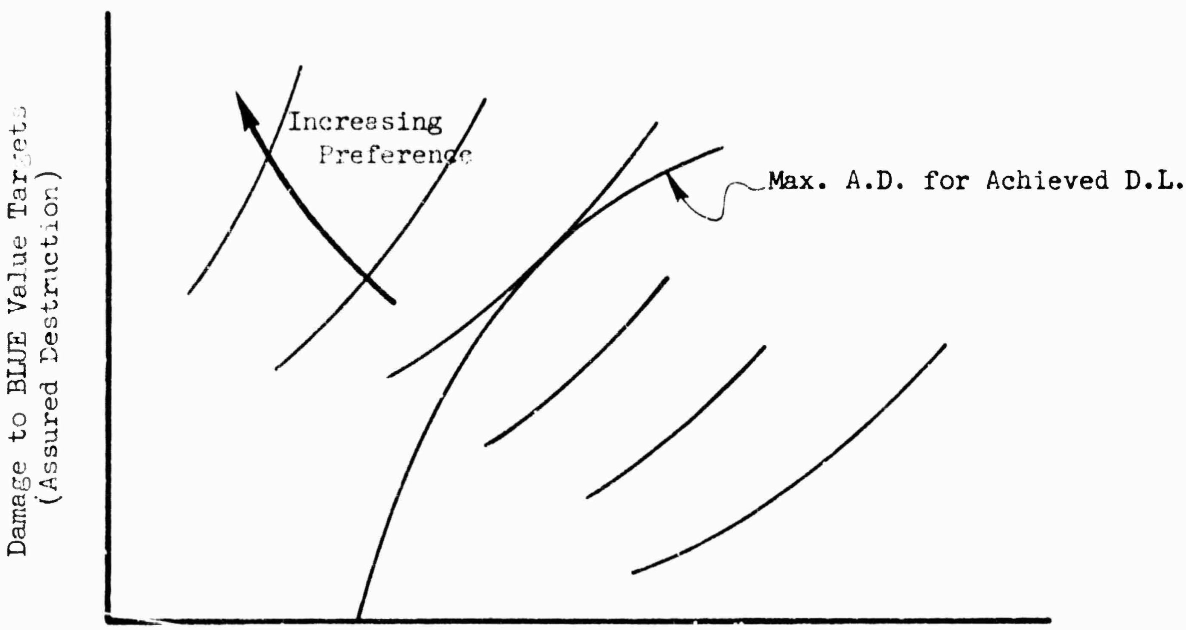


FIGURE 2: POSSIBLE STRATEGIES FOR RED

In that figure a family of constant preference contours is shown. These contours are one way of stating how RED views potential interchanges between less surviving value of his own to obtain increased damage to BLUE.

On any one contour it is hypothesized that RED is indifferent to the various situations represented. For example, if one point on one contour is 52% damage to RED and 76% to BLUE while another point on the same contour represents 65% to RED and 84% to BLUE it means that both situations are equally acceptable to RED. However, another contour, generally assumed to be up and to the left on Figure 2 would represent a separate set of situations which are all more desirable to RED.

A key characteristic of any contour is its slope at a given point. The slope indicates, for that region, the acceptable exchange rate between additional damage to RED and additional damage to BLUE. If the slopes, or value exchange ratio, of all the contours were one it would simply say that RED would be indifferent to accepting 1% more damage in exchange for 1% increased damage to BLUE.

Also indicated is the envelope of the maximum possible assured destruction level for achievable damage limitation levels. The exact location of this curve is a function of all the characteristics of the arsenals and targets and the allocations of those arsenals. The tangent point between that curve and one member of the preference contours is the best attainable result for RED given a fixed arsenal and given the stated preference contours.

The problem is to perform all resource management in such a manner that the specific strategies required to arrive at the tangent point are determined. One approach to this problem is to develop a number of points on the max A.D. envelope one point at a time. Given enough points on the curve, any set of preference contours can be matched to the total curve to arrive at the desired solution, or tangency point.

Development of any single point on the Max A.D. envelope can be obtained by assuming some slope, K of interest and using the relative values for force targets as the mechanism for obtaining the desired strategies that will achieve the point on the envelope with that slope.

In this process, which is described in more detail in Section IV, initial estimates for relative values for BLUE's forces are made. Given those values, that set of weapon-to-target allocations which maximizes the sum of the value destroyed on all targets (force and value) is found. This results in certain surviving weapons for BLUE which he can use to maximize value destroyed on RED.

The Lagrange multiplier method is used in the allocation of BLUE's survivors so that the marginal utility of each of his weapon types is available. A logical next set of estimates for a relative value for BLUE's forces can be shown to be those marginal utilities multiplied by K. The K weighting factor is the direct tie to the slope of the preference contours.

This specific choice for a weapon relative value can be visualized as the expected reduction in RED's payoff if BLUE has one more surviving weapon. Thus, when a new attempt is made by RED to optimize his attack, the choice between attacking force or value targets can be based on the expected increase in RED's payoff if he can reduce BLUE's surviving weapons. The optimum balance between force and value attacks, as biased by the preference ratio K, is that level where additional counterforce attacks results in less RED value saved than the additional BLUE value which could be destroyed by the same weapons. Any individual RED weapon will attack a value target if the BLUE value it can destroy is more than K times the value which could be saved if a BLUE force target was attacked.

Experience has shown that judicious use of the original value estimate set and cycling through the improved estimates for the marginal utility of BLUE's weapons will result in optimal, or near optimal force management. Mathematical proof that global optima are not overlooked occasionally is not available.

D. OTHER EXCHANGE TYPES

The model is designed to handle a number of other exchange types. For example, RED might desire to attack only counterforce targets in a first strike while maintaining a reserve force to deter BLUE from continuing the exchange. In this type of exchange RED must not only allocate weapons-to-targets but must also decide which weapons to hold in reserve.

The conflict in this strategy is that it would be desirable to use a high portion of the force in a first strike in an attempt to seriously weaken BLUE's forces but it is also desirable to maintain an adequate reserve level. Assuming that BLUE's exact response could never be predicted, that response which imposes the maximum requirements on the reserve force can be assumed for the purpose of resolving the conflict. It is desirable then to choose that reserve force for RED which presents the maximum possible deterrence to BLUE against retaliating against RED's value targets. Thus, for the purpose of choosing the best reserve, the nature of the exchange after his first strike is assumed by RED to be as diagrammed in Figure 3.

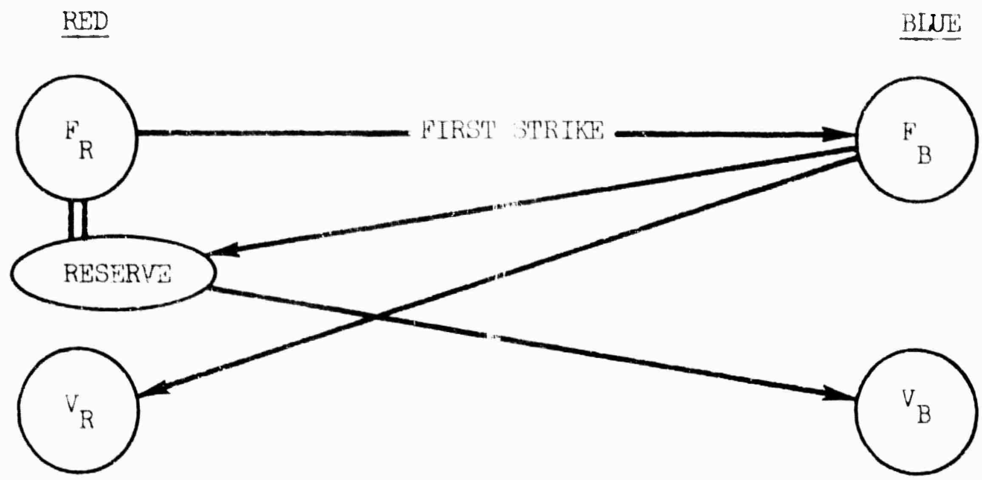


FIGURE 3: COUNTERFORCE FIRST STRIKE

Given that RED assumes the exchange might turn out as indicated in Figure 3, it is necessary to determine, for RED, that reserve force which maximizes the value of the game as measured by his preferences. The same techniques as were described previously are used in making all weapon-to-target allocations, and setting values on weapons. In this case both BLUE and RED must determine such values. (The optimum reserve force is determined by a method which is unimportant to the current discussion so it is only described in detail in Section IV.)

After RED has chosen a reserve and carried out a strike, BLUE can retaliate as indicated in Figure 3 or with a pure counterforce strike of his own. In the latter case BLUE would then have to choose a reserve force. Reversal of the roles indicated in Figure 3 would then occur and BLUE would use a similar logic in picking his reserve force. The model has the capability to analyze any number of such counterforce strikes. At each stage the optimum reserve is chosen based on the assumption that the next retaliation might be against both force and value targets.

An additional type of exchange the model can consider is described in Figure 4.

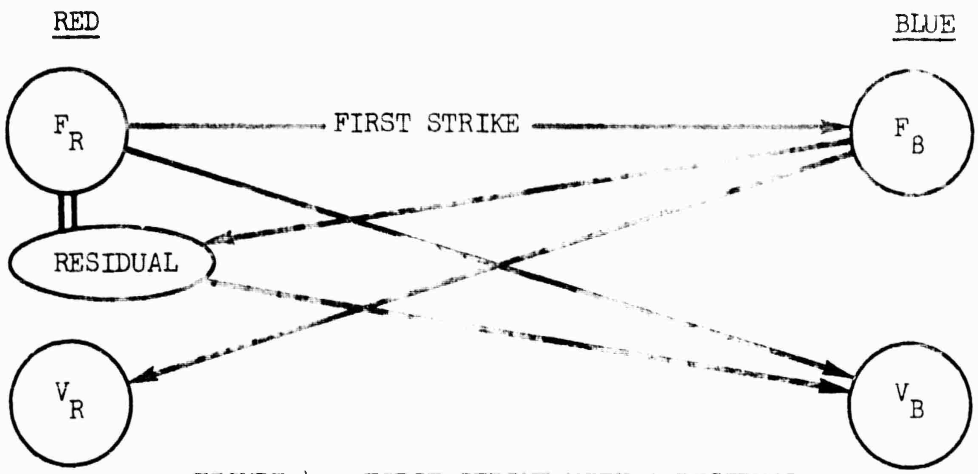


FIGURE 4: FIRST STRIKE WITH A RESIDUAL

This exchange is a more realistic version of that in Figure 1 in the sense that residual weapons possessed by RED are allowed to exist and be targeted.

The major problem in this scenario is planning the countervalue portion of RED's first strike in such a way that he destroys a maximum amount of BLUE industrial value with the <u>combination</u> of his first and second strikes on BLUE value targets. The methodology for accomplishing this aspect of this scenario is described in Section IV.

E. IMPACT OF UNCERTAINTIES

The sequential strike concept can be utilized to go considerably beyond the simple exchanges just discussed. For example, the flexibility exists to analyze the impact of uncertainties. In essence, when one opponent is setting up a strike he must base that strike on his current information concerning his status, the status of his opponent and an estimate of how his opponent might retaliate. Once that strike has been launched, the play in the duel passes to his opponent and the opponent has the same task facing him. In addition, if the strike was set up based on only partially correct information, or assumptions, there can be a difference between the intended and actual effect of the strike.

This concept that a strike is set up based on an estimated situation, but that the actual effect can be different than the intended effect gives the power to the sequential strike method. For example, assume that all elements engaging in an exchange are continually described by three different descriptions which cover:

1) Actual characteristics and status of forces and targets,
2) Characteristics and status of forces and targets as estimated by the owner of those elements, and
3) Characteristics and status of forces and targets as estimated by the opponent of the actual owner of those elements.

For an example of the three types of characteristic , consider a hypothetical ICBM that RED might possess. It might have an actual CEP of 1 n.mi., reliability of .75 and yield of 7.6 MT. But, since no testing program is perfect, RED's estimate of those characteristics might be .6, .80 and 7.0. At the same time, BLUE might be basing his estima[illegible] minimum intelligence data and think the characteristics are .6, .85 an[illegible]

Use of the sequential nature of the program to allow analysis of uncertainties has been implemented in some detail. The procedures use the above described three sets of characteristics for each opponent. The utilization of these three independent sets of parameters then is as follows.

In the previously described types of exchanges, RED's first strike would be based on his estimate of his own situation and his estimate of what BLUE's situation is. Then, before BLUE responds a computation is made, by the model, of the actual damage RED's strike did to BLUE utilizing the actual characteristics, which neither side might know.

BLUE's response would then be based on his estimate of the situation and his actual surviving resources. The sequence of strikes can continue, always basing weapon-to-target allocations on estimated characteristics and basing responses on actual damage done to the opponent and on the opponent's estimate of the current situation.

In order to diagram such a case assume that each opponent knows his own status perfectly and that the scenario to be studied is like that in Figure 3. The sequence can be visualized as in Figure 5.

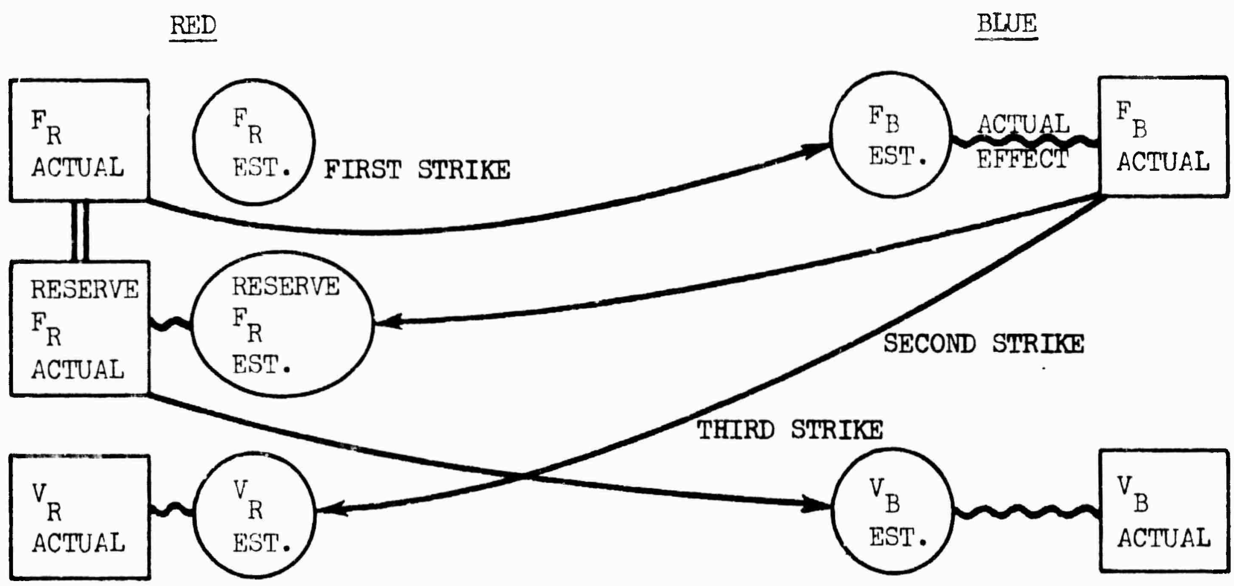


FIGURE 5: COUNTERFORCE STRIKE USING ESTIMATES

In Figure 5 the squares indicate actual characteristics and the circles estimated characteristics. It shows RED setting up a first strike based on an estimate for F_B. The wavy line then indicates that the strike will have some effect on F_B actual. BLUE then would set up a retaliation based on estimates. Note, especially that BLUE would have to estimate RED's reserve. Finally, RED performs his second strike based on his actual survivors from BLUE's attack and his estimate of V_B. At the completion of such a sequence there will be some actual damage levels accepted by each side. The question is - how would those differ if either side had a better estimate of his opponent?

By utilizing the three different sets of descriptions for all elements in the model it is possible to study almost any conceivable topic concerned with optimum force management. For example, some items are as follows:

1) What is it worth to know your own weapon characteristics perfectly?

2) What impact does it make to not know the exact characteristics of your opponents weapons?

3) What is the penalty for not realizing that your opponent has an effective ASW system?

4) What is a system capable of detecting the vacated bases (empty holes) of your opponent worth?

5) What is the effect of the objectives of each opponent not being diametrically opposed?

By appropriate control of the three basic sets of data it is possible to investigate the case of launching at an opponent and having him vacating bases while your weapons are in-flight. It is also possible to investigate the effect of numerous small exchanges rather than a few massive ones with each opponent's viewpoints and objectives changing during the strikes. The only limitation is that all objectives be relatable in some way to a relative preference between damage limitation and assured destruction.

F. CHOICE OF PREFERENCE CONTOURS

A key element of this model is the set of preference contours for each opponent. Unfortunately, a universally acceptable set of contours is not available now and likely never will be. To overcome this problem a number of concepts have been considered before arriving at the one normally used and to be described in this paper.

Considerable logic for choosing various contour families is possible but most of them involve two key thoughts. First, each opponent tends to want to minimize damage to himself, but given obvious limits on that it is reasonable to maximize the difference between the damage to his opponent and the damage to himself. Second, there is usually a realization that once an opponent is destroyed to some reasonable level it is rather foolish to continue destruction on him since he is already an ineffective power.

Combining these two concepts leads to the contour family depicted in Figure 6.

Shown in Figure 6 is a set of parallel contours with a distinct change from a slope value of one to an infinite slope at some fixed level of damage to BLUE for any level of damage to RED. In effect, this set of contours results in maximizing the arithmetic difference between the damage levels of two opponents with some upper limit placed on the allowed damage to your opponent.

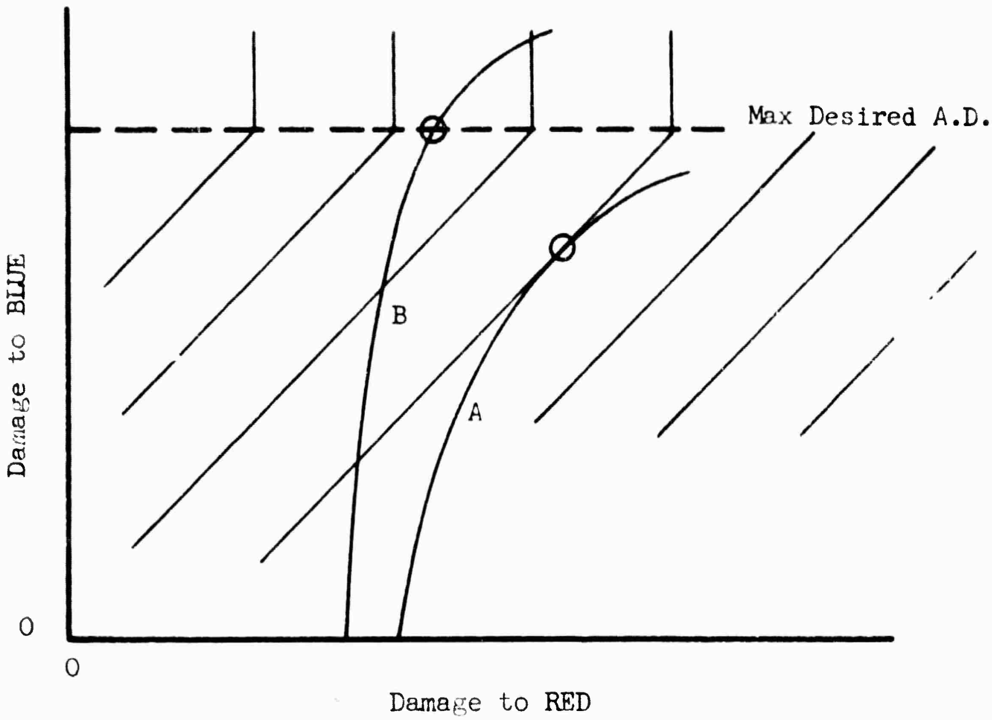


FIGURE 6: PREFERENCE CONTOURS

Application of such a set of contours develops weapon allocations that are very rational. It has been found that use of a max desired assured destruction level eliminates the situations where both sides get destroyed to an extremely high level. Also, the final solution does not appear to be very sensitive to the unity slope assumption. Generally the max A.D. curve is rather steep so that use of other slopes, e.g. 2 does not shift the tangent point very much vertically.

Indicated on Figure 6 are two max A.D. envelopes. If envelope A existed, the optimum point would occur at the point on the envelope with slope equal to one. If B existed, the optimum point would occur at the point on the envelope with damage to BLUE equal to the maximum desired by RED. Given any set of initial arsenals the model can determine such optimum solutions in one run.

Of course, if this simplified contour family is not acceptable, the program can be used to develop the max A.D. envelope a point at a time so that any contour family can be matched to the envelope. In essence, the true capability of the program is reflected in the fact that it can perform the force management function so that for any attainable damage to an opponent a minimum damage occurs to yourself.

III. TYPICAL MODEL APPLICATIONS

To demonstrate the form of typical results obtained from the model a number of topics will be investigated concerning hypothetical RED and BLUE opponents. The arsenals of each and all characteristics necessary to the running of the model are described in Tables I & II. The industrial target complexes are described in Tables III & IV. The target values in Tables III & IV are arbitrary but they do parallel those which could be based on relative population or some typical measure of value. In addition, it will be assumed that each side launches 50% of all bombers on warning. Also, that the availability numbers, which are the fraction of the force considered to be ready for launch, include such items as the on-station fraction for submarines. It will be assumed that all bombers have a probability of .5 of penetrating perimeter defenses of his opponent. (More sophisticated bomber and missile defense assumptions could be made but they would not add significantly to the understanding of the concept of the model.)

TABLE 1: RED ARSENAL CHARACTERISTICS

SYSTEM	TYPE	NUMBER	RELIABILITY	AVAILABILITY	CEP	PAYLOAD
I	ICBM 200 PSI BASE	800	.9	.7	.5	5. MT
II	ICBM 100 PSI BASE	200	.8	.6	1.	25.
III	SLBM 20 SUBS	200	.8	.6	1.	1.
IV	BOMBER 20 BASES	200	.9	.9	.25	10.

TABLE 2: BLUE ARSENAL CHARACTERISTICS

SYSTEM	TYPE	NUMBER	RELIABILITY	AVAILABILITY	CEP	PAYLOAD
A	ICBM 500 PSI BASE	650	.9	.8	0.4	2. MT
B	ICBM 100 PSI BASE	250	.8	.7	0.8	10. MT
C	SLBM 30 SUBS	300	.8	.5	1.0	2. MT
D	BOMBER 40 BASES	400	.95	.8	.25	10. MT

TABLE 3: RED INDUSTRIAL CHARACTERISTICS

CLASS	NUMBER IN CLASS	VALUE OF EACH MEMBER	CUMULATIVE VALUE	AREA
1	1	14.0	14	300 Sq. N. Mi.
2	3	8.0	38	171
3	9	5.0	83	107
4	27	3.0	164	64
5	81	2.0	326	43
6	243	1.2	618	26
7	536	.71	1000	15
TOTAL	900			

TABLE 4: BLUE INDUSTRIAL CHARACTERISTICS

CLASS	NUMBER IN CLASS	VALUE OF EACH MEMBER	CUMULATIVE VALUE	AREA
1	1	24.0	24	1000 Sq. N. Mi.
2	2	18.0	60	430
3	4	12.0	108	290
4	8	9.0	180	216
5	16	6.0	276	144
6	32	4.0	404	96
7	64	2.5	564	60
8	128	1.5	756	36
9	244	1.0	1000	24
TOTAL	500			

A. <u>ASSURED DESTRUCTION VS. DAMAGE LIMITATION</u>

For the given arsenals the exchange type depicted in Figure 1 was analyzed. The resultant max assured destruction envelope is given in Figure 7. Several points should be noted concerning this result. First, note the relatively rapid rise in BLUE damage as RED's damage level increases. This has been found to be very typical when missile defense does not exist. Second, the intercept at a RED damage level of 54% represents the damage which BLUE can do even if RED attacks pure counterforce. Third, the strategy which results if RED desires to maximize the difference between BLUE and RED damage achieves 87.3% damage on BLUE and 70.8% damage on RED.

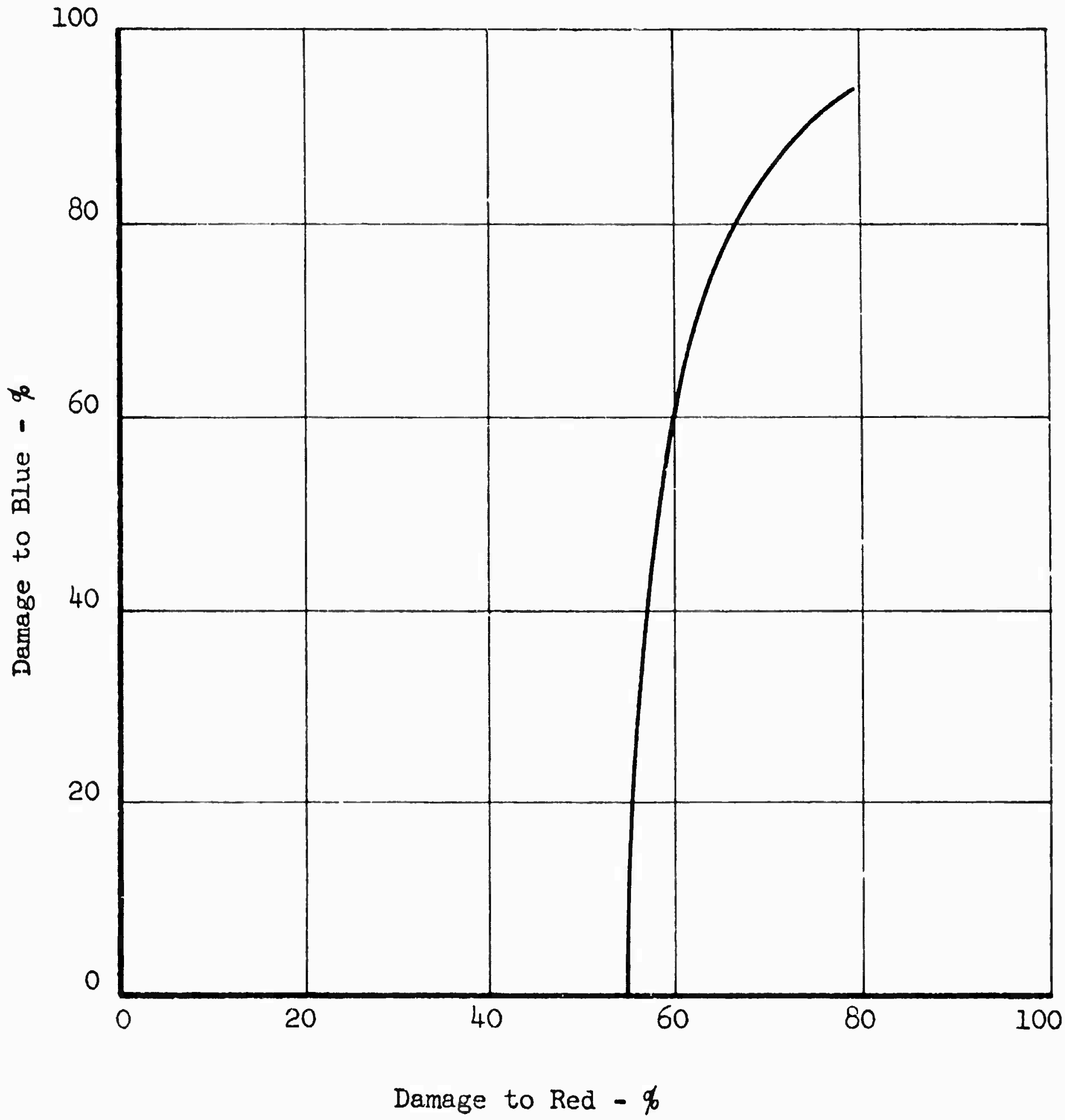


FIGURE 7: ASSURED DESTRUCTION vs. DAMAGE LIMITATION

If RED desires to maximize the difference without exceeding 70% damage to BLUE the optimum allocation of his available weapons are as shown in Table 5. The final result of these allocations, after BLUE optimally allocates his expected survivors is 70% damage to BLUE and 62.8% damage to RED.

TABLE 5: EXAMPLE OF RED WEAPON ALLOCATIONS

BLUE TARGET CATEGORY	TOTAL NUMBER	EXPECTED SURVIVORS	NUMBER ATTACKED	NUMBER OF RED WEAPONS PER TARGET			
				WPN I	WPN II	WPN III	WPN IV
WPN TYPE A	650	614.	71	1	0	0	0
B	250	41.23	250	1	0	0	0
C	30	30.0	0	0	0	0	0
D	40	1.75	40	0	0	2	0
CITY TYPE 1	1	.03	1	0	10	0	0
2	2	.07	2	0	5	0	0
3	4	.42	4	0	3	0	0
4	8	.42	8	0	3	0	0
5	16	1.87	16	0	2	0	0
6	32	4.91	16	0	0	0	4
			16	0	2	0	0
7	64	12.76	38	0	0	0	3
			25	2	0	0	0
			1	1	0	0	2
8	128	33.63	128	1	0	0	0
9	244	175.90	60	1	0	0	0
			40	0	0	1	0

RED allocated almost exactly one-half of his available, non-bomber warheads to the counterforce role. In this allocation all bombers were arbitrarily restricted from the counterforce role because of the problem of strike timing.

B. SYSTEM COMPARISONS

To demonstrate the problems associated with performing system evaluations without a two-sided game the results of this case were compared with an alternate approach. Included in Table 6 are the results of that comparison.

The various measures of effectiveness chosen for RED weapons are those suggested by weapons effects theory and alternates which play down the effect of CEP or yield. The alternates were chosen since it might be suspected that in certain targets those characteristics would have a lesser effect.

Also included is the weapon relative value as obtained from this model. They are based on the Lagrange multipliers obtained when the RED weapons were optimally allocated. The multipliers represent the marginal utility of each weapon so they do describe the true effectiveness of the next additional weapon of each type.

TABLE 6: COMPARISON OF MEASURES OF EFFECTIVENESS

WEAPON TYPE	RELATIVE SYSTEM VALUE FOR GIVEN MEASURE OF EFFECTIVENESS*						
	LAGRANGE MULTIPLIERS	$\frac{R.Y^{2/3}}{CEP^2}$	$\frac{R.Y^{2/3}}{CEP}$	$R.Y^{2/3}$	$R.Y^{1/3}$	$\frac{R.Y^{1/3}}{CEP}$	$\frac{R.Y}{CEP}$
I	1.	1.	1.	1.	1.	1.	1.
II	1.26	.65	1.30	2.60	1.52	.76	2.22
III	.60	.08	.15	.30	[illegible]	.26	[illegible]
IV	.44	3.17	[illegible]	.79	[illegible]	[illegible]	[illegible]

* R = weapon reliability Y = weapon yield

It can be seen that none of the simple measures duplicate the multiplier result. The reason no one measure is adequate is that different weapons would be optimally allocated against different targets so that no single weapons effect parameter is equally dominant in all cases.

For example, in this case additional weapons of Type I would be used in a counterforce role against a hardened target where CEP is important while additional weapons of Type III would be used against a value target where CEP is much less important. This use would release some other weapon for use against a force target. Thus, purchase of one weapon can feasibly affect the use of many other weapons.

Of equal interest is the case where BLUE is doing a system comparison. His problem is like that of RED but compounded by the necessity to estimate the survivability of his weapons. In this case, Table 5 indicates additional weapons of Types A and C would not be attacked but those of Types B & D likely would be. If a different RED arsenal was to be assumed, these conclusions could be changed. In any case, the probability of survival of any additional weapon can be determined by use of the exchange model.

Table 7 describes the set of relative values for all BLUE weapons. Included is a set based on the multipliers alone and a set which factors in survivability as indicated by the model. Note how the values of Systems B & D degrade when survivability is included. As RED increases his arsenals, those values do change. For example, if RED increases his numbers of Type I from 800 to 2000, the relative values are as shown. As the 2000 level is approached, the submarine-based missiles begin to dominate because of their assumed invulnerability.

TABLE 7: BLUE SYSTEM VALUES

WEAPON TYPE	MULTIPLIERS	MULTIPLIERS SURVIVABILITY AT GIVEN RED FORCE LEVEL			
		800	1200	1600	2000
A	1.00	1.00	1.00	1.00	1.00
B	1.26	.78	.78	.78	.66
C	.95	.95	.95	.95	1.26
D	.57	.47	.41	.41	.47

C. OTHER SCENARIOS

As examples of the model application to another scenario, a case similar to that described by Figure 3 was analyzed. The only difference is that a BLUE first strike was assumed.

In the BLUE pure counterforce first strike, the preferred strategy for BLUE was 220 weapons of Type A allocated one to each of RED's Types II and IV. The final damage levels resulting were 83.6% to BLUE and 62.7% to RED. As an example of an alternate strategy for BLUE, he could allocate 490 weapons of Type A, which results in a damage level of 73% to BLUE and 45% to RED. Notice that this latter strategy does not achieve as large a difference between RED and BLUE damage as the preferred strategy does.

D. ANALYSIS OF UNCERTAINTIES

One of the most vital questions in a model of this type is that of uncertainties in assumptions. In a broader sense, this can be equated to the effect of uncertainties, which will always exist, in estimating the current situation of yourself and your opponent.

Some topics of this nature have been investigated to demonstrate the application of this model to such questions. Among those to be discussed here are the following:

1) Is it better to underestimate or overestimate capabilities?

2) Is information concerning your opponent's characteristics worth as much as information concerning yourself?

3) What is the relative payoff between additional offense and better information?

This analysis will be based on the hypothetical arsenals, but the general form of the conclusions is believed to be valid elsewhere.

As an example of the underestimate/overestimate question, the reliability of RED weapons was investigated. The assumed scenario is like Figure 1. The assumed objective for RED is to maximize the difference in damage levels but not to exceed 70% damage on BLUE. The cases considered are described in Table 8.

TABLE 8: EFFECT OF ERRORS IN RELIABILITY ESTIMATES

CASE	ACTUAL RELIABILITIES FOR WEAPONS I,II,III,IV*	RELIABILITIES AS ESTIMATED BY RED	FINAL DAMAGE LEVELS BLUE	FINAL DAMAGE LEVELS RED	GAME VALUE
I	.9,.8,.8,.45	.9,.8,.8,.45	70%	62.8%	7.2
II	.9,.8,.8,.45	.7,.6,.6,.25	83.2%	73.7%	-3.7
III	.7,.6,.6,.25	.7,.6,.6,.25	70%	74.3%	-4.3
IV	.7,.6,.6,.25	.9,.8,.8,.45	50.6%	66.3%	-15.7

*Bomber reliability includes the probability of penetration of perimeter defenses.

The effect of poor reliability estimates is very dramatic. For example, in comparing cases III and IV, the difference is caused purely by RED's <u>not knowing</u> his own reliability. In Case III, he knows they are low and allocates accordingly. In Case IV, he thinks they are high and consequently mis-allocates but the damage is accrued according to his actual reliability. The net effect is about 20% less damage to BLUE and 8% less damage to RED. In terms of the stated RED criteria, Case III has a value of 70 - 74.3 = -4.3 while Case IV has a value of 50.6 - 66.3 = -15.7. Thus, the error caused a large drop in the criteria.

The effect of an underestimate can be observed in Cases I and II. Again, all differences are due to mis-allocation of weapons resulting because RED's knowledge of his own reliability was not perfect. The interesting thing is that an underestimate tends to result in excessive damage to both sides while the overestimate tends to reduce damage to both sides.

If the value of the game is computed according to "maximize the difference but no value is achieved for damage exceeding 70% to BLUE," Case IV is the worst situation. But the true loss due to an underestimate is the difference between Cases I and II while the loss due to an overestimate is the difference between Cases III and IV. Thus, the underestimate caused a change of 10.9 units while the overestimate caused a change of 11.4 units. The two losses are so close that it raises a question as to the advisability of planning an actual strike using all conservative estimates. An additional factor is that this analysis indicates that overestimates might tend to reduce damage levels.

Comparison of Cases II and III raises a very interesting point. They show that, if no value is accorded excess damage to BLUE, thinking your reliability is low (Case II) when it is high produces only slightly better results than when it actually is low (Case III).

A case was also run where RED overestimated his own yield by about 33%. The net results were damage levels of 62.3% to RED and 62% to BLUE. Compared to the perfect information case of 62.8% and 70%, the big loss is in damage to BLUE.

To place the value of good reliability information in proper perspective, consider the number of additional weapons which must be purchased to make up for the imperfect information. In the case of RED underestimating his own reliability, case II, an analysis using additional model runs showed that it would require almost 400 additional Type I weapons to achieve the same damage level as the perfect reliability information case. This suggests that considerable testing expenditures might be warranted.

In contrast to the results presented in Table 8, poor estimates by RED of BLUE force characteristics did not alter the basic results. For example, an estimate by RED of a 500 psi hardness for BLUE Type II, when it was actually 100 psi, did not make RED allocations nonoptimum. This same effect holds true in the case of misestimating reliability and yield. Cases were run where RED misestimated BLUE reliabilities and yields by about the same amounts as described in Table 8. In all cases no non-optimum allocations resulted.

E. <u>VALUE OF INFORMATION</u>

An interesting circumstance is where one opponent has some bit of information which can aid in his resource management. One example is information concerning whether or not a weapon had a successful flight. Another example is the possession of information about which enemy bases have been vacated.

The possession of BLUE empty hole information was analyzed for the case of RED first strike CF/CV where he has a residual force of 25% (the scenario described by Figure 4). To obtain a basis for comparison, the scenario was run twice, once where BLUE did not have empty hole information and once where he did have the information. The results indicated that BLUE could not use the information to reduce his own damage below 70%. The 25% residual assumed for RED was not sufficiently large to attract BLUE weapons. This type of information has not been analyzed extensively enough to know whether or not the same would be true in a real world arsenal.

A case of RED first strike CF/CV where RED has flight success information was analyzed. The net effect was a drop in RED damage from 62.8% to 58.4%. To calibrate the value of this information, the results of the run indicated that it would take approximately 280 additional RED Type I weapons to achieve this same reduction in damage if the information did not exist.

IV. KEY MATHEMATICAL ANALYSES

In the course of implementing the concepts involved in this model, several key mathematical problems had to be solved. They involved the development of:

1) A methodology for optimum allocation of a mixed weapon force to a mixed set of targets.

2) A generalized routine for producing target probability of kill for a given attack level of a given type of weapon. This routine must function for area or point targets, defended and undefended targets and weapons of any characteristics.

3) A method for choosing values to place on counterforce targets so that the optimum allocation of weapons results in a maximization of larger objectives, e.g. maximize the difference in industrial damage to your opponent and industrial damage to yourself.

4) A method for choosing the optimum reserve force when the chosen scenario is a pure counterforce first strike.

5) A method for optimizing a strike when there is a maximum allowed industrial damage to an opponent.

6) A logical method for allocating countervalue weapons in a first strike when there will be a later opportunity to fire at those same targets.

7) A representation of random area defenses for both aircraft and missiles.

8) A method of analysis of an area ABM defense when the defense can preferentially defend his targets after he sees the offensive strike.

9) Approximate methods of analysis for the circumstance where the offense and ABM defense must make plans and deployments without knowing their opponents plans. (Called a pre-commit defense.)

10) A force structuring methodology for optimal distribution of a budget among a specified set of offensive and defensive options.

11) A technique for optimal deployment of a terminal ABM interceptor budget to individual value targets so that the offense damage is minimized. Included within the routine is the capability to allow for fixed investment costs and discrete battery sizes.

12) A subtractive ABM defense model.

13) An analysis method for dealing with weapon retargeting limits.

14) A method for dealing with weapon retargeting limits.

15) A mathematical procedure for generation of hedged allocations.

16) An allocation procedure for use against a list of rank-ordered targets.

17) A weapon defense module (HSD or Safeguard) option.

18) A method for aiding bomber penetration through defense suppression tactics.

19) A new method of linear programming, called generalized upper bounding, as it is used in AEM.

A. OPTIMUM WEAPON ALLOCATION PROCEDURE

In a basic paper on the application of Lagrange multipliers to the optimum allocation of resources Everett (Ref. 1) indicated that the weapon allocation problem of interest here was a natural application of the technique. Following this thought, an attempt was made to apply the generalized Lagrange multiplier method. With some extension of the basic concept, this attempt was successful.

To understand the specifics of the method, it would be worth-while to first review the general problem of optimum allocation of weapons to targets in a mathematical format.

Assume that an arsenal which is to be allocated consists of I different types of weapons with W_i units of any given type i. Also assume that the target system consists of J different targets. The objective is to optimally allocate all weapons to the targets in such a manner that the maximum total value is destroyed on the total target complex where any given target j represents a maximum value of V_j.

Before weapons can be allocated to the targets, it is necessary to know the relationship between level of damage to a target and the number of weapons of all types attacking the target. One common expression of this relationship which holds for many types of targets is as follows:

$$PK_j = 1 - \prod_{i=1}^{I} S_{ij}^{N_{ij}} \qquad \text{(A-1)}$$

where:

PK_j = probability of kill of target j when attacked by N_{ij} weapons of each of I types.

S_{ij} = probability that target j survives one shot from a weapon of type i.

For the moment let us assume that this is the form of the damage function. (It will be shown later how other damage functions can be used.)

Given the above, the weapon allocation problem can be stated in mathematical form. Using the objective of minimizing total surviving value, rather than the equivalent one of maximizing total value destroyed, it is simply to:

Choose a set of N_{ij}, the number of weapons of type i attacking target j, in such a way that the total surviving value, SV, where

$$SV = \sum_{j=1}^{J} V_j \prod_{i=1}^{I} S_{ij}^{\;N_{ij}} \qquad \text{(A-2)}$$

is minimized and the constraints on number of weapons of each type are not exceeded, or

$$\sum_{j=1}^{J} N_{ij} = W_i \text{ for } 1 \le i \le I \qquad \text{(A-3)}$$

It should be noted that the N_{ij} must be integer and that the objective function is non-linear.

Dropping this formulation for a moment, let us review the generalized Lagrange multiplier concept. Everett proved that if X^* minimizes the function $H(X) + \sum_{i=1}^{I} \lambda_i G_i(X)$ in such a way that $G_i(X^*) = B_i$ for $1 \le i \le I$ and positive λ_i^* then X^* also minimizes $H(X)$ subject to $G_i(X) \le B_i$ for $1 \le i \le I$. More importantly, he pointed out that if non-negative multipliers λ_i^* can be found by any convenient technique such that the desired X^* can be obtained the optimality conditions hold even for discontinuous and nondifferentiable functions, H(X) and G(X).

Also of importance is the fact that if the functions, H(X) and G(X) can be expressed as sums of two, or more completely independent functions the above process operates on each of the independent functions with the only connection being the constraints on total resources, B_i and the multipliers, λ_i. That is, each independent objective function, say

$H_j(X)$ where $H(X) = \sum_{j=1}^{J} H_j(X)$ and the associated constraint function,

$G_{ij}(X)$, can be operated on in such a way that X^*_j are obtained for a given

λ^*_i and if $\sum_{j=1}^{J} G_{ij}(X^*_j) = B_i$ for $1 \leq i \leq I$ then the solutions X^*_j

are optimal.

In the context of the weapon allocation problem the constraints on resources, the B_i, are obviously equivalent to W_i. The objective function H(X) is equivalent to the SV of equation (A-2) and the $H_j(X)$ are equivalent to

$V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}}$ the value surviving on target j.

Expressed in words the generalized Lagrange multiplier method applied to the weapon allocation problem is as follows. Determine weapon related multipliers λ^*_i such that if for each target j the solution $\{N_{ij}\}$ which

$$\text{minimizes} \quad V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}} + \sum_{i=1}^{I} \lambda^*_i N_{ij} \qquad \text{(A-4)}$$

also results in $\sum_{j=1}^{J} N_{ij} = W_i$ then the $\{N_{ij}\}$ describe an optimum allocation

of the weapons to the targets.

In order to solve the allocation problem in this way, two non-trivial problems must be solved. First, a technique for minimizing equation (A-4) with integral N_{ij} must be obtained. Second, a method for converging upon the desired multipliers λ^*_i must be developed.

It has developed that ideal solutions to these problems have not been found but that very acceptable approximate solutions are available. Each of these approximations will now be presented.

1. Optimal Target Strategies

Several possibilities exist in attempting to find a minimum to equation (A-4). Among those considered are:

1) Efficient enumeration of all possibilities.

2) Use of the non-integral minimum to focus the enumeration process.

3) Use of "pure" integral solutions as approximate minimums.

4) Use of a sequence of "knapsack" problems to converge on a general solution.

The last two methods were found to be somewhat useful with method (3) being by far the most practical so it is the one in current usage. Each of those two methods will now be described.

a. Optimum Pure Strategy Method

Some consideration of the problem leads to the thought that in many cases the best integral strategy might be a "pure" solution where only one of the N_{ij} is non-zero for a given j. (This is in contrast to a "mixed" strategy where more than one N_{ij} for a given j is non-zero.) This initially appeared reasonable because it was suspected that generally one type of weapon will be preferred for a given target. In such a case two, or more, weapon types will attack only to sort of "fill in," or destroy, what it isn't worth firing one more weapon of the preferred type for.

If only pure integral strategies are to be considered, minimization of equation (A-4) is quite simple. It reduces to finding "I" minimums of the form

$$\text{MIN } H = V_j \, S_{ij}^{\,N_{ij}} + \lambda_i \, N_{ij} \quad \text{for a given } i \tag{A-5}$$

and choosing the specific solution with the overall minimum.

The minimum to equation (A-5) occurs when $N_{ij} = N^*_{ij}$, the integer value for N_{ij} such that increasing by one more unit decreases $V_j S_{ij}^{\,N_{ij}}$ by an amount less than λ_i. That is

$$V_j \, S_{ij}^{\,N^*_{ij}} - V_j \, S_{ij}^{\,N^*_{ij}+1} < \lambda_i \tag{A-6}$$

Solving this equation for N^*_{ij} results in

$$N^*_{ij} > \frac{\log(\lambda_i) - \log(V_j) - \log(1-S_{ij})}{\log(S_{ij})} \tag{A-7}$$

But we are interested in the smallest integer which just makes equation (A-6) true. Therefore the N^*_{ij} of interest is the next larger integer above the right hand value of equation (A-7). Or,

$$N^*_{ij} = \left\lfloor 1 + \frac{\log(\lambda_i) - \log(V_j) - \log(1-S_{ij})}{\log(S_{ij})} \right\rfloor \tag{A-8}$$

where:

$\lfloor A \rfloor$ = the largest integer contained in A

The concept for finding an approximate solution to equation (A-4) is therefore to find solutions for each i by use of equation (A-8) and to choose the best "pure" strategy for a given target, j. If such strategies are found for every target for a given set of

λ_i and if the totals of $\sum_{j=1}^{J} N_{ij}$ meet the constraints, an

approximate allocation optimum will be attained.

The obvious question is--how approximate is the answer obtained by use of "pure" strategies? The result is exact if it is desired to fire only one type of weapon at any single target. In the more general case, though, an approximation does exist and there is no general answer to the question.

As the most direct method of determining the degree of approximation <u>for any individual case</u>, Everett's "Epsilon Theorem" is useful. That theorem (Ref. 1) states that the error involved in using an approximate minimization of the Lagrangian (equation (A-4)) is no larger than the difference between the value of the absolute minimum and the value of the Lagrangian for the strategy used.

It is not possible to find the value of the Lagrangian at the minimum but a lower bound can be obtained. One obvious bound is the value of the Lagrangian if the N_{ij} are allowed to be non-integer. However, this bound can be tightened somewhat by the following logic.

The first point is that we are looking for a bound on the value of the Lagrangian for the best "mixed" strategy. This automatically says that more than one of the N_{ij} must be non-zero. Secondly, it is true that if non-integer N_{ij} are allowed there is one weapon which will achieve as low, or lower a Lagrangian in a pure strategy as any mixed strategy. This best non-integer weapon is the one where the value of $\left[-\frac{\log(S_{ij})}{\lambda_i}\right]$ for that weapon is larger than that for any other weapon. If this is the case, it can be shown that the best mixed strategy in integers cannot have a lower Lagrangian than the value when one of the weapons has $N_{ij} = 1$ and the best non-integer weapon, as defined above is allowed to be optimally chosen. To prove this, consider the following.

In general, if the function $VS^N + \lambda N$ is minimized in non-integers, the optimal N can be found by differentiation. Thus

$$\frac{d}{dN}\left[VS^N + \lambda N\right] = VS^N \log(S) + \lambda \qquad \text{(A-9)}$$

Setting this equal to zero and solving for N results in

$$N^* = \frac{\log(\lambda) - \log(V) - \log\left[-\log(S)\right]}{\log(S)} \qquad \text{(A-10)}$$

This relationship substituted into $(VS^N + \lambda N)$ results in a value for the Lagrangian of

$$LG = \frac{-(\lambda)}{\log(S)} + \lambda\left[\frac{\log(\lambda) - \log(V) - \log\left[-\log(S)\right]}{\log(S)}\right] \qquad \text{(A-11)}$$

This function now can be applied to a general mixed strategy. For example, the value of the Lagrangian for a given set of N_{ij} is expressed as $\text{I.G.} = V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}} + \sum_{i=1}^{I} \lambda_i N_{ij}$ (A-12)

But if all the N_{ij} except for the weapon with max. $\left[\dfrac{-\log(S_{ij})}{\lambda_i} \right]$

are some prescribed value then the value of the N_{ij} for that "best" non-integer weapon can be found by use of equation (A-10). Calling the weapon which has the max. $\left[\dfrac{-\log(S_{ij})}{\lambda_i} \right]$ weapon m, the use of equation (A-10) with

$$V = V_j \prod_{\substack{i=1 \\ i \neq m}}^{I} S_{ij}^{N_{ij}} \qquad \text{(A-13)}$$

results in obtaining an optimum N^{*}_{mj} for whatever the other N_{ij} values are. If all these N_{ij} are substituted into equation (A-12), it yields

$$\text{L.G.} = \frac{-\lambda_m + \lambda_m \left[\log(\lambda_m) - \log\left[V_j \prod\limits_{\substack{i=1 \\ i \neq m}}^{I} S_{ij}^{N_{ij}} \right] - \log\left[-\log(S_{mj}) \right] \right]}{\log(S_{mj})}$$

$$+ \sum_{\substack{i=1 \\ i \neq m}}^{I} \lambda_i N_{ij}$$

or, equivalently

$$L.G. = \frac{-\lambda_m + \lambda_m\left[\log(\lambda_m) - \log(V_j) - \log\left[-\log(S_{mj})\right]\right]}{\log(S_{mj})}$$

$$- \sum_{\substack{i=1 \\ i \neq m}}^{I} \lambda_m N_{ij} \frac{\log(S_{ij})}{\log(S_{mj})} + \sum_{\substack{i=1 \\ i \neq m}}^{I} \lambda_i N_{ij} \qquad \text{(A-15)}$$

But, the first term in equation (A-15) is simply the value of the Lagrangian if weapon m is allowed to be non-integer and it is a pure strategy. Also, since

$\frac{-\log(S_{mj})}{\lambda_m} \geq \frac{-\log(S_{ij})}{\lambda_i}$ we can say that $\lambda_i \geq \lambda_m \frac{\log(S_{ij})}{\log(S_{mj})}$.

Thus, equation (A-15) will be at a minimum when $N_{ij} = 0$ for $i \neq m$ since the second two terms consist of

$$\sum_{\substack{i=1 \\ i \neq m}}^{I} N_{ij}\left[\lambda_i - \lambda_m \frac{\log(S_{ij})}{\log(S_{mj})}\right] \qquad \text{(A-16)}$$

which results in a larger Lagrangian for any non-zero N_{ij}.

It should be noted that if N^*_{mj} had turned out negative it is possible to show that the strategy which minimizes the Lagrangian has all N_{ij}, including N_{mj}, equal to zero.

The above shows that the minimum Lagrangian occurs when the best non-integer weapon exists in a pure strategy. The best mixed strategy must (by equations (A-15) and (A-16)) have a Lagrangian at least as large as $L.G. = L.G.^* + \left[\lambda_p - \lambda_m \frac{\log(S_{pj})}{\log(S_{mj})}\right]$ (A-17) where

$L.G.^*$ = value of best non-integer Lagrangian in a pure strategy

p = weapon which has a minimum value for

$$\lambda_i - \lambda_m \frac{\log(S_{ij})}{\log(S_{mj})}$$

$p \neq m$

This relationship can be effectively used to find a lower bound on the best mixed strategy Lagrangian and thus, by use of the Epsilon Theorem, a bound on the error involved in using pure, rather than mixed strategies.

This error bound has been computed for countless typical and realistic cases with the result that no error ever exceeded 1% and the majority were far less than this value. Thus, there is reasonable assurance that the use of pure strategies is not causing a large error. Of course, this error estimating procedure can be used on any given case.

b. Knapsack Method

The true minimum of equation (A-4) can be attained by use of a sequence of "knapsack" problems as follows. First, consider the sub-problem of minimizing $V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}}$ to the constraint $\sum_{i=1}^{I} \lambda_i N_{ij} \leq L_k$. This problem is equivalent to minimizing

$$\log\left[V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}}\right] \text{ which equals } \log V_j + \sum_{i=1}^{I} N_{ij} \log(S_{ij}) \quad (A-18)$$

under the same constraint. Thus, for a given L_k this sub-problem

$$\text{is equivalent to: minimize } \sum_{i=1}^{I} A_i N_i \text{ where } A_i = \log(S_{ij}) \quad (A-19)$$

under the constraint $\sum \lambda_i N_{ij} \leq L_k$. This is minimization of a linear objective function in integers and it is equivalent to the classical "knapsack" problem. Now, if a sequence of such problems are solved for various values of L_k, the minimum solution to equation (A-4) will be one of the knapsack solutions as long as the correct value of L_k is included in the set considered.

The required solution can be obtained in a reasonably efficient way if all information about equation (A-4) is utilized. For example, the minimum value of equation (A-4) for integral N_{ij} <u>for a given value of L_k</u> must be equal to or greater than the value for non-integral N_{ij} at the same L_k. Also, the minimum non-integral value of equation (A-4) can be shown to be the value attained when all $N_{ij} = 0$ for $i \neq m$ and $N_{mj} = \frac{L_k}{\lambda_m}$. Use of this information allows the easy determination of a span of L_k which must include the overall integral solution.

If L_k is allowed to take on any value, there is some value which results in a minimum to the Lagrangian (equation (A-4)) when non-integral N_{ij} are allowed. As was shown in the previous section, the N_{mj} at the optimum point is given by equation (A-10). Thus, the optimum non-integer L_k must be given by

$L^*_k = \lambda_m N^*_{mj}$ where N^*_{mj} is defined by equation (A-10), or

$$L^*_k = \lambda_m \frac{\log(\lambda_m) - \log(V_j) - \log\left[-\log(S_{mj})\right]}{\log(S_{mj})} \qquad \text{(A-20)}$$

where:

L^*_k = value of the constraint on $\sum_{i=1}^{I} \lambda_i N_{ij}$ which allows absolute minimization of the target Lagrangian for non-integer N_{ij}.

Now consider Figure (A-1). That figure represents the Lagrangian solution space as a function of L_k. Of special importance is that all integral solutions lie on or above the non-integral boundary. In addition, when weapon m takes on integral values for that given L_k, there is an integral solution on the boundary. Also, the lowest integral solution value for weapon m occurs when

$$L_k = \lfloor N^*_{mj} \rfloor \lambda_m \quad \text{or} \quad \lfloor N^*_{mj} + 1 \rfloor \lambda_m. \qquad \text{(A-21)}$$

That is, there is a minimum integral solution for weapon m at one of the integers on either side of N^*_{mj}. Calling this value of L_k the L_k' value there is another value for L_k, call it L''_k, such that L^*_k is between the two and the value of the Lagrangian at point L''_k on the minimum envelope is a constant, LG'.

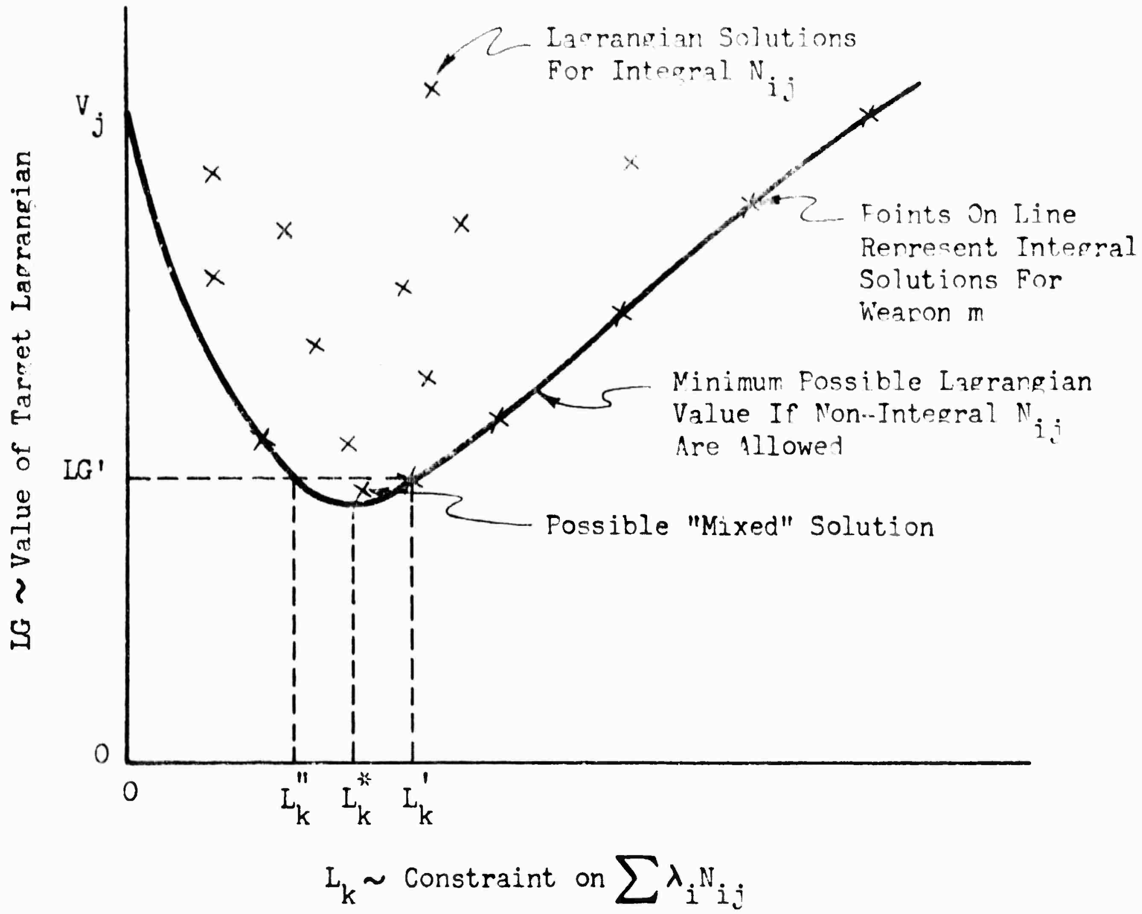


FIGURE A-1 LAGRANGIAN VALUE AS FUNCTION OF CONSTRAINT ON WEAPONS

The span of L_k' and L_k'' must include the best mixed strategy <u>if there is one better than the best pure strategy</u>. This is true because a strategy already is known, namely that represented by equation (A-21) which has a value of LG'. Since no solution, non-integral, or integral outside the span can produce a Lagrangian lower than LG', it is not necessary to look for the best mixed strategy outside the span.

The mechanical process is then as follows. Solve a knapsack problem with $L_k = L_k'$. That integral solution will have a

$\sum_{i=1}^{I} \lambda_i N_{ij} = L_k''' \leq L_k$. Set the new $L_k = L_k''' - \epsilon$, where ϵ is a small number designed to exclude the current set of N_{ij} from the solution but no others. Obtain a new solution, set the new L_k equal to the new $L_k''' - \epsilon$ and continue until $L_k''' \leq L''_k$. This process will obtain the best integral solutions in the span and they can be compared to obtain the minimum Lagrangian solution.

The bulk of the problem with this method is an efficient process for finding knapsack solutions. Gomory's linear programming algorithm (Ref. 2) and his knapsack methods (Ref. 3) were both tried with the method of Ref. 3 being the preferred method. However, the additional computation time was prohibitive.

2. Multiplier Convergence Techniques

Given that a method exists for minimizing the Lagrangian on each target the total weapons allocated to each can be summed to determine if, for the chosen λ_i, the correct total weapons were allocated. If the number of any type allocated does not match the possessed number, the whole set of λ_i must be changed. In general, reducing λ_i causes more weapons to be allocated of type i but since there is a dependency among the λ_i it might also cause less weapons of another type to be allocated. Thus, there is a real problem in deciding how to change the λ_i to converge on the desired number of weapons allocated.

To understand the multiplier convergence method used in this program, consider the following. First, the Epsilon Theorem states that if a strategy used on any target is not the one with the minimum Lagrangian the error involved is ≤ the difference between the minimum Lagrangian and the value of the Lagrangian for the strategy chosen. In other words, the Lagrangian value orders the strategies in a preference sense. The strategy with the minimum Lagrangian is most desirable, the strategy with next higher Lagrangian is next on the list and so on.

Thus, for a given set of λ_i a list of potential strategies for each target, in ranked order, can be obtained. It is only necessary to modify the strategy selection process so the best S strategies, as measured by the Lagrangian are obtained and remembered for each target. If S strategies are available for each target, it might be feasible to select from among the whole set one strategy for each target such that all weapon constraints are exactly met. If this is done, the error involved in using the non-minimum Lagrangian strategy on every target can be obtained by the Epsilon Theorem.

Given this set of S strategies for each target, how does one select the sub-set which minimizes the error caused by not using the preferred strategy? The most directly utilizable process is linear programming. To see this, consider the following.

Assume that strategy h for target j is described by the N_{hij} and an associated total value destroyed VD_{hj} if that strategy is used on target j. (For example, say that there are 3 weapon types and that the 4th strategy which potentially might be used on target 5 is $N_{415} = 0$, $N_{425} = 3$, $N_{435} = 1$ and that the value which will be destroyed is $V_{45} = 1.07$.) Then, the strategy selection process is to choose the strategies to use such that the maximum total value is destroyed on the target system and all weapons are used up. In linear programming language this is equivalent to choosing values for X_{hj} such that

$$\sum_{j=1}^{J} \sum_{h=1}^{S} X_{hj} \cdot VD_{hj} \quad \text{is maximized} \tag{A-22}$$

under the constraints

$$\sum_{j=1}^{J} \sum_{h=1}^{S} X_{hj} \cdot N_{hij} \leq W_i \quad 1 \leq i \leq I \tag{A-23}$$

$$\sum_{h=1}^{S} X_{hj} \leq 1 \quad 1 \leq j \leq J \tag{A-24}$$

Constraint (A-24) essentially says that only one of the S strategies for each target can be chosen. Constraint (A-23) says that the strategies chosen must be such that the number of weapons of each type totally chosen must not exceed the resources. The objective of course is to maximize total value destroyed under these constraints. It can be shown that maximizing total value destroyed is equivalent to minimizing the error resulting because non-minimum Lagrangian strategies were used.

Ideally, there should be another constraint which says that each X_{hj} must equal only 0. or 1.. This is required because integer strategies must be chosen. However, linear programs with integer constraints are notoriously difficult to solve. It was felt initially that the process might naturally lead to integer selection so this

L.P. formulation was used without the integer constraint. It turned out that the majority of the time integer solutions to X_h naturally occur.

Part of the reason is that the N_{hij} are integer and generally one of the S strategies for each target will definitely be the best one. Fractional X_h generally occur on about as many targets as there are weapon types so that the last few remaining weapons of a type can be employed even though there is not enough to completely fulfill the ideal strategy.

An additional saving feature is that rather than having J distinctly different targets the total target complex can often be grouped into J different classes of targets where there are T_j members in each class. In that case the L.P. formulation is the same as the above except equation (A-24) becomes

$$\sum_{h=1}^{S} X_{hj} \leq T_j \qquad 1 \leq j \leq J \tag{A-25}$$

Thus, the X_{hj} become numbers larger than 1 and the only fractional strategy is on no more than one of T_j targets. Thus, there generally won't be more than J fractional strategies out of a total of

$\sum_{j=1}^{J} T_j$ strategies.

Use of the L.P. produces a by-product vastly more useful than just the selection of strategies. Remember that the physical interpretation of a λ is that it represents the marginal utility of a resource (Ref. (1)). This can be qualitatively grasped by consideration of the Lagrangian equation. The Lagrangian is simply a balancing of two functions. The VS^N term represents the return from firing N weapons while the λN product represents the price of buying N weapons at a price of λ per weapon. In turn, the price λ must be the value of that weapon if it is used on some other target.

It turns out that a natural by-product of an L.P. process is a measure of the additional payoff if any individual constraint is relaxed by one unit. These multipliers, or prices as they are called, thus represent, in our case, the same thing as the lambdas involved in the Lagrangian.

Thus, at the completion of an L.P. run with any given set of strategies one of the results is a set of multipliers which form the best possible estimate for a new set of λ_i which can be used to generate a new set of strategies and so on until the λ_i have converged.

Recently Brooks & Geoffrion (Ref. 4) have also pointed out this same concept of using L.P. to find the Lagrange Multipliers. They have a simple proof that the method will lead to a convergence on optimal λ_i.

Our experience over the past three years has indicated several procedures which are required to make the process work for this case. First, as Ref. (5) indicates, convergence will occur only if the specific set of strategies chosen at one stage of the process are retained in the next set of strategies analyzed. Second, the set of strategies used in any given phase should include not only the strategies in the last phase but also more than one new strategy for each target class.

This use of more than one new strategy for each target at each phase is necessary to achieve reasonably rapid convergence to an optimum set of λ_i. Use of several new strategies at each phase allow enough new alternate choices to enter in to definitely improve the estimate for λ_i from one phase to the next.

Another valid question is the choice of a set of λ_i to start the process and the procedure to follow when the L.P. produces some $\lambda_i = 0$. (Which will occur whenever all weapons of a given type cannot be allocated.) Experience has shown that any starting λ_i will allow convergence as long as sufficient strategies are initially inserted into the L.P. process. However, the better the starting λ_i the more rapid the convergence. If $\lambda_i = 0$ do occur, experience has shown that a satisfactory process is to arbitrarily set the new $\lambda_i = .5 \times$ (previous λ_i). This causes weapon i to appear in more strategies and, ultimately to cause a non-zero λ_i to develop.

One technique that has been found to be useful in developing excellent starting λ_i is to use an upper bound theorem that has not previously been published. Since this theorem has other uses in its own right it will now be derived and its possible applications indicated.

The basic Lagrangian solution results in an X^* which minimizes the function $H(X) + \sum_{i=1}^{I} \lambda_i^* G_i(X)$ over all allowed values for X.

In other words
$$H(X^*) + \sum_{i=1}^{I} \lambda_i^* G_i(X^*) \leq H(X) + \sum_{i=1}^{I} \lambda_i^* G_i(X) \quad \text{(A-26)}$$

where X is allowed to take on any value in the space over which the function H(X) is to be minimized.

Of specific interest to our problem is the value of the Lagrangian when X_a denotes the special condition that

$$G_i(X_a) = 0 \text{ for } i \neq a \quad \text{(A-27)}$$

and

$$G_a(X_a) = C_a \quad \text{(A-28)}$$

where:

C_a = the level of resource type a such that $H(X_a) = H(X^*)$ when all other resources are at a zero level $(G_i(X_a) = 0)$.

In other words, the Lagrangian takes on the special value $H(X^*) + \lambda_a^* C_a$ when X_a describes a special condition on resources such that there is some level, C_a, of resource type "a" which can reduce the payoff H(X) to the same level as the mixed level of resources. In this circumstance for $X = X_a$ equation (A-26) becomes

$$H(X^*) + \sum_{i=1}^{I} \lambda_i^* G_i(X^*) \leq H(X^*) + \lambda_a^* C_a \quad \text{(A-29)}$$

or

$$\lambda_a^* \geq \frac{\sum_{i=1}^{I} \lambda_i^* G_i(X^*)}{C_a} \quad \text{(A-30)}$$

It is possible to find other similar solutions for the remaining variables, i.e.,

$$\lambda_j^* \geq \frac{\sum_{i=1}^{I} \lambda_i^* G_i(X^*)}{C_j} \qquad \text{(A-31)}$$

But the solution X^* is defined to be the one such that $G_i(X^*) = B_i$ where B_i is the level of resources of the original minimization problem. If equation (A-31) is multiplied by B_j and then summed for $1 \leq j \leq I$, the result becomes

$$\sum_{j=1}^{I} \lambda_j^* B_j \geq \sum_{j=1}^{I} \left[\sum_{i=1}^{I} \lambda_i^* B_i \right] \frac{B_j}{C_j} \qquad \text{(A-32)}$$

this becomes

$$\sum_{j=1}^{I} \frac{B_j}{C_j} \leq 1 \qquad \text{(A-33)}$$

which represents the upper bound Theorem.

To see how this is useful as an upper bound Theorem, consider what led up to equation (A-33). A mixed set of resources, denoted by B_j were to be allocated in such a way as to minimize some payoff, H(X). In addition, each of the resources were to be substitutable in the sense that there is some level of each resource, C_j, such that if all other resources levels were zero the same payoff as for the mixed set would result. Finally, equation (A-33) says that the B_j/C_j ratios relate in a special way.

Consider the minimization payoff H(X) as a function of the level of resources of one type alone. Generally this function is concave, that is

$$H[\beta Z_1 + (1-\beta) Z_2] \leq \beta H(Z_1) + (1-\beta) H(Z_2) \qquad \text{(A-34)}$$

where

$$0 \leq \beta \leq 1$$

and

Z_1, Z_2 are any two levels of resources of one type.

Thus, if for example there are three types of resources, typical functions might appear as in Figure (A-2).

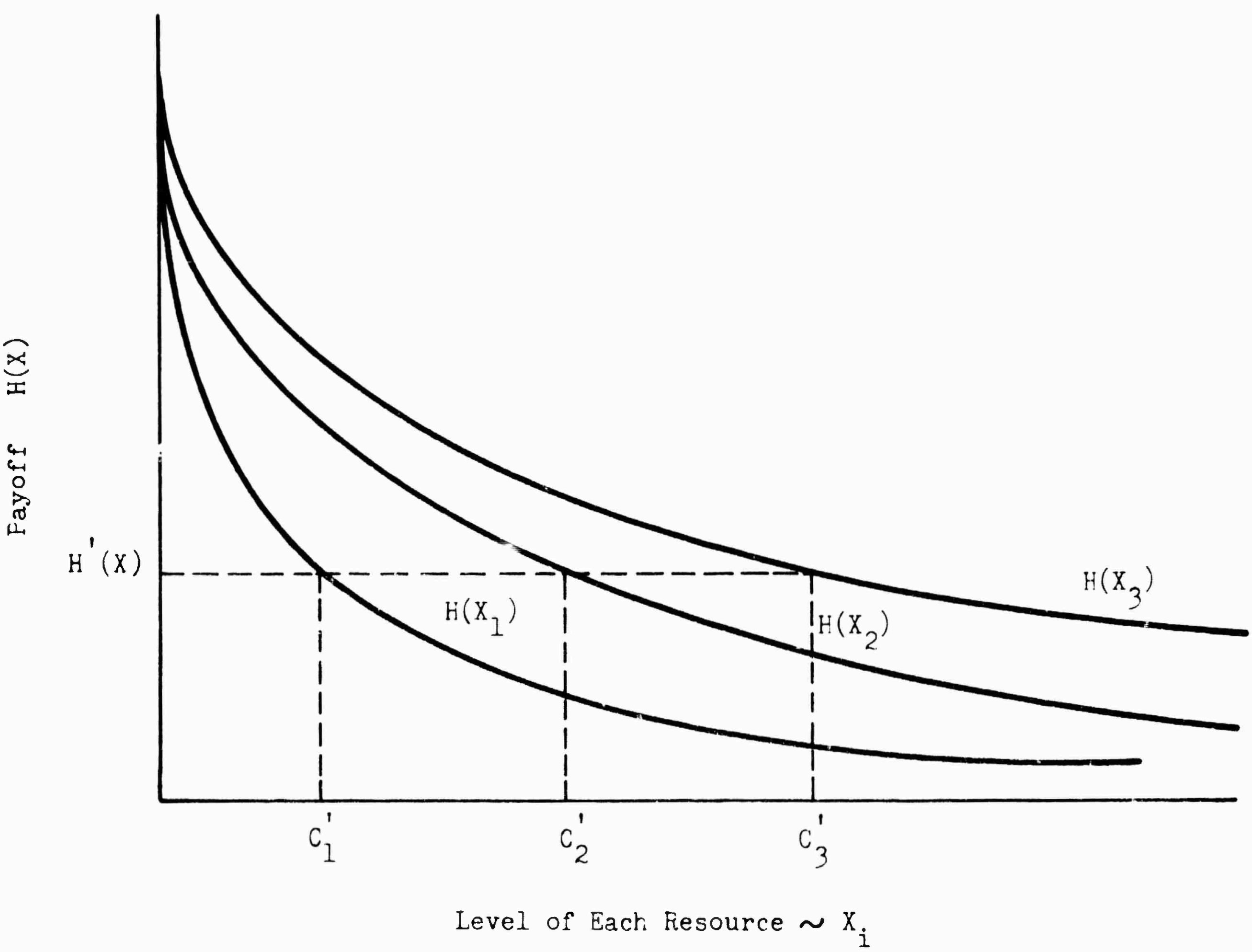


FIGURE A-2 PAYOFF AS A FUNCTION OF THE LEVEL OF INDIVIDUAL RESOURCES

Since these functions are concave, there is some payoff level $H'(X)$ which represents the upper possible payoff such that

$$\sum_{i=1}^{I} \frac{B_i}{C_i'} \leq 1 \qquad \text{(A-35)}$$

where:

C_i' = Level of resource type i to attain payoff $H'(X)$.

Thus, $H'(X)$ represents an upper bound to the payoff which the mixed resources $\{B_i\}$ can attain. Any higher $H(X)$ must achieve a $\sum_{i=1}^{I} \frac{B_i}{C_i'} > 1$ and thus not meet the conditions of equation (A-33).

The interesting thing is that $H'(X)$ has turned out to be very close to the actual $H(X^*)$ in the majority of weapon allocation problems to which this theorem has been applied. Of equal interest is that equation (A-33) can be used to obtain excellent starting λ_i.

The starting λ_i can be obtained simply by noting that λ_a represents the Δ payoff if all B_i $(i \neq a)$ are held fixed and the level of resource type a is increased by one unit. Using the upper bound theorem, it is possible to find the difference between two upper bounds, one at each level of resource type a and use this difference as an approximation to λ_a.

Assuming a small change in the upper bound, it can be shown that

$$\lambda_a \approx \frac{1/C_a}{\left[\frac{B_a - 1}{C_a^2 S_a} + \sum_{\substack{i=1 \\ i \neq a}}^{I} \frac{B_i}{C_i^2 S_i}\right]} \qquad \text{(A-36)}$$

where:

B_i, C_i = previous definitions

S_i = Minus the slope of the "resource type i only" payoff function (as described in Figure (A-2) at the location where $X_i = C_i$.

The upper bound Theorem has the potential of being useful in many resource allocation problems where it is convenient to determine how to allocate one type of resource and expedient to use an estimated payoff rather than allocating the true mixed resources.

B. DEVELOPMENT OF VARIOUS DAMAGE FUNCTIONS

The preceding discussion on weapon allocations has largely assumed that the damage to a target when attacked by a number of weapons of one type is described by the function

$$PK = 1 - (1-P)^N \quad \text{(B-1)}$$

or

$$PK = 1 - S^N \quad \text{(B-2)}$$

where:

P = probability of the target being killed by a single shot from a given weapon.

This function represents reasonably well the destruction of a point target but is deficient for area or defended targets. Thus, it is necessary to consider the meaning of other functional types, how they affect the allocation problem and to describe an efficient method for developing damage functions for all cases of interest.

In general, damage functions will take one of the three generalized forms represented in Figure B-1.

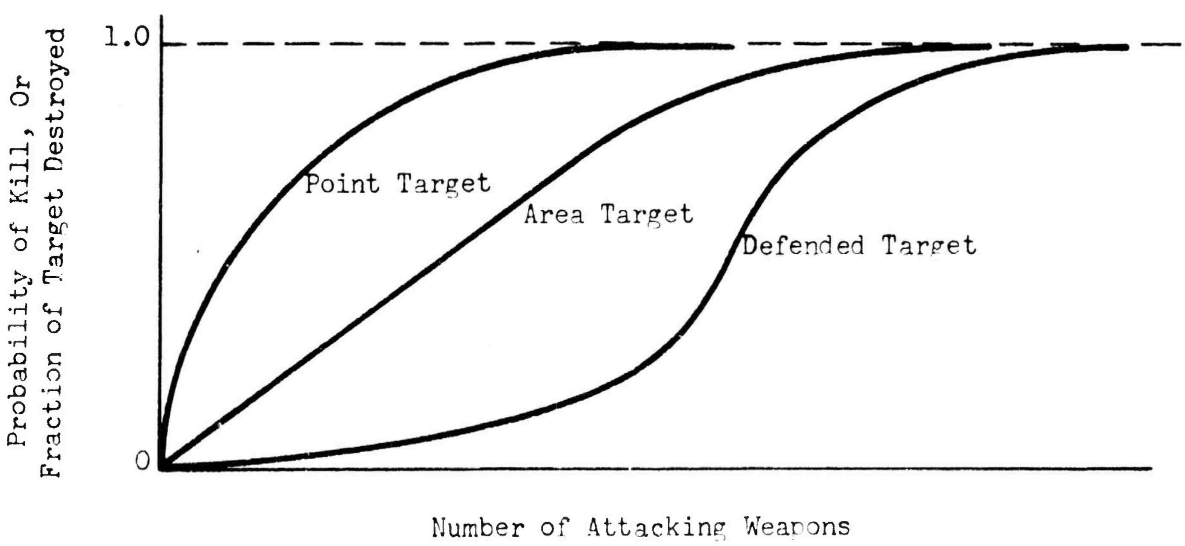


FIGURE B-1 TYPICAL DAMAGE FUNCTIONS

The point target is represented by a function of the equation (B-2) type. The area target has a linear portion which results from attacking an area in such a way that the nuclear effects of one weapon are independent of the effects from another weapon. Eventually no weapons can be impacted without overlapping the effect of another weapon and the linear damage function becomes concave downward. In the case of a defended target, there is generally some level of attack such that below that point only the "leakage" through the defense results in damage while above that point a saturation, or exhaustion occurs and damage accrues quite rapidly.

Given that such damage functions represent the general categories, it is important to realize how such functions react with the Lagrange multiplier method of weapon allocation. Everett points out (Ref. (1)) the most important consideration, namely that the method cannot be guaranteed in conjunction with non-convex functions such as are represented by the defended target function.

The multipliers essentially equate to the partial derivatives of the payoff function at a point and, as a consequence, it is impossible to determine unique λ_i^* in a region where a plane tangent to the payoff intersects the payoff function at some other point. The derivation of this necessary condition follows from the statement of the Lagrangian.

As has been stated previously, an optimal solution, X^* is one such that

$$H(X^*) + \sum_{i=1}^{I} \lambda_i^* G_i(X^*) \leq H(X) + \sum_{i=1}^{I} \lambda_i^* G_i(X) \qquad \text{(B-3)}$$

for any allowed value of X.

This is equivalent to

$$H(X) \geq H(X^*) + \sum_{i=1}^{I} \lambda_i^* \left[G_i(X^*) - G_i(X) \right] \qquad (B-4)$$

But, the right hand side of equation (B-4) is nothing more than the equation for an "I" dimensional plane of slopes λ_i^* passing through the point $H(X^*)$ when $X = X^*$. Thus, if equation (B-3) can be satisfied, there must be no X which results in a payoff less than that described by a plane tangent at X^*. In a region where the payoff H(X) is a non-convex function of $G_i(X)$, it is not possible to use the Lagrange method to find an X^* which leads to a $G_i(X^*) = B_i$ while minimizing H(X) since there is no λ_i^* which satisfies equation (B-3) in such a region.

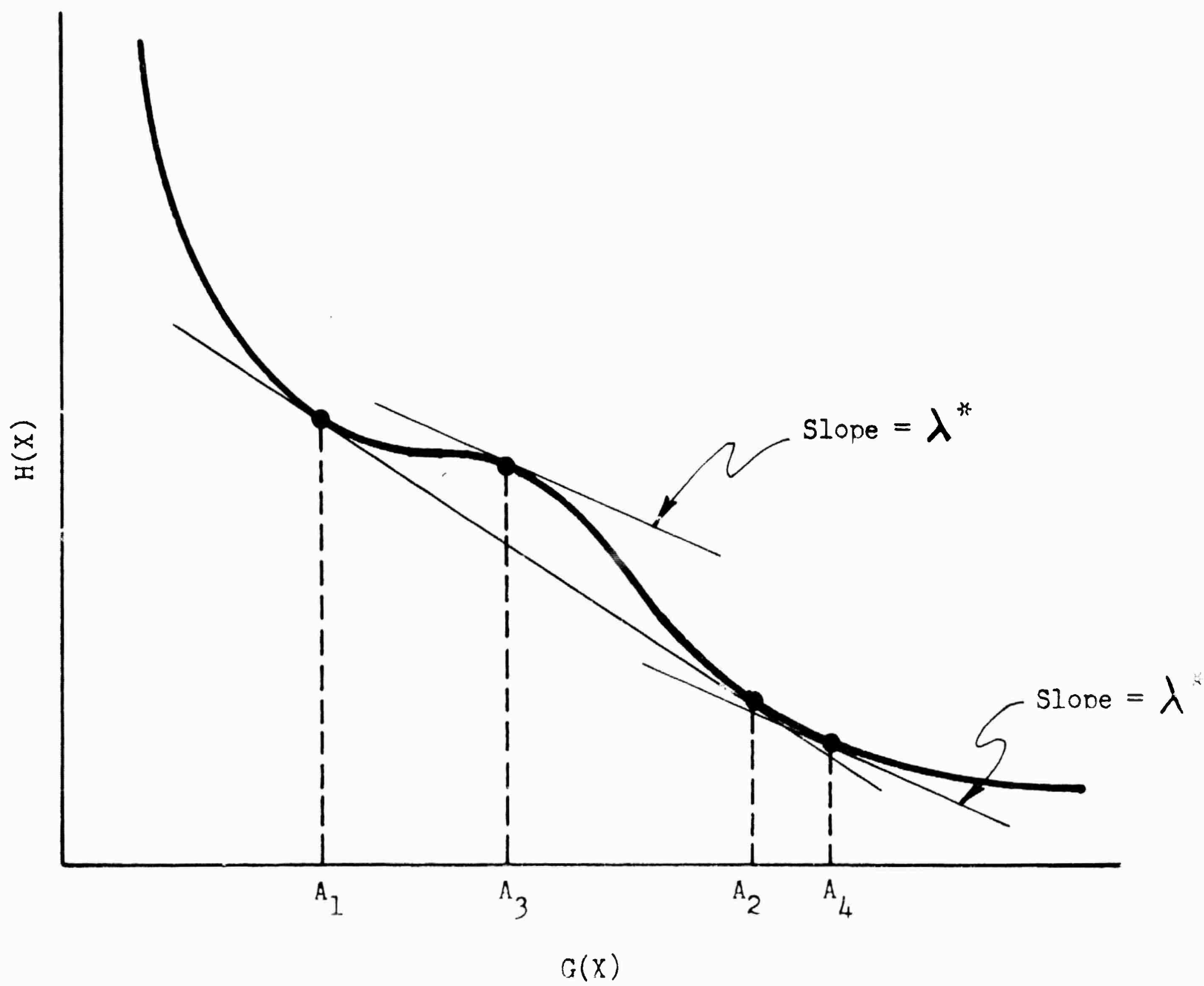


FIGURE B-2 A NON-CONVEX PAYOFF FUNCTION

For example, consider Figure (B-2) which demonstrates a payoff function, H(X), as dependent upon only one type of resource, G(X). In the region where $A_1 \leq G(X) \leq A_2$ the function is non-convex and a plane tangent to the function, e.g. at A_3, intersects the H(X) function at some other G(X). If λ were set equal to the slope, λ^*, and the multiplier method used, the solution obtained would be the one which satisfies equation (B-3) and it is at $G(X) = A_4$, since that tangent does not intersect the payoff at any other resource level.

This limitation concerning the defended target damage function is not as serious as might be thought. Consider the problem of allocating weapons to a target where the damage function is non-convex. As is represented in Figure B-3, there is some number of weapons, N^*, which <u>ideally</u> should be the minimum allocation <u>if the target is attacked at all</u>.

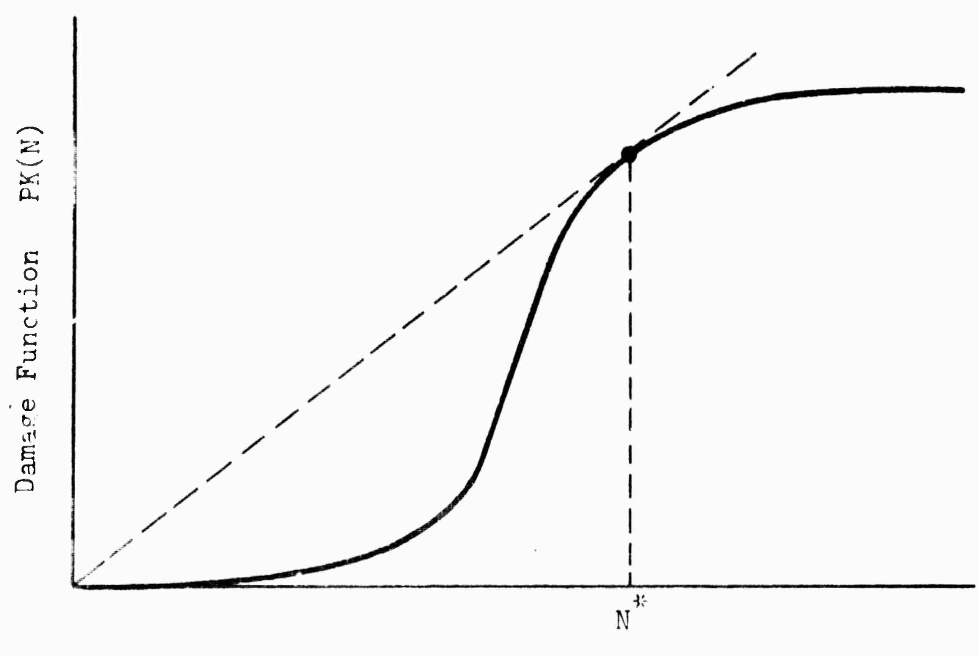


FIGURE B-3 NON-CONVEX DAMAGE FUNCTION

The reason is simply that the average payoff per weapon allocated is less than the maximum possible if $N < N^*$. (The N^* value is that one represented by the tangent point of a line from the origin to the damage function. This tangent line determines the allocation which results in the maximum attainable average payoff per weapon.)

If the attacker has any choice, he will attack at least to the N^* level, if he attacks at all. Thus, the precise shape of the function for $N \leq N^*$ is unimportant as long as N^* is still the point of tangency. More specifically, the same allocation would result for any payoff function $PK'(N)$ such that

$$PK'(N) \leq \frac{N}{N^*} \cdot PK(N^*) \text{ for } N \leq N^* \quad \text{(B-5)}$$

and

$$PK'(N) = PK(N) \text{ for } N \geq N^* \quad \text{(B-6)}$$

In other words, as long as the new function $PK'(N)$ has a payoff equal to or less than the convex approximation to PK(N).

Following this line of reasoning that says an optimized attack will avoid non-convex payoff regions as far as possible leads one to the idea of using only convex payoff function approximations. If any given payoff function is non-convex and a convex representation is used, errors will result only for those targets where an allocation of $N < N^*$ is chosen. In turn, this will occur only for the one target of a class where there is not enough weapons available to achieve the more preferred $N \geq N^*$ allocation.

Accordingly, the method used in this program is to determine the exact non-convex payoff function for a defended target and to then optimally allocate on the best convex approximation to that function. If all resultant allocations have $N \geq N^*$, no error results. In the rare instance where $N < N^*$, it is possible to perform hand computations to indicate the level of errors resulting.

Under the above reasoning, a generalized damage function has the shape represented in Figure B-4.

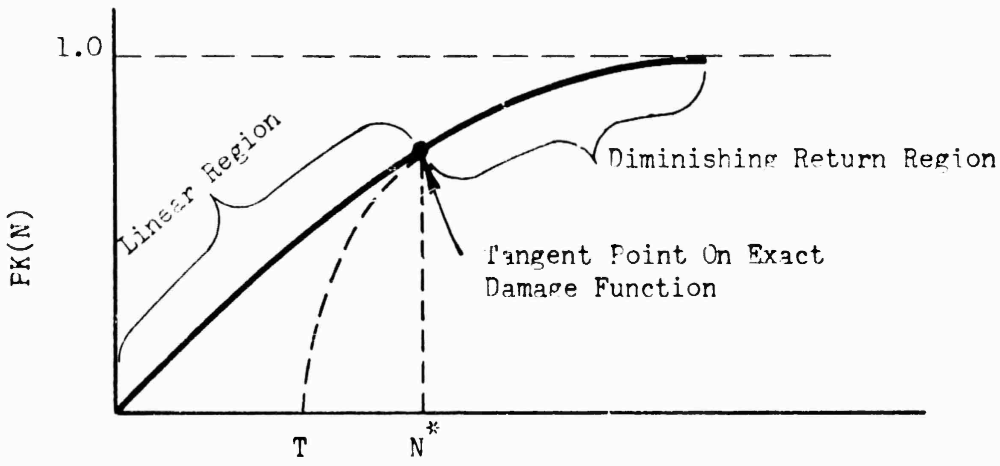


FIGURE B-4 GENERAL TARGET DAMAGE FUNCTION

Represented is a damage function composed of two regions. First is a region where each additional weapon results in a constant delta payoff. This is followed by a region where the return from one additional weapon is less than the last weapon.

Such a generalized function can be represented by a two parameter (T & P) family as follows:

$$PK(N) = \frac{N}{N^*} \cdot PK(N^*) \text{ for } N \leq N^* \qquad \text{(B-7)}$$

$$PK(N) = PK(N) \text{ for } N \geq N^* \qquad \text{(B-8)}$$

and

$$PK(N) = 1-(1-P)^{N-T} \tag{B-9}$$

where

N^* = Point of tangency of a line from the origin to the PK(N) function.

T = A Translation Parameter

P = A Non-Linear Fitting Parameter

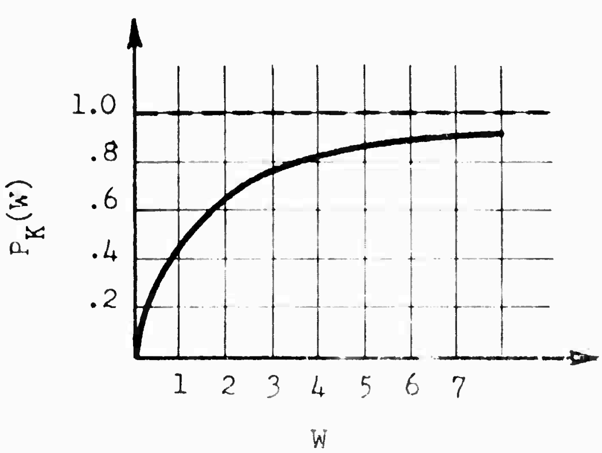
This representation has the special advantage that for point targets the translation parameter (T) becomes 0. and PK(N) results in the correct function. Also, for area targets, the linear portion can be fit exactly while the diminishing return region can usually be approximated quite well by a $1-(1-P)^N$ type of function.

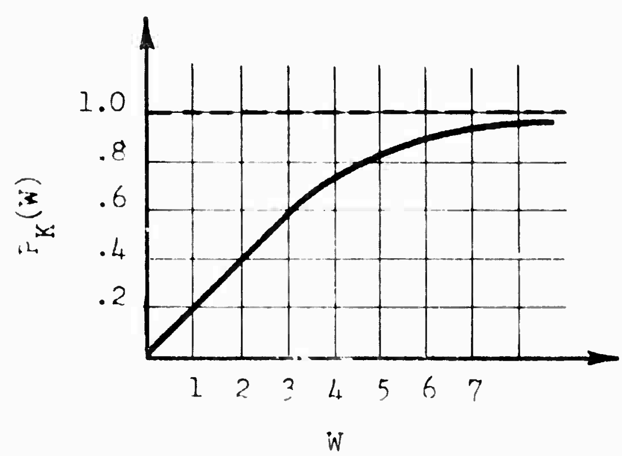
In this section, the methodology for obtaining the correct values of T & P is described. The methodology includes all types of targets, point, area and defended, for all possible combinations of conditions, such as hardness, area, weapon yield, CEP, defense level, etc. Involved in the methodology are basic program subroutines called COMPK and FIT.

1. Perfectly Reliable Weapons Vs. Undefended Target Damage Function

For perfectly reliable weapons against undefended targets, Rand Research Memorandum #RM-2743 ("Mobile System Survival Against a Ballistic Salvo - The Effects of Position Uncertainty and Other Parameters") gives the "expected coverage" of a circular target by W (= 1,2,3,...) optimally patterned weapons. A "cookie cutter" type of damage is used - that is, each weapon type is assumed to have an associated lethal radius (depending upon the weapon yield and the hardness of the target) which determines a circle centered at the point of impact. Total destruction of any portion of the target lying therein, is assumed. Expected coverage for a point target is the probability that at least one lethal circle will contain the point. For an area target, it is the expected fraction of the circular target area which will be included in at least one lethal circle. (The term "kill probability" -- notation, P_K -- is used here instead of expected coverage.)

Data taken from the Rand report tends to graph in one of the ff. fashions:





where, for a particular weapon and target (abbreviated as wpn and tgt hereafter), the precise shape of the curve is dependent only upon the lethal radius-to-CEP ratio and the lethal area-to-tgt area ratio. If there is a linear portion (as in the second sketch), its slope is equal to the lethal area-to-tgt area ratio.

Data like that in the first sketch is closely approximated by the mathematical form

$$P_K(W) = 1 - (1-p)^W, \qquad \text{(B-10)}$$

where

$$0 \leq p \leq 1.$$

The non-linear data in the second sketch follows

$$P_K(W) = 1 - (1-p)^{W-T}, \qquad (B-11)$$

where

$0 \leq p \leq 1$ and $T > 0$.

The linear data behaves as was described above. If a denotes the slope of the linear portion and W_T the point of tangency between the line and the curve, then all of the occurring cases may be described by the set of equations

$$P_K(W) = \begin{cases} aW \text{ for } W \leq W_T, \text{ and} \\ 1 - (1-p)^{W-T} \text{ for } W > W_T, \end{cases} \qquad (B-12)$$

where

$0 \leq p \leq 1$ and $T \geq 0$.

The approach used in the development of COMPK was to first define the kill probability functions for perfectly reliable wpns against undefended tgts, and then introduce reliability and defense effects. The former was resolved in accordance with the ff. procedure:

a) Graphs like the sketches shown above were produced from the Rand report data for a range of values of the lethal radius-to-CEP and lethal area-to-tgt area ratios.

b) For each curve, the plotted data was input to a computer program which produced approximating T and p values according to equations (B-12).

c) The fitted curves were evaluated and checked for agreement with the input data points.

d) It was then necessary to relate the T and p parameters to the two ratios mentioned above which influence the shape of the P_K curve. This was done through a series of fairly complicated curve fitting techniques, specifically tailored to insure that the functions produced would have correct properties and asymptotes. The resulting equations were evaluated over the initial range of values of the ratios, each time producing values for T and p which in turn, specified a unique curve of the form (B-12). This curve was then checked for consistency with the original Rand data.

e) Although wpn CEP's and tgt areas constitute a portion of the input data for AEM, the lethal radius of a wpn-tgt pair does not. A study* of the problem of computing lethal radii produced a pair of equations dependent upon wpn yield and tgt hardness, one equation applying to tgts whose hardness is below 48 psi, the other for harder tgts.

The process described in a) through e) above made possible a determination of the kill probability function for perfectly reliable wpns which are optimally patterned against undefended tgts. The

---

* Based upon data contained in DIA document #PC550/1-2-63 (1 September 1963) entitled "Physical Vulnerability Handbook--Nuclear Weapons."

significant variables are wpn CEP and yield, and tgt area and hardness. Complete details of the process are given in Appendix A.

2. Effect of Unreliable Weapons

The introduction of reliability into the P_K equations was done in an elementary manner. If the Parameter T in equations (B-12) is zero, the tgt is small in comparison to the lethal area of the attacking wpn -- effectively a point tgt. In this case, the interpretation of the parameter p is that it is the single-shot probability of kill, since

$$P_K(1) = 1 - (1-p)^1$$
$$= 1 - 1 + p$$
$$= p.$$

However, in the less-than-perfect reliability case, the true single-shot P_K is given by the product of the in-flight reliability (R) times the probability of terminal kill (p), since both of these events must occur for the wpn to perform effectively. Thus the indicated modification for an unreliable weapon is to define a new parameter ($p' = R.p$) which is then used in equations (B-12) in place of p as the single-shot kill probability.

For an area tgt ($T > 0$), the linear portion of the P_K curve reflects the attacker's ability to target his weapons far enough apart to insure no overlap in the corresponding lethal circles. If the wpn reliability is not perfect, however, there is a definite probability that some of these circles will be missing. Thus the correct modification to the linear portion in the unreliable case is

$$P_K(W) = R.a.W$$

--that is, define a new slope ($a' = R.a$) for the linear part.

In order to insure compatibility as $T \rightarrow 0$ between the point and area tgt methodologies, and in the absence of a clearly defined rationale, the parameter p is also modified in the unreliable wpn -- area tgt case in the same fashion as above (i.e. $p' = R.p$). A new translation (T') and point of tangency (W_T') may then be solved for directly, as a function of the transformed slope (a') and p':

$$T' = \begin{cases} \dfrac{1}{a'} + \dfrac{1 - \ln\left[\dfrac{a'}{-\ln(1-p')}\right]}{\ln(1-p')} & \text{for } a' < -\ln(1-p) \\ 0 & \text{otherwise} \end{cases}$$

$$\text{and} \quad W_T' = \begin{cases} T' + \dfrac{\ln\left[\dfrac{a'}{-\ln(1-p')}\right]}{\ln(1-p')} & \text{for } a' < -\ln(1-p') \\ 0 & \text{otherwise} \end{cases}$$

The resulting curve, although it is only approximately correct in the area tgt case beyond the tangent point, produces P_K values which may be checked according to the more detailed theory described below (by letting the defense effectiveness approach zero). This comparison has been performed several times with the result that the approximate values have not differed by more than 3% from the exact. (A sample is found in Appendix B.) The main reason for continuing to use the approximate method is that considerable saving of computer time is realized.

3. Effect of Reliability and Defense

The development of a capability for modelling a nation's defensive strength and accurately describing the impact of that defense on the outcome of a strategic war game is as multi-faceted a task as the actual defense procurement. Since the offense has the last move insofar as wpn allocation and attack timing is concerned, the defense must be as versatile as possible in order to be capable of dealing with the myriad forms which a sophisticated offensive threat may assume. Area vs. terminal vs. perimeter defenses, ICBM vs. bomber defenses, and civil defense are some of the identifiable components of a versatile national defense in the nuclear age.

Subroutine COMPK deals with terminal defenses only -- ABM, primarily, but with some AAA capability. It is assumed that an attack by a ballistic missile on an ABM-defended tgt may be characterized within the scope of the ff. descriptors:

a. Attacking Wpn
   1) Reliability
   2) CEP (n.mi.)
   3) Yield (MT)
   4) Number of decoys per whd

b. Target
   1) Area (n.mi.^{2})
   2) Hardness (psi)

c. Defense
   1) Number of AMM's
   2) Probability of Acquisition
   3) Probability of Discrimination
   4) Single-shot Probability of Interception

d. Tactical Options

(The tactical situations possible will be described in detail later.)

One of the first things done in the program is to reduce the input number of AMM's to an effective number reserved for whds:

Let N = input no. of AMM's

N_E = effective no. of AMM's

d = decoys/whd

p_D = prob. of discrimination

UFØ = undiscriminated objects/whd

It is assumed that

$$UFØ = 1 + (1-p_D)d$$

that is, that the probability of discrimination applies to decoys only. Then

$$N_E = \frac{N}{UFØ}$$

gives the expected number of AMM's available for whds.

For each of W attacking whds, the ff. outcomes are assumed possible:

a) unreliable

b) reliable, not acquired

c) reliable, acquired, not intercepted

d) reliable, acquired, intercepted

unless the defense has been exhausted or destroyed, in which case the possible outcomes reduce to

a) unreliable, or

b) reliable.

These events form the basis for the detailed methodology referred to previously, since it is possible to enumerate those combinations of the basic events which contribute to tgt destruction, compute their probabilities of occurrence, and compute the expected target damage resulting given their occurrence. In this way, an over-all expected damage, $F_K^*(W)$, is produced for a given value of W.

(The accuracy of this methodology is degraded by a yet unresolved problem - namely, to define the expected tgt damage when 5 whds, for example, are optimally patterned and 3 penetrate. At the present time, CØMPK assumes that the expected damage is the same as that which would obtain if the 3 penetrators were optimally patterned. This assumption results in a slight overestimate of the true expected damage in those cases where the optimal aim points do not all coincide at the center of the tgt.

However, there is in operation a separate IBM 1130 version of CØMPK which facilitates - with a minimum of reprogramming - the study of other, non-optimum damage functions. So far, no acceptable alternative to the damage computation used in CØMPK has been found.)

Currently, a choice of three tactical situations is possible via the input (integer) variable NDØC. If NDØC ≥ 0, the program regards the attack as occurring sequentially in time. It is assumed that the defense is eliminated by the first penetrating whd, the only factors preventing succeeding whds from accumulating further damage being their unreliability or the fact that total tgt damage has already been achieved. It is further assumed that the defense is unaware of the extent of the attack.

In addition, if NDØC > 0, the defense has decided to assign a constant number (namely, NDØC) of its N_E available AMM's to each visible whd until defense destruction or exhaustion occurs.

If NDØC = 0, it is assumed that the defense chooses its successive AMM assignments according to the Prim-Read doctrine described in Appendix C, until the N_E AMM's are depleted or the defense has been killed.

The tactical situation is quite different if NDØC < 0 is specified. In this case, the program assumes that the attack is in the form of a simultaneous salvo of W whds, where the defense does know the extent of the attack, and cannot be killed. It is further assumed that the defense allocates all of its N_E AMM's uniformly (insofar as is possible in integers) to counter the attack. This situation gives the defense all of the advantages, since it can't be killed and does the best job of allocating its AMM's.

If the real situation is that the defense's command and control system can be overloaded by achieving simultaneous arrival in the tgt zone - that is, if the defense can only handle a limited number of AMM intercepts simultaneously - then the input number of AMM's (N) should be reduced accordingly when NDØC < 0 is input.

4. <u>The Computational Procedures For Various Tactical Situations</u>

a. <u>Sequential Strike</u> ˙e

Let $P_n(I)$ = probability that the N^{th} penetration occurs on the I^{th} shot $(I = N, N+1, \ldots, W)$.

PN1(K,W) = probability of exactly K penetrators in W warheads $(K = 1, 2, \ldots, N-1)$.

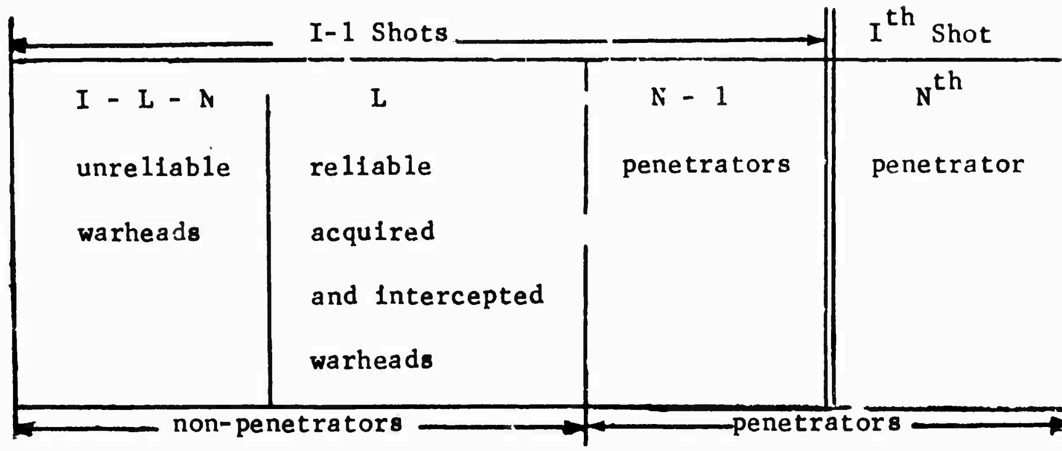
D(I,W) = expected damage given that W warheads are fired and the N^{th} penetration occurs on the I^{th} shot, i.e., expected damage accrued by penetrators N through (W-N).

D1(K,W) = expected damage accrued by penetrators 1 through (N-1).

The main calculation is then governed by the equation:

$$P_K^*(W) = \sum_{I=N}^{W} P_n(I) \cdot D(I,W) + \sum_{K=1}^{N-1} PN1(K,W) \cdot D1(K,W) \quad \text{(B-13)}$$

The $P_n(I)$ values are obtained by accumulating the probabilities of occurrence of those combinations of basic events which lead to the N^{th} penetration on the I^{th} shot. The D(I,W) and D1(K,W) values are computed from weapon reliability and equations (B-12), making use of the optimum damage assumption described earlier. In order that the N^{th} penetrating warhead kills the terminal defenses, we must have the following scheme:



That is to say, there will be some penetrators and some non-penetrators, but we must have N-1 penetrators in I-1 shots and the N^{th} penetrator on the I^{th} shot.

We define p_{N-1} to be the probability of N-1 penetrators in I-1 shots. This probability, of course, acts according to the binomial distribution. Therefore,

$$P_{N-1} = \binom{I-1}{N-1} p_{pen}^{N-1} (1 - p_{pen})^{I-N}, \text{ where } p_{pen} \text{ is}$$

the probability of penetration for a single warhead.

We now define L to be the number of reliable, acquired, and intercepted warheads in I-1 shots. Because the probability of several independent events all occurring is

$$p(A \cap A_2 \cap \ldots \cap A_n) = p(A_1)\; p(A_2) \ldots\ldots p(A_n), \text{ we have}$$

$$p_b = p(L) = \prod_{j=1}^{L} R \cdot p_A \left(1 - (1 - p_I)^{m_j}\right)$$

$$= R^L p_A^L \prod_{j=1}^{L} \left(1 - (1 - p_I)^{m_j}\right)$$

where:

P_A ≡ probability of acquisition

P_I ≡ probability of interception

m_j ≡ the interceptor assignment

The remaining I-L-N non-penetrators are unreliable. If we define,

$p_a = p(I-L-N)$, we have

$p_a = (1-R)^{I-L-N}$

Thus, combining all of these factors, we have for p_n, the probability of the N^{th} penetrator on the I^{th} shot, the following:

$$p_n(I) = \sum_{L=0}^{MIN} p_a \cdot p_b \cdot p_{N-1} \cdot p_{pen}$$

$$= \sum_{L=0}^{MIN} (1-R)^{I-L-N} \left[R^L \, P_A^{\,L} \prod_{j=1}^{L} \left(1 - (1 - P_I)^{m_j} \right) \right]$$

$$\left[\binom{I-1}{N-1} p_{pen}^{N-1} \; (1 - p_{pen})^{I-N} \right] \cdot p_{pen}$$

where:

$MIN = \min \{ I - N, NWUDX \}$

For those sequential strike cases with a constant assignment doctrine (NDØC $\geq$ 1)

$$NWUDX = \left[\frac{NE}{ND\emptyset C} \right]$$

In the Prim-Read case (NDØC = 0), define m_j = # of interceptors assigned to the j^{th} reliable and acquired warhead.

Here, NWUDX is the largest integer such that $\sum_{j=1}^{NWUDX} m_j \leq N_E$,

where the m_j's are determined as in Appendix C.

The second sum in equation (B-13) consists of the damage accrued by penetrators 1 through (N-1) and the probability of exactly one penetrator the probability of exactly two penetrators, ..., up to the probability of exactly (N-1) penetrators. The expected damage is described by (B-14b). The probability PN1 is calculated as follows:

$$PN1(K,W) = \sum_{L=0}^{W-K} \binom{W}{L+K} \binom{L+K}{L} (1-R)^{W-K-L} \cdot R^L P_A^{\,L} \prod_{j=1}^{L} \left(1 - (1-P_I)^{m_j}\right) \cdot R^K \left(1 - P_A + P_A (1-P_I)^{m_j}\right)$$

Assuming that penetrators accumulate damage according to equations (B-12), the calculation of expected damage from W warheads where the I^{th} is the N^{th} to penetrate, is governed by the equations:

$$D(I,W) = \sum_{L=0}^{W-I} \binom{W-I}{L} R^L (1-R)^{W-I-L} P_K(L+N) \qquad \text{(B-14a)}$$

and

$$D1(K,W) = P_K(K) \qquad \text{(B-14b)}$$

where L = random variable denoting the number of additional penetrators (after the N^{th})

R = reliability

and $P_K(L+N)$, $P_K(K)$ are determined according to (B-12).

Defense parameters do not appear in this expression since it is assumed that the I^{th} warhead eliminates the defense.

To avoid the indicated summation in (B-14a), two cases are distinguished. If $W_T \leq 1$ in equations (B-12), then (B-14a) may be shown to be exactly equivalent to

$$D(I,W) = 1 - (1-p)^{N-T}(1-Rp)^{W-I} \qquad \text{(B-15)}$$

where p and T are the parameters appearing in equations (B-12).

If W_T 1, equation (B-14a) is rewritten as

$$D(I,W) = \sum_{L=0}^{W-I} \binom{W-I}{L} R^L (1-R)^{W-I-L} P_K(L)$$

$$+ \sum_{L=0}^{W-I} \binom{W-I}{L} R^L (1-R)^{W-I-L} \left[P_K(L+N) - P_{K(L)}\right]$$

and approximated in this form.

The first summation is the expected damage due to W-I warheads, each with reliability R. This may be approximately computed via the short-cut method discussed in the "Reliability" section (above) and exemplified in Appendix B.

The second sum can be viewed as the average value of a function of a random variable (L), and approximated by the function of the average value (of L). The function here is

$$I(L) = P_K(L+N) - P_K(L)$$

and the average value of L is given by

$$E(L) = (W-I)R.$$

Thus, the overall approximation (for $W_T > 1$) is:

$$D(I,W) \approx P_K'(W-I) + f\left[E(L)\right]$$

where P_K' is the kill probability function resulting when the parameters of (B-12) are modified by R (in accordance with the short-cut logic.)

Therefore, for $W_T > 1$:

$$D(I,W) \approx P_K'(W-I) + P_K\left[N+(W-I)R\right] - P_K\left[(W-I)R\right]. \qquad \text{(b-16)}$$

(The IBM 1130 version of COMPK computes D(I,W) in the $W_T > 1$ case by the more precise equation (B-15). A comparison of the results is given below.)

b. Simultaneous Strike Case

As mentioned previously, the simultaneous strike case is characterized by an unkillable defense which allocates its interceptors uniformly (insofar as is possible in integers) to the visible portion of the attack.

Notation:

N_E = (integer) effective number of defenders

W = number of attacking warheads (=1,2,3,...)

J = random variable denoting the number of reliable warheads ($0 \leq J < W$)

K = random variable denoting the number of reliable and acquired warheads ($0 \leq K < J$)

m_K = minimum number of defenders assigned to any visible warhead

$$= \left[\frac{N_E}{K}\right]$$

m_K+1 = maximum number of defenders assigned to any visible warhead

n_K = number of reliable and acquired warheads to which exactly m_K interceptors have been assigned

N_K = number of reliable and acquired warheads to which exactly m_K+1 interceptors have been assigned

$$= N_E - K \cdot m_K$$

(Then $n_K = K - N_K$)

p_I = single-shot intercept probability

p = penetration probability for a reliable and acquired warhead to which m_K interceptors have been assigned

$$= (1-p_I)^{m_K}$$

P = penetration probability for a reliable and acquired warhead to which m_K+1 interceptors have been assigned

$$= (1-p_I)^{m_K+1}$$

n = random variable denoting the number of penetrating warheads of the n_K to which m_K interceptors were assigned ($0 \leq n \leq n_K$)

N = random variable denoting the number of penetrating warheads of the N_K to which m_K+1 interceptors were assigned $(0 \leq N \leq N_K)$

$J-K+n+N$ = total number of penetrating warheads

R = reliability of each attacking warhead

p_A = probability of acquisition

P_K = kill probability (expected damage) function defined in equations (B-12).

The detailed expression for the overall expected damage in the simultaneous strike case is then given by:

$$P_K^{*}(W) = \sum_{J=0}^{W} \binom{W}{J} R^J (1-R)^{W-J} \cdot \sum_{K=0}^{J} \binom{J}{K} p_A^{\,K} (1-p_A)^{J-K} \cdot \sum_{n=0}^{n_K} \binom{n_K}{n} p^n (1-p)^{n_K-n} \cdot \sum_{N=0}^{N_K} \binom{N_K}{N} P^N (1-P)^{N_K-N} \cdot P_K(J-K+n+N) \qquad \text{(B-18)}$$

where the values p, P, n_K and N_K of the two innermost sums are dependent on the index K, in accordance with their defining equations as given above.

Equation (B-18) is highly combinatorial in the sense that computation time increases astronomically with increasing W. However, it has been programmed for both the IBM 1130 and the CDC 6400 for reasonable values of W. (The 1130 version of CØMPK uses (B-18) with the restriction $W \leq 25$.)

By replacing $P_K(j-K+n+N)$ in (B-18) by J-K+n+N, the expression becomes the expected number of penetrators (PW) given W warheads fired, rather than expected damage. A tedious derivation then shows that the expression for PW reduces to:

$$PW(W) = WR(1-p_A) + \sum_{K=0}^{W} C(K) \binom{W}{K} (Rp_A)^K (1-Rp_A)^{W-K} \qquad \text{(B-19)}$$

where, for $p_I < 1$:

$$C(0) = 0$$

$$C(K) = \left[K(1+m_K p_I) - N_E p_I\right](1-p_I)^{m_K}$$

and, for $p_I = 1$:

$$C(K) = \begin{cases} 0 & (K \leq N_E) \\ K-N_E & (K > N_E) \end{cases}$$

Also, if $R = p_A = 1$, equation (B-19) further reduces to

$$PW(W) = C(W) \qquad \text{(B-20)}$$

where C is as given above.

The apparent interpretation of $C(K)$ is that it is the expected number of reliable and acquired penetrators given K reliable and acquired warheads.

Since $FW(W)$ is computationally feasible for large W, the approach taken in the AEM version of COMPK is to define an approximate, constant probability of penetration (PP) for each of the W attacking warheads, according to the equation

$$PP(W) = \frac{FW(W)}{W} \qquad \text{(B-21)}$$

and then compute overall expected damage using the approximate expression

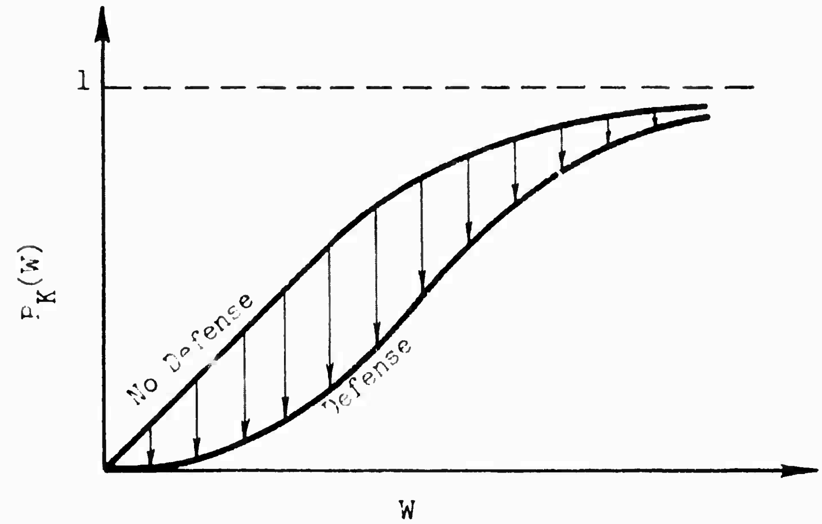
$$P_K^*(W) \approx \sum_{I=0}^{W} \binom{W}{I} [PP(W)]^I \, [1-PP(W)]^{W-I} \cdot P_K(I) \qquad \text{(B-22)}$$

Thus, the pertinent computations in the simultaneous strike case are given in equations (B-19), (B-21), and (B-22). A comparison is given below with equation (B-18) $P_K^*(W)$ values.

5. Obtaining T & P Parameters From Damage Functions

In all defense cases, after several $P_K^*(W)$ values have been generated and stored, a further computational step is necessary in order to present the results in usable form to the AEM main program.

In general, the effect of defense on the P_K function is like that shown in the sketch below.



That is, the offense's kill probability is lowered significantly for low values of W, but this effect diminishes for higher values due to the increasing chance of defense exhaustion and leakage. This tends to introduce a concave region into the P_K function which, without defense, is convex. However, as was discussed previously, the central wpn-to-tgt allocation problem is solved (in AEM) through the use of Lagrange multipliers, an optimization process which is inherently limited to convex functions. Thus, the final computational step is to provide a convex approximation to the stored $P_K^*(W)$ values. This is done in subroutine FIT, which produces new values for the parameters T and p of equations (3), which, in turn, specify a convex function approximating $P_K^*(W)$.

The final fit is a two-phase process. In the first phase, those $P_K^*(W)$ values which will be fit by the non-linear form

$$P_K(W) = 1 - (1-p)^{W-T} \qquad \text{(B-23)}$$

are identified. In the second, least squares estimates of the parameters of (B-23) are obtained for the selected $P_K^*(W)$ values.

If NW values of $P_K^*(W)$ have been computed and stored (i.e., for $W = 1, \ldots, NW$), the program first locates the largest value of W ($=W_{max}$) having the property.

$$\frac{P_K^*(W_{max})}{W_{max}} \geq \frac{P_K^*(W)}{W} \quad \text{for all } W = 1,\ldots,NW.$$

If $W_{max} = 1$, all NW points are fit in phase 2.

If $W_{max} > 1$, the program restricts attention to those $W < W_{max}$ and selects the largest of these ($= W'$) with the property:

$$P_K^*(W'+1) - P_K^*(W') \geq P_K^*(W+1) - P_K^*(W)$$

$$\text{for all } W < W_{max}.$$

In this case, the phase 2 points are chosen to be the $[W, P_K^*(W)]$ pairs for $W = W', \ldots, NW$.

To derive the phase 2 least squares estimates, define

$$y = 1 - q^{x-T} \tag{B-24}$$

as the function of interest. (Equation (B-23) is of this form.)

Given n observations, (x_i, y_i), the sum of squares function to be minimized is

$$S = \sum_i \left[y_i - (1 - q^{x_i - T})\right]^2.$$

For simplicity, the subscripts will not be carried in the derivation, so that

$$S = \sum [y-(1-q^{x-T})]^2.$$

$$\frac{dS}{dq} = 2\sum (y-1+q^{x-T})(x-T)q^{x-T-1}$$

$$= 0.$$

$$\frac{2}{q}\sum (x-T)q^{x-T}(y-1+q^{x-T}) = 0$$

$$\sum xq^{x-T}(y-1+q^{x-T}) = T\sum q^{x-T}(y-1+q^{x-T})$$

$$\frac{dS}{dT} = 2\sum (y-1+q^{x-T})q^{x-T}(-\ln q)$$

$$= 0.$$

$$\therefore \sum q^{x-T}(y-1+q^{x-T}) = 0$$

and $\sum xq^{x-T}(y-1+q^{x-T}) = 0.$

Define

$$y_i' = 1-y_i.$$

Thus

$$\sum q^{x-T}(q^{x-T}-y') = 0$$

and $\sum xq^{x-T}(q^{x-T}-y') = 0.$

$$\sum q^x(q^x - q^T y') = 0$$

and $\sum xq^x(q^x - q^T y') = 0.$

$$\sum q^{2x} = q^T \sum q^x y' \qquad \text{(B-25)}$$

and $\sum xq^{2x} = q^T \sum xq^x y'$

$$\frac{\sum (q^x)^2}{\sum q^x y'} = \frac{\sum x(q^x)^2}{\sum xq^x y'}$$

$$\frac{\sum (q^x)^2}{\sum q^x y'} - \frac{\sum x(q^x)^2}{\sum xq^x y'} = 0.$$

At this point, the Newton-Raphson iterative scheme (Appendix A, eqn. 35) is used to solve

$$F(q) = 0$$

where

$$F(q) = \frac{A}{B} - \frac{C}{D} \qquad \text{(B-26)}$$

$$A = \sum (q^x)^2$$

$$B = \sum q^x y'$$

$$C = \sum x(q^x)^2$$

$$D = \sum xq^x y'.$$

The derivative of F with respect to q is

$$F'(q) = \frac{BA' - AB'}{B^2} - \frac{DC' - CD'}{D^2}$$

where

$$A' = \sum 2q^x x q^{x-1}$$

$$= \frac{2}{q}\sum x(q^x)^2$$

$$= \frac{2C}{q}$$

$$B' = \sum x q^{x-1} y'$$

$$= \frac{1}{q}\sum x q^x y'$$

$$= \frac{D}{q}$$

$$C' = \sum 2x q^x x q^{x-1}$$

$$= \frac{2}{q}\underbrace{\sum x^2 (q^x)^2}_{U}$$

$$= \frac{2U}{q}$$

$$D' = \sum x x q^{x-1} y'$$

$$= \frac{1}{q}\underbrace{\sum x^2 q^x y'}_{V}$$

$$= \frac{V}{q}$$

$$F'(q) = \frac{B \cdot \frac{2C}{q} - A \cdot \frac{D}{q}}{B^2} - \frac{D \cdot \frac{2U}{q} - C \cdot \frac{V}{q}}{D^2}, \text{ or}$$

$$F'(q) = \frac{1}{q}\left[\frac{2BC - AD}{B^2} - \frac{2DU - CV}{D^2}\right] \quad \text{(B-27)}$$

where A,B,C,D,U and V are as defined above. The iteration equation is then

$$q_{K+1} = q_K - \frac{F(q_K)}{F'(q_K)} \quad \text{(B-28)}$$

using (B-26) and (B-27), and, upon convergence, equation (B-25) yields the expression for T:

$$\ln\left[\sum q^{2x}\right] = T(\ln q) + \ln\left[\sum q^x y'\right]$$

$$\ln A = T(\ln q) + \ln B$$

$$T = \frac{\ln A - \ln B}{\ln q} \quad \text{(B-29)}$$

The final results of this process are estimates for the parameters <u>T</u> and <u>p</u> (= 1 - q) of equation (B-23), which then uniquely determine the equation (B-12) kill probability function used in AEM.

<u>Examples</u>

Figures B-5 and B-6 give examples of the defended target computations. Figure B-5 depicts the $F_K^*(W)$ values produced for various tactical situations (identified by NDØC values). The solid curves

FIGURE B-5 EXAMPLES OF DEFENDED TARGET COMPUTATIONS

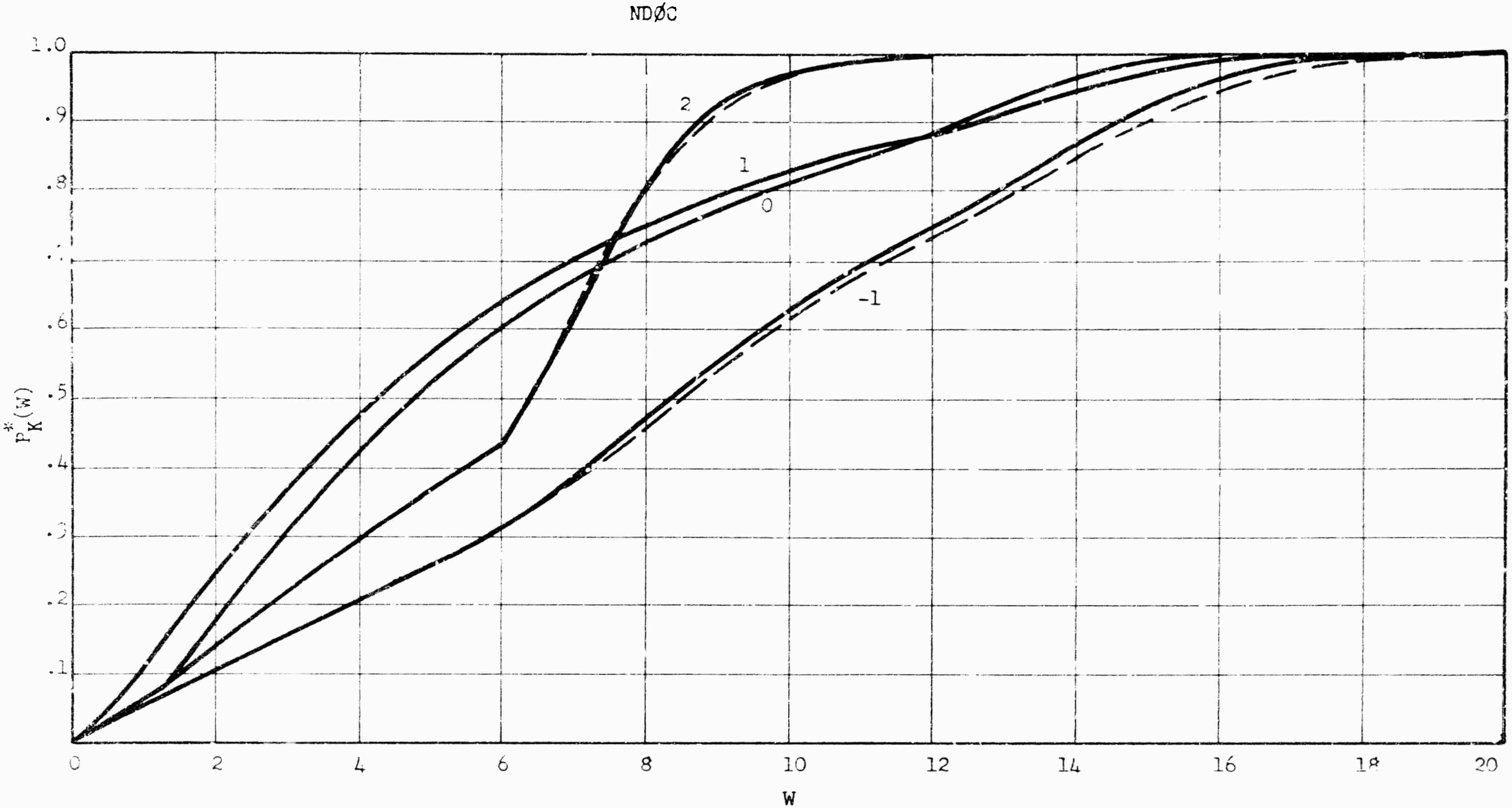
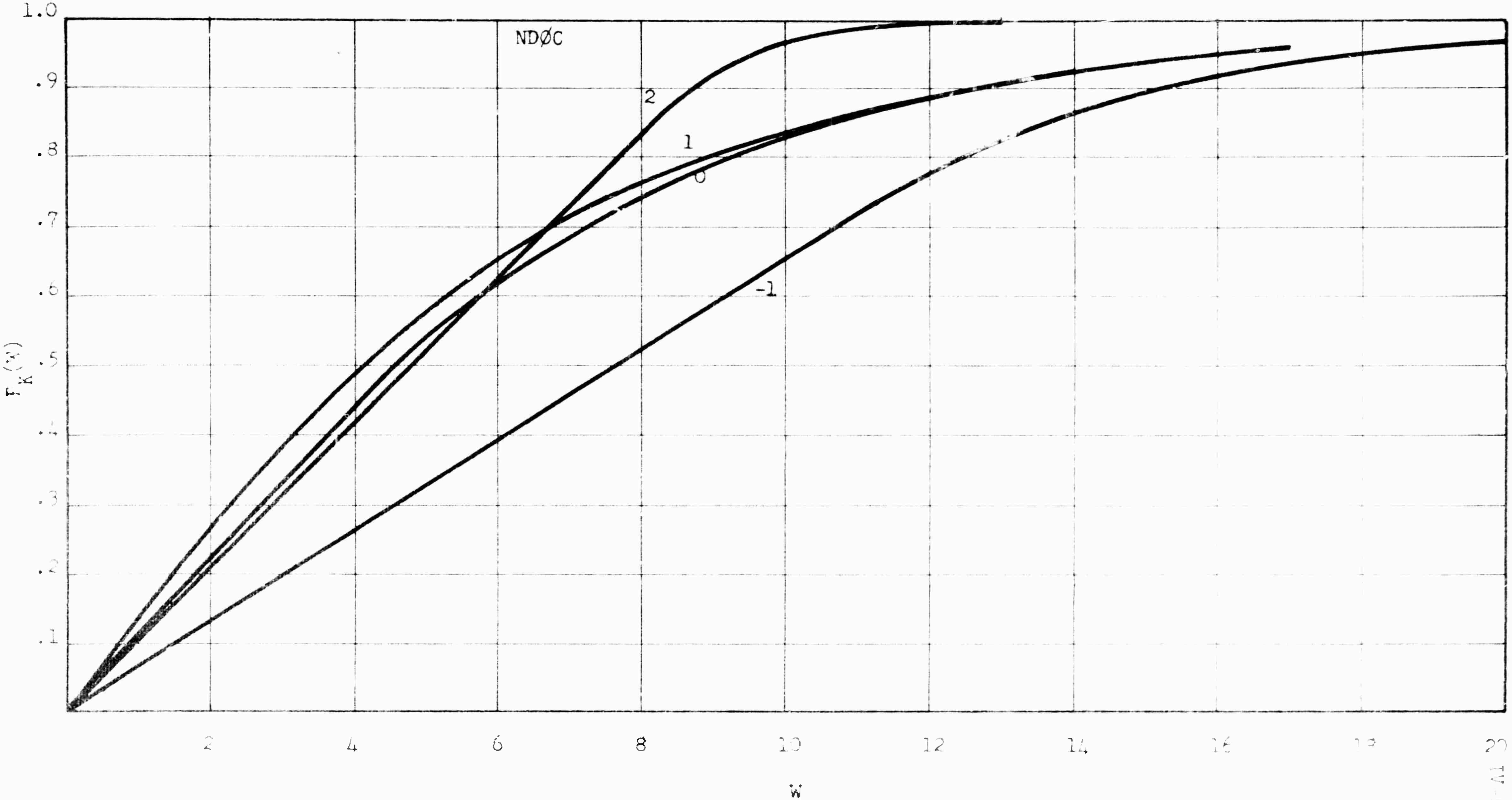


FIGURE B-6 EXAMPLES OF THE FINAL FITTING PROCESS (DEFENDED TARGET)



present the IBM 1130 results, while the dotted curves are those obtained by the approximate methods of CØMPK. (In most cases, the two coincide to within plotting accuracy and only the solid curves are shown.) Figure B-6 gives the convex approximations resulting from the COMPK $P_K^*(W)$ values.

The input data used were:

Reliability = .9

CEP = .5 n.mi.

Yield = 1 MT

No. of Decoys = 2

Hardness = 7 psi

Area = 50 n.mi.^{2}

Probability of Acquisition = .9

Probability of Discrimination = .875

Probability of Interception = .9

Number of AMM's = 15

NDØC = variable

In all cases, the following intermediate computations resulted:

Lethal Radius = 3.15189 n.mi.

LR/CEP = 6.30378

$(L/D)^2$ = .62419

T = .42907

p = .81974

W_T = 1.01841

a = .62420

(T, p, W_T, a: Initial values for the equation (B-12) parameters (perfect reliability, no defense assumed.))

N_E = 12.00000

Specific case results:

a. NDØC = -1:

W' = 8

NW = 20

T = 5.68832

p = .21002

W_T = 11.15739

a = .06494

} Final values for the equation (B-12) parameters (resulting P_K function plotted in Figure 3-6.

b. NDØC = 0:

NWUDX = 11

Prim-Read Assignment (Modified Method, Appendix C):

$$n_1 = 2, n_2 = \ldots = n_{11} = 1, n_{12} = \ldots = 0$$

W' = 2

NW = 17

T = 1.14227

p = .17982

W_T = 4.19536

a = .10822

c. NDØC = 1:

NWUDX = 12

W' = 1

NW = 18

T = .52651

p = .17380

W_T = 2.71191

a = .12579

d. NDØC = 2:

$NWUDX = 6$

$W' = 7$

$NW = 13$

$T = 5.76818$

$p = .54026$

$W_T = 8.36053$

$a = .10366$

6. Bomber Kill Probabilities

A straightforward adaptation of all aspects of the above methodology is used to determine bomber kill probabilities in AEM, with number of SAM's input as the number of defenders. In this case, the assumption of optimum damage for penetrators is more realistic than in the missile case, since a certain amount of in-flight retargeting might be possible.

However, there is a basic drawback.

The basic unit penetrating (or not penetrating) is considered to be a bomb (or standoff weapon) rather than a bomber. Contrary to reality, it is this unit which the program regards as being reliable (or not), etc. (This same difficulty is encountered in MIRV-ed missile cases.) The effect of this faulty conceptualization on the $P_K^*(W)$ values is unknown, but believed small.

C. FORCE TARGET VALUES

Before describing a method for determining force target values, it would be valuable to review the requirement for such items. As was discussed in Section I, the requirement is to develop a program which will maximize the objective function of one opponent engaging another opponent in some given scenario. This term "objective function" can obviously take many forms but in this program it is visualized as a family of preference contours which describe one opponent's relative preference for reducing his own damage (damage limitation) when compared to achieving more damage on his opponent (assured destruction).

These preference contours cannot be built into the model but, instead can be represented in the form of an equivalent objective of maximizing:

$$VD_2 - K \cdot VD_1 \qquad (C-1)$$

where:

VD_2 = industrial value destroyed on the side hitting second

VD_1 = industrial value destroyed on the side hitting first

K = side 1 relative preference for damage limitation compared to assured destruction.

This objective function can be used in conjunction with any set of preference contours in a manner now to be described.

The key to use of preference contours is the ability to derive the maximum A.D. envelope shown on Figure (2). The maximum A.D. envelope indicates the maximum attainable assured destruction for an attainable level of damage limitation. The location of this function in the A.D. vs. D.L.

plane is determined by such items as the resources of each side and the management of those resources.

If a method exists to maximize the above objective function for any given situation, the result must be that point on the maximum A.D. envelope with slope K. This is proven very easily as follows:

Consider Figure C-1, which is a visualization of the VD_2 vs. VD_1 function.

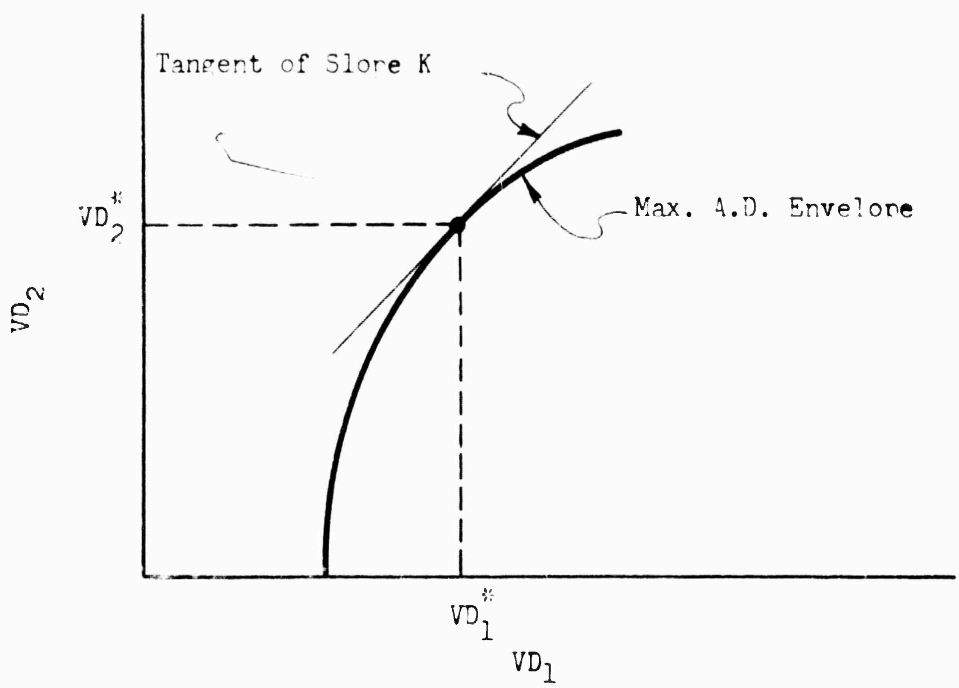


FIGURE C-1 ASSURED DESTRUCTION (VD_2) VS. DAMAGE LIMITATION (VD_1) FUNCTION

Now, at the achievement of a solution where $VD_2 - K \cdot VD_1$ has been maximized the solution VD_1^*, VD_2^* must lie on the envelope. This must be true because for any value of VD_1 any companion VD_2 which does not lie on the envelope could not be the one such that $VD_2 - K \cdot VD_1$ is maximized.

Additionally, the VD_1^*, VD_2^* pair must be at the unique point on the envelope with slope K. This is true because the maximum of $VD_2 - K \cdot VD_1$ means

$$VD_2^* - K \cdot VD_1^* \geq VD_2 - K \cdot VD_1 \qquad (C-2)$$

or

$$VD_2 \leq VD_2^* + K \left[VD_1 - VD_1^* \right] \qquad (C-3)$$

As in the general Lagrangian, the right side of equation (C-3) describes a linear function of slope K passing through the point VD_1^*, VD_2^* when $VD_1 = VD_1^*$. Since VD_2 is $\leq$ this function for any other value of VD_1, the VD_1^*, VD_2^* solution must lie on the max. A.D. envelope at the point where the tangent has a slope of K. (It is important to note that the above must only be true if the max. A.D. envelope is convex so that the unique tangent point does exist).

Since maximizing $VD_2 - K \cdot VD_1$, results in one point on the max A.D. envelope it is possible to obtain multiple solutions for various K values so the complete envelope can be obtained in a numerical approximation form. Thus, given that a program can be developed to maximize the above simple function that same program is useful for any preference contours of interest. The development of a generally useful program resolves down to the problem of obtaining one which will max. $[VD_2 - K \cdot VD_1]$.

Of the many schemes which might be used to maximize the delta damage, the one chosen for this model is the one that exploits the availability of a linear value scale oriented weapon allocation procedure. In Part A of this chapter such a method was described. That method can be used to optimally allocate weapons to a set of targets where each target is represented by a constant value, V_j returned if that target is destroyed.

To see how such an allocation procedure might be used to maximize delta damage consider again the scenario of Figure 1. As RED is setting up his strike he knows what values will be destroyed for each BLUE industrial target he destroys. These values are an input to the program and might be based on population, MVA, industrial floor space or any other value system. However, RED does not possess any similar values for BLUE weapons. Instead, he knows that he wants to maximize the delta damage.

Also, RED realizes that if he did place values on BLUE weapons the net effect is a control over the allocation procedure. That is, high values on BLUE weapons compared to the values on BLUE industry results in high counterforce attacks which result in reduced damage to RED industry when BLUE retaliates. When the BLUE weapon values are low, the opposite effect occurs.

Following the above line of reasoning, it appears reasonable to suspect that there might be some special set of values V_j^* to place on BLUE weapons such that the allocation procedure is indirectly led to an attack allocation which maximizes delta damage. If such a set of values exists, it would be necessary to develop an efficient procedure for finding them.

In summary, the nature of the procedure utilized in this program for the above is as follows:

STEP 1: Choose an arbitrary set of values for RED to place on BLUE weapons.

STEP 2: Optimally allocate RED weapons to all BLUE targets.

STEP 3: Compute expected BLUE surviving weapons.

STEP 4: Optimally allocate expected BLUE survivors to all RED targets.

STEP 5: Make a new set of BLUE weapon values equal to K times the weapon multiplier values resulting from STEP 4 and return to STEP 2. Stop when the value scales converge.

It can be shown that this procedure is guaranteed to find at least a local optimum in the max. delta damage function but a global maximum cannot be guaranteed. However, considerable experience with the procedure has led to the development of sub-steps in the procedure which minimize the possibility of overlooking global maxima.

To see that local optima are guaranteed, consider the procedure at STEP 5 just after BLUE surviving weapons have been allocated. STEP 5 chooses a value for BLUE weapon target j to be $V_j = K \cdot \lambda_i$ where i identifies the BLUE weapon type based at target j. Then, when the procedure returns to STEP 2, RED's objective in the weapon allocation procedure is to maximize the sum of value destroyed on force targets plus the value destroyed on industrial targets, or

$$\text{MAX } H = \sum_{j=1}^{F_B} V_j \left[1 - \prod_{i=1}^{I_R} S_{ij}^{N_{ij}} \right] + VD_B \tag{C-4}$$

where:

VD_B = BLUE industrial value destroyed

F_B = number of BLUE weapon targets

V_j = value of each weapon target

I_R = number of different types of RED weapons

However, this is equivalent to the following:

$$\text{MAX } H = \sum_{k=1}^{I_B} V_k \left[N_k - NS_k \right] + VD_B \tag{C-5}$$

where:

N_k = total number of BLUE weapons of type k

NS_k = number of survivors of BLUE type k weapons

I_B = total number of types of BLUE weapons

This equivalence is obtained simply by grouping all F_B BLUE weapon targets into I_B distinct types and summing up the total survivors as follows

$$NS_k = \sum_{j=1}^{F_B} X \cdot \prod_{i=1}^{I_R} S_{ij}^{N_{ij}} \qquad \text{(C-6)}$$

where:

X = 1 if j identifies a target of type K

X = 0 otherwise

But, STEP 5 chose as a weapon value for BLUE weapons the amount

$$V_k = K \cdot \lambda_k' \qquad \text{(C-7)}$$

where:

λ_k' = weapon λ of the last time BLUE allocated his survivors.

Substitution of (C-7) into (C-5) yields the equivalent RED objective of

$$\max \left[\sum_{k=1}^{I_B} K \lambda_k' \left[N_k - NS_k \right] + VD_B \right] \qquad \text{(C-8)}$$

which is the same as

$$\max \left[VD_B - K \cdot \sum_{k=1}^{I_B} \lambda_k' NS_k + V_F \right] \qquad \text{(C-9)}$$

where:

$$V_F = K \cdot \sum_{k=1}^{I_B} \lambda_k' N_k$$

At STEP 2 this objective function would be maximized and some NS_k^* would result. It will now be shown that the value destroyed on RED by the NS_k^* must be such that the delta damage from the last phase to this phase must not have diminished. In other words, that

$$VD_B^* - K \cdot VD_R^* \geq VD_B' - K \cdot VD_R' \qquad \text{(C-10)}$$

First, note that the solution VD_B^* and NS_K^* maximize equation (C-9), or

$$VD_B^* - K \cdot \sum_{k=1}^{I_B} \lambda_k' NS_k^* + V_F \geq VD_B - K \cdot \sum_{k=1}^{I_B} \lambda_k' NS_k + V_F \qquad \text{(C-11)}$$

for any other combination of VD_B and NS_k attainable. More specifically for the VD_B and NS_k combination resulting from the last set of force values. Thus

$$VD_B^* - K \cdot \sum_{k=1}^{I_B} \lambda_k' NS_k^* \geq VD_B' - K \cdot \sum_{k=1}^{I_B} \lambda_k' NS_k' \qquad \text{(C-12)}$$

where:

* denotes the new solution resulting from the $K \cdot \lambda_k'$ values

' denotes the last solution

However, at the completion of STEP 4 using the previous set of values, there was some accomplished value destroyed by the then existent number of

BLUE survivors. Since those survivors were allocated by the Lagrangian method, it can be said that

$$V_R - VD_R' + \sum_{k=1}^{I_B} \lambda_k' NS_k' \leq V_R - VD_R + \sum_{k=1}^{I_B} \lambda_k' NS_k \qquad \text{(C-13)}$$

where:

V_R = total RED industrial value

NS_k' = number of BLUE weapons surviving RED's attack on last iteration

VD_R' RED value destroyed by BLUE's survivors

λ_k' = optimal set of λ_k

This equates to

$$VD_R' - \sum_{k=1}^{I_B} \lambda_i' NS_k' \leq VD_R - \sum_{k=1}^{I_B} \lambda_i' NS_k \qquad \text{(C-14)}$$

It can be said that equation (C-14) would hold even when VD_R and NS_k take on the values of NS_k^* and VD_R^* which are going to result from the next iteration. This means that

$$\sum_{k=1}^{I_B} \lambda_k' NS_k' \leq VD_R' - VD_R^* + \sum_{k=1}^{I_B} \lambda_k' NS_k^* \qquad \text{(C-15)}$$

Substitution of (C-15) into (C-12) is possible without changing the inequality since the $\sum_{k=1}^{I_B} \lambda_k' NS_k'$ on the right side of equation (C-12) is preceded by a negative and a number $\geq$ than $\sum_{k=1}^{I_B} \lambda_k' NS_k'$ is replacing that term. Doing so yields

$$VD_B^* - K \cdot \sum_{k=1}^{I_B} \lambda_k' NS_k^* \geq VD_B' - K \left[VD_R' - VD_R^* + \sum_{k=1}^{I_B} \lambda_k' NS_k^* \right] \qquad (C-16)$$

This reduces to the desired form which says that the new solution is $\geq$ the previous solution, or

$$VD_B^* - K \cdot VD_R^* \geq VD_B' - K \cdot VD_R' \qquad (C-17)$$

The above proof very simply says that use of the λ_k from BLUE's allocation as a basis for weapon target values in RED's allocation is guaranteed to not result in a degradation to the overall objective function. However, it does not say that another solution which achieves a higher payoff doesn't exist.

As in many other mathematical optimization problems, there is always the thought that the convergence to a global optimum can be partially guaranteed by solving a local optimization [illegible] several times with different starting points. If all of the[illegible]s converge to the same solution, there is some confidence that a global optimum was found.

In this case such a procedure has been found to work very well. In fact, only one solution is generally found if the starting value scales for use in STEP 1 are appropriately chosen. Experience indicated that if a non-global optimum is found the most common error is that BLUE force targets were not attacked heavily enough. It has turned out that if the starting BLUE weapon values are chosen to be very high compared to BLUE industrial values (say a 10 to 1 ratio) the process will converge to a near global maximum in one series of steps through the process.

a. <u>Defense Effect on Force Values</u>

Before leaving this topic of weapon value scales, it would be appropriate to discuss one special case which is important. In certain instances, e.g. when attacking a bomber fleet, attack on a target not only reduces the surviving force by some amount but also reduces slightly the effectiveness of the other survivors. In such a case the appropriate value is not exactly $K \cdot \lambda_i'$.

For example, if a bomber fleet has to penetrate an area defense, the average probability of penetration of one member of the fleet might well depend upon the <u>total</u> number of bombers in the fleet. That is a fleet of W' bombers might have a probability of penetration of P' while $W = W' - X$ bombers would have a probability of penetration $P < P'$. In such a case destruction of one bomber does not reduce the capability of the surviving force by $K \cdot \lambda_i'$ but it does reduce it by

$$V_i \approx K \left[\lambda_i' + MU\emptyset \right] \quad \text{(C-18)}$$

Where:

V_i = approximate value saved if one less weapon of type i survives

λ_i' = value saved because one weapon of type i does not attack a target

$MU\emptyset$ = marginal value saved because one less weapon interacts with the area defense and all the other weapons which do survive have a probability of penetration lower than what they would have if the weapon was not destroyed

MUO = marginal impact of having one less object encountered by the defense

Of interest here is the MUO component of equation (C-1[illegible]). It is the factor that allows for specific consideration of the dilution effect present in random area defenses.

In Section G, random missile and area defenses are presented and the basic equations discussed. These equations relate the bomber (or missile) probability of penetration to the total number of bombers (missiles) in the attack. Thus, the dilution factor is directly computable.

Derived in Section G is the method for computing the marginal impact of having one more defender. This marginal impact is shown to be equal to

$$MUI \approx \sum_{j=1}^{T} \frac{\partial VD_j}{\partial P_p} \left[\frac{\partial P_p}{\partial D} \right] \qquad \text{(C-19)}$$

Where:

MUI = marginal value of having one more interceptor

j = target subscript

T = total number of targets

VD_j = value destroyed on target j under the existing allocation

P_p = appropriate weapon probability of penetration

D = total number of defenders

By use of this relationship and equations (G-8), or (G-11), and (G-9), MUI can be computed.

Using the same equations, it is possible to demonstrate that the marginal value to the defense of having one less object attacking, MUØ, equates to MUI as follows:

$$MUØ = MUI \left[\frac{D}{S_N} \right] \qquad (C-20)$$

Where:

S_N = total number of reliable, acquired and undiscriminated objects presented to the defense

Thus, it is possible to use the random defense marginal utility, MUI, in a given strike (like the second strike in Figure 2) to provide a basis for the MUØ term in C-18. This then leads to a value to place on the force target in strike one such that the attack which maximizes the delta will result.

The two components of equation (C-18) allow for the fact that destruction of a weapon causes degradation of the total force effectiveness due to a dependency on the total number of survivors and for the fact that not only has the force been degraded but there is also one less member in the force.

In general, the only instances where this form of a value is used is in the random defense environment. If it is used, however, the convergence proofs also still hold.

b. Secondary Delta Damage Criteria

In some special cases it is necessary to have a capability for using a secondary criteria for those situations when maximizing delta damage leads to allocations which are only marginally better than other potential allocations. In such cases the use of a secondary criteria is desirable for use to help choose from among alternate strategies generated in the allocation process.

This event usually occurs when arsenals are used which have very high assured destruction capabilities. In such cases, the delta damage computed may fluctuate due to small numerical differences in the strategy sets selected and the resulting force values computed. For these cases, the optimal delta damage normally sought by AEM loses some of its meaning since the expected delta damage is scarcely controlled by the initiator.

In such a circumstance it would be desirable to improve the consistency of case results as opposed to improving the delta damage optimality. This consistency can be obtained by means of a secondary criteria which is queried if the delta damage is within a specified range of the maximum found. The best answer is assumed to be that answer which has the best solution in terms of the secondary criteria that is within a specific range of the maximum delta damage found. Note that the secondary criteria is not an allocation objective and is therefore not maximized by the allocation process. Rather, the results which incidentally produces the best result in the secondary criteria is preferred as long as the delta damage penalty is within a specified tolerance.

There are several candidate criteria to be considered. For instance, an attack plan which achieves the highest OMT target damage may be preferred if the delta damage is not drastically affected. There are many criteria of this type which could be formulated which state a preference between attack plans which result in nearly equivalent damage. However, these preferences could be directly controlled by use of hedges which would include these preferences during the plan generation. Since this development is a post-attack evaluation, to choose between competing attack plans, we are more concerned with a reasonable criteria for a more general case.

There are several causes of delta damage instability. Non-linearities in survivability effects, retargeting effects, and random defense effects frequently prevent the best delta damage from being found. In addition, if both arsenals can accomplish very high damage levels independent of the level of any counter force attack, the resulting delta damage is reasonably insensitive to the allocation. In an extreme case, there is no real benefit in even performing a counter force strike if the opponent can still achieve virtually 100 percent damage in his retaliation. In all these cases it seems prudent to prefer a heavier counter force attack. This is true if one assumes the retaliation may have an increased uncertainty in achieving the best expected damage if weapon survivability is relatively low. Note that AEM consideres only prompt nuclear effects and assumes command and control as being reasonably independent by weapon type. If all operational factors could be considered, it is likely that a counter force attack is in reality more destructive than is currently modeled in AEM.

In addition, delta damage may be the same at more than one level of civilian damage, e.g., if delta damage is 10, the damage levels could be 50-40 or 100-90. In this case the lower damage levels seem more desirable since the retaliation is more sensitive to the attack plan and the implied uncertainty of the retaliator capability to accomplish the best plan is increased.

It was felt that for competing attack plans which result in reasonably close expected delta damages, preference should be given to the plan having the greater counter force attack. The question then becomes one of defining a greater counter force attack. Several options are open: the secondary criteria could be the lowest non-force damage by the initiator, which implies the highest force attack; or the fewest number of retaliating warheads; or the fewest surviving bases after the first strike, etc. Any of these could have been selected. However, most inconsistencies have appeared in the attack on ICBM silos. Since the secondary criteria is not maximized but is used to improve consistencies over a range of force options, this was the selected secondary criteria model, i.e., the attack plan incidentally generated which results in an expected delta damage no less than a specified amount less than the best one generated which results in the fewest surviving ICBM's is the preferred plan.

The amount of delta damage which is allowed to be lost to achieve this secondary criteria is specified by input as follows:

REPTOL = the maximum penalty in delta damage to be allowed while considering the secondary criteria of fewest surviving ICBM weapons.

Let us consider a case which has the following expected results:

Iteration	Delta Damage	Surviving ICBMs
1	-106	120
2	20	375
3 (Best Delta)	32	300
4	27	290
5	31	314

If REPTOL is less than 5, iteration number 3 would be the preferred attack plan since it has the best delta damage (and fewer surviving ICBM than iteration five). If REPTOL is greater than five, iteration number 4 would be the preferred attack plan since there are fewer surviving ICBM's and the delta damage is within REPTOL of the best delta damage.

This criteria thus forms a stateable bias toward the heavier counter force attacks. Whether minimizing the surviving ICBM's within a specified range of the best delta damage provides sufficient consistency will only be determined by analytic utility of AEM.

D. COUNTERVALUE DAMAGE CONSTRAINTS

Even though this program can be used to generate optimum strategies for any set of preference contours (by use of the K factor in the delta damage function) there is a strong requirement to be able to find an optimum solution in one run on the computer without making multiple runs for various values of K. The best current solution to this need is the one described in Section I. Namely, the use of K = 1 plus a maximum allowed industrial value destroyed on your opponent.

As diagrammed in Figure (6), this objective has a solution which occurs in one of two places on the max. A.D. envelope. If the location on the envelope with slope = 1. occurs at an assured destruction level less than the max. allowed level the desired solution is at the unity slope location. However, if the unity slope location is above the max. desired A.D. the desired solution is at that location where the A.D. = max. allowed A.D.

It has been possible to modify the basic weapon allocation and value scale determination process described previously so that the desired one of the two possible solution locations is determined in one computer run. This is achieved by appropriate use of the flexibility of the linear program used to optimally allocate weapons.

Normally, the objective of the program is to maximize delta damage ($VD_B - K \cdot VD_R$) within constraints on weapon and target quantities. The above desired limitation on countervalue destruction can be viewed as another constraint like

$$VD_B \leq VD_{B\ max.} \qquad \text{(D-1)}$$

where:

$VD_{B\ max.}$ = max. desired destruction on BLUE industrial value.

This constraint will not interfere with the solution if the unity slope condition is for a $VD_B \leq VD_{B\ max.}$ and it will locate that point with the minimum possible VD_R at $VD_B = VD_{B\ max.}$ if the second location for the unity slope occurs. Very often this second condition is the one that holds and the program then finds those strategies which achieve a desired A.D. in such a way that RED has a minimum loss. In the context of the current concepts about A.D. and D.L. this is a very compatible statement about the possible objectives of a country.

As simple as this constraint is, it does interfere with the optimum λ_i iteration process in the L.P. if the solution to be obtained is at $VD_B = VD_{B\ max.}$ To realize how this interference occurs, let us review the λ iteration process.

Beginning with some set of λ_i, a set of preferred strategies are found for each target. The L.P. then chooses from among those strategies the set which maximizes total value destroyed. As a by-product, the L.P. also produces a new set of λ_i which should allow determination of an even better set of strategies.

At any step in this iteration the currently most preferred strategy N^*_{ij} for target j is the one with

$$\text{min. } H = V_j \prod_{i=1}^{I} S_{ij}^{N^*_{ij}} + \sum_{i=1}^{I} \lambda_i N^*_{ij} \qquad \text{(D-2)}$$

This specific strategy will improve the answer (Ref. (4) and (5)) only if the following condition holds

$$V_j - H_{min.} \geq \lambda_j \qquad \text{(D-3)}$$

where:

V_j = value of target j

$H_{min.}$ = minimum solution to equation (D-2)

λ_j = target λ obtained from the last L.P.

The new item in equation (D-3) is the target λ. Remember that the weapon λ's obtained from the L.P. represent the amount of value destroyed which would be lost if the I^{th} weapon constraint was tightened by one unit. Similarly, there are target λ's which represent the value destroyed which would be lost if the target j constraint was tightened by one unit. Within these definitions of λ_i and λ_j it is easy to understand equation (D-3).

First, consider equation (D-2). The first term indicates the value surviving on target j if attacked by the strategy $\left\{N_{ij}^*\right\}$. But, those weapons in the attack must be removed from some other target and value destroyed must decrease on those targets. The λ_i essentially represent this amount of value destroyed given up for each weapon of type i diverted from some other target. (The very definition of λ_i dictates that this be true.) Thus, equation (D-2), when minimized, represents the net value surviving in the total target complex if target j was to be attacked by strategy $\left\{N_{ij}^*\right\}$.

However, if target j is to be attacked by this new strategy $\left\{N_{ij}^*\right\}$ the weapons attacking that target in the last L.P. must be diverted elsewhere. In doing so there will be a net loss in value destroyed, which is represented by the appropriate λ_j.

This new strategy for target j will be an improvement over the previous solution only if the new way to attack target j results in a net increase in value destroyed. That is, if

$$V_j - V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}^*} \geq \sum_{i=1}^{I} \lambda_i N_{ij}^* + \lambda_j \qquad (D\text{-}4)$$

Or, if the value destroyed on target j ($V_j - V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}}$) exceeds the loss $\sum_{i=1}^{I} \lambda_i N_{ij}^*$ because weapons were shifted from other targets to this one plus the loss (λ_j) because weapons currently attacking this target had to go elsewhere.

With this visualization of the L.P. iteration process, it is possible to uncover the problems associated with the damage constraint ($VD_B \leq VD_{B\ max.}$)

If the above process is being applied to a countervalue target, the question about a given strategy $\{N_{ij}\}$ being an improvement (as described by equation (D-3)) must be expanded. If the damage constraint is not being met, then any improved strategy can enter into the solution. However, if the damage constraint is exactly met, the new strategy must do more than meet the condition of equation (D-3) because no increased delta damage is allowed on the value targets.

The general L.P. version of equation (D-3) has a form (Ref. (4)) as follows:

$$P(X) \geq \sum_{i=1}^{m} \xi_i' \cdot g_i\ (X) \qquad (D\text{-}5)$$

where:

$P(X)$ = payoff from strategy X

$g_i(X)$ = level of constraint i called upon by strategy X

ξ_i' = constraint i multiplier from the last L.P.

In the case just described P(X) is $V_j - V_j \prod S_{ij}^{N_{ij}}$ and the ξ_i' are the λ_i when $1 \leq i \leq I$ and λ_j when $i = j$. The $g_i(X) = N_{ij}$ when $1 \leq i \leq I$ and $g_i(X) = 1$ when $i = j$ since the strategy applies to only one target.

In the case when there is a damage constraint, there is one additional component of equation (D-5), namely when $i = k$ where k identifies the damage constraint. When $i = k$, the $g_i(X)$ must equal P(X) since that is the amount of the damage constraint used up by strategy X.

Now, what is λ_i' when $i = k$? The definition of λ_k' is the standard one of "the value destroyed reduction if constraint k is tightened by one unit." Thus λ_k' will equal zero if in the last L.P. the damage constraint was not met. In such a case, equation (D-5) reduces to equation (D-3). If in the last L.P. the damage constraint was met, λ_k' will have a finite, positive value and equation (D-3) takes on the new form of

$$V_j - V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}} - \sum_{i=1}^{I} \lambda_i N_{ij} \geq \lambda_j + (V_j - V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}}) \cdot \lambda_k \quad \text{(D-6)}$$

where:

λ_k = damage constraint lambda

(There is an obvious physical interpretation of equations (D-5) and (D-6) like the one for equations (D-3) and (D-4).)

Manipulation of the terms in equation (D-6) results in

$$V_j (1 - \lambda_k) - V_j (1 - \lambda_k) \prod_{i=1}^{I} S_{ij}^{N_{ij}} - \sum_{i=1}^{I} \lambda_i N_{ij} \geq \lambda_j \quad \text{(D-7)}$$

Except for the $(1 - \lambda_k)$ multiplier of V_j in this equation it is the same, effectively, as equation (D-3). This indicates that when strategies are being developed for countervalue targets the optimum strategy is one that minimizes the Lagrangian, H, for that target with an effective value

$$V_j^* = V_j (1 - \lambda_k).$$

This result identifies the effect of adding a damage constraint to the L.P. iteration process. The impact is a very slight modification of the basic process to the extent that the Lagrangian for the countervalue targets should be minimized for an effective value $V_j^* = V_j (1 - \lambda_k)$ rather than for a value of V_j. Given this modification, the process is guaranteed to converge to optimal allocations.

Of equal importance is the thought that almost any type of constraint could be added to the L.P. and the lambda convergence would occur as long as the appropriate process change was identified by the above method. Other examples of such an application will be described in conjunction with the description of optimizing a reserve force selection and optimizing a first strike countervalue strike when it is known that a later strike on the same targets will occur.

E. OPTIMUM RESERVE FORCE CHOICE

In one of the basic scenarios that can be analyzed by use of this model (Figure 3), the side hitting first attacks only counterforce while maintaining a reserve for future strikes. The preferred reserve force is designated to be that one which results in maximum delta damage if the retaliation is against both force and value targets as diagrammed in Figure 3.

Several different techniques have been utilized in the reserve force choice methodology, however only the current, and best method will be described. The method utilizes the basic L.P. process plus the by-products of the multiplier method to determine the optimum reserve in very few iterations by the program.

Consider RED's allocation problem in his first strike counterforce. The allocation process he uses must determine the optimum division of the complete force into those weapons which are to be used in the first strike and those weapons which are to be placed into reserve. Additionally, the weapons being fired in the first strike must have specific targets assigned to them. These divisions and allocations must be such that the maximum delta damage results.

Once RED has decided upon a first strike allocation BLUE then can utilize his survivors to minimize the delta damage that RED is trying to maximize. Thus, the pure counterforce scenario of Figure 3 can be visualized as a first strike by RED followed by a scenario like that of Figure 1 except with BLUE hitting first. Viewed in this way, it can be seen that after RED has hit first it is necessary to go through a force value scale iteration process for BLUE so that his attempt to minimize the delta damage is represented.

After BLUE's optimization process is completed, it becomes appropriate to ask how RED might affect the outcome by striking differently in his first strike. If a different RED allocation which does increase the delta damage can be determined, it would then be necessary to give BLUE another opportunity to re-optimize the delta damage as much as he can. By appropriate use of such a sequence of fictitious plays, it is possible to finally arrive at a point where RED cannot determine a method for improving the delta damage and a max-min solution exists.

Consider the use of BLUE weapon lambdas as a basis for the values to place on BLUE's forces. It was shown in Part C that use of the weapon lambdas does result in convergence to at least a local optimum in the delta damage so it would be suspected that for the weapons RED uses in his first strike the values placed on BLUE forces could be derived in exactly the same manner.

To verify this, consider RED's overall problem of optimizing his first strike in two distinct steps. Assume, first that RED has decided which weapons should go into reserve. Given this situation, his only problem is to optimally allocate the first strike weapons.

Given that RED places a value of λ_B' on BLUE forces, his objective is to use his first strike weapons to:

$$\max \sum_{j=1}^{F_B} \lambda_{B_j}' \left[1 - \prod_{i=1}^{I_R} S_{ij}^{N_{ij}} \right] \qquad \text{(E-1)}$$

where:

F_B = number of BLUE weapon targets

I_R = number of types of RED weapons

λ_{B_j}' = lambda of whatever type of weapon the jth BLUE weapon is —

However, this is equivalent to

$$\max \sum_{k=1}^{I_B} \lambda B'_k \left[N_k - NS_k \right] \qquad \text{(E-2)}$$

where:

I_B = number of different types of BLUE weapons

N_k = total number of BLUE weapons of type k

NS_k = number of survivors of weapon type k

Following the same pattern as in equations (C-8) to (C-12), this can be shown to be the equivalent of

$$\sum_{k=1}^{I_B} \lambda B'_k \, NS^*_k \leq \sum_{k=1}^{I_B} \lambda B'_k \, NS'_k \qquad \text{(E-3)}$$

where:

* denotes the current solution

' denotes the previous solution

However, assuming that BLUE is using a Lagrangian allocation which has found a global minimum to $VD_B - K \cdot VD_R$, the result of his last allocation was such that

$$VD'_B - K \cdot VD'_R + \sum \lambda B'_k \, NS'_k \leq VD^*_B - K \cdot VD^*_R + \sum \lambda B'_k \, NS^*_k \qquad \text{(E-4)}$$

combining (E-3) and (E-4) result in

$$VD^*_B - K \cdot VD^*_R \geq VD'_B - K \cdot VD'_R \qquad \text{(E-5)}$$

which shows that use of the λ'_B for iterative BLUE force values does result in at least a local optima.

To understand the method for changing RED's reserve after he has optimized his attack for a current reserve, consider the marginal effect of changing one weapon from the first strike to reserve.

For the current reserve the weapon has a value as used in the first strike. This value is equivalent to λ'_{Rm}, the lambda of that weapon in the first strike. Also, if the weapon was placed in reserve and survived, it would have a value in RED's second strike. This value is equivalent to λ''_{Rm}, the lambda of that weapon in the second strike.

However, the weapon must survive BLUE's attack before it can accumulate any value in RED's 2nd strike. The probability that it does survive depends upon the nature of BLUE's utilization of the extra weapons that survive because RED has one less attacker in his first strike. It also depends upon BLUE's treatment of RED targets if he knew RED had placed one more weapon in reserve.

This complex interaction results in some net marginal value of the new weapon placed in reserve. The final net value takes on different forms depending upon the nature of BLUE's knowledge about RED's action.

Assume that BLUE has a launch detection or empty-hole detection system. Then if RED fires one less weapon BLUE will add one target to his counterforce system. The net value RED will achieve because he creates a new target can be determined by computing a Lagrangian for this new target. By the Lagrangian it can be said that

$$\Delta V = \min \left[\lambda''_{Rm} \prod_{i=1}^{I_B} S_{im}^{N_{im}} + \sum_{i=1}^{I_B} \lambda'_{Bi} \cdot N_{im} \right] \qquad \text{(E-6)}$$

where:

ΔV = net value to RED for the weapon placed in reserve

λ''_{Rm} = value of the RED weapon placed in reserve

λ'_{Bi} = weapon λ of BLUE weapon type i

S_{im} = probability of survival of a target of type m when attacked by one weapon of type i

The final net contribution to the delta damage of this new RED reserve weapon must be the difference between the gain by the reserve weapon and the loss because there is one less first strike weapon. Thus, the net benefit, $\Delta V'$ of one more reserve weapon in this case becomes

$$\Delta V' \approx \Delta V - \lambda'_{Rm}$$

This is only approximate because the λ'_{Bi} involved in ΔV are marginal values which can change for slight changes in the target structure. It is also approximate because λ'_{R1} might be at such a point that it could change for a small change in RED's attack level. The degree of approximation is probably very small for a one unit change in the reserve level.

If BLUE does not have an empty-hole detection system, he might have a bomb impact system which counts the number of each type of RED weapons impacting on him. In such a case, BLUE does not change the number of targets in his system but instead modifies the value placed on each target by the probability that a weapon is still at the base. This value applies to all bases of a given type.

In this case, BLUE would possibly change his way of attacking the RED targets. The net effect can again be obtained by using the Lagrangian.

Before the extra weapon was placed into reserve, the net surviving value of all weapons of a given type was given by

$$\Delta V_1 \approx N_m \left[\text{Min} \left[\left(\lambda''_{Rm} \frac{N_{RR}}{N_m} \right) \prod_{i=1}^{I_B} S_{im}^{N_{im}} + \sum_{i=1}^{I_B} \lambda'_{Bi} N_{im} \right] \right] \quad \text{(E-8)}$$

where:

N_m = total number of RED weapons of a given type

N_{RR} = current number of weapons in reserve for that type

After placing one more weapon in reserve, the above becomes

$$\Delta V_2 \approx N_m \left[\text{Min} \left[\left(\lambda''_{Rm} \frac{(N_{RR} + 1)}{N_m} \right) \prod_{i=1}^{I_B} S_{im}^{N_{im}} + \sum_{i=1}^{I_B} \lambda'_{Bi} N_{im} \right] \right] \quad \text{(E-9)}$$

Thus, in this case, the approximate total net benefit of adding a weapon into reserve becomes

$$\Delta V' \approx \Delta V_2 - \Delta V_1 - \lambda'_{Rm} \quad \text{(E-10)}$$

Generally the optimum solutions to equations (E-8) and (E-9) will be identical since for any reasonable values for N_m and N_{RR} it would be true that

$\lambda''_{Rm} \frac{N_{RR}}{N_m} \approx \lambda''_{Rm} \frac{(N_{RR} + 1)}{N_m}$ and the optimum integer attack strategy will not change. In such a case (E-10) reduces to

$$\Delta V' \approx PS''_{Rm} \cdot \lambda''_{Rm} - \lambda'_{Rm} \quad \text{(E-11)}$$

where:

$$PS''_{Rm} = \prod_{i=1}^{I_B} S_{im}^{N^*_{im}}$$

= probability of survival of an average target of type m with the current reserve level.

For the purpose of discussion, assume that this last circumstance is the case. Then it will be appropriate to place a weapon in reserve any time $\Delta V'$ is ≥ 0. Or, if

$$PS''_{Rm} \cdot \lambda''_{Rm} \geq \lambda'_{Rm} \tag{E-12}$$

Equation (E-12) essentially is an indication of the slope of the pay-off (delta damage) in the reserve force space. If the relationship of equation (E-12) is true, it simply says that a small increase in the reserve force should increase the delta damage in RED's favor. If

$$PS''_{Rm} \cdot \lambda''_{Rm} < \lambda'_{Rm} \tag{E-13}$$

if simply says that a small <u>decrease</u> in the reserve force should be desirable. When $PS_{Rm} \cdot \lambda''_{Rm} = \lambda'_{Rm}$ for all values of "m" the reserve force must be at a local optimum.

One obvious use of equations (E-12) and (E-13) is in the method of steepest ascents. That is, they can be used to determine which weapons would change the delta damage the fastest, a step could be taken in that direction and a convergence to a new set of λ''_{Rm}, PS''_{Rm} and λ'_{Rm} obtained. This process was attempted but found to be unacceptable because of the difficulty in choosing a step size. Too small a change in reserve force numbers caused slow convergence while large changes caused instability in the solution.

A satisfactory method currently in use is one which internally chooses a step size. This method is based on the idea that RED's first strike objective can be stated to be to maximize the sum of the value destroyed on BLUE forces plus an estimated value destroyed on BLUE value targets by his reserve force. In mathematical form, this is equivalent to

$$\max \sum_{k=1}^{I_B} \lambda'_{Bk} \left[N_k - NS_k \right] + \sum_{m=1}^{I_R} PS''_{Rm} \cdot \lambda''_{Rm} N_{RRm} \qquad \text{(E-14)}$$

where:

N_{RRm} = number of RED type m weapons placed in reserve

$PS''_{Rm} \cdot \lambda''_{Rm}$ = best estimate of value destroyed by each RED reserve weapon

If this is chosen as RED's first strike objective, it will cause the selection of that reserve force choice which comes the closest to a case where $PS''_{Rm} \cdot \lambda''_{Rm} = \lambda'_{Rm}$. (It is not always possible to achieve this equality for each weapon type because of the integer constraints.)

In this objective RED would allocate all his weapons, rather than only the first strike weapons assumed to be allocated for the objective of equation (E-2). The concept of allocation here thus includes allocation of weapons to targets and allocation of weapons to a reserve block. In equation (E-14) any weapon can be allocated to reserve at an estimated payoff of $PS''_{Rm} \cdot \lambda''_{Rm}$ or it can be allocated to a first strike. If allocated in a first strike, the payoff is in terms of a reduced NS_k. The optimum balance is the one where additional weapons placed in reserve bring less payoff than the same weapon fired in the first strike.

Considerable experience with this method has shown that convergence to the correct general reserve force magnitude usually does occur. However, in constrast to the value scale convergence techniques, it is not possible to prove finite convergence to a local optimum.

To understand why finite convergence is not guaranteed, consider the following. First, note that after RED's allocation of his previous reserve force by the Lagrangian method the following can be stated.

$$VD_B' - \sum_{m=1}^{I_R} \lambda''_{Rm} N'_{RSm} \geq VD_B^* - \sum_{m=1}^{I_R} \lambda''_{R_m} N^*_{RSm} \qquad \text{(E-15)}$$

where:

N_{RSm} = RED reserve force survivors of type m

With $N_{RSm} = PS_{Rm} \cdot N_{RRm}$ and the assumption that PS_{Rm} does not change we can say

$$\sum_{m=1}^{I_R} \lambda''_{Rm} PS''_{Rm} (N^*_{RRm} - N'_{RRm}) \geq VD_B^* - VD_B' \qquad \text{(E-16)}$$

which indicates that use of $\lambda''_{Rm} \cdot PS''_{Rm}$ leads to an overestimate of the change in BLUE damage, VD_B' to VD_B^*, when a reserve is changed from N'_{RRm} to N^*_{RRm}. Possible changes in PS''_{Rm} make the estimate even more optimistic since increases in the reserve generally reduces PS''_{Rm}.

By a similar use of equation (E-4), it can be shown that use of λ'_{Bk} as a value estimate for BLUE forces leads to an underestimate on the effect of changing the first strike weapon allocation. But, equation (E-4) applies to a static RED situation and not one where BLUE's targets are changing. This changing of BLUE targets can be shown to cause overestimates to occur. Thus, use of this combined objective function has a mixture of optimistic and conservative factors. The net effect being that there is no way to guarantee that the optimistic estimate of the value of changing the reserve doesn't overshadow the conservatism in the counterforce strike payoff change estimate.

The method is somewhat self-correcting in the sense that going from one reserve to another results in new estimates for the three parameters of interest. These new values tend to correct mistakes made in the previous choice. For example, if too large a reserve force is made, both PS''_{Rm} and λ''_{Rm} tend to drop somewhat which causes the next reserve choice to be somewhat smaller and, thus to correct the error.

Implementation of this concept is rather straightforward. In essence, placing a weapon in reserve is another potential strategy involving use of that weapon. This new type of "strategy" involves use of only one weapon of type "m" at a payoff of $PS''_{Rm} \cdot \lambda''_{Rm}$. In addition, this strategy does not involve any "target" so it does not enter into any target constraints like the strategies involving use of a weapon in the first strike.

Thus, a typical RED first strike linear program at any phase in the process contains strategies which are candidates for counterforce strikes plus these special strategies which represent the payoff for placing weapons in reserve. The lambda convergence process operates as usual.

A special case of the above is one where RED desires no more than a specified damage to BLUE (VD_B). In such a case the preferred reserve is that one which maximizes delta damage under the max. VD_B constraint. Since all damage to BLUE is done by RED reserve weapons, this amounts to the placement of a constraint on the size of the RED reserve. This can be achieved by adding the following constraint to the 1st strike L.P.

$$\sum_{m=1}^{I_R} FS''_{Rm} \cdot \lambda''_{Rm} \cdot N^{*}_{RRm} \leq \sum_{m=1}^{I_R} FS''_{Rm} \cdot \lambda''_{Rm} \cdot N'_{RRm} + VD_{B\ max.} - VD'_B \qquad \text{(E-17)}$$

where:

$*$ denotes new reserve choice

$'$ denotes last reserve result

$VD_{B\ max.}$ = max. allowed damage to BLUE

If $VD'_B < VD_{B\ max.}$ after the last reserve force choice, this constraint allows only enough additional weapons to become reserve such that the desired damage is exactly met. If $VD'_B > VD_{B\ max.}$, it will indicate that weapons must be removed from the reserve.

It is interesting to note that this constraint is used to advantage even when there is no specified limit on BLUE damage. In such a case

$VD_{B\ max.}$ is set equal to the total of all BLUE industrial value. Then the above constraint tends to reduce the possibility that an overly large reserve force is ever chosen and, thus helps eliminate extreme swings in the reserve force choices.

This discussion has been conducted as if the estimated net benefit of one more reserve weapon is given by equation (E-12). However, the method is exactly the same for payoffs as indicated by equations (E-7) and (E-10). The only change is to use the appropriate ΔV in place of $PS''_{Rm} \cdot \lambda''_{Rm}$ in the process just described.

It should also be noted that this method conceivably could be used for rapid convergence to a region near an optimal reserve force. Then another method could be used to refine the solution. However, this has not appeared necessary on the few cases to date where complete convergence did not occur.

A variant of the above method has also been tried and found to add some degree of stability to the process. In this variant the estimates for PS''_{Rm} and λ''_{Rm} are separated and dealt with on an individual basis. This separation occurs as follows.

First, remember that the "strategies" that went into the IF were on the basis of one reserve weapon type "M" always attaining a payoff of $PS''_{Rm} \cdot \lambda''_{Rm}$. This linear payoff estimate falls apart for large changes in reserve force and a better approach is to insert real strateties for the reserve weapons on the third strike targets. That is, indicate how many reserve weapons will achieve what level of damage on which third strike target.

These strategies can be computed for each target just like the strategies for the first strike targets are computed. However, since the reserve weapon must survive the second strike attack, the following relationship holds:

$$W_{Rm} = \frac{W_{3m}}{PS''_{Rm}} \qquad \text{(E-18)}$$

where:

W_{3m} = number of weapons going to a given target in strike 3.

W_{Rm} = number of weapons which must be placed in reserve in order to have W_{3m} weapons available for the third strike stragety.

It is W_{Rm} that is inserted into the specific third strike strategy.

The net effect of this variance is that true strategies are generated in the first strike LP for all first and third strike targets. However, the number of weapons which must be held in reserve for each individual third strike strategy is computed by the use of (E-18). Thus, depending upon the value for PS''_{Rm}, a given third strike strategy might use up considerably more reserve weapons than another strategy which uses a different weapon with a different survival probability.

This approach removes the linearity assumptions inherent in use of the λ''_{Rm}. It does not reduce the problem of correctly estimating PS, however. Estimate of that factor thus is the key issue in this optimal reserve force methodology.

F. MULTI-STRIKE WEAPON ALLOCATIONS

Demonstrated in Figure (4) is another special scenario which can be analyzed by use of this model. This scenario was designed to allow analysis of the special case where RED attacked both counterforce and countervalue in a first strike but BLUE launches a retaliation before all RED weapons are launched. This allows both sides to attack counterforce one time. In this case, the program does not optimize a reserve for RED (since it obviously would be a zero reserve). Instead, there is a specified RED remainder which BLUE can attempt to destroy by counterforce attack.

This scenario is a generalized version of the basic scenario (Figure 1). If the remainder is specified to equal zero, the basic scenario is the result. However, this scenario contains an ingredient not existing in the simpler one.

As RED is setting up his first strike, he must allocate some weapons against BLUE industry. It would be optimum for RED if he could conduct the 1st strike countervalue attack in such a way that the survivors from his remainder are allowed to perfectly augment his first strike. That is, RED should allocate so that the total damage accumulated by both countervalue strikes is maximized. If RED ignores the existence of his remainder, the first strike might be conducted in such a way that the remainder survivors couldn't accomplish much.

There are obvious questions about RED ever being capable of predicting his own survivors. Before such questions are investigated, it is necessary to develop a methodology for RED to use in allocating his 1st strike assuming he knew exactly what his survivors would be. Given this methodology the non-perfect information case can then be analyzed.

One point of view about RED's allocation problem is that he must decide which countervalue targets to label first strike targets and which to label second strike targets. This is a concept very similar to the one described in Part E where RED had to label his weapons as first or second strike weapons.

Viewed in this way RED's problem becomes one of selecting a first strike objective which somehow allocates weapons to targets and targets to a first or second strike category. (Given the allocation is achieved BLUE will minimize the delta damage to the best of his ability by optimum use of his survivors.) The problem is to select an objective which leads to a progressively improved decision about which targets to set aside for attack by RED's second strike.

One obvious limitation of this target allocation concept is that RED first and second strike weapons never attack the same target. A target is either attacked in the first or the second strike or is not attacked at all. This limitation is felt to be reasonable and no effort has been made to avoid it. Some comparisons have been made with the alternate approach of allowing mixed attacks and no significant improvement in total value destroyed was observed.

It can be shown, in a manner similar to the one of Part E, that if RED chooses a specific set of targets to be reserved for his second strike the use of lambda value scales do lead to optimization of the delta damage. The basic problem is to show that there is an objective function which will lead to a target labeling which causes overall maximization of the delta damage.

The similarity of this problem to the optimum reserve force problem extends into the choice of an objective function and the lack of a convergence assurance. As in the reserve force problem, consider the information available to make a decision about the advisability of a small change in a given target allocation. The basic information includes an estimate of the weapon lambdas for BLUE survivors, λ'_{Bi}, and an estimate of RED survivors from BLUE's retaliation. Use of this information can be utilized by making the first strike objective

$$\text{maximize} \quad VD_{BF} + VD_{BV1} + VD_{BV2} \tag{F-1}$$

where:

VD_{BF} = estimated value destroyed on BLUE forces

VD_{BV1} = value destroyed on RED's first strike on BLUE value targets

VD_{BV2} = estimated value destroyed on RED's 2nd strike on BLUE value targets

Each of the above terms are equivalent to

$$VD_{BF} = \sum_{i=1}^{I_B} \lambda'_{Bi} \; (N_{Bi} - NS_{Bi}) \tag{F-2}$$

where:

N_{Bi} = total number of targets containing BLUE type i weapons

NS_{Bi} = number of surviving targets with BLUE type i weapons

$$VD_{BV1} = \sum_{j=1}^{J} V_j \; (N_j - N''_j) \; PK_j \tag{F-3}$$

where:

V_j = value of BLUE value target type j

N_j = total number of BLUE targets of type j

N_j'' = number of BLUE targets of type j set aside by RED for his 2nd strike

PK_j = average probability of kill of BLUE targets of type j as attacked by RED in his first strike

$$VD_{BV2} = \sum_{j=1}^{J} V_j \left(N_j''\right) PK_j^* \qquad \text{(F-4)}$$

where:

PK_j^* = average probability of kill of BLUE targets of type j as attacked by RED in his second strike.

Using this objective results in

1) A choice of which targets should be reserved for RED's second strike.
2) Allocation of RED first strike weapons to counterforce and allowed countervalue targets.
3) Allocation of RED's estimated second strike weapons to specific targets.

This results in a balance such that placing another target in the second strike category (N_j'' increased by one) has less pay-off than reserving that target for the first strike. In RED's first strike, he will choose a specific allocation of his weapons and a specific set of values for N_j'' such that the objective function is maximized. The allocation of his weapons affects the NS_{Bi} factor in VD_{Bf} and the PK_j factor in VD_{BV1}.

As in the reserve force method, it is not possible to guarantee continuous improvement in the delta damage from step to step. However, experience has shown that convergence rarely doesn't occur. In those instances when convergence doesn't occur, the oscillation in delta damage has been very small (a percent or two, at most).

Implementation of the above concept is very straightforward. In the first strike of the scenario, the countervalue and counterforce targets are all represented. The number of attacking weapons is the actual number of RED first strike weapons. Additionally, there is inserted into the L.P. an additional set of constraints to represent the estimated number of RED attackers of each RED weapon type in his second strike on BLUE value targets. These weapons are constrained to attack countervalue only.

The net result of optimization of this type of L.P. is a maximization of equation (F-1). After completion of the first strike, BLUE then proceeds to optimize his attack with his survivors (NS_B). After BLUE optimizes his attack a new estimate of RED attackers in his second strike is obtained and the process is repeated. After some iterations, the estimate of RED survivors does not change and the process stops.

If RED has a damage limitation on BLUE, such a constraint can be inserted into the first strike L.P. in the same manner as previously described for other cases. No interference with the convergence process occurs.

Note that the above process allows RED to set his first strike based on what he believes his remainder survivors will be. The result when he misestimates the number of survivors can be obtained by use of the standard misestimate features in the model. That is, RED will misestimate his survivors because he misestimates BLUE's attack plan, some weapon characteristics or some other data. The impact of that misestimate is partially reflected in RED's first strike plan being non-optimum and the degree of non-optimality can be determined by use of the misestimate option discussed in Section I.

Basically, the process is as follows. RED sets up his first strike based on his best estimate of his survivors. The program "locks" in that plan. It then proceeds to compute the real RED survivors and allocates them in the best way possible consistent with RED's level of retargeting capability. The result is some damage by RED's forces but not the optimum level.

G. RANDOM AREA DEFENSES - BOMBER AND MISSILE

In this model area defenses are designated to be those defenses active between the launch phase of weapon flight and the exposure of the weapon to terminal defenses. It is assumed in this representation that unreliable missiles fail during the launch phase so that only reliable missiles confront the defense. Aircraft are assumed to be either unavailable or able to attempt penetration of the perimeter defenses. The reliability factor applies to the bomb or Air-to-Surface Missile (ASM) at the time of arming which is assumed to occur just prior to encountering terminal defenses. These assumptions are presented pictorally in Figure G-1.

In general, area defenses affect offensive forces by either a reduction in numbers of weapons that can impact on all (or a subset) of targets or an increased uncertainty that a particular weapon will reach its target. This section describes the currently programmed random bomber and missile defenses in terms most like the latter of these two effects.

Bomber defenses included in the model are perimeter (manned interceptor) defenses sensitive to number of bombers attempting penetration and resulting in uncertainty of a bomber arriving at the arming phase of its attack.

Random area missile defenses are non-killable defenses which attempt intercept against incoming RV's and decoys chosen at random, without regard to RV destination or sequence of fire. The effect is an increase in the uncertainty that a particular weapon will arrive in the target area.

All area defenses considered are "whole country" in coverage, i.e., no target lies outside the influence of the area defense if such defense is present.

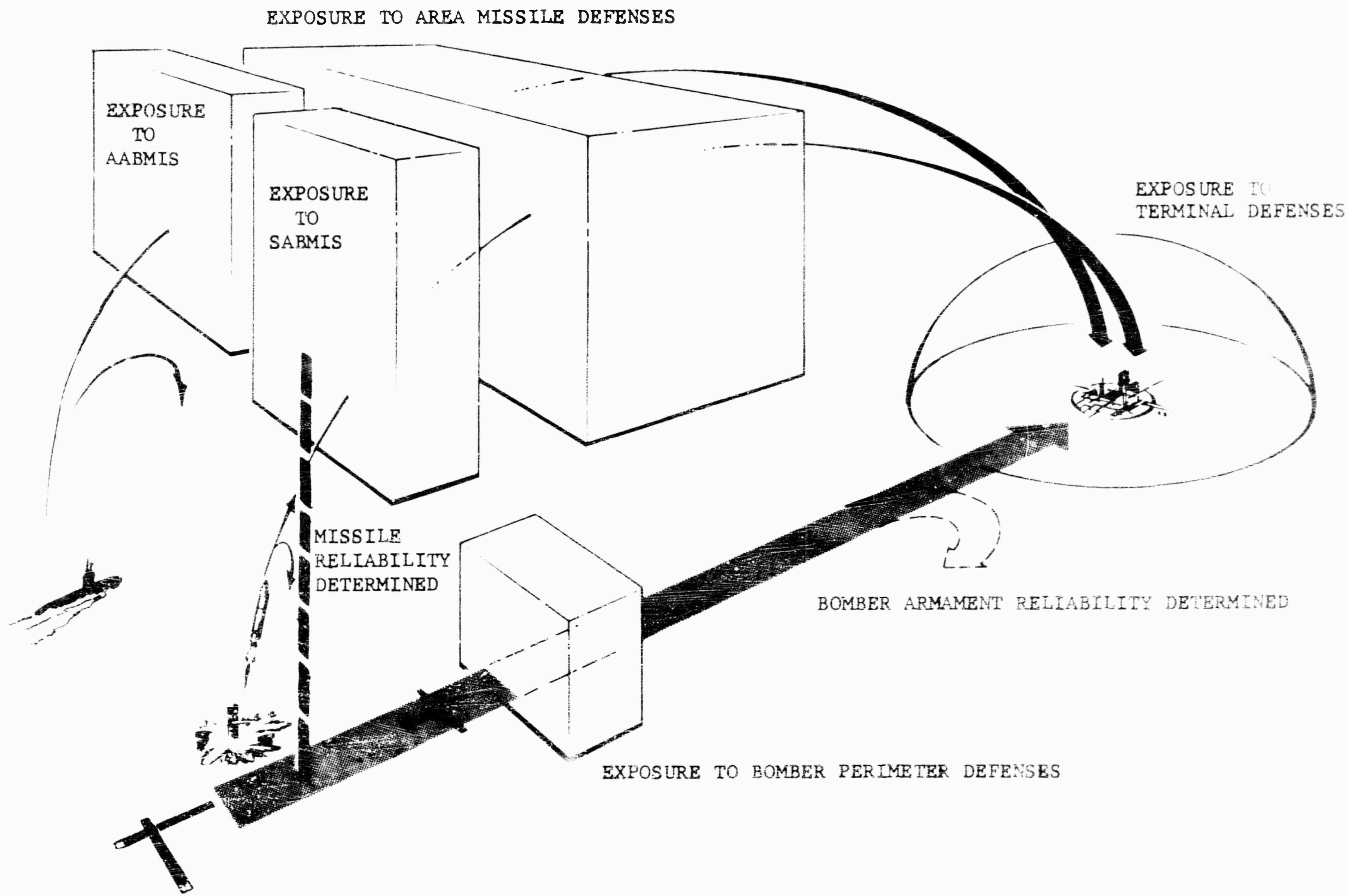


FIGURE G-1 SCHEMATIC OF BOMBER AND MISSILE DEFENSE ARENAS

1. Bomber Area Defense

a. General

The probability of strategic penetration encompasses a massive number of events and their associated probability. In designing a model to reflect the most prominent of these events and still allow minimum computational requirements must necessarily be a compromise which, hopefully, does not overly offend the arsenal exchange analyst. It may be of benefit to elaborate on the events considered in most penetration analyses, if for no other reason than to gain an appreciation of the scope of the problem.

The threats to an airbreathing penetrator are not only varied, but relatively cheap and therefore numerous, have high cumulative effectiveness (dependent on penetrator aids), may be reuseable (dependent on time), and possess at least some degree of autonomy. The spectrum of threats varies from supersonic (hypersonic?) manned interceptors armed with air-to-air missiles to literally hand-thrown rocks.

There are three major threats: manned aircraft, surface-to-air missiles, and anti-aircraft artillery (AAA), generally considered in penetration studies. Although AAA is frequently ignored, it is reasonably effective against low level penetrators if in sufficient quantity (e.g., North Vietnam). The problem of massing along all suspected penetration corridors prohibits inclusion of AAA as a major threat in our study. The probability of random encounters with AAA while at low altitude, particularly while in the vicinity of military installations may be quite high, but it is assumed that careful planning of flight paths minimizes the likelihood of these conditions and is therefore ignored.

The most versatile defender is still the manned interceptor. It can operate effectively with little command and control, although early warning, real-time data processing, and target vectoring enhance the inherent capability many fold. Armament generally consists of guided or unguided air-to-air missiles with fire control systems having varied range and altitude capabilities, or cannons which are manually or electronically fired. The effectiveness of the manned interceptor is dependent on such parameters as probability of detection, identification, and conversion to the target, evasive action by the target, armament of the interceptor, and armament effectiveness. The number of intercepts possible during penetration is dependent on the number of available interceptors, number of passes an interceptor can make during a sortie, number of passes per intercept, and number of sorties an interceptor can make during the battle.

The other defense of concern in this analysis is the surface-to-air missile (SAM). Although SAM's may be deployed as perimeter defenses and scattered throughout any penetration corridor, it was assumed that these defenses could be evaded or made ineffective except during the terminal attack. The modeling of SAM defenses in a terminal mode is described in Chapter IV - Section B.

Additional considerations included in detailed penetration studies include the probability of clobber (flying into the ground during low altitude penetration) and crew safety during attack (i.e., displacement from nuclear blasts of the bombers weapons). Effective terrain avoidance radar and aerodynamic indifference to gust loading preclude probability of clobber as a major consideration. The advent of stand-off ASM's preclude the constraints of crew safety. Since these developments, if not current, are being heartily pursued, these additional bomber characteristic constraints have been ignored.

b. Bomber Area Defense Model

The model for inclusion of a manned interceptor perimeter defense is an extreme aggregation of the events discussed above. Stated simply, a bomber either was or was not subjected to the perimeter defenses and if so, either survived the encounter or did not, i.e.,

$$P_p = (1 - P_E) + P_E (1 - P_{AI})^{\frac{I}{B}} \qquad \text{(G-1)}$$

where: P_p = probability of bomber penetration

P_E = probability of encountering perimeter defenses

P_{AI} = probability of bomer kill by an interceptor pass

I = total expected interceptor passes available

B = total number of bombers presented to the defense.

Since P_E, P_{AI}, and I are input data, a discussion of these parameters is desirable.

The probability of encountering the perimeter defense (P_E) includes the probability a particular aircraft was acquired by the defense either through early warning or local observations, the probability that communications are adequate to inform the interceptor base or airborne interceptors, the probability that the command and control is adequate to vector a sortie against this penetrator, the probability that an interceptor is available for the sortie (includes POL, armament, reliability of critical components, geography, and combat range/time specifications), and the probability of successful vectoring of the interceptor so that the penetrator is acquired by the fire control system inside the interceptor conversion barriers. Because penetration requires a finite time that may be large with respect to any of the above activities, P_E represents the cumulative probability of being within at least one conversion cone of an interceptor.

P_{AI} or pass lethality is an aggregate containing weighed effectiveness of all armaments that may be used which includes weapon reliability, warhead configuration, and round-to-round dispersion or CEP, aspect angle of convergence, fire control accuracy and reliability, lethal area of the target (a function of the aircraft, warhead, and aspect angle), and evasive or defensive action by the penetrator.

The number of expected passes available to the defense (I) is the product of number of aircraft, aircraft availability number of passes per sortie, and expected number of sorties during penetration.

Obviously, the expected number of intercepts (or passes) against a particular penetrator varies with exposure (penetration time), flight plan (geography of interceptor bases in relation to penetrator path), and flight profile (Hi-Low-Hi, Low-Low, etc.).

The model assumes no penalty to the acquired penetrators in terms of anticipated defense level as P_E varies, i.e., I/B and P_{AI} are not functions of P_E. This assumption precludes the possibility of an "optimum" P_E and dictates a linearly decreasing P_P as P_E varies from zero to one.

The probability of "losing" a target after an unsuccessful pass or the inability of the next attacker to acquire the target should all passes be unsuccessful is not included in the model. The variables P_E and I are not affected by previous ICBM strike nor by any stand-off ASM capability employed as a defense suppression missile (DSM).

With the assumption that in-flight communications are minimal, the net effect of a kill by the perimeter defense is the uncertainty that a particular bomber (bomb) will reach its programmed destination. Since this behavior is similar to the effect of bomb reli ity, P_P is modeled as a reliability modifier, i.e., $R^* = P_P \cdot R$. The resultant reliability, R*, is used in computing the expected kill of a particular target.

Certain cases must be carefully analyzed to insure the assumptions of this model do not mislead the user. For instance, when the bomber force on a side is of considerable strength and the perimeter defense is effective, large variation in the probability of penetration derived here may occur dependent on the previous attack against bombers. This variation may range between bomber dominance of value target destruction to bomber subordinance to every ICBM type. The optimum is most likely at neither extreme. It is possible that the model assumptions for this case would indicate the lower extreme as the optimum. Sensitivities to the assumptions by varying inputs to the model may then be desirable.

It is realized that the degree of aggregation in this model is extreme. However, most alternate methods are incompatible with the Arsenal Exchange Model in computational efficiency. At the current time it is planned that future activities will include investigations of results from comprehensive studies to determine whether additional parameters should be introduced. Comparative analysis with more sophisticated models has not been performed and therefore, we have no current way to measure the accuracy of the model until such comparative analyses are made.

Attempts to optimize future arsenals may include balancing the ICBM/Bomber forces to enhance the capabilities of each and to force continued development of two or more types of defense by the opposition. This model is not necessarily appropriate for such investigations and will be re-evaluated when that capability exists (see Section IV-J).

c. <u>Effects of Area Bomber Defense Model On Value Scales</u>

The variable B in equation (G-1) (number of bombers) is computed as number of bombers of all types off-on-warning plus available survivors of any previous attack. Therefore, lowering B by a previous attack lowers P_p for the retaliating bombers. This subtle benefit may be thought of as a second value for attacking bomber bases, i.e., destroyed bombers do no damage and a reduction in the bomber force enhances the defense, decreasing the value of each bomber in the retaliating bomber force. Chapter IV, Section C contains a discussion of the method used in analyzing the impact of this factor on the value placed on a bomber base when it is a target.

This ability to relate defense capability to weapon value and thereby, to ICBM attacks on these weapons could not be achieved by an inputed reliability modifier. For this reason, it is felt that this bomber penetration model is a logical first step toward the real impact of bomber penetration capabilities.

2. Random Area ABM

a. Rational

The initial attempt at modeling area ABM effects was similar in scope to the area bomber defense discussed above. The primary assumptions are the offense estimates the number of RV's that will be killed by the defense, but has no control over which particular ones, and the defense attrits RV's without benefit of deducing destination, but rather by random choice. The option of the offense to suppress the defense by direct attack was not modeled.

This defense may be thought of as "medium capability" lying between offensive dominance over the defense (terminal or subtractive) and defensive dominance over the offensive (preferential). However, exact determination of random defense capability with respect to optimally deployed terminal and preferential has not been performed. This model has sufficed as an initial ABM area defense representation, but multiple strike scenarios make implementation of this type of defense difficult. Therefore, it is a limited program capability. These model limits will be discussed in sub-paragraph d.

b. The Random ABM Model

The effect of this type of area defense is essentially the same as a reliability degradation. As noted in Figure G-1, weapon reliability for missiles is assumed to be determined prior to possible acquisition by area defenders. Pragmatically, this reduces the attack as viewed by the defense, and increases the uncertainty of safe arrival in the allocation planning done by the offense.

The model states penetration may occur in any of three ways:

- The RV is not "acquired" by the defense, i.e., no attempt is made to intercept.
- The RV is acquired but there are no remaining interceptors.
- The RV is acquired and an unsuccessful intercept attempt is made.

Stated mathematically:

$$P_P = (1. - P_a) + P_a (1 - D/S_N) + P_a \; D/S_N (1. - P_I)^I \qquad \text{(G-2)}$$

Where:

P_a = Probability of acquisition

D = Defense intercept capability

S_N = Total number of reliable, acquired, and undiscriminated objects presented to the defense.

P_I = Probability one interceptor kills the RV

I = Number of interceptors fired per intercept attempt

The connotation of acquisition in these discussions is not exclusively the act of "seeing" an incoming weapon (which may be assumed to always happen), but, includes all factors other than exhaustion which preclude intercept attempts. Such factors as time delays, nuclear blackout, queues in the fire control systems, and launcher availability contribute to lack of "acquisition."

Defense intercept capability is the smaller of; (a) the total number of interceptors ($\overline{D}$) divided by interceptors per intercept attempt (I) or; (b) the total objects at the defense, as defined by S_N below.

$$D = \text{MIN} \left\{ \frac{\overline{D}}{I}, S_N \right\} \quad \text{(G-3)}$$

$$S_N = \sum_{N_W^*} P_a \cdot R \cdot S_W \cdot (1. + (1. - P_D) \cdot d) \quad \text{(G-4)}$$

Where:

N_W^* = Number of missile weapon types

R = Reliability of the weapon

S_W = Number of available, surviving RV's fired

P_D = Probability of detecting that an incoming object is a decoy

d = Number of decoys per RV

Given the probability of penetration, P_P, as computed from the above relationships, it is utilized as a direct modifier of the RV reliability. This "effective" reliability is then used in the appropriate damage computations.

For the circumstance where $\frac{\bar{D}}{I}$ exceeds S_N, this model has been generalized. This generalization allows distribution of the excess interceptors over the objects (S_N) in units equal to I. That is, once each object has been allocated I interceptors, it is assumed that another pass through the objects is made with each one getting another I interceptors, etc. until the excess defenders are allocated.

The effect is that equation (G-2) becomes:

$$P_p = (1 - P_a) + P_a \left(1 - \frac{\bar{D}}{S_N \cdot I} + IE\right) * E^I$$

$$+ P_a \left(\bar{D}/(S_N \cdot I) - IE\right) (1. - P_I)^I * E^I$$

where:

IE = Largest integer in $\bar{D}/(S_N \cdot I)$

E = $(1 - P_I)^{IE}$

c. Effects of Random ABM On Value Scales

As with the perimeter bomber defense, the probability of penetration is dependent on the total number of weapons fired. The similarity indicates the need for value scale modification in this case also. The Chapter IV-C discussion is appropriate for this situation.

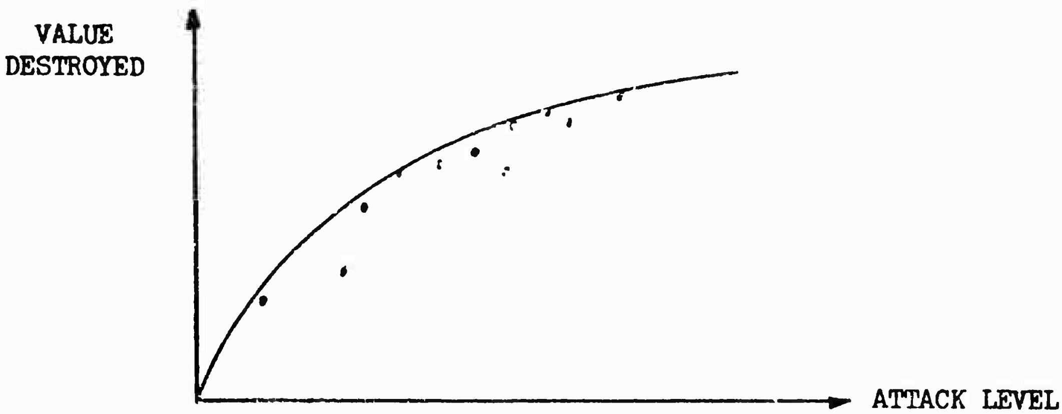
d. Limitations of the Random Area Defense Model

The major limitation in using this model is related to its dependence on knowing in advance the number of incoming objects (S_N) on a given strike. For those scenarios allowing a choice of defending against more than one strike, the number of attacking weapons in each strike can be a floating variable. For example; if an optimum reserve force is to be chosen, (see Figure 3) the first and third strikes may experience considerable variation in force size and composition as various levels of reserve are analyzed. A CV/CF first strike with a CV third strike (See Figure 4) also has a variable number of objects per target since value targets may be attacked either in the first or third strike and the attacker can be splitting his attack between counterforce and countervalue attacks.

An additional problem exists in that a CF first strike scenario involves the offense making a decision about how large the first strike should be with the decision dependent upon the effectiveness of his weapons in that strike. However, he cannot say how effective his weapons are until he decides how large his attack will be. In addition, because of the form of the probability of penetration model, there is a non-convex nature to the weapon effectiveness as a function of the total force size, which emphasizes the offensive problem of predicting individual weapon effectiveness.

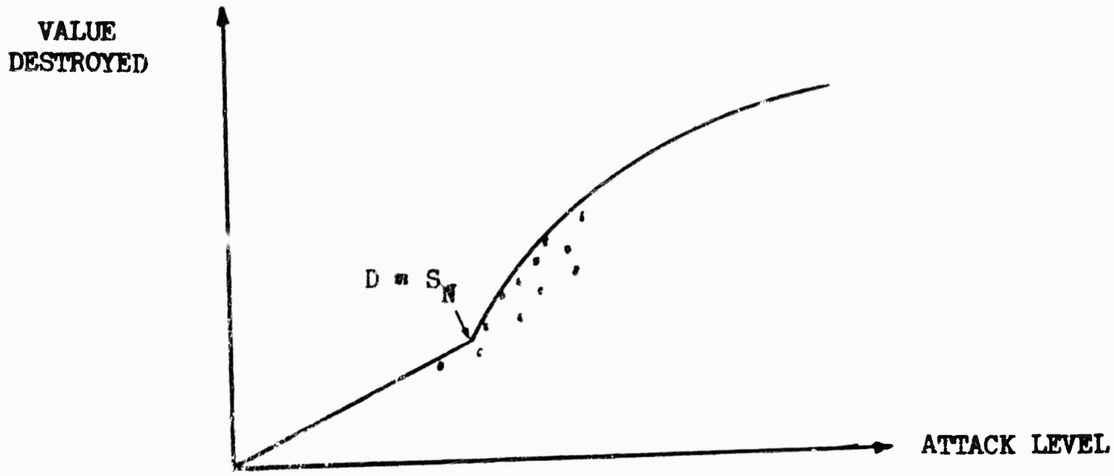
For example, the break point at $D = S_N$ in the computation of P_P causes a change in computation from a linear to an exponential type of damage function. This break is reflected in value destroyed as a function of attack level.

A hypothetical plot presenting this problem may enhance understanding.



The solid line represents maximum value destroyed (i.e., the value destroyed if the attack level is of the optimum composition and allocation) while the dots represent the value destroyed by a non-optimum force allocation of the same magnitude. Determination of the optimal allocation for the total target structure is obtained by cellular optimization as discussed in Section IV-A.

The inclusion of random defenses modifies this curve as follows:



The function is now non-convex over a portion of the possible attack. (It must be remembered that the curve is hypothetical in that each point on the curve is an optimized arsenal allocation. It should also be noted that these functions have as many dimensions as the number of weapon types.)

The slope of the function in any dimension is approximated by the appropriate weapon lambda. However, this approximation is dependent on the convexity of the objective function, which is not guaranteed when random defenses are present. This implies a dependence between the "optimized" cells, i.e., the optimum lagrangian for a cell is dependent on the number and composition of other cellular strategies chosen.

This can be stated directly if a cellular Lagrangian for an individual target is examined:

$$H = V \cdot (1 - RP)^{N} - \lambda N$$

Where:

H = the Lagrangian value

V = value of the target

R = the degradation in kill certainty due to the action of the random defense = P_P

P = probability of kill by a single RV of normal reliability

N = number of weapons fired

λ = Lagrangian cost

But R (= P_p) by equation (G-2) is dependent on S_N, which by equation (G-4) is dependent on the total number of RV's and decoys allocated. Only if S_N is constant can the optimum Lagrangian for each target be computed -- and only if the optimum Lagrangians are available can S_N be established.

Since P_P is peculiar in that computation is either linear or exponential, an additional problem is exposed. If either were always true, iteration rules could be implemented (e.g., lambda modification) that would allow conversion. Estimations of compositions of forces needed to exceed the D = S_N breakpoint may be computed but are themselves a large set of vectors having as many elements as weapon types and implementation in the Lagrangian or linear program is not generally feasible.

It is felt that these problems could be resolved, perhaps by nonlinear programming techniques. However, the concept of random defense seems very approximate and it was suspected that a more logical and fruitful approach would be to look at other reasonable area defense doctrires. An understanding of the impact of the defense doctrine could be obtained by modeling a subtractive defense (if the defense is really that indiscriminate) or precommit (if the defense is really that much in control).

Therefore, <u>the capability remains for a side suffering one attack during the game to possess random area defenses</u>. It is not anticipated that any further level of effort will be devoted to modeling multiple strike capability for random defenses. If compatible defenses are desired for a variety of scenarios, then this defense doctrine should not be used.

e. <u>Defense Marginal Utility</u>

In several uses of this program, e.g., optimal budget allocations (Section J) and force target values (Section C), it is necessary to know what one more unit of defense is worth in terms of less value destroyed on the defender. This section derives the method used to determine such a defense marginal utility.

In general the value destroyed on a given target is expressed by the relationship of equation (G-5). (See Section B.)

$$VD = V \left[1. - (1 - p)^{W-T} \right] \qquad (G\text{-}5)$$

Where:

V = value of the target

W = number of attacking warheads ($\geq T$)

p, T = damage function parameters

As weapon reliability varies, such as result from random area defenses, the p and T parameters of the damage function are modified in quite a complex manner. In general, as reliability is reduced, the effect is to lower the p parameter and increase the T parameter. As a result, it is possible to approximate equation (G-5) as follows:

$$VD \approx V \left[1 - (1 - P_p \cdot p^*)^{W - \frac{T^*}{P_p}} \right] \qquad (G\text{-}6)$$

Where:

p^*, T^* = damage function parameters with probability of penetration equal to 1

P_p = probability of penetration as defined by equation (G-2)

If the number of defenders was to change by one unit, the delta change in the value destroyed <u>on that target</u> can be approximated by:

$$DVD \approx \frac{\partial VD}{\partial P_p} \left(\frac{\partial P_p}{\partial D} \right) \qquad (G\text{-}7)$$

Where:

DVD = delta in the value destroyed on this target because one more interceptor is present in the area defense

By use of equations (G-6) and (G-2), the two components of (G-7) can be developed as follows:

$$\frac{\partial VD}{\partial P_p} \approx V_p^* \left[W - \frac{T^*}{P_p} \right] \left[1 - P_p p^* \right]^{(W - \frac{T^*}{P_p} - 1)} \qquad (G\text{-}8)$$

$$- \frac{T^* V}{P_p^2} \left[\ln \; (1 - P_p p^*) \right] \left[1 - P_p p^* \right]^{(W - \frac{T^*}{P_p})}$$

$$\frac{\partial P_p}{\partial D} \approx \frac{P_a}{S_N} \left[(1 - P_I)^I - 1. \right] \qquad (G\text{-}9)$$

These equations apply under the assumption that a slight change in total interceptors will not cause the allocations to change. (That is, W remains constant on the target.) Additionally, it is assumed that $D/I \leq S_N$. That is, the offense dominates the total attack. (If this latter assumption is not true, a minor modification to the process is necessary.)

Given the above relationships, the total marginal utility of a defender can be obtained by computing DVD for every target being attacked and sum over all such targets. This then will accumulate the small delta effects caused by a unit change in the number of interceptors. This is the process used in the program to compute the defense marginal utility. The relationship resulting is as follows:

$$MUI = \sum_{j=1}^{T} \frac{\partial VD_j}{\partial P_p} \left(\frac{\partial P_p}{\partial D} \right) \qquad (G\text{-}10)$$

Where:

MUi = marginal utility of one more interceptor

T = total number of targets

j = target subscript

In [illegible], equation (G-8) has been found to lead to accurate predictions of DVD when slight changes in reliability occur. However, [illegible] is also [illegible] such things as measuring the interaction of [illegible] probability and target capacity. (See Chapter IV-0.) In this latter circumstance the reliability of a given weapon might change rather considerably and the basis for (G-8) gets to be a little [illegible].

A better (more stable) computation procedure to replace (G-8) when larger reliability deltas exist is the following:

$$\frac{\partial VD_j}{\partial P_p} \approx \left[2 W_j \quad \lambda_w + \quad \lambda_j \right] \cdot \left[\frac{\Delta P_p}{2R} \right] \qquad \text{(G-11)}$$

where:

λ_w = weapon lambda for the weapon type attacking target.

W_j = number of weapons attacking target j.

λ_j = target type j target lambda.

ΔP_p = delta change in weapon reliability.

R = basic weapon reliability for the weapon attacking target j.

This relationship can be used in (G-10).

The logic behind this approximation is as follows. First, weapon lambdas have been found to be rather stable in behavior and they usually have a value which is proportional to weapon reliability. Second, the value destroyed in a given strategy can be expressed as:

$$VD = W \lambda_w + \lambda_i \qquad (G-12)$$

where the symbols have the same meaning as in (G-11). Normally, VD is not quite proportional to the reliability change, whereas the average between W λ_w and VD is close to being proportional. These facts led to an experiment which utilized (G-11) as an approximation and that experiment revealed that the approximation of the whole was more accurate than the (G-8) method. Thus, it is the one in current usage.

f. Extension to Forward Area Defenses

In addition to the normal random area missile defense assumed to operate over a country, there is a capability to allow for forward defenses which represent additional berriers that the missiles must pass through. There are two such forward deployed defenses allowed: AABMIS (Airborne Anti-Ballistic Missile) and SABMIS (Shipborne Anti-Ballistic Missile) defenses.

In this program these defenses are of the random type with the main distinction being that AABMIS is assumed to be used exclusively against SLBM's and SABMIS is used exclusively for ICBM defense. In either case, their effect is to reduce the number of objects arriving at the normal area defense, or at the terminal defenses.

Because these defenses can take action very early in the missile launch phase, there is a provision for destruction of the missile prior to complete dispersion of all warheads or decoys carried by the missile. The factor is allowed for by replacement of equation (G-4) with the following:

$$S_N = \sum_{N_W^*} P_a \cdot R \cdot M_W \cdot \left[1 + PDWF \left[WHPC - 1 + WHPC \cdot d \, (1 - P_d) \right] \right] \quad \text{(G-11)}$$

Where:

WHPC = Number of warheads carried by each missile

PDWF = Decimal fraction of objects deployed prior to defense kill

M_W = Number of missiles of type W in the attack.

There is no other modification of the basic random defense methodology.

The probability of penetration for the forward-based defenses is utilized as a direct modifier of the RV reliability. Thus, it will impact on all damage computations, and upon the number of objects arriving at any national area defense.

H. PREFERENTIAL DEFENSE

Within the context of this program the concept of preferential defense is used to denote the situation where the defense has the last move in the offense-defense interaction. That is, when preferential defense exists, the sequence of events is as follows.

First, the offense sets up his strike plan using an estimate of the number of interceptors the defense has and knowing that the defense will choose to defend the targets which allows him to save the maximum value. Then, the offense launches the attack and the defender inspects the total attack and decides which specific weapons he should defend against in order to minimize total damage to himself.

Within this concept two problems are of major importance. First, the offense must develop a plan for his strike which will leave him with a maximum value destroyed after the defense has acted. Second, there must be some allowance for a leaky defense. That is, a defense which cannot do precisely what his optimum is.

This section describes the methods used in this program to address each of these problems. The process is under some revision, mainly to reduce running time, but the fundamental concept is that which will be described.

1. Optimal Offense Strategies

A common characteristic of the offense-defense interaction problem is that, ideally, the first move should be made so that the last move opponent has a spectrum of possibilities which all have a constant return to him. For example, in the terminal defense problem, the defender should spread his interceptors so the offense has a return per weapon on each target ≤ some constant value. The defense should not put interceptors on a target, if possible, while some other target is still bringing a higher return per weapon to the offense.

In the case of preferential defense, the offense should attack each target in such a way that when the defense is looking for a place to put his interceptors he sees a constant return for each interceptor. To further understand this concept, consider Figure H-1, which is representative of the perfect attacker and perfect interceptor problem.

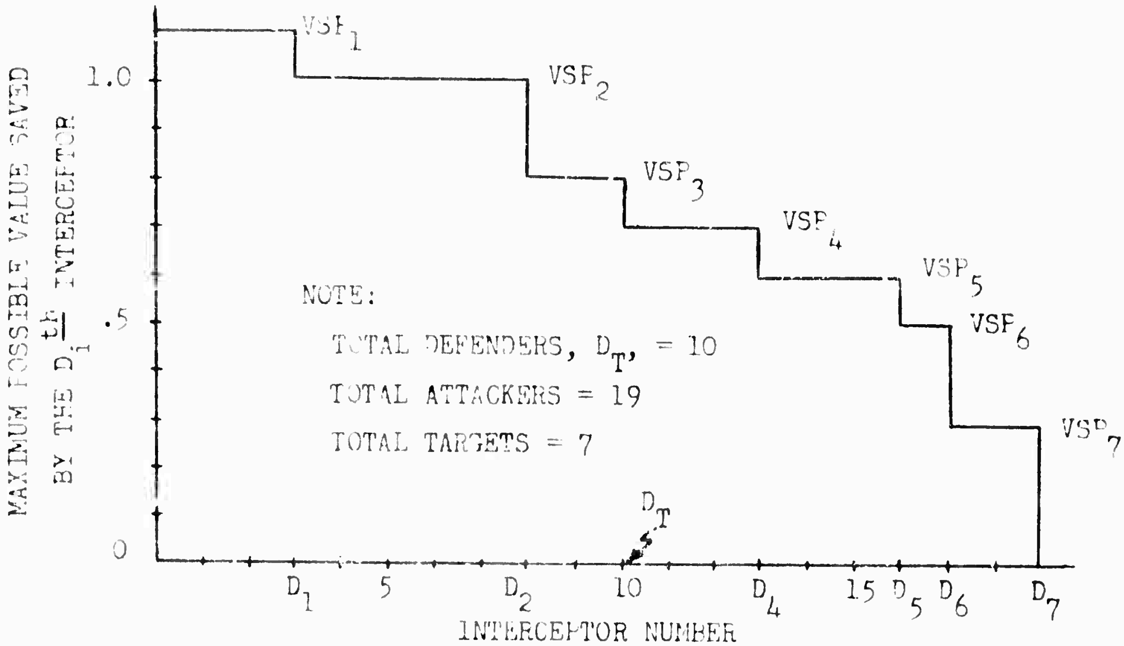


FIGURE H-1 DEFENSE VIEW OF HIS OPTIONS

Figure H-1 is the result of the defense inspection of all offensive strategies and ordering them in terms of the value saved per interceptor fired if a specific attack on a target is intercepted. In this figure, the defense found one offensive strategy which would require D_1 interceptors to nullify the attack at a return of VSP_1

per interceptor. The second best strategy would require $D_2 - D_1$ interceptors at a return of VSP_2 per interceptor, etc. The figure is a staircase with each step representing the defense potential at each of his seven targets for the specified offensive attack on each target.

If the defense has D_T interceptors, the use of a function like that of Figure H-1 will indicate to the defense how to allocate his interceptors. Since the offense can also look at his strategy from the defense point of view, if he knows D_T, he can determine if better strategies exist for himself.

In the case of Figure H-1, there likely is a better set of strategies. For example, if the offense took one attacker from strategy 7 and added it to strategy 1, he would be better off as long as that extra attacker in strategy 1 did not reduce VSP_1 below VSP_3. The logic is as follows. The attacker gives up VSP_7 units value destroyed when he switches to strategy 1, but if the new VSP_1 is still greater than the value of the defense's last interceptor, VSP_3, the defense will remove a defender from strategy 3 and add it to the new strategy 1. The effect is a net gain to the offense of $VSP_3 - VSP_7$.

If the offensive strategies all present an essentially constant return to the defender for his D_T interceptors, the offense has one of the necessary conditions for an optimum set of strategies. In such a case, the offense should carefully analyze his options because he might gain nothing by switching attackers from one target to another.

The basic problem is to develop a method which will lead the offense to such a special set of strategies. The solution method in this program is as follows.

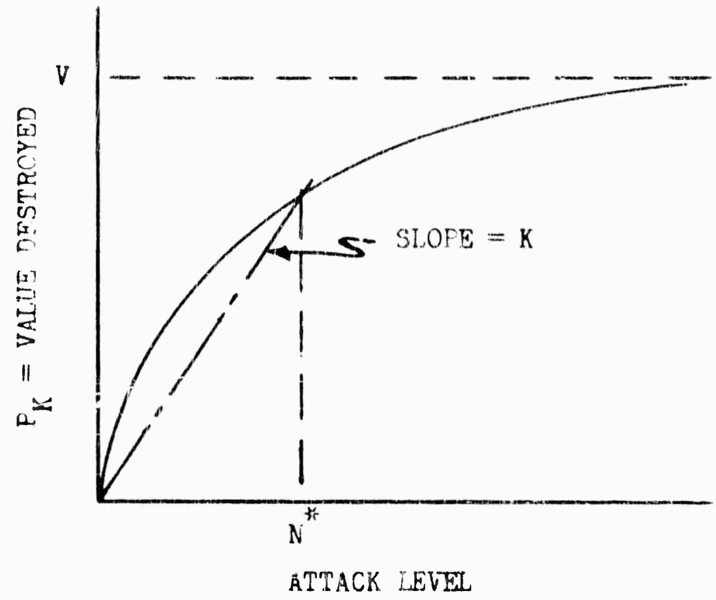
As described in Chapter IV - Section A, this program uses a Lagrangian process to optimally allocate weapons to targets. In this process, the bulk of the offensive problem is centered around the determination of an optimal Lagrangian strategy, target by target. In brief, given a current set of weapon lambdas, the best strategy is to find that set of integer N^*_{ij} such that the following function is minimized.

$$H = V_j \prod_{i=1}^{I} S_{ij}^{N_{ij}} + \sum_{i=1}^{I} \lambda_i N_{ij} \qquad \text{(H-1)}$$

The integer constraint is implied since it enters by virtue of the fact that integer N_{ij} are the only ones allowed.

In this preferential defense problem, it is possible to develop optimal offense strategies through the use of another implied constraint in the above process. Since the offense desires to develop strategies which meet the special constant return condition, he can consider only those integer N_{ij} which return the defense a pay-off $\leq$ a constant value. Then, he will develop a set of strategies which maximize value destroyed under the condition that no strategy shall allow a defense pay-off over some specified amount.

Due to the convex nature of the typical target damage function, there will be a minimum level of attack allowed if this maximum defense pay-off condition is inserted. Consider the following typical damage function at some target.



In the case of perfect attackers and defenders, the average defense pay-off for any attack level is simply P_K/N. Since the P_K function is convex, the average return per defender will exceed K if $N < N^*$ and it will be less than K if $N > N^*$. Thus, if the offense wants strategies of defense return $\leq K$, he should allow only strategies with $N \geq N^*$.

But, which amount of pay-off per interceptor shall the offense allow? The answer is that he should find one such that the defense return pay-off function looks basically like that represented in Figure H-2.

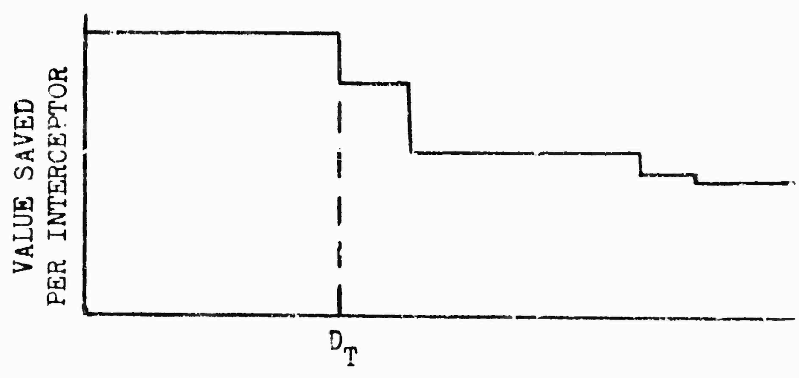


FIGURE H-2 IDEAL OFFENSE CONDITION

In this case, the offense has constrained himself only to the degree that the defense has approximately a constant return for his D_T interceptors.

It is possible for the offense to constrain himself too little, or too much. For example, if the offense solves a large number of offensive strategy sets for various constrained pay-off levels, the situation of Figure H-3 will exist.

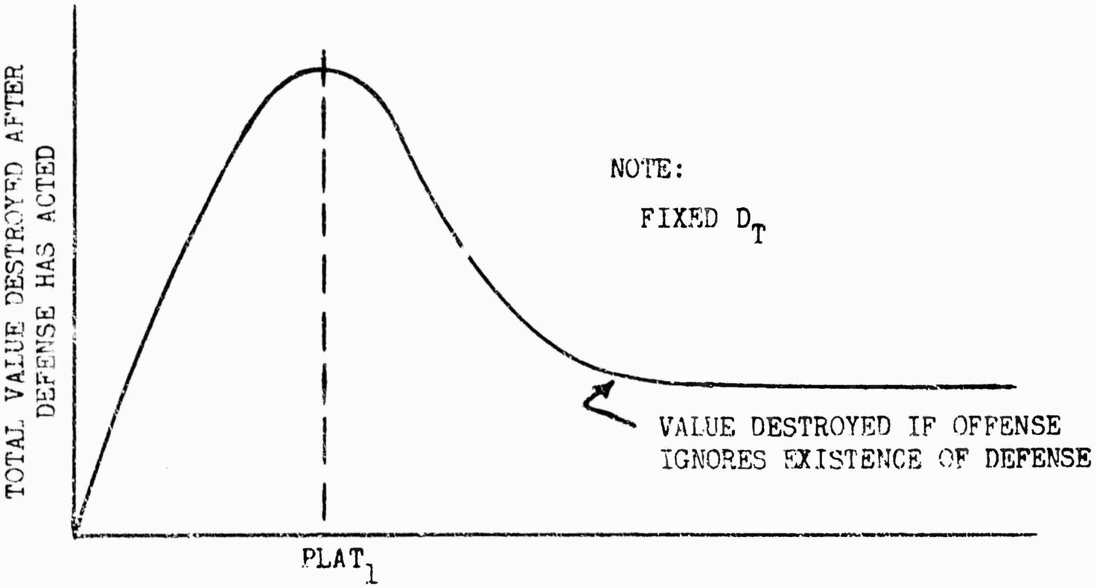
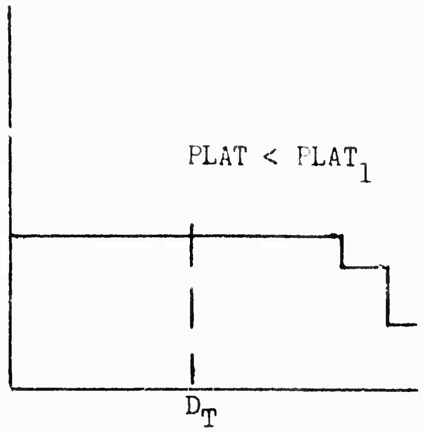
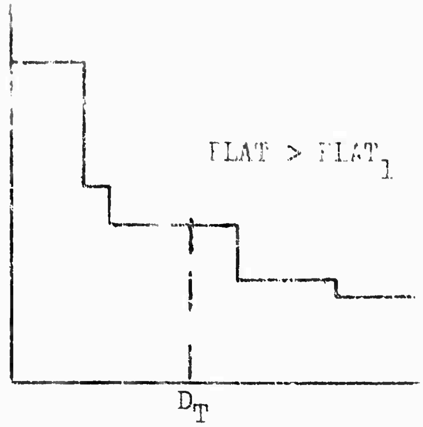


FIGURE H-3 EFFECT OF THE OFFENSE SELF-CONSTRAINT ON DEFENSE PAY-OFF

There is some constraint level, $PLAT_1$, which causes strategies to occur which result in a defense situation like that of Figure H-2. Other levels lead to defense options like those below.





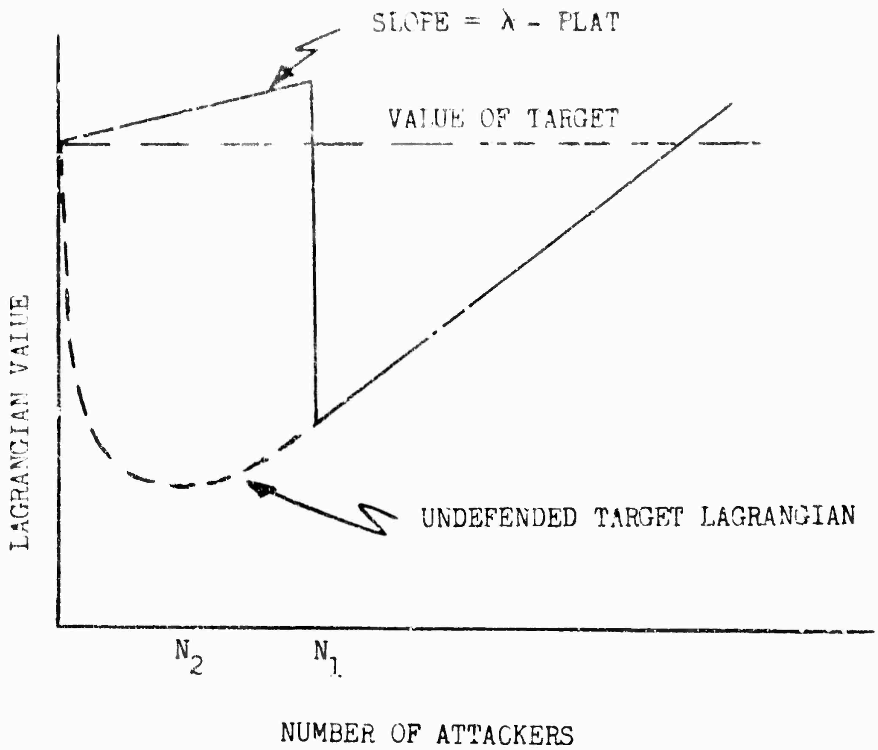
Either of these cases results in less final pay-off to the attacker.

The ideal of Figure H-2 is impossible to achieve in all cases because of integer effects. It is generally true that the offense can only achieve a set of strategies which have an approximately constant return to the defense. How this impacts on the process will be discussed further into the section.

This PLAT concept can be viewed simply as an artificial device with no interpretation of the final $PLAT_1$. In such a case, simple numerical search of the PLAT variable in order to develop $PLAT_1$ would be the strongest possibility.

There is an alternate point of view, however, which interprets the offense constraint level of PLAT to be a defense Lagrange multiplier. In this interpretation, PLAT takes on a little more significance, and utility.

One impact of this interpretation is in visualization of the Lagrangian minimizing function. If the defense is assumed to have a marginal utility of PLAT, the Lagrangian function for a specific target would diagram as follows.



There are two distinct regions in this figure. Out to an attack level of N_1, the average return to each defender exceeds PLAT so he will defend the target. Assuming no leakage, the Lagrangian function increases at a rate of λ - PLAT. This represents the delta pay-off to the defender since the attacker used one weapon but did not achieve any damage. After an attack level of N_1, the return is no longer large enough to the defense to defend the target so the defense drops completely out of the picture. The Lagrangian function then coincides with the no-defense function.

If the target truly was undefended, the optimal attack level would be at N_2. But, the presence of the preferential defense causes an attack level of N_1 in order to minimize the Lagrangian.

Continuing this idea of PLAT being a defense multiplier, after the offense optimizes his attack for the given PLAT, it will be an optimal strategy for the offense. Like the use of weapon lambdas, this PLAT corresponds to some level of defense unknown until the total offense attack is set. If the total defense level, which is counted by finding all strategies leading to a defense pay-off ≥ PLAT, is not the amount desired, PLAT must be modified.

With either interpretation, there must be developed a rapid process for converging upon the optimal level for PLAT if fixed defense resources must be analyzed. The computational success of the whole concept depends upon that process.

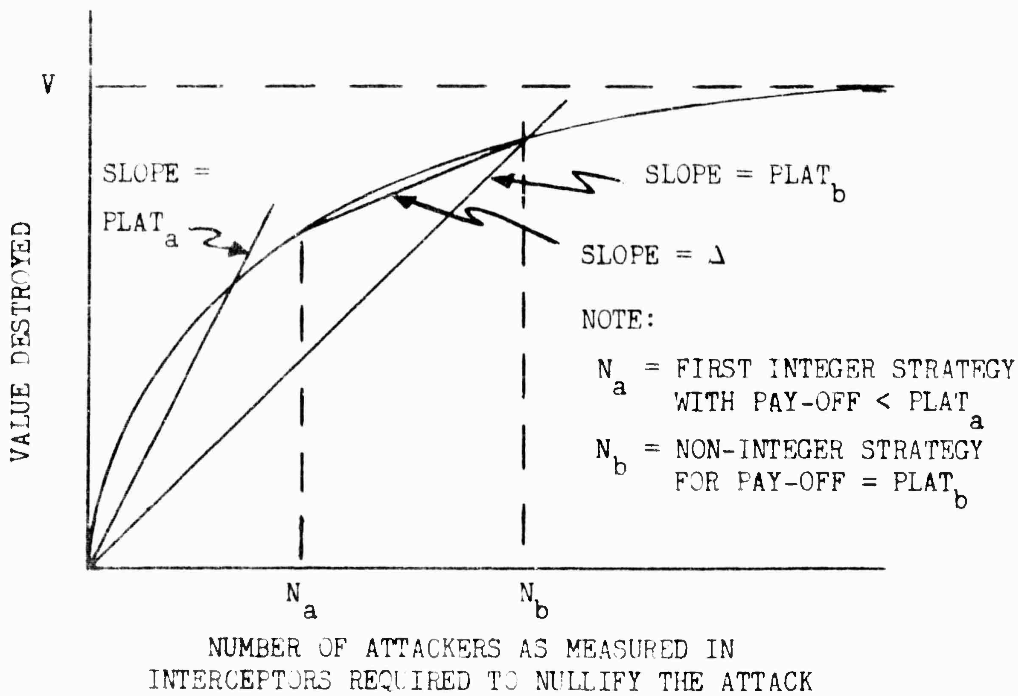
a. The PLAT Convergence Process

The first attempt at a method for determining an optimum PLAT for the offense involved usage of a Fibonacci search process. Since the offensive pay-off as a function of PLAT (Figure H-3) is basically a unimodal function, such a process should work. However, there are considerable small local optima, caused by integer effects, in the function of Figure H-3. This means that the search process can be misdirected and the approach was dropped.

The alternative of simply using a numerical stepping procedure across a selected span for PLAT was felt to be too time-consuming. Each choice of PLAT involves a new offensive optimization so one must avoid large numbers of PLAT iterations.

The current method is acceptable in running time and there is a good chance that it can be improved even further. Basically, it uses a maximum amount of information in predicting a new estimate for $PLAT_1$ given the offensive solution for some current, non-optimum PLAT. This information revolves around the set of necessary conditions for an offense optimum.

To aid in understanding of the process, consider the following figure.



Assume that a current PLAT = PLAT$_a$ was used to allocate weapons to this target. Without knowing the precise shape of the function between N_a and N_b, an estimate for the non-integer number of attackers to achieve a PLAT = PLAT$_b$ is as follows:

$$N_b \approx N_a \frac{(PLAT - \Delta)}{(PLAT_b - \Delta)} \qquad \text{(H-2)}$$

Where:

N_a = No. of attackers to achieve defense pay-off = PLAT ≤ PLAT$_a$

N_b = Non-integer attackers required for PLAT$_b$

Δ = Estimated average slope of the damage function between N_a and N_b

PLAT = Average pay-off per defender when the attack is exactly at N_a

$PLAT_b$ = Next desired defense pay-off

It is important to note here that in all discussions in this section the number of attackers is measured in units of intercepts required to defend against the attack. For example, if the attackers and interceptors are perfect but the attacker carries decoys, the attack would be measured in intercepts required to nullify the attack. This would be equal to (1 + d) times the number of attackers, where d = number of perfect decoy   RV.

If a new arbitrary $PLAT_b$ is chosen, all strategies which are returning a defense pay-off > $PLAT_b$ must have additional attackers. An estimate of the number of attacking objects required can be obtained by use of equation (H-2). The total becomes

$$D' \approx \sum_{m=1}^{M} \frac{D_m \left(PLAT_m - \Delta_m\right)}{\left(PLAT_b - \Delta_m\right)} \qquad \text{(H-3)}$$

Where:

M = last ordered strategy with defense pay-off > $PLAT_b$

D_m = current defense level required to nullify strategy m

m = subscript to identify a strategy as ordered by the defense according to his pay-off per interceptor.

$FLAT_b$ = new FLAT

$FLAT_m$ = actual defense pay-off at strategy m

But, D_m ($FLAT_m$) also represents the extra value saved by defending against strategy m. Thus equation (H-3) can be rewritten to yield:

$$D' \approx \sum_{m=1}^{M} \frac{VS_m - \Delta_m D_m}{FLAT_b - \Delta_m} \qquad \text{(H-4)}$$

Where:

VS_m = value saved by the defense if strategy m is defended against.

In order to use this equation, some assumptions must now be made about Δ_m. This parameter obviously varies from target to tar et but some bounds on its value can be obtained.

Before going into the usage of the above relationship, it is necessary to emphasize the point that it was based on an estimate of the <u>non-integer</u> number of attackers to attain a desired defense pay-off. This means that, at best, use of it should be considered as a guide to the selection of new FLAT levels.

If the current FLAT is a long way from the optimum, the non-integer estimate is adequate. As one approaches the optimum, integer effects begin to dominate and an alternate method must be used.

Several prospects for usage of the relationship exist. First, we know that $\Delta \geq o$ since the value destroyed on a target is asymptotically approaching the value of the target. Thus, we can say:

$$D' \geq \sum_{m=1}^{M} \frac{VS_m}{PLAT_b} \qquad \text{(H-5)}$$

Use of this equation leads to a useful lower bound estimate on $PLAT_b$ as follows:

Assume that the current $PLAT_a > PLAT_1$. This means that $PLAT_b < PLAT_a$ and, as a consequence, that there is some minimum number of strategies M^* such that

$$D' = D_T \text{ and } PLAT_{M^*+1} \leq PLAT_b' = \frac{\sum_{m=1}^{M^*} VS_m}{D_T} \qquad \text{(H-6)}$$

$PLAT_b'$ = lower bound estimate of PLAT.

In other words, if the first M^* strategies all are to yield a payoff per interceptor of $PLAT_{M^*+1}$, the estimated total interceptors required would equal D_T. Thus, $PLAT_b \trianglerighteq PLAT_{M^*+1}$ and we have a candidate for a new PLAT. Also, this identifies which strategies likely would be the ones for the offense to act upon.

On the other hand, an estimate for the next defense payoff to allow $PLAT_b$ could be the same as the average return per interceptor in the last set of offensive strategies. Use of that average value cannot hurt the defense but possibly could help the offense. Thus, we can estimate that

$$PLAT_b = \frac{\text{(VALUE SAVED BY DEFENSE)}}{\text{(TOTAL NUMBER OF INTERCEPTORS)}}$$

$PLAT_b$ = upper bound estimate of PLAT.

One interesting side light on the integer effect is that use of this PLAT can hurt the defense, and help the offense, when the offense goes to the integer strategy which brings pay-offs just less than this estimate. In other circumstances, the estimate can lead to an offense mistake because of the integer effect.

One additional estimate for $PLAT_b$ can be obtained by assuming that D_m is approximated by the lowest defense pay-off offered by the offense in the last set of strategies. (VSP_7 in Figure H-1.) An indicator that this is true comes about because the offense is using a Lagrangian process but he didn't add another attacker to the strategies chosen by the defense. This means that his pay-off there must have been lower than the pay-off he could get by creating a new strategy.

Use of that estimate in equation (H-4) results in

$$\overline{PLAT_b} \approx \sum_{m=1}^{M^{**}} \frac{VS_m - PLAT' \cdot D_m}{D_T} + PLAT' \qquad \text{(H-7)}$$

Where:

$PLAT'$ = minimum defense pay-off per defender in last strategy set

M^{**} = smallest number of ordered strategies that $PLAT_{M^{**}+1} < PLAT_b$

To demonstrate the nature of these estimates, consider Figure H-1. In that example, there are seven strategies for the defense to operate on. In the notation just used, the three estimates just discussed are as follows:

Lower Estimate: (Using H-6)

$$M^{*} = 2 \qquad PLAT_b' = \frac{3.3 + 5.0}{10} = .83$$

$$PLAT_{M^{*}+1} = PLAT_3 = .80$$

Estimate is $PLAT_b \geq .83$

Upper Estimate:

$$PLAT_b \leq \frac{3.3 + 5.0 + 1.6}{10} = .99$$

Mid Estimate: (Using H-7)

$$PLAT' = .2 \qquad M^{**} = 2$$

$$\overline{PLAT}_b = \frac{3.3 + 5.0 - .6 - 1.0}{10} + .2$$

$$= .87$$

Several different ways of using these estimates is possible. One effective way is to use the upper estimate but always to choose the integer strategy that results in a defense pay-off just under the estimate. This process seems to converge in the most stable fashion. The next best, sometimes faster, method is to use the mid estimate and to take the integer strategy which comes the closest to it. At the present time, additional experimentation is being conducted to determine the best method overall.

As one approaches the final $FLAT_1$, all of these estimates tend to converge and the precise way one handles the integer effects seems to be the final factor in achieving an optimum. This ingredient enters in as follows.

As the optimal ($FLAT_1$) is approached from a level of FLAT which is too high, the offense develops strategies which present the defense with nearly a constant pay-off on each of several strategies. When this occurs, all of the above estimates look essentially alike, but with small differences.

The small differences occur because of the integer effect. First, with a mixed weapon force, each strategy can involve different weapons to integer levels. Secondly, the target groups are of different types and the net effect is the staircase function discussed previously. In this case, the differences between strategies can be small but they do exist.

At this point, there is a question as to the best manner to handle the integer effect. For example, say a new FLAT is obtained by use of the previously described upper estimate. Since strategies must involve integer weapons, this exact FLAT cannot be attained on each strategy. If one always chooses an integer such that the pay-off is just less than this FLAT, there is a slight, but perceptible lowering of the average defense pay-off. As a result, the next estimate for FLAT will be slightly lower than the previous one, etc. In this way, the process can slowly lead to lower and lower FLAT's and the optimal PLAT will be passed.

As the FLAT gets progressively smaller, another effect begins to appear. Namely, there are more strategies having a constant defense pay-off than the defense can defend against. For example, the defense might have 100 interceptors, but there are enough strategies for him to place 300 interceptors, all at a constant pay-off.

This raises an interesting question. Should the offense ever deliberately lower the defense pay-off even when he must lower it on more strategies than the defense can defend against? In other words, is it a sufficient condition for offense optimality when he produces a set of constant defense pay-offs or is it only a necessary condition?

The answer to the above set of questions is that the constant pay-off objective is only a necessary condition. This is obvious from the fact that the offense can impose a very small PLAT on himself with the net effect that he overresponds to the defense and provides the defense with a constant return per interceptor on far more targets than the defense can defend. In doing so, he allows many targets to go unattacked and penalizes himself in value destroyed.

Quite obviously, there is some connection between the defense level and the proper level of PLAT. One necessary relationship between the two comes about as follows. At any given PLAT, there is a set of strategies which were impacted upon by the FLAT in that the ideal no defense Lagrangian strategy was not allowed. If all of those strategies were to be modified to fit another FLAT (either larger or lower), we can use equation (H-2) to estimate the new total number of attackers in those strategies. This results in

$$A_E' \approx A_E \frac{(FLAT - \Delta)}{(FLAT_b - \Delta)} \qquad (H-8)$$

Where:

A_E = total number of attackers currently included in strategies controlled by FLAT.

FLAT = average defense pay-off in the above strategies

$FLAT_b$ = next defense pay-off

Δ = average slope in the damage functions involved in the above strategies.

A_E' = new total attackers if FLAT is changed to $FLAT_B$

In the case where the optimal FLAT has been passed, the above A_E will be $\geq D_T$, the total defenders. If the attacker has some place else to use his attackers, he would like to impose a new FLAT on himself so he can balance the benefit to himself and not overly respond to the defense.

If he allows FLAT to go to $FLAT_b$ and the A_E' still is $> D_T$, he can estimate the impact on the defense pay-off to be

$$\Delta DF \approx D_T \; (FLAT_b - FLAT) \qquad (H-9)$$

Where:

ΔDF = Total defense value change for a given FLAT change when the defense still gets a constant return per interceptor.

Meanwhile, the defense now has either released some attackers for use elsewhere (if $FLAT_b > FLAT$), or brought in more attackers (if $FLAT_b < PLAT$). His delta pay-off then must be:

$$\Delta OP \approx (A_E - A_E^{'}) \; P_o \qquad \text{(H-10)}$$

Where:

ΔOP = offense delta pay-off

P_o = offense estimate of his marginal utility per attacker

The offense would like to find such a place that the $\Delta DP = \Delta OP$. Then he estimates that no further adjustment in FLAT should be necessary. Equating the two equations results in

$$PLAT_b \approx \frac{A_E \; P_o}{D_T} + \Delta \qquad \text{(H-11)}$$

One can use this relationship to make a check at any time to determine if the optimal PLAT has been passed. At the completion of any step in the PLAT convergence process the offense can look at his strategies to estimate the parameters in the relationship and decide which direction to go on PLAT.

Experience has demonstrated that use of the relationships in this section do provide a very good method for isolating a near-optimum PLAT. The upper estimate operates very well in cutting PLAT down if it is too large, while that of equation (H-11) indicates proper increases in PLAT if it is too small.

However, in the region of an optimal PLAT, there is no current way to converge on the final precise value other than slowly stepping FLAT across a small span and simply watching the total offense net pay-off. The best approach is to start with a high FLAT, use the high estimate to continually cut it down and as the optimal FLAT is approached, the estimate changes automatically slow down appropriately. Meanwhile equation (H-11) is used to keep a check on the process to make sure the optimal FLAT has not been passed.

One cannot just stop at a decrease in offense pay-off because of the local optima caused by the integer effect.

Using this process obviously does not guarantee precisely optimal offense strategies but the accuracy has been acceptable. When all of these estimate procedures point toward a certain FLAT level, it is felt that adequate results do occur. No precise error computations are possible but one confidence generating factor is that when offense and defense parameters are varied, the offense pay-off values plot very smoothly. If poor local optima were being obtained, this situation would not exist.

One option not considered to date is that of relaxing the integer weapon constraint just to help convergence to a correct starting region for FLAT. The reason this has not been done is that in the computationally similar weapon allocation process relaxation of the integer constraint does not materially aid in the final weapon convergence problem. Thus, one would suspect that the final convergence, which must allow for integers, would still be a difficult computational problem.

Going back to the Lagrangian view of FLAT, it is obvious that for a given change in FLAT there are discontinuties in the defense versus FLAT function. That is, if one plots FLAT versus the defenses required to nullify all strategies where the pay-off essentially equals FLAT, there are instances where an ϵ change in FLAT results in a large change in the equivalent defense level. If the defense level of interest is in this region of discontinutiy, a special procedure must be developed to delve down into the discontinuity. (Reference (1) contains a discussion of this "gap" problem.) If a specific defense level is to be exactly optimized, a special procedure for delving deeper into the strategies is sometimes necessary in the case where the PLAT variable does not have a unique correlation with a given defense level. Such a procedure has not yet been deemed necessary.

2. Impact of Imperfect Defenses

It is desirable to be able to analyze the impact of imperfect defenses so the preceding concepts were analyzed to develop such a capability. As in the other ABM routines, this capability should include the traditional parameters of:

P_A = Probability that the defense does not have the opportunity to fire at a given object.

P_I = Probability that the defense does intercept an object with one shot.

P_D = Probability that a decoy is discriminated.

EXO = Number of area decoys per RV.

ACODE = Fixed number of interceptors to fire at each undiscriminated, but acquired object.

R = Weapon reliability.

Within these parameters there are several approaches that could be taken. For reasons of simplicity and compatibility, the one selected here is that of an expected value model. This model computes the probability that an average RV will arrive at and penetrate the defense and the expected number of interceptors launched for each RV fired, if the defense defends against a given strategy.

Built into the model is the assumption that the defense leaves constant the number of interceptors fired at each object. Also, there is the assumption that the defense fires at every acquired, undiscriminated object at each target he decides to defend against.

In the expected value sense, the probability that any given RV arrives at the target and penetrates the defense is the following:

$$R^* = R \left[(1 - P_A) + P_A (1 - P_I)^{AC\emptyset DE} \right] \qquad \text{(H-12)}$$

This simply computes all the appropriate event probabilities to identify reliable penetrators.

Meanwhile, the defense expects to fire an average of RINT interceptors at each launched RV, where:

$$RINT = AC\emptyset DE \cdot P_A \cdot R \left[1 + (1 - P_D) \cdot EXO \right] \qquad \text{(H-13)}$$

This is simply based on the expected number of reliable, but undiscriminated objects arriving at the area defense.

Within the above relationships, one can visualize the impact of an imperfect defense. Consider the following figure.

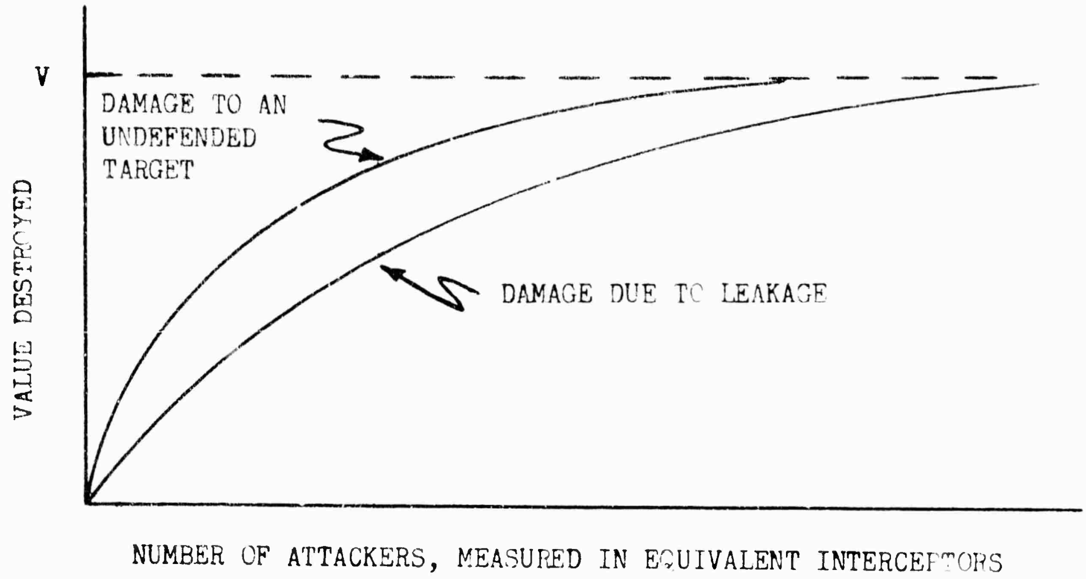


FIGURE H-4 DAMAGE FUNCTION FOR AN UNDEFENDED OR DEFENDED TARGET

Two damage curves are demonstrated. They represent damage as a function of attackers for the target in a defended and undefended mode. The damage to the target if it is defended is due to the leakage because of the imperfect defense. It is computed by assuming that the effective weapon reliability is modified from R to R^* if the defense decides to defend that specific target.

The pay-off to the defense for any specific attack level must be the difference between the two functions at that attack level. Such a delta function appears as follows.

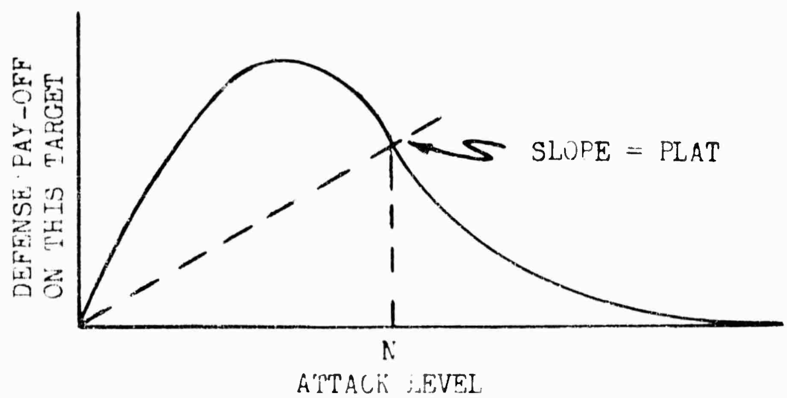


FIGURE H-5 DEFENSE PAY-OFF VERSUS ATTACK LEVEL

For any specific attack level, say N, the defense can compute his average pay-off per defender. In the above figure, this is represented by PLAT.

It is apparent that this representation of an imperfect defense meshes very well with the perfect defense model. Both finally come down to each strategy being measured in terms of the average defense pay-off if the strategy is defended against and the expected number of defenders required in the defense against that strategy.

In accordance with this similarity, the offense here uses the same mechanism to develop an optimal set of strategies. He estimates the defense parameters plus the number of defenders and proceeds to compute a PLAT which will provide the defense with a constant return per interceptor.

In this case, the offense must use the delta function to determine which attack level is necessary to limit the defense pay-off. Once a PLAT is chosen the offense uses the same process to optimize his attack for that PLAT.

In development of a process for converging to an optimal PLAT, several facts are worthy of discussion.

First, note that the estimating procedure previously described generally involved estimating the change of attackers for a desired change in PLAT. The various estimates followed from different estimates of the slope of the defense pay-off versus attack level function.

In that case, the defense function always had a positive slope. However, in this case, the slope is positive or negative depending upon the attack region one is in and the degree of leakage. Since those estimates were based on only a positive slope, they will all be high estimates if a negative slope is in existence.

If the initial estimate is high, the previously described stepping procedure will operate. It possibly will converge slowly, but it will function.

Experience with the s ame stepping procedure for PLAT in perfect and imperfect defenses has shown that one procedure is adequate. However, at the present time, additional analyses are being conducted to determine if an effective procedure can be obtained for speeding up the process in the leaky defense case. Until, and if one is developed, the described procedure will be used.

I. PRE-COMMITTED AREA DEFENSES

1. General

If the defense chooses to defend a target with a certain level of defense a fraction of the time and is allowed the option of changing this level without exposing the change to the offense, the resultant is a distribution of defensive possibilities facing the offense. If it is also assumed that for each defensive level (D) there corresponds an optimum attack strategy (A), the distribution in defense level [P(D)] forces a distribution in attack strategy [P(A)] to produce the maximum expected damage from the entire offensive attack. This defensive doctrine and the associated defense/offense distributions which minimize the maximum expected damage achievable by the offense is called Pre-commit.

Initial work done by R. J. Galiano (Reference 6) on perfect attackers and perfect defenders indicated Pre-commit is roughly mid-way between preferential and subtractive defenses. His definitions of equivalence with who moves last may enhance conceptual understanding:

TABLE I-1 RELATION BETWEEN DEFENSE DOCTRINES

DEFENSE DOCTRINE	LAST REALLOCATION (WHO MOVES LAST)	ATTACK (A) CONFRONTED BY (AT A TARGET)
SUBTRACTIVE (TERM.)	OFFENSE	D DEFENDERS
PRE-COMMIT	SIMULTANEOUS	D DEFENDERS WITH PROBABILITY P(D) $0 \leq D < \infty$
PREFERENTIAL	DEFENSE	A DEFENDERS OR ZERO DEFENDERS

2. Approaches to the Problem

Neither a comprehensive model nor a general approximation is currently programmed as an Arsenal Exchange Model (AEM) capability. The current capability is limited to the offense dominant condition. The work presented here is to be viewed as background support for anticipated expansion of the capability to all situations. Paragraph 3 presents the status and direction of current work in this area.

Two approaches have been explored: a) min-max solution of a constrained game and b) analytic evaluation of uniform defense distributions. In addition, some equivalence with optimum terminal and imperfect preferential defense has been noted.

a. The Game Theoretic Approach

Zero-Sum, Two-Person Games

The uncertainty in resource allocation by offense and defense and the defense objective of minimizing the maximum offensive pay-off is explicitly stated mathematically as a zero-sum, two-person game. The following discussion on the theory of zero-sum games (provided by A. H. Silver, Martin Marietta Corporation) is a useful background for understanding how pre-commit defenses can be viewed as such a game.

Zero-Sum, Two-Person Games are by definition games with only two participants (persons, teams, firms or nations) in which one participant wins what the other loses. A fundamental concept in game theory is that of a "strategy." A "strategy" is a complete enumeration of all actions each competitor will adopt for every contingency that might arise, whether the contingency be one of change or one created by the moves of the opposing players. However, a strategy should not be interpreted too naively. The reason a player does not change a strategy during the course of a game is not that the strategy has committed him to a predetermined sequence of moves he must make regardless of what his opponent does but rather that it enables him to make a move in any circumstance that may arise. At this point, our attention is restricted to two types of strategy. A "pure" strategy is a decision, in advance of all plays, always to choose a particular course of action. A "mixed" strategy is a rule to choose a course of action, in advance of all plays, in accordance with some particular probability distribution.

In attempting to obtain an optimal strategy for each competitor, the criterion of optimality used will be the "max-min" criterion (Reference 7). Roughly, this criterion may be visualized as follows. A player lists each of his potential (mixed or pure) strategies together with the worst outcome, from his point of view, that can result from combinations of his competitors' potential strategies. He chooses the strategy that corresponds to the best of these worst possible outcomes.

By definition, the game has a solution if the players maintain their initial strategies as giving them the maximum expected gain after repeated plays. The value of a solvable game to a player is his expected gain in one play of the game, with both players utilizing their stable optimal strategies.

A game is defined to be a triplet $\{X, Y, K\}$, where X denotes the space of strategies for Player A, Y signifies the space of strategies of Player B and K is a real valued function of X and Y. The following assumptions are made:

(a) X is a convex, closed, bounded set in Euclidean n-space.

(b) Y is a convex, closed, bounded set in Euclidean m-space.

(c) The pay-off, K, is a convex linear function of each variable separately.

Thus, the pay-off for arbitrary mixed strategies X and Y is given by the expression:

$$K(X, Y) = \sum_{j=1}^{m} \sum_{i=1}^{n} x_i a_{ij} y_j \qquad \text{(I-1)}$$

Obviously, certain special strategies, consisting of vertex points of X, are denoted by

$$\alpha_i = (0, \ldots 0, 1, 0, \ldots 0) \; (i = 1, \ldots n) \qquad \text{(I-2)}$$

where 1 occurs in the i^{th} component. These are Player A's pure strategies. Similarly,

$$\beta_i = (0, \ldots 0, 1, 0, \ldots 0) \; (j = 1, \ldots m) \qquad \text{(I-3)}$$

are Player B's pure strategies. Since

$$K(\alpha_i, \beta_j) = a_{ij} = A(\alpha_i, \beta_j) \qquad \text{(I-4)}$$

then the i, j element of the matrix array "A" expresses the gain to Player A when Player A utilizes the pure strategy α_i and Player B employs the pure strategy β_j.

Suppose that Player B is compelled to announce to Player A what strategy he is going to use and that he states y_o. Then, Player A, seeking to maximize his pay-off, will obviously choose his strategy as x_o so that

$$K(x_o, y_o) = \max_x K(x, y_o) \tag{I-5}$$

Then, the best thing that Player B can do under these circumstances would be to announce y_o such that

$$\max_x K(x, y_o) = \min_y \max_x K(x, y) = \nu_1 \tag{I-6}$$

where ν_1 can be interpreted as the most that Player A can achieve if Player B employs strategy y_o.

Suppose now that Player A has to announce his strategy x_o. Since Player B is sure to choose y_o such that

$$K(x_o, y_o) = \min_y K(x_o, y) \tag{I-7}$$

Player A can best protect himself by choosing x_o such that

$$\min_y K(x_o, y) = \max_x \min_y K(x, y) = \nu_2 \tag{I-8}$$

where ν_2 can be interpreted as the most Player A can guarantee himself independent of Player B's choice of strategy.

Von Neumann has established that for a matrix game $\nu_1 = \nu_2 = \nu$ (see Reference 7). The following proof develops the simple criterion to determine when $\nu_1 = \nu_2$ and points up the meaning of "optimal" strategies.

If there exist $x_o \in X$, $y_o \in Y$ and a real number ν such that

$$K(x_o, y) \geq \nu \quad , \text{ for all } y \in Y$$

and

$$K(x, y_o) \leq \nu \quad , \text{ for all } x \in X$$

then $\nu_1 = \min\limits_y \max\limits_x K(x, y) = \nu = \max\limits_x \min\limits_y K(x, y) = \nu_2$ (I-9)

and conversely. Proof:

(a) Since $K(x_o, y) \geq \nu$ for all y, it follows that

$$\min_y K(x_o, y) \geq \nu \text{ and } \max_x \min_y K(x, y) \geq \nu \ .$$

Similarly, $\max\limits_x K(x, y_o) \leq \nu$ and $\min\limits_y \max\limits_x K(x, y) \leq \nu$.

Thus, $\max\limits_x \min\limits_y K(x, y) \geq \min\limits_y \max\limits_x K(x, y)$ (I-10)

But $\min\limits_y \max\limits_x K(x, y) \geq \max\limits_x \min\limits_y K(x, y)$

Therefore, $\min\limits_y \max\limits_x K(x, y) = \max\limits_x \min\limits_y K(x, y)$. (I-11)

(b) Conversely, choose y_o such that

$$\max_x K(x, y_o) = \min_y \max_x K(x, y) = \nu \qquad \text{(I-12)}$$

and choose x_o such that

$$\min_y K(x_o, y) = \max_x \min_y K(x, y) = \nu$$

Then $K(x, y_o) \leq \nu$ and $K(x_o, y) \geq \nu$. (I-13)

This completes the proof.

Thus, the "optimal" strategy indicates that Player A can guarantee himself the value $\nu_2 = \nu$, and by judicious play, Player B can prevent Player A from achieving more than $\nu_1 = \nu_2$. Unless Player A has additional information about Player B's mode of behavior, Player A should play so as to achieve ν . If he departs from the course of action that assures him the value ν , his ultimate yield might be less than ν . Thus, it is in this sense that the common value ν is called the "value" of the game to Player A. Conversely $-\nu$ is the value to Player B.

Formulation and Play of the Pre-commit Game

Lagrangian multipliers were used to resolve the globular problem of multiple target and weapon types on a cellular basis (see Reference 8). Use of this technique allows the following strategy formulation.

$$H(i,j) = VD(i,j) - \lambda i + \mu j \qquad (I-14)$$

where: $H(i,j)$ = The Lagrangian value if i weapons are allocated against a target defended by having j defenders.

$VD(i,j)$ = The resultant value destroyed.

λ = Lagrangian cost per weapon.

μ = Lagrangian cost per defender.

These pay-offs are structured in matrix form as follows:

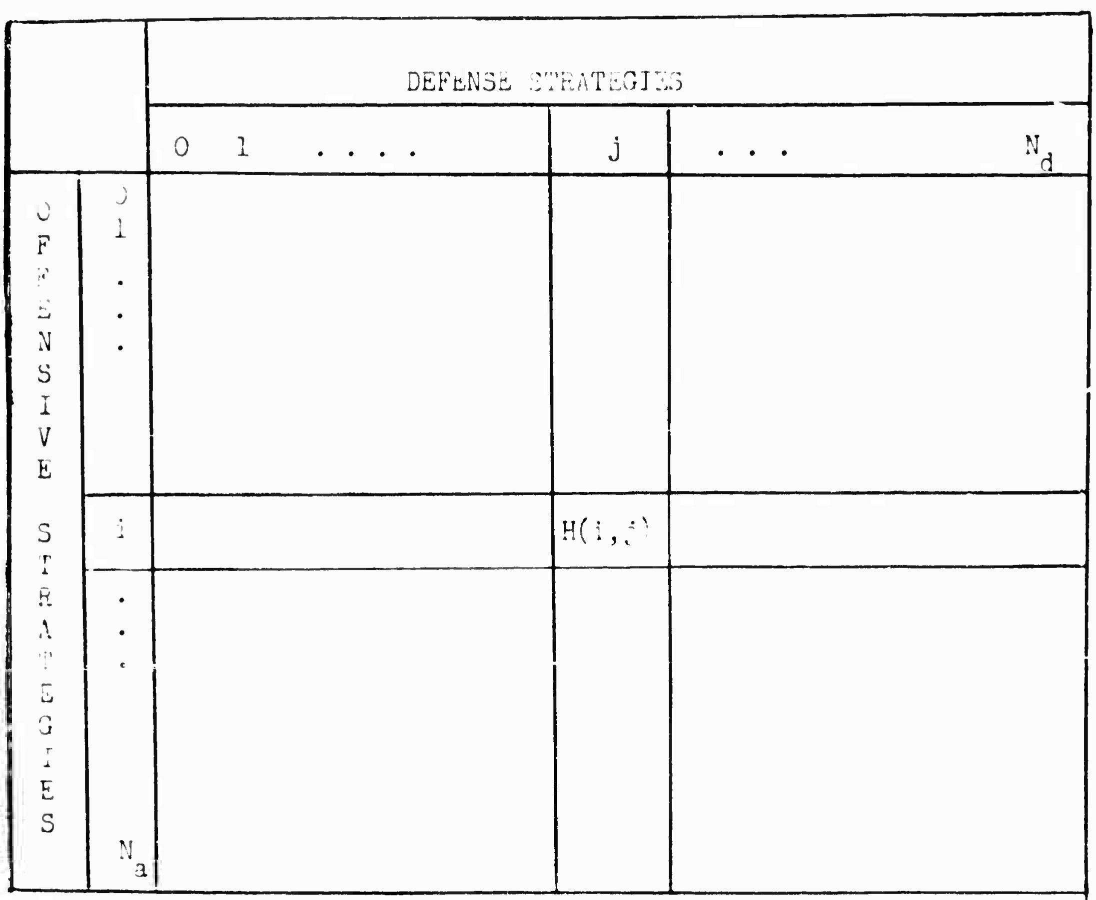


FIGURE I-1 GENERAL GAME MATRIX

Where N_a and N_d denotes the limits of the game kernel, i.e., any row R representing an attack strategy greater than N_a is dominated (each term less than) by the first row in the kernel, and any column C representing a defensive strategy greater than N_d is dominated (each term greater than) by the first column of the kernel.

The following example illustrates a typical game formulation and typical results.

The resultant game matrix for perfect defenders protecting a target of value one against an attack by weapons having a single shot probability of kill of .5 is presented in Figure I-2. This matrix is generated by the following Lagrangian form:

$$H(i,j) = \begin{cases} j \cdot \mu - i \cdot \lambda \ ; i \leq j \\ V(1 - P_{KSS})^{i-j} - i \cdot \lambda + j \cdot \mu \ ; \ i > j \end{cases}$$

and uses Lagrangian cost of $\lambda = .2$, $\mu = .001$.

		DEFENSIVE STRATEGIES					
		0	1	2	3	4	5
OFFENSIVE STRATEGIES	0	0	.001	.002	.003	.004	.005
	1	.30	-.199	-.198	-.197	-.196	-.195
	2	.35	.101	-.398	-.397	-.396	-.395
	3	.275	.151	-.098	-.597	-.596	-.595
	4	.1375	.026	-.048	-.297	-.796	-.795
	5	-.0312	-.0614	-.123	-.247	-.496	-.995

FIGURE I-2 SAMPLE GAME MATRIX

Note that the sixth row strategy (5 attackers) is less than the first row in every term. Therefore, it will never enter the solution. Likewise all subsequent attack strategies (> 5) will be dominated by the first row. Since the defense is perfect, the defense will never exceed the maximum attack. Hence, it is sufficient to solve the game when $N_d = N_a = 5$ (zero through four attackers and defenders).

Two methods of computing the min-max solutions of these pay-off matrices have been developed, namely, Brown's method of fictitious play and a simplex algorithm.

Brown's method (see Reference 9) is an iterative process which involves the normalization of cumulative history vectors for both players. The game may be terminated at the end of any iteration cycle and the bounding values computed. The following table presents the results of the preceding game as a function of the number of iterations:

TABLE I-2 EFFECTS OF NUMBER OF ITERATIONS ON GAME SOLUTIONS

NUMBER OF ITERATIONS	GAME VALUE BOUNDS UPPER/LOWER	GAME VALUE DELTA	MEAN ATTACK $\sum P(A) \cdot A$	MEAN DEFENSE $\sum P(D) \cdot D$
10	.025/-.05	.03	.3	.25
100	.0144/.012	.0024	.09	1.44
1000	.0126/.0119	.0008	.09	1.266
10000	.01214/.01184	.0003	.0893	1.2148
--	.012 *	--	.09 *	1.2 *

* LINEAR PROGRAM SOLUTION

The error after ten thousand iterations is less than 3 percent in this example. The running time varied roughly linearly with number of iterations but with the square of the number of strategies. As larger games were considered, this technique proved too burdensome.

The equivalence between game theory and linear programming was investigated as a means to preclude convergence problems. The algorithm presented in Danzig (see Reference 5) was adapted. The last row in the above table is the linear program solution to that game. Additional comparisons were performed to confirm identical solutions.

Usage of Results

It should be remembered that each game corresponds to one "strategy" as thought of in the Arsenal Exchange Model. Direct implementation would thus entail cellular solution for each weapon/ target combination and development of some techniques to allow proper convergence of the associated lambdas and mu. Such changes would require major reprogramming and would of necessity grossly increase running time. Therefore, much effort has been devoted to capitalizing on the insight gained from these games to allow reasonably accurate approximations that are compatible with the current model. A brief discussion of these efforts follows to establish a basis for the analytic approach.

Results of some initial investigations gave some hope of approximating the distributions of offensive and defensive strategies from known variables. However, no general hypothesis could be resolved. With increased experience, game results continued to deny the existence of such an hypothesis. The following examples (Table I-3) might better explain the sequences. Three examples are shown. The first is from a simple case, the second and third are more representative of the target values and kill functions used in the Arsenal Exchange Model.

The similarity between lambda and defense strategy frequency (and mu and attack strategy frequency) in case 1 is apparently misleading. Observe the non-constant probabilities in cases 2 and 3. Prediction of such variations seem impossible to date. Prediction of the beginning and ending of the distributions has proved equally difficult.

Analysis of mean level of effort as a function of lambda and mu is more promising in form. Figure I-3 illustrates the variation with respect to mu. Similar curves describe the variation with respect to lambda with transposed behavior of attack and defense levels. The form is easily stated as: a linear increase in attack level until mu obtains some critical level (not always at mu = lambda), then decays exponentially to the optimum Lagrangian level if there was no defense; and a constant level of defense until mu obtains some critical level (not necessarily the same mu at the attack break point), then exponentially decaying to zero.

There may be relationships that allow prediction of these components. But, great difficulty was experienced in constructing general functions to predict: 1) the slope of attack level increase in its linear region, 2) the level of defense when constant, 3) the critical values of mu which denote the change

TABLE I-3 STRATEGY DISTRIBUTIONS

CASE	1		2		3	
TARGET VALUE	1.		3.		1.4	
$P_K SS$	.5		.179		.39	
P_K TRANSLATION	0		3.1		2.23	
λ	.2		.2		.04	
μ	.15		.15		.07	
LEVEL OF EFFORT	ATTACK	DEFENSE	ATTACK	DEFENSE	ATTACK	DEFENSE
0	.4	.4	.28	.86	0	.6
1	.15	.2	.08	0	0	0
2	.15	.2	0	0	0	0
3	.3	.2	0	0	0	0
4	0	0	0	0	0	0
5			0	0	0	0
6			0	0	0	0
7			0	.002	.215	.041
8			0	.102	.061	.058
9			.64	.036	.057	.035
10			0	0	.039	.022
11					.014	.028
12					.022	.040
13					.034	.066
14					.036	.092
15					.031	.064
16					.025	.006
17					0	0
18					0	.075
19					.135	.006
20					.196	0
21					0	0

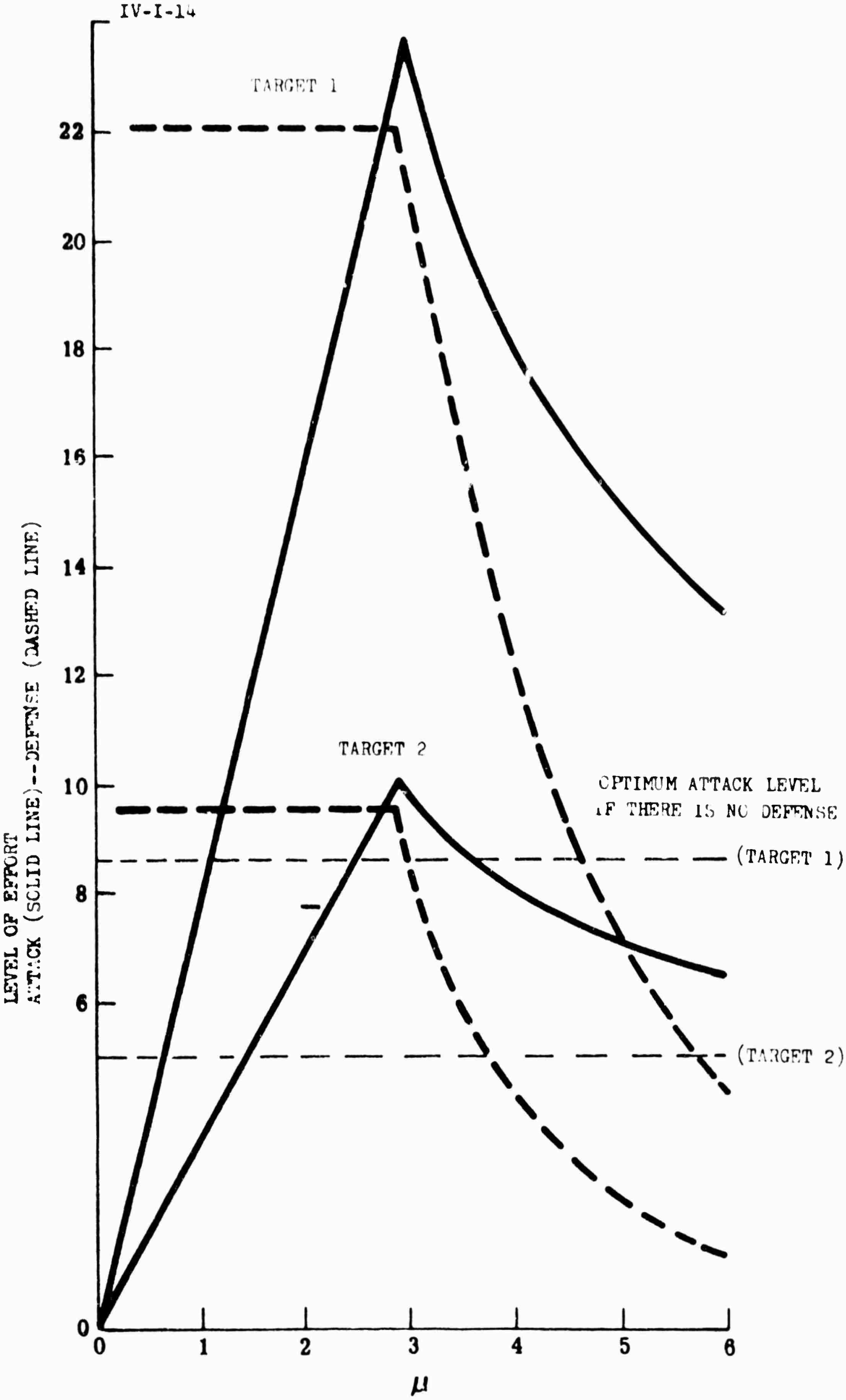


FIGURE I-3 VARIATION OF MEAN LEVEL OF EFFORT VS. μ

in functions, and 4) the exponential path leading to the asymptote. It is possible that a more rigorous attempt will be made to resolve these difficulties.

The difficulty experienced in predicting game results made direct approximation by analytic functions unsupportable. However, some results are used as supporting rationale in the analytic approach. Additionally, these games have been invaluable in assessing the applicability of the results of the analytic model as well as the approximation of pre-committed defenses by optimally deployed terminal and imperfect preferential defenses.

b. Analytic Methods

The game theoretic solution applies to a single target and weapon type where weapon and defense resource constraints are expressed as Lagrangian cost. The technique for inclusion of additional basic capabilities of the Arsenal Exchange Model (e.g., multiple weapon types, damage limitation constraints, optimum budget expenditures) is not now known. The impact of these considerations can therefore not be determined directly. Therefore, an analytic approach compatible with the cellular concept used in the Arsenal Exchange Model which approximates game solutions has been developed.

Although the optimum (min-max) solution is obtained by game theory, if frequency functions are arbitrarily chosen for both the offense and defense and the min-max of these functions can be obtained by calculus, the definition of pre-commit has not been violated. It is just that an arbitrary set of frequency functions is not an optimal pre-commit.

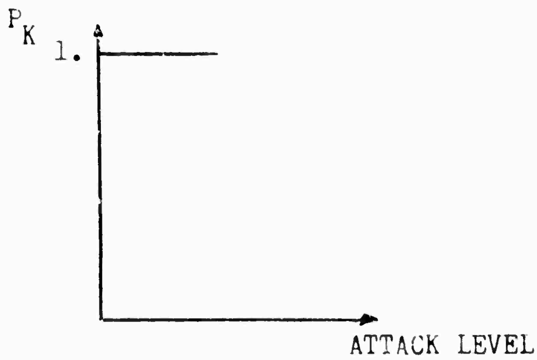
It can be demonstrated that the precise frequencies generated by the game solution are not necessarily critical. Table I-4 contains these distributions for the sample game presented in Figure I-2 and Table I-2. The variations in the defense strategy distribution are quite pronounced while the average game value is rather stable, being within ten percent after 100 iterations. It was concluded from a variety of such investigations that the game value (and therefore, value destroyed) could be approximated with reasonably arbitrary strategy distributions.

TABLE I-4 EFFECTS OF NUMBER OF ITERATIONS ON STRATEGY DISTRIBUTIONS

NUMBER OF ITERATIONS	AVERAGE GAME VALUE	DEFENSE DISTRIBUTION			
		0	1	2	3
10	.015	.1	0	.2	.7
100	.0132	.39	.01	.37	.23
1000	.0123	.293	.36	.135	.212
10000	.01187	.393	.191	.224	.192
LINEAR PROGRAM	.012	.4	.2	.2	.2

Additional substantiation that the defense frequency is stable for diverse kill functions comes from analyses of the two perfect kill functions pictured below. This analysis showed the result of uniform defense distributions between zero and twice the mean attack minus the translation, i.e., $P(D) = K,\ 0 \leq D \leq 2\bar{A}-T$.

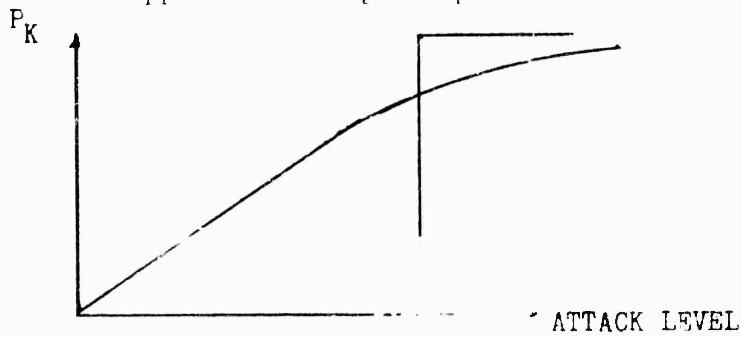
FUNCTION (1)
(NO TRANSLATION)



FUNCTION (2)
(TRANSLATION = T)

The probability of no defense, $P(D = 0)$, is expressed as $1 - \overline{D}/\overline{A}$. (If $\overline{D} > \overline{A}$, $P(D = 0) = 0$. and $P(A = 0) = 1 - \overline{A}/\overline{D}$.) These conclusions are supported by algebraic derivation in References 6 and 10.

The following overlay of a typical AEM damage function represents an approximation by the perfect kill function type (2).



The apparent lack of sensitivity to precise strategy distributions and the existence of algebraic simplicity for these diverse looking, perfect kill functions resulted in the suggestion that assumption of a uniform, zero to twice mean offensive attack level, defense distribution in the analytic model might be reasonable.

Another simplification results if the following relationship is assumed.

Define the net value destroyed (υ) as

$$\upsilon = \sum_{R_1} \sum_{R_1} P(A) \cdot P(D) \cdot VD\ (A,D) \qquad \text{(I-15)}$$

Assume there exist in some new region R_2 a vector $P(D^*)$ = $\left[P^*(0), \ . \ . \ . \ , P^*(D) \ , \ ,\right]$ where the sum, $\sum_{R_2} P^*\ (D) = 1.$ such that

$$\upsilon = \sum_{R_2} P^*(D) \cdot VD\ (\bar{A},D) \qquad \text{(I-16)}$$

Since VD $(\bar{A},D)$ decreases as D increases, it is necessary that $VD(\bar{A},D) \geq \upsilon$. This assumption allows min-max analysis by considerations of $P^*(D)$ only, i.e., the result of random selections on the part of the offense and defense may be investigated by considering a random variation in defense selection alone.

<u>The Analytic Model</u>

An analytical pre-commit doctrine based on the above logic when perfect defenders oppose perfectly reliable weapons and both may choose continuous (non-integer) strategies has been developed. The probability of killing the target considers the total number of defenders at a target class and the fact that all targets in this class are not defended, i.e.,

$$P_{K(M)} = P(D = 0)\ P_{K/M} + P(D > 0) \int_0^M P(D)\ P_{K/(M-D)}\ dD \qquad \text{(I-17)}$$

$P_{K(M)}$ = Probability of kill with m firings where defense is present.

$P_{K/M}$ = Original probability of kill with m firings (no defense).

$P(D)$ = Probability of defense level D, where $P(D)$ is defined by the uniform defense distribution.

such that $0 \leq D \leq \frac{2D^*}{S}$

where D is a possible number of defenders at a target

D^* is number of defenders at the class

S is number of defended targets in a class.

Since P(D) is continuous, the probability $P(D = D_o)$ for every $D_O > 0$ is equal to zero.

But by limits

$$\underset{\Delta \to o}{\text{Limit}}\ P(D_o < D < D_o + \Delta) = \underset{\Delta \to o}{\text{Limit}}\ (D_o + \Delta - D_o)\frac{S}{2D^*} =$$

$$\underset{\Delta \to o}{\text{Limit}}\ \Delta \frac{S}{2D^*} = \frac{S}{2D^*}\ dD \qquad \text{(I-18)}$$

Therefore:

$$P_{K(M)} = \frac{N_T - S}{N_T}\ P_{K/M} + \frac{S}{N_T}\int_0^M \frac{S}{2D^*}\ P_{K(M-D)}dD \qquad \text{(I-19)}$$

where S of N_T targets are defended.

Figure I-4 illustrates the original damage function which is defined as:

$$P_{K/M} = \begin{cases} \frac{M}{T_o}\ P_o; & M \leq T_o \\ 1 - (1-P)^{M-T}; & M > T_o \end{cases} \qquad \text{(I-20)}$$

NOTE: $P_o = 1 - (1-P)^{T_o - T}$

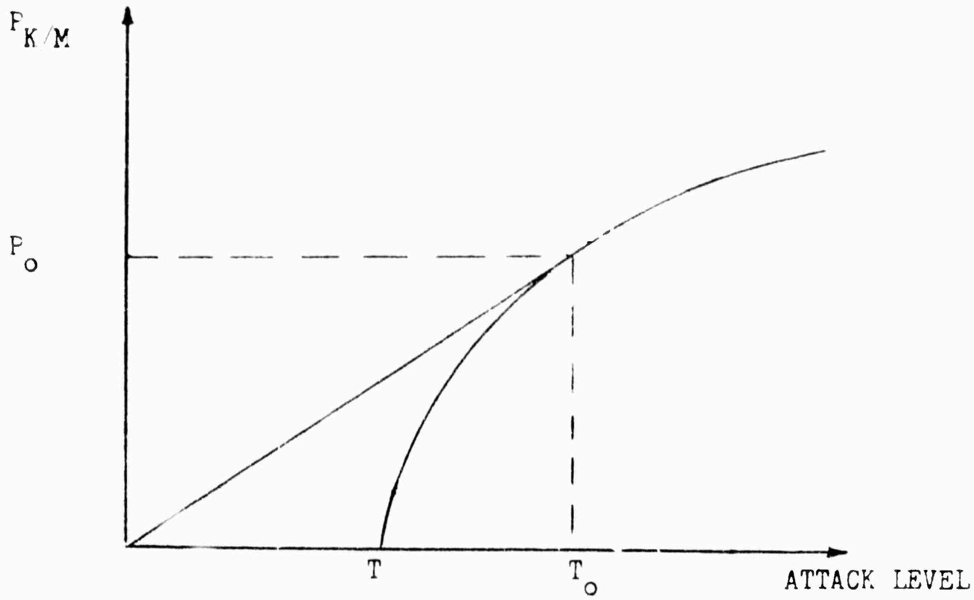


FIGURE I-4   TYPICAL KILL FUNCTION

The net effect of the addition of defense is a flattening of this function. Assuming the existence of a tangent point on the new function ($P_{K(M)}$), it is the objective of the defense to minimize the maximum pay-off per attacking weapon (i.e., minimize the slope to the tangent point on the new function $P_{K(M)}$ since the defense is assumed ignorant of the precise attack level to be suffered).

That is, we are looking for that attack level M_o ($M_o > T_o$) such that $\frac{\partial P_{K(M)}}{\partial M} = \frac{P_{K(M_o)}}{M_o}$, which is the tangent point of the new function. It is assumed that M_o will be reached without possible exhaustion of defense resources at a target since S is continuous in the region and $\lim_{S \to 0} \frac{2D^*}{S} = \infty$. Thus equation (I-19) becomes:

$$P_{K(M)} = \frac{N_T - S}{N_T}\left[1 - (1-P)^{M-T}\right] + \frac{S^2}{2D^* N_T}\left[\int_0^{T_o} D\,\frac{P_o}{T_o}\,dD + \int_{T_o}^{M} 1 - (1-P)^{D-T}\,dD\right] \qquad \text{(I-21)}$$

and completing the integration,

$$P_{K(M)} = \frac{N_T - S}{N_T}\left(1 - Q^{M-T}\right) + \frac{S^2}{2D^* N_T}\left[T_o\left(\frac{P_o}{2} - 1\right) + M - \frac{Q^{M-T} + P_o - 1}{\ln Q}\right] \qquad \text{(I-22)}$$

where $Q = 1-P$.

By noting the tangent point (P_o, T_o) also comprises the maximum value $P_{K(M)}$:

$$\frac{\delta}{\delta T_o}\left[\frac{1}{T_o} - \frac{(1-P)}{T_o}^{T_o - T}\right] = -\frac{(1-P)^{T_o-T}\ln(1-P) + }{T_o} \cdot \frac{(1-P)^{T_o-T} - 1}{T_o^2} = 0$$

$$\text{or,} \quad T_o = \frac{-P_o}{(1-P_o)\ \ln Q} \qquad \text{(I-23)}$$

Equation (I-22) is rewritten to facilitate manipulations:

$$P_{K(M)} = \frac{N_T - S}{N_T} \left(1 - Q^{M-T}\right) + \frac{S^2}{2D^* N_T} \alpha \qquad \text{(I-24)}$$

where α is defined by the integral to be

$$\alpha = \frac{P_o^2 - 2P_o + 2}{2(1-P_o)\ln Q} + M - \frac{Q^{M-T}}{\ln Q} \qquad \text{(I-25)}$$

NOTE: α = area under original P_K function to an attack level M. Derivation of the optimum defended subset (S_{opt}) i.e., that S which minimizes $P_{K(M)}$ is accomplished by differentiation:

$$F = \frac{\delta P_{K(M)}}{\delta S} = -\frac{(1-Q^{M-T})}{N_T} + \frac{2S}{2D^* N_T} \alpha = 0$$

$$S_{opt} = (1-Q^{M-T}) \frac{D^*}{\alpha} \qquad \text{(I-26)}$$

Which minimized equation (I-24) if and only if

$$\frac{\delta F}{\delta S} = \frac{\delta^2 P_{K(M)}}{\delta S^2} = \frac{\alpha}{D^* N_T} > 0$$

By the derivation of α, it has been defined as the area under the damage function from zero to M attack level, which is always $\geq$ 0. Therefore, the defended subset S_{opt} in equation (I-26) minimizes equation (I-24).

Replacing S by its optimum value, equation (I-24) becomes:

$$P_{K(M)_s} = (1 - Q^{M-T}) \left[1 - (1 - Q^{M-T}) \frac{D^*}{2 N_T \alpha} \right] \qquad \text{(I-27)}$$

If $\frac{D^*}{N_T} > M$, it is assumed that $S = N_T$ (all targets are defended) and equation (I-24) becomes:

$$P_{K(M)_s} = \frac{N_T}{2D^*} \sim \qquad \text{(I-28)}$$

Since the defended fraction of the target class (S) is predetermined, equation (I-17) must be modified to allow attacks exceeding $\frac{2D^*}{S}$;

$$P_{K(M;\ M \; \frac{2D^*}{S})} = P_{D=0}\, P_{K/M} + P_{D>0} \int_0^{\frac{2D^*}{S}} P(D)\, P_{K/M+D-\frac{2D^*}{S}} dD \qquad \text{(I-29}$$

since, $M-\frac{2D^*}{S}$ attackers are always "safe" from the defense.

Assuming $M > T_o$ = initial target point (I-29) becomes

$$P_{K(M;\ M > \frac{2D^*}{S})} = (1 - \frac{S}{N_T}) \;\; (1 - Q^{M-T}) + \frac{S^2}{2D^* N_T} \alpha^* \qquad \text{(I-30)}$$

$$\alpha^* = \int_0^{\frac{2D^*}{S}} P(D)\, P_{K/M+D-\frac{2D^*}{S}} \; dD$$

The simple analytic approximations sought by this analysis were now obviously not simple. Consideration of additional encumbrances such as multiple weapon types suggested a few more complex analysis would be necessary before a usable analytic model was developed. Therefore, only the cases where no initial translation in the kill function existed will be discussed in the remainder of this section.

If $T = 0$, α^* becomes:

$$\alpha^* = \int_0^{\frac{2D^*}{S}} \left[1 - Q^{M - \frac{2D^*}{S} + D} \right] dD$$

$$= \frac{2D^*}{S} - \frac{Q^M}{\ln Q} + \frac{Q^{M - \frac{2D^*}{S}}}{\ln Q} \qquad \text{(I-31)}$$

The model represented by these equations accepts input of number of defenders (per class), number of targets in the class, and damage function of an attacking weapon if no defense is employed. Equations (I-27) and (I-28) are used to find the tangent point of the function $P_{K(M)_S}$. This function is the resulting damage function if the defense could predict the exact attack level and picked the optimal subset of targets, S_{opt}, to defend. The value of S_{opt} at this point is then computed by equation (I-26), locked in, and a new function $P_{K(M)_{S=K}}$ is com[illegible] using equations (I-24) and (I-30). The final function represents the expected damage as a function of attack level when the defense has chosen to defend a subset of his targets which is optimal for the attack level which would bring the attacker a maximum return per attacker.

The equivalence of this model with the algebraic derivation (References 6 and 10) for perfect weapons and perfect defenders has been computed to insure the desired compatibility. The corresponding notation used in these references is included to facilitate rapid acceptance by a suspicious reader:

Notations:

$P_{K(M)_s}$ = H = pay-off function

$\frac{D^*}{N_T}$ = D = average defense level per target

M = $\bar{N}$ = expected attack level

S/N_T = S_o = probability that a target is defended.

For perfect weapons, $P_{K/M} = 1.$, $Q = 0.$, $T_o = P_o = 0.$ Therefore the area under the function (α) is $1.\cdot\bar{N}$, or from equation (I-25)

$$= \frac{(0.)^2 - 2.(0.) + 2.}{2.(1-0.)\ln(0.)} + M - \frac{(0)}{\ln (0)}$$

and since $\lim_{X \to 0} \frac{1}{\ln(X)} = 0$

Equation (I-27) becomes

$$P_{K(M)_s} = H = 1 - \frac{D^*}{2M\ N_T} = 1 - \frac{D}{2\bar{N}}$$

Equation (I-26) when divided by N_T becomes:

$$\frac{S_{opt}}{N_T} = S_o = \frac{D^*}{N_T M} = \frac{D}{\bar{N}}$$

and equation (I-28) reduces to:

$$P_{K(M)_s} = H = \frac{N_T}{2D^*}\ M = \frac{\bar{N}}{2D}$$

Comparison of Analytic and Game Theoretic Results

A typical comparison of game results with the analytical model is presented in Figure I-5. It will be noted that the analytic model is apparently better for the defense if $\overline{A}/\overline{D}$ is low. However, the game results considered integer points only while the analytic model uses continuous offense and defense strategies. The impact of continuous versus discrete interpretation is presented in Table I-5 by allowing small discrete steps in the strategies inserted into the game.

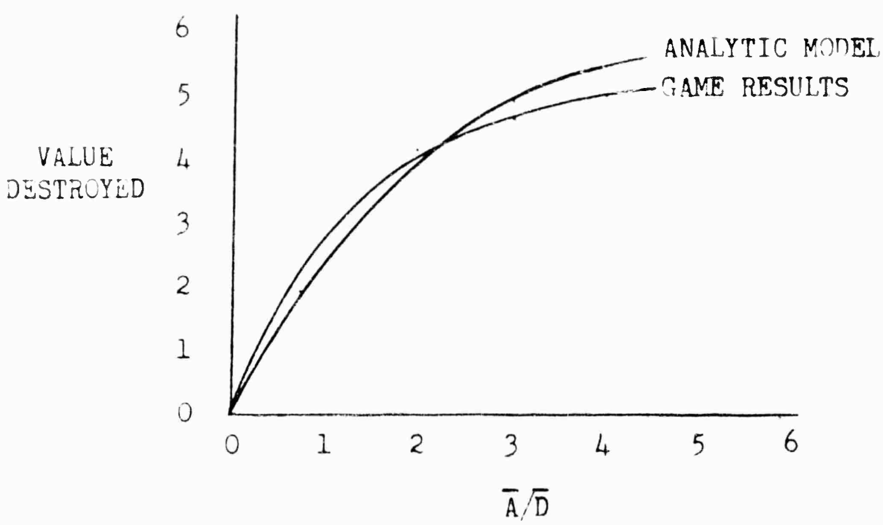


FIGURE I-5 COMPARISON OF ANALYTIC AND GAME THEORETIC RESULTS

TABLE I-5 EFFECT OF INTEGER STRATEGIES ASSUMPTION

GAME SOLUTIONS					ANALYTIC ** SOLUTION
CASE *	DISCRETE STRATEGY INTERVAL	AVERAGE ATTACK	AVERAGE DEFENSE	RESULTANT VALUE DESTROYED	VALUE DESTROYED
	1.0	1.35	1.2	.27	.278
1	0.25	1.678	1.2626	.3356	.3361
	0.20	1.7523	1.2636	.3505	.3507
	1.0	.75	.75	.1875	.1952
2	0.1	1.07	.8169	.26747	.26757

* Both cases are for perfect defenders having Lagrangian cost = .15 defending a target of value 1. against an attack having a single shot kill of .5. The Lagrangian costs for the attacking weapons (λ) is .2 in case 1 and .25 in case 2.

** The analytic solutions were obtained by defending a target by the average defense resulting from the game solution and computing the damage as if the attack level were the average attack from the game solution.

This evaluation demonstates the basic inaccuracy of the continuous assumption. It is of interest to note that in all of the above calculations the average attack produced by the game was less than the tangent point described by the analytic model, i.e., evaluation of damage was done on the linear rise to the tangent point of the function $P_{K(M)_{S} = K}$. The possible relation to the linear behavior of the game solutions in certain regions (see Figure I-3) has not been determined.

Additionally, a discrete analytical approach has not been attempted. This is largely due to the combinational complexity inherent in this technique. This complexity appears even more burdensome when it is realized that the analytic model does not consider weapon reliability nor defense leakage which are additional combinational encumbrances.

The results are generally in good agreement with game theory and future efforts to express game results analytically would most likely be fruitful. However, caution is required in that the effort expended in this area should not be out of proportion to the projected utility of the end result. The current forecast of this utility suggests an adequate approach would be the development and implementation of approximations to the game theoretic results in terms of damage and resource expenditure. Questionable results from such approximations could be challenged by appropriate analysis using game matrices. It is anticipated that such approximations as are developed will require periodic review as they are utilized which may necessitate further efforts to resolve pre-commit defenses analytically.

Approximations for Exponential Kill Functions

Analyses of a large number of cases using this model suggested a general approximation for those cases having no initial PK translation, namely, there exists a translation (T) in conjunction with the original PK such that the area defined by the resultant tangent point (P_o', T_o') equalled twice the average defense per target, i.e.:

$$P_o' \approx \frac{S}{N_T} \;;\; T_o' \approx \frac{2D^*}{S} \;;\; \therefore \; P_o' T_o' \approx \frac{2D^*}{N_T} \qquad \text{(I-32)}$$

and since from equation (I-23)

$$T_o' = \frac{-P_o'}{(1-P_o') \ln Q}$$

it is possible to compute an approximate P_K at the tangent point with use of fixed parameters and no extensive mathematical process.

$$P_o' \approx \frac{\ln Q \, D^* + \sqrt{D^{*2} \ln^2 Q - 2N_T D^* \ln Q}}{N_T} \qquad \text{(I-33)}$$

The following equivalences with the slope to the tangent point (a) were also noted.

$$a = \frac{P_{K(M_o)}}{M_o} = \frac{P_{K(M_o)}}{M_o} \approx \frac{(1-Q^{M-T})^2 D^*}{2N_T} = \frac{S^2}{2D^* N_T}$$

The above relationships can be shown by demonstration, however, a mathematical proof is not now available.

The final curves of $P_{K(M)_{S=K}}$ when an initial translation was present were difficult to fit to the standard PK-with-a-translation format. It is suspected that a proper fitting analysis might produce similar approximations.

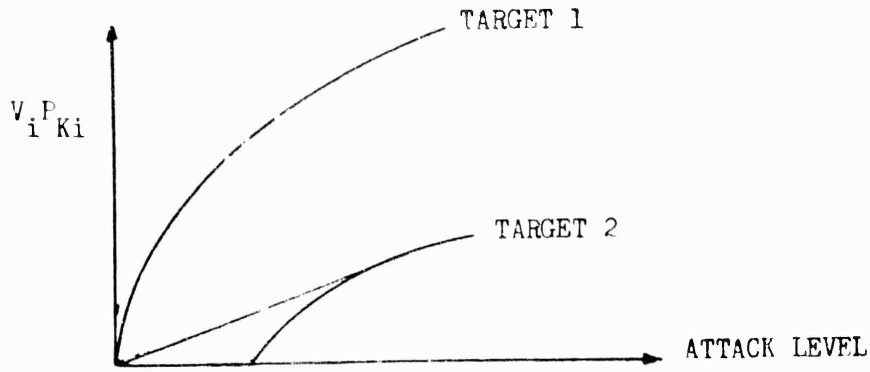
c. Approximating Pre-commit By Using Optimum Terminal Defenses

An effort to equate pre-commit defenses with currently programmed defense models was conducted to investigate any qualities unique to pre-commit defenses and to test the feasibility of approximating pre-commit by manipulation of the input of existing defense models. The above approximations allowed a direct comparison of the analytical pre-commit model with optimum terminal defenses.

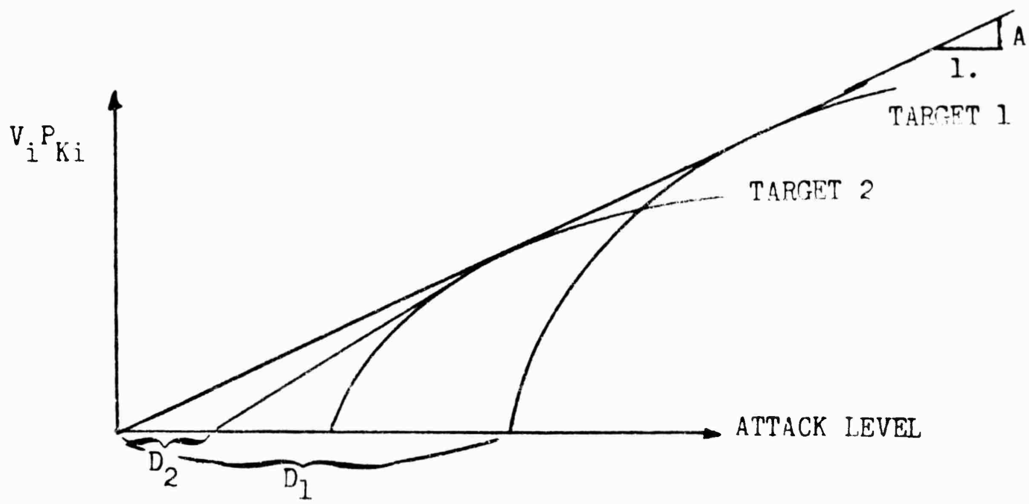
Two models were developed for this comparison. The first model allows determination of an optimum terminal deployment (continuous) given a number of interceptors and the second involved the usage of the above approximations.

Optimum Continuous Allocations of Perfect Terminal Defenders

Optimum allocation of perfect defenders ($\overline{D}$) among I sets of targets having N_i numbers of value V_i is achieved when any attack by weapons having a damage function P_i, T_i against these targets obtains pay-off $\leq$ a constant value per attacker and all defenders are used. This is accomplished by deriving the minimum/maximum average pay-off (i.e., the tangent point of the damage function on each target and making the slope constant for each target). Since perfect defenders result in a linear increase in translation, resolution of the minimum slope tangent to each damage function as illustrated below produces the desired effect.



ORIGINAL DAMAGE FUNCTIONS



DAMAGE FUNCTION AFTER DISTRIBUTION OF DEFENDERS

where: $D_1 N_1 + D_2 N_2 = \overline{D}$

When optimal deployment is attained, the slope to the tangent points (A) is a constant for each target, i.e.,

$$A = \frac{V_i}{M_i} \left[1 - (1 - P_i)^{M_i - T_i - D_i} \right] = K \tag{I-35}$$

Where M_i is the attack level at the tangent point on target i.

Letting $q_i = 1 - P_i$

$$V_i \, q_i^{M_i - T_i - D_i} = V_i - A \, M_i \tag{I-36}$$

$V_i \, q_i^{M_i - T_i - D_i}$ = Value surviving at the tangent point

AM_i = Value destroyed at the tangent point.

T_i = The original translation in the kill function.

D_i Number of defenders at target i.

Removal of value allows the normal computation of the tangent point by equating the slope of this damage function to the slope of the individual kill functions.

$$M_i = (T_i + D_i) + \frac{\ln(a_i) - \ln(-\ln q_i)}{\ln q_i} \qquad \text{(I-37)}$$

(See Equation (13) Appendix A)

$a_i = \frac{A}{V_i}$ the scope of the kill function on target i.

or

$$D_i = \frac{1}{a_i} + \frac{1}{\ln q_i} - \frac{\ln a_i}{\ln q_i} + \frac{\ln(-\ln q_i)}{\ln q_i} - T_i \qquad \text{(I-38)}$$

This expression is summed to equal the number of defenders.

$$\text{Let } C' = \sum_i N_i \quad \frac{1}{\ln q_i} + \frac{\ln(-\ln q_i)}{\ln q_i} - T_i$$

(since these terms are not affected by changes in the defense.)

$$\bar{D} = \sum_i N_i \left[\frac{1}{a_i} - \frac{\ln a_i}{\ln q_i} \right] + C' \qquad \text{(I-39)}$$

Replacing a_i by $\frac{A}{V_i}$;

$$\bar{D} = \sum_i N_i \frac{V_i}{A} - \sum_i N_i \frac{\ln A - \ln V_i}{\ln q_i} + C' \qquad \text{(I-40)}$$

$$\bar{D} = \frac{1}{A} \sum_i N_i V_i - \ln A \sum_i \frac{N_i}{\ln q_i} + \sum_i N_i \frac{\ln V_i}{\ln q_i} + C' \quad \text{(I-41)}$$

Let $C_1 = \sum_i N_i V_i$

$$C_2 = \sum_i N_i / \ln q_i$$

$$C_3 = C' + \sum_i N_i \ln V_i / \ln q_i$$

equation (I-41) becomes

$$\bar{D} = \frac{C_1}{A} - C_2 \ln A + C_3 \quad \text{(I-42)}$$

Newton-Raphson iteration was employed to find the value for A that optimally distributes the interceptors using the following equations:

$$F(A) = \frac{C_1}{A} - C_2 \ln A + C_3 - \bar{D}$$

$$F'(A) = \frac{\delta F(A)}{\delta A} = -\frac{C_1}{A^2} - \frac{C_2}{A}$$

Upon convergence, the D_i were computed by equation (I-38).

Optimal Allocations of Pre-commit Defenders (Using Approximations)

The allocation of defenders to a number of target classes while using a pre-commit doctrine is a two-stage operation. The number of defenders to commit to a class ($D_i^* = N_i \cdot D_i$) and the number of targets in each class (consisting of N_i targets) to be defended to any specified level (D) must be established. If there are $\bar{D}$ total defenders to be allocated ($\sum_i D_i N_i = \bar{D}$), the process should minimize the maximum accrued damage suffered by any attack. The approximations to the analytic model discussed above allow a simple calculation of the effect on the damage function if D_i^* is known. The approximation used (equation I-33) states that the damage incurred when the most efficient use is made of attacking weapons (i.e., at P_o' which is the damage at the tangent point on the resultant $P_{K(M)_{s=constant}}$ curve) may be found by solution of the following quadratic:

$$P_o' = \frac{\ln q_i \; D_i^*}{N_i} + \frac{\sqrt{D_i^{*2} \ln^2 q_i - 2 N_i D_i^* \ln q_i}}{N_i}$$

The following model assumes the minimum damage will occur when for every target i, the slope to the tangent point (P_o', T_o') times the value of the target (V_i) is equal,

$$V_i \frac{P_{o_i}'}{T_{o_i}'} = V_j \frac{P_{o_j}'}{T_{o_j}'} \qquad \text{for every } i, j$$

and exactly $\overline{D}$ defenders are used.

Using the relationship stated in equation (I-23) and defining C to be the slope of the value destroyed function to the tangent point, the problem is to find that C_R such that

$$C_R = V_i \, (P_{o_i}' - 1) \ln q_i \quad \text{(for every i.)} \qquad \text{(I-44)}$$

and exactly $\overline{D}$ defenders are required.

Substituting equation (I-33) into equation (I-44) and solving for the resultant C;

$$C = V_i \ln q_i \left[\frac{\ln q_i D_i^* + \sqrt{D_i^{*2} \ln^2 q_i - 2 N_{T_i} D_i^* \ln q_i}}{N_i} - 1. \right] \quad \text{(I-45)}$$

It should be noted that

$$\sum_i N_i D_i < \bar{D} \quad \text{implies } C > C_R \quad \text{(I-46)}$$

and $\sum_i N_i D_i > \bar{D}$ implies $C < C_R$

Equation (I-45) is rewritten to allow the solution of individual D_i.

$$D_i^* \left[1 + \sqrt{1 - 2 N_i / D_i^* \ln q_i} \right] = \frac{C N_i}{V \ln^2 q_i} + \frac{N_i}{\ln q_i} \quad \text{(I-47)}$$

The following approximation was made to remove the square root:

$$\sqrt{1 - X} = 1 - \frac{X}{2} - \epsilon \quad \text{(I-48)}$$

Using equation (I-48), equation (I-47) becomes:

$$D_i^* \approx N_i D_i = \frac{C N_i}{V_i \ln q_i} + \frac{2 N_i}{\ln q_i} \quad \text{(I-49)}$$

$$\sum_i D_i^* \approx \sum_i \frac{CN_i}{V_i \ln^2 q_i} + 2 \sum_i \frac{N_i}{\ln q_i} \approx \bar{D} \quad \text{(I-50)}$$

The solution was obtained by step-wise iteration.

Comparative Analysis

Since this is an approximation model, comparisons with the results of the analytic pre-commit model were made. The error was found to be almost non-existent.

A comparison between the above models (pre-commit using approximations and optimal terminal) is presented in Table I-6. This case is comprised of four target classes which may be attacked by a single weapon type resulting in a simple exponential damage function. The translation shift due to terminal defenses is the same as the number of terminal defenders at a target. The average number of pre-commit defenders per target and the resultant kill function translation are also presented. The last column is the optimum spreading of 146 translation units to establish the error in the equal damage slope calculations used in the pre-commit approximations.

TABLE I-6

COMPARISON OF OPTIMUM TERMINAL AND PRE-COMMIT DEFENSES

TGT CLASS	NUMBER IN CLASS	TGT VALUE	P_{KSS}	NO. OF TERMINAL DEFENDERS PER TARGET ($\Sigma = 100$)	AVERAGE NO. OF PRE-COMMIT DEFENDERS PER TGT	RESULTANT TRANSLATION FROM PRE-COMMIT MODEL	EQUIV. TERM. DEFENDERS TO APPROX. PRE-COMMIT ($\Sigma = 146$)
1	2	10	.2	13.28	11.359	17.8674	17.8685
2	5	5	.3	5.7255	5.208	7.9107	7.9105
3	10	3	.4	3.0388	2.9033	4.3000	4.2966
4	50	1	.5	.2884	.444	.5538	.5548

Cumulation of the resultant pre-commit translations yields 145.9783 units of displacement. Therefore, the worth of a pre-commit defender in this case relative to optimum terminal defenses is 1.46:1. The slope to the value destroyed at the tangent point is .322 for pre-commit and .391 for optimum terminal giving a .825:1 advantage to pre-commit on maximum damage per offensive weapon.

The advantage of pre-commit over optimal terminal could not be resolved into a singular ratio by this analysis even though only one attacking weapon type was evaluated at a time. However, the analysis indicates there may be some equivalence in the more general cases, perhaps including such mathematical anomalies as multiple attacking weapon types.

The compatibility of these results leads to a comparison between optimum terminal deployment and the game theoretic solution. Table I-7a shows a target structure and pertinent parameters. The results of the game analysis for three separate levels of defense resources are presented in Tables b, c, and d. Iterations on the number of optimally deployed terminal defenders to yield the same total damage (if the mean attack produced by the game matrices were employed against each target) were conducted and are also presented in these tables.

TABLE I-7a COMPARISON OF OPTIMUM TERMINAL DEFENSES AND GAME THEORETIC SOLUTIONS

TARGET CLASS	NUMBER IN THE CLASS	VALUE OF EACH	PROBABILITY OF KILL P_{KSS}	TRANSLATION
1	1	14.	.09	15.72
2	2	6.2	.12	8.1
3	3.75	3.	.179	3.1

TABLE I-7b

CASE 1: 100 ATTACKERS, 50 PRE-COMMIT DEFENDERS

TARGET CLASS	GAME RESULTS (λ = .158215; μ = .146) MEAN ATTACK	MEAN DEFENSE	VALUE DESTROYED PER TARGET	OPTIMUM TERMINAL EQUIVALENT (EVALUATED AT GAME MEAN ATTACK) EQUIVALENT TERMINAL INTERCEPTORS	VALUE DESTROYED PER TARGET	RATIO: TERMINAL TO PRE-COMMIT
1	41.32	24.22	6.537	40.1286	6.55	1.59
2	15.39	7.32	2.4352	10.7648	2.4397	1.47
3	7.41	2.97	1.1725	4.0910	1.1746	1.375
TOTALS	99.89	50.00	15.805	77.00	15.834	1.54

TABLE I-7c

CASE 2: 100 ATTACKERS, 100 PRE-COMMIT DEFENDERS

TARGET CLASS	GAME RESULTS (λ = .10985; μ = .065) MEAN ATTACK	MEAN DEFENSE	VALUE DESTROYED PER TARGET	OPTIMUM TERMINAL EQUIVALENT (EVALUATED AT GAME MEAN ATTACK) EQUIVALENT TERMINAL INTERCEPTORS	VALUE DESTROYED PER TARGET	RATIO: TERMINAL TO PRE-COMMIT
1	36.46	43.38	4.005	75.6314	4.007	1.745
2	16.865	15.498	1.8526	25.2791	1.8535	1.63
3	7.796	6.82475	.8564	10.6161	.8568	1.555
TOTALS	99.42	99.97	10.922	166	10.927	1.66

TABLE I-7d

CASE 3: 100 ATTACKERS, 150 PRE-COMMIT DEFENDERS

TARGET CLASS	GAME RESULTS ($\lambda = .0846$; $\mu = .0388$) MEAN ATTACK	MEAN DEFENSE	VALUE DESTROYED PER TARGET	OPTIMUM TERMINAL EQUIVALENT (EVALUATED AT GAME MEAN ATTACK) EQUIVALENT TERMINAL INTERCEPTORS	VALUE DESTROYED PER TARGET	RATIO: TERMINAL TO PRE-COMMIT
1	37.45	62.34	3.168	111.2235	3.169	1.785
2	16.647	23.70	1.408	40.1594	1.409	1.69
3	7.9135	10.745	.6695	17.4553	.6697	1.625
TOTALS	100.4	150.04	84.956	257	84.978	1.71

These examples show that while equivalence may be found, prediction of the proper ratio between pre-commit and the equivalent number of terminal defenders is not obvious. As an example, if a constant fifty percent ratio is assumed, the error in total value destroyed is 1.265%, 5.73%, and 8.05% for Cases 1, 2, and 3 respectively. In such a case one might accept 1 to 4% error but the 8.05% is certainly unacceptable. It is thus obvious that additional analysis would have to be done to establish proper rules leading to the least error assumptions and a maximum error computed.

It is of particular interest to note that these comparisons ignore computation of the mean attack except by game play. Both the analytic pre-commit and optimal terminal models were developed from a defensive point of view when complete uncertainty about the attack level is assumed. Since the major task of the Arsenal Exchange Model is the allocation of weapons, an honest approximation of the game attack strategy is mandatory. In comparisons between the analytic models and game solutions, invariably the average attack from the game solution was to the left of the tangent point of the final analytic kill function. However, great care was taken to provide programmatic indifference in this linear region, regardless of target class (equal value destroyed to tangent point). Therefore, if the same total level of resource is assumed, attacking five of ten targets to the tangent point is the same as attacking all ten to half the level (regardless of which five targets are attacked). Thus, it is unimportant that the actual attack on each target be precisely predicted.

d. <u>An Approximation Using Preferential Defense</u>

From the offensive point of view, the random defense level at a target can be represented as an additional uncertainty that a given RV will penetrate the defense. In this representation, the degree of uncertainty is a function of the relative probability that the defense picks a specific defense level.

At the same time, the offense knows that the defense level at a target will be chosen in such a manner that the defense tends to match the expected attack level on that target. Since the defense does not know the exact attack level, he will sometimes be high and sometimes low, but always in the correct range.

This kind of thinking led to the concept of modeling the effect of pre-commit by use of an equivalent, imperfect preferential defense. This representation allows for the random defense level at a target while offering a reasonable logic for the offense and defense choices at each target.

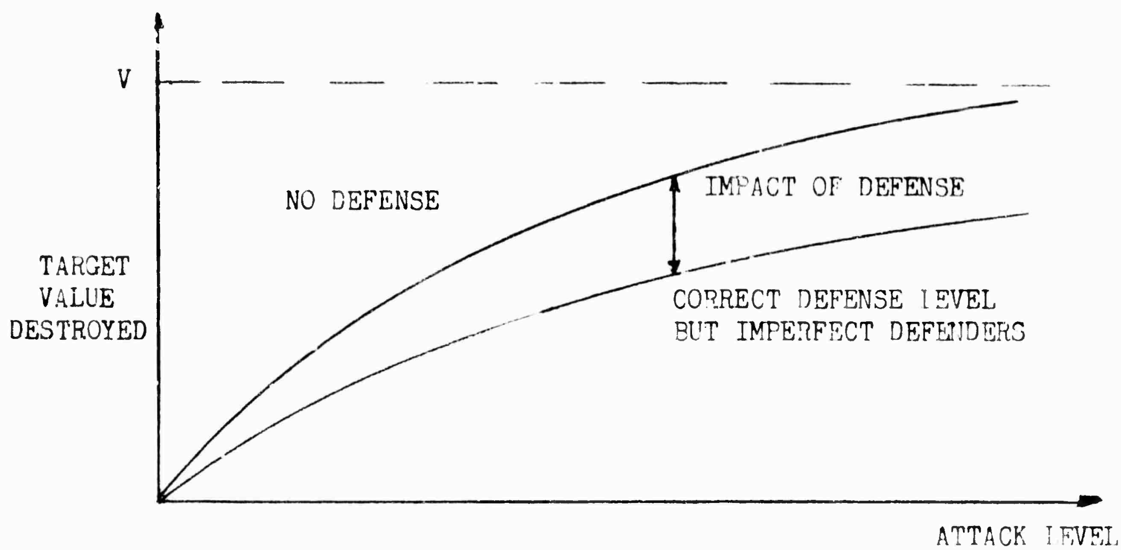
The preceding parts of this section presented evidence that the minute details of the random strategies are probably not crucial in the final determination of total value destroyed. Additionally, an analytical model was developed around the assumption that a reasonable defense strategy is to pick a subset of targets to defend and to defend each target to a level somewhere from zero to twice the expected attack level. That model has been shown to result in a final value destroyed very close to the optimal which could be achieved if both offense and defense played with game theoretic strategies, even though this specific defense doctrine is not optimal for all types of target damage functions.

Following along the same line of reasoning led to the idea of a preferential defense approximation as to be described. The whole concept revolving around the assumption that the defense correctly predicts the expected attack level on each target and then places a random defense level at e... target. The defense level coming from a distribution uniform from zero to twice the expected attack level.

Meanwhile, the offense knows that the defense can predict the expected attack level but also knows that the defense level will still not be precisely correct. This giving him enough information to optimally design an attack which will exploit the random nature of the defensive allocation.

In Section IV-H, the logic for determining optimal offense and defense strategies was presented. Specifically, this logic allowed for an imperfect defense, i.e., a defense which allows RVs to penetrate because of non-unity kill probability or because a specific RV could not be fired at. Within the context of pre-commit, there is an additional cause for leakage, namely when the defense has less interceptors than he needs at a target.

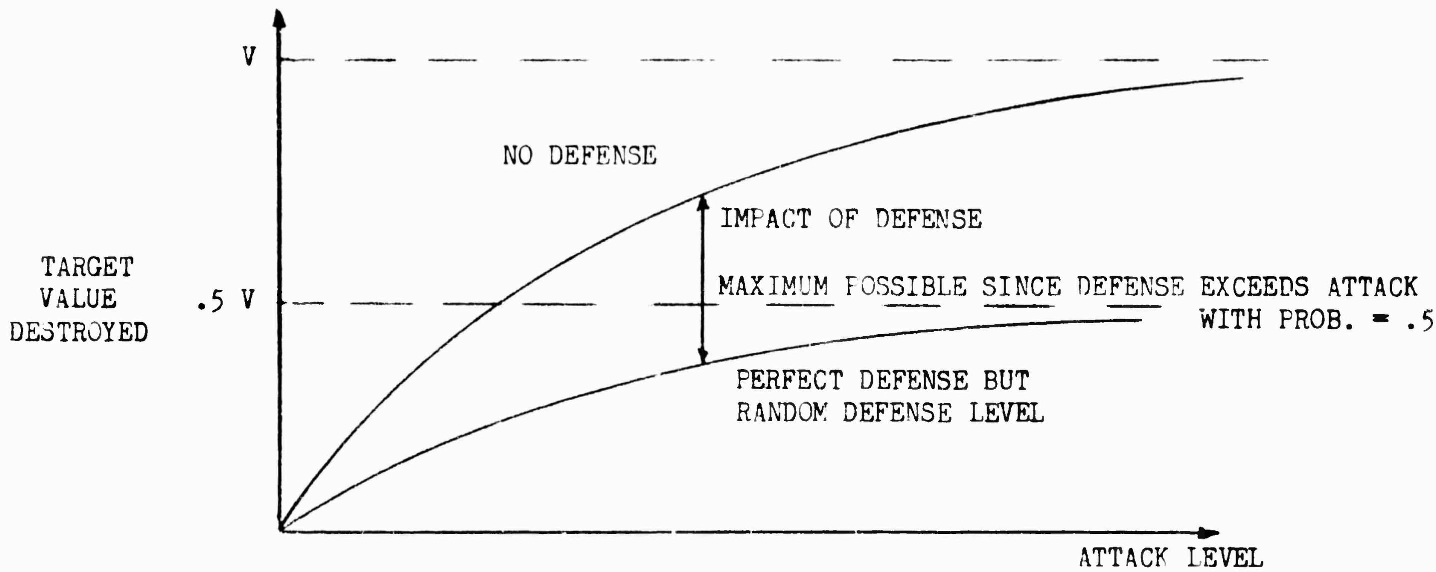
The key ingredient in developing optimal offense strategies when the preferential defense is imperfect is the delta damage function as in the following figure.



This figure demonstates the impact of an imperfect defense as measured by the value destroyed, which is a function of the attack level and the presence, or lack of a defense.

In the case of pre-commit, the same ingredient exists. If a given target is undefended, the damage function is the normal, no defense function. If the target is defended, but to a random level, the damage is reduced at any given attack level with the size of the reduction dependent upon the kind of defense randomization.

For example, consider the case where the only cause of leakage is the random defense level. In such a case, if the defense has the zero to twice the expected attack level form, there is a probability of .5 that no penetrations will occur and a probability of .5 that from zero to all attackers will penetrate. The above damage function then looks as follows:



This presence of a pre-commit changes the form of the defense impact but will not modify the methodology for choosing optimal defense strategies. The only change necessary to the previously described leaky preferential logic is to specialize the equations which compute the defense impact as a function of the attack level.

The basic equations for the imperfect preferential defense (see Chapter IV - Section H) resulted in a delta damage relationship as follows:

$$VS = V\left(P_{K/N} - P_{K/(N,D)}\right) \quad \text{(I-51)}$$

Where:

VS = Value saved by the defense if an attack level = N occurs and if the defense chooses to defend this target.

V = Value of the target.

$P_{K/N}$ = Target probability of kill for attack = N and no defense.

$P_{K/(N,D)}$ = Target probability of kill for attack = N, where each RV has a degraded reliability because of the defense.

When the defense is really a pre-commit defense with some random defense level, this equation becomes

$$VS' = V\left[P_{K/N} - .5\ P'_{K/(N,D)} - .5\ P''_{K/(N,D)}\right] \quad \text{(I-52)}$$

Where:

$P'_{K/(N,D)}$ = Target probability of kill when the defense level matches or exceeds the ideal for the attack level.

$P'_{K/(N,D)}$ = 0 if the defense is perfect.

$P''_{K/(N,D)}$ = Target probability of kill when the defense is uniformly likely to be from zero to exactly the ideal for the attack level.

If one assumes that the interceptors are perfect, but that an incorrect defense level was committed to the target, equation (I-52) can be expanded somewhat. First, it is necessary to indicate an effect of the defense on the arriving objects. Assume, for example, that the defense simply subtracts from the arriving objects. Then, if the defense is uniformly likely to be at a level anywhere from zero defenders to a level of exactly N defenders, we can state that

$$P''_{K/(N,D)} \approx \frac{1}{N} \int_0^N \left[1 - Q^{N-T} \right] dT \qquad \text{(I-53)}$$

where:

N = number of attackers.

Q = probability of single-shot survival.

T = subtraction due to a defense level of T.

(This equation is based on the assumption that the probability of a defense level of amount T to T + dT is equal to $\frac{1}{N}dt$.)

Solution of this equation results in

$$P''_{K/(N,D)} \approx 1 + \frac{1}{\ln Q} \left[\frac{1}{N} - Q^N \right] \qquad \text{(I-54)}$$

Note that the other terms in equation (I-52) for the perfect interceptor case amount to

$$P_{K/N} = 1 - Q^N \tag{I-55}$$

and

$$P'_{K/(N,D)} = 0. \tag{I-56}$$

Thus, all parts of (I-52) can be computed for a given attack strategy simply by knowing the attack level and the single-shot survival probability.

If the interceptors are not perfect, the same sort of development can be used to arrive at a somewhat more complex version of these relationships. In such a case, however, VS' can still be easily computed.

Using the (I-52) relationship, the preferential defense approximation to pre-commit ope- -es as follows. The offense assumes that the defense can predict his attack on each target, but that the defense chooses his distribution from the special distribution of zero to twice his attack level. The offense then uses the relationship to predict which targets the defense would choose to defend and picks an attack level for those targets so that a constant return per interceptor is attained by the defense. At the completion of the sequence the attacker has converged on a mean attack level at each target and the defense has selected targets to defend. Also, the total value destroyed in the presence of a randomized defense is then available.

The quality of the approximation was investigated by making comparative runs for this method and for identical cases in the game theoretic method previously described. The results of these comparisons are presented in Table I-8.

TABLE I-8

COMPARISON OF GAME THEORETIC RESULTS WITH THE PREFERENTIAL DEFENSE APPROXIMATION

EXPECTED ATTACK LEVEL	EXPECTED DEFENSE LEVEL	GAME THEORETIC RESULTS	PREFERENTIAL APPROXIMATION RESULTS
454 TYPE 2	172	529	523
427 TYPE 2	34	666	662
92 TYPE 1	58	382	383
74 TYPE 1	58	295	286
78 TYPE 1	6	597	596
91 TYPE 1	13	592	604
75 TYPE 1	100	677	677
400 TYPE 2			
400 TYPE 2	100	520	528

The target structure used was an approximation of a real population listing and the weapons were both small (Type 2) and large (Type 1) yield weapons. (Classified data can be furnished if necessary.)

As can be seen, the approximation is very good (errors in the 1 to 2% range). However, it should be noted that all cases represent circumstances where the offense is dominant. This is the ideal circumstance for the approximation since the assumption about the defense behavior is most directly applicable then.

A brief investigation of the defense dominant condition verified the suspicion that the approximation would then degenerate. As the defense level comes close to matching the attack level, the defense should play a stronger role but this approximation does not properly account for such an effect.

In the near future, the defense dominant condition will be analyzed in more depth. The assumption being that the previously described approximation can be broadened to adequately consider such circumstances.

It is worth-while noting an interesting characteristic of the approximation. Inspection of the optimal offense strategies and comparison with the mean attack level on each target as produced by game theory showed a remarkable similarity. It appears that the optimal mean attack strategies can be developed by use of the approximation.

A useful by-product of the approach was the verification that the IAT convergence process for imperfect preferential defense must be functioning properly. If the process was not functioning, there could never be the close match as seen in the previous table.

3. Status and Conclusions

Analysis of the pre-commit defense doctrine has been roughly divided into three areas of work:

- Formulation and solution of optimum min-max games using discrete integer strategies and Lagrangian constraints.
- Analytic formulation of a near optimum pre-commit doctrine for continuous attack and defense strategies.
- Applicability of other defense doctrines (i.e., optimum, terminal and preferential) in approximating the effects of a pre-commit defense doctrine.

No working model for the pre-commit defense doctrine is currently resolved that is completely compatible with the Arsenal Exchange Model. The problems associated with each of these areas are as follows:

Solution of Pre-commit By Game Theory

The game yields the optimum pre-commit solution and for that reason is the basis for determining the accuracy of any approximation or assumption used to model the effect of this defense doctrine. Since each game solution corresponds to one contending strategy (weapon to target and defense to target allocation), direct implementation of this technique would greatly increase computational complexity and program running time. Additionally, the process for iteration on Lagrangian cost should this method be used has not been developed.

Approximations to game results based on game inputs have not been successful. However, such approximations may be developed in the course of testing future assumptions.

Solution of Pre-commit by Analytic Means

The analytic model developed herein has shown the weakness of assuming continuous offensive and defensive strategies. It has demonstrated the insensitivity to precise strategy distributions and has provided additional encouragement that a near-optimum pre-commit doctrine may be simply approximated. The development of a discrete analytic model is not anticipated.

Approximation of Pre-commit By Developed Defense Doctrines

Optimum terminal defense deployment has been shown to approximate the effect of a pre-commit doctrine in the cases analyzed. However, the equivalent number of terminal defenders was 40 to 70% greater than pre-commit. Resolution of this enhancement factor into a constant or a simple function has not been accomplished. Additional analysis will be necessary to ascertain the accuracy of using this technique where mixed weapon types having wide variations in numbers and capabilities are involved.

The equivalence with a modified, imperfect preferential defense has been analyzed. This approximation is very accurate when the number of defenders is less than the reliable attack level. Efforts will be made to broaden this approximation to allow reasonable accuracy when the defense is equal to and greater than the attack. Despite this limitation, the current version of AEM has programmed a pre-commit approximation of this form. Until improvements, or an alternate method is found, its usage will be restricted to the offense dominant condition.

J. OPTIMIZED BUDGET ALLOCATIONS - CURRENT STATUS

At the present time there is a capability within the program to allocate a budget among a specified list of offensive and defensive options. This capability is still considered to be in an R&D phase since extensive operating experience has not yet occurred. However, the capability has checked out quite well in an extensive testing process that was conducted. The purpose of this section is to indicate potential applications, describe the current status of the methodology and discuss any limitations of the process as it currently exists.

Before delving into the mathematics of the process, it would be useful to discuss the requirement for such a capability and the type of applications visualized. In this manner, it is possible to indicate the conditions which should be met by the final product.

1. Potential Applications

a. Uncertainty Improvement Break-Even Costs

One important application of this model is to utilize the uncertainty option to evaluate the impact of misestimates in any of the characteristics of the resources involved in an exchange. Measurement of the impact of various kinds of misestimates then can lead to a discussion of the relative desirability of expenditures to reduce the uncertainties.

It is often useful in this process to have available a break-point expenditure which represents the maximum allowed cost to reduce the misestimate. This cost exists because there is an alternate, competing way to overcome the misestimate. For example, assume that the misestimate involves one side's estimate of his own weapon reliability. He can overcome lack of knowledge of his own reliability by either purchase of more weapons or by testing to determine the reliability more exactly. The break-even cost for testing is therefore the cost of the additional weapons required to exactly equal the improvement in total value destroyed that would occur if the testing was done and better knowledge of how to use the weapons was available.

An accurate break-even cost can be obtained with an optimum budgeting routine. First, it is necessary to determine the delta damage change obtained for a selected improvement in a given misestimate. Then, using a selected list of force options it should be possible to determine the force mixture which achieves the same delta damage change at a minimum cost.

The ideal feature of the optimum budgeting routine to use in this process would be one with the capability to specify the desired delta damage and have the program determine the minimum cost to meet the objective. Lacking this ideal, it is possible to estimate the required budget, optimally distribute the budget, look at the obtained delta damage, make a new estimate of the budget, and thus eventually converge on an estimate of the exact budget to obtain the desired delta damage.

b. <u>New Weapon System Requirements and Effectiveness Evaluations</u>

Another common application of this program is the analysis of various force options--both offensive and defensive. In such an analysis each option can be inserted into the total arsenal and an individual run made to determine the benefit attained. If the best expenditure of a budget involves a mixed offensive force or a mixture of offense and defense, this process leads to a considerable number of combinations.

After a few attempts at repetitive case runs with variable options and costs, it becomes obvious that a built-in optimization would be very useful. With such a capability the emphasis can be upon obtaining correct costs and other inputs rather than upon the mechanical process of making computer runs. Generally, threats and characteristics are very variable and the analyst is interested in determining result sensitivity to such variations. With a total budget optimization routine, it becomes practical to determine such sensitivities.

c. Evaluation of Arms Control Agreements

In the near future the emphasis on an arms control agreement might lead to constraints on the composition of a total arsenal. Examples of such constraints are limitations on total offensive throw-weight, total RVs, total megatonnage, etc.

In such a case, it would be useful to determine the most effective arsenal that meets the appropriate conditions. Such an arsenal can be obtained by use of the optimized budgeting routine by appropriate definition of the "budget" in units other than dollars. For example, if the agreement involves limitations on the parameters listed above, each option can have a multi-dimensional "cost." (Namely, the throw-weight, number of RVs and megatonnage used up if one unit of a given option is selected for the arsenal.) Then the program can allocate the "budget" in such a way as to maximize force effectiveness within the arms control agreements.

d. Future Enemy Responses/Actions

Another useful option available with possession of an optimized budgeting routine is the analysis of enemy responses or actions when one side takes a certain step. For example, assume BLUE has several options open to him. Before selection of one option, it would be useful to know the direction his selection would drive his opponent.

By playing RED's options against BLUE, where BLUE chooses a specific option, it should be possible to gain insight into RED's tendency to respond by more offense, more defense, more countervalue attacks, etc. Thus, study of arms race directions is conceptually feasible.

In this application, it would be mandatory to have the capability to select an optimum force mix for any of the scenarios described for this model. By having the capability for any scenario it would be possible to evaluate your opponent's options in each situation, e.g. first strike or second strike.

With the above background in potential applications, the basic structural modifications to the program can be discussed.

2. The Basic Mathematical Process

The current process for budget allocation can best be explained in stages that roughly equate to the sequence of developments that resulted in the process. This sequence was as follows:

a) Development of a method for optimally allocating a budget to offensive options alone.

b) Development of a method for dividing a budget between offense and defense.

c) Development of a method for dividing a defense budget between a list of defense options.

Each of these stages involved unique problem areas of their own and each one built upon the concepts and processes of the previous stages.

One argument for de-coupling the process into such a sequence of steps is that a valid result possibly can be obtained even though each independent step is not perfectly refined.

Under this philosophy, it is better to separate a large problem into controllable sub-packages that can be operated upon independently. In this way each stage can be made to compensate for inaccuracies in data passed on by previous stages.

The validity of this philosophy will have to be judged after the final product is in use.

a. <u>The Offensive Budget Allocation Procedure</u>

In Chapter IV, Section A, the basic Lagrangian process for optimally allocating weapons to a target structure was developed. Very briefly, in review, the process involves starting with a set of Lagrangian strategies for each target. All such strategies are inserted into a linear program which contains appropriate constraints on number of weapons of each type and number of targets of each class. The L.P. produces new Lagrange multipliers to use in developing better strategies and the process cycles until optimal allocations are obtained.

Of interest to the problem of optimally distributing an offensive budget is the structure of the L.P. Each strategy is inserted into the L.P. in the form of a column. The entries in the column indicate the value destroyed if the strategy is chosen, the number of weapons of each type required and the target to which the strategy applies. (See Figure (J-1).)

This column can be interpreted as a description of the "resources" expended in order to achieve the given value destroyed, or value returned. In this sense, the total weapons of each type are resources and the total number of targets of each type are resources.

Conceptually, the above interpretation leads very naturally to the idea of simple insertion of one more constraint, or "resource" which represents the budget. Then each strategy must include in its description another component, namely, the amount of the budget used up if that strategy is chosen. Visualization of the L.P. with this new constraint can be obtained in Figure (J-1).

Fundamentally, this concept is the essence of the offensive budget optimization process contained in the program. However, additional complexity results for mechanization of the concept for various scenarios.

b. <u>Impact On The Lagrangian Process</u>

Contained in Section D is a discussion of the impact on the Lagrange multipliers when a damge constraint is inserted into the L.P. Similarly, when a budget constraint, or constraints, are inserted into the L.P., it is necessary to reinterpret the multipliers.

Referencing equation (D-5), it was stated that a new strategy will improve the answer if

$$P(X) > \sum_{i=1}^{m} \xi_i' \cdot g_i (X) \qquad (J-1)$$

Where:

$P(X)$ = payoff from strategy X

$g_i(X)$ = level of constraint i called upon by strategy X

ξ_i' = constraint i multiplier from the last L.P.

Use of this equation resulted in the conclusion that a best new strategy must be one that minimizes

$$H = V_j (1 - \lambda_k) \prod_{i=1}^{I} S_{ij}^{N_{ij}} + \sum_{i=1}^{I} \lambda_i N_{ij} \qquad (J-2)$$

Where:

V_j = target value

λ_i = weapon multipliers

λ_k = damage constraint multiplier

N_{ij} = weapons of type i on this strategy for target j

Based on equation (J-1), the addition of a budget constraint must modify equation (J-2) to the following:

$$H = V_j (1 - \lambda_k) \prod_{i=1}^{I} S_{ij}^{N_{ij}} + \sum_{i=1}^{I} \lambda_i N_{ij} + \sum_{i=1}^{I} \lambda_m C_i N_{ij} \qquad \text{(J-3)}$$

Where:

C_i = unit "cost" of each weapon of type i

λ_m = budget constraint lambda

The last two terms can be grouped to obtain an effective weapon lambda of

$$\bar{\lambda}_i = \lambda_i + \lambda_m C_i \qquad \text{(J-4)}$$

and, since $V_j (1 - \lambda_k)$ is an effective value, equation (J-3) then appears like the standard Lagrangian function.

The impact of the budget constraint is thus very direct and conceptually there is no problem with the convergence routines discussed previously. However, it should be noted that the effective weapon lambda $\bar{\lambda}_i$ is not quite appropriate for use as a force value scale as described in Chapter IV, Section C.

The effective weapon lambda measures the cost-effectiveness impact of a certain option on the budget. However, when an opponent places a value on that weapon, it must be in terms of its effectiveness alone. He doesn't care if it is cost-effective, only how purely effective it is against him.

Thus, if appropriate lambdas are necessary for use as a value scale, they must be developed after a specific force mix is chosen. They can be developed by allocating exactly the chosen force mix against the given target system under the assumption that the budget constraint is no longer in existence.

The presence of two terms in equation (J-4) is worthy of comment as it leads to additional understanding of the process being used. The λ_i measures the constraining effect of the normal weapon constraint while the $\lambda_m C_i$ measures the constraining effect of the budget. Normally the λ_i term would be assumed to equal zero as the budget would constrain the problem, but in the case where a current arsenal is being augmented this conclusion is not valid.

When no weapons of a given type are in existence, the impact of the normal weapon constraint is completely relaxed by inserting a number of weapons in the constraint column larger than that which the total budget could purchase. However, when a given number of a certain weapon is in existence, the weapon constraint must equal the number in existence since no more than that number can be purchased without additional investment costs.

Thus, if a current arsenal is to be augmented by any one of several options, the L.P. would contain the existing numbers of the current arsenal and either no constraint, or a completely relaxed constraint on all future options. Accordingly, the costs utilized would represent only O&M costs for the existing weapons and investment plus O&M for the future options.

In this circumstance the future options would have a $\lambda_1 = 0$ but the current systems would have both multiplier terms non-zero if they are in the optimum force mixture.

The availability of a budget constraint multiplier (λ_m) is especially important since such a parameter is useful in its own right. Basically, it indicates the marginal change in payoff for a unit change in the budget. Thus, it is a very useful measure of the cost-effectiveness of the total arsenal.

c. The Survivability Problem

The preceding discussion is representative of the budget optimization problem in any situation where there is no possibility of an attack on the force being purchased. An example of such a situation is a force optimization for RED (Figure 1) in the massive first strike scenario.

However, if the budget optimization is for a side in a situation where his forces will be under attack (BLUE in Figure 1), the additional ingredient of survivability must be considered. This survivability factor causes a degree of complexity which is very difficult to overcome. In fact, only approximate solutions are possible without going to a level of non-linear analysis which is currently unacceptable.

Consider the problem of BLUE optimizing his forces against a possible counterforce attack by RED in the scenario of Figure 1. The obvious way for BLUE to implement the optimization process is for him to insert the appropriate budget constraint into the L.P. which represents his retaliatory strike on RED.

If he does so, this constraint must represent his surviving weapons, either directly or by implication, for whatever force mix the budget constraint causes to occur.

In other words, BLUE desires to come up with the force which after an attack by RED does a maximum damage to RED value targets. If this is BLUE's goal, his force optimization process must include an estimation of the number of survivors of each weapon type as a function of a given total mixture.

The problems caused by this survivability factor can be demonstrated by a discussion of the two methods currently available for including it in the process.

The "Effective" Cost Method

The simplest method considered to date is to insert the budget constraint into BLUE's L.P. as previously described with the exception that the cost per RV of a given type has a modifier to include the estimated survival of that RV. In the L.P. the objective is to maximize total damage on RED value targets under the constraint that the BLUE weapons must not cost more than a certain budget. Since BLUE is allocating survivors, his cost per RV must be the estimated cost per surviving RV, which would be:

$$C_i' = \frac{1}{PS_i} C_i \qquad \text{(J-5)}$$

Where:

C_i = basic cost per available BLUE RV of type i

PS_i = estimated average probability of survival of a type i RV

C_i' = effective type i cost per RV

Using this effective cost the L.P. will proceed to optimally spend the budget on the assumption that exactly PS_i survivors will result for every RV of type i purchased.

In this approach the survivability problem resolves down to the appropriate estimation of the PS_i parameters. The utility of the approach thus is a direct function of the process used in estimating the PS_i.

The basic version of this method of estimating the PS_i as currently programmed is as follows:

a) First BLUE optimizes his budget under the assumption that all weapons purchased will survive.

b) RED then optimizes his counterforce and counter-value attack against the force chosen in step (a).

c) Based on the allocation of RED attackers in step (b), an estimation is made of the way RED would attack an average weapon of each BLUE type.

d) BLUE uses the PS_i estimates to come up with an effective BLUE weapon cost, C_i', and then re-distributes his budget. The process then returns to step (b).

Unfortunately, the process described in steps (a) to (d) is not always convergent. After BLUE makes an estimate of the PS_i he may choose a mixture which causes RED to change his attack, which would cause the estimates of PS_i to change. Quite often an individual PS_i can go from a low level to a high level, etc. as the RED attack keeps changing for BLUE's varying force mix.

This tendency to oscillate in the PS_i estimates has been controlled by the simple technique of weighting the new PS_i estimate by the history of previous estimates so that BLUE would tend to change his PS_i estimate more slowly. Quite obviously, this modified process is not guaranteed to converge either, but considerable experience with the program in this form has demonstrated a reasonable stability.

Quite a number of different cases have been checked out by making several runs of various mixtures chosen by an analyst. These runs demonstrated that an optimum, or very near optimum result had been arrived at.

If one desires to determine the minimum budget to arrive at a specified damage level, this approach will not work very well. In such a case, BLUE not only changes his force mix at each new estimate of the PS_i, but he also changes his estimate of the budget required to meet his objective. This causes more gross changes in BLUE's total force mix and as a result the averaging method of estimating the PS_i breaks down.

Because of the limitations connected with this effective cost method an alternate approach was programmed. At the current time both methods can be utilized and, since they are based on such different concepts, they can be used to substantiate any given result.

The "Effective" Defense Method

Experience with other areas in this program, e.g., determination of the optimum reserve force (Figure 3), has demonstrated the extreme desirability of performing as much of the optimization process as possible within the L.P. In this circumstance, this would mean that an attempt should be made to represent RED's attack on BLUE more directly within BLUE's budget optimization L.P.

Following this logic, an attempt was made to analyze possible routes for representation of RED's attack in BLUE's L.P. This analysis led to the concept of an "effective" defense in the form described below.

Analysis of the form of RED's attack on BLUE for a variety of BLUE force mixes indicated that the total level of RED's counterforce attack remained fairly stable. What changes as a function of the specific nature of BLUE's mix is the way RED uses his counterforce attack. (RED's use of his counterforce attack is oriented toward the objective of minimizing the effectiveness of the total surviving BLUE force.)

Following the above observations, it was recognized that one way to conceptualize an attack by RED on BLUE's bases is to view it as another kind of "defense." This defense is one that has a form something like a preferential defense in that RED can launch at the individual BLUE weapons in a manner to match his own objectives.

The current version of the program does include a preferential defense option (See Section H) so it was decided that a fruitful alternate approach to the effective cost method might be an effective defense method. This representation would be an adaptation of the true preferential defense logic with specialized inputs and controls.

By represenation of RED's counterforce attack as an equivalent defense, it is possible to analyze BLUE's budget optimization process in the following manner.

BLUE sets up his optimization L.P. using the actual weapon costs to insert a budget constraint. He then optimally allocates his budget to buy a force mixture which is optimal for the given target system, as defended by a preferential - like defense of special characteristics. This defense is represented by two characteristics. First, it consists of X number of effective intercepts, and second, there is a set of specified levels of intercepts to nullify each type of RV possessed by BLUE.

The number of effective intercepts represents the expected size of RED's counterforce attack while the intercepts required per RV represents the ability of RED to destroy a given BLUE target type. For example, say that in a previous iteration for some specific BLUE force mix RED attacked by TNR_i of his type i weapon. In addition, for the BLUE targets he attacked there were some survivors of each BLUE target type.

Then the appropriate equivalent defense parameters for such an attack would be:

$$X = \sum_{i=1}^{I_R} \frac{TWR_i \lambda_{Ri}}{\lambda_{R1}} \qquad \text{(J-6)}$$

Where:

X = effective RED counterforce size in terms of equivalent type 1 weapons

TWR_i = total RED counterforce size of type i weapons

λ_{Ri} = RED type i weapon lambda

I_R = total number of RED weapon types

Measurement of RED's counterforce attack in terms of one equivalent weapon type is necessary since the attack is likely mixed but the defense is restricted to only one type of interceptor. The weapon lambdas measure quite well the relative effectiveness of various weapons so they provide a very good equivalencing relationship.

The equivalent intercept requirements are as follows:

$$INT_j = \frac{\sum_{i=1}^{I_R} \frac{TN_{ji} \lambda_{Ri}}{\lambda_{R1}}}{DNB_j W_j} \qquad \text{(J-7)}$$

Where:

INT_j = number of equivalent intercepts to destroy one incoming RV of type j

TN_{ji} = total number of RED weapons of type i attacking targets of type j

DNB_j = total BLUE targets of type j destroyed in the attack represented by TN_{ji}

W_j = available RV's per BLUE base of type j

Equation (J-7) simply summarizes the average number of equivalent type 1 RED weapons required to destroy one BLUE RV.

If the previous BLUE force did not contain any targets of type j, an estimate for INT_j can be obtained simply by doing a minimum Lagrangian process on such a target to see how it would be attacked if BLUE did possess one in his arsenal.

Given that this equivalent defense method is used, how does it represent RED's attack more precisely than the effective cost method? The answer is that this method represents more nearly the fact that RED can be saturated by targets to the extent that he does not attack all targets of a given type.

For example, let's say that BLUE has two weapon types and that in his first force mix he bought N_1 of type 1 and zero of type 2. Then, he allowed RED to attack that mixture and found that he had S_1 survivors of type 1 and that the probability of survival of a type 2 weapon, if he had bought one, would be PS_2. Additionally, say that RED only attacked NA_1 of the N_1 targets with one weapon each and a resultant probability of survival for an individual target at PS_1.

The effective cost modifier method would use this information to develop the following parameters:

$$CM_1 = \frac{N_1}{(N_1 - NA_1) \quad NA_1 \; PS_1} = \frac{N_1}{S_1} \tag{J-8}$$

$$CM_2 = \frac{1}{PS_2} \tag{J-9}$$

Where:

CM = effective cost modifiers

In the same circumstance the effective defense method would utilize the three parameters as follows:

$$X = NA_1 \tag{J-10}$$

$$INT_1 = \frac{NA_1}{N_1 - S_1} = \frac{1}{1 - PS_1} \tag{J-11}$$

$$INT_2 = \frac{N_2}{(1 - PS_2)}$$

Where:

N_2 = expected number of attackers on each type 2 BLUE target

Note that this method carries the RED attackers, NA_1, as a separate parameter and also brings in the information that each type 2 target attracts an attack of N_2 RED weapons.

The net effect is that the effective defense method presents BLUE with more information on which to plan his budget optimization. Experience has shown that in the cost method the cost modifiers oscillate quite easily while the defense parameters described above are slowly changing. The result being a more stable process in the equivalent defense representation.

One result of this representation is that BLUE can more realistically consider the impact of total aim points on his surviving force mix. Cases have been observed where BLUE purchased certain mixed weapon types simply because the mixture itself caused problems to RED that individual types might not cause.

In contrast to the effective cost method, this procedure does have a reasonable chance to determine the minimum budget to meet a fixed objective. Such a utilization has not been emphasized but testing of the option resulted in a force procurement which very nearly met the fixed objective. Definitely convergence cannot be guaranteed here, however.

d. The Retargeting Problem

Given that the effective defense parameters are known the representation of RED's attack on BLUE utilizes the preferential defense logic described in Section H. However, the preferential defense is not precisely equivalent to RED's offensive attack because it does not give BLUE the opportunity of redirecting his survivors as he could if he possessed retargeting.

Consider the circumstance when BLUE does not have any retargeting capability. First, BLUE doesn't know which of his bases will be destroyed. Second, RED doesn't know where each BLUE weapon is targeted. Thus, at most, each side can only state a probability of any given weapon actually arriving at a specific target.

This probability can be approximated as

$$XPAQ_j = 1 - \frac{1}{CM_j} \qquad (J\text{-}13)$$

$$= \frac{\text{Number of Targets Destroyed of Type } j}{\text{Total Targets of Type } j}$$

Where:

CM_j = effective cost modifier for BLUE weapon type j

$XPAQ_j$ = probability that when RED decides to intercept a BLUE weapon, he actually destroys one headed for the target he wants to defend. (Assuming that he knows which weapon types will attack each target.)

Such a probability can be utilized in the effective defense method by making the defense probability of acquisition equal to XPAQ. In other words, represent this lack of knowledge about targeting as a kind of defense leakage.

When this is done, the equivalence to the case of no retargeting is reasonably close. (In many cases the impact of retargeting possession is quite small so that an optimum force mix would turn out to be the same with or without retargeting.)

The most successful representation of a full retargeting capability for BLUE has been to provide BLUE with extra RED value targets to match his loss because of RED's preferential defense. That is, BLUE designs a force to attack RED with the equivalent defense. After the attack, an inspection of which targets RED defended is made. Then BLUE is given a new opportunity to choose a force mix where he has extra target to match the loss of targets due to the equivalent defense. When the final convergence is reached, BLUE's forces which survive the equivalent defense have a number of targets to shoot at which exactly matches the original target structure.

Experience with the process has shown little change in the force mix as this variation in targets occurs. However, the total value destroyed obviously varies. If one is interested in determining the exact damage BLUE's survivors will do on RED, the process must be allowed to converge. Otherwise an underestimate of the value destroyed will occur.

One problem with the effective defense method is its interaction with a real preferential defense. In cases where the side being attacked does have a preferential defense, it is necessary to describe two separate kinds of defense. Currently, the program cannot handle such a case but work is progressing in the development of such a capability.

In practice it has been found that the effective cost method usually provides a good solution so it is the most commonly used procedure. However, in unusual cases, the effective defense method has been very valuable as a cross-checking procedure. Using one method to validate the other method thus has been a valuable feature of the two-method approach.

Additionally, concepts and procedures involved in the development of the defense method have been invaluable in improving the cost method. This factor alone has made the cost method much more practical than it would otherwise be.

3. Scenario Implementation

All the preceding discussion emphasized the budget optimization process in the simple two strike scenario because it demonstrates the survivability problem most directly. In the more complex three strike scenarios (Figures (3) and (4)), the discussion is still valid if the side striking second is the one performing the optimization. In the first strike option, however, there are some additional considerations.

a. A Counterforce First Strike

In the case of a counterforce first strike the program performs the function of optimally dividing the total force between first strike and the reserve. (See Section E.) This process is performed within the L.P. associated with the first strike and it uses as part of the process estimates of the probability of survival of a weapon placed in reserve and the estimated utility of that weapon in the third strike of the scenario.

In this budget optimization process the first strike is the natural L.P. to insert the budget constraint into. That strike essentially is in control of the critical optimizations for the side hitting first.

However, since the force mix is not known until after the first strike is conducted it is not possible to estimate the appropriate probability of survival in the special circumstance of your opponent possessing a system which counts the size of attack on himself.

This situation arises because of the technique used in optimizing the reserve force. The process is one that operates on the assumption that a candidate reserve force is analyzed and an estimate is made of the average probability of survival if a small change is made in the reserve force.

If the opponent counts the size of attack on himself, he then determines an average probability that each base is still occupied and attacks accordingly. This probability uses the ratio of attack size to force size so it is a function of the force mix. But the whole purpose of the first strike L.P. in the budget optimization problem is to determine the force mix. Thus, there is an incompatibility in that the force mix is a function of survivability but the best estimate of survivability involves knowing the force mix.

When the first strike opponent possesses no information about the attack on himself, or if he knows which weapons were launched at him, he attacks based on a constant estimate of base occupancy. Thus, the probability of survivability estimate for the first strike does not depend on the force mix details and no problems result.

Since the two extremes in assumptions do not cause any problems, it is felt that no effort should be made to develop a methodology for the case where attack level information is known. If a specific case is felt to be sensitive to the assumption, the two extremes could be run and the optimum force mix bounded by the result.

For the effective cost method implementation of the budget optimization into the first strike is straightforward. The normal budget constraint is inserted with the simple exception being that two cost units are involved. For weapons to be fired in the first strike, the directly input unit cost is used since they do not have to survive an attack. The usual survivability - modified cost is put into the L.P. for all weapons designated to go into reserve.

The program then buys a force mix and designates them either as reserve or first strike weapons. Given that the reserve is known, the opponent then gets an opportunity to optimize his attack. Then new estimates of reserve force survivability are made and the first strike is repeated, etc.

At the current time the effective defense method is not programmed for this situation since a few problems in the special process involved must be resolved.

b. <u>A Counterforce/Countervalue Strike With A Residual</u>

In the first strike of this scenario the L.P. does not conduct an optimization of the residual since it is a specified input. However, it does determine how to optimally split the first strike attack between counterforce and countervalue when there will be a final strike countervalue.

In this circumstance the first strike L.P. contains not only the real first strike, but also the planning of the third strike for the estimated residual survivors. Thus, all necessary information to conduct the budget optimization is available in the L.P.

Since the percent residual is specified the estimated attackers in the third strike is an easily computable percent (ignoring survivability) of the weapons bought for the first strike. In fact, the constraints on the third strike attackers can be written in terms of the number of first strike weapons.

The budget constraint for this situation charges no cost for the residual weapons but, instead charges any weapons purchased for a first strike the proportional amount to allow for the residual weapons. For example, if weapon type 1 costs C_1 and the residual is specified to be a decimal fraction of R_1, the budget constraint charges $C_1/(1-R_1)$ units for all first strike weapons of type 1 purchased. Then, the constraint on estimated third strike attackers is written as being $\leq R_1$ times the total number of type 1 weapons purchased.

By formulating the budget and weapon constraints this way there is correct allowance for the specified residual to occur and no residual weapons can be purchased without the corresponding first strike weapons having been purchased.

Survivability is factored into the process as follows. For the effective cost method, the third strike weapon constraints are simply modified to read $\leq R_1$ times the number of type 1 weapons purchased for the first strike times the estimated probability of survival of the residual weapons. The effective defense method is not currently programmed for this scenario.

c. <u>Problems With Random Area Defenses</u>

Due to the technique used to formulate the random area defense models, both bomber and missile, no budget optimization can be conducted against a target set possessing such defenses. The problem is simply that the random defense effectiveness is a function of the total number of attackers but in the budget optimization problem the total mix is a function of the effectiveness of each RV type. The situation typical of the problems with random defense as discussed in Chapter IV - Section G thus impacts here.

Additional effort is being made to overcome this problem. In certain circumstances the problem can be resolved and there is some hope that the total problem can be overcome. Currently, the impact of a random area defense would have to be approximated by an average, but constant reliability degradation.

4. Optimum Offense - Defense Allocations

At the present time there is an operating capability to optimally allocate a given budget among a list of offensive options and a list of defense types. That is, the defensive choices are restricted to one option for ABM area defense, one for ABM terminal defense, etc. The final result is an allocation of the defensive budget among the various types of defense, but no choice of the best candidate among several of one type.

This section will present a discussion of the method of formulation of the offense-defense allocation problem and the associated methodology.

The programmed process for dividing the budget between offense and defense builds upon the methodology for the offense allocation

BEST AVAILABLE COPY

as just described. The technique revolves around the concept of balancing the budget split in such a way that the offense and defense marginal utilities balance.

The process consists of iteration procedures which search for one way to divide up the budget so that one more unit of budget on any offensive or defensive option would result in the same return in pay-off. When this balance point is developed, the budget allocation problem is assumed to have been solved. (There are theoretical questions about local optima but these have not yet been addressed.)

a. The Offense-Defense Budget Split

After considerable experience with the optimizations involved in this program, it has become apparent that chances for convergence are greatly enhanced whenever the L.P. contains a maximum amount of the optimization process. If the L.P. can be provided with the bulk of the decision-making process, the likelihood of a convergence is simply greater.

This logic suggests that the proper place to divide a budget between offense and defense is within an L.P. Following such logic is difficult since the use of offense and defense occur in different strikes and each L.P. is basically a single strike representation. The connection between strikes does occur, however, in several instances, e.g., use of weapon lambdas for force target values, optimization of a reserve, multiple strike targeting and the effective defense method just described. Thus, there is some hope that the L.P. can be modified to connect offensive and defensive expenditures within one L.P.

There is one very obvious way to connect offensive and defensive expenditures. Say that BLUE is optimizing a budget as in Figure (1). He can start with an initial offense-defense division, say 50% to each. Then, in RED's first strike BLUE can use a special defense expenditure subroutine to proportion his defense budget optimally between area and terminal, bomber and missile defenses. Assume that BLUE obtained a marginal estimate of the utility of his defense budget in terms of the value saved by the last unit of budget spent. Then BLUE could use that information in his offensive attack by allocating budget to offense only as long as his marginal return to offense exceeds the marginal return to defense. Eventually the process could be continued until the two types of expenditure would be balanced in marginal utility.

The appropriate place to insert this marginal defense utility is in the pay-off row of the L.P. directly above the budget constraint slack variable. By inserting it at that location, there is a pay-off returned per unit of budget not expended on offense. Given the insertion is made in this way, the L.P. will automatically divide the budget with budget being spent on offense only as long as the return per dollar exceeds the predicted defense budget marginal utility.

This concept has been developed as the basis for the offense-defense budget splitting methodology. The methodology inherently assumes that there is one specific division of the budget where the value of the last offense dollar equals the value returned by the last defense dollar. When such a circumstance is located, it is assumed that the splitting problem has been solved.

The methodology proceeds as suggested previously. First, an arbitrary split of the budget is made. This split results in an estimate of the marginal utility of the defense dollars. Assuming that the defense marginal utility is going to remain constant, the next offensive L.P. for the budget optimizer is operated on so that dollars are spent for offense only if they bring more return than the estimated defense marginal return.

Once a new candidate offense-defense budget split is determined, the defense budget is allocated to all defense types by a methodology to be described in the next section. After this is done the opponent of the optimizer is allowed to optimally attack the newly purchased offense and defense mix. The result of the attack is a new estimate of the survivability of each offensive option and a new estimate of the marginal utility of the defense budget.

If the offense and defense marginal utilities balance, the process is stopped. If they do not balance, a new estimate of the balance value is made and that estimate is used in the offensive L.P. to generate a new budget split.

The process continues until the correct estimate of the balance point of marginal utilities is developed. The key to the process is the development of a procedure for rapid convergence to an estimate of the marginal utility balance point.

The problem can be visualized by consideration of Figure (J-2).

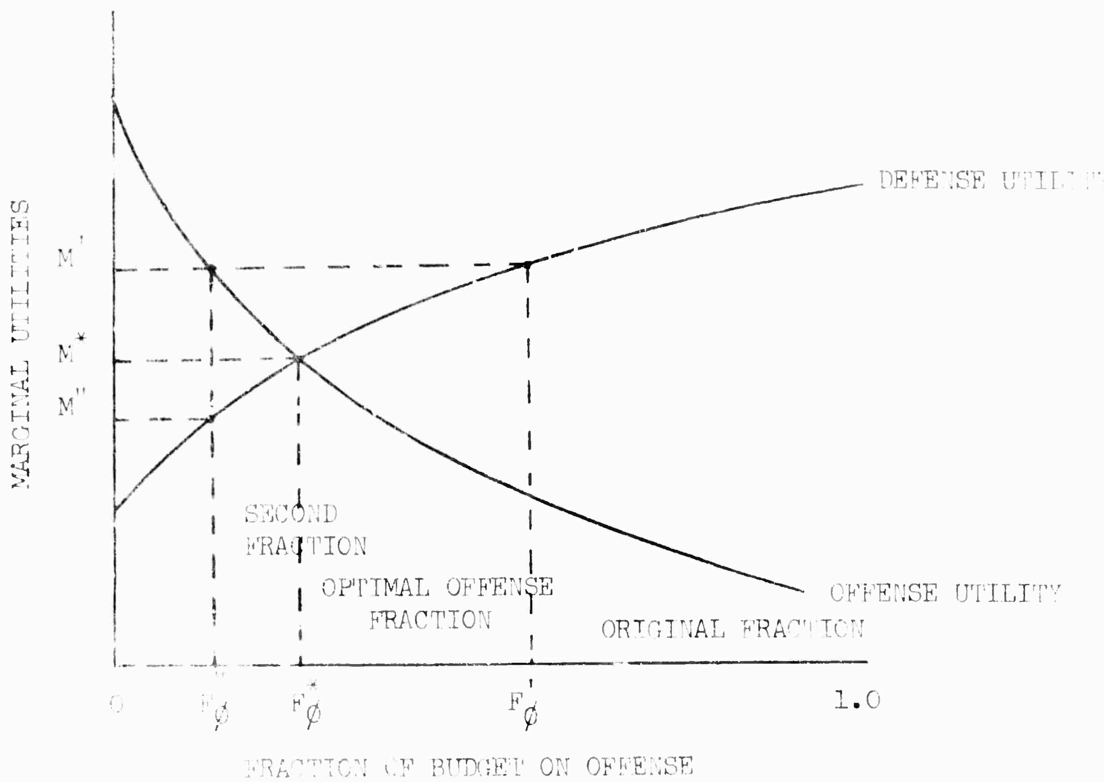


FIGURE (J-2): MARGINAL UTILITIES VS. THE BUDGET

The figure demonstrates an idealized interaction between offense and defense marginal utilities as a function of the fraction of the budget spent on offense. The goal is to find that utility, M^*, such that the marginal utilities of offense and defense are equal.

Indicated on the figure is the arbitrary starting fraction F'_ϕ and the defense utility M' that results. Using M' as an estimate for M^* will result in a second candidate fraction,

F''_ϕ that results in a new defense utility of M''. One could use M'' as the new estimate for the desired M^x and determine a new candidate fraction, etc.

However, the shapes of the two functions are not always as smooth as diagrammed and a more sophisticated convergence scheme is necessary. For any given case the shapes of the function can be irregular and the convergence procedure has to be adaptable to such irregularities.

In order to minimize computer running time the programmed procedure does not develop the functions of Figure (J-2) directly. Instead, the procedure involves a succession of estimates for M^x based on the history of all previous estimates and the current delta between M' and M''.

In actuality the two functions of Figure (J-2) are not fixed since they can fluctuate as a function of the detailed way a given budget split is allocated to offense and defense options. Figure (J-2) can be drawn only if one offense and one defense type are involved. As more options exist, the functions are more difficult to develop.

Because of these considerations the programmed procedure is the numerical, historical average procedure mentioned previously.

b. Distribution of a Defense Budget

For any given allocation of budget to defense it is necessary to have a sub-procedure for distributing the budget among all the defense types. Such a sub-procedure is the topic of this sub-section.

Because the offense-defense problem has been organized into sequential steps, namely:

a) Division of the budget

b) Distribution of the defense budget

c) Determination of the true marginal utility of the distribution from (b)

d) Re-division of the budget

it is hopefully possible to have an approximate procedure in step (b) that will yield reasonable results in step (d).

With the problem separated into stages like this each stage can be made to compensate for inadequacies of other stages, and each stage can be designed to take advantage of certain flexibilities.

For example, step (a) above requires only an estimate of the marginal utility that will balance the budget between offense and defense. (Thus, it is reduced to a one-dimensional optimization problem.) Then step (b) is given a budget and the only requirement in step (b) is to take the <u>relative</u> defense capability of each option into account when distributing that budget.

When step (b) arrives at an end product, step (c) then measures the result of that product in terms that can be used by step (a) in deciding how much total defense budget to have. The only problem in step (b) is that the defense option interactions must be taken into account.

If one conceptually takes into account the potential interactions of all defense types, he soon realizes that a hopeless mess could exist. Theoretically, depending upon the exact offense purchased, the various kinds of defense could interact in a very complex manner so that determination of a precise budget distribution would be nearly impossible.

However, if one backs off from the theoretical possibilities, there is a rational set of options. First, defenses against bombers might not have too strong an interaction with missile defenses since the two weapon types are quite independent. In the case of area and terminal ABM, the interaction is potentially stronger, but side analyses have indicated that if defense leakage is absent, it is usually optimal to buy either terminal or area defense, but not both.

If defense leakage does exist, the same analyses indicated that one defense type will be the best to spend budget on first and it is only as the budget climbs that a mixture then becomes valid.

Such considerations led to the idea of allocating the defense budget by use of defense pay-off characteristic functions. Typical such functions are indicated in Figure (J-3).

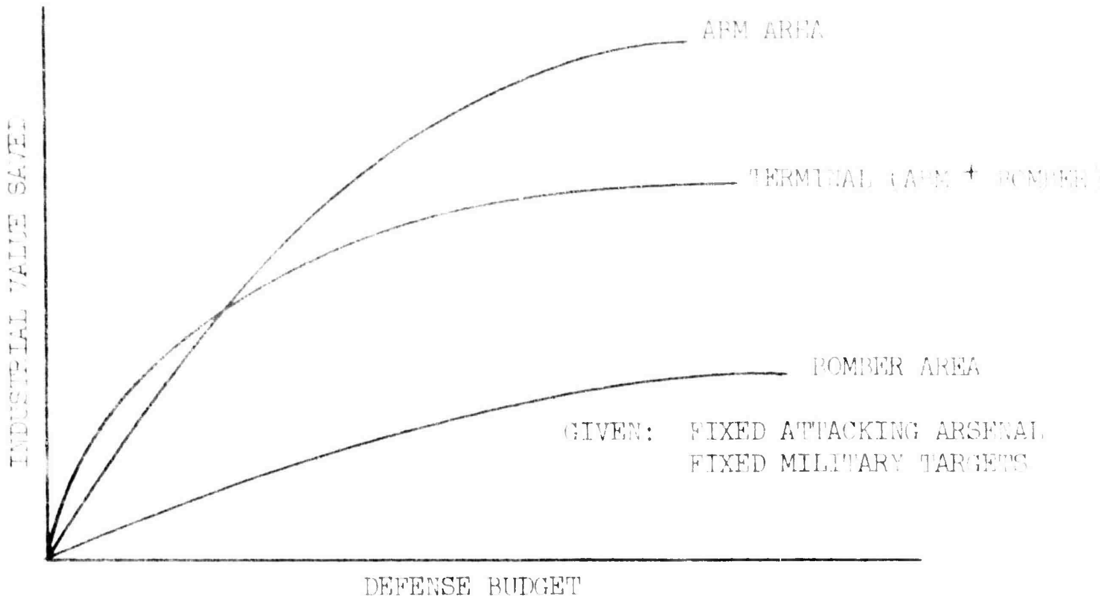


FIGURE (J-3): DEFENSE PAY-OFF CHARACTERISTIC FUNCTIONS

Each of these functions represents the industrial value saved as a function of the defense budget, when the total budget is spent only on the given defense type. (Note that there is only one terminal defense function. Section K discusses the fact that terminal defenses of all types can be balanced in one process to yield the appropriate function.)

The given functions exist for a specific choice of offensive bases for the offense budget and for a given attacking arsenal. If either of these factors changes, the functions obviously will change.

The next basic assumption is that as the offensive choices vary, the above functions remain reasonably fixed <u>with respect to each other</u>. That is, the total value saved can change but for a fixed budget spent on each defense type the ratio of pay-offs remains reasonably constant. This assumption has been borne out by some computational checks but no mathematical proofs exist.

Given the above assumption, it is feasible to assume that one can use these functions to distribute the defense budget by use of a marginal process. That is, in the above example, spend the money first on terminal defense because it yields the maximum initial slope, then buy some ABM area and, finally, if enough budget exists, buy some bomber area defense.

At each step in this process the concept is to spend money on the defense type which maximizes the marginal return per dollar. At the end of the budget distribution phase each defense marginal return should be as nearly equal as possible.

The success of this type of defense allocation depends mainly on the assumption that budget spent on one type of defense does not strongly effect the relative value of budget spent on another type of defense. If defense expenditures do in fact bring independent pay-offs, then this process is reasonable.

Several analyses of potential distributions of defense budgets were done by a combinatorial technique and compared to the results from this process. The result was a verification that this process did lead to optimal, or near optimal distributions. Again, only these experimental tests are available to validate the process.

Some degree of validation can be developed by comparing the predicted marginal defense utilities at the end of step (b) with those that are actually computed in step (c) of the procedure described above. This also has been done and, in fact, the results can be used to obtain a qualitative feeling for the adequacy of the process. In general, the results verify that the distributions are very reasonable.

This concept of validating the process by comparison of the "predicted" and "actual" marginal defense utilities resulted in a procedure for self-correcting any biases in the marginal utility prediction process. If, in making comparisons of the predicted and actual marginal utilities, there is a significant difference between them, the prediction process is modified so it would yield an accurate prediction for the case where the difference was detected. This should then improve the prediction process for similar cases.

This self-correction procedure operates as follows. Assume that at a given step in the budget allocation process the defense budget was allocated to each defense type in amounts

DB_1 = terminal defense budget

DB_2 = area ABM budget

DB_3 = area bomber budget

Then, the actual marginal defense utilities would be obtained in step (c) at this mixture. Finally, the defense pay-off functions (like in Figure J-3) used in the defense budget split process would be modified so that the slope of each characteristic function at levels of DB_1, DB_2, or DB_3 would be equal to the marginal utilities from step (c).

In summary, the defense budget distribution process utilizes the defense pay-off characteristic functions. These functions represent the pay-off for defense dollars spent on individual defense types but the assumption is that they also indicate the relative desirability of each defense type and thus are an ideal mechanism for the distribution process.

5. Result Validation

It is obvious that this whole process is designed to yield results in a reasonable length of time at the potential expense of a guarantee concerning result optimality. Accordingly, a number of validation procedures are available in the program.

Given that a budget optimization run is made, a tabular summary of the important steps in the process is printed. In addition, the best answer obtained is described in detail. Use of this information usually will resolve the question of doubts about the result validity.

Given that some questions do exist, there are several options for additional validation runs. They are as follows:

a) Choose several arbitrary budget distributions in the vicinity of the stated optimum and make normal runs with these choices. If another better allocation exists, one of the runs should indicate the possibility.

b) Make a restricted budget optimization where the dollars to offense [illegible] a specified amount. This will drive the all[illegible] process into another region of possibilities and might indicate a better optimum.

c) Make a restricted budget optimization where the dollars to offense and defense are specified, but where the distribution of the budget to individual options is made by the computer. This would cause the budget allocation process to have a simpler problem and thus might yield a better solution.

Each of these types of check-out are possible and they will normally resolve any questions concerning result validity.

6. Insertion of Biases or Preferences

The preceding discussion has indicated the current AEM procedure for searching out a "near-optimal" force mix when a candidate list of force components is available to choose from. This procedure has been built on the assumption that the final force mix should be that one which maximizes total damage, or delta damage, for a given budget.

Such a force mix is often of theoretical interest only. Since there are forces in-being and a political system which views with some doubt any attempt to radically alter that existing force, it is often true that a mathematical optimum is not an attainable option in the real world.

For example, in the circumstance where the U. S. has 1,000 Minutemen of certain capabilities, it would be very difficult to convince anyone that scrapping the current Minuteman force would be the thing to do in order to get a 3% improvement in cost-effectiveness. Uncertainties in cost and other features of new systems could very easily swamp such a small expected benefit.

However, if one could say that a new system would lead to a 350% improvement in cost-effectiveness, there might be some interest in such an idea - at least to the point of trying to confirm the 350%.

Considerations like the above, and countless others which could be presented, amount to the existence of biases toward certain individual weapon types or biases toward certain force mixes. That is, certain weapons should be chosen for the force unless other weapons are very clearly superior to them. Of course, the level of bias might vary from circumstance to circumstance.

Recognition of this bias ingredient to the budget allocation problem led to a task designed to develop techniques whereby the analyst could express such biases in a convenient and realistic manner. The resultant techniques that were implemented into AEM will now be discussed.

a. The Variety of Bias Forms

Some contemplation of the situations where biases do exist, or the situations where it would be realistic to express some biases, soon convinces one that no single bias factor approach is possible. There are too many diverse situations where the type of bias to be expressed is unique to that situation. Accordingly, the approach has been to implement a variety of the major bias insertion procedures so that the analyst would have a choice of procedures when a given situation arose.

A basic premise of the work done in this bias problem is that absolute biases which say - "weapon type X must be bought" - are not of primary interest. It is felt that such biases are of interest but there are a number of ways to insert such biases into the program without adding any new features. For example, the analyst could make the unit cost of system X = zero and then subtract all real costs for N units of system X from the budget. Then, by indicating that a maximum of N units could be purchased, all at zero cost, the final force mix would include the required N units of system X and the proper amount of real budget would be spent.

Consideration of possible bias forms first led to the basic type of bias that is as follows - "weapon type X must be bought, unless doing so would impose too high a degree of non-optimality into the final force mix." This type of conditional bias is of interest and insertion of such a conditional bias into the model was achieved.

One key aspect to the conditional bias problem is finding a means to express the allowed degree of non-optimality. A second key aspect is that of providing a means for expression of variations in biases from system to system.

The most logical expression of allowed degree of non-optimality seems to involve the classical notions of cost-effectiveness. That is, it seems natural for an analyst to think in terms of cost-effectiveness degradation. In such terms, he could visualize a 10%, or 20%, or whatever, impact on the value returned for his budget expended.

If bias factors can be related to cost-effectiveness types of impacts, it should at least be possible for the analyst to logically express the amount of bias he really feels. This relating of bias factors to cost-effectiveness numbers forms the basis of formulation of the bias expressions.

It should first be noted that in the circumstances of this budget optimization procedure it is always true that the budget is fixed. Thus, the objective is to maximize the return from the budget expenditure. In the case of constant budget, a degradation in cost-effectiveness is therefore equivalent to a degradation in pay-off or effectiveness.

Thus, one logical expression of the above type of bias toward system X might be as follows. "Weapon type X must be bought, unless there is some other system which would result in at least a Y % higher pay-off for the same budget spent."

Further consideration of possible situations where a bias would be natural leads quickly to the realization that an alternate expression of bias is often toward a force mix, and not necessarily toward individual systems. In such a case, the bias expression could well sound as follows - "Choose a force mix composed of N_1 units of system 1, N_2 units of system 2, N_3 units of system 3, etc. unless some other force mix would result in at least a Y % higher pay-off for the same budget spent."

For example, the preferred force mix might be the existing arsenal. In that case the N_i units of system i would be those in the existing arsenal. Then, the analyst might desire to stick with the existing arsenal unless a force mix could be found that was at least Y % better than the existing one.

Such a circumstance could be demonstrated as follows:

Arsenal Option	Value Destroyed
Existing Arsenal	915.7
Alternate Force Mix Number 1	972.0
Alternate Force Mix Number 2	1007.3
.	.
.	.
.	.
.	.
Alternate Force Mix Number 3	851.4

If such a circumstance existed, alternate force mix number 2 would be chosen if no bias toward the existing arsenal was expressed. However, if a bias of anything over 10% was expressed, the choice would be the existing arsenal.

Other situations could also be visualized for use of such a force mix bias. For example, a bias toward a balanced force of ICBM, SLBM and A/C components might exist. In such a case the analyst could prefer a balanced mix unless an unbalanced one provided at least Y % more damage than the balanced mix.

Further review of bias forms revealed that another form that does not fit into the above categories is as follows. Take the situation where some degree of change to a given arsenal is allowed, but where wholesale changes are not desirable. In such a circumstance, the bias might be expressible in the form - " spend at least 80% of the budget on systems X, Y, and Z." With this bias, it would be guaranteed that too large a change in force posture could occur.

A variation of this type of bias would be one where certain numbers of specific systems must be purchased, but where no upper limit would be expressed. This might be as follows - "purchase at least N_X of system X."

Variations of biases where the units are not dollars, e.g., "buy at least X units of yield," can always be utilized in AEM simply by viewing the budget as measured in those non-dollar units.

In summary, the insertion of biases into AEM budget optimization procedures has been approached in terms of allowing a variety of bias forms. This variety currently can be categorized as follows:

1) A bias toward specific individual systems.

2) A bias toward a specific force mix.

3) A control over the minimum purchase level of a specific force mix.

4) A control over the minimum purchase level of a specific individual system.

b. Implementation of Bias Forms

After reviewing the variety of biases desired, each of the basic four forms were implemented by appropriate modifications to the budget optimization process. This implementation took the following form.

b.1) Individual System Biases

Basically, a bias toward an individual system can be achieved by giving that system an LP pay-off advantage of some specified amount whenever the allocation process is considering whether or not the system should be purchased. This advantage works out very nicely in the L.P. procedure used in the weapon allocation process.

Consider the situation where a given strategy involves weapon type 4 and the analyst has indicated a 25% bias toward that system. In the implemented procedure a 25% bias means that, all other things being equal, weapon type 4 will be given credit in the L.P. for attainment of 25% more pay-off than would be attained by an otherwise identical weapon. Thus, when the LP chooses the optimal strategies, the one for weapon type 4 will be chosen unless a strategy by another unbiased weapon type achieves at least 25% more true pay-off for the same budget expenditure.

Note that this pay-off bonus of 25% is an artificial bonus that is used only to bias the LP toward weapon type 4. The true damage is that which actually can be achieved by the given weapon type.

In order to demonstrate the procedure more completely, consider its effect on the Lagrangian minimization process.

Equation (J-1) is the basic equation for deciding whether or not a given strategy will be chosen by the L.P. It says that: strategy X will lead to an improvement if the LP pay-off, P(X) is greater than $\sum_{i=1}^{M} \zeta_i' \cdot g_i(X)$, where $g_i(X)$ = level of constraint i called upon by strategy X and ζ_i' = constraint i multiplier from the last LP.

Now, in this case the pay-off, P(X), is multiplied by an appropriate bias factor, so that the equivalent of equation (J-3) turns out to be

$$\text{Maximize } H = V_j \prod_{i=1}^{I} B_i - V_j (1 - \lambda_K) \prod_{i=1}^{I} S_{ij}^{N_{ij}} B_i - \sum_{i=1}^{I} \lambda_i N_{ij} + \sum_{i=1}^{I} \lambda_m C_i N_{ij} \qquad \text{(J-14)}$$

where: all terms as defined for (J-3) except,

B_i = bias multiplier if $N_{ij} \neq 0$

= 1. if $N_{ij} = 0$

(Note that equation (J-3) calls for minimization of H while (J-14) calls for maximization.)

In the case of a 25% bias toward weapon type 4, we would have $B_4 = 1.25$ for any strategy involving that weapon type. It can be seen that such a B_4 multiplier would provide an advantage to weapon type 4 by virtue of increasing the pay-off for any strategy that uses that weapon type.

In summary, a specific weapon system bias is dealt with in the LP by increasing the LP pay-off for any strategy that utilizes certain weapon types. This artificial increase in the pay-off results in a modification to the Lagrangian search process as indicated by equation (J-14).

b.2) Force Mix Biases

The approach to biasing the LP toward certain force mixes, rather than specific weapons is similar to, but crucially different from the method just described. The basic similarity is that the LP pay-off is given some biased incentive to select the desired mix. The difference lies in the manner of computing and utilizing the incentive.

Consider the case where seven weapon types exist and a specific force mix of types 1 to 3 is preferred. In this circumstance, the analyst would like to indicate the mix he prefers and some measure of his level of preference.

If a simple multiplying factor, like B_i, were to be used, there would be no indication of the level of mix desired, only the weapon types involved in the preferred mix. Suppose, however, that a given mix was preferred unless it achieved ΔB units less pay-off than some other mix. Then, one could view the value of ΔB as the absolute level of bias toward the preferred mix. This value of ΔB

units could be divided up among the preferred mix weapons in such a way that purchase of the preferred mix would result in accumulation of the ΔB extra incentive points.

For example, take the case where 150 type 1, 215 type 3 and 55 type 8 were the preferred mix. Also suppose that the degree of preference was of an amount equal to 150 total pay-off units. That is, this mix should be chosen unless some other mix would accrue at least 150 more units of pay-off than the preferred mix. Then, it is possible to distribute the 150 incentive points over the preferred mix as follows:

$$\Delta b = \frac{150}{150 + 215 + 55} \qquad \text{(J-}\ldots$$

$$= .357$$

where: Δb = incentive, or bias, per warhead in the preferred mix

Addition of Δb units of bias in a strategy for each of the preferred mix weapons will cause a total accumulation of ΔB units of benefit when the exact numbers of weapons involved in the mix are purchased.

This constant bonus factor per RV can be implemented into the Lagrangian process very simply by recognizing that it is a linear term in the LP pay-off. It thus turns out that the equivalent of equation (J-14) for this case is:

$$\text{Maximize } H = V_j + \sum_{i=1}^{I} \Delta b_i N_{ij} - V_j (1 - \lambda_K) \prod_{i=1}^{I} S_{ij}^{N_{ij}}$$

$$- \sum_{i=1}^{I} \lambda_i N_{ij} \qquad \text{(J-16)}$$

where:

$$\Delta b_i = \frac{\Delta B}{\sum_{i=1}^{I} N_i^*}$$

ΔB = total force mix preference measure

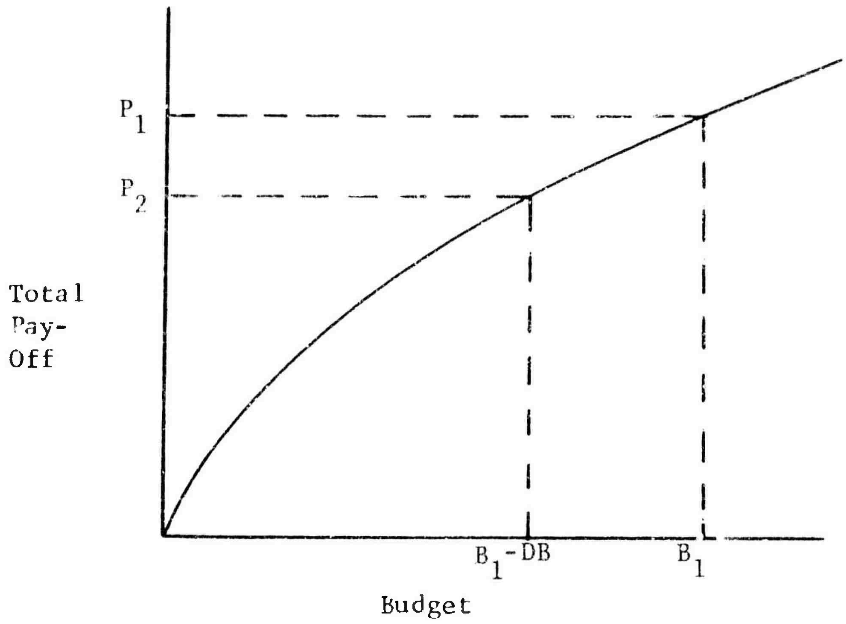
N_i^* = number of RV's of type i in the preferred mix

Δb_i = bonus per RV allotted only if weapon type i is in the preferred mix

The LP procedure is therefore very simply modified to augment each LP strategy by an amount of Δb_i for each preferred RV in that strategy. This, in turn impacts on the Lagrangian process as represented by equation (J-16).

It should be noted that this approach to bias adds a constant return for each preferred RV purchased while the method described in (b.1) added a constant proportional return for each preferred RV selected. Thus, the two approaches differ quite significantly.

Where would the analyst come up with the ΔB bias level input? One approach might be as follows. First, he could run a series of unbiased optimizations for various budget levels. This might result in the following kind of plot.



This plot shows the relationship between budget level and total force pay-off.

Now, suppose that B_1 budget units exist but that the optimal force for that budget is not desirable for some reason or another. Further, suppose that the analyst is willing to spend B_1 budget units inefficiently but only as long as the penalty is no larger than that resulting from a reduced budget of DB units. In other words, dropping the pay-off from P_1 to P_2 i acceptable because it would have about the same impact as having a budget of DB less units.

This would lead to a $\Delta b = P_1 - P_2$ and it would provide some rationale for choosing ΔB. Another rational would simply be to allow $\Delta B = f\ P_1$, where f is some fraction ≤ 1.

It should be noted that this procedure will not guarantee that the precise preferred mixture will be chosen over <u>any</u> other mix. For example, consider a mix which differs from the preferred mix only very slightly. Such a mix might accrue a significant fraction of the ΔB bonus points and, at the same time, gain significant true damage advantage by differing to a slight degree from the preferred mix.

This feature of the approach is not thought to be a weakness, but instead can be of some utility since it could indicate which components of the force mix are the most binding. For example, consider a hypothetical sequence where the bonus points are varied through a sequence of runs. This sequence might result in:

FORCE MIX CHOSEN BY WEAPON TYPE

ΔB	1	2	3	4	5	6
0	150	0	0	75	400	0
100	50	0	50	150	250	0
200	0	0	125	175	100	10
300	0	0	175	200	50	35
400	0	0	200	200	25	65
500	0	0	200	200		100

Note that the preferred mix of weapon types 3, 4 and 6 is gradually approached as the bias level is increased. More importantly, note that weapon type 5 is very strongly desired in the mix and that weapon type 6 is the one that forces ΔB to the highest level in order to create the preferred mix. Such a sequence can be used to generate preferred mix rankings.

b.3) A Minimum Acceptable Force Mix Purchase

It could happen that a bias exists toward certain forces but only to the degree that a large fraction, but not 100% of the budget, must be spent on those forces. Such a bias form was implemented by appropriate use of the auxiliary budget constraints currently allowed in AEM.

AEM allows insertion of up to four budget categories simultaneously. These four different constraints allow specification of several totally different budget conditions. It is through appropriate use of such an option that the minimum force purchase can be imposed.

If one views these auxiliary budget constraints as freely usable conditions, it is possible to convert a constraint of the form

$$\sum_{i=1}^{I} C_i N_i \leq B_k \qquad \text{(J-17)}$$

where: C_i = unit cost of weapon type i

N_i = number of units of type i purchased

B_k = budget type k amount

to one of the form

$$\sum_{i=1}^{I} C_i^* N_i \geq B_k^* \qquad \text{(J-18)}$$

where: C_i^* = special unit cost of weapon type where $C_i^* = C_i$ when type i is in the preferred mix and where $C_i^* = 0$ when is is not

B_k^* = minimum acceptable budget spent on the preferred weapon types.

This conversion is accomplished in two steps. First, the analyst must input the appropriate C_i^* values. Second, the LP must be modified to accept $\geq$ constraints in place of the $\leq$ variety.

This second step was recently accomplished in AEM so it was only necessary to appropriately modify internal bookkeeping procedures to exploit that option.

Appropriate use of the option presented in (J-18) can allow the analyst the ability to control the blend of the force mix in terms of old versus new, existing versus advanced, ICBM versus SLBM, etc. For example, use of the four different budget constraints could guide quite nicely the blend of the force mix.

b.4) <u>A Minimum Specific System Purchase</u>

It is obvious that the technique of (b.3) could be used to control the amount of purchase weapon by weapon type. However, the existence of only four optional budget conditions makes such an approach somewhat limited. It was felt that the most ideal method would be to implement the notion of hedges (See Chapter IV-P) in the context of budget optimizations.

Hedges can be utilized in AEM in a variety of ways and appropriate use of such hedges would offer very high utility in expression of biases. In this specific case, one hedge of the form: the number of weapons of type X fired at all targets must be GE N, would guarantee purchase of at least N units of system X.

In addition, other hedge forms, like value destroyed hedges and number of targets attacked hedges are possible and it would be very desirable to use those hedge forms in this bias procedure. Accordingly, appropriate internal program changes were made so that the full hedging option could be used with these bias options.

c. A Summary

Appropriate modifications have been made to AEM so that an analyst using the budget optimization procedure can exert realistic bias, or preference controls over the optimization process. Through appropriate use of such controls, it should be possible for the analyst to allow for non-mathematical considerations that a normal max pay-off criteria could never consider. A variety of options have been provided so that specific situations could be adapted to.

7. An Overview

In summary, [illegible] is the following:

1) Non-linearity [illegible] practical [illegible] method which guarantees convergence in all cases has been developed.

2) The most significant problem-causing factor is the requirement to purchase forces with proper allowance for their survival against attack. If only first-strike forces (Fig. 1) were to be purchased, convergence would be guaranteed.

3) If one is looking for near-optimal solutions, the process seems to be very adequate. The proper attitude is to view all results as a reasonable budget distribution which possibly should be validated by other processes.

4) Experience with the budget optimization process as currently programmed indicates that convergence to an optimum, or near optimum solution generally occurs and enough information is available to indicate when such does not exist.

5) There is currently no promise that a routine will be developed in the immediate future to determine the minimum budget to meet precise damage objectives. The whole emphasis has been on development of a process to optimally spend a given budget.

6) The allocation of a defense budget to specific defense types has been solved by a logical, but approximate, process which assumes that the interaction between defense types is weak.

Given the total picture as outlined above, it appears that a final product of some utility for the applications described in Part (1) is available. Most of those applications can achieve considerable benefit even if near-optimal results are achieved. Additionally, some relaxation of the constraint to be optimal for precise situations is possible in many cases and this helps the process considerably.

FORCE TGT STRATEGIES			VALUE TGT STRATEGIES			CONSTRAINT SLACK VARIABLE STRATEGIES	PAYOFF & CONSTRAINT COLUMN	
	-1.63			-140.05		LOCATION OF MULTIPLIERS AT COMPLETION		
	0			0		1	W1	WEAPON CONSTRAINTS
	1	TYPICAL STRATEGY		0	TYPICAL STRATEGY	1	W2	
	0			3		1	W3	
	0			0		1	W4	
	0			0		1	T1	TARGET CONSTRAINTS
	1			0		1	T2	
	0			0		1	T3	
	0			0		1	T4	
	0			0		1	T5	
	0			1		1	T6	
	0			0		1	T7	
	0			0		1	T8	
	0			+140.05		1	CVD	← VALUE TARGET DAMAGE
	0			0		1	RD	← RESERVE FORCE SIZE CONSTRAINT
	$1\,C_2$			$3\,C_3$		1	B	← BUDGET CONSTRAINT

NOTE: W_i = TOTAL ALLOWD WEAPONS OF TYPE i.

T_j = TOTAL ALLOWED TARGETS OF CLASS j.

CVD = MAX ALLOWED UNITS OF VALUE DESTROYED ON VALUE TARGETS.

RD = MAX ALLOWED RESERVE FORCE POTENTIAL.

B = TOTAL AVAILABLE BUDGET.

C_i = COST PER UNIT WEAPON OF TYPE i.

FIGURE J-1 TYPICAL L.P. FORMAT FOR RED IN FIGURE (1)

K. TERMINAL ABM DEFENSE - OPTIMAL DEPLOYMENT AND UTILIZATION

1. The Problem

The question of how best to employ terminal ABM interceptors is a resource allocation problem for the defense just as the weapon-to-target assignment is an allocation problem which the offense must solve to achieve full benefit from his resources. Some complication is introduced, however, by the fact that the defense acts first, chronologically, in deploying an ABM system, and because (given the present state of the art of intelligence collection), it is probable that the offense will be aware of the resulting deployment, and will reoptimize his plan in the light of that knowledge. These considerations lead to the following formulation of the defense's problem:

Choose a D which realizes

$$\min_{D} \left\{ \max_{A/D} \quad P(A,D) \right\} \qquad \text{(K-1)}$$

where the notation is heuristically intended to mean

A = what the attacker does

D = what the defender does

P(A,D) = pay-off to attacker if he selects action A and the defense selects action D.

It is evident that the merit of a particular defense action is measured by the resulting reduction in maximum attacker pay-off.

If the problem is cellular in nature - i.e. if (1) the attacker and defender each divide their resources among several "cells," (2) the pay-off in one cell is dependent only upon properties of the cell and the resources of each opponent committed there, and (3) the overall [illegible] of the pay-off [illegible] the [illegible] then the [illegible] is applied [illegible].

In the AHV, [illegible] individual target. The attacker chooses a vector $N_j = (N_j^1, \ldots, N_j^{NW})$ of resources to commit to the j^{th} target ($j = 1, \ldots, NT$). The components N_j^i are non-negative integer numbers of weapons of the i^{th} attacking type ($i = 1, \ldots, NW$). (The attack strategy vectors N_j are currently restricted to be "pure" – i.e., at most, one of the components is positive.)

The defense chooses a (non-negative, integer) scalar D_j = the number of terminal interceptors to allocate to the j^{th} target. The pay-off in the j^{th} cell is then

$$V_j \cdot P_{K_j}(N_j, D_j)$$

where V_j = value of the j^{th} target

and $P_{K_j}(N_j, D_j)$ = expected coverage at target j if N_j attackers and D_j defenders are committed there.

The overall pay-off is then

$$\sum_{j=1}^{NT} V_j \cdot P_{K_j}(N_j, D_j)$$

Defining $\mathcal{S}$ = set of all pure attack strategies,

W_i = total number of weapons of i^{th} attacking type,

and D_o = total number of interceptors, the AEM version* of formulation (K-1) of the defense problem is to

$$\text{choose } \left\{ D_j \right\} j = 1, \ldots, NT$$

$$\text{satisfying } \sum_{j=1}^{NT} D_j \leq D_o$$

$$\text{and minimizing } G(D_1, \ldots, D_{NT}), \text{ where} \qquad \text{(K-2)}$$

$$G(D_1, \ldots, D_{NT}) = \max_{\substack{N_1, \ldots, N_{NT} \\ N_j \in \mathcal{S}}} \sum_{j=1}^{NT} V_j \cdot P_{K_j}(N_j, D_j)$$

$$\text{subject to } \sum_{j=1}^{NT} N_j^i \leq W_i \quad (i = 1, \ldots, NW)$$

= maximum attacker pay-off for the specified defense allocation.

Due to the practical restrictions of computer storage and program running time, it was clear that the techniques of Section IV-B could not be used to produce the damage functions, $P_{K_j}(N_j, D_j)$, for every type of weapon and target and various levels of terminal defense. Thus, before a solution to (K-2) could be attempted, some other method of describing the impact of terminal defense on the basic weapon-target damage function was necessary. This has been done

---

* This formulation is not precisely compatible with the Arsenal Exchange Model in that target classes have been excluded from the discussion here. Their omission simplifies the explanation and notation. Any extensions necessary to treat the grouped target case are straightforward and will be indicated by footnote.

for the sequential strike cases (see p. IV-B-17 ff.) only, (so that it is currently not possible to optimize the deployment of terminal interceptors which are to be used in the NDJC < 0 tactical situation). It is felt that the resulting simpler damage functions are adequate for purposes of terminal defense deployment; however, the more accurate methods of Section IV-B are used to compute estimated and actual damage once the deployment is determined.

2. <u>Simplified Damage Functions</u>

An effort to obtain a simplified damage function was initiated by use of the method of Section IV-B through a parametric study of the impact of all the defense parameters on the exact damage function. This analysis demonstrated that the precise approximation of the exact damage function requires a procedure so complex that very little advantage is gained from the use of the approximation.

The problem, basically is that the AEM uses the two parameter function (T,P) to represent the damage function and the presence of defense can modify both of the parameters. The relationship between a precise set of defense parameters and both damage function parameters is so unique that very little generalization is possible.

Of the two parameters, the translation parameter, T, modification is the easiest to understand.

In fact, for a perfect defense, all of the defense impact centers in the translation parameter since modification of T can represent the situation where nothing penetrates until the defense is exhausted. It is only when leakage exists that the P parameter is also modified.

An additional factor is that the usage of a simplified damage function is visualized to be only for the purpose of deploying a set of interceptors and one would suspect that the optimal deployment would be somewhat insensitive to the precise damage function. Given the deployment, the exact damage function can then be computed and actual damage then computed.

Such considerations led to the concept of representation of the defense impact as a T parameter modification only. Such a representation will be exact if the defense is perfect and, hopefully, will result in near-optimum deployments if it is not perfect.

In the sequential strike cases (NDØC$\geq$0), first penetration of the defense is an event of particular interest (and utility in obtaining a T - only approximation) since it is assumed that the first penetrating warhead eliminates the remaining terminal defenses. On pp. IV-B-18 and 19, a fairly complicated expression is given for the probability that a specific warhead is the first penetrator. In view of this, it is remarkable that the expected point of first penetration has the following simple form (as may be verified by a lengthy derivation):

$$W^* = \frac{1}{R}\left[\,1 + d_1 + d_1 d_2 + \ldots + d_1 \ldots d_K\right] \qquad \text{(K-3)}$$

where:

R = reliability of the attack weapon

K = NWUDX (number of warheads until defense exhaustion)

d_j = probability that the defense stops the j^{th} incoming warhead.

In the notation [illegible],

$$d_j = p_A \left[1 - (1 - p_I)^{n_j} \right] \qquad \text{([illegible])}$$

where the n_j's are the interceptor assignments.

Example 1

If $ND\emptyset C > 0$, then

$$n_j = ND\emptyset C \qquad (j = 1, \ldots, K)$$

$$d_j = p_A \left[1 - (1 - p_I)^{ND\emptyset C} \right]$$

$$= \text{constant}$$

$$= d$$

$$W^* = \frac{1}{R} \left[1 + d + d^2 + \ldots + d^K \right]$$

$$= \frac{1 - d^{K+1}}{R\,(1 - d)} \qquad (d < 1)$$

Example 2

If the defense is perfect, then

$$p_A = p_I = 1 \quad \text{and} \quad n_j = ND\emptyset C = 1$$

$$d_j = d = 1$$

$$W^* = \frac{K + 1}{R}$$

Example 3

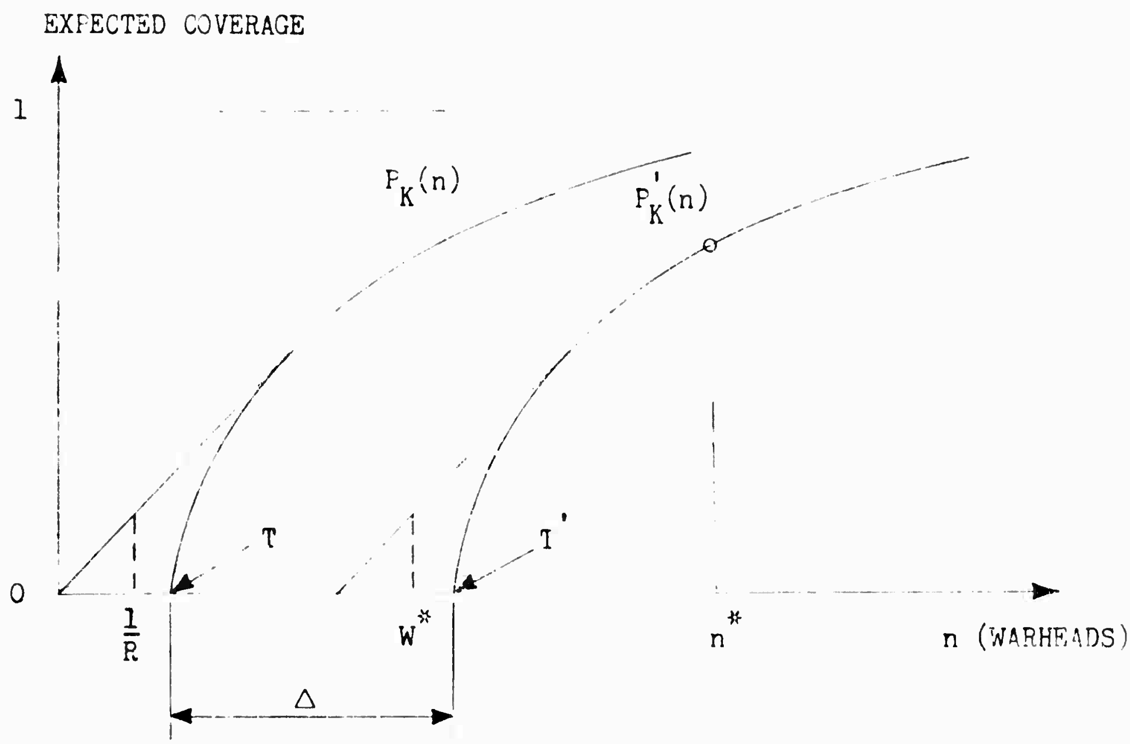
If there is no defense, then

$$d = 0 \quad \text{and} \quad W^* = \frac{1}{R}$$

The sketch below shows a typical damage function, $P_K(n)$, for an undefended target. If a terminal defense is introduced which is assumed to be vulnerable to the first penetrating warhead, then a reasonable approximation, $P_K'(n)$, to the real damage function is obtained by requiring that

$$P_K' (W^* + X) = P_K (\frac{1}{R} + X) \qquad \forall \; X \geq 0$$

- i.e., that the ordinates of the two functions agree beyond their respective points of expected first penetration.



The appropriate T-only modification is obtained by finding the T parameter shift, Δ, such that this result occurs. Thus, define

$$\Delta = W^* - \frac{1}{R}$$

and then the parameters of $P_K'(n)$ are

$p' = p$, and

$T' = T + \Delta$.

The Lagrange method ignores any cavities introduced into the damage function by the translation shift, so that if the target implied by he sketch is attacked at all, it will be attacked by at least n^* warheads of this type.

or any case where the defense fires a constant number of interceptors until he is exhausted, the defense impact can then be evaluated by computing a T parameter shift, Δ , by use of the W^* relationships presented in the above examples.

a. <u>Optimal Assignment Doctrine</u>

The preceding analysis leads to an interesting and useful by-product - namely, an optimum interceptor assignment doctrine ("optimum" in the sense of postponing first penetration as long as possible). That is, suppose the defense wants to choose K and $\{ n_j \}_{j = 1, \ldots, K}$ which maximize W^* subject to a stockpile constraint

$$\sum_{j=1}^{K} n_j \leq N_E$$

where

N_E = effective number of interceptors (see pp. IV-B-14).

Note that the following problem is equivalent:

$$\text{choose } K, \left\{ n_j \right\} \; j = 1, \ldots, K \quad \text{maximizing}$$

$$F(n_1, \ldots, n_K) = d_1 + d_1 d_2 + \ldots + d_1 \ldots d_K$$

$$\text{subject to } \sum_j n_j \leq N_E$$

$$\text{where } d_j = p_A \left[1 - (1 - p_I)^{n_j} \right] \qquad \text{(K-5)}$$

A non-integer solution to (K-5) in general is a point of tangency in K-space between the plane $\sum_j n_j = N_E$ and a contour surface of the function to be maximized

$$F(n_1, \ldots, n_K) = F_o.$$

However, no simple solution technique is available for the system of equations resulting from the tangency requirement. Therefore, the following alternative approach has been used successfully in approximating the solution to (K-5).

Consider what occurs as the stockpile parameter N_E is increased. At first all N_E interceptors are assigned to the first warhead seen by the defense, so that $K = 1$ and $n_1 = N_E$. Eventually, however, a value $N_E = N_{E_1}$ is attained where it is better for the defense to split his stockpile into $n_1 + n_2 = N_E$, where $n_1 > 0$ and $n_2 > 0$ for $N_E > N_{E_1}$. And, continuing to increase N_E, a sequence $\left\{ N_{E_K} \right\}$ $K = 1, 2, 3, \ldots$ is produced, where each

N_{E_K} [illegible] size at which the defense begins to develop a positive n_{K+1} in its optimum assignment.

At these boundary points, the K + 1 dimensional tangency requirement is satisfied by $n_1, \ldots, n_K > 0$ and $n_{K+1} = 0$, and the resulting system of equations is solvable for $n_1, \ldots, n_K$.

Thus the solution technique for a problem where N_E is specified is to bracket N_E by two values N_{E_K} and $N_{E_{K+1}}$ (such that $N_{E_K} < N_E < N_{E_{K+1}}$) and then integerize the resulting n_j's to obtain an optimum or near-optimum integer solution to (K-5). As will be shown later, this approach produces assignment doctrines for the terminal defense which appear to be superior to those produced by the modified Prim-Read methodology described in Appendix C. Thus, the damage functions produced in AEM when NDØC = 0 is specified, now assume this maximum "postponement" type of assignment doctrine for the defense.

The K + 1 dimensional tangency requirement (at the $n_{K+1} = 0$ boundary) ultimately results in the following system of equations, expressed in terms of the d_j's for simplicity:

$$(d_1 + d_1 d_2 + \ldots + d_1 \ldots d_K)\left(1 - \frac{d_1}{P_A}\right) = (d_1 \ldots d_K)\, d_1$$

$$(d_1 d_2 + \ldots + d_1 \ldots d_K)\left(1 - \frac{d_2}{P_A}\right) = (d_1 \ldots d_K)\, d_2$$

$$\vdots \qquad\qquad\qquad \vdots$$

$$(d_1 \ldots d_K)\left(1 - \frac{d_K}{P_A}\right) = (d_1 \ldots d_K)\, d_K$$

The $j^{\underline{th}}$ equation $(j = 1, \ldots, K-1)$ may be written as follows:

$$\frac{d_j}{1 - \frac{d_j}{p_A}} = \frac{d_1 \cdots d_j + \cdots + d_1 \cdots d_K}{d_1 \cdots d_K}$$

$$= \frac{1 + d_{j+1} + \cdots + d_{j+1} \cdots d_K}{d_{j+1} \cdots d_K}$$

$$= \frac{1}{d_{j+1} \cdots d_K} + \frac{\cancel{d_{j+1}}\,(1 + d_{j+2} + \cdots + d_{j+2} \cdots d_K)}{\cancel{d_{j+1}}\,(d_{j+2} \cdots d_K)}$$

$$= \frac{1}{d_{j+1} \cdots d_K} + \frac{d_{j+1}}{1 - \frac{d_{j+1}}{p_A}}$$

Thus, defining

$$a_j = \frac{d_j}{1 - \frac{d_j}{p_A}} \qquad \text{(K-6)}$$

we have the recursion

$$a_j = \frac{1}{d_{j+1} \cdots d_K} + a_{j+1} \qquad j = 1, \ldots, K-1$$

or equivalently

$$a_{j-1} = a_j + \frac{1}{d_j \cdot \cdot \cdot d_K} \qquad (j = 2, \ldots, K).$$

The last equation in the system is

$$a_K = 1$$

and (K-6) may be rearranged to read

$$d_j = \frac{a_j \, p_A}{a_j + p_A}$$

Thus, the following simple recursive solution is obtained for any value of K:

$$a_K = 1$$

$$d_j = \frac{a_j \, p_A}{a_j + p_A} \qquad \text{(K-7)}$$

$$a_{j-1} = a_j + \frac{1}{d_j \cdot \cdot \cdot d_K}$$

The n_j's are then produced by inverting (K-4):

$$n_j = \frac{\ln\left(1 - \frac{d_j}{p_A}\right)}{\ln\left(1 - p_I\right)} \qquad j = K, K-1, \ldots, 1 \qquad \text{(K-8)}$$

Table K-1 shows the first few optimum boundary assignments resulting when (K-7) and (K-8) are applied for the case $p_A = p_I = .9$.

TABLE K-1 OPTIMAL INTERCEPTOR ASSIGNMENTS

N_{E_K}	K	n_j
.32	1	.32
.97	2	.65
1.87	3	.89
2.95	4	1.08
4.19	5	1.24
5.56	6	1.37
7.04	7	1.48
8.63	8	1.59
10.31	9	1.68
12.08	10	1.77

The first column gives the stockpile sizes for which the assignments are exactly optimum. The second gives the number of attacking warheads to which interceptors are assigned. The last column lists the non-integer interceptors to be assigned to successive warheads seen by the defense, in reverse order. For example, if the stockpile size is 4.19, the optimum assignments are $n_1 = 1.24$, $n_2 = 1.08$, $n_3 = .89$, $n_4 = .65$ and $n_5 = .32$, in non-integers.

Intuitively, the best integerization of this doctrine for a stockpile size of 4 interceptors would be $n_1 = n_2 = n_3 = n_4 = 1$. However, another possibility is $n_1 = 2$, $n_2 = n_3 = 1$.

To compare the two, compute

$$d(1) = .9\left[1 - (1 - .9)^1\right] = .81$$

$$d(2) = .9\left[1 - (1 - .9)^2\right] = .891$$

and $F(1, 1, 1, 1) = .81 + (.81)^2 + (.81)^3 + (.81)^4$

$$= 2.43$$

$$F(2, 1, 1) = .891 + .891\,(.81) + .891\,(.81)^2$$

$$= 2.20$$

Thus the first doctrine is clearly optimum.

Subroutine CØMPK generates the continuous n_j by the above recursion relation, and accumulates them to produce the N_{E_K} values. As soon as the specified stockpile is exceeded, the corresponding n_j's are rounded to the nearest integer (except that zero is not allowed), the i[illegible] sum is matched to the specified value of N_E, and K is rea[illegible]sted to agree with the integer solution.

To illustrate the advantages of this process over the modified Prim-Read approach, consider the example given on pp. IV-B-34 with NDØC = 0. The Prim-Read assignment of the $N_E = 12$ interceptors was $n_1 = 2$, $n_2 = \ldots = n_{11} = 1$. The postponement doctrine resulting from the above recursion and integerization is $n_1 = n_2 = n_3 = 2$, $n_4 = \ldots = n_9 = 1$. A comparison of the resulting damage functions is given in Figure K-1.

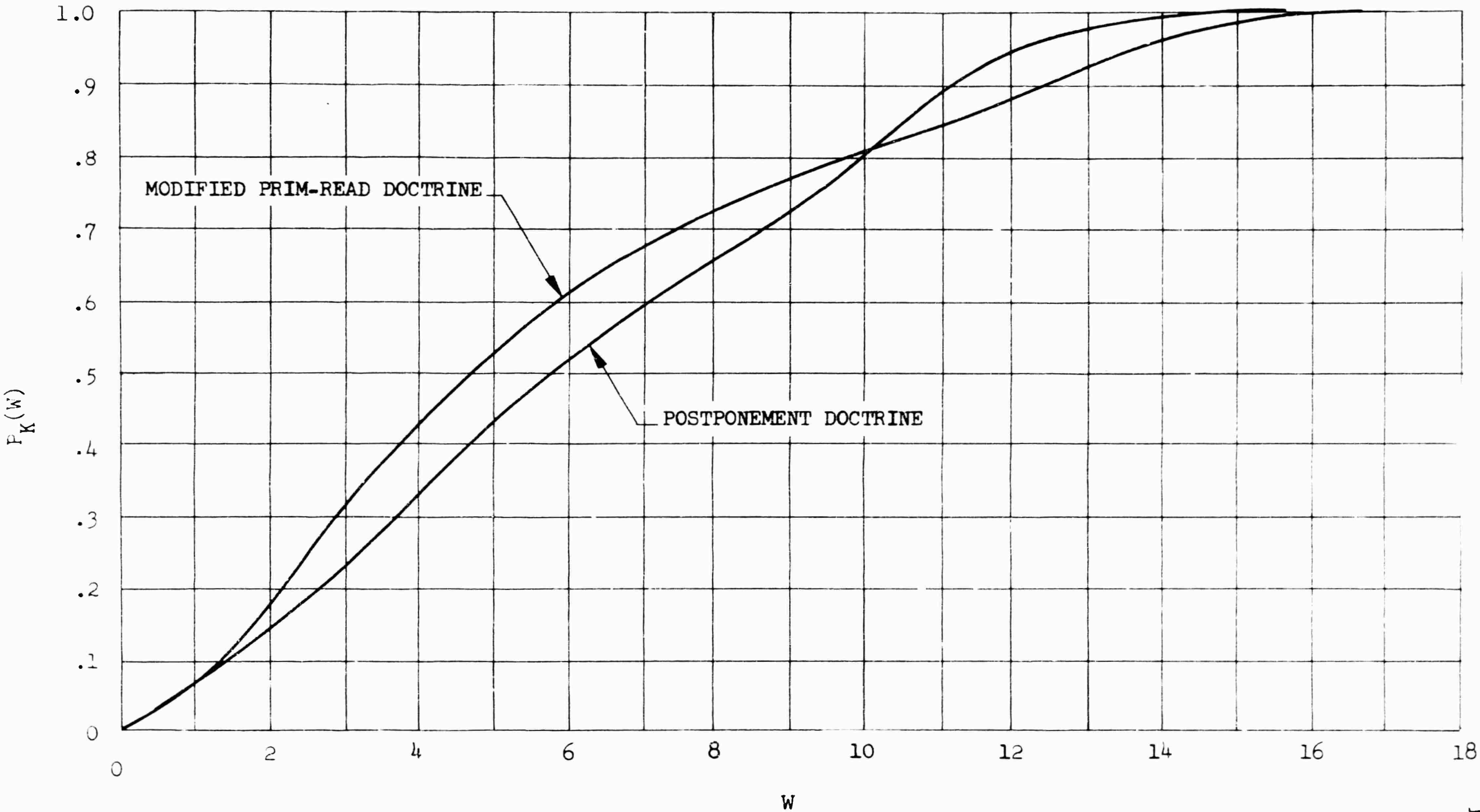


FIGURE K-1 COMPARISON OF INTERCEPTOR ASSIGNMENT DOCTRINES

b. Simplified Damage Functions - Optimal Assignment Case

The recursion (K-7) is not in itself a satisfactory tool for quickly approximating the effect of any given number (N_E) of optimally assigned terminal interceptors on the basic damage function. However, it was instrumental in producing such a tool.

Observe that for fixed values of p_A and K, equation (K-7) may be used to produce the d_j values (j = 1, . . . , K). Without knowing the assignment $n_1, \ldots, n_K$, the function $F(n_1, \ldots, n_K)$ is then computable via the relation

$$F = d_1 + d_1 d_2 + \ldots + d_1 \ldots d_K.$$

One is thus inclined to regard F as a function of p_A and K only, and indeed Figure K-2 gives a plot of $F(K,p_A)$.

Also, from equation (K-8), notice that

$$n_j \ln (1 - p_I) = \ln (1 - \frac{d_j}{p_A})$$

Therefore

$$\sum_{j=1}^{K} n_j \ln (1 - p_I) = \sum_{j=1}^{K} \ln (1 - \frac{d_j}{p_A})$$

$$\text{or } N_{E_K} \ln (1 - p_I) = \sum_{j=1}^{K} \ln (1 - \frac{d_j}{p_A}) \qquad \text{(K-9)}$$

We thus define

$$Q(K, p_A) = \sum_{j=1}^{K} \ln (1 - \frac{d_j}{p_A})$$

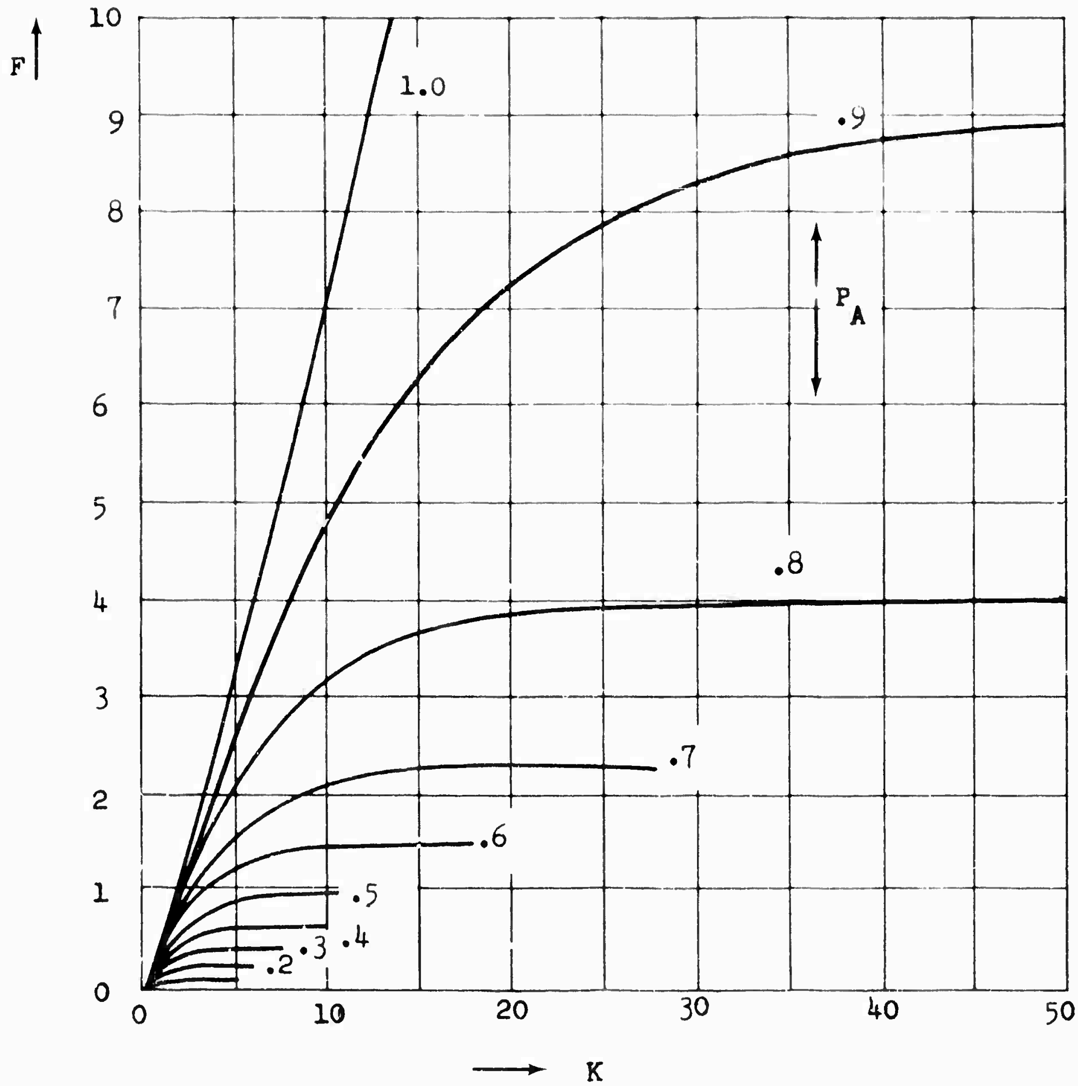


FIGURE K-2 F - FUNCTION

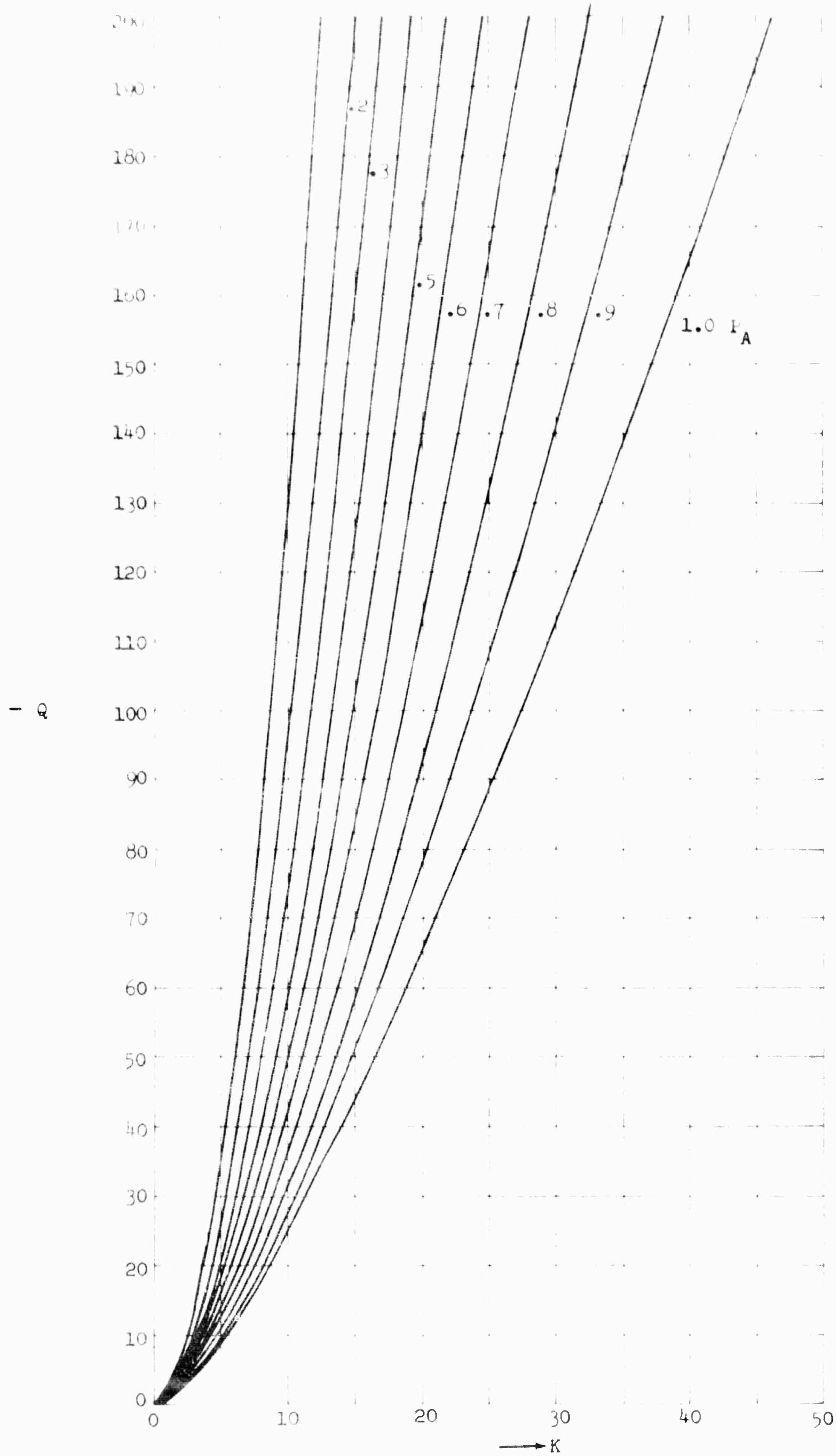


FIGURE K-2 Q - FUNCTION

and, like F, compute its values directly from the recursion. A plot is given in Figure K-3.

The purpose of all this is as follows. The approach will be to first approximate the functions $F(K, p_A)$ and $Q(K, p_A)$ analytically. Then, for arbitrary specified values of p_A, p_I and N_E, in view of equation (K-9), find the (not necessarily integer) value K^* satisfying

$$K^* = Q^{-1} (N_E \ln (1 - p_I),\ p_A).$$

Finally, $F(K^*, p_A)$ is evaluated to produce the numerator of the desired translation shift

$$\Lambda = W^* - \frac{1}{R}$$

$$= \frac{1 + F(n_1, \ldots, n_K)}{R} - \frac{1}{R}$$

$$= \frac{F(n_1, \ldots, n_K)}{R}.$$

This gives the approximate impact on the undefended target damage function of N_E optimally assigned terminal interceptors. Note that use of the procedure described above does not involve evaluation of the recursion, solving for the assignments (n_j), or interpolating between solutions at adjacent boundary points $(N_{E_K}$ and $N_{E_{K+1}})$, and thus represents a considerable economy of effort in the computations.

The following expressions were produced by a tedious fitting process to data generated from the recursion (K-7), and have proven adequate as approximations to the functions F and Q:

[illegible]

where [illegible]

and $[illegible] \approx -K^{C}$ (K-12)

where $$C = 1.38\left[1 - (1 - p_A)^{1.3865} + (1 - p_A)\, p_A^{-\frac{.3865}{1.3865}}\right] \quad \text{(K-13)}$$

Thus, from (K-12):

$$K^* \approx \left[-Q(K^*, p_A)\right]^{\frac{1}{C}}$$

or $$K^* \approx \left[-N_E \ln(1 - p_I)\right]^{\frac{1}{C}} \quad \text{(K-14)}$$

As outlined above, the procedure is to apply (K-14), and then (K-10).

For example, let $p_A = .9$, $p_I = .8$, $N_E = 30$. Thus,

$$-N_E \ln(1 - p_I) = 48.3$$

$$b = .036$$

$$c = 1.467$$

$$K^* = 14.1$$

$$F = [illegible]$$

With detailed grids, Figures (K-2) and (K-3) yield the more accurate results

$$K^* = 14.6$$

$$F = 6.2$$

Notice from equation (K-4) that as K and $n_j \rightarrow \infty$, $d_j \rightarrow p_A$ and thus

$$F \rightarrow p_A + p_A^{\ 2} + p_A^{\ 3} + \ldots$$

$$= \frac{p_A}{1 - p_A}$$

It is clear that the approximation (K-10) for F has the same asymptote. Also,

$$W^* \rightarrow \frac{1 + \frac{p_A}{1 - p_A}}{R} = \frac{1}{R(1 - p_A)}$$

which is the same limit as that of the constant assignment doctrine expression given in Example 1 of the section entitled "Simplified Damage Functions."

3. Optimum Deployment

Having designed suitable approximations to the functions $P_{K_j}(N_j, D_j)$ for sequential strike cases, it is meaningful to discuss the solution to the defense's deployment problem as formulated in (K-2). In Reference (11), Pugh suggests a Lagrange multiplier approach solving the following unconstrained minimaximization problem as a means of approximating the solution to (K-2):

Find multipliers $\hat{\mu} \geq 0$ and $\hat{\lambda}_i \geq 0$ $(i = 1, \ldots, NW)$ and choose

$\left\{ \hat{D}_j \right\}$ $j = 1, \ldots, NT$ and $\left\{ \hat{N}_j \right\}$ $j = 1, \ldots, NT$ achieving

$$\min_{\{D_j\}} \left(\max_{\{N_j\}} \left[\sum_{j=1}^{NT} V_j P_{K_j} (N_j, D_j) - \sum_{i=1}^{NW} \hat{\lambda}_i \sum_{j=1}^{NT} N_j^i \right] + \hat{\mu} \sum_{j=1}^{NT} D_j \right)$$

such that

$$\sum_{j=1}^{NT} \hat{N}_j^i = W_i \qquad (i = 1, \ldots, NW)$$

and $$\sum_{j=1}^{NT} \hat{D}_j = D_o$$

In Reference (11), it is pointed out that this approach produces a defense option minimizing the maximum "profit" available to the attacker within the defense's resource constraint - that is, a deployment $\left\{ \hat{D}_j \right\}$ is found which solves

minimize $F(D_1, \ldots, D_{NT}) =$

$$\max_{\{N_j\}} \left[\sum_{j=1}^{NT} V_j P_{K_j} (N_j, D_j) - \sum_{i=1}^{NW} \hat{\lambda}_i \sum_{j=1}^{NT} N_j^i \right] \qquad \text{(K-15)}$$

subject to $\sum_{j=1}^{NT} D_j \leq D_o$.

The deployments produced in AEM result from an attempt to solve (K-15) directly, without explicit use of a Lagrange multiplier, μ, for the defense.

Note that the maximum profit function may be written as follows:

$$P(D_1, \ldots, D_{NT}) = \sum_{j=1}^{NT} \max_{N_j} \left[V_j \cdot P_{K_j}(N_j, D_j) - \sum_{i=1}^{NW} \hat{\lambda}_i N_j^i \right]$$

$$= \sum_{j=1}^{NT} p_j(D_j)$$

where

$p_j(D_j)$ = maximum profit available to the attacker at target j

$$= \max_{N_j} \left[V_j \cdot P_{K_j}(N_j, D_j) - \sum_{i=1}^{NW} \hat{\lambda}_i N_j^i \right]$$

Thus, at every target there is a maximum attacker profit function of the form

$$p(D) = \max_{i} \max_{N_i} \left[V \cdot P_K(N_i, D) - \hat{\lambda}_i N_i \right]$$

where N_i has been redefined to mean the number of attacking weapons of type i. Since

$$P_K(N_i, D) \approx 1 - S_i^{N_i - T_i - \Delta_i(D)}$$

with the function $\Delta_i(D)$ as derived above,

$$p(D) = \max \max \left[[illegible] S_i^{N_i - T_i - \Delta_i (D)} [illegible] \hat{\lambda}_i N_i \right]$$

$$[illegible] \left[[illegible] S_i^{[illegible] T_i \, \Delta_i [illegible]} + \hat{\lambda}_i N_i \right]$$

Therefore, [illegible]

$$h(D) = \min_{N_i} \left[S_i^{N_i - T_i - \Delta_i (D)} + \lambda_i N_i \right] \qquad \text{(K-16)}$$

= minimum attacker "loss" function at any target

and it is clear that the defense's goal of minimizing overall attacker profit is the same as maximizing overall attacker loss.

Thus, $h_j (D_j)$ may be regarded as the defense's pay-off function at target j, and is computable for integer values of D_j via equation (K-16). The $h_j (D_j)$ have a generally concave shape, as is illustrated in the sketch below.

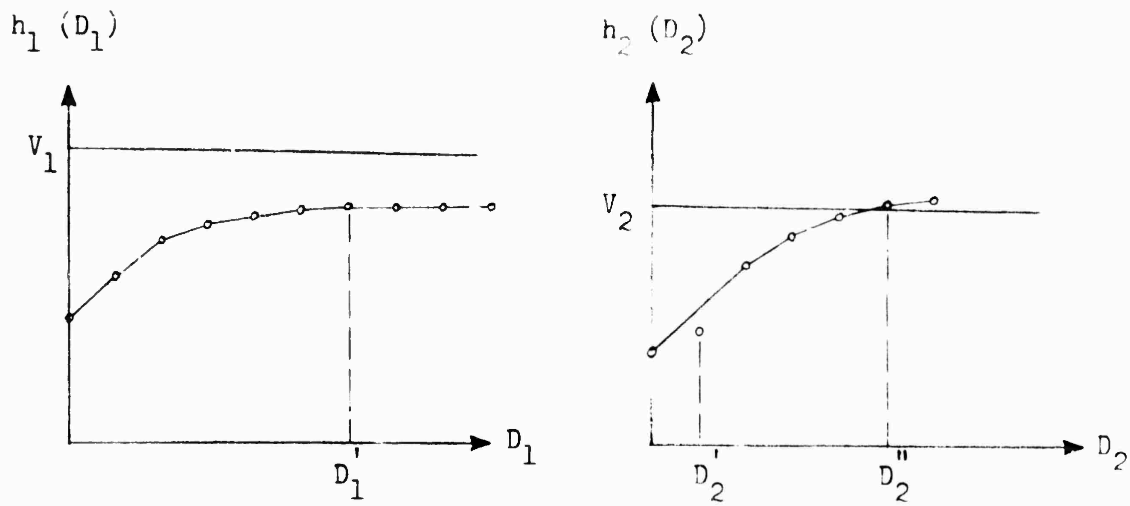
Since only one type of resource is involved, the defense can sequentially allocate his interceptors to targets at which the maximum benefit/interceptor is realized and be assured that, when his stockpile is depleted,*

$$\sum_j D_j \leq D_o \qquad \max \left[\sum_j h_j (D_j) \right]$$

will have been achieved.

---

* In the grouped target case, the incremental deployments are made simultaneously for all like targets. If stockpile exhaustion occurs mid-way through a target class, it is split into two classes, one at the higher level of deployment and one at the lower.



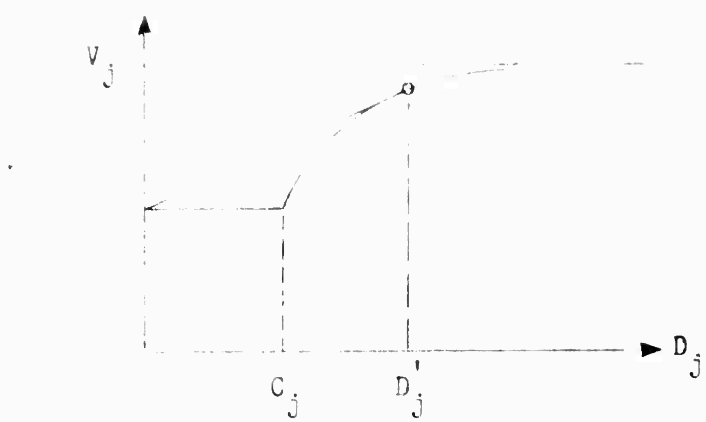
The sketch illustrates some of the phenomena which can occur. D_1' is a point at which the optimum attack on target 1 switches from missiles to bombers. As the slope criteria indicates, no further benefit accrues to the defense by adding (ABM) interceptors.

The pay off h_2 (D_2) has a slight cavity at D_2'. Therefore the program has a look-ahead feature which isolates cavities like this and will deploy around them.

When the attacker loss exceeds the value of the target, as at D_2'', the interpretation is that the defense level is sufficient to preclude any attack whatsoever by the offense. In this situation no further addition of interceptors is warranted or permitted.

The foregoing discussion has assumed that a defense stockpile consisted solely of interceptors. A more realistic view is that there are certain fixed installation costs which enter the deployment problem - i.e., radars, computers, site-preparation costs, etc. must be taken into account before a single interceptor is deployed.

If these costs can be lumped into an equivalent cost in interceptors, C_j, then the effect on the deployment methodology is as follows:



That is, the defense receives no benefit until he has paid the price, C_j, (in interceptors) of the defense installation. Because of the look-ahead feature of the program which skips over cavities, the first potential deployment increment available to the defense at this target is then D'_j.

Also, it is sometimes desirable to deploy interceptors in batteries, or in incremental steps other than unity. There is a program input which accomplishes this, if desired, with no essential change to the above theory.

Finally, in order to solve (K-15) completely, it is necessary to have attacking weapon lambdas ($\hat{\lambda}_i$) which are optimum in the sense that they must be compatible with an attack against the optimum deployment sought. Since these are not known at the start of the solution process, an iteration is performed. Starting with any deployment, a strike is optimized against the defense, producing λ_i's as a by-product. These determine a new deployment, another strike is conducted, new λ_i's are produced, and so on. This process converges to reasonable defense deployments, but sufficient evidence exists to indicate there is no guarantee of optimality. As Pugh, Reference (11), indicates, the double Lagrangian method is not guaranteed to result in optimal solutions. The method described here is directly comparable to the double Lagrangian so it too is not guaranteed.

In the process employed in this program, special steps have been taken to encourage convergence. For example, special "effective" weapon lambdas are sometimes used in order to help direct the defense to a deployment which provides the offense with a constant return per weapon.

However, usage of the deployment routine has shown little interest in having deployments optimal for some precise attack condition. Rather, one usually desires a deployment which is reasonable for a general attack condition. Such deployments do result from the process described above.

4. Balancing of Missile and Bomber Defenses

As discussed in the sketch of the previous subsection, a target can be defended against missiles to such an extent that the target will then be attacked by bombers. It is then necessary to add bomber terminal defense before continuing with more missile defense.

There is an option in this subroutine to detect that such a condition exists and it will appropriately add a balanced defense increment to the target (given a cost for bomber terminal defense). The increment is defined as a level of missile defense and a

[illegible]

[illegible] rder [illegible] the same delta [illegible].

[illegible] defense [illegible] bomber plus missile) [illegible] delta [illegible]. [illegible] and delta can [illegible] defenses [illegible] target represent a preferred [illegible] another target.

5. Deployment of Dedicated Area Defenses

Within the context of the Arsenal Exchange Model, dedicated area defenders is the name attached to long-range interceptors which per[illegible] intercept function against exo-atmospheric objects, but only on those objects directed at some specific terminal target. Thus, they could appropriately be considered the same as long-range terminal interceptors. Their function is usually viewed simply as a defense barrier which is far enough away from a target so that short-range interceptors can be directed at objects which survived the long-range intercept attempts. In addition, they might have an advantage because of the fact that intercepts would occur outside the atmosphere.

In this section we will describe deployment procedures which can take a fixed number of such interceptors and optimally distribute them to specific targets so that the maximum benefit is achieved when a given arsenal is attacking a target set that includes the defended targets. An implied assumption in the methodology will be that the attacker knows the exact deployment outcome. This can be visualized as a two-move game where the defender moves first.

a. <u>The Methodology</u>

A comprehensive methodology to deploy short-range terminal defenders has just been described. It seemed most appropriate to adapt that methodology to this task to as large a degree as follows.

The problem to be addressed here basically is as follows. A defender possesses some number of dedicated area interceptors to deploy. He expects an attack by an arsenal of some specified characteristics after he has completed his deployment. There is a set of targets (T) of varying values and characteristics which are candidates for receipt of the interceptors. The defenders' objective is to distribute, or deploy the interceptors to the individual targets in such a way that the resultant decrease in total damage to all targets is maximized.

This can be expressed (as in K-1) in mathematical notation as follows:

Choose a D which realizes

$$\underset{D}{\text{Min}} \left\{ \underset{A/D}{\text{Max}} \quad P\ (A,\ D) \right\} \qquad \text{(K-17)}$$

where the notation is intended to mean

A = the action chosen by the attacker

D = the deployment plan

P (A, D) = pay-off to the attacker if he selects action A after deployment D is chosen.

The term inside the bracket implies that the attacker wants to maximize damage while the Min D outside the bracket indicates that the defender wants to minimize the maximum possible attacker pay-off.

There are some inherent problems in attaining the global optimum strategy, D^*. These problems were addressed earlier and will not be re-stated here. It is important only to understand that the methodology utilized in AEM always produces "good" deployments, usually "near-optimum" deployments are achieved, but optimality cannot be guaranteed.

The basic deployment concept previously described in mathematical form can also be visualized as follows. First, generate an attack plan on the targets in their "before-deployment" defense condition. Then, evaluate the net marginal pay-off to the attacker at each of the targets being attacked. This will produce a result like that in column two of the following table. This net marginal pay-off at target i is the loss to the attacker if target i were removed from his attack option list.

TABLE K-2 ATTACKER NET PAY-OFF BY TARGET BY DEFENSE LEVEL

Target	Net Pay-Off To Attacker At Deployment Level					
	None	1	2	3	4	5
1	60	56	52	48	44	40
2	110	110	110	110	110	110
3	87	80	74	70	67	64
4	42	38	35	35	35	35
5	15	11	5	0	0	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
T	22	18	14	12	10	9

Then compute a similar net pay-off for each target as if one defender was added to that target. (This being done as if all other target deployments were held const[illegible] at the starting level.)

This would result in the data presented in column three of Table K-2. A similar procedure might then be conducted in order to arrive at the complete table.

Several [illegible] important to note about the table. First, the pay-offs [illegible] attacker should be decreasing at a rate which itself is decreasing. Second, some targets (like target 5) might not be worth deploying to beyond some level where the attacker would stop his attack. Third, some targets (like target 2) might not show any benefit from deployment if, for example, the target is being attacked by bombers and the defense is only good against missiles.

Once such a table has been constructed it is possible to develop a candidate deployment simply by putting the first interceptor on the target where the biggest decrease in attacker pay-off occurs, the second interceptor on the next largest decrease, and so on. In the case of Table K-2, the first would go to target 3, the second again to target 3, the third to target 4, and so on.

The [illegible] candidate deployment is not necessarily optimum because all of the net pay-off figures in the table are based on the starting defense condition and, as such, the pay-off decreases are only good estimates of the deployment benefits. To be optimal, a new attack plan should be generated after each individual interceptor is assigned. (If there were hundreds of interceptors to assign, such a scheme would be very time-consuming.)

Within AEM the procedure is to deploy all interceptors by use of results like those in Table K-2, then to generate a new attack plan, re-compute Table K-2 and [illegible] if adjustments to the deployment are necessary, and so on. In practice, four to ten deployments and adjustments usually are adequate.

An issue not brought out in Table K-2, but which is addressed in the actual methodology is the concept of fixed installation costs at a site before any interceptors can be deployed and then deployment of interceptors in groups, or batteries. Such factors affect the computation of the attacker benefit tables but do not modify the basic concept.

A second issue which is addressed in the methodology is that of defenses which have been in existence prior to a new deployment. Such pre-existent defenses can exist and they, too, only affect the attacker benefit computations.

b. Methodology Implementation

Since AEM already had procedures for computation of deployment effects on net attacker pay-offs (like Table K-2) and all associated deployment logic, it was not necessary to implement the procedure over again for dedicated area defenses. Instead, this new deployment problem was resolved by modifications of input routines, internal bookkeeping procedures, and data management functions.

The key issue in the task was the data management functions. Early versions of AEM had routines with built-in assumptions that the defenses being deployed were the standard terminal category.

Therefore, items such as defense effectiveness factors were automatically taken to be those for the terminal defense input by the analyst. Then, once a deployment was achieved, all print-outs of those results were automatically tied to the terminal defense print-outs.

Necessary implementation tasks in order to convert to dedicated area defenses therefore included modification of the data management routines so that the correct defense factors were used in the computation of the attacker benefits and so that all outputs could be properly associated with the dedicated area defense outputs.

c. Option Utilization

In order to make the analyst job as simple as possible all inputs necessary for dedicated area defense deployment are designed to be used in common with those for the standard terminal deployment option. It is only necessary to input one extra parameter which indicates the number of dedicated area defenders to be deployed. Because of the inherent nature of the methodology, standard terminal and dedicated area defenders cannot be deployed simultaneously in one case. It can only be done by first deploying one type of interceptor and then in a following run deploying the other interceptor type.

L. MIXED-INTEGER VARIABLE PROCEDURES AND APPLICATIONS

1. An Overview of the Problem

The basic linear programming allocation procedure (See Chapter IV-A) about which the whole AEM program revolves has been found to have many powerful features. For example, damage constraints (IV-D), hedging allocations (IV-P), and budget optimizations (IV-J) all take extensive advantage of the option to insert miscellaneous types of constraints, which can allow the analyst to more nearly evaluate his true problem.

As this exploitation of the power of LP techniques has progressed, it has become apparent that one additional avenue of development which could lead to similar pay-offs occurs in the area of mixed-integer (MIP) variable LP's. As a first exploration of such concepts, such an option was addressed and reported upon in a separate document (Reference 13). As a result of that work, the effort to be described in this chapter was undertaken.

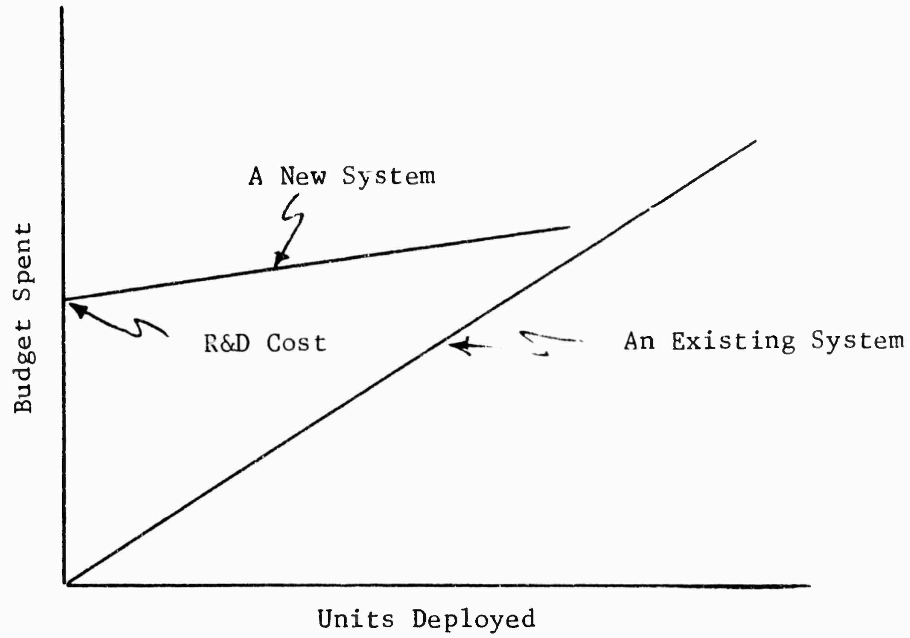
Before describing an example of a MIP option, it would be useful to review some of the current LP features. Basically, the non-integer LP traditionally used in AEM chooses from among a candidate set of weapon-to-target strategies that sub-set which maximizes total pay-off for the specified limits on number of weapons, number of targets and other auxiliary constraints. In doing so, integer numbers of weapons are fired at targets, but the LP is not constrained to using those integer weapons on an integral number of targets. For example, a strategy of six weapons of type 4 might be used to attack 8.1 targets of some class.

This has [illegible] been much of an issue in AEM because the number of targets which can be fractionated is always ≤ the number of weapon types. Since 10-20 weapon types and several thousand targets are typical for AEM, the effect of fractionating a few specific targets is minimal.

However, there are a number of applications for AEM which require that at least some of the variables picked by the LP must be only at integer values. In fact, most of these applications require that selected variables be at values of 0 or 1. Within these applications, it is always true that the majority of the LP variables are allowed to be non-integer. Hence, the name of mixed-integer LP, meaning that some variables must be integers while others can be fractions.

A classic example of a MIP application occurs in the budget optimization process when R&D funds must be spent before any purchase of certain weapon types can occur. This application was the primary one addressed in Reference 13 and it is the only one which has been implemented in AEM to date.

Take a circumstance where some number of offensive weapon types are available for augmentation of a force. Certain of these types might already have gone through development, while others require full R&D. This key difference impacts on the relationship between budget spent and units deployed as demonstrated in the following figure.



Note that the existing system has a higher unit cost (a higher slope) but that the new system must first sink the R&D penalty. The type of relationship demonstrated for the new system is of the classic "fixed-charge" problem form. That is, a fixed charge occurs before any useful objects are acquired. Then, each object comes at some unit cost.

The part the LP must play in such a budget allocation process must be as follows. One, it must decide which system types should be purchased, knowing that some of them require R&D. Two, it must decide how many of each type to buy. Three, it must decide how to allocate each of the purchased weapons to specific targets.

The key issue in this process is simply that no R&D is necessary unless at least one unit of a certain type is purchased. Once the R&D has been paid then it is possible to consider how many of each type to buy.

This type of problem can be expressed in a MIP format if the following type of budget constraint occurs in the LP.

$$X(1)R(1) + X(2)R(2) + .. X(n)R(n) + Y(1)U(1) + Y(2)U(2) + .. Y(n)U(n) \leq B \quad \text{(L-1)}$$

where: B = Total budget

R(i) = R&D cost for system i

U(i) = Unit cost for system i

Y(i) = Number of units of type i purchased

X(i) = 0, 1 variable such that

$X(i) = 0$ if $Y(i) = 0$

$X(i) = 1$ if $Y(i) > 0$

In this equation the X(i) variables are constrained to 0, 1 while the Y(i) can take on any values. The X(i) perform the function that no R&D cost occurs for a system unless at least one unit of that system is purchased.

Up to this time AEM has not had any way to deal with the fixed charge aspects to this problem. Insertion of a MIP option does resolve that deficiency however.

In order to bring out the concepts behind MIP in AEM this R&D problem will be used in all further examples in this chapter. This is most appropriate because it is the only implemented MIP option in AEM but it also does lend itself to presentation of the methodology. It should be noted however that other MIP applications are in the planning stage at this time.

2. The R&D Problem In LP Format

The AEM formulation of the R&D problem in budget allocations is demonstrated in tableau form in Figure L-1. This tableau is set up for a three-weapon, six-target class problem. A basic outline of the tableau is as follows.

a. The Rows and Columns

Each column has certain non-zero entries in certain rows. These entries indicate how many units of those rows will be utilized if the LP chooses that column to a level of one. For example, choosing column 4 means that one target of type 1 will be attacked by 42 weapons of type 2 and that 42 U(2) units of the budget will be expended. (The meaning of the other row entries will be brought out later.)

b. The R&D Decision Columns

The first three columns basically indicate which weapon types have been chosen by the LP for purchase. For example, if column 1 was chosen to a level of one, it would indicate that weapon type 1 has been chosen for use.

c. The Target Attack Columns

Columns 4 through 15 are used to indicate which targets are attacked and which resources are expended.

d. The Budget Constraint

This row controls the budget expenditure and it is basically the equivalent of equation (L-1).

e. The Total Weapon Purchase Constraints

These three rows in essence guarantee that no weapons will

	R&D Decision Variables			Target Attack Variables												
Payoff Row	0	0	0	-37	-18	-14	-11	-10	-9	-9	-6	-6	-5	-4	-5	
Weapon Constraints					14					12	8			6		≤ W1
				42			[illegible]		14				7			≤ W2
						37		[illegible]								≤ W3
Target Constraints				1												≤ [illegible]
					1	1										≤ [illegible]
							1									≤ [illegible]
								1	[illegible]	1						≤ [illegible]
											1	[illegible]				≤ [illegible]
													1	[illegible]		≤ [illegible]
Budget Constraint	R(1)	R(2)	R(3)	42 U(2)	14 U(1)	37 U(3)	• • • •									≤ [illegible]
Total Weapon Purchase Constraints	-125				14					12	8			[illegible]		≤ [illegible]
		-85		42			17		14				7	[illegible]		≤ [illegible]
			-107			37		16				3			[illegible]	≤ [illegible]
Integer Variable Limit Constraints	1															≤ [illegible]
		1														≤ [illegible]
			1													≤ [illegible]

FIGURE L-1 THE LP FOR AN R&D PROBLEM

be used to attack targets unless the appropriate R&D decision variable is set to one. For example, if X(1) = 0, then the first of these rows says that the sum of the weapons of type 1 used in the target attack columns must be $\leq$ 0. Thus, none will be fired. However, if X(1) = 1., then up to 125 of weapon type 1 could be fired at the real targets. The number of 125 is entered in column 1 on the basis of that being the maximum possible purchase of type 1. This can be computed from

$$WM(i) = \frac{B - R(i)}{U(i)} \qquad (L-2)$$

where: WM(i) = maximum feasible purchase of weapon i

f. <u>The Integer Variable Limit Constraints</u>

These rows serve a purpose which will become clear when the specific MIP algorithm is described.

If a MIP algorithm can be developed so that variables one through three would only take on the values of zero or one, then this LP tableau can be used to solve the budget optimization problem even if there are R&D costs associated with some of the force options. In the next sections such an algorithm will be described.

3. <u>The MIP Algorithm Search Process</u>

As was explained in Reference 13, the current status of MIP algorithms is somewhat cloudy. There are quite a number of codes available in commercial and non-commercial form. Each of these codes has advantages and disadvantages. Further, it has been found by a number of investigators (private communication with

R. E. Woolsey of the Colorado School of Mines) that any given code might work well on one class of problems, but poorly on another class.

Therefore, it seemed imperative to conduct further research into some of the codes in order to find one that seemed most compatible with AEM problems. This research rather rapidly narrowed down to three options - a code called BBMIP, a code called RIP, and a theoretical concept called Benders' decomposition. Each of these options were investigated, with the following results.

a. BBMIP

This code was developed at IBM, Reference 14, on the basis of theory developed by Land and Doig, Reference 15. A copy of the code was acquired from R. E. Woolsey and some degree of experimentation with the code was conducted. Some of this experimentation was reported upon in Reference 13.

Basically, the code operates as follows. First an LP is solved without regard to the integer conditions. Then the program proceeds to enumerate all the possible integer solutions by constraining the variables appropriately. Meanwhile a dual simplex LP algorithm is used to establish bounds and thus eliminate large subsets of possible enumerations.

In summary, study and experimentation with the code revealed certain basic facts. First, it provided conclusive evidence that the code would solve AEM-like problems in somewhat of a reasonable computational time. Second, it was apparent that the LP code used interior to BBMIP suffered somewhat from computational accuracy problems. These problems demonstrated themselves in erratic convergence speed and, in some cases, convergence to non-optimal solutions. Third, use of BBMIP would require extensive reprogramming in order to make it less susceptible to accuracy problems and to make computer storage requirements acceptable. This storage issue was found to be very crucial since lack of adequate storage meant that numerous LP tableaus referenced in the branch and bound procedure could not be saved and the running time would increase drastically.

Overall, it was concluded that BBMIP offered an option which probably could be implemented, but which might well lack flexibility for future growth.

b. RIP

This code was developed by Geoffrion, Reference 16, as a follow-on to work he reported upon in Reference 12. A copy of this code was also obtained from R. E. Woolsey for which we are very grateful. Experimentation with the code on true AEM-like problems was not immediately possible since in its acquired form it was capable of being used only on all-integer problems.

This code, too, is based on branch and bound or implicit enumeration concept. As a start on the code, Geoffrion describes in Reference 17 his reformulation of Balas' algorithm, with the basic idea being to minimize computer storage. Then in Refe[illegible] describes a significant improvement that was obtained by use of an imbedded linear program to generate surrogate constraints that help in minimizing the work necessary for the enumeration. In that sense it draws upon the concepts of Reference 15.

Even though experimentation with the code was not conducted, analysis of the theory was done in order to determine if the code was a candidate for use in AEM. It should be mentioned that the theory did provide a method for converting the code to a MIP problem structure.

This theoretical analysis revealed that there was some doubt about the utility of the code's imbedded LP option for this AEM problem. That LP depends upon entries in the integer columns for generation of tight surrogate constraints. In the structure of Figure L-1, it can be seen that those columns do not contain extensive entries. Thus, there was some doubt about the use of RIP in AEM.

About this same time considerable research was being conducted into the Benders' decomposition idea, so further research on RIP was held back until that concept was further evaluated.

c. <u>Benders' Decomposition</u>

Benders' decomposition is not an existing computer code, at least not in available form. Instead, it is a mathematical procedure for solving MIP problems by decomposing the problem into an iterative procedure, where the integer and the non-integer problems are solved separately. This concept was developed by J. F. Benders, Reference 18, and it has received considerable attention by various investigators. An excellent discussion of the concept is contained in Reference 19.

The key concepts are as follows. Note in Figure L-1 that the integer variables have been grouped away from the non-integer variables. Second, note that one could arbitrarily decide which weapons to pay the R&D on. Once such a decision was made, the integer variables have all been specified - that is X(i) = 1 for all weapon types where the decision was to pay the R&D. Otherwise, X(i) = 0. Given that those variables are specified, it is not necessary to have a MIP program. All that has to happen is that appropriate changes be made to the LP right-hand side (RHS), like subtracting the appropriate R&D costs from the budget.

The steps in the above paragraph are essentially those that would occur if the analyst simply guessed at which weapon types to pay R&D on. After the appropriate target allocations are done for that guess, Benders' procedure really is initiated.

The Benders' procedure provides a means for analyzing the output from the target allocation LP. That analysis then leads to a new, hopefully better, selection of which variables, X(i), should be set at 1 and which ones set to 0. Once that has happened, a new RHS is computed and a new weapon allocation performed. This new LP is analyzed and a new selection for the integer variables occurs. The iteration continues until appropriate mathematical criteria indicate that optimality has been reached.

One advantage of this procedure is that the integer and non-integer variables are separated into two distinct problems. This allows separate methodologies to be developed on the two smaller problems. Such a separation can result in a significant simplification and speed-up of the whole process.

As more and more candidate specifications of the integer variables are tried, a library of LP output information is accumulated. This updated information is used to determine new candidate specifications by utilization of an all-integer LP technique. (The logic behind the all-integer part of the problem will be presented later.) Thus, in Benders, it is necessary to have one code for solving all-integer LP problems and another code for solving non-integer LP problems.

After a review of Benders' procedure, it was decided to conduct some experimentation with it. This experimentation used an AEM LP for the non-integer code and BBMIP and RIP for two separate ways to solve the all-integer problems. It was felt that such experimentation could be conducted on problems which could be solved also by BBMIP alone, when used in its mixed integer form. The part it played in the Benders' procedure was only in an all-integer form.

By doing this experimentation, it was felt possible to at least compare very directly three candidate solution approaches to MIP in AEM. Namely, BBMIP in mixed-integer form, the AEM LP matched up with BBMIP in all-integer form, and the AEM LP matched up with RIP for the all-integer code.

In summary, the experimentation revealed that:

1) The AEM LP/RIP combination was better than the AEM LP/BBMIP combination.

2) The AEM LP/RIP combination produced answers at least as fast as pure BBMIP, without the numerical accuracy limitations.

3) The AEM LP/RIP approach seemed most compatible with the structure of AEM.

After the experimentation was completed, an analysis was made of the pro and con arguments for each approach. The result of that analysis is contained in the next section.

4. Final Choice of the MIP Approach

After all of the experimentation and analyses were complete, it was necessary to make a final decision about which technique to implement. The factors mainly involved in that decision were as follows:

1) Which approach is most compatible with AEM structure?

2) How do storage requirements and computer running time compare?

3) Is any information obtained if total convergence to optimality does not occur?

It turned out that Benders' procedure using RIP was the winner in all of these respects. The basic facts are worth presenting since they reveal some of the subtleties of the decision.

a. AEM [illegible] Compatibility

One of the key aspects to AEM weapon allocations is that a column-generation procedure is necessary. It is simply not possible to place into one LP all the possible strategies for each target. This column generation uses the LP multipliers to converge on better, and better LP strategies until optimality is reached.

A main problem with a true MIP code, like BBMIP, is that any LP row which contains entries from an integer variable column produces a multiplier of some uncertain characteristics. Some research into the impact of integer variables on the LP multipliers has been done but no clear position on their behavior could be uncovered. It seems that totally new theories about the meaning of multipliers in MIP codes are being developed.

Since column generation is a necessity in AEM, and the multipliers form a crucial part in that generation, it was felt that use of BBMIP presented some possibilities for future problems. Those possibilities do not exist when using an AEM LP/RIP combination since the integer variables are all specified during the time column generation is being performed. Thus, it has a distinct advantage.

Another compatibility issue is in favor of AEM LP/RIP because of the structure of AEM. Basically, most AEM problems do not need MIP. Therefore, any problem which doesn't need

MIP can simply be solved by use of the standard AEM LP and RIP would never be entered. Thus, no conflict would occur.

Meanwhile, use of BBMIP would possibly involve use of it in a non-integer form. In that form, it did not appear to be as efficient as the standard AEM LP.

Other structure aspects to the problem also seemed to be in favor of AEM LP/RIP.

b. Storage Requirements and Running Time

In this area the storage requirements are all in favor of a Benders' approach. Since two separated and smaller problems are solved sequentially the storage necessary at any one time is less than for an all-at-once MIP code.

Running time indications were that on smaller problems BBMIP and Benders were about the same. Indications were that Benders could be sped up but there was no indication that a large speed advantage was guaranteed.

All in all, the Benders' approach again seemed best by these measures of effectiveness.

c. Information On Non-Optimal Stops

A key issue in any allocation procedure is the information content in any case where the problem is stopped before final optimality. It is always nice to have a feasible answer on hand even if it is non-optimal.

In the case of Benders, there always exists a feasible answer plus an estimate of the maximum possible level of non-optimality. Thus, it presents an ideal situation.

BBMIP, however, might not have a feasible solution at the stopping point. It tends to not work directly from feasible-to-feasible solutions. Thus, it appears that Benders is again the best approach.

5. Implementation of Benders' Procedure

We will now present some details of the Benders' procedure implementation. The presentation will not be involved with theoretical proofs of convergence, etc., since such are available in other references. Instead, the emphasis will be upon physical understanding of the procedure as it applies to AEM.

a. Basic Benders

In order to grasp the basic Benders' decomposition procedure, consider the following matrix notation description of an LP problem:

maximize $C'X + D'Y$ (L-3)

subject to $AX + BY \leq E$ (L-4)

$X = 0, 1 \quad Y \geq 0$ (L-5)

The notation is as follows:

A = MxN matrix

M = number of constraints

N = number of 0, 1 variables

B = MxR matrix

r = number of non-integer variables

E = M vector

C' = N vector, transposed

D' = r vector, transposed

X = the 0, 1 variables

Y = the non-integer variables

In terms of Figure L-1, the above notation is equivalent to:

m = 13, $n = 3$, $r = 12$

A = the entries in the first three columns, excluding the top row

B = the entries in the last twelve columns, excluding the top row

E = the right-hand side values

C' = the entries in the top row for the first three columns

D' = the entries in the top row for the last twelve columns

The basic Benders' procedure starts out as follows. First, select a candidate specification of all the X variables. Call that candidate the vector X^o. Once each of those variables are specified the $C'X$ and AX parts of the LP problem become constants. Therefore, the above problem turns into a subproblem.

$$\text{maximize} \quad D'Y \qquad \text{(L-6)}$$

$$\text{subject to} \quad BY \leq E - AX^o \qquad \text{(L-7)}$$

This new problem has no integral variables and it can be solved by any standard LP code.

Now, the crucial question becomes - is there a better specification of the X variables that should be tried?

Benders [illegible] finding [illegible] better candidate by analy[illegible] from the solution to the above LP problem. [illegible] the procedure calls for forming the following [illegible] equation.

$$Z \leq E'U^{o} + C'X - (AX)'U^{o} \qquad \text{(L-8)}$$

where U^{o} = M vector of multipliers obtained from the solution of the reduced LP problem

Z = estimated maximum possible pay-off to the original MIP problem

By Benders' theory, the pay-off to the original MIP problem must satisfy the above relationship.

It is very interesting to consider what the right-side of (L-8) really amounts to. Term by term the right-side becomes [illegible] following:

$E'U^{o}$ - the estimated utility of the resources in the MIP problem, as measured by the original RHS, E, and the marginal utility estimates, U^{o}, out of the LP.

$C'X$ - the direct pay-off contributions of the integer variables

$(AX)'U^{o}$ - the estimated change in pay-off due to specification of any individual integer variable to 0, or 1. This estimated change being a function of the entries in each of the columns of A and the marginal utility estimates, U^{o}.

These terms are most easily grasped if a simple example is carried out to some extent. For the tableau of Figure L-1, assume that the original specification of X was $X^o = [0, 1, 1]$. Then, for purposes of our example, assume that the output LP multipliers were as follows: $U^o = [.5, .3, 1.1, 1.7, 10., 4., 3., 2., 0., 14., 25., 0., 0.]$ Using these values and the RHS of Figure L-1 results in:

$$E'U^o = .5\ W_1 + .3\ W_2 + 1.1\ W_3 + 17.\ T_1 + 10.\ T_2 + 4.\ T_3 + 3.\ T_4 + 2.\ T_5 + 14.\ B$$

$$C'X = 0$$

$$(AX)'U^o = (14\ R(1) - 125.25)\ X_1 + 14\ R(2)\ X_2 + 14\ R(3)\ X_3$$

Now, assuming that values exist for the W, T, B and R variables, it can be seen that equation (L-8) really turns out to be of the equivalent grouped form.

$$Z \leq BC_0 + BC_1\ X_1 + BC_2\ X_2 + BC_3\ X_3 \qquad (L-9)$$

where: BC_i = constants as found by the definitions for the components of (L-8)

Essentially, equation (L-9) is an estimate of the maximum possible MIP pay-off as a function of the specification of the 0, 1 variables. The coefficients for each variable are simply computed from the LP output on the basis of the resource drain, or contribution, if a specific variable was set to a value and if the current constraint marginal utilities are known from the LP.

A remarkable thing about equation (L-9) is that it is the exact equivalent of the procedure outlined in the non-overlapping island defense procedure, equations (M-8) and (M-9). In fact that whole procedure is equivalent to a modified Benders.

Since it is the objective to make the MIP pay-off maximized, it is clear that any 0, 1 variable for which BC is positive should be specified as 1. If the BC is ≤ 0, the specification should equal 0. Thus, computation of (L-9) will indicate the next 0, 1 variable specification to try and the non-integer LP can be solved again.

Once the new LP problem has been solved again, a new relationship like (L-9) can be formed. This one will be different from the previous one because the U^o values will have changed.

Benders' procedure now utilizes both of these generated relationships to develop a new candidate specification for X. The general procedure becomes:

maximize Z

subject to

$$Z \leq BC_0' + BC_1' X_1 + BC_2' X_2 + BC_3' X_3$$

$$Z \leq BC_0' + BC_1' X_1 + BC_2' X_2 + BC_3' X_3$$

$$\vdots$$

$$Z \leq BC_0^i + BC_1^i X_1 + BC_2^i X_2 + BC_3^i X_3$$

where: BC^i = the Benders' coefficients generated after X specification number i

X_i = 0, 1

As successive candidate X specifications are tried, new constraints are added to this problem. This maximize Z problem is an all-integer problem in the sense that the X variables are all integral. Therefore, it requires some sort of integer programming code to solve it.

The total Benders' proceudre can be summarized as follows:

Step 1: Choose a candidate specification for the integer variables.

Step 2: Solve the non-integer LP problem on the basis of the candidate $\hat{}$ specification.

Step 3: Extract the multipliers from the LP solution of Step 2.

Step 4: Form an auxiliary constraint of the (L-9) variety.

Step 5: Solve the integer LP problem formed from all the auxiliary constraints generated to date.

Step 6: Return to Step 2 with the new specification.

The whole procedure can terminate by use of bounding information generated as this iterative sequence is followed. First, the solution of each LP is a feasible solution to the MIP problem. Thus, the highest pay-off LP solution in the sequence must be a lower bound on the MIP solution. Second, the Z values obtained in the solution of the integer program of

Step 5 i[illegible] [illegible]id upper bound to the tru MIP solution. When the upper and lower bounds are acceptably close, the procedure can be terminated.

b. <u>Benders In AEM</u>

The items necessary for implementation of the basic Benders' procedure in AEM are rather clear. The whole structure of AEM revolves around an LP, so that structure will perform the target attack optimization once a specification is set for the integer variables. The major new item is an all-integer code necessary for the Z maximization necessary each time a new specification is generated.

In the current mode of AEM, this all-integer work is performed by a modified version of RIP. Basically, RIP was taken and reduced to its bare bones form, without the LP surrogate constraint option. This modified RIP was then added to in order to make it interface with the main AEM structure. A special subroutine, called BENDER, was written to compute the Bender equation coefficients and to feed RIP.

Since RIP is a basic branch and bound code, it is not deemed necessary to describe its workings in detail. The description of RIP provided in Reference 17 should be referenced if the reader desires to understand the approach of branch and bound. For those unfamiliar with the notion of branch and bound, the following very brief resume is provided.

If one has an integer problem with N (0,1) variables, there are 2^N possible solutions. Since 2^N is finite, it is obvious that one could enumerate all of the solutions, and thus determine the optimal one for a problem at hand. Branch and bound is simply an efficient way to conduct such an enumeration through implicit procedures.

Say that one has a partial solution, that is, some number, M of the variables have been specified. Then, depending upon the problem structure, it is often possible to devise bounding arguments which indicate whether any complete specification of that partial solution could produce more pay-off than some earlier discovered feasible solution. If the bounding argument indicates that no completion of the partial solution could lead to an improvement, then by implication a whole sub-set of the 2^N solutions will have been enumerated. Then a different set of solutions can proceed to be evaluated.

So, branch and bound is a procedure for evaluation of all 2^N solutions either explicitly, or more often implicitly. The whole procedure amounts to a bookkeeping task and being clever in devising tight bounding arguments.

One slight deviation away from classical Benders was taken in AEM after some experimentation with the procedure. This deviation essentially can be considered as a two-phase Benders, where phase 1 is a procedure used to generate good initial sets of specifications of X. The two-phases idea came about as follows.

If one chooses some initial guess for an X specification, it is most often true that the Benders' procedure then produces a new specification which is a direct complement of the previous one. That is, any variable which had value 1. will now be set at 0., and vice versa. This general kind of behavior can continue for several candidate specifications until enough Benders' equations exist to stabilize the whole procedure to the point where less dramatic changes in specifications are obtained.

The reason for this is that the Benders' equations are linear estimates of very non-linear behavior and it requires a number of such equations before the non-linear aspects can be deduced.

Knowing that the first few specifications are apt to lead to poor long-range estimates, it was decided that it would be worth-while to have some procedure for making Benders change specifications more slowly. This idea is directly comparable to the idea of - don't change more than one island at a time in the subtractive island procedure.

Such a "slow-down" control over Benders could be implemented a number of different ways. However, the current approach involves utilization of the LP structure of Figure L-1 in a special way. Looking back at that figure, note that in the R&D decision columns there is a pay-off indicated. Also, note the three extra integer variable limit constraints at the bottom of the LP. (We have ignored those rows up to now.)

Now, suppose that the first three pay-off row entries had values as follows:

If X_i = 0; pay-off = + M where M is an arbitrary constant

If X_i = 1; pay-off = - M

Then, solve such an LP problem with the three bottom rows being i[illegible]ced and using the standard LP code. In the solution that would be obtained, it will have occurred that X_i would = 0 if + M was a very large number and if it was the column pay-off and X_i = 1 if -M had been the pay-off. The so-called integer variable limit constraints serve the purpose of limiting the X_i values to ≤ 1.

This technique is simply a way to solve the LP for some determined specification of X without doing the standard Benders' modification of the RHS as called for in equation (L-7). The $\pm$ M entries guarantee that the desired values for X_i will occur.

Suppose that the X specification was only a guess. It seems clear that in such a circumstance it might be worthwhile to <u>encourage</u> the LP to satisfy the specification without <u>demanding</u> satisfaction. For example, an LP where M = 10 might indicate a preference for the LP to choose X_i as desired, but it certainly would not preclude a different choice.

This concept of encouraged specifications is the key of phase 1 of the AEM Benders' procedure. During phase 1 choices are made by KIP for the supposed next best choice for an X specification. This choice is then used to set up a special version of the LP, where selected values of M are used to encourage the LP to meet the specification. However, if the LP decides that the provided specification is very poor, it will overrule the $\pm$ M indicators and pick a different specification.

If the encouraged specification is overruled, it usually happens that a fractional specification occurs. That is, $0 \leq X_i \leq 1$. This is no catastrophe since the resultant Benders' equation is still legitimate.

The values for M are currently selected by a heuristic rule based on an estimate of the penalty accepted if the encouraged specification is not met. For example, if RIP indicates that $X_i = 1$ is correct, an estimate is made for the penalty if $X_i = 0$. This estimated penalty is then used as the value of M for that column in the LP. A similar logic works for $X_i = 0$.

After some number of passes with these encouraged sets of specifications, a convergence on some best set occurs. At that time the freedom of the LP in overruling the specification is removed and phase 2 is entered. In this phase true Benders is followed until optimality is reached.

Use of this two-phase procedure has been found to be less erratic in behavior than pure Benders. A behavior pattern that is not removed, however, is the following. Quite often the optimal answer is found some significant amount of time before the Benders' bound indicates optimality. In such a case, it would be valid to terminate early, except that no early termination rules have yet been found.

6. Other MIP Applications

At this time (June 1973) this MIP capability has been utilized in several ways besides the R&D budget situation. These other areas are as follows:

1) Use of Benders in the ABM island defense situation.

2) Creation of bomber defense districts which are separate and unique from the ABM defenses.

3) Use of either-or types of hedging conditions.

4) Modelling of two-island random defense situation.

Beyond these applications, there are undoubtedly countless others. The practicality of others remains to be seen until more computational experience occurs with current and planned capabilities.

M. SUBTRACTIVE DEFENSES

1. General

Among the simpler approaches to defense modeling is the assumption that a "price of admission" is charged the offense prior to its destruction lay-down. For instance, with D perfect defenders, the first D reliable objects presented to the defenses are negated. This assumption will be called subtractive defenses here since it deletes some of the arsenal but does not affect the optimal lay-down procedure for the remaining weapons.

The assumption of prior action is implicit in that the offense may choose the weapon type(s) deleted. This generally allows a higher damage achieved than if the defense could make the choice or even if random selection were adopted. Figure M-1 shows a hypothetical total damage function for offense choice deletion, defense choice deletion (type known by offense), and random deletion. Offense versus defense choice deletion is primarily a function of weapon type efficiency after defense negation. Random deletion is illustrated lower since arrival of any weapon to any target is now more uncertain, hence modifying the entire allocation process.

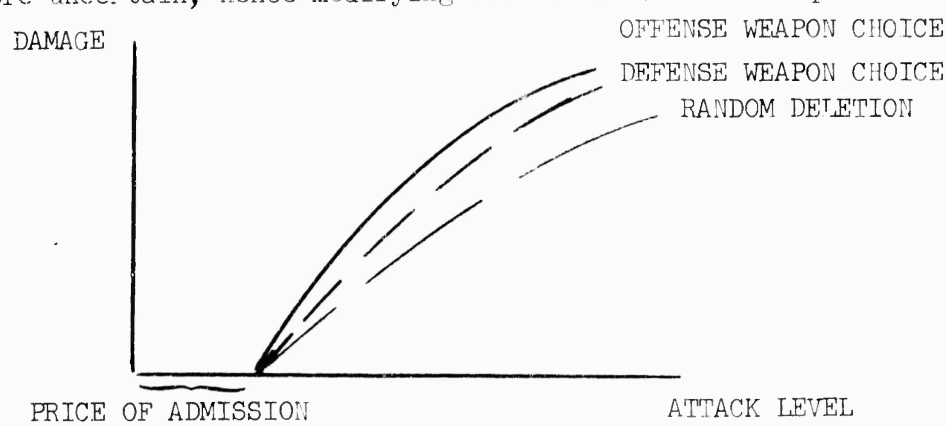


FIGURE M-1 COMPARISON OF SUBTRACTIVE AND PERFECT RANDOM DEFENSES

An exception occurs when the defense is imperfect and the attacking arsenal is relatively small. Consider Figure M-2. Unless the attack level is greater than X, allowing the defense to exact its penalty is less severe than attempting destruction of the defenses (if the "price of admission" can in fact be paid). Moreover, damage is accrued due to defense leakage with an attack level less than the price of admission.

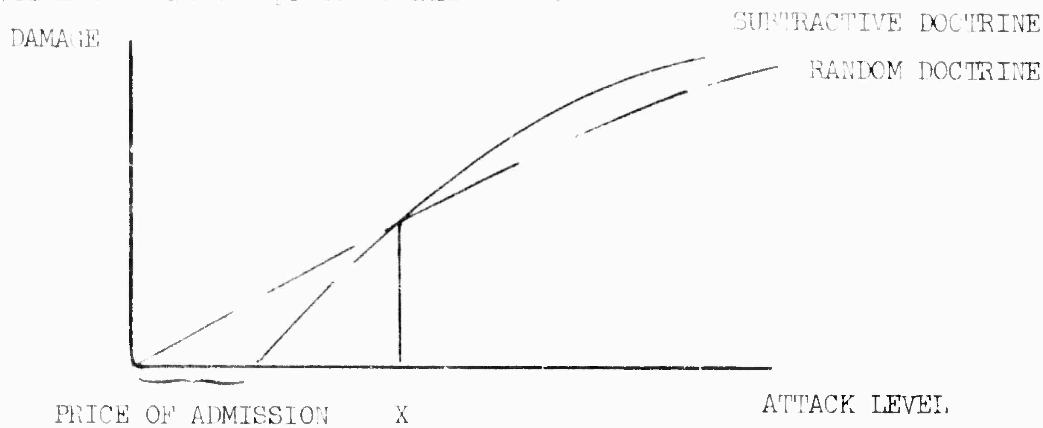


FIGURE M-2 COMPARISON OF SUBTRACTIVE AND IMPERFECT RANDOM DEFENSES

We have elected to model this defense/offense action as if the offense attempted defense destruction in the first wave of an attack. This interpretation is compatible with the basic assumption of subtractive defenses where the offense chooses the weapon type and, if successful, suffers no consequent defensive penalties in subsequent allocations (against counterforce and/or countervalue targets).

Defensive installations are described by number, hardness, area, number of defenders per installation and their capabilities against each possible attacker. A price of admission is computed by the subtractive model for each attacker. This price is then equated to the proper kill function translation with near perfect kill for the next weapon (translation plus one). The assignment of a strict equality on the target constraint in the LP assures the price will be paid, if possible. Since the remaining weapons will be optimally allocated against real targets and the defenses are assumed destroyed (99.99% assumed), the proper lambdas are produced. In this way, the process selects by iteration the optimal way (for the offense) to pay for admission.

Of major consideration to this model is the ability to assess quantitatively the penalty for poor judgment. In the case of an overestimate of the price of admission, no pragmatic harm is done, the penalty being the damage that could have been done if the price had been correctly estimated. However, when the price has not been fully paid, the penalties are more exacting. We have assumed that regardless of the number of installations or their true content, all surviving defenders may be used as whole country, randomly acting interceptors. This view is admittedly defense optimistic in light of damage assessment capabilities (destroyed radar faces, failures of command and control, etc.) which precludes some areas from defensive protection even if whole nation coverage were attained prior to the attack. It should be noted that there is no option on defense doctrine in this case. Even if the original defenses were more sophisticated (e.g., precommitted or preferential), the residual defense acts randomly.

2 The Subtractive Model

The price of admission is computed as the more economical of interceptor exhaustion ([illegible]$_X$) or installation kill through leakage

(N_K). All installations are [illegible]
hardness, area, and number of defenders, [illegible]
computational sequence for each [illegible]
precluded from this attack).

The number of attackers of type i required [illegible] the defense is:

$$N_{X_i} = S / (N_D \cdot P_{A_i} \cdot R_i \cdot O_i) \quad \text{(M-1)}$$

Where: S = number of defenders per installation

N_D = firing doctrine of the defense (e.g., two at each object)

P_{A_i} = Probability of acquisition of an incoming object of type i by the defense (i.e., firing can be accomplished)

R_i = reliability of weapon type i

O_i = number of objects observed by the defense for each incoming warhead of type i

$O_i = 1 + (1 - P_D) * d_i$

P_D = probability of determining that a decoy is a decoy

d_i = number of decoys per warhead

It should be noted that as the probability of acquisition becomes lower, more attackers are required for exhaustion [illegible] weapons are required to provide an "acquired" target [illegible] defenders. Therefore, a second [illegible] for the offense to capitalize on this [illegible].

The ability to penetrate the defense (P_P) is computed as the expected value with no assumed degradation of the defense until a sufficient confidence of installation kill is achieved, i.e.:

$$P_P = 1 - P_A + P_A (1 - P_I)^{N_D} \quad \text{(M-2)}$$

Where: P_I = single interceptor probability of kill

This probability is treated as a reliability modifier as discussed in Section IV-B-2 to compute a probability of installation kill. The price of admission via installation kill (N_K) is then computed as:

$$N_K = \ln (1 - C_D) / \ln (1 - P) - T + 1 \quad \text{(M-3)}$$

Where: C_D = the desired confidence of defense kill (input)

P and T are probability of installation kill descriptors. (They are functions of the defense installation hardness, area and defense level.)

The allowed strategy (N_F) is then found:

$$N_F = \text{MIN} \{ N_X, N_K \} \quad \text{(M-4)}$$

[illegible] price of [illegible] computed. This [illegible] determine the number of [illegible] wave [illegible].

$$[illegible] \sum_{i} [illegible] \qquad (M-5)$$

Where: N_i = number of times this strategy was used

The second uses the assumption that remaining installation capability is directly proportional to probability of survival (DPS) (computed from use of D_P and resultant installation P and T parameters), such that:

Defenders used = DUSE = $D_P + (\overline{S} - D_P)$ (1. - DPS) (M-6)

and

RDEF = $\overline{S}$ - DUSE

When: RDEF = number of random area defenders to be used (nearest integer).

= expected number of defenders not used, and surviving the first wave

$\overline{S}$ = number of initial defenders per installation times number of installations

3. Limitations

The model is very straightforward within the framework of the assumptions. However, a zone of infeasibility may exist where all weapons are used to partially pay the price of admission. If the defenses are imperfect, another doctrine should be assumed. However, if the defenses are perfect, the attack should not have been allowed. There is no internal control of this region. Since the contingencies are case sensitive, and control of the model is absolute by input, the analyst could best decide upon the proper corrective action to be taken. The most dangerous circumstance is in a three-cycle game (defensive targets only on the third cycle) when the weapons used against defenses should have been used against force targets (cycle one as opposed to cycle three) It is much more difficult to detect when random defenses should have been allowed, even though the price of admission has been paid. If the defenses are imperfect and the arsenal not allocated against defenses is small, an additional run assuming random defenses should be made for comparison. The analyst must be wary of all cases where defense dominance can occur.

The choice of required kill confidence must be less than one but may be arbitrarily close to one. If mis-estimates are to be assessed during the case, random defenders will be generated even if the desired kill was achieved. This is important since the random defenders are less valuable as the number increases. Therefore, a slight residual may cause a large variation in the answer. The selection of perfect defenders assures exhaustion will be used, producing no random defenders if pertinent errors are not incurred.

4. Extension to Non-Overlapping Islands

a. The Basic Mathematics

The previous discussion implied that there was one group of defense installations which could defend any target in the whole nation. This assumption of national coverage can be by-passed if one assumes, instead, that the defenders have limited range and that all targets are divided into distinct defended islands, i.e., each target in an island can be defended only by the defenders associated with that island. In other words, the interceptor defense zones do not overlap each other and a target is a member of only one island.

Given this basic assumption of non-overlap, the process for allocating weapons to a total target set can operate as follows.

First, recognize that the problem for the attacker is to decide which defended islands to attack and which islands to avoid. Given that the attacker selects a given combination of islands to attack, he must pay the subtractive defense price for each of the islands in the attack, but he ignores all defenses and targets in the unattacked islands.

The attackers' basic decision therefore is to decide whether or not the option of attacking the targets in a specific island are worth the subtractive price of admission to that island. This decision is obviously a function of the value and vulnerability of the targets in the island and the defense level at the island.

Fortunately, the AEM has sufficient data for the attacker to converge on the optimum decision for the total island attack grouping. This data is used in an iterative process which will now be described. The iterative process exploits the fact that once a given combination of islands is chosen for attack the targets from those islands can be lumped together and the non-island subtractive model described in Sections (1) to (3) will apply.

The process is as follows:

Step 1 - Assume that the optimal answer is to attack some combination of islands. Accordingly, allocate the total attack so as to pay the entry price on all those islands and to maximize the total damage on the combined target set. Do not let the attack go against targets in the islands which are not being attacked. This being accomplished simply by not listing those targets in the linear program.

Step 2 - Retrieve the results of the linear program used in accomplishing the above allocation. (See Section IV.A.) These results include Lagrange multipliers for each constraint involved in the L.P. for the allocation. Specifically, these multipliers indicate how much value would be lost (gained) if a single target of each type was removed from (added to) the target set.

Step 3 - Compute the estimated net benefit for including each island in the attackers' option. This benefit being computed as follows:

$$NB_S = NG_S - NP_S \qquad \text{(M-8)}$$

where:

NB_S — net benefit if island S is included in the attackers' list.

NG_S = Expected gain if all targets in island S are included in the attackers' list.

NP_S = Expected price which must be paid in order to destroy the subtractive defenders at island S.

Computing each term in (M-8) is possible by use of the L.P. by-products as follows:

$$NB_S = \sum_j TN_{js} \cdot \lambda_j - (V_d - \lambda_d) \; TN_{ds} \qquad \text{(M-9)}$$

where:

TN_{js} = Number of targets of class j in island S.

TN_{ds} = Number of defense installations in island S.

V_d = Value placed on each defense installation.

λ_j = Target multiplier from the L.P., which equals the expected gain if the attacker had one more target of type j to attack.

λ_d = Defense installation multiplier from the L.P.

The first component of equation (M-9) sums up the expected gain in target damage if the attacker could attack the targets in island S. The second component sums up the expected loss if weapons which currently attack other targets must be diverted to attack the defenses in island S.

Step 4 - Determine if the net benefit was positive for all islands being attacked in Step 1 and negative for all islands not attacked in Step 1. If so, the assumption in Step 1 was correct and the allocation is at least at a local optimum. If not, proceed to Step 5.

Step 5 - Form a new island attack combination by use of the Benders procedure available in AEM. (See Chapter IV-L). This procedure joins the latest set of island net benefits to all previous net benefit sets and utilizes an all-integer code to choose the best possible specification of which islands to attack.

Step 6 - If this new island attack combination has never been considered before, return to Step 1 and start over with this new combination. If it has been considered before an optimal solution is guaranteed.

The basic advantage of this process is that a requirement for large storage space in the allocation process is by-passed by going to the iterative procedure.

For example, if one wanted to work a case with 25 target classes and fifteen islands, an alternate approach would be to create 25 X 15 = 375 equivalent target classes and thus represent the fact that targets of a class are still differentiated by their island identifier.

This iterative process by-passes such a procedure, which would require excessive computer storage, in favor of making several iterations through a problem which always has twenty-five target classes. The main idea is that a procedure for finding the optimal combination of islands to lump together into one equivalent target set can be faster than solving an enormous one-step problem.

The interface with the Benders procedure as included in step 5 is very straightforward. In essence, the NB_s values computed by equation M-9 are equivalent to the BC^i coefficients utilized in equation (L-9) and the maximization problem on page IV-L-20. The step-wise procedure on page IV-L-21 is the exact equivalent of the 6 step procedure just described here. Therefore, the mathematical implementation of this procedure is described in Chapter IV-L if the reader is interested in it.

b. <u>The Impact of Misestimates</u>

One of the key attributes of AFM is its ability to analyze the impact of misestimates in the input parameters. In this non-overlapping island option, there are some special aspects to such analyses which will now be discussed.

If one desires to analyze misestimates which apply to the island configurations, there are four areas which are most critical. These are misestimates of:

1) Defense performance

2) Defense interceptor stockpile

3) Interceptor range

4) Interceptor farm location

Type 1 and 2 misestimates have a similar impact in that they affect the accomplishment of killing the subtractive defense. For example, if one underestimates the level of defense, or the performance, the net effect is that the required defense price is not paid and the attack on the non-defense targets is vulnerable to the defense that was not destroyed.

However, if one overestimates the defense level, or performance, the impact is that more than enough weapons were used on the defense. Thus, unnecessary wastage of weapons occurred.

In the island defense case, one could misestimate the defense price by amounts that vary from island to island. Thus, in actuality, there could be varying levels of misestimate penalty from island to island.

Such an island-dependent misestimate penalty is quite difficult to implement in AEM, however. The reason is that AEM lumps together all targets into classes when the allocation process begins. (Up to that time, the targets are classified island-by-island.) Thus, the allocation ends up by saying how many targets in a class to ... but the island association is lost.

For example, say that islands 3, 5 and 7 were being attacked and their target class configuration was thought to be:

Number of Targets By Class By Island

Island	Class = 1	2	3	4	Defenses
3	5	18	0	17	10
5	3	7	12	10	0
7	2	3	8	2	8
Total	10	28	20	29	18

AEM would choose an optimal allocation for the total targets of all classes (and the defense targets) in all islands. For example, such an allocation might say to attack eight of the defense targets by one type of strategy, seven by another strategy and a different attack on the remaining three.

The key point is that the island identifier is not associated with these strategies. Thus, there is no information as to which strategies apply to the targets in any given island.

One could make some nominal assumptions that would arbitrarily proportion out the strategies to the specific islands. However, the programming required to do this and to complete the island-by-island misestimate impact was not felt to be warranted at this time. Such programming is feasible, however.

Instead, AEM currently approximates the misestimated defense level by aggregating all defense level misestimates into one total figure for all islands attacked and then assumes that the surviving defenders have nation-wide capability and are random in their effect on the attackers. (Thus, it is similar to the technique discussed in Section 2 of this chapter.) In essence, any surviving defender is assumed to attack randomly over all attackers and is not confined to those attacking his island.

By lumping all island misestimates into one total misestimate, this approach ignores all island-by-island distinctions and gives credit to the offense for proper proportioning of his attack to the islands. He only pays a penalty if his total attack on all the islands is improperly sized. In return, however, the defense gets to use his excess defenders to defend any target in any island. Thus, we have a blend of offense optimistic and defense optimistic aspects to the problem. No accurate measurement of the impact of these contrasting assumptions has been made.

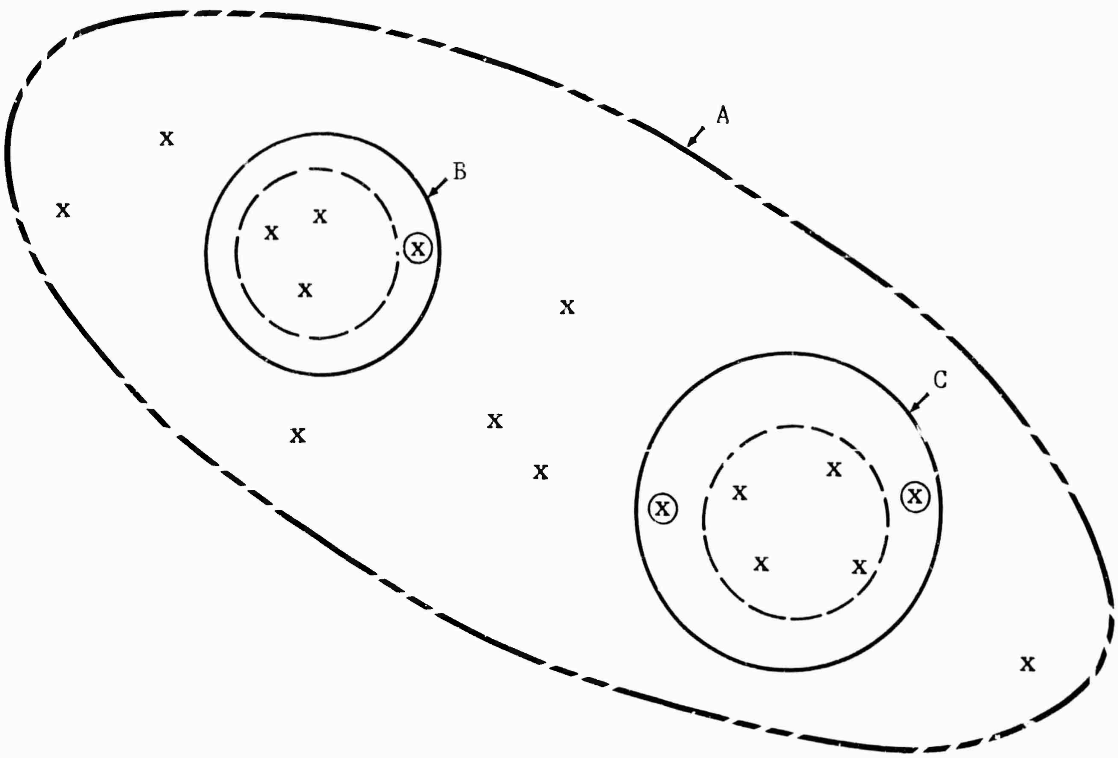
There is a separate, but similar, logic for the handling of misestimates in interceptor range. In AEM such a misestimate would be reflected in a misestimate of the number of targets in a class in an island. For example, if the range was shorter than anticipated, there would be a number of targets less than anticipated in a given island. The reverse being true for a longer range interceptor.

Before delving into the range estimate problem, it is necessary to point out a basic fact that might not be obvious in this island analysis. This fact is that all targets must be associated with some island and this means all targets which are not in an actual defended island must be considered as belonging to an <u>undefended</u> island. This requirement is inherent by virtue of the mathematical process utilized.

A basic follow-on fact is that misestimates in range demonstrates itself only in incorrect association of targets with a specific defended island <u>or</u> with the undefended island. This happens by virtue of the fact that all islands are non-overlapping and the zone between all defended islands belongs to the undefended island.

Thus, misestimating the boundary of a given island always transfers targets from that island to the undefended island, (or in reverse) and never to another defended island. (This being true as long as the battery location itself is not in doubt.)

With these basic facts, consider the case where the range was overestimated. (Such a case is diagrammed in Figure M-3.) In such a case, some targets associated with a defended island will properly go to the undefended island. (Visualize the dashed line as the actual range . . .) However, this has no serious impact at all since the price was paid on the defense in that island and the target in essence is "undefended" anyhow. The only impact is that the attacker might not have paid the price if he had known about the nondefended nature of the target.



A = The undefended island.

B = First defended island.

C = Second defended island.

x = Targets

(x) = Targets which will be defended or undefended, depending upon correct range line.

- - - = Smallest likely range line.

——— = Largest likely range line.

FIGURE M-3 A DIAGRAM FOR RANGE EFFECTS

In the case of underestimating interceptor range, however, the problem is more serious. (In Figure M-3, the actual range would now be the solid lines.) In such a case, a target was thought to be undefended but in actuality it was defended. Two variations are important when this happens. If the target moved into an island for which the subtractive price was paid, there is no impact since the defense will already have been killed. However, if the target moved into an unattacked island, the defense might well be able to nullify any attack on this target.

The treatment of this circumstance in AEM is currently as follows. First, it is assumed that any targets which went into a defended, but unattacked, island are completely protected by that island and no damage occurs on such targets. Second, it is assumed that the attacker does not make a deliberate effort to avoid attacking targets which are near a boundary of an island not being attacked. He simply attacks the targets he has been allowed to attack and he does not hedge his attack.

Then, for all targets in a class which are attacked, AEM computes the number of targets which moved into a defended and unattacked island and removes an appropriate share of such targets from the damage list.

For example, take the previously described case where islands 3, 5 and 7 were being attacked. In that case, it was assumed that target class 2 had 18 members in island 3, seven in island 5 and three in island 7. If, however, the range had been underestimated, it could be that some of the seven targets in island 5 (which has no defenses) actually were in defended islands. Say, for example, that all seven were actually in other islands as follows: three were in island 7, two were in island 4, and two were in island 8.

Now, since island 7 has had its defenses killed, any attack on the three additional targets that were in that island will do their prescribed damage. However, by our assumptions, the attack will do no damage on the other four targets, which were in defended and unattacked islands. AEM will compute this loss of damage on the four targets (if they were attacked).

Misestimates of interceptor farm location is much more difficult to treat in a consistent manner. In such a case targets can go from one defended island to another defended island, or into the undefended zone. In essence, the sky is the limit in terms of types of transfers which could occur.

No feasible and totally consistent resolution of this circumstance has been found. It now appears that the only way to describe the impact of misestimating farm location would be to identify the change in association with each individual target and, to identify that target in all allocations. Such detailed tracking of targets is counter to the basic aggregated nature of AEM.

Therefore, AEM does not have at this time a special treatment for farm location misestimates. The only option available is to input the misestimate in terms of numbers of targets identified with a given island. Thus, the impact will be similar to a misestimated range problem. There will not be any way to indicate the case where an island loses one member of a target class and gains another member of the same class.

c. <u>Bomber Interactions With Islands</u>

There is a potential conflict in this subtractive island model when one considers the interaction of the defense with bombers. Basically, the question revolves around the vulnerability of the bombers to the island defenses. Once can visualize at least four main possibilities, i.e.:

1) The bombers are totally unaffected by the defenses.
2) The bombers have their own set of island defenses to penetrate.
3) The bombers are affected by the defenses but they cannot kill, or exhaust the defenses.
4) The bombers are affected by the defense but they can pay the defense price.

At the present time AEM is programmed for condition two, that is, the bombers have their own set of islands and all targets must be described as belonging to one of those islands. Since a target could well belong to a missile island and a bomber island, this automatically leads to a method for dealing with overlapping islands. Such a methodology is described in Section 6 of this chapter.

5. Deployment of Subtractive Island Defenders

This task has the same basic function as the terminal defense deployments described in Chapter IV-K, namely to develop a procedure for deployment of a type of defense interceptor among a set of candidate locations so that benefit obtained from the interceptors is maximized. It is similar also in terms of the concepts used in accomplishment of the task. Basically, computations must be performed in order to compare benefits to be attained by various deployment options and then to provide a method for converging upon the best deployment.

It would be helpful to recap the concepts involved in subtractive island defenses (as described in Section 4) and thus set the stage for the methodology description.

Within AEM all targets are aggregated into composite classes, where all members of a class have essentially similar characteristics. With no defenses this aggregation can usually represent the urban targets of interest in about ten classes. As defenses are inserted, this low level of aggregation is difficult to maintain without sacrificing some degree of accuracy. However, in the case of subtractive island defenses, the aggregation is still possible, hence the attractiveness of using such a defense representation.

First of all, island defenses are those defense situations where limited range interceptors make it possible to defend reasonably small regions (hence the name islands), but regions that are large enough to include a number of separated and distinct targets. For example, a region might be several hundred miles in diameter.

Secondly, subtractive defenses are defenses which are rather low in intelligence in the sense that they shoot at the first arrivers in their island, regardless of the aim point of the arriver, or the lethality expected. Thus, the name subtractive, since they subtract a price from the total attack on the island and once the price is paid all follow-on attackers enter without any interference from the defense.

If these island subtractive defense installation are located so that no overlap exists between islands, one can visualize the total target set in a country as being broken into sub-sets where targets in each sub-set all belong to the same island, and not to any other island. Any individual target class can then have a separate set of identifiers to indicate how many members of the class belong to island 1, to island 2 and so on.

If a total set of targets are described in this manner, and if the defense level (number of interceptors) associated with each island are specified, then the attacker must decide which islands are attractive enough to warrant attack. By definition, those islands chosen for attack must be such that there is adequate payoff received in an island after paying the subtractive price (or killing the island defense).

Given such island descriptors, it is possible to devise iterative procedures for determining the optimal combination of islands to attack for a given total attacking arsenal. Section 4 of this Chapter described just such a procedure.

The problem in this section is to devise a procedure which will find out how many defenders should be deployed at each island in order to minimize the damage attained by the attacker when he chooses which islands to attack. Basic assumptions in the task are as follows:

1) The attacker knows about the deployment chosen.

2) The geographical location is specified and, hence, the number of targets in each island is given.

3) The task is to assign numbers of interceptors to the islands and, by virtue of assumption number 2, the number of targets in each island is not to be changed.

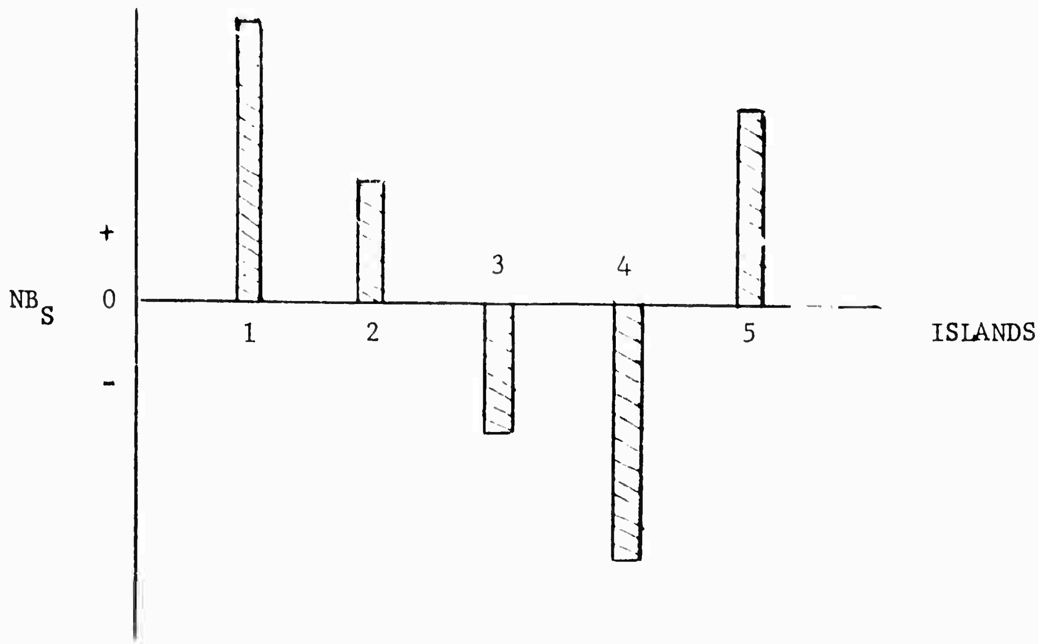
a. The Methodology

It would be appropriate at this time for the reader to review the essential ingredients of the optimal island attack procedures described on pages IV-M-9 to IV-M-13. That procedure forms the basis for the deployment procedures.

Th[illegible] proc[illegible] [illegible] the described procedure [illegible] very rapid and, while no mathemat[illegible], [illegible] sequences have resulted in finding the optimal combination [illegible] to attack.

The items in the procedure that are most pertinent to this discussion are the island net benefits described by equations (M-8) and (M-9). In essence, any island which has a positive net benefit NB_S will be attacked and any island which has a negative NB_S is not worthy of attack. As those equations indicated, one of the major components in computing NB_S is the defense level at the given island. The defense level is the item to be chosen in the deployment so there should be much valuable information in the net benefit computation procedures.

Consider the following sequence. Choose a starting island defense deployment by some arbitrary means. Given that deployment, compute the optimal attack on the islands and the resultant net benefits. For a five-island case (AEM can handle up to 15 islands), the NB_S values might plot as follows:



Note especially that not only do positive and negative values exist, but different magnitudes also occur. Island 4 has a large negative NB_S. This means that the subtractive price on that island is so large that the island is definitely not worth attacking. Meanwhile island 1 has considerable margin in positive net benefit, which means that the targets in that island are well worth the subtractive price for the island.

It is a basic thesis of defense deployments that an optimal deployment is one where all defenders engage the attack in some manner. By definition, any defender is of no value if it is so placed as to never have any opportunity for engagement. This says that some of the defenders on islands 3 and 4 must be misplaced because the island negative NB_S means no attack will enter those islands.

Additionally, island 1 is the most prime candidate for additional defense since his NB_S has the largest positive margin and a few more defenders at the island would simply extract a larger price from his attack. However, excessive additional defense could drive the net benefit on island 1 to zero, or even negative.

The number of defenders to drive any island NB_S to zero can be estimated by setting $NB_S = 0$ and solving equation (M-9) for TN_{ds} follows:

$$TN^{*}_{ds} = \sum_{j} TN_{js} \cdot \lambda_j / (Vd - \lambda d) \qquad \text{(M-10)}$$

where:

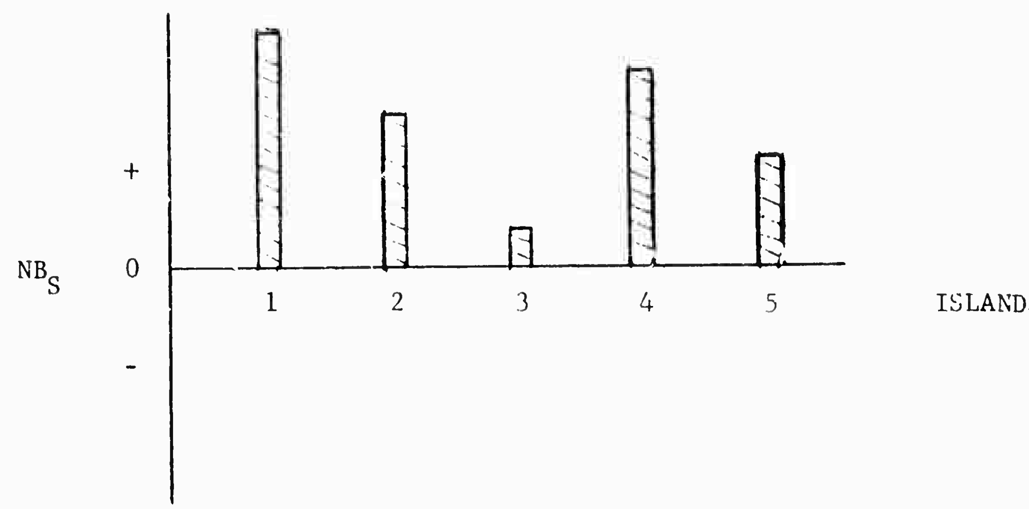
TN^{*}_{ds} = estimated number of defense installations at island 5 to have a zero net benefit.

This is just an estimate because changing the deployment very significantly from the starting deployment will change the λ factors in equation (M-10) and the impact of a given number of defenders would be misjudged.

One must always remember that the λ factors are only marginal slope indicators at some given condition point. They are linear and therefore cannot predict non-linear functions over too broad a change in condition.

Despite the imperfect nature of the estimate provided by equation (M-10), it still forms a basis for making some intelligent deployment changes away from the starting deployment.

Before delving further into the deployment procedure, it would be well to discuss another aspect to the deployment issue. Consider the following net benefit plot:



In this case all the net benefits are positive, thus all islands will be attacked, and all defenders engaged. By our previous optimality condition, there must be an optimal deployment of defenders. However, it has been another basic thesis in this task that a better deployment is one where all the NB_S values are equal, even if they are all positive.

The basic reasoning behind this thesis is as follows. Optimal deployments should also minimize sensitivity to changes in the assumed defense performance, the attacking arsenal size, etc.

In any case where an island NB_S is [illegible] condition change might drop it [illegible], if all NB_S values are nearly the same [illegible] minimal sensitivity to assumptions should [illegible].

Following this line of thought, equation (M-9) can be used to determine, for island S, the estimated number of defense installations in order to have an NB_S value of some specified amount, or goal.

With this concept in mind, the basic island deployment procedure is as follows:

Step 1: With a starting deployment, compute an optimal island attack and all associated NB_S values.

Step 2: Evaluate the NB_S terms in equation (M-9) in order to determine what defense level at each island would more nearly equalize all the NB_S values.

Step 3: Using the new deployment resulting from Step 2, return to Step 1 and compute new NB_S factors. This process will be repeated until the NB_S terms are all adequately equal to each other.

The details of the implementation of this procedure are provided in the next section.

b. Methodology Implementation

The total implementation process can be viewed as a reflection of the 3-step process just presented in the methodology section. The first phase thus consisted of development of a start-up procedure for getting an initial deployment to begin the iteration process.

A basic start-up deployment is a necessary first step in arriving at an optimal deployment. At the present time the start-up is achieved by letting the program compute a deployment which is proportional to the total value of all targets included in each island. That is, for example, an island will get 20% of the defenders if it contains 20% of all value which has been included in islands that are to be defended.

Once this start-up deployment is chosen the program computes the optimal attack plan on the defended islands. The main product of interest to the deployment process are the NB_S factors obtained as a result of the attack and the λ values for all target and defense installations.

One complication in this step of the implementation results from the nature of the procedure used to compute the optimal island attack plan. Recall that a 7-step procedure was necessary in order to find the island attack plan. These 7 steps often result in a circumstance such that several island attack plans are nearly optimal and convergence on the optimal plan is achieved only by trying all of the near optimal plans in order to sort out the best plan. When such an event occurs there is some question about which attack plan should be used to provide the NB_S values.

Evaluation of a number of such cases demonstrated that the most stable and consistent values for NB_S and λ_d could be obtained by using the following procedure. First, NB_S and λ_d values were obtained for all the island attack plans uncovered while searching for the optimal plan. Then, a weighted average of the NB_S and λ values was computed, where the weighting factor was the ratio of the total value destroyed in the given attack plan to the value destroyed by all of the attack plans combined.

For example, take a case where three attack plans were evaluated and the NB_S for island one was as follows in each case (+ 150, + 100, + 125). Also, for this example, say that the attacker payoff for each attack plan was (1010, 890, 955). Then, the weighted average NB_S for island one would be computed as follows:

$$NB_S^* = \frac{(150 \times 1010 + 100 \times 890 + 125 \times 955)}{(1010 + .890 + 955)} \qquad \text{(M-11)}$$

$$= 130.8$$

Only the near-optimal attack plans were used in this weighting procedure and one clearly optimal plan would dominate the weighting process.

These weighting procedures were useful mostly in those situations where the NB_S values would change quite drastically even though small changes would occur in the total attack plan damage. Not using the weighting process could result in oscillations in deployment plans.

Given that NB_S and λ_d values are available, the procedure then requires that the defenders be moved from certain islands to others in order to equalize the anticipated NB_S values for all islands. Equation (M-9) can be used to compute what the equalizing deployment should be.

However, experience proved that too complete and total a shift in deployment could create convergence problems. The explanation lies in the fact that the λ 's are only slope indicators for a given deployment. The effect of large shifts in deployment are simply not predicted well by the λ 's.

The most stable procedure developed was to allow shifting only up to one-half of the defenders away from any one island and no more than doubling the defenders on any one island at any single deployment stage. This semi-control of the step-size in deployment changes created the necessary stability.

Given that the appropriate numerical control steps were taken, the implementation process for this deployment procedure consisted basically of internal data management input procedures and output formatting. As always in AEM, these modifications required care in execution but no sophisticated math was necessary in their completion.

c. <u>Option Utilization</u>

The option can be utilized very simply by indication of the island makeup by target class, the number of defenders to be deployed, and the existing deployment if there is one.

The necessary input variables are described in The Arsenal Exchange Model Handbook.

6. Extension to Overlapping Islands

a. The Basic Problem

In section 4. of this chapter a methodology was presented for the circumstance of non-overlapping subtractive island defenses. That methodology was developed in order to resolve one of the key issues in area ABM defense modelling. That issue is simply that defenses do have limited range, and national coverage defenses are not realistic.

The work of section 4. allowed for finite range defenses of a special category, namely, where none of the defense zones overlapped. In other words, any individual target was a member of only one defense zone. The targets would thus all be disjoint sets and the result would be that a country would be divided into regions where certain regions, or islands, had defenses and other regions had no defenses.

As the number of defense installations increases, however, this assumption of non-overlapping defense coverage becomes rather weak. Thus, it would be very appropriate to consider means for extending the non-overlapping methodology to the circumstance where at least some of the zones do overlap. Such an extension has been accomplished and this section will describe the methodology.

b. Some Considerations

A key factor which has previously precluded full develpment of overlapping defense options in AEM is the nature of the aggregation in AEM. Basically, AEM is designed to operate with 20 to 50 target classes. By limiting the target class distinctions, considerable speed and storage advantages have been achieved.

In essence, the presence of overlapping defenses can be viewed as another target distinction factor just like target area and value normally are considered as such. For example, if one target lies in a zone protected by defenses 2, 5 and 8, while another identical target lies in a zone protected by defenses 2, 14 and 15, it is necessary to have these targets separated into different classes.

However, such increases are not compatible with AEM structure. For example, take a hypothetical case of only five defense zones. If any degree of overlap is allowed for, there could be a theoretical maximum of 2^5 different defense overlap regions. For example, a region might be that area which defenses 1, 3 and 5 could all protect. Since any member of a standard target class could be a member of any overlap region, there would have to be allowance for $2^5 \cdot NT$ target classes, where NT is the standard AEM target class numbers.

Even if such storage was available, there is a question of data input. AEM is typically used in a parametric manner where defense levels, etc. are varied. If each target in a list of 2000 targets had to be identified with specific defense zone coverages for each variation, the input job would be so large that an input generator would have to be developed.

Such considerations resulted in a feeling that approximate modelling of the overlap problem would have to be developed before such a capability could be added to AEM. The idea was that some approximations might be possible such that the basic impact of overlap could be logically dealt with and still be compatible with the agregated nature of AEM.

c. Degrees of Overlap

As was just discussed, it is not compatible with AEM technology to deal with large numbers of islands with extensive degrees of overlap. AEM conceptually must deal with aggregated situations and the presence of a few (say 15 or less) islands for a complete country. Therefore, it seems reasonable to develop a methodology which would be designed to deal with situations where mostly simple overlaps occur. That is, where a given target is generally defended by, at most, two defense islands.

One obvious case which is important is that of distinct missile and bomber defense islands or zones. In such a case none of the missile, or bomber islands might overlap with another island of its own type but it probably would overlap with certain of the opposite type of defense.

For example, consider Figure M-4. That figure demonstrates a situation where 3 missile and 2 bomber defense regions exist. The two bomber regions happen to protect some targets that are

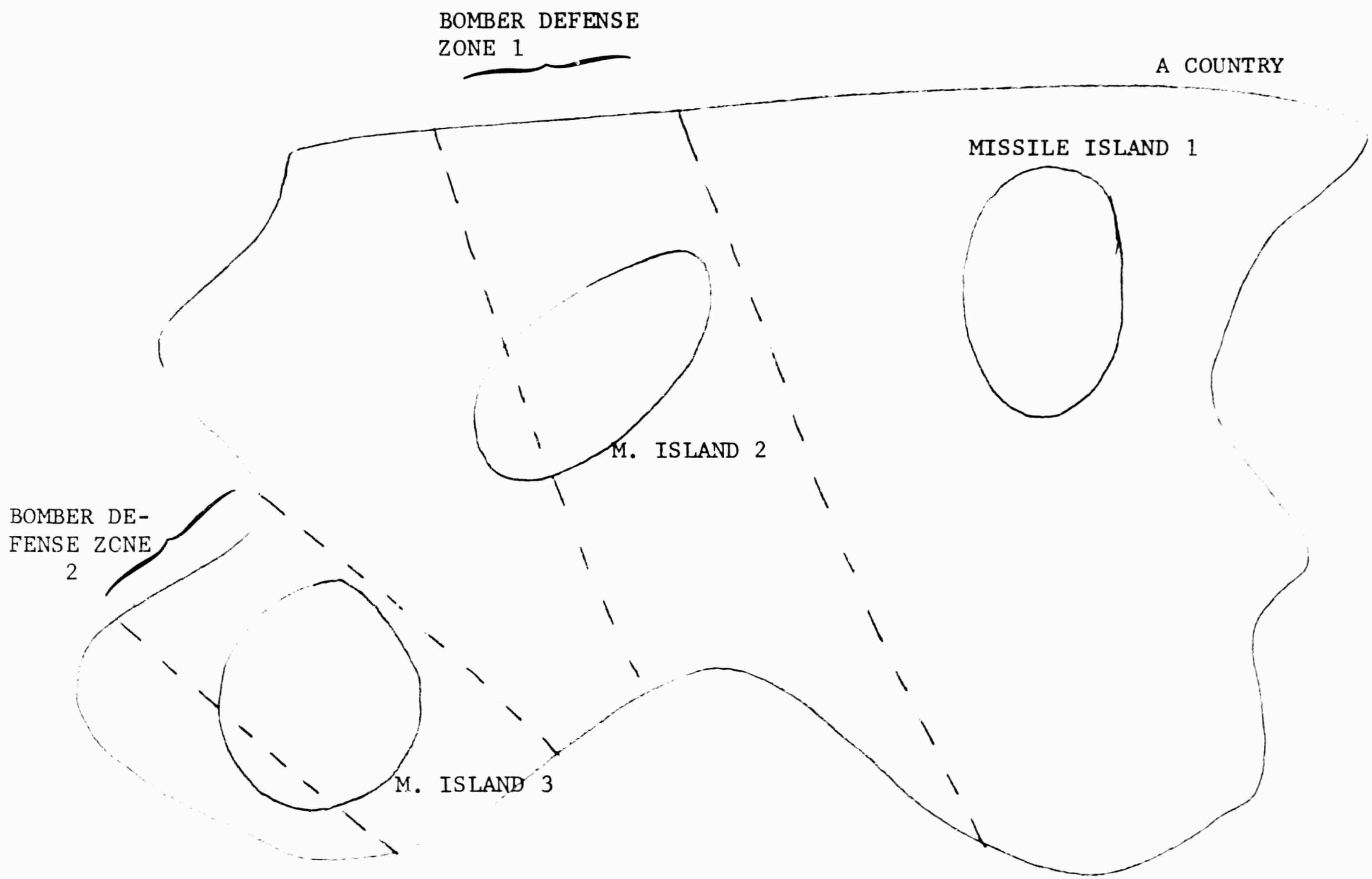


Figure M-4. Bomber and Missile Islands

also protected by missile islands 2 and 3. Each island, however, contains some targets that are protected only by a single defense type.

It was felt that dealing with this special type of overlap was quite important. Therefore, a methodology was developed which will allow for overlap between missile and bomber defenses and will allow overlap even between given weapon type defenses. However, because of data storage problems it is not feasible to have large numbers of complex degrees of overlap. Any single region, however, can have any desired degree of overlap.

Basically, the analyst must describe the target list make-up (in terms of numbers of targets by class of defense zone) in each unique defense zone. A unique defense zone is one where all targets have exactly the same islands defending them. In Figure M-4 there are 8 such defense zones (including a single zone for all undefended targets). This input problem is limited mostly because there is currently a limit in AEM of 15 unique zones.

d. <u>The List of Uncovered Targets</u>

The basic concept of island defenses leads to a requirement to compute the correct list of uncovered targets as a function of the islands which were chosen for attack and the islands which were not chosen for attack. This computation is basically the one required in step 1 of the island attack optimization procedure outlined on page IV-M-10.

If the analyst has defined every category of defense zone coverage such a computation is mainly one of list search and addition, For example, if islands 1, 2 and 8 are to be attacked it is necessary to inspect each defense zone to see if it is defended only by one or more of these islands, or if it is totally undefended. If a zone is in the above categories all targets included in that zone are uncovered by the attack on islands 1, 2 and 8, and are therefore eligible for attack themselves.

This simple type of uncovered target computation is done in AEM.

e. Separate Missile and Bomber Target Lists

If some of the island defenses are missile defenses and some are bomber defenses it is necessary to compute three subcategories in the uncovered target list. The three are: targets which can only be attacked by missiles, targets which can only be attacked by bombers and targets which can be attacked by any weapon type. Then the LP must be constrained so that the appropriate conditions on allowed target attacks are met.

This means that a typical LP will have 3 times as many target constraints as normal. For each target class there will be 3 separate constraints which guarantee that the correct attack occurs on the uncovered targets. Then, when a strategy column is inserted into the LP the appropriate target constraint entry must be made. That is, the Lagrangian process must indicate which of the 3 target constraint types the strategy is designed for.

This effect on the Lagrangian and the LP column generation is very direct. Basically, the target constraint lambdas indicate which of the constraint types are least binding and therefore which category of target constraint would allow the most effective attack. For example, consider the case where there are 200 targets eligible for each weapon category attack, 50 only eligible for missiles and 25 only for bombers. If a current LP has an attack on 105 of the 200, 45 of the 50 and 75 of the 25, the next target attacked by bombers would be from the list of 200 because the bomber has attacked all 75 confined to his attack.

Another effect of this distinction between missile and bomber target attack constraints shows up in computation of the island net benefit from attacking a specific island. Equation (M-9) shows how that computation is a function of the target multipliers (or lambdas) and the number of uncovered targets, TN_{js}. When there are missile and bomber uncovered targets the computation of island attack net benefit must span over [illegible] all the categories. This would then require [illegible] the properly matching [illegible] and TN_{is}. Otherwise the [illegible].

f. [illegible] Bomber Defense Prices

In this formulation it is necessary to describe a defense "price" for the bomber islands, in addition to a price for the missile islands. It is recognized that the standard concept of bomber area defenses runs counter to such a price concept. First, such defenses are more nearly viewed as random defenses, with a price proportional to the attack level. Second, such defenses are not usually independent as island defenses usually are. Typically, one must penetrate a zone before he can even enter another zone.

These facts were recognized prior to the modelling of the current independent, price-type of bomber islands. However, the plan was to make the current capability a first step in a development cycle. The second step will be to provide price computation procedures and dependency among the islands. After that next step realism should be much more acceptable. However, until that stage is reached the analyst must either utilize or not utilize the current option on its current merits.

N. ESTIMATES AND MISESTIMATES

Allocations are performed in the AEM on the basis of single valued data. Such data inputs may be classed into two groups, those which are explicit (e.g., numbers, hardness, yield), and those which imply assumed distributions (e.g., reliability, C.E.P.). This data is used by the weapon allocation routine to produce an allocation which will achieve the optimum expected value damage if the expected values of these variables are correct. This section will address the measurement of the impact of errors in these expected value estimates. The errors introduced by implied, inherent, or other distributional factors will not be addressed (see Section IV-L). Therefore, error assessment can occur in one value, namely the difference between the damage attained by optimum allocations and the damage attained as a result of a misestimate of the expected value of one or more parameters.

Allocations are normally performed using the assumptions of the side commencing the war. That is, the side going first assumes his estimates of all parameters are not only correct, but shared by his opponent. Obviously, if he estimated a certain weapon would have a yield of 2.0 megatons where it really has a yield of 1.5 megatons, his expected kill of targets attacked with this weapon is less than anticipated. Such an assumption may not be catastrophic. The allocation may also be optimal for 1.5 megatons (if the change in expected kill would not cause a reallocation of forces). In this case, the penalty for not knowing the yield was 1.5 megatons is zero, but the penalty for not having two megatons yield is positive.

If a side's estimate of his own forces are not accurate, the effect is like looking at oneself in an imperfect or wavy mirror, Similarly, a misestimate of an opponent is analogous to a view of that opponent through a distorted or wavy glass. Hence, cases where an unrecognized error exists are often referred to as "wavy" cases.

If a side believes the response will be based on a different set of estimates than his own, he is assuming his opponent is looking through a wavy glass at him and in a wavy mirror at himself (since a side believes his own estimates to be wholly true). Thus, the phrase "wavy-wavy" is frequently used to describe a case having these assumptions. Wavy and wavy-wavy capabilities are now discussed separately.

1. Misestimates (Wavy)

The concept of misestimates is centered on the difference in the planning function and the consequence of the plan. The side initiating the war allocates his weapons and predicts a retaliation based totally on his estimates of both sides and his objectives (assured destruction, damage limitation, etc.). Figure N-1 shows a schematic of a two-strike war with misestimates. The allocation on the first strike is generated on BLUE's estimates of both sides (including the expected response). Note that BLUE assumes RED has precisely the same knowledge as BLUE. The actual expected damage to RED is then calculated. The true survivors are used by RED to generate his response based on his estimates of his weapon capabilities, BLUE's defenses and characteristics of his value and non-retaliatory, military targets (OMT for other military targets). The true expected damage for this plan is then computed.

It should be remembered that the allocations are based on a single value for each parameter and the allocations are optimal if these values are correct. Therefore, the penalty incurred due to

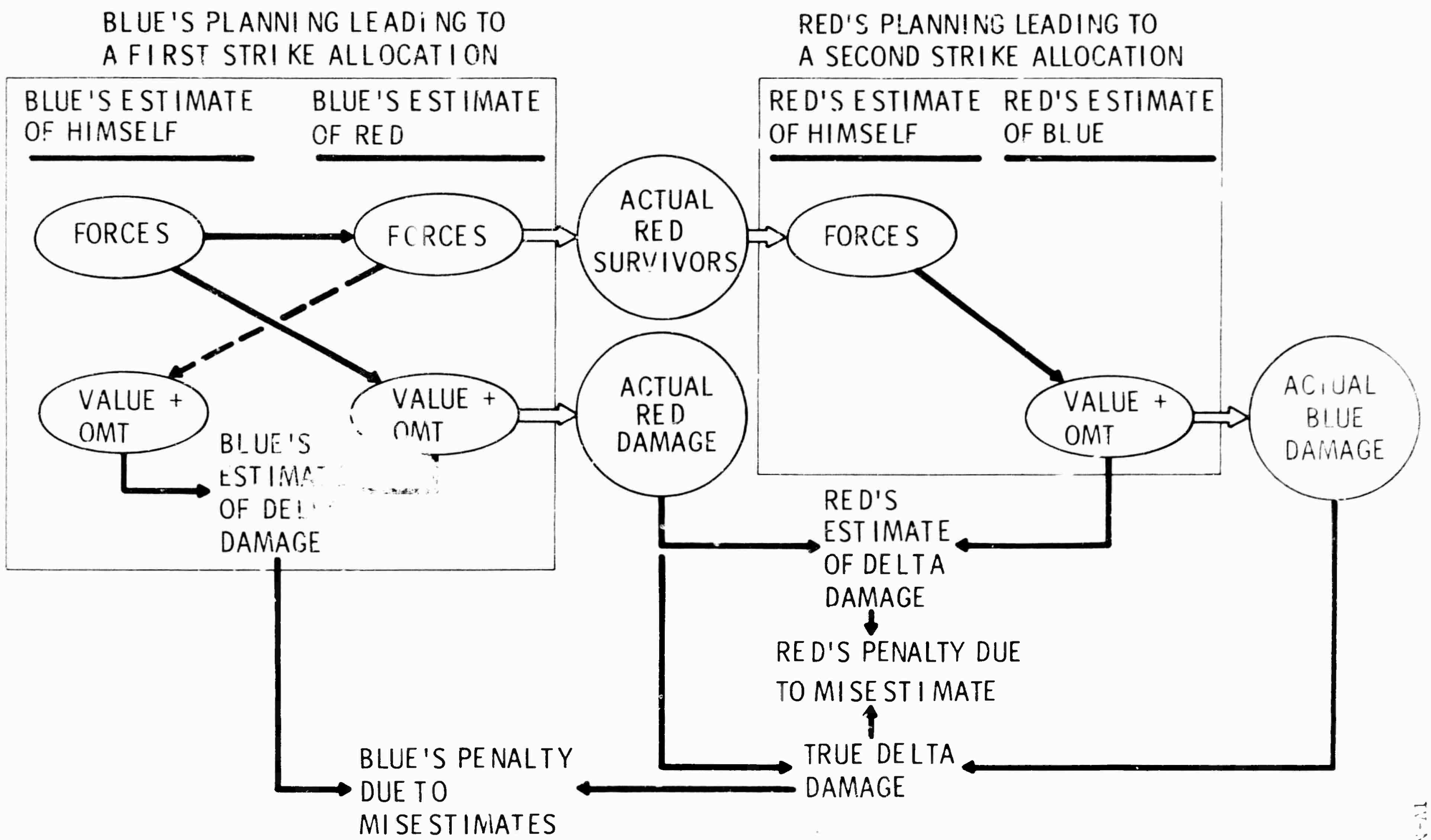


FIGURE N-1 MISESTIMATES ON A TWO-STRIKE WAR misestimates in the AEM is a raw measure for the scenario, opponents, estimates, and errors in estimates postulated. This measure might be improved if the allocation recognized some uncertainties in the estimates do exist. Hedging against these uncertainties would reduce the penalty in most cases but would also tend to decrease the expected damage of an allocation (should the estimates be correct).

2. Estimate of Estimates (Wavy-Wavy)

The concept of estimates of estimates is a natural growth of the misestimates capability discussed above. In essence, the assumption that my opponents' estimates are the same as mine is removed. However, my opponents' estimates are assumed to have singular values.

Figure N-2 shows a schematic of a two-strike war where estimates of estimates are used and misestimates are evaluated. The only significant change from Figure N-1 is in BLUE's planning. BLUE predicts a response based on assumptions other than he would make based on his knowledge and estimates. For example, if BLUE has deployed many terminal defenders of poor effectiveness but has designed his testing program and all other overt actions to indicate a good effectiveness, he might predict a retaliation based on the better defense. To maximize BLUE's value received, he must compute the value of RED's forces in terms of how he believes they will truly effect him. This computation is in two parts. Using BLUE's estimate of RED's view of all parameters to predict the response, BLUE uses his own best estimates to evaluate how this response would really affect the damage to his value and OMT targets (double arrow B). Once BLUE's estimate of

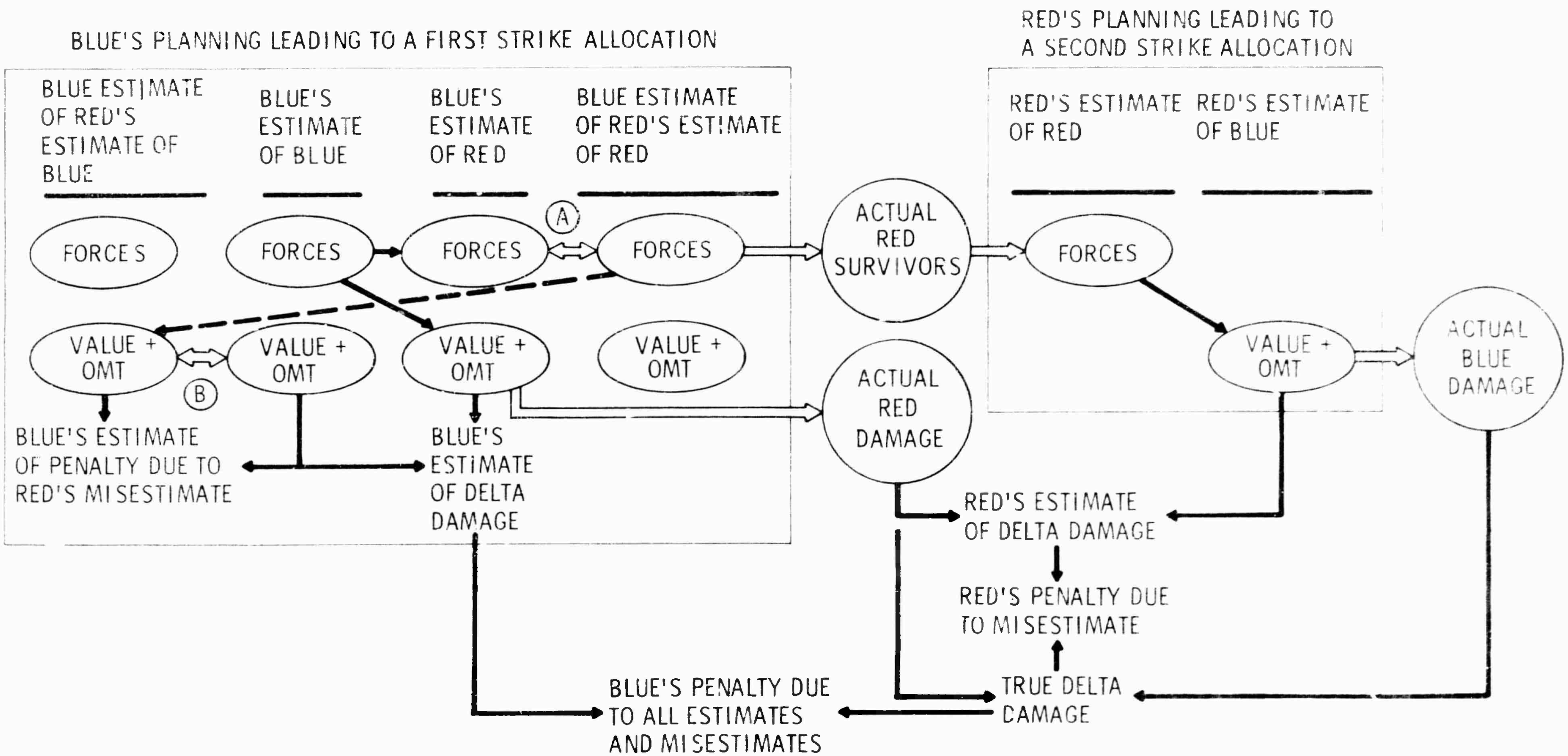


FIGURE N-2 ESTIMATES OF ESTIMATES WITH MISESTIMATES (TWO-STRIKE WAR) his damage suffered has been found, BLUE must estimate the BLUE value saved if a RED weapon base were destroyed (force target values). In the AEM, force values are based on the marginal utility of a weapon, i.e., the weapon lambda (see Section IV-C). Therefore, an effective weapon lambda must be computed. The technique for this computation is discussed in paragraph 3.a. of this section. This computation of the effective force values is denoted in Figure N-2 by double arrow A between BLUE's estimate of RED forces and BLUE's estimate of RED's estimate of RED forces.

a. Some General Applications of Wavy-Wavy

Some primary applications of wavy-wavy are values of intelligence ploys, values of covert changes in arsenal capabilities (especially in an arms control environment), and the effects of certain types of battle management information possessed by the opponent. The first two areas follow from the objective of wavy-wavy. My opponent will respond using different estimates than my own. If we hypothesize some control over my opponent's estimate of me, we might wish to know whether it is better to appear stronger or weaker than we truly believe we are. It is possible to determine the relative values of planting such information (if believed) using estimates of estimates. Similarly, the value of changes in capabilities which are assumed to be unknown by the opponent may be evaluated. The third area mentioned is not so obvious. However, the ability to assume the opponent does not possess precise knowledge of my reserve force composition is required (three or more strikes in the war).

The following paragraph explains the types of battle management information available in the AEM.

b. <u>Applications of Wavy-Wavy for Battle Management Analyses</u>

Using estimates of estimates, it is possible to correctly determine the effect of imperfect knowledge about the first strike attack. This knowledge (in the AEM) is important only if there will be (or there is expected to be) another strike by the initiator, i.e., a three-strike war. Consider three levels of information as follows:

1) Knowledge of which bases were emptied on the first strike (empty hole information).

2) Knowledge of the number of weapons by type sent in the first strike (attack level information).

3) Knowledge of the number of nuclear impacts suffered during the first strike (NUDETS information).

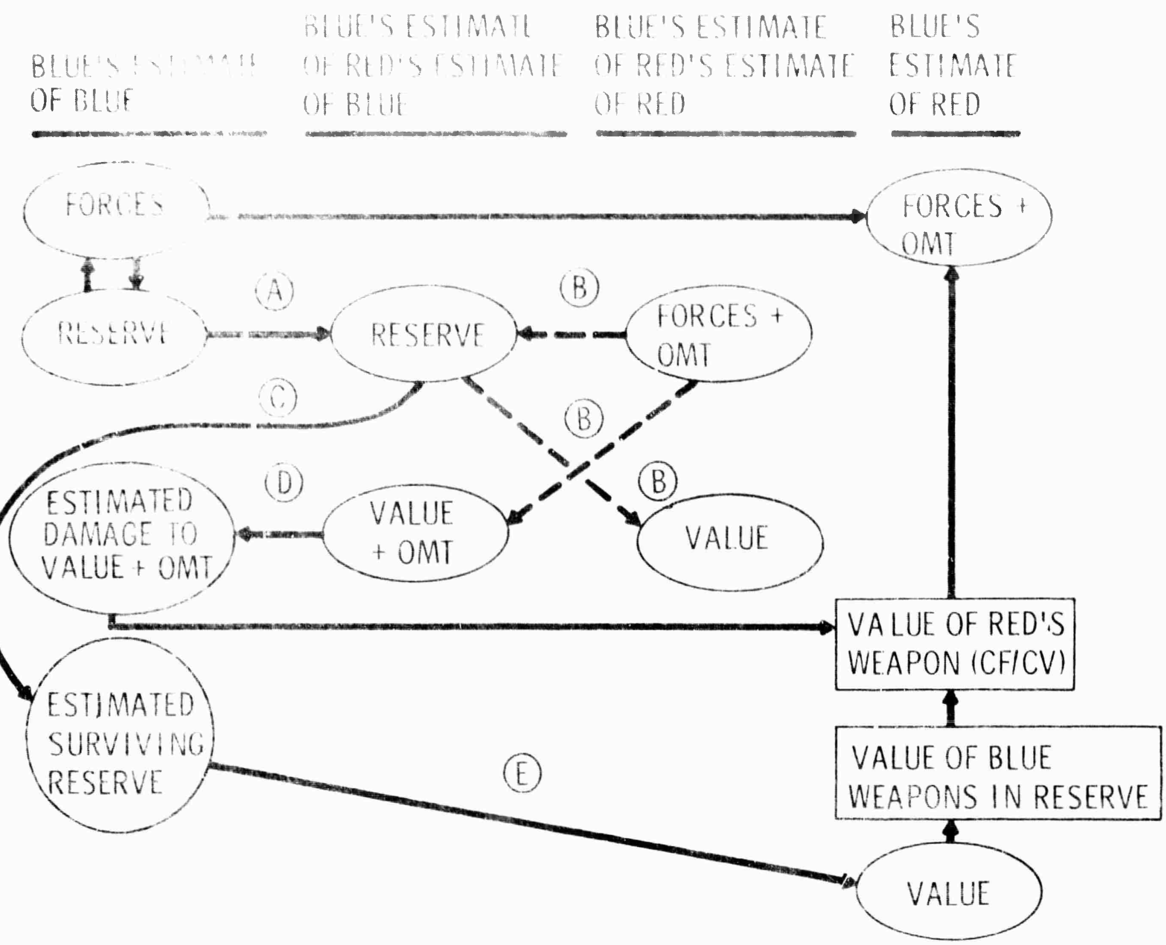
These are, in descending order, types of battle management information potentially gathered between an attack and a response. In the AEM, these are treated as inclusive sets, i.e., having any empty hole information implies complete attack level information, and having any attack level information implies complete NUDETS information. However, such information, as exists, may be incomplete. As pointed out previously, normal cases inherently assume perfect ALINFO, since the third strike weapons are assumed to be known by the opponent (his estimate is my estimate). Using wavy-wavy, other assumptions can be made to the lower limit where the retaliator only knows his survivors (no NUDET capability).

Figure N-3 illustrates the interactions of wavy-wavy on a three-strike war with no misestimates evaluated. Because the response is separated from the initial attack and the associated estimate, it is possible for the response to consider more (or less) weapons in the reserve than are actually there (arrow A). If BLUE believes RED has no capability to measure the attack in strike one, he assumes RED will behave as if all weapons are in the reserve holding RED's value targets at risk. Additionally, RED may be unaware of a covertly purchased weapon system which is assumed to be invulnerable to the response. Note that since BLUE's reserve may be attacked, BLUE must estimate the true value of his weapons to RED in the second strike prior to estimating the value saved by attacking RED weapons.

3. Techniques and Problems

The evaluation of estimated damage versus actual damage is primarily concerned with the kill functions themselves, since the allocation is not changed and all misestimate impacts occur in the kill functions. Therefore, the parameters which greatly influence achieved kill are generally more critical. Such parameters are reliability (or effective reliability modifiers by means of defenses or target capacity), target hardness and area, weapon yield and C.E.P., and terminal defense effectiveness.

Uncertainty in the number of targets of a type, e.g., empty



(A) BLUE ESTIMATES THE NUMBER OF WEAPONS RED WILL BELIEVE ARE IN THE RESERVE.

(B) BLUE'S ESTIMATE OF RED'S RESPONSE (INCLUDING CONVERGENCE TO A VALUE FOR BLUE'S RESERVE FORCE).

(C) ESTIMATED DAMAGE TO THE RESERVE USING BLUE'S BEST ESTIMATE OF RED CAPABILITIES.

(D) ESTIMATED DAMAGE TO BLUE VALUE AND OMT TARGETS USING BLUE'S BEST ESTIMATE OF RED CAPABILITIES.

(E) EXPECTED DAMAGE TO RED BY BLUE ON THIRD STRIKE.

FIGURE N-3 THREE-STRIKE WAR WITH RESERVE FORCE USING ESTIMATES OF ESTIMATES silos, hidden deployment of weapons, will also affect the answer. In this case, if more "targets" are attacked than truly exist, it is assumed that the targets attacked include all true targets. If fewer are attacked than exist, all those attacked were assumed to be true targets.

The value of a target may be different than estimated. While this is obviously true of force targets, true or corrected force target values are not normally computed (unless estimates of estimates are involved). However, if the plan was based on erroneous or obsolete data or on values which are arbitrary, non-retaliatory targets may have different "planning" and "true" values.

An allocation based on estimates may thus have kill functions, target values, and numbers changed and the "actual" expected damage caused by that allocation computed. The difference between the "estimated" damage and the "actual" damage [illegible] en the expected change caused by the discrepancy.

a. The Effect of Strategy Value Changes On Lambdas

When an error in kill functions or in target value is made, the resulting value received by a strategy is also changed. Similarly, the marginal utility of the weapon involved is changed. Since lambdas are used in the AEM for computations of force values, this change in lambda must be treated. If the error is a general one, involving all targets or all weapon types, averaging the slope of the correct kill functions for the strategies involved might yield a good approximation. However, the error might involve only one of several weapon types and then on only one target. Fortunately, by use of the multiplier matrix from the linear program, no heuristic rules for such cases are required. A brief synopsis of the linear program formulation in the AEM is now presented.

A linear program is composed of a set of matrices and vectors having initial and final definitions. We speak of a strategy matrix(es) which is composed of the amount of each particular resource (weapons, targets, budgets, etc.) used to gain the corresponding value (normally value destroyed) in the pay-off vector ($\overline{P}$). The set of those strategies selected by the programming process form a matrix, B, which is called the basis. All columns which are members of S but not members of B were not necessary to the solution. The AEM makes use of an identity matrix (I) to provide an initial basis. This may be interpreted as the initial feasible solution; i.e., use no resources, which results in a program value of zero. The initial construction is now shown as:

$\overline{P}$	$\overline{L}$	V
S	I	$\overline{C}$

where:

$\overline{P}$ is the pay-off matrix (vector).

S is the complete strategy matrix.

$\bar{L}$ is the marginal cost or lambda matrix (vector), initially zero, i.e., there is no pay-off for not firing weapons nor for not attacking targets.

I is the initial feasible solution matrix (an identity matrix which may be interpreted no resources used).

V is the scalar value of the matrix or linear program solution value.

$\bar{C}$ is the solution matrix (vector) initially equal to this constraint or resource value (since nothing is used).

If the simplex algorithm is used, after some iterative cycle (or pivot) k, which produces a better solution, the construction has been changed such that the feasible solution contains some strategies of the S matrix. The set of selected strategies is still identified by an identity matrix, although unordered. Additionally, the pivotal operations used transform the I matrix into the inverse of the basis. Thus, after the Kth pivot, which replaces a strategy in the basis with a new strategy, the resulting matrix I is the inverse of the new basis ($I_K = B_K^{-1}$).

Note that the I matrix and B matrix are square. Obviously, no more strategies can be in the basis than there are constraints. Just as important is the fact that precisely the same number of strategies will be in the basis as there are constraints. This means that some columns which were in the initial solution (I matrix) may also be in the final basis. This is the notion we call "GIVBAK" or weapons (and targets) not used in the allocation.

This inverse may be used to enter new strategies into the basis without modification of the original and is called the simplex method using multipliers or more simply, the multiplier technique (see Reference 5, Chapter 9).

If the basis is expanded to include the $\overline{P}$ vector and an identity vector (to square the matrix), the additional row in the inverse will yield the marginal cost (or lambdas in our case), i.e., $\overline{L} = \overline{P} \times B^{-1}$. In the AEM formulation, the inverse of the basis is saved. As may be seen, if the basis does not change, i.e., no strategy is changed, the values of $\overline{P}$ may be freely changed to yield effective lambdas, or:

$$\overline{L}^{*} = \overline{P}^{*} \times B^{-1}$$

where:

$\overline{P}^{*}$ is the new pay-off vector and

$\overline{L}^{*}$ is the resultant lambda vector.

One key concept of estimates of estimates is the ability to predict a response based on a $\overline{P}$ vector which is different than the initiator truly expects to exist. Therefore, while the $\overline{P}$ vector is used to generate the response, the $\overline{P}^{*}$ vector, which corresponds to the initiator's best estimates, may be used to generate effective lambdas ($\overline{L}^{*}$) for the purpose of generating force target values discussed in paragraph 2. of this section.

The effective lambdas produced by the above may very well be negative. This is not an obvious statement but may be easily understood in the following way. Assume two different weapons are selected to attack a certain target based on estimated values for $\overline{P}$. We further assume the P* value for one of the weapons is very low while the other weapon has a high P*. If the high P* weapon has a higher ratio of P* to number of weapons used on this target than on any other strategy involving that weapon, the weapon with the low P* value is preventing the maximum attainable damage by using some of the target resources and that weapon will have a negative lambda.

There are two points of view concerning what weapon lambda value to use in this case. The correctness of the method used must be conditioned not only on why a negative lambda exists but on the probability that it will continue to exist. For instance, if the low P* value occurs because of covertly deployed random defenses, the desired effect of low resultant error might be largely nullified if no attack on that system occurred. Similarly, if target area has been covertly increased by population dispersion, the low P* might result from a low yield weapon being improperly allocated. If so, no attack on that system would be desirable. In short, simply knowing a weapon possesses a negative lambda does not dictate that its value as a target is also negative.

The confidence in the supposed retaliation must also be considered. If the low P* value is caused by sending a bomber to a target having covertly deployed SAM defenses, an alternate allocation involving only missiles against this target might be an equally good plan based on the assumed estimates. If this were true, planning on a bomber assignment to the target hardly seems certain.

Additionally, if the retaliation arsenal includes separately targetable launched-on-warning weapons of if his plan is based on estimated prelaunch survival which is felt to be incorrect by the initiator, the resultant strategy set is no longer a square basis and cannot be directly inverted (see sub-paragraph 3-b).

With a view of the above considerations, AEM HEDGE estimates the value of one additional weapon by type as if that additional weapon were to be randomly added to an existing strategy employing that weapon type. This technique is not mathematically elegant but is guaranteed to produce non-negative weapon lambdas and still be sensitive to the conjuncture allocations for initiator and retaliator and assumed P* values for the retaliator. Thus:

$$\lambda_i = \frac{\sum_{S(i)} \left(VD(N_k + 1) - VD(N_k) \right) \cdot SOL(k)}{\sum_{S(i)} SOL(k)} \qquad (N-1)$$

where:

S(i) denotes the set of strategies which involve weapons of type i.

N is the number of weapons in the k strategy involving weapon type i.

SOL(k) is the number of times the k^{th} strategy was selected.

$VD(N_k)$ is the damage attributable to allocating N_k weapon of type i in the k^{th} strategy.

b. Misestimates of Prelaunch Survivability

The ability to misestimate prelaunch survivability has been added to AEM HEDGE. The impact on retargeting limits is discussed in Section IV-O; however, the mechanics of the procedure are presented here since misestimates are involved.

The procedure used requires that the retaliators' estimate of his strike assumes a slightly different connotation, i.e., the strike generated and printed is the total force plan and not necessarily the final allocation. This total force plan results in a partitioning of the targets into a group attacked by weapon type one, a group attacked by weapon type two, etc. If the retargeting limit is greater than one, the weapon type commander may adjust the actual strategies used against his particular group of targets. The resulting allocation is then evaluated to obtain actual expected destruction of all target groups. Figure N-4 illustrates this program sequence.

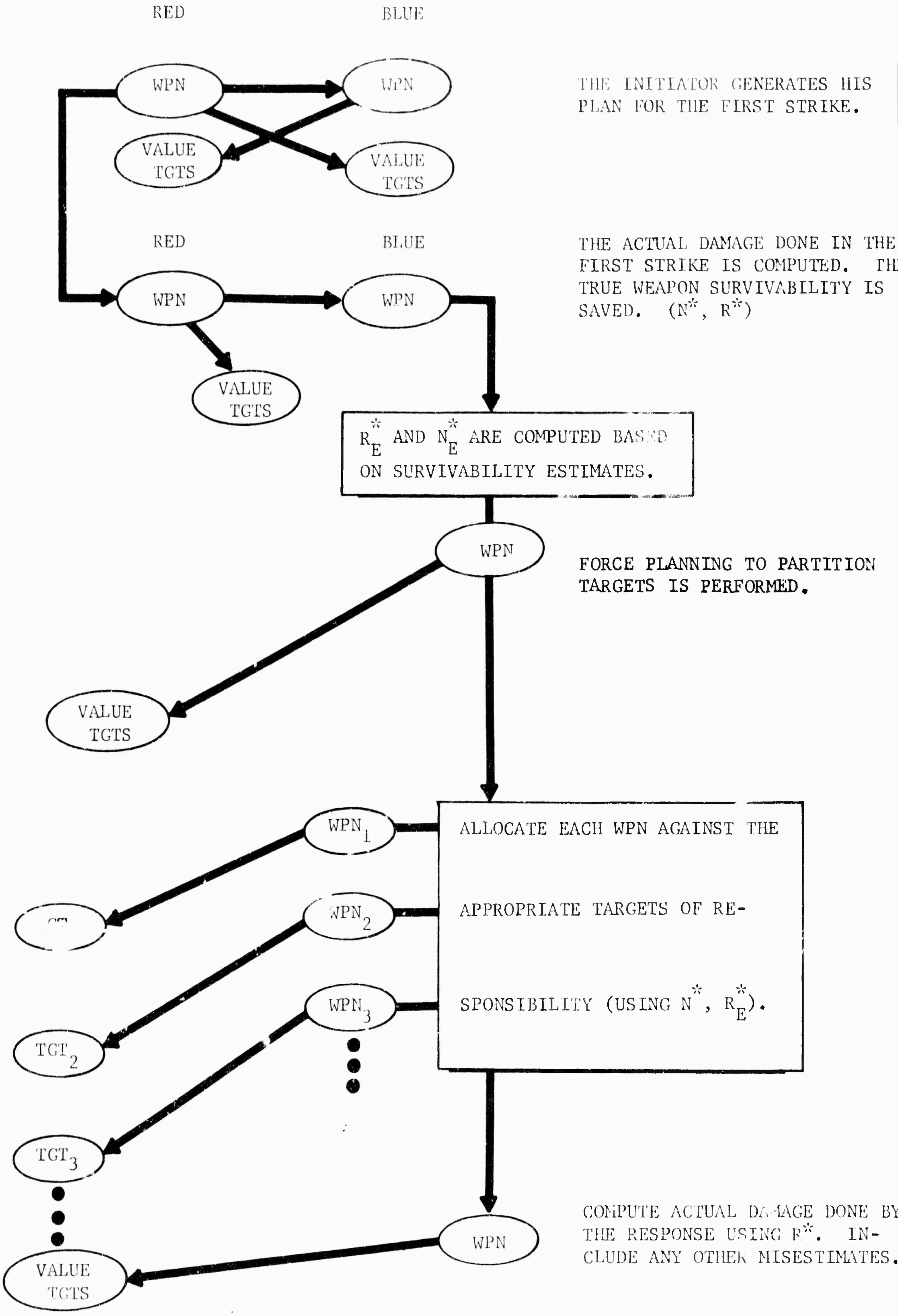


FIGURE N-4: PROCEDURE FOR EVALUATING PRE-LAUNCH SURVIVABILITY MISESTIMATES

Each target grouping reflects only those targets attacked by that particular weapon type in the total force plan. Thus, targets not attacked in the total force plan cannot be attacked by the weapon type commander. Nor, can targets which belong to one group be attacked by any other weapon type. In addition, the retargeting level is assumed at the most convenient demarkation level in AEM, i.e., by weapon type. It seems more realistic that wing commander or force component commander would do the retargeting (e.g., retargeting within a group of fifty missiles, or retargeting the entire Minuteman force) but such an option is not natural to AEM.

This technique also is based on the assumption of only pure strategies being in existance. Thus, cross-targeting like occurs with hedges is not allowed. It is anticipated that this restriction will be removed in the near future.

c. <u>Misestimates of the Number of Separately Targeted Weapons Which Are Launched on Warning</u>

The capability of assuming the particular warheads which will be launched on warning has also been added to AEM HEDGE. These weapons are pretargeted and assume invulnerable. However, if the number assumed is not correct, the penalty is assessed by use of the procedure used for prelaunch survival misestimates.

Consider the case where the actual number launched on warning (OA) is less than assumed (OE). This implies a change in the available vulnerable warheads per base from those estimated (WE) to the actual (WA). If we assume the base survival (PS) was correctly estimated, we still must consider the effect on both the vulnerable and invulnerable components of this weapon system. It is not essential that the same number of weapons be available in the estimate (OE + WE) as in the actual (OA + WA). The technique is designed for changes in the number allocated regardless of the cause.

For the targets assigned to the OE weapons, no strategies are changed. However, an effective reliability modifier ($RMOD_0$) is computed as:

$$RMOD_0 \ (OA < OE) \ = \ \frac{OA + (OE - OA) \cdot PS}{OE} \qquad \text{(N-2)}$$

This $RMOD_0$ is then considered as an increased uncertainty arrival for the OE weapons in evaluating the actual damage.

For the vulnerable weapon, if OA + WA = OE + WE, there is no change in strategies or estimated damage. However, WA is equated with WE to insure proper bookkeeping (the launched on-warning portion of the weapon system will have WA - WE vulnerable warheads per base). If OA + WA ≠ OE + WE, there is an implied change in the number of vulnerable weapons to be allocated if a retargeting capability exists. A new actual available warheads per base is calculated as WA' = OA + WA - OE. These WA' warheads are assumed to have the same aim points as the WE weapons assumed. This is equivalent to a change in N as discussed in prelaunch survivability estimates.

Let us consider the case of 1000 warheads having an assumed availability of .8 with 400 warheads estimated to be separately targeted and launched on warning (OE = WE = 400). If the actual number launched on warning were only 200 (OA) and the base survival were .5 (PS), then all strategies involving the pretargeted OE weapons would be evaluated with an increased uncertainty of arrival of:

$$RMOD_0 = \frac{200 + (400 - 200)\ .5}{400} = .75$$

Additionally, if the actual weapon availability is .6 instead of .8, the true number of vulnerable weapons to be allocated is:

$$WA' = 200 + 400 - 400 = 200$$

instead of the 400 assumed in the estimates. However, the effective reliability due to retargeting limits would not change.

If OA is greater than OE, there is no change in the separately targeted OE weapons. However, there is an implied change in PS for the vulnerable weapons (PS*):

$$PS^* = \frac{WA \cdot PS + (OA - OE)}{WA + (OA - OE)} \qquad \text{(N-3)}$$

Using this PS*, a new N* is computed and used in accordance with the plan depicted in Figure N-4.

Thus in our example, if OA were truly 600, the effective PS for the vulnerable weapons would be:

$$PS^* = \frac{WA \cdot .5 + (600 - 400)}{WA + (600 - 400)}$$

If no availability error is made:

$$PS^* = \frac{200(.5) + 200}{400} = .75$$

and if the actual availability is .6:

$$PS^* = \frac{0(.5) + 200}{200} = 1.$$

But only 200 weapons may be allocated instead of the estimated 400 since all available (600) warheads are launched on warning with 400 being sent to the targets assigned to the OE weapons.

This analysis could be expanded to misestimates of numbers of weapons to allocate due to any reason and without specifying a launch on warning capability. However, before this development occurs, the limitations of the technique which assumes pure strategies and retargeting by weapon type should first be removed to provide a more general and compatible technique. Such developments should be considered in the near future.

d. Restrictions and Limitations

All allocations are evaluated on singular valued data for the primary weapon if more than one weapon type is sent to a target. This assumption may cause an undue emphasis on the penalties projected if only the primary weapon is degraded.

The scenario cannot be violated, i.e., number of strikes to be planned and which side initiates the conflict. The first restriction is largely due to the structure of the model, but can be removed by use of separately targeted weapons which are launched on warning. Otherwise, once an allocation is made, all factors which preclude a safe arrival are treated as reliability which may be vastly different than the reliability used to generate the allocation (or plan).

It is important to note that the area target kill function formulation used in the AEM is based on optimal aim points for the uncertainties involved (reliability, C.E.P.). We have no algorithm for the effect of poorly chosen aim points nor for defenses which deny certain aim points; therefore, the "true" damage functions reflect optimal aim point corrections even though the characteristics which lead to those aim points were not known. This is of course inconsistent with the intent of misestimate (wavy) and estimates of estimates (wavy-wavy) analysis since no errors in aim points are permitted.

This deficiency should be considered in light of other assumptions made in the model. Area targets are basically considered flat disk with uniformly distributed value (the Q-95 option allows a circular normal value spread with a sigma radius). In reality, the value being attacked is more likely located in known or suspected pockets throughout a noncircular area. The AEM kill functions have been shown to closely approximate national blast fatalities found by detailed grid lay-downs. It is therefore suspected, but not proven, that the aim points inconsistency is probably more academic than real. Analyses to illuminate and resolve this apparent inconsistency have been suggested and are being planned.

Another limitation is also apparent within the Prim-Read terminal defense doctrine. The current formulation of this doctrine implies knowledge by the defense of the expected damage function attributable to an arriving weapon. Therefore, the defense assigns interceptors to incoming weapons in such a way as to delay penetration until defense kill is imminent. Since the defense characteristics are allowed to be different by attacking weapon type, the defense is assumed to know which weapon type is attacking; hence, the defense has perfect knowledge of attacking weapon type.

4. Summary

The AEM has the ability to evaluate an allocation against different data than produced the allocation. It is therefore possible to compute the difference in expected value destroyed due to data differences. By interpretation of data into "estimates" for purposes of allocation and "actual" for purposes of evaluation, a method of computing the penalty due to misestimates has been developed. Similarly, by allowing different data to be used during a retaliation (the initiator's estimate of the retaliator's estimates) than the initiator "estimates" data, the ability to compute the value of knowledge about the opponent's knowledge has been developed.

The evaluations are based on single valued data which is always assumed to be correct. The allocations are optimal for the particular parameter values and hedges assumed. If hedging allocations are used, only the damage attributable to the primary weapon is evaluated. Additionally, if hedges cause cross-targeting, prelaunch survival estimates and misestimates of separately targeted weapons which are launched on warning will not be properly performed. The removal of these restrictions is anticipated.

O. WEAPON RETARGETING

1. General

One physical characteristic of offensive weapons that has not been discussed to date is the limitation on retargeting of weapons after an enemy strike. Basically, most weapons have guidance systems which can easily direct an individual weapon to any one aim point out of a finite group of aim points which were selected prior to the initiation of a war. A requirement to go to an aim point that was not preselected can generally be satisfied only at the expense of considerable time delays. This feature of redirecting a weapon to some aim point we call retargeting.

The whole topic of modelling the impact of target capacity limitations is very complex and gets into very detailed combinatorial analyses. Feeling that such detail is inappropriate to AEM, modelling of the approximate effect of target capacity limits has been done. This chapter discusses the form of that modelling today.

2. End Point Conditions

As an introduction into the concept of modelling the effect of retargeting limits without modelling the details, consider the impact of having no ability to retarget in contrast to the impact for unlimited retargeting.

Consider the case where each weapon has only one target, which must be set before an attack occurs and cannot be changed after the attack. How would this limitation affect the original targeting of the weapons and their utilization in retaliation? If an estimate of the probability of survival, S, for the weapon existed, the fact that a weapon might not survive can be viewed as an additional component of reliability.

It will be assumed that, by definition, reliability is the probability that a weapon will be launched and detonated within some distance, as described by a CEP, of its target. Furthermore, the unreliable weapons are not known until launch time and it is assumed that the assigned targets for the unreliable weapons cannot be shifted to a reliable weapon. Thus, a reliable weapon is a weapon that performs its assigned function with some probability, R.

If not surviving an attack is viewed as another way to fail to launch then, by the above definition of reliability, the probability of survival, S, is another component of reliability. Thus, if the standard reliability is R, the effective reliability, R^* would be:

$$R^* = S \cdot R$$

If this effective reliability is used in the original targeting, the maximum utility of the unity target capacity will be attained even though retargeting cannot occur.

For this case of a one target limit, AEM would therefore operate as follows. The initial strike allocations would occur and the resultant probability of survival, S, computed for all weapons. The new effective reliability, $R^* = R \cdot S$, for each weapon would be computed and the retaliation allocations would be conducted using those effective reliabilities and a number of weapons equal to the original number possessed.

The above process accounts for the fact that the side being attacked cannot retarget but that he would allocate targets before the exchange to account for the possibility that any weapon might not survive, with probability S. Implied is that he can estimate perfectly what value S is.

Consider now the case where a very large target capacity exists. (For this discussion, a very large number can be considered to be the total number of weapons.)

In such a case no matter which weapons survive retargeting can take place so that weapons surviving can be targeted to the exact targets desired.

For this case, AEM would operate as follows: The initial strike allocations would occur and the resultant expected number of survivors computed for each weapon. Leaving reliability at $R^* = R$, the computed expected survivors would be allocated in retaliation.

This process accounts for the fact that the side being hit can retarget after an attack so any surviving weapon can go to any target desired.

Consideration of these two extreme cases points out an interesting relationship. In summary, for the case of capacity = one, if you have N weapons of reliability R and probability of survival after an attack of S, the inability to retarget can be considered the equivalent of retaliating with N weapons of effective reliability $R^* = R \cdot S$. For the large target capacity case, complete retargeting can always occur and this is equivalent to retaliating with an effective number of weapons $N^* = N \cdot S$ with reliability R.

Thus, even though the expected number of survivors is always $N \cdot S$, the possession of retargeting capability is summarized in the concept of having an effective number of weapons with an effective reliability to retaliate with.

For target capacity limit of one, the effective number is N and the effective reliability is R · S. For a large target capability, the effective number is N · S and the effective reliability is R. These effective numbers completely account for the fact that the si retaliating will do his best to utilize whatever capacity he has.

In summary, consideration of these end points has led to the following relationships.

First, consider the definitions as previously described:

T = target capacity (each weapon)

N = number of weapons before attack

S = attack survival probability (each weapon)

R = weapon reliability

N^* = "effective" number of weapons

R^* = "effective" reliability

The results of the study of the extreme cases T = 1 and T = are then as follows:

T	N^*	R^*
1	N	R·S
∞	N·S	R

These cases are pleasing to treat since it is possible to define here just what N^* and R^* are.

It is implied that the targeting takes place in two phases - once before the attack, and then a certain amount of reshuffling afterward (depending on T).

For T = 1, no reshuffling is allowed, and the uncertainty of which weapons survive the attack must be taken into account in phase I targeting.

For $T = \infty$, complete reshuffling is possible, and the real targeting is done exclusively in phase II, using survivors of the attack.

In both of these cases, N^* is the number of weapons targeted and R^* is the uncertainty facing the targeteer at the time.

3. General Limits of Target Capacity

For the intermediate cases ($1 < T < \infty$) things are not so clear. It seems reasonable to assume, however, that since both extremes can be accurately dealt with by modification of numbers of weapons and their reliability, the intermediate cases be handled in the same fashion. Thus, we choose to describe N^* and R^* simply as "effective" numbers which will result in the correct optimum expected damage for intermediate target capacities, though they may or may not have real interpretations.

Since correct damage is the goal, it makes sense to preserve a heretofore unnoted property of the extremes:

$$N^* R^* = N \cdot R \cdot S \qquad (0\text{-}1)$$

that is, that the reliable surviving weapons are the same as the expected value, whatever the value of T.

The job of targeting is to establish a correspondence between weapons and targets in such a way that effective use of the weapons is guaranteed. Consideration of the target system is a pertinent aspect of this process, and is currently handled in AEM through the Lagrangian. If there is some obstacle preventing a weapon from reaching a valuable target (e.g., low reliability, poor CEP, defense), then doubling up will occur in the final allocation as long as a reasonable marginal return per weapon is achieved.

In finite target capacity cases, however, a new ingredient is injected into this process - namely, that there is some probability (after an attack on weapons) that a valuable target will be unreachable by the surviving weapons due to their retargeting limitations. Since this is, after all, the weapon's fault (and not the target's), it is proper to degrade weapon capability.

If we define an aim point as being above the terminal defense of a target (if any exist), we may declare that an aim point is covered by an arriving warhead. We further assume that aim points are independent and unique. Therefore, an aim point is covered if a warhead arrives at that point whether any other warheads arrive. By this definition, a maximum of N·R·S aim points can be covered. The question is how many of these points are indeed targeted.

Each weapon is assumed to have T aim points which are prestored in its guidance. We will assume these are unique destinations and that there exist T weapons possessing each possible aim point. Therefore, an aim point may be covered with probability R if at least one of the T weapons possessing that aim point survives. We further assume these weapons are independently survivable. The probability that an aim point is covered by the surviving weapons (Pc) is thus:

$$Pc = 1 - (1 - S)^T \qquad (0\text{-}2)$$

We may now state that the total probability that an aim point is attained (R^*) is the probability that the aim point is covered by the survivors times the probability that a launched weapon reaches the target, or:

$$R^* = R \cdot \quad 1 - (1 - S)^T \qquad (0\text{-}3)$$

This R^* may be described as the effective reliability of the weapon type. We now must address the effective number of weapons of this type (N^*) which are allocated.

We note that the actual aim points covered by the surviving force is N·S. The effective number required to cover these aim points is $N^* \cdot Pc$. If in fact N·S, aim points are selected by the targeteer, then:

$$N^* = \frac{N \cdot S}{Pc} \qquad (0\text{-}4)$$

This assumption has been made in AEM. Note that the property of the aim points described in equation (0-1) is preserved and that equation (0-4) is smoothly behaved between these extremes. No proof is currently offered that this assumption is optimal.

4. Impact On Force Values

In a manner similar to the effect of random defenses, a limited retargeting ability by a weapon does have a measurable effect on the force value to be placed on that weapon type. For example, if a certain weapon type has a target capacity of one, then destroying one of those weapons does not change the number of "effective" retaliating warheads, instead it modifies the "effective" reliability of the weapons. And, all of the weapons of the type have their reliability changed. If a value is placed on one such weapon when it is a force target, that value must reflect the true impact resulting from destruction of one such target.

In the general case, destruction of a single target of some weapon type will change the reliability and number of "effective" retaliating warheads. If we use equations (0-3) and (0-4) to describe the effective retaliation forces, then we can derive:

$$\frac{\Delta R^*}{\Delta NTD} \approx \frac{\partial R^*}{\partial S} \cdot \frac{\partial S}{\partial NTD} \qquad (0\text{-}5)$$

where:

$\frac{\Delta R^*}{\Delta NTD}$ = Change in reliability of all weapons of a type when one force target of a type is destroyed.

$\frac{\partial R^*}{\partial S}$ = Change in reliability for a unit change in probability of survival.

$\frac{\partial S}{\partial NTD}$ = Change in overall probability of survival when one more target of a type is destroyed.

If we assume, for simplicity, that there are N targets, then we have

$$S = 1 - \frac{NTD}{N} \qquad (0\text{-}6)$$

where:

NTD = Number of targets destroyed.

It follows then that

$$\frac{\partial S}{\partial NTD} = - \frac{1}{N} \qquad (0\text{-}7)$$

and, using (0-3)

$$\frac{\partial R^*}{\partial S} = RT\ (1 - S)^{T-1} \qquad (0\text{-}8)$$

and

$$\frac{\Delta R^*}{\Delta NTD} \approx - \frac{RT}{N}\ (1 - S)^{T-1} \qquad (0\text{-}9)$$

In a similar fashion, we can derive:

$$\frac{\Delta N^*}{\Delta NTD} \cong \frac{N^*}{S} \cdot \frac{\partial S}{\partial NTD} \qquad (0\text{-}10)$$

where:

$\frac{\Delta N^*}{\Delta NTD}$ = Change in effective number of retaliating warheads when one force target of a type is destroyed.

and

$$\frac{\Delta N^*}{\Delta NTD} \cong \frac{1 - (1 - S)^T - S(1 - S)^{T-1}(-T)}{\left[1 - (1 - S)^T\right]^2} \qquad (0\text{-}11)$$

Now, the appropriate value to be placed on this weapon type can be computed using equations (0-9) and (0-11). We know that changing a retaliatory force by one warhead reduces damage by λ units (see Section IV-C). We also know that, for a fixed force, the loss in retaliatory damage can be computed when there is a small change in reliability (see IV.G.e.).

Thus, the value to be placed on a force target can be computed as:

$$V = \frac{\Delta N^*}{\Delta NTD} \cdot \lambda + \frac{\Delta R^*}{\Delta NTD} \cdot \frac{\Delta VD}{\Delta R} \qquad (0\text{-}12)$$

where:

V = Force target value

λ = Marginal weapon utility

$\frac{\Delta VD}{\Delta R}$ = Delta change in retaliatory damage for Δ R change in reliability.

It is interesting to note that $\frac{\Delta N^*}{\Delta NTD} \rightarrow 0$ as $T \rightarrow 1$ and $\frac{\Delta R^*}{\Delta NTD} \rightarrow 0$ as $T \rightarrow \infty$. Thus, the two terms in equation (0-12) have a different impact, depending upon the exact retargeting capability.

The process used to compute force values for any given situation would then operate as follows:

1) Side attacking the weapons uses arbitrary initial force values.

2) Side possessing the weapons uses the relationships of (0-3) and (0-4) to lead to his retaliation.

3) After the completion of the retaliation in step (2), there will be available weapon lambdas (λ in (0-12) and it is simple to compute the resultant small change in retaliatory damage, ΔVD for some small change in weapon reliability, ΔR. Using these results and relationships (0-9), (0-11) and (0-12) will lead to new force values.

5. Prelaunch Survivability Estimates

The preceding discussions have described the targeting philosophy for some survivability factor S. If the survivability is known prior to targeting, this philosophy has been shown to be correct at the end points where a target capacity of one implies only the survivability is known and an essentially infinite target capacity where the particular surviving warheads are known. A more realistic case involves targeting based on an estimated survivability (SE) which may not correspond to the actual S.

To model the effect of such estimates, two levels of targeting have been assumed. The first level involved total arsenal targeting in accordance with survival estimates for each weapon type which occurs before the war starts. This allocation process, denoted as plan generation in AEM HEDGE, determines the targets a particular weapon type may attack and candidate strategies to be used. (If target capacity is one, the candidate strategies will be used.)

The second level of targeting is performed by the weapon type commander once the attack has occurred. This targeting allows some adjustment of the strategies as generated by the arsenal plan. This adjustment is made within the targets assigned to the weapon type commander and within the retargeting capability of his particular weapons. However, the aim points of his weapons are not allowed to change, i.e., he retargets with improved knowledge of his survivors but can only change the number directed to prestored aim points which were determined by the arsenal plan.

This procedure is approximated in AEM HEDGE by using the estimated survivability (SE) to compute an R_E^* and an N_E^* which are used to generate the plan. The true survivability (S) is then used to compute the true N^*. These N^* weapons at reliability R_E^* are then retargeted against the same aim points denoted by the targets previously assigned. Then, an evaluation of the strike damage is made under the assumption that the N^* weapons really had a reliability of R^*.

That is, if A aim points were originally assigned to a particular weapon type:

$$A = R_E^* \cdot N_E^* \qquad (0\text{-}13)$$

These aim points must be maintained during any post attack retargeting. Therefore, on the average, some fraction of A, A^* will be retargeted.

$$A^* = \frac{N^*}{N_E^*} \cdot A = \frac{N^*}{N_E^*} \cdot R_E^* \cdot N_E^* = N^* \cdot R_E^* \qquad (O-14)$$

The above process identifies the aim points, A^*, and then evaluates the true damage for those aim points.

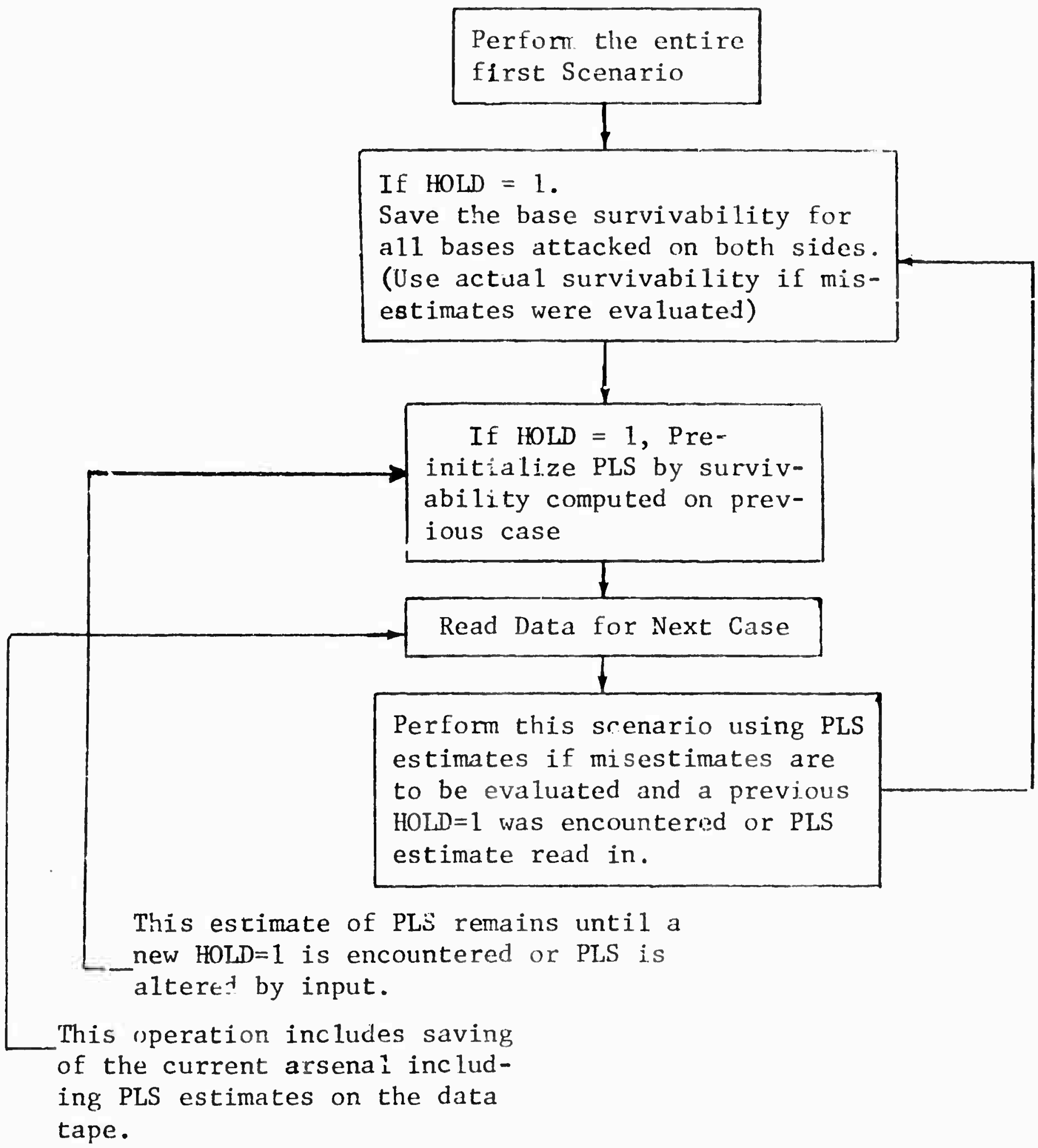
This retargeting technique is precise if survivability is correctly predicted since $S_E = S$ produces $N_E^* = N^*$ which implies $A^* = A$. If $S_E \neq S$, N_E^* is different from N^* only if a retargeting capability exists. If a sufficiently large retargeting limit is considered, $R^* = R_E^* = 1$. Therefore, the number of aim points is changed from N_E^* to N^* with the further limit that no new target can be added to the list.

The current formulation of estimated survivability does not allow for contingent targets should N^* be greater than N_E^*. Nor can hedges currently be used in conjunction with misestimated survivability. Hence, the effect is probably over emphasized in this formulation. The latter restriction will most likely be removed in the near future.

6. Internally Computed Prelaunch Survivability Estimates

Prelaunch survivability has always been computed internally in AEM for all strikes when forces may have suffered a previous attack. In the case where an estimated survivability (SE) occurs, there is a question of proper procedures for arriving at SE. With this in mind, a procedure has been provided for analyst aid in developing SE.

This procedure involves computing a prelaunch survivability internally in one scenario and using that result in later scenarios. The primary control for this option is a variable HOLD which pre-initializes the prelaunch survivability for the next case as the one computed for this case. This allows the estimated S to be saved on a data tape if desired for later non-concurrent analyses. The program flow is illustrated below:



7. Summary

It must be remembered that this effort has been concerned only with modeling the effect of retargeting limits and survival estimates and is not a detailed, rigorous formulation. However, it has been shown that the effect is precise at the extremes of retargeting limits and is smoothly behaved between these limits. Additionally, retargeting at the weapon type level may be employed to investigate the effects of mis-estimating survivability. The latter procedure involves retargeting within reselected aim points which are generated by the estimated survivability.

P. GENERATION OF HEDGING ALLOCATIONS

1. The Problem

The Arsenal Exchange Model has traditionally been able to deal with measurement of the impact of uncertainties. (This capability is described in an earlier chapter, IV-N, Estimates and Misestimates.) The traditional approach centered upon the notion of strict evaluation of penalties due to uncertainties, rather than development of allocations which would minimize the penalties. The philosophy was based on the idea that understanding of penalties could lead to actions designed to reduce uncertainties.

There is another view of uncertainties, however, which is probably of equal importance. This view, which will be addressed in this chapter, is that some uncertainties cannot be eliminated, or even reduced, and allocations should be developed which would minimize the impact of such uncertainties.

Considerable work has been done by many individuals in problems like that encountered when dealing with uncertainties in strategic allocations. Review of that work reveals a fairly basic assumption that is unrealistic for this application, however. Most often, one finds that a necessary ingredient to the development is an assumption that probabilities, or probability density functions, are available to describe the uncertainties. For example, if event A is an uncertain event, most approaches require specification of P_A, where P_A is the probability that event A will occur.

In the context of strategic analyses, like those AEM is involved with, such probabilities are impossible to develop. For example, take the case where event A is the situation where all ICBMs fail because of some previously undetected defense mechanism. In such a case it is impossible to arrive at any rational probability of event occurrence.

Most often there is a finite, but small, number of events which are of very low liklihood but whose occurrence could be catastrophic. In such a case the analyst would like to develop allocations which minimize sensitivity of the allocation payoff to the occurrence of one, or more, of the events. Such allocations can be called hedging allocations.

The AEM concept for dealing with such low probability events is to require that an analyst specify in some quantitative manner what result he would accept as "adequate" if such an event did occur and the allocation was not optimized exactly for the event. Specification of such goals results in placement of side conditions, or hedges, that the allocation must try to satisfy.

A hedging condition should be considered as a specification of some requirement or goal to be met for the specified side whenever that side is conducting an allocation of weapons to targets. These conditions can, at the present time, fall into three separate categories as follows:

1) A condition on the total damage achieved on a specified set of target classes by a specified set of weapon types.

2) A condition on the total number of weapon RV's allocated against a specified set of target classes by a specified set of weapon types.

3) A condition on the total number of targets hit by a specified set of weapon types over a specified set of target classes.

In most cases such hedging conditions cause the allocation to give up some damage in order to satisfy them. In that sense, a hedged allocation is usually "non-optimum" by a max damage criteria but the allocation will meet the specified hedging conditions if at all possible, and it will do so at a minimum penalty to the max damage objective.

In order to demonstrate what is meant by a hedge, consider the following examples: (Not in AEM input format.)

Example 1: Have Side 2 attain at least 250 units of value destroyed on value targets and have weapon type 7 be the only type of weapon achieving the damage.

Example 2: Allocate no more than 175, Side 1, ICBM RV's against OMT targets.

Example 3: Allocate Side 1 weapons so that if all missile pen-aids fail the damage received from the allocation would be at least 500 units.

These examples demonstrate certain kinds of side conditions which the analyst wants to insert so he can influence the sensitivity to some uncertainty.

Example 1 might have been generated because weapon type 7 is a submarine type that is very invulnerable and allocation of that type to cities guarantees adequate assured destruction in the event that side 1 conducts a pre-emptive attack on side 2.

Example 2 might have been generated in order to force a mixed attack on side 2 military targets. This hedge would do so in an indirect way by limiting the number of ICBM weapons launched at such targets.

Example 3 basically is a hedge against the event where all pen-aids fail. No probability of the event is available, but the analyst would be satisfied with an allocation that achieves some "adequate" level of damage if the event does occur.

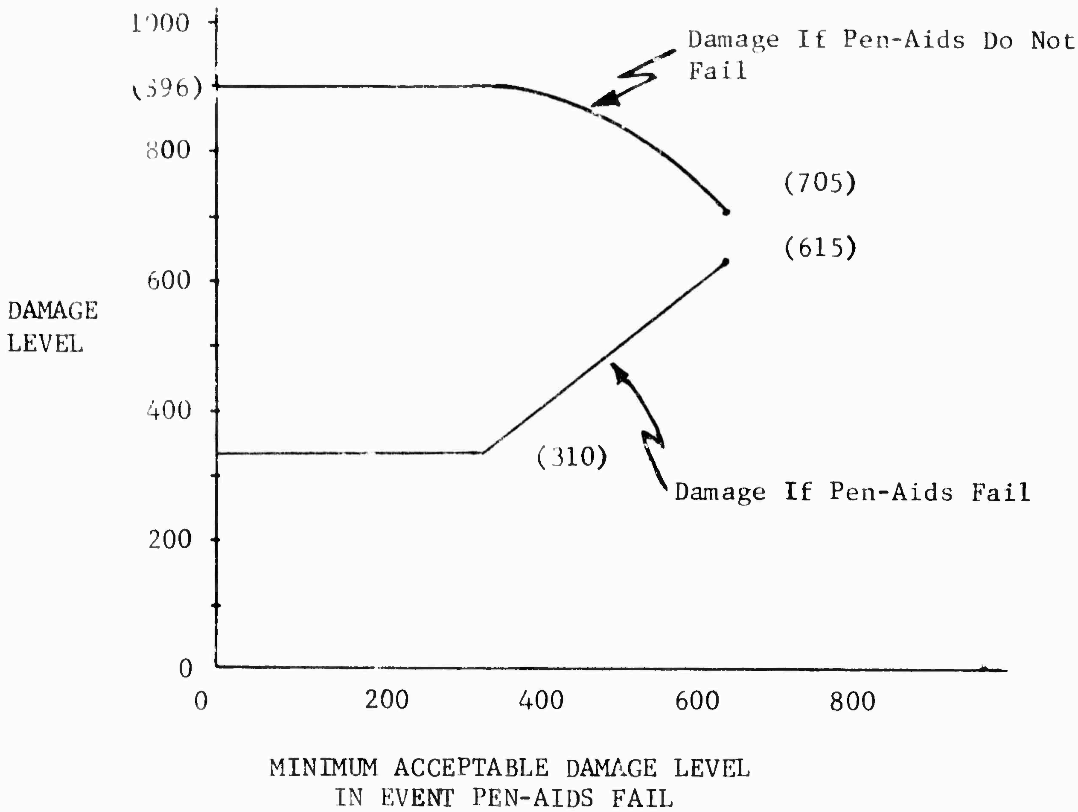
This latter concept of an "adequate" level of damage can best be visualized if one considers more fully the situation of example 3. Assume, for an example, that side 1 can allocate for the condition where his pen-aids work and he knew they would work or for the condition where they fail, but he knew they were going to fail. Two such allocations might result in 896.1 and 615.3 units of payoff, respectively for the work/fail events. These two allocations could then be evaluated to see how much payoff would occur for the opposite event. Such evaluations might yield the following:

ALLOCATION BASED ON PEN-AIDS	ACTUAL EVENT WAS: PEN-AIDS WORKED	ACTUAL EVENT WAS: PEN-AIDS FAILED
WORKING	896.1	310.0
FAILING	705.3	615.3

This table demonstrates a basic point. If the allocation was based on the pen-aids working, and they failed, the result (310.0) is far worse than if the assumption originally had been that the pen-aids would fail (615.3). This is usually an inherent characteristic of allocations which are based on single measures of effectiveness--namely that some low probability event can cause catastrophic drops in payoff.

In this example the desired hedging allocation would have the characteristic that at least 500 units of damage would be achieved if the pen-aids failed, even though the primary assumption was that they would not fail. Such an allocation might, for example, result in 815.0 units of damage if pen-aids worked and 500 units if they failed. This is not as good, in either event, as the best possible but it does minimize to some degree sensitivity to the event occurrence.

The analyst has basic control by his specification of the "adequate" damage value. In this case, where 500 damage units was defined as adequate, a certain allocation would result. If the 500 was raised to 600, the allocation would shift toward emphasis of the pen-aids event. All possible inputs for the desired goal might result in allocations whose damage plots as follows:



By development of this whole plot the analyst could understand completely the implication of any definition he is interested in for "adequate".

The analyst might feel that he would never accept anything as low as 500 units of damage. In such a case this type of plot could immediately tell him what goal to require for the pen-aids fail event and it would also show how big a penalty that goal is imposing if they do not fail.

In summary, a hedging allocation is an allocation generated on the basis of maximizing against a primary objective while, at the same time, satisfying a number of side conditions or goals. The basic intent of such allocations is to arrive at an allocation which has no "Achilles Heel" where certain specified events would result in unacceptable payoff levels.

2. The Methodology Base

The mathematical structure for development of hedging allocations is available in AEM because of the linear programming (LP) approach used in all allocations. (See Chapters IV-A and IV-D.) The presence of an LP offers the opportunity to insert miscellaneous constraints, or conditions, into the allocation process in a very matural manner. The way this works out in the case of hedges will now be addressed.

Basic understanding of the hedging condition impact must begin with the approach used in AEM for all weapon allocations. Basically, the idea is as follows:

Step 1: Generate, on some arbitrary basis, a candidate allocation of all weapons to individual targets in the target set. This candidate allocation takes the form of a set of "strategies", where each strategy is a proposed allocation of some number of weapons to some specific target. For example, a typical strategy might say to allocate 17 weapons of type 3 to a single target of class 8.

Step 2: Given the candidate set of strategies, insert all of them into an LP which is to choose from among all the strategies so as to maximize total payoff, while not exceeding constraints on the number of weapons and targets.

Step 3: At the completion of Step 2, evaluate the strategies chosen and determine if any new candidate strategies should be searched for, or if the given allocation is optimal.

Step 4: If the optimal allocation has not yet been determined, go into a search mode to find new candidates and insert them into an LP, then return to Step 2 with the new candidate set.

The crucial step in this whole procedure revolves around Step 4. It is in this step that presence of hedging constraints impact most strongly.

To help visualize a typical LP tableau consider Figure P-1. This is a tableau as might occur for a 2-weapon type, 6-target class problem with 3 hedges. (Note that the payoff row has negative entries. This occurs because a minimizing algorithm is being used to perform a maximization objective.)

PAYOFF ROW	-18.1	-17.0	-29.0	-31.7	-42.6	-40.7	-89.1	-67.2	-118.3	-95.6	-66.1	-57.4		
WEAPON CONSTRAINTS	3	0	7	0	8	0	0	9	5	0	0	3	≤	50
	0	5	0	8	1	3	7	2	3	9	7	0	≤	82
TARGET CONSTRAINTS	1	1											≤	[illegible]
			1	1									≤	[illegible]
					1	1							≤	14
							1	1					≤	22
									1	1			≤	89
											1	1	≤	14
HEDGING CONSTRAINTS	3		7		8								≤	25
					42.6	40.7	89.1	67.2	118.3	95.6			≤	[illegible]
		1		1		1	1	1	1	1	1		≤	[illegible]

FIGURE P-1: A Typical LP Structure

There are two candidate strategies for each of the 6 target classes. For example, the first candidate for target class 1 indicates that 3 weapons of type 1 would attain 18.1 units of payoff if it were chosen. The entry of a 1 in the first target class constraint indicates that choosing this first strategy will use up 1 of the 5 targets of class 1. The target constraint row for class 1 assures that no more than 5 targets of that type will be attacked.

There are 3 hedging conditions in the tableau. Hedge 1 is a condition that says: do not fire more than 25 of weapon type 1 on targets in classes 1 through 3. Hedge 2 is a condition that says: do not attain more than 250 units of damage in target classes 3 through 5. Hedge 3 is a condition that says: do not attack more than 35 targets with strategies that involve weapon 2.

The job of the LP algorithm, as called for in Step 2, is to choose that combination of strategies which lives within the constraints and, at the same time, maximizes total payoff. In mathematical notation this is equivalent to:

Choose X(i) such that:

$$\sum_{i} X(i)VD(i) \text{ is maximized} \qquad \text{(P-1)}$$

and

$$\sum_{i} X(i)E(i,j) \leq R(j) \text{ for } 1 \leq j \leq C \qquad \text{(P-2)}$$

where:

$X(i)$ = number of targets to be hit using strategy i.

$VD(i)$ = payoff attained each time strategy i is used one time.

$E(i,j)$ = entry in row j and column i.

$R(j)$ = constraint value for constraint j.

C = total number of constraints.

It is important to realize in this formulation that the X(i) can take on non-integer values. That is, for example, X(1) can be <u>any</u> value from 0 up to 5. The limit of 5 occurs simply because there are only 5 targets of class 1 and that is the target class involved in strategy 1.

An alternate matrix notation for equation (P-2) is of interest because it emphasizes the columnar aspects to the problem. That form is as follows:

$$\begin{Bmatrix} 3 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 0 \\ 0 \end{Bmatrix} X(1) + \begin{Bmatrix} 0 \\ 5 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{Bmatrix} X(2) + \ldots \begin{Bmatrix} 3 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{Bmatrix} X(12) \le \begin{Bmatrix} 50 \\ 82 \\ 5 \\ 8 \\ 14 \\ 22 \\ 89 \\ 14 \\ 25 \\ 250 \\ 35 \end{Bmatrix} \qquad (P-3)$$

Each of the I column matrices indicated in the brackets corresponds to a column in the LP tableau. Thus, addition of other candidate strategies simply amounts to augmentation of the tableau with another column, and another bracket in equation (P-3).

It is very important to understand the nature of the entries in each of these columns. Basically, each row in an LP should be visualized as a resource constraint. This is true even for constraints which normally one would not think of as "resources". A constraint is always a "resource" in the sense that only so many units are available for use and the LP must "dole out" the resources in as efficient a manner as possible. In this light the entries in a column are the amounts of resources of each constraint type used up if that strategy is chosen to a level of 1.

For example, column 1 says that 3 units of weapon type 1 resource will be used, 1 unit of target type 1 resource and 3 units of hedge condition 1 resource if that column is chosen.

This notion of column entries will be returned to, but for the moment consider another aspect to the LP algorithm. At the completion of LP convergence with a given candidate set of strategies, the values of X_i have all been set. In addition the LP produces some auxiliary output which is very important. This other output can be described in many ways, but for our purposes it can best be viewed as marginal utility measures for the constraints.

These marginal utilities, herein called lambdas, indicate the marginal change in payoff for a unit change in a given constraint

<u>if all other constraint values and tableau entries are held constant</u>. These lambdas provide a relative "scarcity of resource" measure. If the lambda for constraint 5 is equal to 10.7 and the lambda for constraint 3 is equal to 3.5, it indicates that the last unit of constraint 5 is worth about 3 times that of the last unit of constraint 3.

Using such lambdas, any column which is a candidate for being added to the LP can be "costed-out" to determine how big a demand it would place upon the limited resources. For example, column 1 of the tableau requires 3 units of constraint 1, 1 unit of constraint 3 and 3 units of constraint 9. Thus, if column 1 was brought into the LP solution, it would tentatively cost an amount equal to:

$$P(1) = 3 \cdot \lambda(1) + 1 \cdot \lambda(3) + 3 \cdot \lambda(9) \qquad \text{(P-4)}$$

where:

P(1) = expected price to pay if column 1 is chosen to level one.

λ = current marginal constraint values.

However, if column 1 is chosen, 18.1 units of payoff will be accrued. Thus, the <u>net</u> benefit of bringing in column 1 is:

$$NP = 18.1 - 3 \cdot \lambda(1) - 1 \cdot \lambda(3) - 3 \cdot \lambda(9) \qquad \text{(P-5)}$$

where:

18.1 = return for bringing in column 1.

NP = net payoff for bringing in column 1.

= (return) - (cost).

This concept of "pricing-out" a column is key to the whole AEM LP procedure and, especially, to the hedging constraints.

The mathematical form of equation (P-5), see Reference (4) and Chapter IV-D, can be stated as follows:

$$NP(i) = PX(i) - \sum_{j=1}^{C} \lambda(j)\, G(i,j) \qquad \text{(P-6)}$$

where:

PX(i) = expected payoff from column X(i).

λ(j) = constraint j marginal utility measure.

C = total number of constraints.

G(i,j) = level of resources of type j called upon in column i.

Determination of the best possible strategies follows from equation (P-6). The best strategy at any given iteration stage is that one which "prices-out" most positive, or in other words the best G(i,j) values for a column are those which result in the largest possible NP value.

Primarily, equation (P-6) is used in steps 3 and 4 of the basic AEM LP iteration process. At the completion of any given LP this equation can be used to find out if any new candidate strategies should be added to the LP and, at the same time, determine if the iteration process can stop.

The crucial step in using (P-6) is in finding the best values for the column entries (G (i,j)). Since hedging conditions result in entries in the column, the presence of hedges therefore make use of (P-6) more complex.

Before delving into the use of (P-6) in hedging allocations it would be of some use to consider what varieties of hedges might be of interest. This would indicate possible entry sources.

3. Types of Hedges

There are obviously countless types and varieties of hedges possible. However, at this stage of development it seems appropriate to consider three special types which offer considerable flexibility. These three types are classified as follows:

Value Hedges: A condition on the value destroyed on a specified set of targets by a specified set of weapons.

Weapon Hedges: A condition on the number of RV's of a specified category allocated to a specified set of targets.

Target Hedges: A condition on the number of targets of specified classes attacked by a specified category of weapons.

These types are of special interest because they are involved with the three crucial elements of allocation: how many weapons attack; which targets; and how much damage will they do?

All of these three types of hedges are demonstrated in the tableau of Figure (P-1). Hedge 1 is a weapon hedge; hedge 2 is a value hedge; and hedge 3 is a target hedge.

In addition there is a variant of the value hedge which is of interest. The value destroyed by a given strategy is a function of three things: the value of the target; the number of weapons allocated; and the damage function parameters for the given weapon on the given target. It is in the damage function that the impact of many crucial uncertainties assert themselves.

For example, if one is uncertain about any weapon effectiveness parameters, like yield and CEP, or about any target vulnerability parameters, like hardness and defense level, the uncertainty can be expressed as an uncertainty about such things as probability of single-shot kill.

The importance of damage function uncertainty leads to the desire to specify value hedges which allow for uncertainty in damage function parameters. Thus, a broader definition of a value hedge might be as follows:

<u>Value Hedges</u>: A condition on the value destroyed on a specified set of targets by a specified set of weapons when the probability of kill for the weapons is computed on the basis of some described weapon and target characteristics.

Such a hedge might be as follows: attain at least 90% damage on all value targets even if side 1 weapon reliabilities are .10 lower than expected. This uncertainty about reliability reflects itself in uncertainty about the damage such weapons will achieve on the targets.

Even though many other hedge varieties are possible, these three types seem to contain the essence of the hedging problem.

In order to demonstrate more fully the notion of hedging types, consider the following examples:

Example 1: (Weapon Hedge)

Have side 1 attack the 5th city class on side 2 by at least a total of 50 RV's of the SLBM type.

Example 2: (Value Hedge)

If side 2 has 500 random area defenders instead of the nominal side 1 estimate of 250 defenders, side 1 would still like to try to do at least 650 units of damage to the value targets on side 2.

Example 3: (Weapon Hedge)

Each target in the first OMT target class (which has ten targets) on side 1 must on the average be attacked by 2 ICBM, 3 SLBM, and 5 A/C RV's.

Example 4: (Target Hedge)

All targets in the last OMT class and the first two value target classes on side 2 must be attacked by some weapon.

Example 5: (Value Hedge)

Side 2 is unsure about his own weapon reliability (values of .7, .8, or .95 are possible, with .8 being nominal). Separate runs for RL = .7 and .95 were made before the current hedging run. In those cases, the damage attained against side 1 were: 805 units when RL = .7 and 902 when RL = .95. Side 2 would be satisfied with one allocation that maximized damage for RL = .8, while attaining at least 780 units damage if RL = .7 and 850 when RL = .95.

Example 6: (Target Hedge)

The OMT targets on side 1 are so important that side 2 wants to attack all of them by some weapons of type 3, some of type 2 and some A/C.

4. Evaluation of A Hedging Strategy

Given that some set of hedges occur in an LP, it is necessary to utilize equation (P-6) to determine if any new strategies exist which price out positive and therefore should be added to the LP in the next iteration step. Expressed in a partitioned form, this amounts to the following:

Find a strategy for which NP(i) is maximized, where:

$$NP(i) = V \cdot PK - \sum_{M=1}^{NWP} LW(M) \cdot N(M) - LT$$

$$- V \sum_{M=1}^{NV} LVH(M) \cdot PKV(M)$$

$$- \sum_{M=1}^{NW} LWH(M) \cdot NWH(M)$$

$$- \sum_{M=1}^{NT} LTH(M) \cdot NTH(M) \qquad (P-7)$$

where:

V = value of a target of the class being attacked.

PK = probability of kill on this target for the allocation being studied.

NWP = number of weapon types.

N(M) = number of weapons of type M in the allocation.

LW(M) = constraint lambda for weapon type M.

LT = lambda for the target constraint.

NV = number of value hedges.

LVH(M) = constraint lambda for value hedge number M.

PKV(M) = probability of kill as computed by the rules for value hedge M.

NW = number of weapon hedges.

LWH(M) = constraint lambda for weapon hedge M.

NWH(M) = entry in weapon hedge M constraint row.

NT = number of target hedges.

LTH(M) = constraint lambda for target hedge M.

NTH(M) = entry in target hedge M constraint row.

Equation (P-7) is quite a complex mess to understand in one piece so a partitioned explanation is in order. The simplest way to understand (P-7) is to assume that someone has picked a candidate strategy for a given target and (P-7) is to be used to see if that strategy is a good one. Having a candidate strategy means that N(M) is specified for $1 \leq M \leq NWP$. For example, one might have N(1) = 0, N(2) = 3, N(3) = 0 and N(4) = 8, for a 4-weapon case.

In order to make this equation understandable it will be convenient to conduct sample computations for a typical case for each term in equation (P-7). Data necessary for the sample computations will be as follows:

Target Value, V = 105. and it is a force target.

Number of Weapons, NWP = 2.

Number of Value Hedges, NV = 2.

Number of Weapon Hedges, NW = 3.

Number of Target Hedges, NT = 2.

Candidate Strategy, N(i), is N(1) = 5, N(2) = 3.

Lambdas from last LP are:

LW(1) = .89, LW(2) = 1.55, LT = 67.0

LVH(1) = .05, LVH(2) = .52

LWH(1) = .5, LWH(2) = .8, LWH(3) = .7

LTH(1) = 2.1, LTH(2) = 3.2.

The hedges are:

Value Hedge i: Value destroyed on all force targets must be at least 750 units even if 975 random ABM exist. (Nominal estimate is that 500 exist.)

Value Hedge 2: Value destroyed on all urban targets in the list must exceed 50% of the value in the list.

Weapon Hedge 1: The number of weapon type 1 attacking all force targets must not exceed 210.

Weapon Hedge 2: The number of weapon type 2 attacking all force targets must not exceed 100.

Weapon Hedge 3: The number of weapons of any type attacking force targets must not exceed 350.

Target Hedge 1: The number of targets hit by weapon type 2 must exceed 200.

Target Hedge 2: The number of OMT targets hit by all weapons must exceed 150.

The appropriate single-shot probability of survival numbers are as follows:

WEAPON	SINGLE-SHOT SURVIVAL IF	
	500 ABM	975 ABM
1	.55	.68
2	.72	.81

Further explanation and use of all these numbers will occur in the term-by-term discussion.

Computing V·PK

Given that the N(M) are known, one should be able to compute the probability of kill for the given target. For example, if the target is a point target with no defense, the following equation holds:

$$PK = 1 - \prod_{M=1}^{NWT} PSK(M)^{N(M)} \qquad (P\text{-}8)$$

where:

PSK(M) = probability of single shot survival against the target class being attacked, when attacked by weapon type M.

Given that PK is known and the value of the target, V, is known, the first term in equation (P-7) can be computed.

However, this is where the first real complexity arises. If the N(M) includes more than one non-zero value a mixed strategy exists. In addition, if the target is an area target, or a defended target, the PK value depends upon the degree of partnership employed by the various weapon types. For example, a defended city will have one kill level if several weapon types are time sequenced to arrive in a well ordered manner and a different kill level if they arrive with no order. Also, in an area target the aim points might, or might not have been chosen to allow for mixed strategies. All of these "ifs" amount to a real problem in computing PK for a general circumstance.

Within AEM a simplification was made in computing PK. First, it was recognized that realistic computation of PK implies a lot of knowledge about very detailed war planning performance. Second, it has been found that mixed strategies usually occur only when hedges force them. This usually means that multiple weapon types are allocated to a target only in case the primary weapon does not show up. Thus, the idea is to guarantee some level of damage and not to take credit for the combined damage which might occur if all weapons show up exactly on schedule.

This logic led to the idea that in <u>each</u> mixed strategy there would be one weapon type identified as the primary weapon. All other types would be secondaries in that strategy. Then, the PK value taken credit for would be the damage expected from the primary weapon alone. The secondary weapons would be taken credit for only in satisfying the hedges. This primary weapon is identified as that weapon which would get the maximum kill if he was the only weapon type arriving at the target.

If the previously presented numbers are used, we find that:

$$V = 105$$

$$PK(1) = 1 - .55^5 = .95$$

$$PK(2) = 1 - .72^3 = .363$$

Since the PK from weapon 1 is higher, it will be designated as primary weapon for the target. This would result in

$V \cdot PK = 105 \cdot (.95) \approx 100.$

Computing $\sum_{M=1}^{NWP} LW(M) \cdot N(M)$

The second term in equation (P-7) is quite straightforward to compute. N(M) is known and the weapon lambdas are assumed known from the current LP solution. Therefore, this term can be computed directly.

From the data we can form:

$$\sum_{M=1}^{4} LW(M) \cdot N(M) = .89\ (5) + 1.55\ (3) = 9.10$$

Computing LT

This term is simply the target lambda from the current LP and it therefore is a known value. In our example LT = 67.

Computing $\sum_{M=1}^{NV} LVH(M) \cdot PKV(M)$

The next term in the so-called "Lagrangian" equation (equation P-7) is the contribution of the column entries because of any value type hedges that might exist. Again, the lambda factor presents no problem since the lambdas are available from the last LP. However, the PKV(M) factor, which is the probability of kill as computed for the specified value hedge number M, can be somewhat of a problem.

Essentially, all of the issues discussed in the section on computing V·PK will arise here also. If the strategy is a mixed strategy, computation of PKV(M) is exactly the same as computation of

PK, except for the possibly different damage function parameters involved.

The approach in AEM is to compute PKV(M) for each of the weapon types in the mixed strategy and to take credit for the largest individual PKV(M). Thus, one of the weapon types in the mixed strategy will be designated as the weapon responsible for satisfaction of the hedge.

With this approach, it is necessary to compute separate PKW values for each eligible weapon type in the mixed strategy. (It should be noted that usually all weapon types in a mixed strategy will not be eligible for a given hedge.) The PKW values are the probabilities of kill for each weapon type, or:

$$PKW(i,M) = 1. - PSS(i,M)^{N(i)} \qquad \text{(P-9)}$$

where:

PKW(i,M) = cumulative probability of kill for weapon type i in hedge number M.

PSS(i,M) = probability of single-shot survival for one weapon of type i.

N(i) = number of weapons of type i in the strategy.

In this equation the PSS ingredient is of special interest. The general value hedge must allow for uncertainties in damage function parameters and those uncertainties show up in the PSS factor.

The situation is as follows. Weapon type i will have a nominal probability of single-shot survival for the target being attacked (this is PSK(i) in equation (P-8)). But, since there is uncertainty about some factors used in computing PSK, like weapon characteristics, alternate values will show up in the PSS.

In our typical situation we have the following case. Weapons are being allocated on the assumption that 500 random ABM defenders exist. However, there is some possibility that as few as 250 exist, or as many as 975. The analyst wants to protect himself against the high defense level so he specified a value hedge which indicated a goal if such a defense level existed. This higher defense level then resulted in the PSS values indicated in the single-shot survival table.

The appropriate values for PKW must then be computed for each hedge and each weapon type by use of equation (P-9) and numbers like those in the table. The correct value for PKV(M) will then be the maximum PKW(i,M) for that target, or:

$$PKV(M) = \max_{i} \left\{ PKW(i,M) \right\} \qquad \text{(P-10)}$$

Given that PKV(M) is available, the complete term

$$\sum_{M=1}^{NV} LVH(M) \cdot PKV(M)$$ can then be computed.

Using the probability of survival table in our example yields the following computation:

Value Hedge 1: This hedge applies to force targets and the target being studied is in the force class. We then compute:

$PKW(1,1) = 1 - .68^5 = .855$

$PKW(2,1) = 1 - .81^3 = .468.$

Since PKW(1,1) is max, weapon 1 will be designated as the primary hedge weapon, for this hedge at least.

This results in:

$$V \cdot LVH(1) \cdot PKV(1) = 105(.05)(.855) \approx 4.5$$

Value Hedge 2: This value hedge applies only to urban targets and, thus, will not call for an entry in this column.

$$V \cdot LVH(2) \cdot PKV(2) = 0.$$

Computing $\sum$ LWH(M)·NWH(M)

This computation is rather straightforward. The lambdas, LWH, are available from the last LP. NWH(M) is the number of weapons in the mixed strategy which are eligible for satisfaction of weapon hedge number M. Simple knowledge of N(i) and the statement of the weapon hedge allows direct computation of this component.

In our example, 3 weapon hedges exist so it will be necessary to go through 3 computations:

Weapon Hedge 1: Weapon 1 is involved in this strategy so the hedge does apply:

$$LWH(1) \cdot NWH(1) = .5\ (5) = 2.5$$

Weapon Hedge 2: This hedge does apply since weapon 2 is included in the mixed strategy:

$$LWH(2) \cdot NWH(2) = .8\ (3) = 2.4$$

Weapon Hedge 3: This is a force target so the hedge does apply and we get:

$$LWH(3) \cdot NWH(3) = .7\ (5 + 3) = 5.6$$

Note that NWH(3) is the sum of all weapons in the strategy. This occurs because the hedge said that all weapons attacking force targets must not exceed 350.

Computing $\sum$ LTH(M)·NTH(M)

Again, the computation of this term can be done very easily. The LTH are available from the last LP and NTH(M) will always be 1. or 0. If target hedge M does not involve the target under investigation, it will be a 0. If the hedge does include this target, NTH(M) will be a 1. In our example, target hedge 1 applies to all targets, while target hedge 2 applies only to OMT targets. Thus, NTH(1) will be a 1 and NTH(2) will be a 0. This results in:

LTH(1)·NTH(1) = 2.1 (1) = 2.1

LTH(2)·NTH(2) = 3.2 (0) = 0.

Computing NP

Now that all individual terms are available the total value for NP can be found. In the example we have been using the terms add as follows:

$$NP(i) = 100 - 9.1 - 67 - (4.5 + 0.) - (2.5 + 2.4 + 5.6) - (2.1 + 0.)$$

$$= 100 - 93.2$$

$$= 6.8$$

Since NP(i) has turned out positive, it can be concluded that the candidate strategy would be a good one to add to the LP tabl .

The more crucial question, however, is one of finding that strategy set, N(M), which would make NP as large as possible. Such a strategy would result in the most rapid improvement to the allocation process. The issue of finding such optimal strategies will be addressed next.

5. Generation of Optimal Strategies

The basic mathematical problem in generati n of optimal strategies is to find N(M) such that NP is maximized for a given target. (This is one case where it is far easier to express the objective than it is to carry out the problem.) A grasp of the difficulties in this objective can best be obtained by working up to the most general case from the simplest possible case.

The simplest situation exists when only one weapon type is being allocated and there are no hedges. In that circumstance equation (P-7) reduces to:

$$NP(i) = V \cdot PK - LW(1) \cdot N(1) - LT \qquad \text{(P-11)}$$

In addition we know that:

$$PK = 1 - PSK(1)^{N(1)} \qquad \text{(P-12)}$$

so that the objective is to: choose N(1) such that:

$$NP(1) = V(1 - PSK(1)^{N(1)}) - LW(1) \cdot N(1) - LT$$

is maximized.

Maximizing this function is thoroughly described in IV-A and the derivation will not be repeated here. The basic procedure, however, is to compute:

$$N^*(1) = \left| 1 + \frac{LOG(LW(1)) - LOG(V) - LOG\ (1 - PSK(1))}{LOG\ (PSK(1))} \right| \qquad \text{(P-13)}$$

where:

$|A|$ = the largest integer contained in A.

$N^*(1)$ = the integer value for N(1) which maximizes equation (P-11).

A more complex situation occurs when there are hedges in existence. In such a case equation (P-7) reduces to (where we still have only 1 weapon type):

$$NP(1) = V \cdot PK - LW(1) \cdot N(1) - LT - V \sum_{M=1}^{NV} LVH(M) \cdot PKV(M) - \sum_{M=1}^{NW} LWH(M) \cdot NWH(M) - \sum_{M=1}^{NT} LTH(M) \cdot NTH(N) \qquad (P\text{-}14)$$

In this reduction the only issue has been to indicate that a single weapon type exists. The next reduction is to express PK and PKV in terms of survival probabilities and N(1). Before doing so, it is important to note that only some of the hedges will involve this weapon type on the target class being analyzed. Thus, the summations in equation (P-14) only occur for those hedges that do apply. All of these issues lead to:

$$NP(1) = V(1 - PSK(1)^{N(1)}) - LW(1) \cdot N(1) - LF$$

$$- V \sum_{M=1}^{NV} DD(M) \cdot LVH(M) \cdot (1 - PSS(1,n)^{N(1)})$$

$$- \sum_{M=1}^{NW} DD(M) \cdot LWH(M) \; N(1)$$

$$- \sum_{M=1}^{NT} DD(M) \cdot LTH(M) \cdot 1 \qquad \text{(P-15)}$$

where:

DD(M) = dirac-delta type function which takes on value of 1 when the hedge applies and a value of 0 when it does not apply.

It is possible to group these terms in a slightly better form:

$$NP(i) = V \left[1 - PSK(i)^{N(1)} - \sum_{M=1}^{NV} DD(M) \cdot LVH(M) \cdot (1 - PSS(1,M)^{N(1)}) \right]$$

$$- \left[\left(LW(1) + \sum_{M=1}^{NW} DD(M) \cdot LWH(M) \right) \cdot N(1) \right]$$

$$- \left[LT + \sum_{M=1}^{NW} DD(M) \cdot LTH(M) \right] \qquad (P\text{-}16)$$

This can be restated in an even simpler form as:

$$NP(i) = K - V\ PSK(1)^{N(1)} - \sum_{M=1}^{NV} VE(M)\ PSS(i,M)^{N(1)}$$

$$- LWE \cdot N(1) \qquad (P\text{-}17)$$

where:

K = all constant factors in equation (P-16) that do not use N(1).

VE(M) = effective value for value hedge number M.

= $V \cdot DD(M) \cdot LVH(M)$

LWE = effective weapon value for all combined weapon hedges.

= $LW(1) + \sum_{M=1}^{NW} DD(M) \cdot LWH(M)$

In order to visualize this equation consider the example numbers used in demonstration of computations for (P-7). Use of those numbers here would result in:

$$K = 32.75$$

$$V = 105$$

$$PSK(1) = .55$$

$$VE(1) = 5.25$$

$$VE(2) = 0$$

$$SS(1,1) = .68$$

$$LWE = 1.39$$

which produces the following for NP(i):

$$NP(i) = 32.75 - 105\ (.55)^{N(1)} - 5.25\ (.68)^{N(1)} - 1.39\ N(1) \qquad \text{(P-18)}$$

For this example, it would be necessary to find that value for N(1) which maximizes (P-18). (Just for interest's sake, N*(1) = 7 is the correct value.)

It is not possible to produce a closed form equation which will yield the optimal N(1). The most consistent procedure found to date is to use a Newton-Raphson iteration procedure which uses a derivative method to converge u[illegible] the correct value. Such a procedure unfortunately can take considerable computation time when there are literally thousands of times the procedure will be used in a given LP problem.

In evaluation of several procedures for maximizing equations like (P-17) and (P-18) it was observed that most often one of the terms of the form VPS^N would dominate the computations. For example, in (P-18), it is the $105(.55)^{N(1)}$ part of the equation which in essence determines the optimal N(1). The $5.25(.68)^{N(1)}$ has little effect because the coefficient is so much smaller and the optimal N(1) must be integer.

This observation resulted in a fairly rapid procedure for approximately maximizing equation (P-17). This procedure was to find various candidate N(1) values by solving a sequence of maximizations. Each candidate N(1) was found by acting as if all parts of the equation of the form VPS^N were non-existent, except for one of the parts. That is, if the equation had 4 terms of the VPS^N form, an N(i) candidate would be found by setting all VPS^N parts to zero except for one of them. Then, a different part would be set to zero and another N(1) found. Finally, all of these candidate values for N(1) would be evaluated by use of the whole equation and the one which led to the overall maximum would be accepted as the optimal N(1).

In the above example only 2 parts exist so one could proceed as follows:

<u>Stage 1</u>. $\text{Max}\left\{32.75 - .05\,(.55)^{N(1)} - 1.39\,N(1)\right\}$

<u>Stage 2</u>: $\text{Max}\left\{32.75 - 5.25(.68)^{N(1)} - 1.39\,N(1)\right\}$

<u>Stage [illegible]</u> Evaluate equation (P-18) with the N(1) found in stage 1 and the N(1) found in stage 2. Choose the one that produces the largest value for [illegible].

At the present time comparisons are being made to determine if, in real data situations, this procedure is adequate. The issue of computing time to accomplish the Newton-Raphson method is also being investigated. However, at the moment it does appear that this approximation is adequate.

The next level of complication in the generation of optimal strategies occurs when more than one weapon type exists. In that circumstance all issues come together in equation (P-7). The issues are so complex that they are rather difficult to describe, but describing them is necessary before the problems can be understood.

Consider a typical general case with 12 weapon types, 5 value hedges, [illegible] weapon hedges and 2 target hedges. In such a circumstance some weapon types might apply to some hedges and not to others. The net effect can be displayed as in the following table:

WEAPON	CAN WEAPON TYPE I CONTRIBUTE TO HEDGE J?									
	HEDGE 1	2	3	4	5	6	7	8	9	10
1	Yes									
2	Yes	Yes								
3								Yes		
4										Yes
5		Yes					Yes			
6							Yes			
7	Yes								Yes	
8	Yes						Yes			Yes
9	Yes			Yes						
10		Yes		Yes		Yes				Yes
11		Yes				Yes				Yes
12						Yes		Yes		

Within this table all weapons which can qualify for contribution to a given hedge are labeled yes. All blanks imply that no, the weapon cannot qualify. In addition, all weapons are assumed to qualify for the basic objective of maximizing target damage. Note that columns 3 and 5 have no yes entries. This will occur when those specific hedges do not apply to a given target class. Thus, the table will change if the same question is asked for a different target class. If such a table was constructed for the example we have been working with, it would appear as follows:

WEAPON	CAN WEAPON TYPE I CONTRIBUTE TO HEDGE J?						
	1	2	3	4	5	6	7
1	Yes		Yes		Yes		
2	Yes			Yes	Yes	Yes	

It should be clear from these tables that finding out which weapon types should come into a strategy and to what level is a very complex issue. Take the above table. Weapon 1 can qualify for hedges 1, 3 and 5. He is all alone in satisfying hedge 3 but will be competing for hedges 1 and 5 with weapon 2. Meanwhile, weapon 2 is also working on hedges 4 and 6.

It is clearly possible that at one level for weapon 1, hedge 3 would have a good contribution but weapon 1 would lose the competition for entering into hedges 1 and 5. Meanwhile, at a higher level, weapon 1 would win all competitions for all hedges he qualified for.

This concept of being in competition for hedge entries has formed the basis for the current approach to maximizing equation (P-7). In essence, each weapon type is presumably trying to find an attack level such that he gets assigned to as many entry locations in the column as possible. Meanwhile, the overall choice for which weapon types do enter into the column rows is made on the basis of which combination produces the largest value for NP(i).

An equivalent way of expressing this concept is in terms of assignments. Each hedge row and the top row entry of value destroyed is to have a weapon type assigned to it in a given column. This primary assignment is to be made for the column in such a way that NP(i) is maximized.

The manner in which the assignments are carried out at present in AEM is as follows. A series of competitions are set up with the series being run in a special order. First, a competition is held to determine which weapon type will be tentatively assigned the job of being the primary weapon in getting overall damage to the target. Second, a competition is held to determine which weapon type will be assigned the job of being the primary weapon in satisfaction of hedge number 1. Then, competitions will be held for hedges 2, 3 and so on until all hedges have weapons assigned to their fulfillment.

Some weapons might be designated as primary on more than one hedge. In fact, it often happens this way.

The assignment of being primary weapon to a given hedge is made in the following way. Assume that some number of assignments have been made. Determine, by an analysis like that made in producing the previous table, which weapon types are eligible for the next assignment. Then, go through the following sequence:

Step 1: Find out the optimal attack level for a given weapon type as if he were going to be chosen as the assigned weapon for all as yet undertermined assignments for which he qualifies. This is found by construction of an equation like (P-15) and then consequent maximization.

For example, in our last table, say that assignments have been made for primary weapons on the target and satisfaction of hedges 1 and 2. Now, it is time to assign hedge 3 to some weapon type. Weapon 1 qualifies so it is possible to form an equation like (P-15) where it is assumed that weapon 1 is also going to be assigned to hedge 5. Maximization of that equation will produce a candidate N(1).

Repeat this procedure for all weapon types eligible for the given hedge assignment.

Step 2. Assign the given hedge to that weapon which yields a maximum value for equation (P-7), where eligible weapon types are in at the attack levels found in Step 1.

Step 3: Tentatively choose as the attack level for the winning weapon type the value found in step 1.

Step 4: Return to step 1 and carry out a competition for the next unassigned hedge.

The result of these steps is an assignment of given weapon types at given attack levels to the strategy. Thus, all N(M) are specified and a procedure like that in section 4 can be followed in order to compute all column entries and evaluate the final Lagrangian.

Why does this procedure work? The answer lies in the basic nature of the Lagrangian process. First, mixed strategies are usually not optimum unless some hedge forces their entry. Because all hedges are really secondary objectives to the primary objective of maximizing total damage, the best weapon types for a strategy are usually those that simultaneously do a good job of damage attainment and hedge satisfaction.

This fact makes it most important to find single weapon types that do a good job in several row entries in a column. Thus, most often, the Lagrangian will be maximized by choosing weapons that are in at levels which were chosen on the assumption that they would win all competitions for which they qualified. Large levels of mixed strategies only occur when there are numbers of hedges which can be satisfied only by a few weapon types.

This sequential procedure "stacks the deck", so to speak, in favor of weapon types that qualify for many hedges, but this is exactly what is most often the right thing to do. However, the procedure does not short-change the hedges which can be met only by one weapon type since a competition will be held for that hedge assignment and the only eligible weapon type will win the competition.

6. Computational Considerations

The above procedure has been tested on quite a number of hedging test cases, is felt that in most circumstances it does do an acceptable job in arriving at near-optimal allocations. There are, however, some alternate schemes which are being evaluated for their applicability. As was mentioned, the single weapon type maximization (P-15) has been solved by a Newton-Raphson procedure, and the only reason why it is not being used is that computation time seems excessive.

This issue of computation time is worthy of some discussion because it indicates how much effort must be devoted to approximation procedures.

Consider a 15-weapon, 30-target class, 7-hedge case. Optimal allocations for such a case can usually be obtained in 7 to 10 LP iterations. At each LP iteration each weapon-target combination must be considered as a candidate for inclusion as a new LP strategy. This amounts to about 450 solutions of equation (P-13) for every LP if no hedges exist.

However, in this case it might be necessary to consider each weapon type as a candidate for each hedge even if the weapon type was not designated as the primary weapon for target damage. But, the procedure of Step 1, as just described, said that equation (P-15) would be solved for each weapon type eligible for a given hedge. In turn, the approximation for a solution to (P-15) involved solving an equation like (P-13) as many times as there were value hedges still unassigned to any weapon. This all amounts to the possibility of solving equation (P-13) many more times because of the hedges.

As a general formula, the number of solutions to (P-13) in a hedging LP iteration can be estimated by the following relationship:

$$PAS \leq NWP \cdot NTG \cdot (\frac{NH^2}{2} + NH + 1) \qquad (P\text{-}19)$$

where:

PAS = number of required solutions to (P-13).

NWP = number of weapon types.

NTG = number of target types.

NH = number of hedges.

In practice PAS rarely approaches this magnitude because most hedges only apply to certain weapon and target types. In addition, the NH^2 factor in (P-19) occurs because of value hedges of certain varieties, and those varieties do not occur in every case.

Nevertheless, the total strategy generation time is proportional to PAS and PAS can very easily go up linearly with NH. Thus, even if the approximations currently in use in AEM are followed, the running time could increase by a factor from 1 to 10. Thus, it is imperative that considerable effort be expended on finding rapid, but accurate approximations to use in solving (P-7).

In the forementioned testing of these procedures several good usage rules have been uncovered. For example, because of the way that hedge assignments occur in a descending order, it is possible to affect the quality of the approximation by simple change of order in the list of hedges. Such a change of order is especially worth trying when in some case a given hedge was not met. Changing the order of hedge specification might lead the allocation into a different and better approximation route.

In general, the procedure is slightly biased to the idea of not putting extensive mixed strategies into the LP. As a result, the most common non-optimal allocation situations are those when a mixture was not specified and it should have been. Inequality hedges, which will be discussed in more detail, of the $\leq$ form tend to discourage mixtures, while other inequalities might encourage them. It has been found that the procedure works best when the hedge specification order has $\leq$ hedges first in the list and $\geq$ hedges last.

The presence of infeasible hedges warps the allocation since all possible effort is made to satisfy such a hedge. At the same time, however, weapons not involved in the hedge satisfaction problem will be allocated optimally. Thus, allocations when infeasible hedges exist will still convey considerable information.

Another issue in hedging allocations is that weapon lambdas are warped to some degree. Therefore, their use as a part of the force value scale problem (Chapter IV-C) is not valid. Accordingly, hedges are only allowed in the first strike of any scenario.

Hedges for the side going in any other strike can be evaluated only by use of a mis-estimate option within AEM (see Reference 11). Other computational considerations and guidelines are also presented in that same reference.

7. Relative Hedge Preferences

Implementation of these hedging concepts required modification of the linear program (LP) so that it would properly deal with all forms of inequality constraints. Prior to this hedging development the LP could only accept $\leq$ constraints. With hedges, however, it was cricial to allow $=$ and $\geq$ types of constraints.

The method chosen for implementation of these inequality constraints is known as the big M method. In this method the following equation transformation first occurs (using an $\geq$ constraint as an example). Start first with a general equation of the form

$$\sum_{i=1}^{n} C(i)\ X\ (i) \geq B \qquad \text{(P-20)}$$

This is equivalent to

$$\sum_{i=1}^{n} C(i)\ X\ (i) - Y = B \qquad \text{(P-21)}$$

Where Y is called a slack variable and it define the level by which $\sum C(i)X(i)$ exceeds B.

If one simply inserted an equation of the form of eq (P-21) into an LP the [illegible] problem is one of devising an initial feasible solution. That is, one would have to determine values for the X(i) so that some value for Y which is ≥ 0 would result. This problem results because negative values for the LP variables are not allowed.

Given such an initial set of X(i) the LP could then proceed to an optimal set of X(i). In this process only feasible values for X(i) would be allowed.

In many applications of linear programming finding the required feasible X(i) to start with can be a serious problem. This can be overcome by adding a second slack variable to equation (P-21). Doing so results in

$$\sum_{i=1}^{n} C(i)X(i) - Y + Z = B \tag{P-22}$$

In this new form any starting X(i) is allowed because Z can be positive, and thus satisfy equation (P-22) for a zero value of Y, or Y can be positive when Z is zero.

However, the desired condition requires that Z equal zero since we started with an $\geq$ constraint and any positive value for Z would violate that requirement. This is possible by using the big M concept as follows.

Visualize an LP tableau where the payoff (or objective) row contains a <u>penalty</u> of amount M for every unit of the variable Z in the final solution. That is, if the optimal X(i) results in Y=0 and Z=Z* then the objective is reduced by an amount of Z* times M.

If M is an arbitrary, but very large number, this penalty can be so great that the LP will make every possible effort to find an optimal X(i) such that Z=0. The net result is satisfaction of the original requirement for $\geq$ B.

This approach was selected for use in AEM for one major reason. It is not always true that a feasible X(i) exists. The analyst could easily specify a condition that the existing arsenals could simply never meet. In such a circumstance the LP should still attempt to minimize Z and thus communicate to the analyst exactly how infeasible the requirement was. By use of a big M infeasibility is discouraged but not precluded.

The negative fact that use of a very large value for M can cause numerical convergence problems in the LP was felt to be overcome by this ability to allow infeasibilities to exist if they were necessary.

The use of this big M technique also had the fortuitous capability to allow specification of preferences among a set of hedges as required by this task. The manner in which this capability was exploited is described as follows.

a. <u>A Need for Preferences</u>

The notion of a preference among hedges is only important in the circumstance where it is not feasible for a given [illegible] goal to satisfy every one of a set of hedges. If all hedges could simultaneously be satisfied it would not be necessary to ask for a preference list.

When it is not feasible to satisfy every hedge in a set, however, the analyst would normally prefer that certain hedges be satisfied in preference to others that might be easier to satisfy. Such a preference is conveniently allowed for by variable big M values.

Consider a set of hedging constraints like that of equation (P-22). Further assume that each equation has its' own value for M. In such a case the normal - maximize total value destroyed - type of objective gets converted into

$$\text{maximize TVD} - \sum_{j=1}^{H} M(j) \bullet Z(j) \qquad \text{(P-23)}$$

where:

TVD = total value destroyed

M(j) = big M for hedge j

H = total number of $\geq$ hedges

Z(j) = amount by which hedge j is not satisfied

Now, if all Z(j) can not be simultaneously driven to zero the LP will automatically drive for a specification which still maximizes equation (P-23). If all the M(j) are equal, but still very large, the LP would therefore attempt to minimize $\sum Z(j)$.

This would naturally result in satisfaction of those hedges which are least binding before satisfaction of the more constraining hedges. This course of events is what an analyst might desire to preclude by some sort of preference specification.

Consideration of equation (P-23) leads to the obvious thought that selection of unequal values for M(j) could lead to a preference matching allocation. For example, if M(1) is 10 times M(2) the LP would be 10 times as interested in reducing Z(1) as it would be in reducing Z(2). Such a technique for analyst control over the M(j) was devised as follows.

b. Expression of Preferences

At first glance it might appear that an option which would allow analyst input of M(j) directly would be adequate. This has been done for AEM HEDGE; however, there can be difficulties for the analyst in selection of such values.

First, the analyst will not usually know exactly how difficult each hedge is to satisfy. Second, because of the numerical accuracy problems with very large values for M(j) it is not feasible to simply input numbers such as

$M(1) = 10^5$

$M(2) = 10^{10}$

$M(3) = 10^{15}$

Etc.

The computer very rapidly would develop problems in round-off. In fact, anyc [illegible] at has an M(j) larger than 10^6 has been found to be error-prone. As an alternate treatment for M(j) sizing consider the following interpretation for each big M.

When an analyst inserts a hedge he usually can express some desire for satisfaction of that hedge in terms of comparison with the total value of all targets in the target structure. For example, he might say "I would like to satisfy this hedge about as much as I would like 20% more damage to the total target structure." A second less desirable hedge might be as acceptable as 10% more damage to the total target structure.

When one thinks in terms of this kind of preference it is more feasible to expect an analyst to choose a value that he can relate to. He normally understands the basis for the value scales of his targets and he can scale his hedge preference to those scales.

In these terms each hedge will be satisfied as long as doing so will not detract from the TVD by more than M(j) units per each extra unit of hedge j requirement met. This concept can be expanded even further as follows.

When the analyst specifies a hedge requirement he must designate a value for B(j) which is the goal value for the hedge. Now, his preference could be in either of two forms:

1. I would like to totally satisfy hedge j about as much as 20% more damage to the total target structure, or

2. For each unit I come closer to satisfaction of hedge j, that is for each reduction in Z(j) by one, I am willing to accept 20% less damage to the total target structure.

The difference between these two is the following. In type 1. the analyst will give up 20% more damage in order to drive Z(j) all the way to zero. In type 2. the analyst would give up 20% more damage in order to reduce Z(j) by a single unit.

All of these options have been programmed into AEM HEDGE. They allow the analyst to specify a preference for a given hedge in any one of the following ways:

Option 1: A value for M(j) is input directly and no program modification of that value occurs.

Option 2: A value, A(j), is input by the analyst and the program computes M(j) as equal to A(j)/B(j).

Option 3: A value, A(j), is input by the analyst and the program computes M(j) as equal to A(j)* TV, where TV is the total value of all targets in the target list.

Option 4: The same as Option 3 except M(j) is equal to A(j)* TV/B(j).

By appropriate use of these options the analyst should be capable of expressing his preferences in an acceptable manner, without requiring abstract selection of M(j).

An allocation will then be guided by the M(j) values and the final optimal allocation will be one where a hedge will be satisfied up to the point where more accomplishment in that hedge results in unacceptable reductions in the TVD.

8. Automatic Hedge Set [illegible]

The capability of [illegible] special sub-goals, or allocation constraints, [illegible] need for special treatment of infeasible hedges. Since the analyst can create very binding hedges if he so desires, it can happen that a given arsenal is incapable of meeting the hedges and the program must then decide how to deal with the situation.

The previous section discussed one option in dealing with such a circumstance. That option is simply to offer the analyst the capability to express relative preferences among the hedges. This preference is then used by the program to satisfy the hedges in such a manner as to minimize the undesirability of not meeting the complete set of hedges.

This section is directed at a second option. Namely, to allow selected growth in the arsenal as necessary to fulfill all of the hedges. When this is accomplished in such a manner as to minimize the required growth in the arsenal the analyst then is shown how much additional arsenal would be required in order to meet all of his hedges.

The analyst obviously could make several runs at different force levels with the hedges and then interpolate to determine the minimum force augmentation necessary to achieve satisfaction of all the hedges. Within this task, however, the goal is to achieve that determination within a single run if at all possible.

a. The Methodology

Several possibilities exist for development of an appropriate methodology for this objective. First, it is possible to make the program conduct a single parameter search and interpolation process just like an analyst could accomplish by making a sequence of individual runs. In this type of approach, however, one would have to expect an increase in running time that would be significant. The approach is essentially like making a number of separate runs and it is most likely that at least three such equivalent runs would be necessary in order to accomplish acceptable convergence. The only savings in this approach would be in terms of reduced analyst interface with the computer.

A second, more desirable approach has been pursued. This approach makes maximum utilization of the flexibility offered by the linear programming (LP) capability within AEM. Within the approach automatic convergence to a minimum required force augmentation is possible.

It is advisable to discuss adaptation of the weapon constraint row to the purpose of this task and the interaction of that row with the rest of the constraints.

In each LP there are several types of constraints. First, there are $\leq$ types of constraints on the number of weapons of each weapon type and on the number of targets of each target class. Second, there are the hedging constraints, which can be of any inequality form, and they can require satisfaction of conditions which the specified set of weapon types are incapable of achieving.

If one of the weapon constraint rows was converted from an $\leq$ form into an $\geq$ form the effect could be useful to the purposes of this task. A weapon constraint of an $\geq$ form, in essence, totally unbinds that weapon row. This allows unlimited growth in the arsenal, as that weapon type could be used in any amount as necessary to accomplish damage on the targets, or satisfaction of the hedges.

The problem with this inequality reversal is that it allows unlimited growth in the number of weapons of that type. As a result, enough weapons would be added to the arsenal to achieve 100% total target damage. In other words the arsenal would grow beyond the minimum level necessary to satisfy the set of hedges.

After some experimentation with this unbounded arsenal problem it was possible to devise a precise technique for controlling the arsenal growth so that only the minimum necessary augmentation would occur. This was accomplished by requiring the analyst to insert one additional control hedge which could be used in the LP to cut off the arsenal growth at the preferred point. This control hedge operates as follows.

The analyst estimates how much total damage could be accomplished by the unaugmented arsenal. This estimate most likely would be obtained from the run which first showed the analyst that he had a set of infeasible hedges. It could also be a damage goal which is adequate for his total damage objective. For example, 25% national fatalities or some similar accepted goal.

This control hedge is treated in a special manner within the LP. The form of the hedge is like any standard value destroyed hedge, namely:

HEDGE(L*,J) = V (ALL) BY (ALL) GE AMO

where:

L* = the special control hedge number

J = the side with the augmentation problem.

AMO = the estimated total value destroyed by the unaugmented arsenal.

However, the program has been told to treat this control hedge in a special manner when it comes to taking credit for total value destroyed beyond the level of AMO.

Under normal circumstances the above hedge would specify that AMO is to be the minimum total damage acceptable, but that any excess above that is allowed and even desirable. In this control hedge the LP is modified so that any excess above AMO is acceptable but it is not desirable at all. This allows the total damage to grow, as would occur when the arsenal is augmented, but as soon as theall hedges are met further growth will be stopped. In order to understand how this is accomplished, consider the following.

A normal constraint of $\geq$ form is usually inserted into the LP in the form

$$\sum_{i=1}^{n} C(i)\, X\,(i) - Y + Z = B \qquad \text{(P-24)}$$

where:

$C(i)$ = constants

n = total number of non-slack variables in the LP

Y, Z = artificial slack variables

B = constraint amount

This form of insertion would occur also for the special control hedge.

In addition, the payoff, or objective row is normally constructed as follows:

$$\text{maximize TVD} - \sum_{j=1}^{H} M\,(j)\; Z\,(j) \qquad \text{(P-25)}$$

where:

TVD = total value destroyed on all targets by all weapons

H = total number of $\geq$ constraints

$M(j)$ = big M for row j (see Section 7)

$Z(j)$ = amount by which row j $\geq$ constraint is not satisfied.

The $M(j)$ factor are used to drive the $Z(j)$ terms to zero and thus accomplish the satisfaction of all the $\geq$ constraints. In this problem, however, the arsenal is not adequate enough to drive all the $Z(j)$ to zero.

When the special control hedge for this problem exists, however, the above objective row is modified to be of the form

$$\text{maximize TVD} - Y - \sum_{j=1}^{II+1} M(j)\, Z(j) \qquad \text{(P-26)}$$

In this new form the variable Y appears. From equation (1) it can be seen that when Z=0 and Y $\geq$ 0 it is the excess by which that control constraint exceeds the constraint amount, AMO.

Observe that as the force is augmented in order to drive all of the Z(j) to zero the Y term in the objective function will exactly nullify any desire to increase the force augmentation just to increase TVD. This occurs because the control hedge specifies that total value destroyed by all weapons on all targets was to be $\geq$ than AMO. Therefore, the Y variable measures how much the TVD has grown beyond AMO and its presence in the objective row cuts off any incentive for such growth. Note that it does not preclude such growth as long as some of the Z(j) terms are being driven to zero.

Experimentation with this technique demonstrated excellent performance. Computer running time with the addition of the control hedge was not noticeably affected and determination of a minimum arsenal augmentation did occur as theorized.

The analyst need only do three things. First, he must estimate the unaugmented arsenal damage, AMO. Second, he must construct the additional control hedge in the form previously described.

Finally, he must [illegible] which hedge is the control hedge by insertion of [illegible] L* [illegible] variable.

LIMHEG (J) [illegible] L*

where:

L* - the hedge number which is designated to be the control hedge.

This variable identifies the proper Y variable to insert into the objective row. None of the other hedges would have any modification to the objective function.

Before leaving this discussion of the methodology it is necessary to point out one caution in usage of this form of force augmentation computation.

In certain defense types, namely random area defenses, a probability of defense penetration is computed prior to LP time by use of the ratio of defenders to total attackers. Since this computation occurs <u>before</u> determination of the final force augmentation level the LP will be using probability of kill values which do not take credit for the impact of the extra arsenal on the probability of penetration.

The net effect is that the LP will choose a conservative sized force augmentation which is an upper bound to the true minimum required. The degree of conservatism can only be determined by making a second run without the control hedge but with the previously determined force augmentation.

9. Non-Alert Weapon Allocations

One of the classical problems in weapon allocations is that of devising war plans which are not overly sensitive to precise pre-war knowledge about the conditions and performance of weapons when the war actually begins. It is obvious that one must know quite a lot about his own capabilities and the capabilities of his opponent just to devise a reasonable allocation. What is desired, however, is to avoid the requirement for allocations finely tuned to precise situations.

AEM HEDGE has numerous features designed to provide aid in development of such allocations. These features allow measurement of the impact of making misestimates and development of hedging allocations which guarantee achievement of specific goals even when certain defined misestimates do occur.

Therefore, it is natural to desire to develop a capability for dealing with weapons which might or might not be available at war start time. None of the current AEM HEDGE options deal with such misestimates in a truly optimal procedure.

In order to help understand the issues in such a task, consider the following situation. Blue has 1000 weapons of a certain variety. His nominal estimate for their alert rate is 85%. Thus, on the average, 850 of the 1000 are available for launch at any given time. However, which 850 it is keeps changing and on some occasions he has more, or less than 850 available.

Blue would like to be able to make maximal use of whatever weapons are available at the instant of demand [illegible]. He could do this quite easily if he simply waited until the demand time occurred and then generated an allocation which would be optimal for the specific weapons available at that time. This would result in the maximal usage of the alert weapons.

A major problem with such a scheme is that revising and generating war plans is not such an instantaneous event. Many aspects must be considered and time is always short. The command and control requirements would be strenuous and in many ways such a concept would be considered infeasible.

A second option one might consider is to use the alert rate factor as a modifier of reliability during the allocation process. For example, if the Blue weapons had a reliability of .90 Blue could allocate all 1000 weapons at an effective reliability of .85 X .90. Then, when the war began Blue would not have to adjust any allocations.

This concept does not maximize th[illegible] from the alert weapons since allocations with 850 weapon[illegible] reliability of .9 will achieve more total damage than an allocation of 1000 weapons with reliability of .765. It also has the feature of being sensitive to the estimated alert rate. If the alert rate really is .9 the capabilities of the arsenal will not be taken advantage of to the highest level.

It desired to devise a procedure for generating allocations that adapt by themselves to the day-to-day alert conditions and require a minimum of re-allocation in order to gain maximum utility from the weapons. Such an allocation would hopefully b of minimal sensitivity to the nominal alert estimate. Some sensitivity will obviously exist, however.

a. 1 e Methodology

The concept developed in this section follows from certain basic premises concerning characteristics of a desirable allocation of the non-alert weapons. These premises include the following:

1. It is desirable to have an allocation that includes the total set of weapons, both alert and non-alert.

2. The allocation should not require a distinction as to which specific weapons are on alert, or on non-alert. At most, the fraction on alert can be a basic assumption.

3. The allocation should, if possible, include as a sub-set an allocation which would be reasonable if only the alert weapons were allocated.

Other premises could be added; however, the above set should allow acceptance of an allocation as one which would yield a reasonable, even if non-optimal, allocation of the alert plus the non-alert weapons.

The premises can be further expanded upon as follows. First, if one assumes that re-allocations at war time are impractical it is vital that all weapons should be included in the plan. Then, whichever ones are on alert can go towards accomplishment of the objectives in the plan. Second, on the long term, weapons go in and out of alert status. Thus, in order to minimize re-allocations over time it would be desirable to not require a daily re-assignment weapon by weapon. Third, since some re-allocation probably is possible as long as time urgency does not exist, it would seem intuitively reasonable that the allocation obtained by allocation of precisely the alert weapons should be attainable as a sub-set of the total allocation.

Within the above premises the following method of operation would seem reasonable. First, all weapons are allocated to some target or another. Second, if an important weapon became unavailable his target position could simply be switched with an alert weapon which at that time has as his primary target one which is lower on the list. This re-assignment could occur as often as time allowed, but it would always involve one-to-one switching of positions, with no major reshuffling of the complete plan.

A technique for development of a plan with the above characteristics has been accomplished in this task by expansion of the hedging option in AEM HEDGE. This expansion requires the following procedure.

Allocate the total (alert + non-alert) weapons as if they would all be on alert. However, allocate them in such a way that if only a share of the weapons actually were on alert the damage attained by that share would be nearly as good as would be achieved if the precise alert weapons had been known. The net effect of this "guarantee acceptable damage by the alert share" concept is modeled as a proportional reduction of the attackers to each and every target.

For example, take the case where n weapons are being allocated to a given target under the assumption of some PSSK that would exist if all weapons were on alert. However, if the alert share was only 75% the PSSK might be somewhat different. This other PSSK, call it PSSKP, would be lower; for example, if there were random defenses whose effectiveness depended upon the total number of attackers in the whole arsenal. In such a case the damage might be

$$VD(i) = V(i)\ (1-\ (1-PSSK)^{n}) \qquad (P-27)$$

where:

V(i) = value of target i

n = number of attacking weapons on target i

PSSK = probability of single shot kill

The damage for the proportional share of alert weapons on that target might be

$$VDH(i) = V(i)\ (1-\ (1-PSSKP)^{AV \cdot n}) \qquad (P-28)$$

where:

AV = weapon alert rate

PSSKP = probability of single shot kill if only the expected alert weapons exist.

In the special allocations generated [illegible] option the objective is to maximize the total [illegible] according to the VD(i) relationship, but have the total to be destroyed according to the VDH(i) relationship be some acceptable goal. This two-pronged objective results in an allocation of all weapons in such a way that if only the estimated alert fraction occurs the total damage will be acceptable.

In this use of what is to be called an availability hedge the analyst must specify what is "acceptable" total damage when only some specified fraction is on alert. This goal can be obtained by the analyst making a run in order to determine what maximal damage could be obtained if the true alert weapons were known at war plan generation time.

One way of looking at the allocations generated by this hedging plan is as follows. Consider it to be a plan which could be obtained by taking a separate plan which involves only the alert weapons and proportionally assigning the non-alert weapons to that plan. That is, if 10 alert weapons are to go to a target and the alert rate is 50%, then make a new plan which has 10/.5 = 20 weapons to that target.

However, the basic alert weapon plan and the final total weapon plan are related to each other in such a way that the final plan maximizes damage if all weapons were on alert while providing for acceptable damage if only the expected alert rate occurs.

A plan which does well at the two conditions of 100% alert and the expected alert hopefully would be a reasonable plan for any alert rate in between. No testing procedure for this hope was devised during the task effort, however the hypothesis seems reasonable.

One perspective on the capability offered by this option is that it allows the analyst another way to develop a plan that hedges against a type of uncertainty. In certain cases this option might be of some use while in others a different set might be more appropriate. The whole intent is to offer a spectrum of capabilities which an analyst could use as an exploratory tool.

10. Some Applications

There are certainly countless numbers of ways that this hedging condition option can be applied. The programming of the option has been done in a flexible manner so that the analyst should be able to devise sets of hedges which at least come close to any specific application. This section will discuss three examples of applications that have been tried and found to be of some use. The discussion of the examples will illustrate how cases not normally thought possible in previous AEM versions are now open to the analyst.

a. Limit the Number of Targets Hit By One Bomber

A [illegible] problem in bomber allocations in AEM is created because there has been no control over the number of targets any individual bomber could hit. For example, a single bomber carrying five bombs could hit up to five separate targets. Considering that defenses between targets, etc. are in existence and that the bomber would have to serially penetrate a number of targets it is certainly unreasonable not to penalize the bombers in some way.

With hedges the analyst cannot penalize the bombers but he can make his own decision about the number of targets per bomber he thinks is reasonable. Then, he would simply state a target hedge that said: Targets of all types hit by bombers must be LE N•B, where B is the number of bombers and N is the desired limit on targets per bomber. Such a hedge would offer some reasonable control.

b. Force Damage Requirements

In previous versions of AEM an analyst could specify an assured destruction objective, where assured destruction is defined as an industrial target damage goal, or upper limit. This was achieved in the weapon allocation process by insertion of a damage constraint into the weapon allocation linear program (see IV-D). Since that approach has been reliable and successful, it was felt that a similar option which applies to force damage goals would be a natural application of hedges.

In any strike where force targets are open to attack a value hedge can be specified so that a force damage requirement is met, if possible. Essentially, it is only necessary for the analyst to state:

1) The percentage of force damage he places as a requirement.

2) A specification as to the equality condition on the requirement, i.e., is it a minimum, a maximum, or an exact goal.

3) The weapon type, or types, he desires to take part in achievement of the damage requirements.

Item 3), above, is of special interest since it implies that the analyst cannot only specify a goal, but also which weapons meet the goal. Thus, it is possible to demand use of what the analyst thinks of as t[illegible]est" counterforce weapons.

c. <u>Reserve Force By Weapon Types</u>

The three strike, reserve force scenario (see IV-E) has typically been difficult to use because of the extremely difficult mathematical problems occurring when truly optimal reserves are desired. To help alleviate these problems, AEM has been previously modified to allow analyst specification of the reserve force. The presence of hedges now allows a third option - to allow analyst control over the types of weapons going into reserve without giving the complete specification.

[illegible] of the allowed hedging [illegible] added to the L.P. and they might [illegible] guide the [illegible] reserve into, or out of, certain [illegible]. [illegible] is the essence of the new option open through hedges.

For example, one can place upper [illegible] limits on the number of warheads of a given type going into the first strike attack, a total bound on the number of warheads of a given category, e.g., ICBM's, going in the first strike, or a limit on the total warheads of all types going in the first strike. These are primary examples but other forms could be used if desired.

A complete discussion of the detailed parameters necessary to use the hedging option is appropriate only to a document like the Arsenal Exchange Model Handbook. For the purposes of this report, it would be of interest, however, to re-summarize the types of constraint controls the analyst has.

By input one can design constraints which specify:

1) Which Force target classes are involved in the given constraint.

2) Which attacking weapon types, or categories are involved in the constraint.

3) The type of inequality involved, that is $\leq$, $=$, $\geq$.

4) The constraint value.

5) The general form of the constraint, i.e., is it a constraint on damage, or on weapon numbers?

For example, say there were ten force target classes and the [illegible] had five weapon types, two types being ICBM's and three types being SLBM's. An analyst might then generate constraints of the following types.

1) Do not attack Force targets with more than 500 total warheads.

2) Achieve at least X units of damage on the Force targets.

3) Attack target classes 3 through 7 only by ICBM class 2.

4) Allocate at least 250 SLBM warheads against Force targets.

It is obvious that the constraint design process could go on almost without end. However, the analyst must be careful lest he overcontrol the reserve force selection process. In essence, he must use the constraints in a reasonable manner.

Q. RANK ORDERED ATTACK ALLOCATIONS

1. The Problem

The concept of allocations within AEM has always involved the objective of maximizing total damage over a complete target set, without putting conditions on which targets must be destroyed. If no adequate value system to rate targets one to the other is possible, such an allocation objective becomes difficult to accept.

An alternate scheme which has some advantages is the notion of a ranked target list. Such a list indicates a strict order of preference and no balance of several targets in the list can be achieved when compared to a higher ranked target. Allocations to targets which are ranked in such an absolute manner can be conducted in AEM. This chapter describes the procedures used in such allocations.

2. The Methodology

In order to clarify the specific methodology used in obtaining rank ordered allocations some discussion of the rank-ordered attack concept would be useful. The concept deviations away from the standard weapon allocation process in AEM can thus be isolated.

In the classical concept of rank-ordered attacks, there is usually an ordered list of targets, a specified set of weapons to allocate against the targets and the condition that the final allocation must take on a form such that no target in the list is attacked unless all targets higher in the ordered list have been destroyed at least to some specified kill level. The overall objective of the process is then to produce an allocation which results in the required kill level being achieved on as many targets in the list as possible. In other words, to achieve maximum penetration down the ordered list.

The normal aspects to the problem which make determination of such an allocation non-trivial are: (1) the presence of several weapon types and (2) blending of this rank-ordered type of allocation for a sub-set of the targets in a larger list, where the remaining targets are to be attacked by a "maximize total damage" form of objective. Since both of these conditions are normal for AEM analyses, the final methodology must be capable of handling such variations.

In essence the above situations impact most strongly in two areas. They are reflected in the necessity to guarantee that: (1) no targets are skipped over when trying to penetrate the rank, while at the same time, (2) the easiest way possible to satisfy the rank conditions must be discovered. These two sub-objectives fight each other and a delicate balancing problem can result.

If ten different varieties of attacking weapons exist in an arsenal and if the targets are rather diverse in their vulnerabilities, by virtue of defense levels, etc., it can be a very large combinatorial problem to determine exactly which weapon type should satisfy the rank objective on each target so that deepest possible penetration of the list is attained.

However if only one weapon type existed, there would not be any combinatorial problem to consider and the job of computing the depth of penetration could be achieved simply by allocating weapons to each individual target in the list, in the order of the rank, until all weapons had been allocated. The number of weapons to allocate to any individual target is clearly determined by the desired kill level specified for the target.

In the circumstance where the rank-ordered list of targets can be completely penetrated, the allocation process must then be capable of switching to the normal "maximize total damage" on an additional list of non-ordered targets. In this case, it is very important that the ranked list was penetrated in the most efficient manner possible so that the "maximize total damage" objective can be truly maximized.

Consideration of the above aspects led to the following AEM methodology concept. Basically, the idea is to exploit the current AEM structure, which is totally dedicated to the idea of maximizing total damage, but to trick that methodology into achieving the real rank-ordered objectives.

For complete understanding of the current AEM allocation methodology, the reader should refer to Chapter IV-A. However, for our purposes here, it will suffice to say that a fundamental part of the allocation process is involved with a linear program whose objective it is to maximize total damage for a candidate set of individual weapon-to-target allocations for all the targets in the target list. A key item in the L.P. is the value which will be destroyed if any candidate weapon allocation to a specific target is chosen.

The primary technique, in manipulation of the AEM allocation process, therefore, is to modify the L.P. value destroyed in a way that will drive the L.P. process to a rank-ordered attack objective while the L.P. is actually maximizing total damage. However, this manipulation must be done very carefully so that the exact rank-ordered format is achieved.

Consider the following scheme. In order to impose the rank-ordered objective on a standard L.P. of AEM format, we must make it allocate weapons to a given target only if all targets above that one in the list are being attacked. This can be achieved by placing specially constructed, artificial value destroyed entries in the pay-off function of the L.P.

The artificial values would be selected so that the value destroyed per weapon allocated for any given weapon type would be largest for the first target in the ranked list, slightly smaller for the next ranked target and so on down the list. Given such a specially constructed pay-off function, no weapon of a type would be allocated to any given target as long as higher targets on the list were not being attacked.

However, this step alone will not guarantee that the correct weapon type will be assigned to each target so that maximum list penetration will occur. In order to insure that there is no favoritism to any given weapon type, it is necessary to guarantee that any strategy which achieves the desired kill on a given target contributes the same absolute value destroyed as any other strategy which gets the desired kill on that target.

In order to clarify how these special constructions occur, consider a case where three target classes exist and two weapon types exist. Further, consider the following hypothetical set of candidate strategies which are being provided to an L.P.

Target Class	Strategy	Value Destroyed Mod 1	Value Destroyed Mod 2
1	5 Weapons of Type 1	8.0	17.0
	2 Weapons of Type 2	3.0	17.0
2	3 Weapons of Type 1	4.0	10.0
	8 Weapons of Type 2	10.0	10.0
3	1 Weapon of Type 1	1.0	2.0
	2 Weapons of Type 2	2.0	2.0

Now, for the six strategies which are provided to the L.P., consider the value destroyed mod 1 column. Note that for each individual weapon type, the value destroyed per weapon on target class 1 exceeds that for class 2, which exceeds that for class 3. For example, for weapon type 2, the appropriate values are 3/2, 10/8 and 2/2 respectively.

However, note that the absolute value destroyed on each target is not constant. Especially, note that on class 2, weapon 2 is given credit for ten units of value destroyed, while weapon 1 would get the same kill level but only get credit for four units destroyed. This could well force weapon 2 to be used on this target.

If we construct another value destroyed variety (the mod 2 column), we can retain the desired ordering as in mod 1 but at the same time give each weapon credit for the same value destroyed on a given target. This is accomplished by working up the list from class 3 and insuring both conditions simultaneously.

In order to describe the technique in a more rigorous manner, consider the following recursive type of process. Let T be the total number of target classes in the ranked list. Further, assume that W weapon types exist and that all weapons have a potential strategy for each target type, where the strategy consists of some number of weapons assigned to that target. Denote this number of weapons as N(I,J), where I is the target subscript and J is the weapon subscript.

Now, according to our previous discussion, we desire to assign value destroyed units to each of the strategies and these assignments must be such that:

$$VD(I,J)/N(I,J) > VD(I+1,J)/N(I+1,J) \qquad (Q-1)$$

for all $(1 \le I < T)$

where:

VD(I,J) = value destroyed for the strategy for weapon J on target I

and such that

$$VD(I,J) = VD(I,K) \qquad (Q-2)$$

for $(1 \le K \le W)$ and $(1 \le I \le T)$

Equation (Q-1) simply forces the correct descending order for the weapon value destroyed units while equation (Q-2) says that all strategies on any target must be given credit for a constant value destroyed.

The recursive process would then start with the lowest ranked target class (I = T) and simply assign any arbitrary VD value to each strategy. For example, VD(3,1) = VD(3,2) = 2.0 in column 4 of the table.

Given this start, one can use equation 1 to compute

$$VD(2,1)/N(2,1) > VD(3,1)/N(3,1)$$

and

$$VD(2,2)/N(2,2) > VD(3,2)/N(3,2)$$

or

$$VD(2,1) > 2\cdot(3)/1 = 6$$

and

$$VD(2,2) > 2\cdot(8)/2 = 8$$

Theoretically we could choose any numbers that satisfies the above equations, as long as (from equation 2) VD(2,1) = VD(2,2). In this case, the value of ten was chosen, as is indicated in column 4 of the table. We can then go on to determine that

$$VD(1,1) > 16+2/3$$

and

$$VD(1,2) > 2+1/2$$

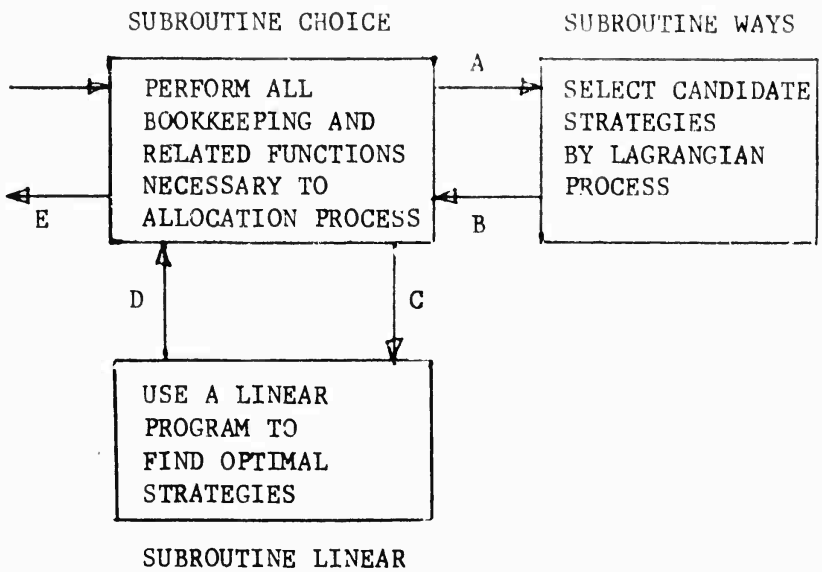
Since we also have the condition VD(1,1) = VD(1,2), we have chosen the value of 17 (see the table).

Thus, we have demonstrated that it is possible to use this recursive process to develop appropriate value destroyed units for any given set of strategies so that the L.P. will produce a rank-ordered attack. These values will essentially trick the L.P. into doing a rank-ordered attack even though it is programmed to maximize the total value destroyed.

In the case where target classes beyond clars T exist but they are not to be included in the ranked list, the only effect on the above process is involved in equation (Q-1). In that case, if I = T, then target class I + 1 must be arbitrarily defined as that member of the unranked list where weapon J will achieve the maximum value destroyed per weapon (VD(I+1,J)/N(I+1,J)). Thus, in such a case, equation (Q-1) must be satisfied for the last member in the list so that attacking the lowest ranked target is preferable to attacking an unranked target.

3. The Program Modification Scheme

Basically, the modifications to AEM were isolated in the optimal strategy generation loop. This loop can be diagrammed as follows:



In this operation, the logic flow is from Choice to Ways by Link A, return to Choice by Link B, more computations in Choice, proceed to Linear by Link C, return to Choice by Link D and then either repeat the process if the optimal allocation has not yet been found, or return to other AEM functions, by Link E, if it has been found.

The rank-ordered recursive process operates in Choice at the time the return from Ways occurs via Link B. At that time all the candidate L.P. strategies are available and the proper VD parameters can be set so that the rank condition will be imposed. Logic in Choice simply scans all the strategies and generates the appropriate VD units by equations (Q-1) and (Q-2).

4. Problem Areas

There are some additional problems caused within AEM by these rank conditions that should be discussed.

First, the [illegible] values for certain realistic cases can become rather large numbers (on the order of 10^8) and, as a result, the LP process encounters computer round-off problems. Thus, it was necessary to modify subroutine Linear so that these problems are minimized. However, more computational experience might reveal that additional work is necessary.

Second, in many cases, the rank condition might well be satisfied through natural allocations because of the inherent nature of realistic target systems. Thus, in order to minimize computer time, routines were added to Choice and Linear so that the rank-ordered conditions would only be imposed if the normal "maximize total value destroyed" allocation would <u>not</u> satisfy the rank conditions.

Third, since AEM uses weapon Lagrangian values in the force target value computation, it must be recognized that imposing rank-ordered conditions can conflict with the normal two and three-strike scenarios where the scenario objective is to maximize delta damage. In essence, the rank conditions are partially reflected in modifications to the weapon lambdas and this modification is not at all clear-cut.

In multi-strike scenarios, it is recommended that rank condition cases should be run as mis-estimate cases where neither opponent knows about his opponents' rank objective. This by-passes the conflict between max delta and rank attack objectives by simply letting each side attack his opponent as if the retaliation would be a max damage attack and not a rank attack.

Finally, there are some conceptual conflicts between rank-ordered attacks, island defenses and preferential defenses. For example, if one has island defenses, the normal AEM logic is to determine, for a given attack arsenal, which islands, and associated targets, should be attacked. However, if there are rank attack conditions, does this mean that there must be attack on any island which contains any ranked targets? Or, does it mean that in any island attacked the rank conditions must be followed? AEM is currently programmed for this latter practice.

When preferential defenses exist, there are convergence conflicts between the rank attack process and the correct attack plan through the defenses. This conflict is such that the final closure of the program on the correct attack plan might not occur. It is therefore recommended that a combination of this variety should be used with caution.

R. A VARIETY OF SMALL TOPICS

RA. The Assured Destruction Scenario

Since the assured destruction (AD) scenario is such a common analysis procedure in strategic analyses, it was deemed worthwhile to mechanize the scenario as a special AEM option so input of such a case would be as simple as possible. This section describes the implementation of concepts behind the scenario as viewed in AEM.

1. Definition and Task Approach

The implementation of an AD scenario is concerned primarily with data manipulation rather than mathematical developments. Thus, the methodology discussion will involve basic definitions of the scenario sequence of interest as it was programmed and the form of the data manipulation necessary to task accomplishment.

Assured destruction (AD) as described herein will be defined as the ability of a force to incur damage to an opponent after suffereing a severe counterforce attack. The AEM has been compatible with analysis of assured destruction capability for some time through thoughtful use of inputs. However, the job of input management for an in-dpeth study of a wide range of force postures was prohibitive. Thus, in this AEM option it is an objective to add control to AEM to set the proper parameter values if this type of case is desired. In this way the analyst [illegible]

The assured destruction scenario is based on the [illegible] game (a CF/CV attack followed by a CV retaliation) where the first strike is primarily counterforce. By current definitions, in an AD run if a weapon type *may* attack force targets, it *must* attack force targets in the first strike. It is also assumed that all first strike weapons are on generated alert to insure the largest practical counterforce attack. (Retaliating weapons are assumed to be on day-to-day alert.)

Unless otherwise specified by the analyst all other (non-force) military targets are removed from the game. This is done to increase the attack on retaliatory weapons in the first strike. In addition there is normally a limitation of the retaliatory response to urban targets only.

In certain circumstances the analyst might desire the inclusion of additional targets in assured destruction cases. Such a capability is offered in AEM through the use of special variables as follows.

A target parameter (TTYPE) which denotes a basic target class category and a scenario control (INCLUDE) which specifies those target categories which may be attacked on each strike. These variables are defined as

TTYPE(I,J) = 1 denotes that targets of class I possessed by side J are retaliatory force targets (weapon bases),

= 2 targets of class I possessed by side J are nonretaliatory, other military targets (OMT)

= 3 targets of class I possessed by side J are civilian value targets

= 4 targets of class I are non-retaliatory, Nth country targets which may be attacked when side J is attacked.

= 5 targets of class I on side J describe ABM defense installations for a missile subtractive defense island

= 6 targets of class I on side J describe a bomber defense installation for a bomber subtractive defense island.

INCLUDE(L,J)= all target TTYPE's which may be attacked by side J on strike L (L=1,2,3). Thus, INCLUDE(1,2)=14 would specify retaliatory force targets and Nth country targets only are to be attacked on a first strike performed by side 2.

For the particular case of assured destruction scenarios, the control INCLUDE, would normally allow only the retaliatory forces and civilian value targets to be attacked in the first strike with only those weapons designated as having no counter force capability being allowed on the civilian followed by a retaliation against only the civilian value targets of the initiator. This control may be overridden by specifying INCLUDE in the input array to include Nth country, OMT, or other target categories as desired.

The full use of these variables is described in the AEM-HEDGE Handbook. They are mentioned here only to indicate a capability.

In an AD type of run it may be desirable to investigate higher confidence in achieved damage. This is facilitated indirectly by means of a target hardness (ADHARD), used only in AD runs, which may be different than that used in ordinary analysis (HARD). Therefore, weapons may be easier to kill (ADHARD $\leq$ HARD) yielding a lower weapon survival than would normally be anticipated and the urban target may be harder to kill (ADHARD $\geq$ HARD) during the retaliation to gain confidence in the achieved damage.

A key part of the AD scenario is an option which would allow an investigation of the assured destruction level if one or more force components could not retaliate (retargeting is allowed for each component failure). As a result of this objective, AEM was programmed so the following sequence of responses is computed after the normal assured destruction computation: 1) ICBM's only, 2) SLBM's only, 3) aircraft only, 4) ICBM's plus SLBM's, 5) ICBM's plus aircraft, and 6) SLBM's plus aircraft. If a component is not in the basic force, they are deleted from the sequence, e.g., if there are no aircraft weapon types (JTYPE=3), response 3, 5, and 6 would not be computed.

Within the above definitions concerning the AD sequence, AEM input data and internal controls are set so that the analyst can indicate a desire to have an AD run and all other input mods would follow automatically [illegible] sets up the desired two-strike game, manipulates the alert rates and hardness levels as previously discussed and conducts the two-strike run.

At the completion of the two-strike run, the appropriate sequence of alternative responses in the second strike, as defined above, then occur in the form of individual one-strike games. That is, e.g., the ICBM only response is a one-strike game, the SLBM only response is a separate one-strike game and so on.

2. Option Utilization Cautions

If the assured destruction option is elected, there is virtually no further control of the scenario possible by input.

It is also important to note that the first strike is based on computed force values as if all components of the force will retaliate. The sequence by components at the end of the game assumes component failure was known by the retaliator prior to the war. This sequence is optimal if this knowledge exists.

RB. Impacting Megatons

1. General

The strike summaries in AEM HEDGE include both impacting equivalent megatons and true impacting one megaton warheads. The difference is in the method of computation which is described here so the analyst may decide which (if either) computation is meaningful in his analysis.

2. Impacting Equivalent Megatons (EMT)

This parameter is based on a scaling law of the form:

$$EMT = W_I \cdot Y^{2/3}; \quad Y \leq 1.$$

$$EMT = W_I \cdot Y^{1/2}; \quad Y > 1.$$

where

EMT is the impacting equivalent megatons

W_I is the number of impacting warheads

Y is the weapon yield in megatons

The question then remains how to determine the number of impacting warheads.

Consider an undefended point target attacked by a weapon of reliability R. If A weapons are allocated, $A \cdot R$ are expected to impact. If D perfect defenders are added, a translation shift of T units in the damage function where $T = D/R$ is caused and the number of impacts is $(A - T) \cdot R$.

There may be other target types having areas greater than zero and/or imperfect defenses. For these cases, an effective reliability R^*

and an effective translation T' are computed. The [illegible] weapon, damage parameters (P', T') are computed for the [illegible] and target (with no defense). The effective reliability is [illegible] ($R^* = P/P'$ where P is the imperfect single shot kill). To allow [illegible] area targets which normally have a translation, T' is modified by reliability considerations to produce T^*. Thus, if the target truly has no translation, ($T' = \emptyset$) e.g., point target, T^* will also equal zero. If the target has a positive area, the difference in T and T^* is an approximation of the defense price. The general formulation is then:

$$W_I \approx R^* \cdot A - (T - T^*)$$

If $T^* = \emptyset$, the approximation is exact with the exception of targets having infinite area with value distributed according to the circular normal distribution. In this case W_I is only an approximation.

3. True Impacting One Megaton Warheads (TEMW)

This computation requires the derivation of the damage function for a perfectly reliable, one megaton weapon which is impervious to defenses. The weapon is further described by a .25 nautical mile CEP. The question then asked, strategy by strategy, is how many of these "perfect" weapons would be required to obtain the same expected damage on the targets attacked by this strategy. An integer condition is applied as follows: consider a strategy used to [illegible] target with expectations of killing 90% of all ten targets. If the "perfect" one megaton weapon has a perfect damage function ($P = 1$, $T = \emptyset$), nine true one megatons would be equivalent (one on each of nine targets) in terms of expected damage. In general, if PK denotes the expected damage for a selected strategy used S times:

$$TEMT = (X \cdot I) + (S - X) \cdot (I + 1)$$

where

$$X = \frac{S\ (PK - PI)}{(PU - PI)}$$

$$PI = 1 - (1 - P)^{I}$$

$$PU = 1 - (1 - P)^{I + 1}$$

$$I = \left| \frac{\ln\ (1 - PK)}{\ln\ (1 - P)} \right|$$

P = "perfect" weapon PKSS for the target involved

Note that $PI < PK < PU$.

RC. A Specialized Bomber Model

There are two complex issues which are not dealt with in standard AEM bomber relationships. These issues are: (1) the fact that bomber loadings are generally mixed in nature, and (2) bomber penetration of area defenses is a complex function of the bomber types penetrating and the manned interceptor types in the defense. These issues have been addressed to some degree by insertion of a specialized bomber model into AEM. This section will discuss some of the key issues pertinent to that model.

1. Bomber Loading

In the real world a bomber can obviously carry a complex combination of offensive weaponry. For example, gravity-drop bombs, air-to-surface missiles and air-launched decoys are all candidates to be loaded onto a given bomber. There might be reasons why some of the many possible combinations would never exist on a single bomber, but the general problem still remains.

AEM has a feature which makes such mixed loads an even more complex issue. All internal weapon allocations and bookkeeping in AEM implies a one-to-one correspondence between weapon types and base types. That is, a given base is not allowed to contain more than one kind of weapon. Thus, mixed load bombers cannot even be allowed in AEM.

Within the specialized bomber model, herein called BOMBER, these aspects are worked out to somewhat of a satisfactory conclusion.

First, however, it should be recognized that the analyst can do any aggregation he desires and totally avoid the use of BOMBER. However, on demand BOMBER will perform the following functions.

For one of the sides in the two-sided exchange, the analyst can input (see the AEM Users' Handbook for full details).

B(1), B(2), B(3)	=	Inventory numbers of three different kinds of bombers
SRAM	=	Total number of air-launched missiles in inventory
SCAD	=	Total number of air-launched decoys in inventory
BASES	=	Total number of bases occupied by the total set of bombers

The loading part of BOMBER simply distributes these total inventories in such a manner as to maintain consistency in the number of weapons available for use and the number of bases which contain them. In doing so, however, it creates artificial basing categories so that a given base type will contain only one type of weapon. For example, there will be a category of B(1) that carries only gravity-drop bombs, a category of B(1) that carries only SRAM, a category of B(2) that carries only SCAD, etc.

The intent of this distributed loading is to live within the AEM limitation that each base type can only possess a single weapon type, but not force the analyst to compute some artificial loadings by hand.

While performing this bomber loading function appropriate detail is paid to such issues as: bombers not available because of training and withhold options, alert rates, load capacities and general bomber type (of which three are allowed). These issues simply modify the loading procedures used in partitioning out the total inventories to the various single-weapon base types that are created.

The precise loading rules cannot be presented here, but basically the procedure is as follows. First, add up the total number of bombers and divide by the total bases to obtain an average base loading. Then, take the total inventories of SRAM and SCAD and distribute them among the three bomber types according to their capacities. In doing so SRAM bombers and SCAD bombers are created. Assign remaining bombers as gravity-drop bombers. Then use the average base loading to convert all bases into bases of each type.

At present in AEM not all possible base types are created. Reference to the AEM Users' Handbook will provide an understanding of six types that are created and the analyst inputs available to control the whole process.

This creation of artificial base types is acceptable for AEM type analyses for the following basic reason. It is usually true that all bomber bases get attacked to a rather high level, whatever the assumed loading. Thus, there is no misleading survivability effect because of the artificial base type. In such a case, the off-on-warning weapons are the only survivors and from case-to-case this fact remains rather constant.

Since the number of bombers surviving [illegible], it is then possible to compute bomber penetration probabilities [illegible]. The procedure for computing such probabilities [illegible] next section. Then, when the bomber weapons are allocated, there is no distinction of which bomber carried each weapon; the strategy simply says how many weapons of a given type are allocated to a given target.

2. Bomber Penetration

A number of bomber penetration studies have been conducted by many groups over the years in an attempt to develop aggregated bomber penetration models. These efforts have usually been concerned with such issues as geography, multiple interceptor types, tactics, number of passes per interceptor, acquisition and interception probabilities, etc. One such continuing effort has been conducted by the Air Force group in the Office of the Assistant Chief of Staff, Studies and Analysis (AFCSA). The penetration relationship to be presented here is from that source.

The basic AFCSA equation used in this penetration model is as follows:

$$B(i) = B(i-1) \left\{ 1 - PK \left[1 - \exp \left(- PD \cdot I/B(i-1) \right) \right] \right\} \qquad \text{(RC-1)}$$

where: $B(i)$ = number of bombers surviving the i^{th} wave made by I interceptors

PK = probability that an interceptor kills the bomber in a single pass

PD = probability that an interceptor detects a bomber and converts to an attack position

I = number of reliable interceptors available for possible engagement of the bombers in the i^{th} wave

This equation is based on several key assumptions. First, it is assumed that the interceptor control environment is adequate enough to vector the interceptors into the vicinity of the bomber raid. Second, it is assumed that interceptors then encounter bombers in the raid in a random manner, with no overall control of the final vectoring. Third, it is assumed that some number of interceptor waves engage the bomber raid and that each succeeding wave encounters only survivors from previous waves.

It should be observed that the interceptor effectiveness is contained in the parameters PK and PD. Other than these two parameters the key issues are the ratio of interceptors to bombers (I/B), and the number of waves allowed to the interceptors.

If there are multiple interceptor types, this equation can be expanded to the following:

$$B(i) = B(i-1) \left\{ 1 - PK(j) \left[1 - \exp -\sum_j PD(j) \cdot I(j)/B(i-1) \right] \right\} \qquad \text{(RC-2)}$$

where: PK(j) = PK by interceptor type j in a single pass

PD(j) = Detection and conversion probability by interceptor type j

I(j) = Number of interceptors of type j

In addition, presence of multiple bomber types can be represented by computing the above equation for each bomber type and by making PK and PD functions of j and the bomber type.

This basic equation has been adapted for use in AEM as follows. Basically, each of the bomber categories created in the loading part of BOMBER I is identified as belonging to one of five different penetration classes. For each penetration class there are sets of parameters describing the effectiveness of each interceptor category against the class. (There are up to six different interceptor types allowed.)

For example, one bomber category might be thought of as a good penetrator against some interceptor types but not against other types. For this bomber category, and for each interceptor category, the interceptor effectiveness is described in terms of PK and PD values, the number of turnarounds, or waves, the interceptor could make on that bomber category, and the number of passes the interceptor could make in any single wave attack.

Given that effectiveness values are available for each interceptor type against each penetration class, the program then uses equation (RC-2) to compute the number of bombers of that class which will survive wave number 1. The survivors of that wave are then engaged in another wave, and so on. The number of passes a given interceptor type can perform in a given wave is used as a multiplier of I(j) in the equation.

The number of waves is a parameter which must be chosen on the basis of the expected geometry of the penetration routes, aircraft performance, etc. It also turns out that the development of the basic equation by AFCSA revealed that good prediction of penetration can be achieved only if the wave parameter takes on certain values. This led to the following choice of the number of waves.

The basic assumption is that four waves will occur and that one-fourth of all available and reliable interceptors will take part in each wave. However, if four waves result in a ratio of interceptors per bomber less than .3, the number of waves is cut back to a number of waves such that

$$W = \left| \frac{I}{.3B} \right| + 1 \qquad \text{(RC-3)}$$

where: W = number of waves

$|A|$ = largest integer included in A

I = total available and reliable interceptors

B = number of arriving bombers

Use of this procedure for selection of the number of waves has resulted in adequate accuracy of computation of penetration losses when the PD and PK parameters are also appropriately chosen.

3. The BOMBER Output

Once the loading and penetration computations are completed BOMBER creates special AEM format inputs for the analyst. These variables are: the number of bases of each artificial type, the number of bombers on each base type, the number of warheads of each type off-on-warning and the reliability of each warhead type (which includes penetration probability).

These inputs are generated in the front of AEM, just where the inputs get into the program. Because of this location none of the factors are affected by the scenario outcome. This is somewhat critical in the case of penetration, but, as has been discussed, it is not generally a problem since only off-on-warning weapons are generally survivors and that is non-scenario outcome dependent.

4. <u>A Generalized Capability</u>

More generalized bomber loading and penetration options have been developed from these simple concepts. Discussion of those capabilities in AEM is the subject of reference 21. Contained therein is considerable insight into additional aspects to the issues presented here. Since the procedures discussed in reference 21 are mostly of a bookkeeping variety they are not reproduced here.

RD. RV Impact Limits

The current version of AEM has a rapid method for calculating an approximate number of impacting RV's which is correct for perfectly defended point targets and is a good approximation for targets with defense leakage. This method, which was described in IV-RB, was adapted to allow control over the maximum allowed number of RV's impacting on a point target by appropriate control over the strategies allowed to enter the weapon allocation process.

Consider the following equation for computing the number of RV impacts on a target when the defense is perfect.

$$I = A \cdot R - D \qquad (1)$$

where: I = Number of impacts.

A = Number of RV's of a given weapon type allocated to a target of some given class.

R = RV reliability for the given weapon type.

D = Number of RV's lost to terminal defense located at the target.

The $A \cdot R$ part of the equation is simply the expected arrivers at the defended target and the D subtraction indicates the reduction of those arrivers by the defense.

In terms of AEM processes, this equation can be transformed in the perfect defense case to the following for a point target:

$$I = (A - ANZ) \cdot R \qquad (2)$$

where:

ANZ = Damage function translation = $\frac{D}{R}$ parameter.

If one considers the R and ANZ terms in equation (2) as "loss factors," equation (2) is generalized in AEM for the leaky defense case by the following approximation:

$$R \approx \frac{P_K}{p}$$

where:

P_K = Damage function single shot kill parameter.

p = Damage function single shot kill parameter for the same target in a no defense, perfect reliability weapon circumstance.

Substituting this relationship into (2) we get:

$$I = (A - ANZ) \cdot \frac{P_K}{p} \qquad (3)$$

Solving (3) for A (the maximum number of RV's to be allocated to a target):

$$A = \frac{I \cdot p}{P_K} + ANZ \qquad (4)$$

where I is now the desired limit on number of impacts. The method thus reduces down to use of equation (4) to compute the maximum number of allocated RV's such that the expected impacts on the target will not exceed I. This number is used in AEM to prohibit any allocations which would exceed I for any weapon-target combination.

S. WEAPON DEFENSE MODULES

1. General

A weapon defense module is defined as the common defense of multiple weapon silos by a single defensive system. The silos are assumed to have adequate separation in distance to be considered separate targets inside the module. However, any silo within the module may be defended if the defense is operative. The defense is vulnerable and will be considered as an attack option.

A weapon defense module is described by the number of radars and interceptors possessed by the defense and the number of silos in the module. Only the radars and silos are assumed vulnerable (an attack on the interceptors is not considered). Ladder down, blackout, and other variations dependent on sequence and timing of the attack are not considered. Therefore, defenses are suppressed only by blast damage on the radars from a penetrating warhead. In addition, leakage parameters are assumed constant independent of numbers of warheads or warhead type.

These assumptions prohibit detailed analysis of various systems with these models. However, the effects of such modular defenses or exchange analysis can be analyzed. There are two spearate models in AEM which operate in different environments (Safeguard and Hard Site Defense). There is currently no way in AEM for any weapons to be simultaneously defended by both defense types.

The Safeguard assumes high altitude, endoatmospheric interceptors which are used in a shoot-look-shoot mode against incoming warheads.

All warheads are assumed to arrive simultaneously in any anti-radar attack. The radar attack level is identical for all radars in terms of allocated warheads. The radars are netted and commonly defended, however, the survival of more than one radar may be required to maintain a credible defense. Interceptors are used against a random selection of warheads attacking silos if the defense survives the radar attack.

The Hard Site Defense model assumes low altitude intercepts (after terminal decoys are aerodynamically stripped). Only one interceptor is assigned to each incoming warhead directed against radars. However, a variety of firing doctrines may be used if the warhead is directed against silos. The radars are completely netted and of such capability that only a single radar survivor is required. Survival may be enhanced by the creation of a collection pool of interceptors which are assigned to one radar which has successfully used its share of interceptors.

These models are very different in mathematical derivation. The Safeguard model uses expected values almost exclusively, while the Hard Site Defense model uses the distributions implied by certain mean values. The formulation of each model is discussed below.

2. Safeguard Defenses

These defenses protect a larger area than Hard Site Defenses due to the assumption that high altitude intercepts are accomplished. Therefore, the number of silos within a module may be reasonably large and still assume sufficient separation for independent targeting. In addition, radars are considered sufficiently separated to preclude multiple kill or "accidental" kill. The anti-silo attack arrives simultaneously and may be preceded by a simultaneously arriving anti-radar attack. All radars are attacked by an identical number of warheads. (Even though it may not be necessary to kill all radars to nullify the defense.)

Since Safeguard operates in the high altitude environment, decoys must be considered. In keeping with other defense models in AEM, the following definitions are made:

SGPA - The probability an object is acquired by the Safeguard radars and is placed in the firing list.

SGPD - The probability an incoming decoy is determined to be a decoy.

SGPI - The single shot lethality of a Safeguard interceptor against an incoming warhead.

OBJ - The number of terminal decoys accompanying each arriving warhead.

Decoys are assumed to be the equivalent of a warhead in terms of acquisition and intercept lethality. This assumption is dubious unless intercept lethality is totally described by in-flight reliability (interceptor warhead lethality is perfect).

Acquisition type leakage is most likely due to the mass of the attack and probably should be a function of the attack size where the simultaneous tracking capability has been exceeded or queues develop in data transmission channels. The level of model detail prohibited serious analysis of these considerations. Nevertheless, the factor is included to allow whatever unopposed leakage the analyst might desire as a constant factor.

Until the defense is exhausted, one interceptor is assigned to each acquired and undiscriminated object. A second interceptor will be sent only if the first failed. No objects are assigned more than two interceptors.

The objective of the model is to find, for a given total attack (MA), that radar attack which maximizes the expected number of silos destroyed. Thus, if there are R radars and M weapons sent against each, there is some probability (PKR) that the defense has been nullified. If the defense was killed, the effectiveness of the remaining attack (MA - R · M) will not be affected by the defense.

We first address those factors considered during the radar attack. The question of how large M should be cannot be answered until both radar and silo attacks are considered together. Therefore, we shall begin by determining how large M can be.

There exists a maximum expected number of intercepts which may be accomplished by a stated number of interceptors (INT). For every acquired and undiscriminated object, one interceptor will be used. If it is not successful, a second interceptor is fired. The expected number of interceptors used against each acquired and undiscriminated object (INT/O) (assuming no discrimination by blast) is thus:

$$\begin{aligned} INT/O &= 1 \cdot SGPI + 2 \cdot (1 - SGPI) \\ &= 2 - SGPI \end{aligned} \qquad (S-1)$$

Note that O may be either a warhead or decoy.

The defense is expected to be exhausted when the number of objects acquired by the defense = INT/(2 - SGPI), or

$$INT/(2 - SGPI) = N^{*} \cdot r \cdot SGPA \cdot (1 + (1 - SGPD) \cdot OBJ) \quad (S\text{-}2)$$

Where r is the in-flight reliability of the attacking weapons and N^{*} is the expected number of warheads allocated against the module to achieve exhaustion. Therefore, we may assume the radar attack is successful if more than N^{*} warheads are used in the radar attack (since expected values are assumed). Hence, the maximum radar (MAXR) attack considered is

$$MAXR = \left| \frac{N^{*}}{R} + R \right| \quad (S\text{-}3)$$

Equation (S-3) is a limit only in the expected value. We rewrite Equation (S-2) to further explore the assumptions involved by this definition of MAXR.

$$N^{*} = \frac{INT}{(2 - SGPI) \cdot r \cdot SGPA \cdot (1 + (1 - SGPD) \cdot OBJ)} \quad (S\text{-}4)$$

Note that r, SGPA, SGPD, SGPI are means of distributions, each having a possible upper limit of one and lower limit of zero. If all are equal to one, N^{*} is exact. Note also that N^{*} increases with increasing SGPI and SGPD, and decreases with increasing r and SGPA. If the defense is perfect, indeed, if SGPI, SGPA are 1, there is no advantage in attacking the radars at all.

If leakage is introduced, the advantage of a radar attack is to prevent all INT interceptors from being used. Therefore, the relative advantage of a radar attack may be described in the limit as

$$(MA - M \cdot R) \cdot PKR > \left(1 - \frac{N^*}{MA}\right) MA; \quad (MA > N^*)$$

The left hand side of the above inequality considers the extreme case where the silo attack accrues damage only if the radars have been killed and represents the expected number of attackers in the silo attack. The right hand portion of the inequality is the expected number of attackers in the silo attack if the radars are not attacked but are finitely limited to N^* random deletions from MA attackers. The left hand portion is limiting since other leakage does exist and the defense is not infinite and is therefore a lower bound on the expected attackers. We may then find an upper bound on M since it appears negatively in this lower bound by stating the equations as an equality and setting PKP equal to one. (Note that if $N^* > MA$, the MAXR in Equation (S-3) is greater than $\frac{MA}{R}$.)

Thus: $MA - M \cdot R = MA - N^*$

$$M \cdot R = N^*$$

$$M = \frac{N^*}{R} = \text{upper bound on } M$$

Therefore, no radar attack exceeding MAXR as defined in Equation (S-3) need be considered.

However, there exists some positive probability that the defense is nullified prior to the exhaustion limit of MAXR. This is true because of leakage and because only a subset of R must be killed to nullify the defense. We denote this subset by defining RK as the number of surviving radars when the defense becomes inoperative (i.e., R - RK must be killed to nullify the defense). We now consider the event tree for the radar attack in Figure S-1.

There are thus two closely related ways to kill the radar and three ways for the radar to survive. The probability of killing the radar with one warhead is thus:

$$PKRSS = r \left[(1 - SGPA) + SGPA * (1 - SGPI)^2 \right] \qquad (S-5)$$

since, if acquired, the defense must be unsuccessful on both intercept attempts. The probability of the radar being alive after M warheads are allocated is then:

$$PRS(M) = (1 - PKRSS)^M \qquad (S-6)$$

This assumes no capability degradations as a function of attack size. We must now compute PKR(M), since any RK radars may survive and the defense be inoperative. We will employ the binomial expansion such that the probability that X of R radars survive an attack by M warhead against each radar is:

$$P(X) = \binom{R}{X} PRS(M)^X \; (1 - PRS(M))^{R-X} \qquad (S-7)$$

and the probability that the defense has been nullified by this attack is:

$$PKR\ (M) = \sum_{i=0}^{RK} \binom{R}{i} PRS(M)^i \; (1 - PRS(M))^{R-i} \qquad (S-8)$$

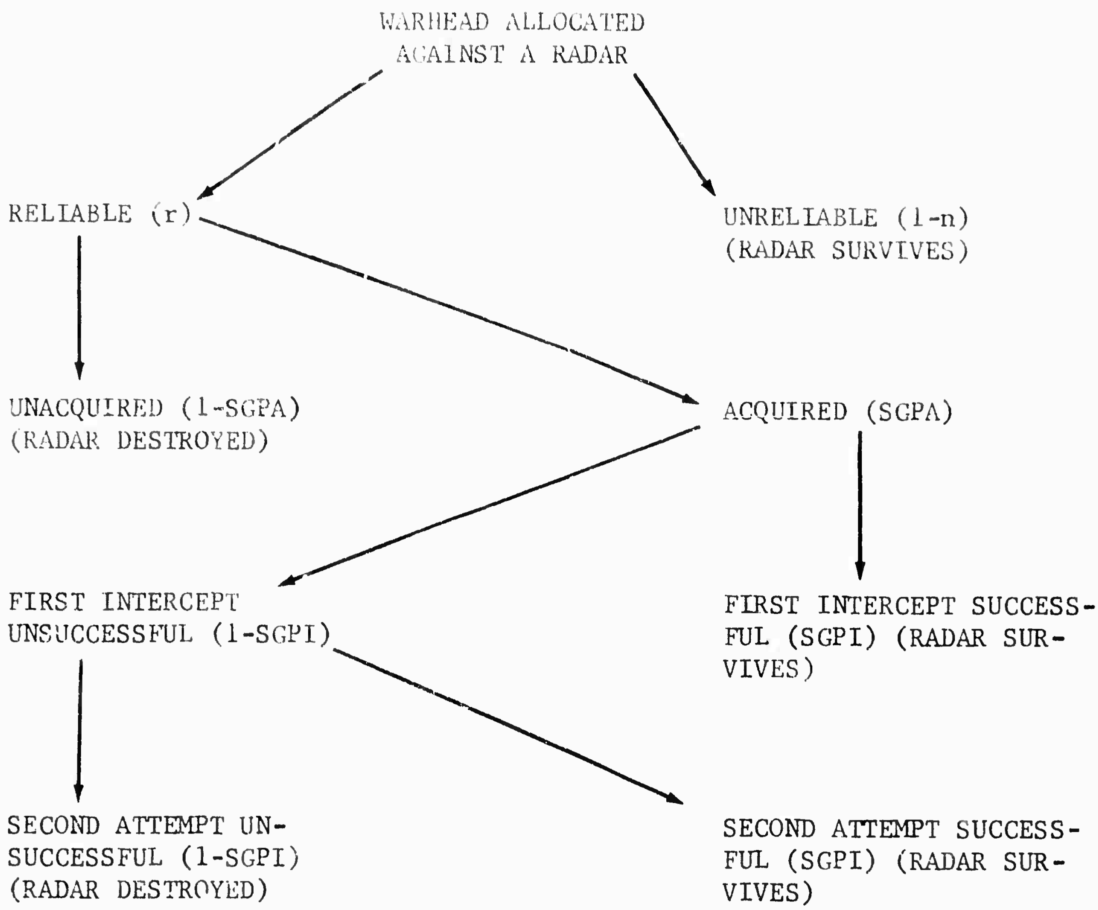


FIGURE S-1: SAFEGUARD RADAR ATTACK EVENT TREE

We are now prepared to consider the events which can occur in the silo attack. Figure S-2 depicts all possible events concerning an allocated warhead.

As may be seen in this tree, there are five ways to penetrate and only three ways not to penetrate (unreliable weapon, killed by first intercept, killed by second intercept). From these eight events we can deduce the probabilities for a single warhead to penetrate (PP) and not to penetrate (PNP) as follows:

$$\begin{aligned} PP = {} & r\,(1 - PKR) + r\,(1 - PKR)\,(1 - SGPA) \\ & + r\,(1 - PKR) \cdot SGPA\,(1 - PDR) \\ & + r\,(1 - PKR) \cdot SGPA + r\,(1 - PKR) \cdot SGPA \\ & \quad \cdot PDR\,(1 - SGPI)\,(1 - PDR') \\ & + r\,(1 - PKR) \cdot SGPA \cdot PDR \cdot (1 - SGPI) \\ & \quad \cdot PDR' \cdot (1 - SGPI) \end{aligned} \qquad \text{(S-9)}$$

$$\begin{aligned} PNP = {} & (1 - r) + r\,(1 - PKR)\,SGPA \cdot PDR \cdot SGPI \\ & + r\,(1 - PKR)\,SGPA \cdot PDR \cdot (1 - SGPI) \cdot PDR' \cdot SGPI \end{aligned} \qquad \text{(S-10)}$$

PKR is the probability of killing the radars (M attackers per radar).

PDR is the probability the defense has not been exhausted prior to first intercept attempt.

PDR' is the probability the defense was not exhausted on the first intercept attempt.

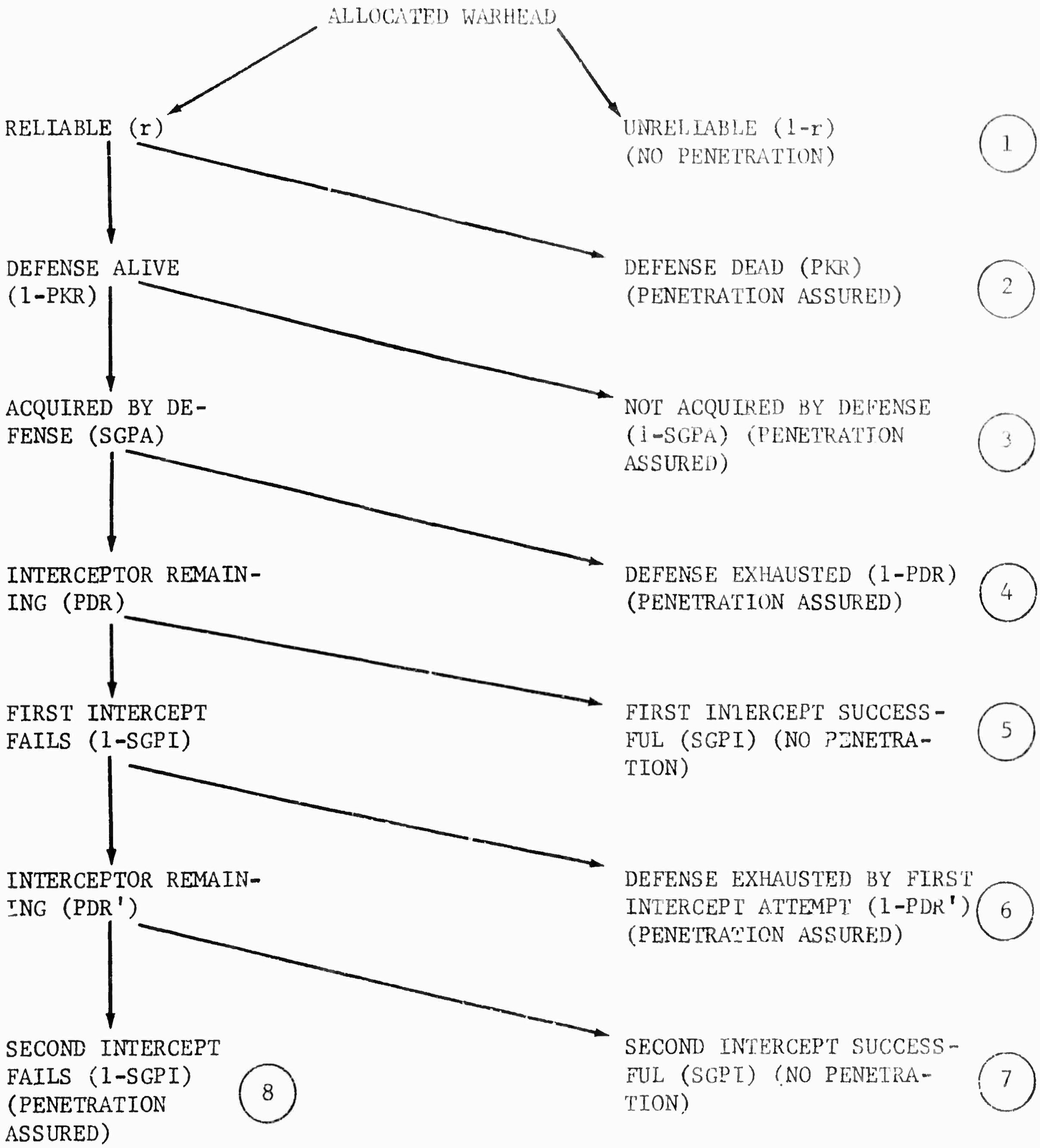


FIGURE S-2: SAFEGUARD DEFENSE SILO ATTACK EVENT TREE

From Equation (S-4):

$$PDR = 1 \qquad MA \leq N^*$$

$$PDR = \frac{N^*}{MA} \qquad MA > N^* \qquad (S-11)$$

If we assume PDR' involves the very next interceptor, i.e., a hold is placed on another interceptor until the success of the first attempt is ascertained, 1 - PDR' may be considered as the partial of PDR with respect to INT, i.e., the probability no interceptor is available for holding at the time of first intercept attempt is:

$$1 - PDR' = \frac{\partial\ PDR}{\partial\ INT} = \frac{1}{MA} \quad \frac{\partial\ N^*}{\partial\ INT}$$

$$(S-12)$$

$$= \frac{1}{MA\ (2 - SGPI) \cdot RL \cdot SGPA\ \ 1 + (1 - SGPD)\ OBJ)}$$

$$\approx 0 \quad \text{(for large MA)}$$

or, $\quad PDR' \approx 1.$

Noting that PP = 1 - PNP, we are more content to work with Equation (S-10). Incorporating the results of (S-12), we rewrite (S-10) as:

$$PNP = 1 - r + r\ (1 - PKR)\ PDR \cdot SGPA$$

$$\cdot\ SGPI\ (1 + (1 - SGPI)) \qquad (S-13)$$

and the probability of killing a silo (PKS) by an allocated warhead as:

$$PKS = (1 - PNP) \cdot PKSS \qquad (S-14)$$

where: PKSS is the single shot probability of kill for a perfectly reliable weapon against a silo.

Two terms in Equation (S-13) are controlled by the offense (PKR, PDR) if he chooses to employ an initial radar attack. The objective of this model is to find that radar attack M (hence, PKR, PDR) such that the greatest expected silo destruction, SK, is enjoyed for a stated total attack size MA:

$$\max \left\{ SK = N \left[1 - (1 - PKS)^{L} \right] + (NS - N) \left[1 - (1 - PKS)^{L+1} \right. \right\} \qquad (S-15)$$

where: N of NS silos are attacked with L warheads each.

The remaining silos attacked with L + 1 warheads each.

This model selects the maximum SK for a considered attack MA by straightforward enumeration of all integer M values which are less than MAXR defined in Equation (S-3) or MA if less than MAXR. Note that $L \cdot N + (L + 1)(NS - N) = MA - M \cdot R$ where M warheads are directed against each radar.

Examples of Safeguard functions and considerations in AEM HEDGE are presented in paragraph IV-S-4.

3. Hard Site Defense (HSD)

The HSD model is similar to the Safeguard model in that the objective is to maximize expected silo destruction for a stated attack level MA which may include a radar attack prior to a silo attack. However, since HSD is a low attitude defense, the module size is probably smaller.

Radars are assumed to be completely netted so that all must be killed to nullify the defense. Any radar attack is assumed sequential in nature with one-on-one assignment of interceptors. No acquisition type of leakage is assumed and all decoys are assumed to be aerodynamically stripped prior to defensive action. These differences dictate comparatively small numbers of objects and interceptors.

Additionally, whereas all radars are commonly defended in the Safeguard model, they independently provide for their own defense in HSD. Therefore, each radar is assumed to have a percent of the total interceptors it may use for its own defense specified by INT/R (R is the number of radars and INT is the total number of interceptors). However, a specified percentage of these defenders are donated by each radar to a common pool. This pool of interceptors will be assigned during the radar attack (if necessary) to additionally defend one radar which has successfully used all available interceptors in his particular initial batch of INT/R interceptors minus his donation.

This assignment of additional defenders is assumed to be random among all radars who have successfully used their initial interceptors. Other rules might be considered, e.g., the radar which had the most available initial defenders and used them all successfully. However, this would assume some knowledge about the number of warheads allocated against that radar but which were unreliable. It also assumes that a proportional attack is yet to come, i.e., this radar has already survived a greater portion of the total attack directed against it. This concept has credibility if warheads are directed against the radars in waves of R. With a more random attack ordering, it is less clear what the rule should be. By assuming random solution of the additional defense assignment, we may assume a random attack order (at least in the eyes of the defense).

The radar attack results in the deletion of some available interceptors and some probability that the defense is inoperative. The silo attack is weighed by these considerations. If at least one radar survives, all remaining, available interceptors will be used in the silo defense and will employ one of the following doctrines: subtractive, random, Prim-Read, or preferential.

Basically, subtractive defenses attempt intercepts on the first INT_S reliable warheads, where INT_S is the number of available interceptors remaining after the radar attack. Random defense is more defensively optimistic in that the entire silo attack appears simultaneously, allowing multiple interceptor assignments to warheads if fewer reliable warheads than INT_S appear. Prim-Read defense attempts to allocate a decreasing number of interceptors to reliable warheads as they sequentially appear in order to approximate the random defense at that attack which produces the maximum return per attacking warhead. Preferential defense is assumed to see the entire attack, to know where each warhead will impact, and to assign one on one to maximize the number of silos saved.

It should be noted that the random doctrine is superior if INT_S is greater than the number of reliable attacking warheads. The preferential defense is superior for silo [illegible] respect to INT_s, Prim-Read is in between the subtractive and random doctrines. However, in all cases, the [illegible] the damage capability of the attacking force.

Distributional variations are not treated in the silo attack. If warhead reliability is low, the silo attack will be large and the expected silo destruction is relatively insensitive to small changes in the attack. If reliability is high, the distribution of events is small and gathered at the expected value. While a medium value of reliability might be worthy of distribution analysis, it is not done. Since several silos are in the module, the errors of expected value analysis are felt to be within AEM tolerances.

The mathematical discussions pertinent to both the radar attack and silo attack are presented below.

a. <u>Hard Site Defense-Radar Attack</u>

In this defense module, two levels of defense are possible since one radar may be assigned an additional number of defenders if all initial available interceptors have successfully been used. For convenience, we denote these as lower defense events for the initial defenders, and upper defense events for the one radar receiving the additional defenders. As mentioned previously, all possible events of arriving warheads and available interceptors are considered as independent event probabilities Thus, I of M attackers may arrive at the radar and

J of L interceptors may be available for allocation. We will consider three independent states in the lower defense events for a radar having M attackers allocated to it:

1) The radar survives the attack without being additionally defended.

2) The radar was killed prior to using all initial defenders.

3) The radar qualified for the additional defenders.

The third state implies that the radar is killed if it does not win the additional defenders. There are in addition two states for the single radar in the upper defense events, i.e., additionally defended:

4) Radar is additionally defended and survives.

5) Radar is additionally defended and is killed.

We will consider the three states for the lower defense events as independent. One radar in state three (if any) also has two possible independent states in the upper defense events. It is now pertinent to define the probabilities of each state from an analysis of defense events.

Let: SP(N) denote the probability of a radar being in state N; PA(I) denote the probability of I arriving warheads at a radar (M warheads allocated); PL(J) denote the probability of J interceptors being available in the lower defense (L total interceptors); PU(K) denote the probability of K interceptors being available in the upper defense (U total interceptors); and PK denote the interceptor lethality against a warhead. We may then define SP as follows:

$$SP(1) \quad \sum_{i=0}^{L} PA(i) \sum_{j=i}^{L} PL(j) \cdot PK^{i} \qquad \text{(S-16) (a)}$$

$$SP(2) = \sum_{i=1}^{M} PA(i) \sum_{j=0}^{\min(i-1,\,L-1)} PK^{j} \, (1 - PK)$$

$$\sum_{m=i+j}^{L} PL(m) \qquad \text{(b)}$$

$$SP(3) = \sum_{i=1}^{M} PA(i) \sum_{j=0}^{i-1} PL(J) \cdot PK^{j} \qquad \text{(c)}$$

SP(3) is used to generate SP(4) and SP(5) for that radar additionally defended by:

$$SP(4) = \sum_{i=1}^{M} PA(i) \sum_{j=0}^{i-1} PL(j) \, PK^{j}$$

$$\sum_{k=i-j}^{U} PU(k) \, PK^{i-j} \qquad \text{(d)}$$

$$SP(5) = \sum_{i=1}^{M} PA(i) \sum_{j=0}^{i-1} PL(j) \, PK^{j} \left[\sum_{k=0}^{i-j-1} PU(k) + \right.$$

$$\left. \sum_{k=i-j}^{U} PU(k) \sum_{n=0}^{k-1} PK^{n} \, (1 - PK) \right] \qquad \text{(e)}$$

We define PA(i) by the binomial distribution on in-flight reliability (r) such that

$$PA(i) = \binom{M}{i} r^{i} (1 - r)^{M-i} \qquad \text{(S-17)}$$

The variables PL(j) and PU(k) are similarly defined by the binomial distribution on interceptor availability (DAV) such that

$$PL(j) = \binom{L}{j} DAV^{j} (1 - DAV)^{L - j} \qquad \text{(S-18)}$$

$$PU(k) = \binom{U}{k} DAV^{k} (1 - DAV)^{U - k} \qquad \text{(S-19)}$$

$$L = \frac{INT}{R} (1 - AD)$$

$$U = INT - L \cdot R$$

AD = Fractional donation by each radar to the common pool.

The above equations are more easily understood by considering the event tree in Figure S-3.

Note that if there is no additional defense, there are three ways for the radar to be killed and one way to survive. If additional defense exists, there are four ways for the radar to be killed and two ways to survive - but only for one radar in state SP(3). Therefore,

$$SP(1) + SP(2) + SP(3) = 1 \qquad \text{(S-20 (a))}$$

and

$$SP(4) + SP(5) = SP(3) \qquad \text{(b)}$$

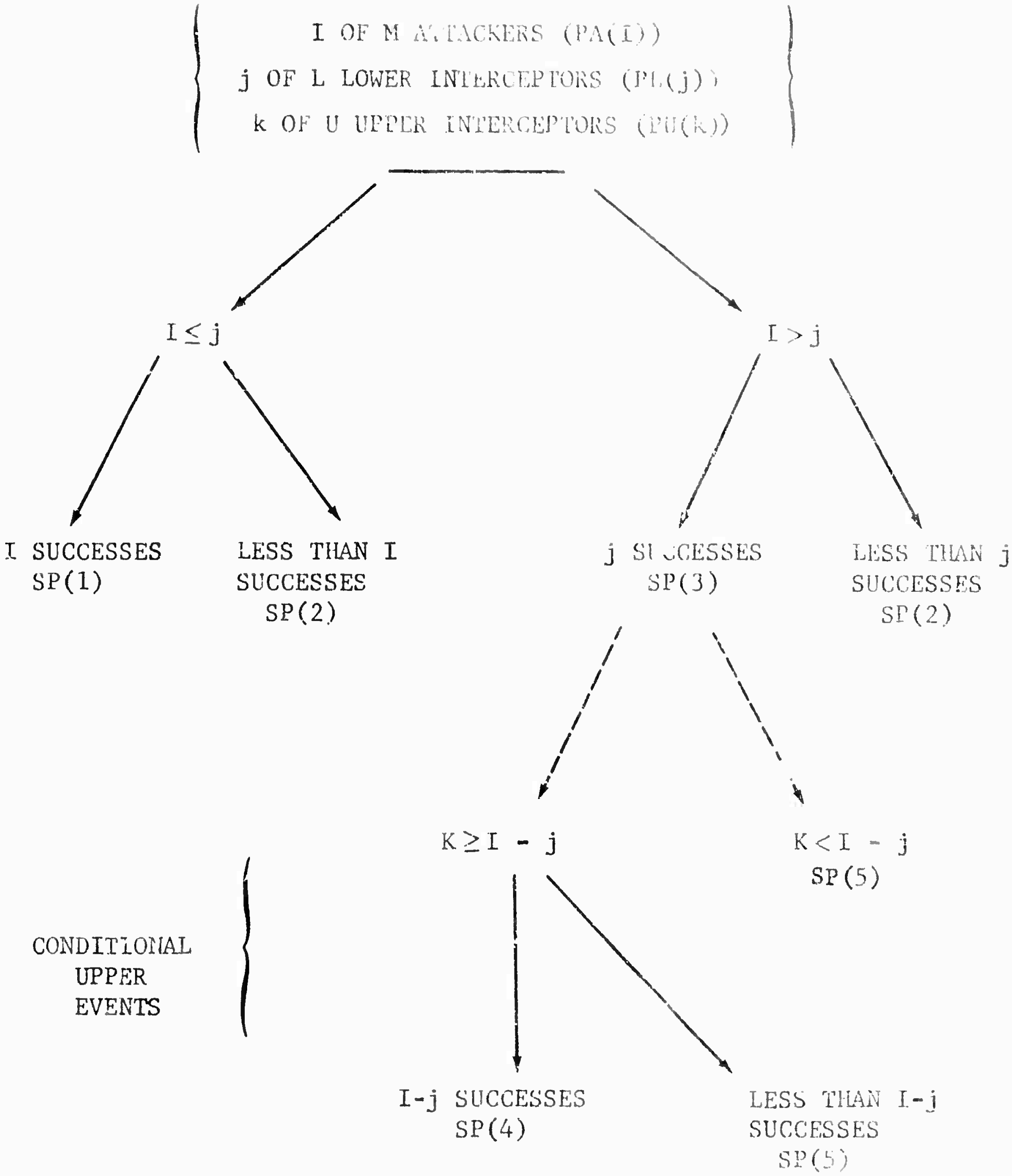


FIGURE S-3: EVENT TREE FOR HARD SITE DEFENSE RADAR ATTACK STATE PROBABILITIES

We can then say the survivability of at least one radar is based on SP(1) for R-1 radars and SP(1) + SP(4) for one radar, or conversely,

$$PKR = SP(2)^{R} + \sum_{i=1}^{R} \binom{R}{i} SP(2)^{R-i} \, SP(4) \qquad \text{(S-21)(a)}$$

$$PRS = 1 - PKR \qquad \text{(b)}$$

where PRS is the probability of radar survival.

If PRS were the only result of interest, this formulation is sufficient. However, we are also concerned with the number of interceptors remaining for the silo defense. The assumptions previously stated are now augmented by further stating an implied assumption in the state probabilities: Namely, available interceptors not used by a killed radar become unavailable until the silo attack. Therefore, to determine the number of remaining available interceptors, the number of such interceptors depends on the state each radar is in after the radar attack.

Thus, we will compute the expected number or available interceptors in each state and combine these numbers according to the probabilities of each outcome.

$$EI(1) = \sum_{i=0}^{L} PA(i) \cdot PK^{i} \sum_{j=1}^{L} PL(j) \left[j - i \right] \cdot 1/SP(1) \qquad \text{(S-22)(a)}$$

$$EI(2) = \sum_{i=1}^{M} PA(i) \sum_{j=0}^{i-1} PK^{j} (1 - PK) \cdot$$

$$\sum_{m=j+1}^{L} PL(m) \left[m - j - 1 \right] \cdot 1/SP(2) \qquad \text{(b)}$$

$$EI(3) = 0. \qquad \text{(c)}$$

$$EI(4) = \sum_{i=1}^{M} PA(i) \sum_{j=0}^{k-1} PL(j) \; PK^{j}$$

$$\sum_{k=i-j}^{U} PU(k) \; PK^{i-j} \left[k-j \right] \cdot 1/SP(4) \qquad \text{(d)}$$

$$EI(5) = \sum_{i=1}^{M} PA(i) \sum_{j=0}^{k-1} PL(j) \; PK^{j}$$

$$\sum_{k=1}^{U} PU(k) \sum_{n=0}^{k-1} PK^{n}(1 - PL) \; (k - n) \cdot 1/SP(5) \qquad \text{(e)}$$

We must remember that if the radars do not survive the number of available interceptors remaining is immaterial. Thus:

$$INT_{S} = \sum_{i=0}^{R-1} \cdot \begin{pmatrix} R \\ i \end{pmatrix} SP(1)^{R - I} \cdot EI(1) \; (R - 1) \cdot (1 - SP(1))$$

$$+ \left\{ \sum_{k=0}^{i-1} \begin{pmatrix} I \\ K \end{pmatrix} \frac{SP(3)^{i-k-1} \cdot SP(2)^{k}}{(1 - SP(1))^{2}} \left[\; k \cdot EI(2) \right. \right.$$

$$+ EI(4) \cdot \frac{SP(4)}{SP(3)} + EI(5) \cdot \frac{SP(5)}{SP(3)} \Bigg]$$

$$+ SP(2)^{i} \left[EI(2) \cdot i + U \cdot DAV \right] \Bigg\}$$

$$+ \sum_{k=0}^{R-1} \binom{R-1}{R} SP(3)^{R-k} \cdot SP(2)^{k}$$

$$\left[k \cdot EI(2) + EI(4) \cdot \frac{SP(4)}{SP(3)} \right] \qquad (S-23)$$

This equation accounts for all variations when at least one radar survives. Table S-1 has been constructed for a case where R = 3 to illustrate the variabilities considered above.

Of these 13 cases based on R = 3 and some attack level M on each radar, only cases 7, 9, 11 and 13 result in complete defense kill. Note that as M increases, SP(1) decreases which decreases the likelihood of the first six cases.

Figure S-4 depicts the variability of the probability that the radars are killed (PKR) and INT_S as a function of the number of attackers allocated against each of these radars. It is interesting to note how smoothly the functions behave, PKR being largely linear for attack levels between 20 and 50, and INT_S having a linear or a linear reciprocal behavior. In an expected sense, there are 22.5 available intercepts per radar in the lower events. Since warhead reliability is also .9, 22.5 of 25 attackers are expected to arrive. Thus, if expected values

TABLE S-1 EXPECTED SURVIVING HARD SITE DEFENSE INTERCEPTORS AS DETERMINED BY FINAL STATE OF THREE RADARS ATTACKED BY M WEAPONS EACH

CASE	NUMBER OF RADARS IN STATE 1	NUMBER OF RADARS IN STATE 2	NUMBER OF RADARS IN STATE 3	ADDITIONAL ABM REMAINING	INITIAL ABM REMAINING	SURVIVE?
1	3	0	0	DAV·U	3 · EI(1)	YES
2	2	1	0	DAV·U	2·EI(1) + EI(2)	YES
3		0	1	EI(4) + EI(5)	2 · EI(1)	YES
4	1	2	0	DAV·U	EI(1) + 2·EI(2)	YES
5		1	1	EI(4) + EI(5)	EI(1) + EI(2)	YES
6		0	2	EI(4) + EI(5)	EI(1)	YES
7	0	3	0	DAV·U	3 · EI(2)	NO
8		2	1	EI(4)	2 · EI(2)	YES
9				EI(5)	2 · EI(2)	NO
10		1	2	EI(4)	EI(2)	YES
11				EI(5)	EI(2)	NO
12		0	3	EI(4)	0	YES
13				EI(5)	0	[illegible]

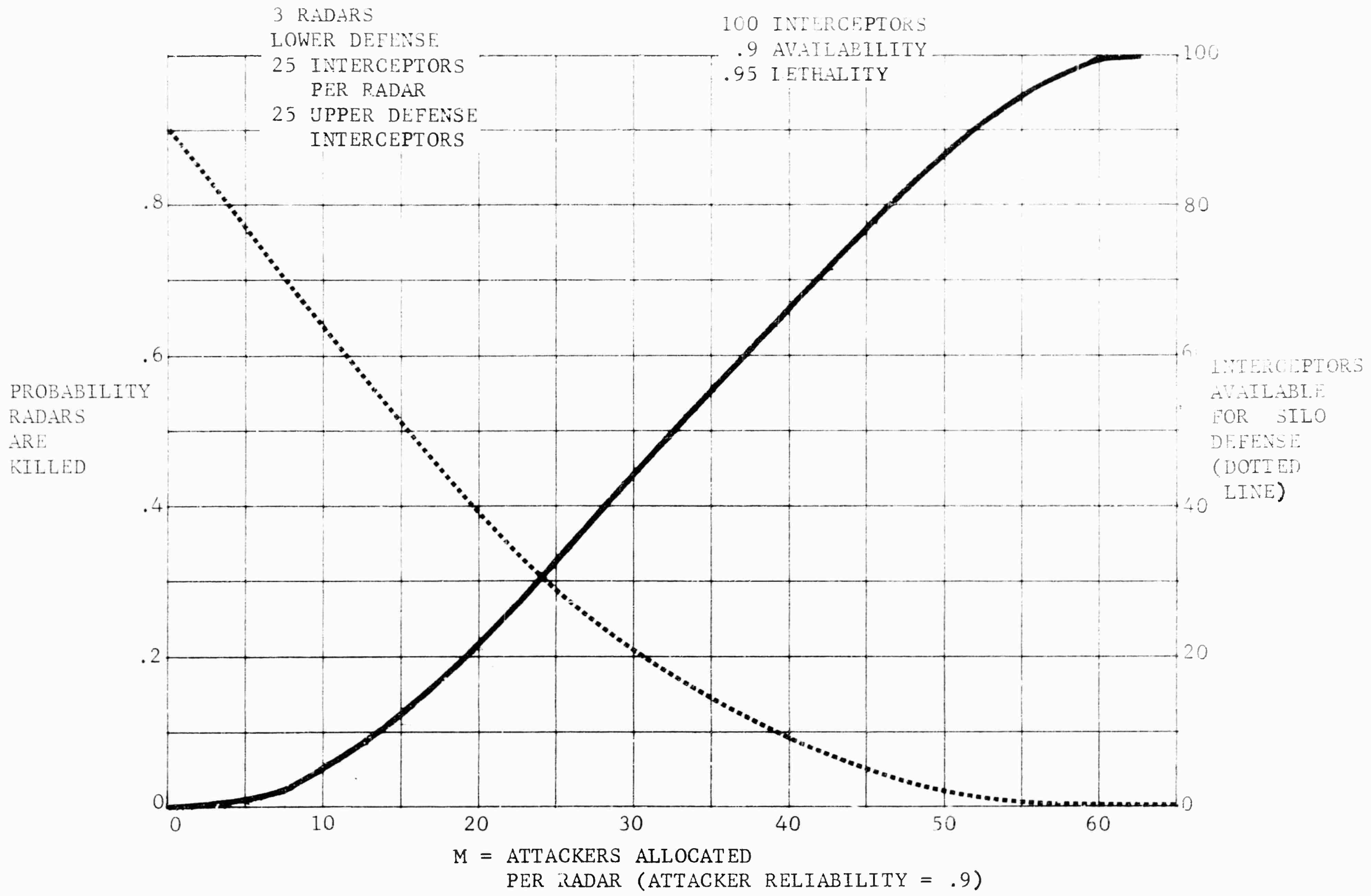


FIGURE S-4 HARD SITE DEFENSE RADAR ATTACK EXAMPLE were directly used, an attack of 51 per radar would guarantee radar kill. By considering the variability of all possible events, an attack of 51 per radar results in only a 90% kill confidence but reduces the number of interceptors remaining for silo defense to one.

It is perhaps due to this behavior that the number of values chosen for M is relatively unimportant as long as the extremes are included, i.e., in general, the radar attack will be very heavy or non-existent. There are some parametric values and silo defense doctrines where this is not true. Therefore, ten M values are considered by the model as candidate radar attack strategies.

We now direct our attention to the silo defense doctrines which are considered as a separate, later attack then the radar attack. Thus, all available interceptors not used in radar defense may be used in silo defense if at least one radar survives.

b. Hard Site Defense - Silo Attack

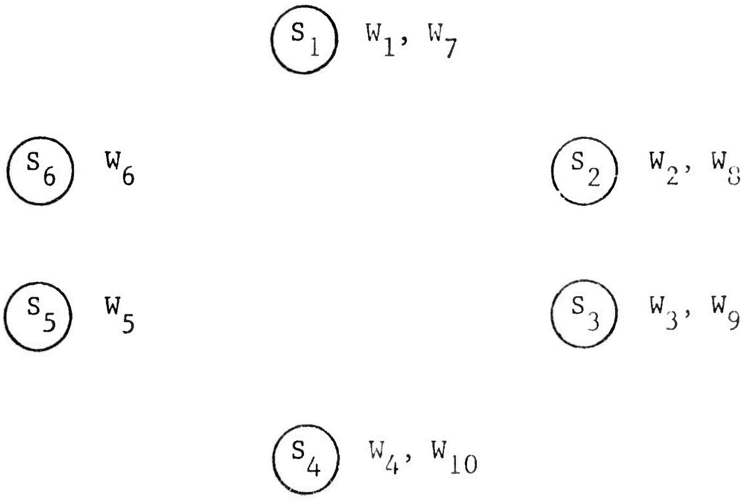
The defense may choose any one of four doctrines to defend the silos. As previously mentioned, these are: subtractive, random, preferential and Prim-Read. Each doctrine is employed in an expected value manner since the errors involved by doing so are felt to be tolerable within the AEM framework. Each doctrine is discussed separately below.

For all the ensuring discussion, MA is the total attack size (radar and silo), R the number of radars, NS the number of silos,

INT_s the available interceptors if the defense is alive, PRK is the probability the radars have been killed, M the number of warheads directed at each radar.

i. Subtractive Doctrine

Attempts will be made against the first INT_s reliable warheads sent against the silos. This doctrine is best understood if sequential arrivals are considered and the first INT_s reliable warheads are "thrown away." This is achievable by cyclic targeting, i.e., the first INT_s/R aim points, or silos, are covered by some later attacker as in the following figure:



S_i is the i^{th} silo in the module to be attacked.

W_j is the j^{th} allocated warhead against the module.

If weapon reliability (r) is perfect and INT_s less than or equal 4, each silo will suffer at least one impact. If weapons one and two were directed against the first silo, three and four against the second, etc., the defense might save the first two silos entirely.

Let N denote the number of silos attacked with L warheads, N1 denote the number of silos attacked with L + 1 warheads. Let D denote the number of silos defended by J interceptors and D1 denote the number defended by J + 1. MS is the total silo attack (MS = MA - M · R), and INT^* the effective number of interceptors.

$$INT^* = MIN \left\{ \left| \frac{INT_s}{r} + .5 \right| , \quad MS \right\} \qquad \text{(S-24) (a)}$$

$$L = \left| \frac{MS}{NS} \right| \qquad \text{(b)}$$

$$N1 = MS - L \cdot NS \qquad \text{(c)}$$

$$N = NS - N1 \qquad \text{(d)}$$

$$J = \left| \frac{INT^*}{NS} \right| \qquad \text{(e)}$$

$$D1 = INT^* - J \cdot NS \qquad \text{(f)}$$

$$S = NS - D \qquad \text{(g)}$$

By the cyclic attack plan assumed, we define QD as the probability of silo survival for the first INT^* attackers and Q for all additional attackers as:

$$QD = 1 = PK \cdot PKSS \qquad \text{(S-25) (a)}$$

$$Q = 1 - PKSS \qquad \text{(b)}$$

Any silo can potentially be in one of four states:

1) Defended by J, attacked by L

2) Defended by J + 1, attacked by L

3) Defended by J, attacked by L + 1

4) Defended by J + 1, attacked by L + 1

We now define K1 through K4 as the number of silos in each state:

$$K1 = MIN \{ N, D \} \quad (S-26)(a)$$

$$K2 = MAX \{ D1 - N1, 0 \} \quad (b)$$

$$K3 = MAX \{ 0, N1 - D1 \} \quad (c)$$

$$K4 = MIN \{ N1, D1 \} \quad (d)$$

The following table clarifies the above equations. Note that for each condition the number of silos sums to NS (NS = D1 + D = N1 + N).

Condition	Number By State K1	K2	K3	K4
N1 = D1; N = D	D = N	0	0	N1 = D1
N1 > D1; N < D	N	0	N1 - D1	D1
N1 < D1; N > D	D	D1 - N1	0	N1

We may now state the expected silos destroyed as:

$$\begin{aligned} SK = {} & K1 \, (1 - QD^{J} \, Q^{L - J}) \; + \\ & K2 \, (1 - QD^{J + 1} \, Q^{L - J - 1}) \; + \\ & K3 \, (1 - QD^{J} \, Q^{L + 1 - J}) \; + \\ & K4 \, (1 - QD^{J + 1} \, Q^{L - J}) \end{aligned} \quad (S-27)$$

ii. Random Doctrine

The random doctrine assumes simultaneous arrival for the silo attack with no saturation limits. The impact of this defense is a reliability degradation for the silo attacking weapons. There is no restriction on the number of interceptors assigned to each reliable warhead. Therefore, all INT_S defenders will be used regardless of the number of weapons in the silo attack.

Let N denote the number of silos attacked with L warheads and N1 denote the number of silos attacked with L + 1 warheads. The expected number of arriving warheads, E(A), in the silo attack is then:

$$E(A) = MS \cdot r \qquad \text{(S-28)(a)}$$

where

$$MS = MA - R \cdot M \qquad \text{(b)}$$

We may now define an effective reliability modifier for allocated warheads as:

$$r^* = (1 - PK)\ INT_S / E(A) \qquad \text{(S-29)}$$

if integer effects are not considered. However, since an interceptor cannot attack a fraction of a warhead, the following effective reliability is used:

$$r^{*} = \frac{W\,(1 - PK)^{J} + W1\,(1 - PK)^{J+1}}{E(A)} \qquad \text{(S-30)}\ \text{(a)}$$

where

$$E(A) = W + W1 \qquad \text{(b)}$$

$$J = \left| \frac{INT_s}{E(A)} \right| \qquad \text{(c)}$$

$$W = INT_s - J * E(A) \qquad \text{(d)}$$

which applies to each allocated warhead. The expected silos destroyed is then:

$$SK = N\,(1 - (1 - r^{*} \cdot PKSS)^{L}) + N1\,(1 - (1 - r^{*} \cdot PKSS)^{L+1}) \qquad \text{(S-31)}$$

It should be noted that this equation is used for evaluating silo destruction if the defense is killed by settling $r^{*} = 1$.

iii. Preferential Doctrine

This doctrine limits the defense to one interceptor per reliable warhead, but assumes the defense can predict which silo each warhead is attacking and how many warheads will arrive at the silo. A practical way to view this scheme is simultaneous arrival with impact prediction. The defense then selects a portion of the silos to defend which have the fewest arrivals to the point of interceptor exhaustion.

For any attack MS there are N silos attacked by L warheads and N1 = MS - N silos attacked by L + 1 warheads. We may then define RF = r · L and RF1 = r · (L + 1) as the expected number of warheads arriving at these silos.

All warheads attacked suffer a degradation of 1 - PK. We now take the offensive point of view and suggest that the L class of strategies be subject to defense until defense exhaustion or until the L + 1 class of strategies are defended against. If r · MS is greater than INT_s, all INT_s interceptors will be used. Therefore, one strategy may be only partially defended against. We define ND as the number of L class strategies totally defended, NDP as the number of L class strategies partially defended (= 0 or 1), ND1 as the number of L + 1 class strategies totally defended, ND1P as the number of L + 1 class strategies partially defended (= 0 or 1). Note that if NDP = 1, ND1 and ND1P both equal zero.

$$\begin{aligned} SK = {} & ND \cdot (1 - Q)^{L} + NDP \cdot (1 - Q)^{ID} (1 - PKSS)^{L - ID} \\ & + ND1 \cdot (1 - Q)^{L + 1} + ND1P \cdot (1 - Q)^{ID} \cdot (1 - PKSS)^{L + 1 - ID} \\ & + (N - ND - ND1) (1 - PKSS)^{L} \\ & + (N1 - ND1 - ND1P) (1 - PKSS)^{L + 1} \end{aligned} \tag{S-32}$$

where

$$Q = (1 - PK) \cdot PKSS$$

ID = number of warheads attacked in the fractional strategy

Note that

$$ND + NDP \leq N$$

$$ND1 + ND1P \leq N1$$

If $NDP = 1; \quad ND1, ND1P = 0$

$$INT_s = ND \cdot r \cdot L + NDP \cdot ID$$

$$+ ND1 \cdot (L + 1) \cdot r + ND1P \cdot ID$$

iv. Prim-Read Doctrine

The purpose of Prim-Read is to approximate the results of a simultaneous arrival random doctrine where the attack is sequential. It is a more clever use of interceptors than the subtractive doctrine where the first INT_s reliable warheads are attacked. Consider Figure S-5 which depicts the module kill functions for the subtractive and random doctrines.

At some point (x in Figure S-5) for a stated INT_s, the offense receives a maximum kill per allocated warhead. It is assumed that some firing doctrines can be found such that the expectations of a sequential attack cannot exceed this pay-off per warhead. Such a doctrines might be two on each of the first x_1 arriving warheads, one on each of the next x_2 and on every other of the next x_3, etc. Again, this firing doctrine is not computed in this model but is assumed to exist. The dotted line in the above figure illustrates the expected module PK if this doctrine is used. Thus, the module PK is greater than subtractive for low attack and greater than or equal to the random doctrine everywhere.

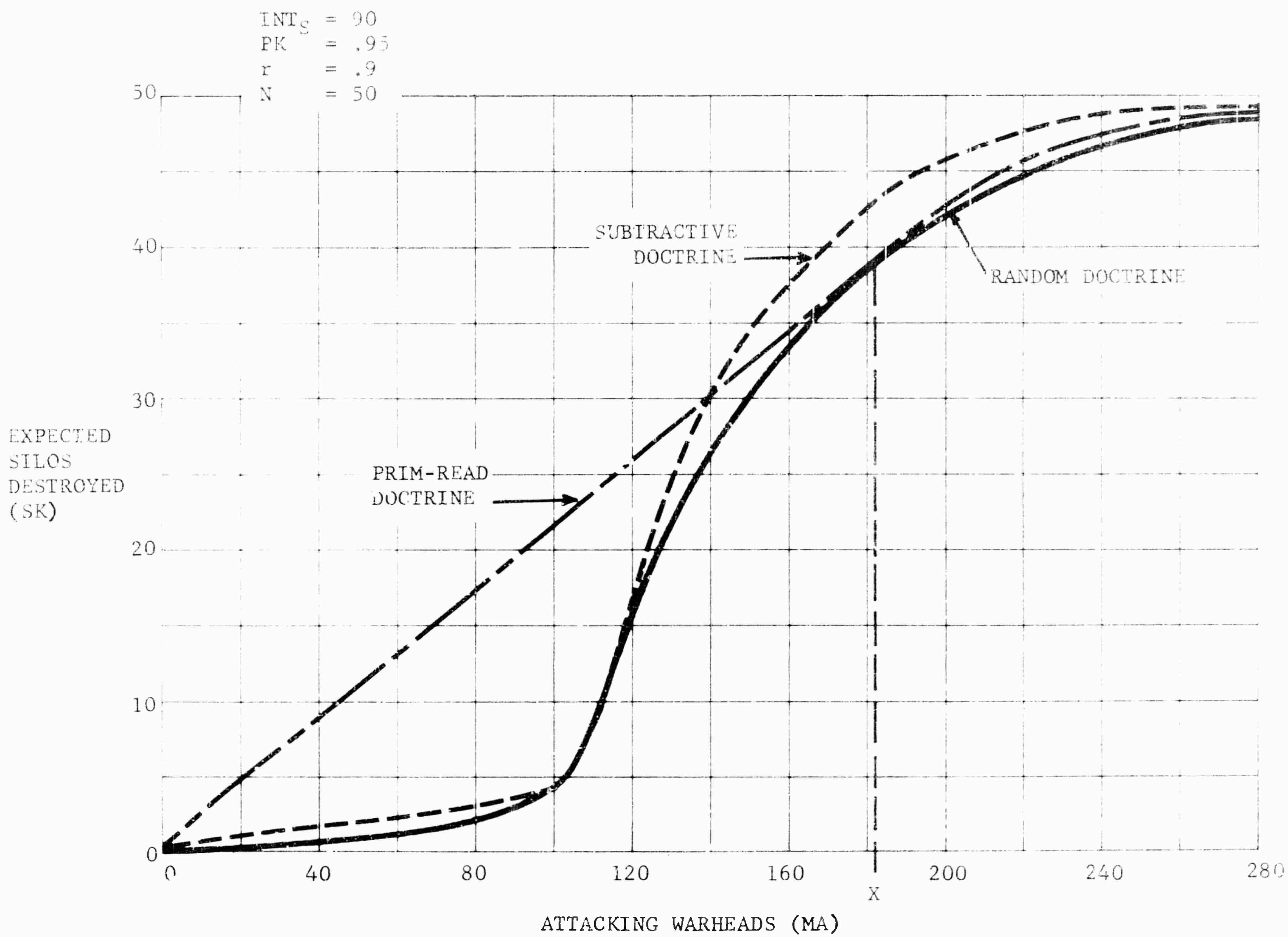


FIGURE S-5 PRIM-READ COMPUTATION COMPARED TO RANDOM AND SUBTRACTIVE DOCTRINES

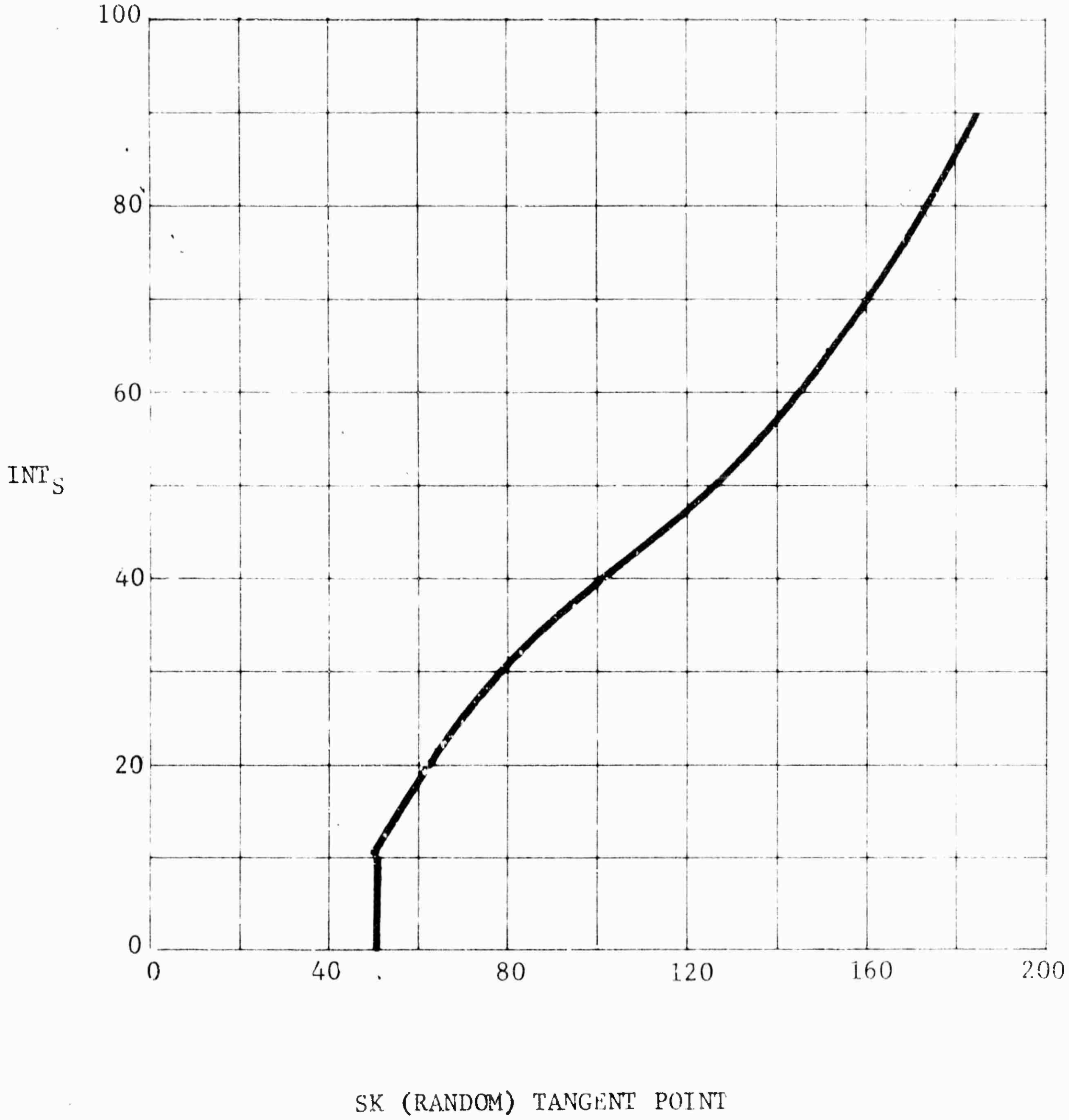


FIGURE S-6 VARIATION OF X WITH INT_S FOR PRIM-READ DOCTRINE

The procedure used in this model is as follows. For each value INT_s resulting from a radar attack strategy, find the attack level x such that the average value per attacking warhead (v) is maximized if random defenses are assumed, i.e.,

Find x such that

$$\max \left\{ v = \frac{SK\ (random)}{X} \right\} \quad \text{is found.}$$

We then assume for any silo attack MS < X, each warhead obtains J units of damage. If the attack is greater than x we assume a weighted average of SK (random) and SK (subtractive). It should be noticed that the precise damage past x is relatively unimportant since by definition we bound the function between subtractive and random doctrines. The point x occurs after the defense is exhausted unless the probability of intercept (PK) is very small. As PK decreases, the difference between SK (random) and SK (subtractive) also decreases. It is also true that while the effect of a decreasing firing doctrine on attacks greater than x is not computable without precise definition of the firing sequence, the greatest defense employment is during early arrivals. Therefore, at some point Prim-Read will be no better than subtractive. The following equations are used:

$$SK = v \cdot MS \qquad MS \leq x \qquad \text{(S-3[illegible])}\ \text{(a)}$$

$$SK = \frac{SK\ (random) \cdot X + SK\ (subtractive)\ (MS - X)}{MS} \qquad MS > x \qquad \text{(b)}$$

4. Kill Functions For A Defended Weapon Module

Both the Safeguard and HSD models produce the maximum SK values for a stated attack level. However, the desired attack level is not known prior to the allocation which is generated by Lagrangian analysis of the entire weapon and target system. It would be possible to store a spaced collection of SK, MA pairs for each candidate weapon type against each module type which would be considered as special linear programming strategies. This approach might be too restrictive unless the SK, MA pairs selected were carefully chosen (and occasionally updated with finer cuts if a desirable region is found).

It was felt that if a functional relationship were found between SK and MA of sufficient quality to generate a desirable MA, the Lagrangian process could be directly applied. Therefore, only the function and pertinent parameters would be stored in AEM. This would greatly reduce storage requirements and possibly save computation time.

In every case, module defense kill is an "S" shaped function of the attack level. However, as is typical of such functions, if any attack is to occur on the target, the attack will be at least as large as that attack yielding the maximum return per allocated warhead. For this reason, the technique described in Appendix A for finding a translated "P" parameter fit to such data was attempted. This formulation was even more desirable since the Lagrangian process in AEM is built on this concept.

Several figures have been drawn to illustrate the goodness of fit between the functions and the P and T parameter fit which best describes the functions. All figures assume a 50 silo module whose defense has three radars and 100 interceptors attacked by a weapon having PKSS of [illegible] against a silo and a .9 in-flight reliability. In all cases, the interceptor lethality is .95.

Figure S-7 depicts a Safeguard module which must further be described by noting the attacking weapon has no decoys and that if only one radar survives, the defense is assumed to be killed.

Figures S-8 through S-11 depict the four Hard Site Defense doctrines. This defense is further described by each radar having 25 interceptors in the lower events with 25 interceptors in a common pool for upper events. Figure S-12 compares the four Hard Site Defense doctrine for this case.

Of particular interest is the attack split into silo and radar attack for a given attack level. Table S-2 shows this variation for Safeguard attack levels. Tables S-3 through S-6 show this information for the four Hard Site Defense doctrines.

5. <u>Computational Considerations</u>

The Safeguard model causes no noticeable i crease in computation time. However, the fitted P and T are generally pessimistic at the attack level generated by the Lagrangian process. Therefore, after each attack where a strategy was used against a Safeguard module, the correct survivors are computed before the retaliation plan is generated. While this process is not ideal, the impact of the evaluation is in the right direction. This is true since the true survivors are less than anticipated and the lower the number of survivors the higher the lambda, and hence, force values. Therefore the next

TABLE S-2: SAFEGUARD MODULE ATTACK

TOTAL ATTACK	ANTI-SILO	ANTI-RADAR	EXPECTED SILOS DESTROYED
25	13	12	3.0
50	26	24	11.3
75	45	30	23.0
100	52	48	32.7
125	83	42	38.9
150	102	48	44.7
175	130	45	46.7
200	152	48	48.5

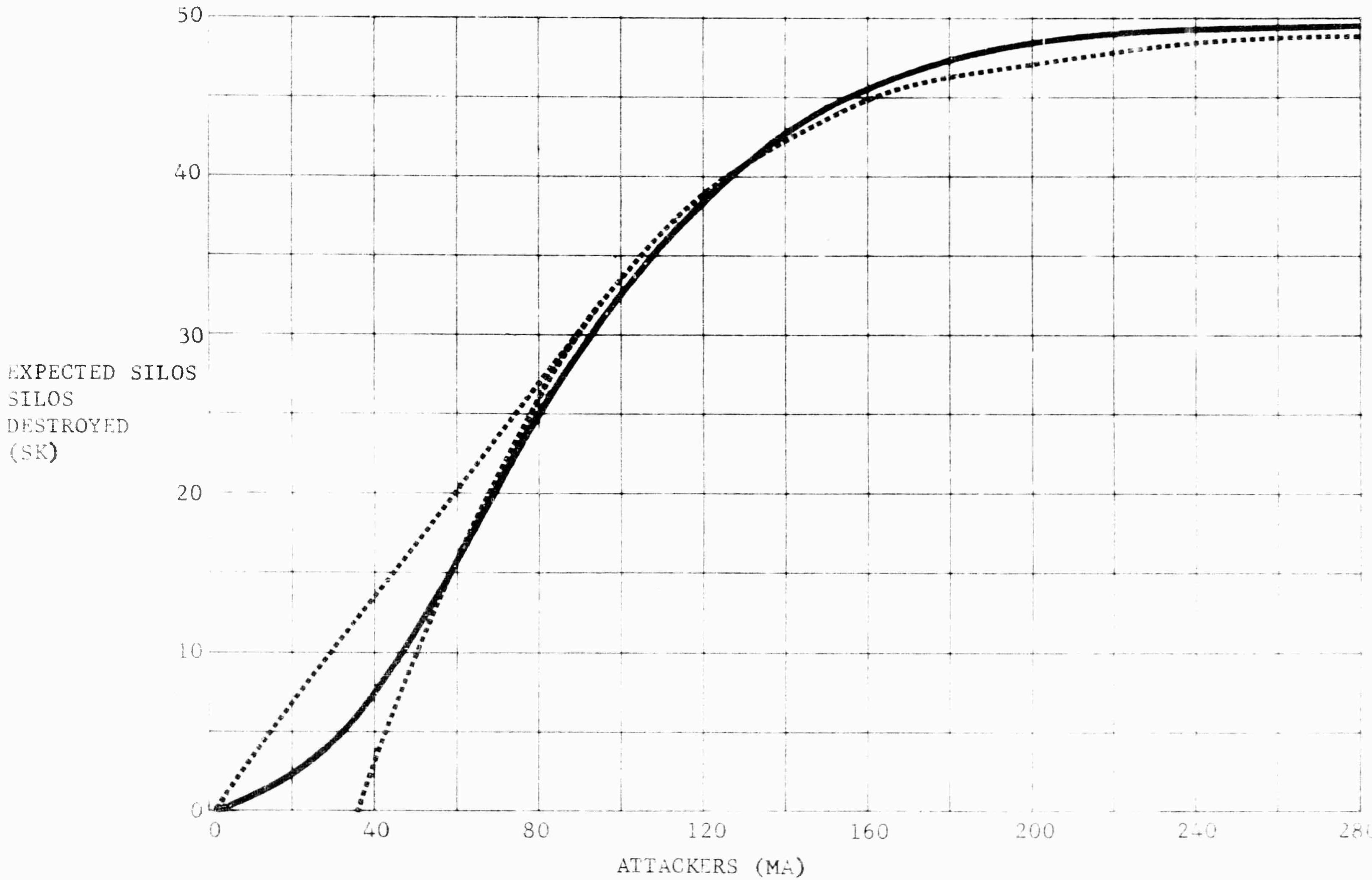


FIGURE S-7 SAFEGUARD EXAMPLE WITH TWO PARAMETER FIT

TABLE S-3: HARD SITE DEFENSE MODULE
ATTACK - SUBTRACTIVE DOCTRINE

TOTAL ATTACK	ANTI-SILO	ANTI-RADAR	EXPECTED SILOS DESTROYED
35	35	0	1.3
70	40	30	2.8
105	45	60	8.9
140	140	0	30.3
175	115	60	42.0
210	150	60	47.0
245	245	0	48.7

Note that by use of integer assignments of warheads to silos and interceptors to warheads, the optimal split of anti-silo and anti-radar is not smooth. If sixty attackers were using anti-radar at the attack of 140, the expected silo destruction would be almost identical to that obtained with no radar attack.

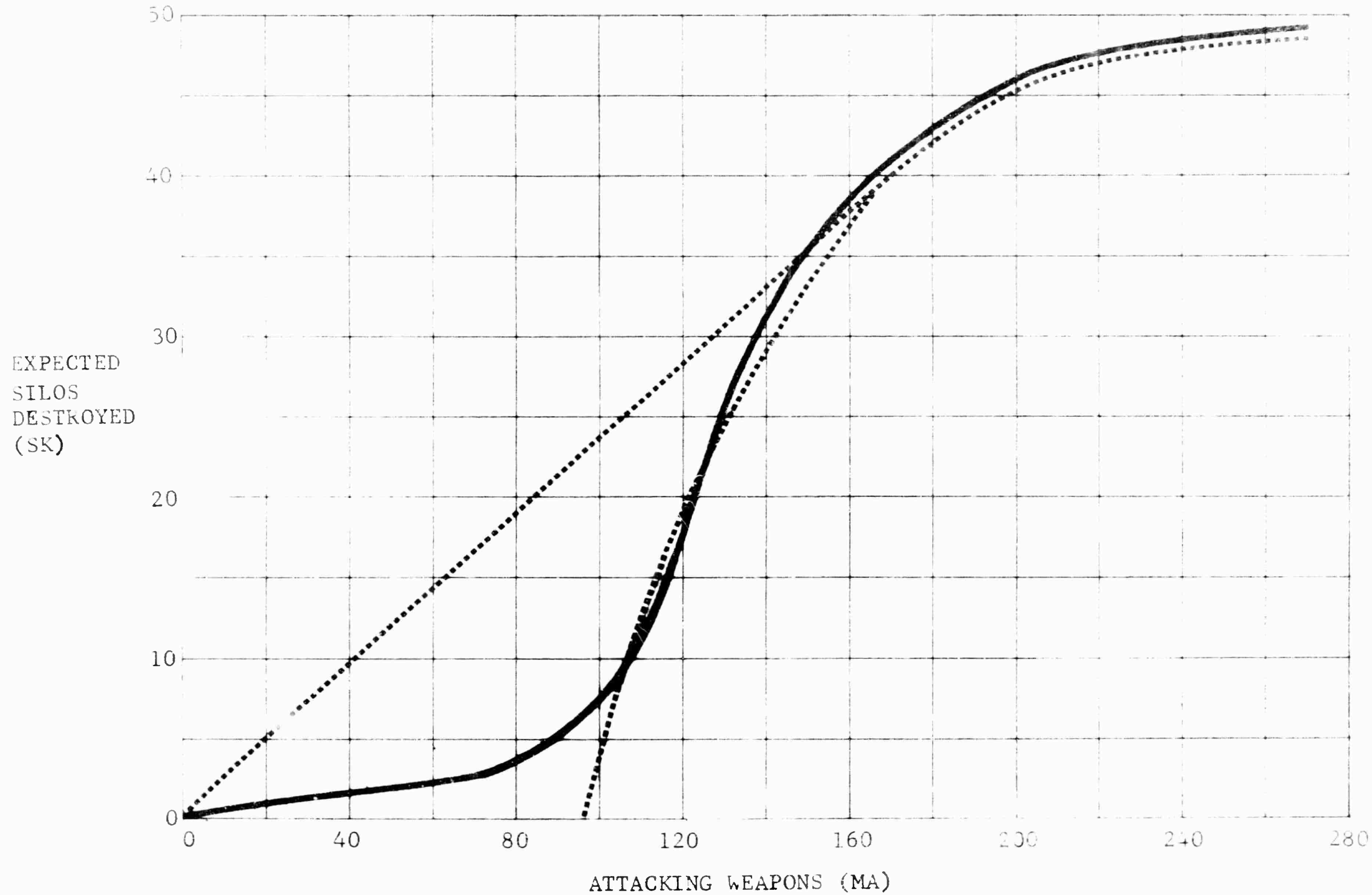


FIGURE S-8: HARD SITE DEFENSE EXAMPLE OF SUBTRACTIVE DOCTRINE AND TWO PARAMETER FIT

TABLE S-4: HARD SITE DEFENSE MODULE
ATTACK - RANDOM DOCTRINE

TOTAL ATTACK	ANTI-SILO	ANTI-RADAR	EXPECTED SILOS DESTROYED
35	5	30	.2
70	40	30	1.8
105	45	60	8.3
140	80	60	27.7
175	115	60	34.3
210	150	60	45.4
245	155	90	48.1

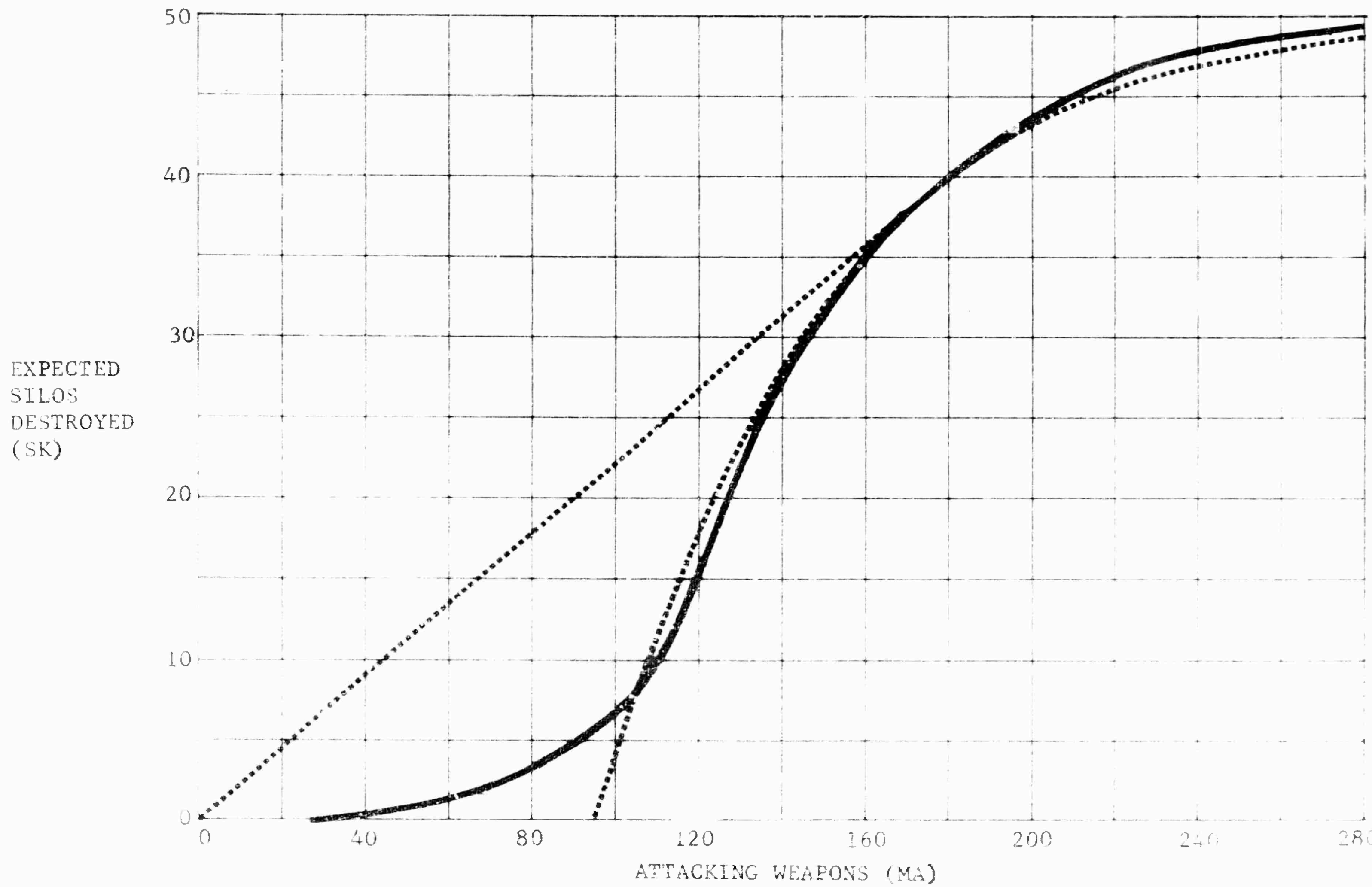


FIGURE S-9: HARD SITE DEFENSE EXAMPLE OF RANDOM DOCTRINE AND TWO PARAMETER FIT

TABLE S-5: HARD SITE DEFENSE MODULE ATTACK - PREFERENTIAL DOCTRINE

TOTAL ATTACK	ANTI-SILO	ANTI-RADAR	EXPECTED SILOS DESTROYED
35	35	0	1.3
70	40	30	2.8
105	45	60	8.9
140	50	90	2[illegible].5
175	85	90	35.7
210	120	90	42.0
245	125	120	46.0
280	160	120	48.1

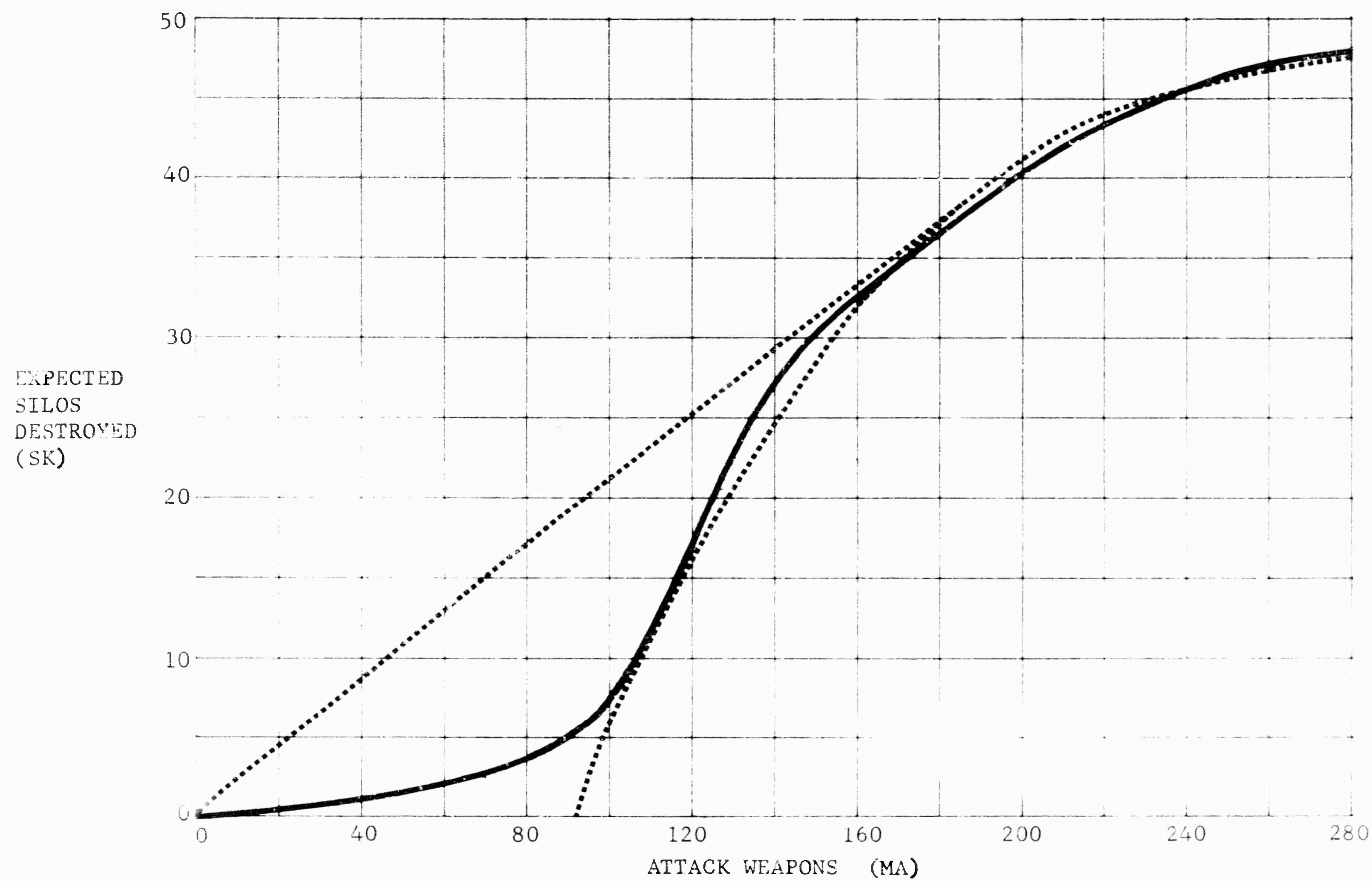


FIGURE S-1C: HARD SITE DEFENSE EXAMPLE OF PREFERENTIAL DOCTRINE AND TWO PARAMETER FIT

TABLE S-6: HARD SITE DEFENSE MODULE ATTACK - PRIM-READ DOCTRINE

TOTAL ATTACK	ANTI-SILO	ANTI-RADAR	EXPECTED SILOS DESTROYED
35	35	0	7.5
70	70	0	15.0
105	105	0	22.5
140	140	0	30.0
175	115	60	39.6
210	150	60	45.9
245	155	90	48.3

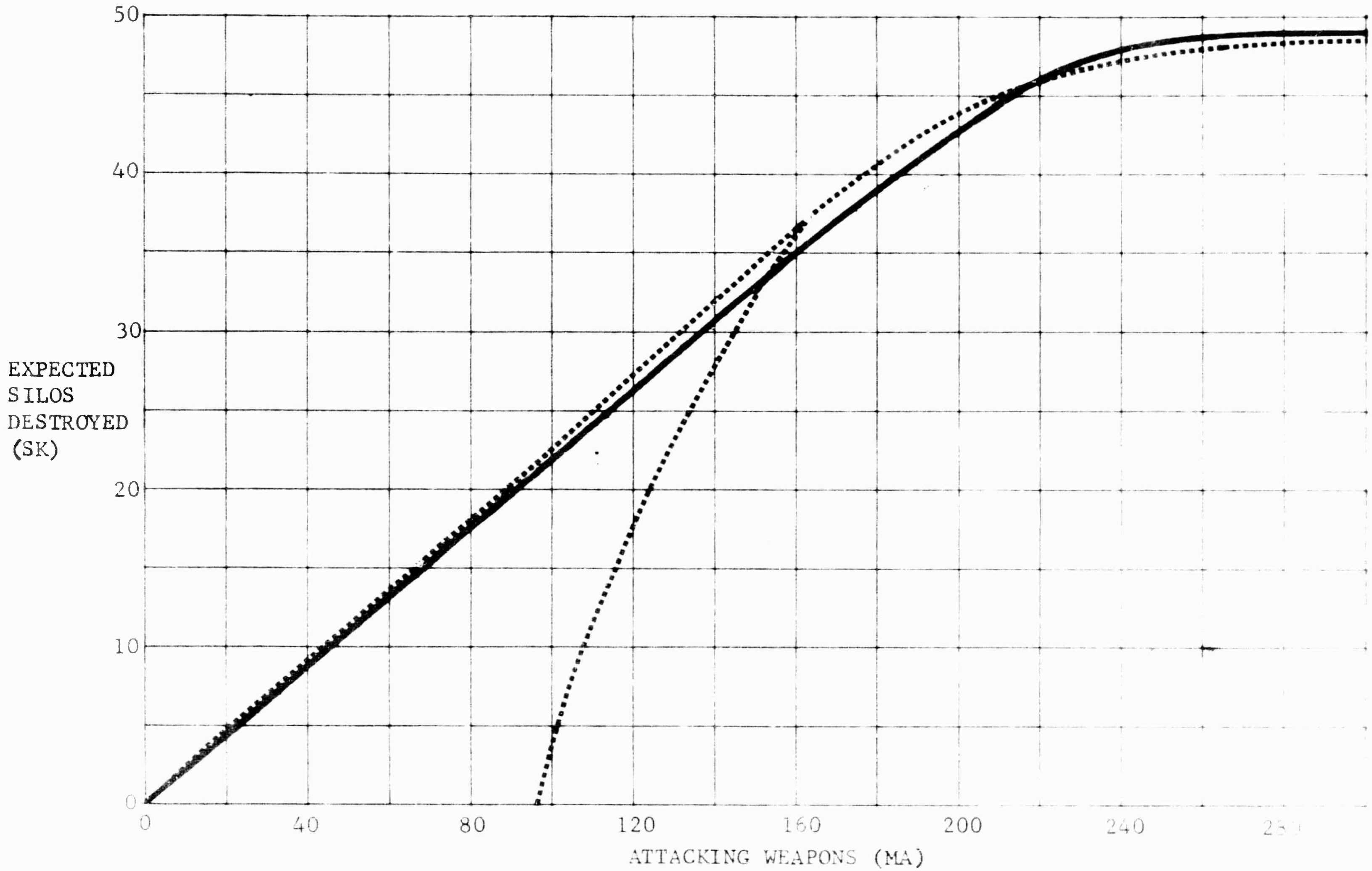


FIGURE S-11: HARD SITE DEFENSE EXAMPLE OF PRIM-READ DOCTRINE AND TWO PARAMETER FIT

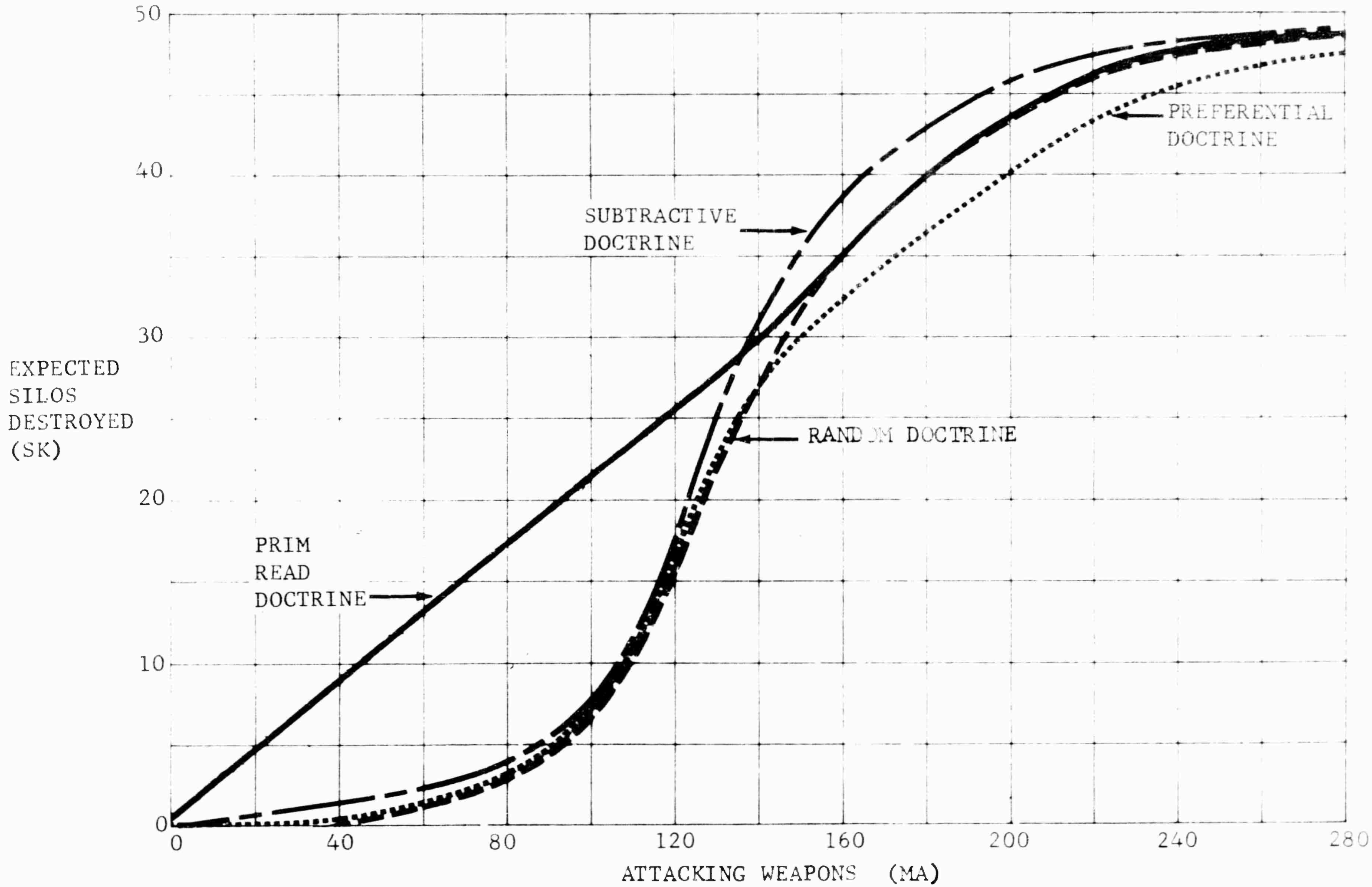


FIGURE S-12 COMPARISON OF HARD SITE DEFENSE DOCTRINES [illegible] against the module should be [illegible] great, if all other factors remain constant.

The Hard Site Defense model is likewise [illegible] the likely attack regions. The procedure used to correct the survivors is identical to the Safeguard discussion above. However, this model does affect computation time. This additional time is almost entirely spent in evaluating various attack levels against the radars. If the number of initial interceptors per radar is small, the total number of events to be considered is small and computation time rapidly decreases. Additionally, very high or very low values for attacker in-flight reliability, interceptor availability, and interceptor lethality also tend to decrease considered events.

6. Summary

Two weapon defense module types are modeled in AEM HEDGE. They correspond to a high altitude endoatmospheric defense called Safeguard, and a low altitude defense called Hard Site Defense. The Safeguard model always employs a shoot-look-shoot firing doctrine against one or two waves of attackers. If there are two waves, the first is directed against the defensive radars. The Hard Site Defense model has four separate firing doctrines for defense of silos. The silo attack may be preceded by a sequential radar attack where a one-on-one firing doctrine is assumed. Each radar is independently defended during the radar attack and if all available interceptors are successfully used, may compete for a pool of additional interceptors reserved for that purpose.

In both cases, several constant interval attack levels are assumed. The radar attack which maximizes the expected silo destruction is found by enumeration. These values are then mathematically fit with a P and T parameter for purposes of strategy generation.

It is not currently possible to simultaneously defend a weapon with both defenses in AEM HEDGE. It is anticipated that this restriction will eventually be removed.

T. BOMBER DEFENSE SUPPRESSION

1. The Defense Suppression Environment

Defense suppression, as it is used here, denotes attack strategies directed at defensive installations which protect desirable targets. There are two possible benefits from assigning defense suppression missions (DSM) to various attackers. If successful, the effect of the defenses is totally removed from target attack strategy considerations. Even if DSM is unsuccessful, some number of interceptors may be used against the DSM. Therefore, unless the defense has the time and capability to reload, the number of interceptors available for target defense is reduced by DSM. An exception to this rule exists if the defense has no capability against the DSM weapons but does against the target kill weapons, e.g., ICBM against SAM or bombers against ABM.

The DSM model incorporated into AEM has a basic assumption regarding the separability of attacks against a target and its defense.

In reality, it is probable that some collateral effects of either DSM or target kill assignments are encountered. This is true if the suppression is by exhaustive overflights of the defense but against some target or if the target is within a radar's lethal radius of the defense (accidental DSM). There is also some probability that a near-by detonation jiggles a wire loose in the defensive network or the radar is covered with dust.

The relative timing of the DSM and target attack may also be important. If the DSM is not totally successful and the defense has both the capability and sufficient time to reload, the weapons used in DSM were essentially wasted. There is also the possibility of DSM by saturation. This technique requires precise arrival timing where a relatively large number of objects are directed at the defense immediately followed by the target attack. The assumption is that the defense will be so busy defending itself that the target attack is essentially unmolested. Other time-sensitive factors could be of interest, e.g., ladder down, blackout, use of debris clouds carried by the ambient wind, etc., where specific concern for the tactics of the DSM attacks are considered.

The mathematical implication of this assumed separability is that the expected value destroyed against a target protected by a defense of INT interceptors, assuming DSM with D attackers of type k and a target attack with N attackers of type i, is given by

$$VD = V \cdot \sum_{j=o}^{INT} P(j \mid D_k) \cdot PK(N_i \mid j) \qquad \text{(T-1)}$$

where

V = The target value

$P(j \mid D_k)$ = The probability that j operational defenders survive, given a DSM allocation of D_k attackers.

$PK(N_i \mid j)$ = The cumulative probability of damage of N_i attackers against the target, given j operational survivors of the DSM attack.

Equation T-1 illuminates the crucial factors involved in DSM. The generation of strategies against defense installations, the likelihood of successfully suppressing the defense, the degradations of a defense which has been attacked but survives, and the impact of a surviving defense on the target strategy generation process.

The complete solution of (T-1) is beyond the current capability of AEM. In fact, prior to the inclusion of a DSM capability in the program, research was conducted to determine what kind of a DSM model could reasonably be incorporated. (See reference 33). The conclusion of this analysis was that a high confidence DSM capability was possible, and it is the option which has been implemented for the case of bomber defenses which protect a single target.

The restriction of DSM strategies to those which have a high confidence of succeeding, while mathematically restrictive, is sound military doctrine. This is the same assumption which has been made in the subtractive island defense models. Under this assumption, the defense may be viewed as a separate target which is independent of the target it protects. Hence separate strategies may be generated against each. The linking condition is the requirement that if the true target is attacked, then a DSM allocation of sufficient size to have a high confidence ($\geq$ .99) of suppressing the defense must also be generated. (The two allocations may be done with different weapon types, however.)

With only high confidence DSM strategies permitted, equation (T-1) simplifies to

$$VD = V \cdot P\ (o \mid D_k) \cdot PK\ (N_i \mid o) \cong V \cdot PK\ (N_i) \qquad (T\text{-}2)$$

where $PK(N_i)$ is the normal AEM damage function. If (T-2) applies, then generating a strategy for weapon type i also requires determining that weapon type k which maximizes the Lagrangian function

$$H = V \cdot PK\ (N_i) - N_i\,\lambda_i - D_k\,\lambda_k \qquad (T\text{-}3)$$

Subject to the condition that D_k is of sufficient magnitude that

$$P(O \mid D_k) \cong 1. \qquad (T\text{-}4)$$

Since 0 operational survivors can occur either because there are no remaining interceptors or because the defense installation/radars no longer survive, (T-4) includes the condition that the DSM weapons have destroyed the defense prior to the launching of all interceptors (as will be shown later, because of the attack geometry for realistic systems, this is a very likely condition).

2. The Bomber Defense Barrier

The high confidence DSM model in the current version of AEM HEDGE treats the case of SAM bomber defenses which protect a single target. Since high confidence DSM is required, these bomber defenses act as a defense barrier which must be eliminated before bomber-delivered weapons can be allocated to the target.

The bomber defense barrier is pictured in Figure T-1. Up to three distinct SAM types (SAM1, SAM2, and SAM3) are modeled, which are assumed deployed in independent sites protecting the target. Any number of sites of each type is permitted. [illegible] be allocated DSM weapons separately, with no collateral damage to adjacent sites assumed. Similarly, no netting between sites is assumed, so that each site engages only the attackers directed to that site. Figure T-1 also shows the relationship of the SAM1/SAM2/SAM3 defense barrier to the other AEM bomber defense options.

It would have been possible to develop a purely probabilistic DSM model. However, analysis of the disparity of attacker/defender characteristics for systems of interest showed that this would have significantly distorted the true situation. This is because of the variance in defense capability caused by such variables as attacker speed and altitude, and defense reaction time, interceptor speed and flyout range. When such considerations are included, a very large stockpile of interceptors may be useless because of the attacker speed and slow defense reaction time. Consequently an attack geometry model has been included for the case of bomber delivered weapons. The purpose of this model is to determine the maximum number of encounter arenas which could actually occur between interceptors and attackers. Probabilistic considerations then determine the outcome of each single encounter arena, and the totality of such arenas. This allows, for any stated number of attackers, the computation of the probability that the defense is killed. It has been assumed that a single penetrator will render the defense inoperative.

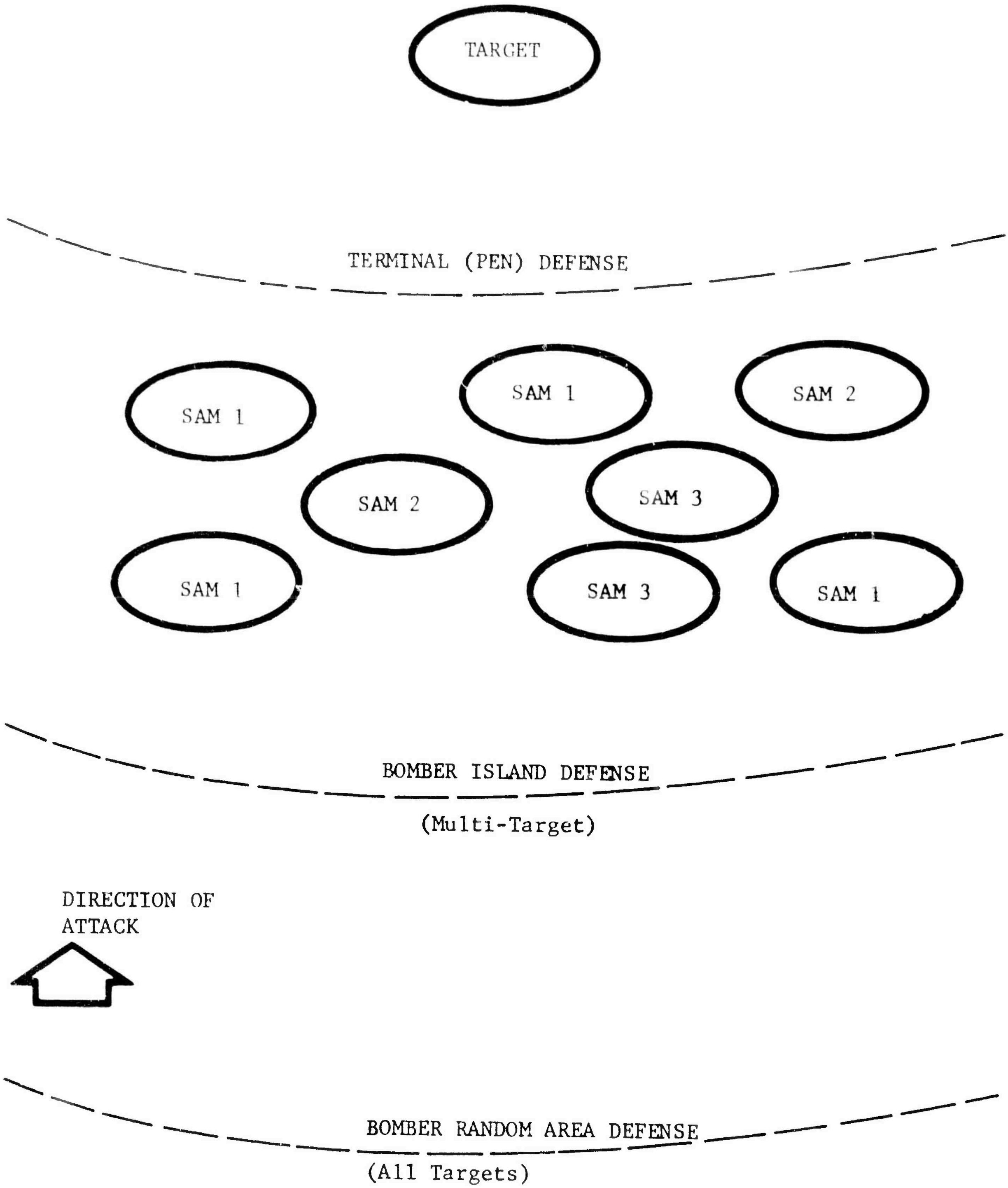


Figure T-1: Bomber Defenses

No attack geometry model has been included for the case of DSM against the [illegible] SAM sites with ICBM's or SLBM's. For SAM systems of interest, the PSSK against these attackers is so small that the only consideration for high confidence suppression is the reliability of the attacker. However, a capability has also been included to allow the user to specify via program input the number of reliable attackers required for suppression. Hence, if SAM systems which have a capability against ICBM's/SLBM's are of interest, they can be included by this method.

The inclusion of an attack geometry model for ICBM's/SLBM's would require detailed trajectory specifications and also greater variability in firing doctrines than is reasonable for bomber-delivered weapons.

The following quantities are used to specify the attack geometry/encounter model. To simplify the notation, a single weapon type and a single SAM type will be assumed unless otherwise specified. In the actual model, the required computations are performed for each weapon type against each SAM type (SAM1/SAM2/SAM3).

N = The number of warheads/RV's allocated for DSM against a single SAM site.

R = The non-reprogrammable reliability of the attacking warhead. (This may include degradations for retargetting and area defense penetration).

H = The altitude of the attacker.

VT = The attacker velocity.

RF = The interceptor maximum flyout range.

VM = The interceptor velocity.

HT = The radar antenna height.

a_m = The radar mask angle.

DIS = The minimum intercept distance.

TD = The initial interceptor launch delay time.

TBS = The interceptor refire time (time between shots).

L = The number of launchers.

I = The number of interceptors which can be loaded and fired from each launcher.

A_s = The reprogrammable reliability of an interceptor.

R_s = The in-flight reliability of an interceptor.

PK = The single shot probability of kill of an interceptor against the specific attacker under consideration.

Using these quantities, the procedu[illegible] for computing the number of warheads required for high-confider[illegible] will now be computed. The crucial issues are:

a. How many salvos could the defense possibly fire?

b. What is the probability that the attacker penetrates a single salvo?

c. What is the probability that the defense survives?

d. How many warheads are required for high confidence DSM?

Each of these areas will now be discussed separately for bomber delivered weapons. The computation for multi-site ... cussed, fo wed by the computation for the case in which the DSM price in reliable RV's is specified. Multi-site suppression will then be discussed, followed by the strategy generation procedure, and the treatment of misestimates.

3. The Attack Geometry Model

For bomber-delivered weapons, it is assumed that the defender has line-of-sight radar, with a specified mask angle a_m, and will launch as soon as possible after detection occurs (subject to initial delay time and flyout range considerations). The attacker is assumed to launch at some pre-specified coordinates which are known to be within range of the SAM site for his weapons. Furthermore, the attacker is assumed to fly at an altitude sufficiently low so that when detection occurs by the SAM site, launch of the DSM weapon can be assumed to have occurred.

A spherical earth is assumed. This results in the geometry of Figure T-2. (Distances on the surface have been exaggerated for clarity).

In determining line-of-sight distances, an equivalent earth radius $R = 4/3\ R_e$ is used to correct for atmospheric refraction at altitudes below 20 kft. (See references 34-35). If the mask

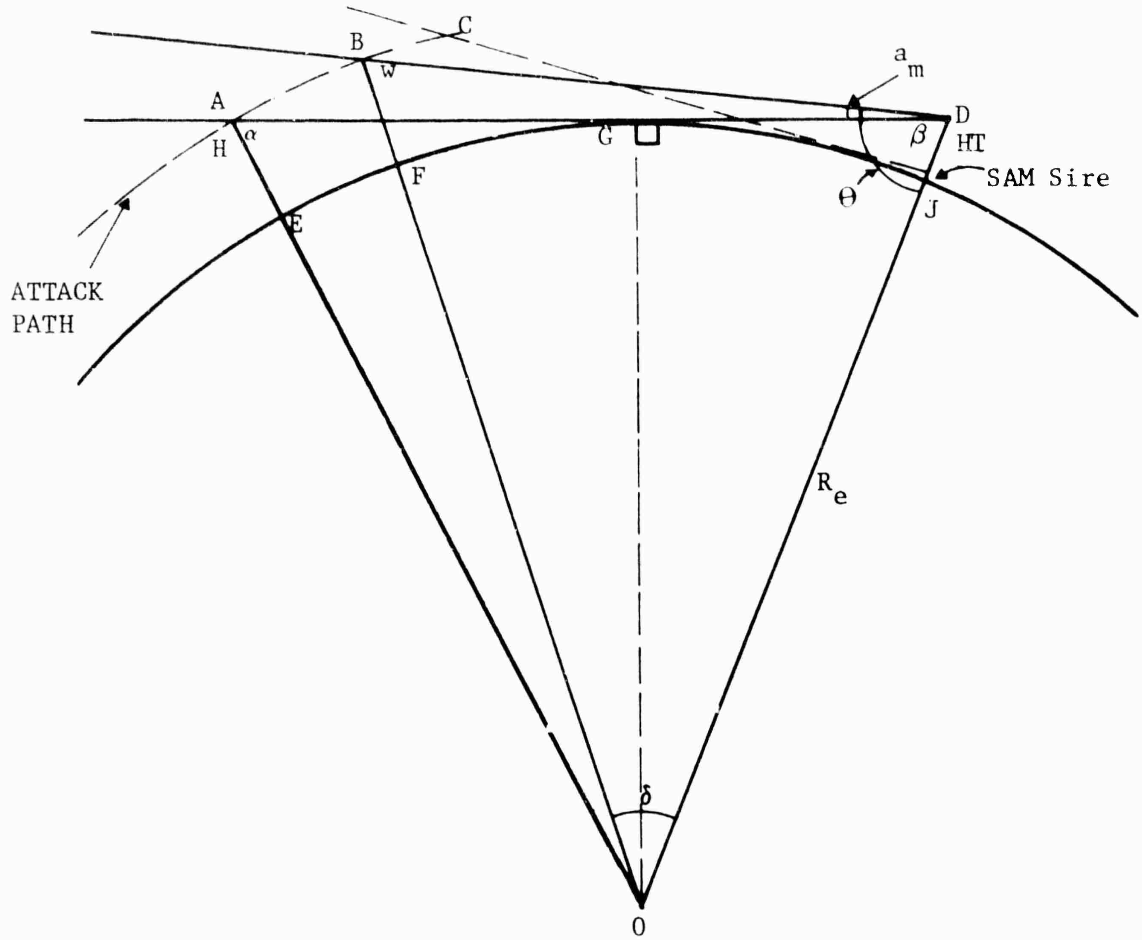


Figure T-2: Attack Geometry

angle were zero. Detection would occur at point A. Hence

$$\beta = \cos^{-1}(R/(R+HT)) \qquad \text{(T-5)}$$

and thus $\theta = \beta + a_m$ corrects for radar masking, during which time the attacker has moved to point B. Using the law of sines,

$$\sin w = (R+HT)\ \sin\theta/(R+H) \qquad \text{(T-6)}$$

and hence w can be calculated from the relationship

$$w = \tan^{-1}\left(\frac{\sin w}{\cos w}\right) \qquad \text{(T-7)}$$

The separation angle is thus given by

$$\delta = \pi - (\theta + w) \qquad \text{(T-8)}$$

and the separation distance is computed as

$$\overline{JF} = \delta \cdot R_e \qquad \text{(T-9)}$$

This distance assumes an antenna of height HT. A line-of-sight distance from the attacker to the site itself (base of the antenna) strictly is not possible until the attacker moves to point C. However, if HT is small, distance $\overline{BC}$ is negligible. Hence it will be assumed that the potential engagement begins at point B, and along the slant range $\overline{BJ}$. (This neglects possible curvilinear trajectories of both the attacker and defender). Consequently the engagement occurs over a slant range distance

$$SR = (\overline{JF}^2 + H^2)^{\frac{1}{2}} \qquad \text{(T-10)}$$

The problem now is to determine how many engagements occur along the distance SR. Assume a single attacker and an unlimited number of perfectly reliable interceptors, all fired from a single launcher. The attacker is traveling with velocity VT, and the interceptor with velocity VM. However, the interceptor has a limited range RF, and must intercept at a distance of at least DIS from the SAM site. Furthermore the first interceptor cannot be fired until a time TD has elapsed, and succeeding interceptors are fired with separation times of TBS. It is assumed that TBS is of sufficient magnitude that the interceptor launches can be treated as a sequence of discrete events, of sufficient spacing so that the engagement of the i-th interceptor with the attacker is independent of the (i+1)st interceptor engagement. (This assumption is realistic for actual systems considered.)

It is also assumed that if SR $\leq$ RF, the defender will fire as soon as possible. However, if SR > RF, the defender will wait, if possible, and time the interceptor launch so that the first intercept occurs at the maximum intercept distance RF. Consequently define

DØ = The distance the attacker travels during the initial delay period (=TD·VT).

DF = The distance the attacker travels during the time it takes the interceptor to reach RF. (=VT·(RF/VM))

NOB = The number of attacker-interceptor engagements (fly-by's) which occur along the distance SR.

The first condition which can occur is that DØ $\geq$ SR. This means that no launches can occur. Hence NOB = 0. If DØ < SR, then two cases happen. In either instance, an effective engagement distance RE, different from SR, is defined to compute NOB. RE is the distance between the attacker and the site when the first interceptor is launched.

a. RF $\geq$ SR. The defender fires as soon as possible. Hence

RE = SR - DØ

b. RF < SR. The defender may wait, or shoot.

1. DØ + DF > SR-RF. The defender shoots. Hence RE = SR-DØ.

2. DØ + DF $\leq$ SR-RF. The defender waits until the attacker is close enough. Thus RE = SR+DF.

Given the effective engagement range, the number of possible engagements is computed as

$$NOB = \left[(RE/VT)/TBS + 1 \right]$$

where [ ] denotes the integer part.

Previous computations have assumed a minimum intercept distance of zero. If this is not the case, the computations remain valid by replacing SR by SR-DIS.

Figure T-3 illustrates the relationships between the distances of interest.

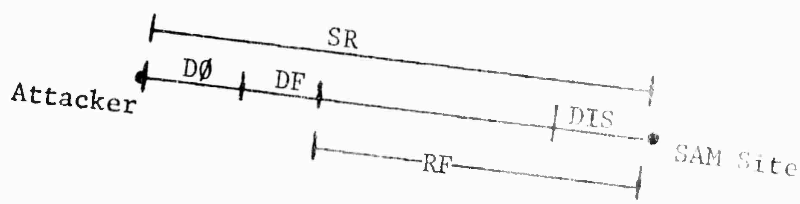


Figure T-3: Attack Geometry Distances

To illustrate the results of parametric variations on the number of engagement arenas, Figure T-4 has been prepared for three sets of delay time/refire time pairs. Standard conditions for other variables were the following:

Interceptor range (RF)	:	75000 ft.
Interceptor velocity (VM)	:	800 ft/sec.
Antenna height (HT)	:	30 ft.
Radar Mask Angle (a_m)	:	2 degrees
Minimum Intercept Distance (DIS)	:	0 ft.

Using these standard conditions and TD=20 secs, TBS=20 seconds, and the same altitude/velocity variations given in Figure T-4, no engagements are possible (NOB=0) at any velocity for altitudes below 600 ft. The maximum number of engagements at higher altitudes is one.

Attacker Velocity (ft/sec)	Attacker Altitude (Ft) 100	200	300	400	500	600	800	1000
600	2	6	9	13	17	20	28	35
800	1	4	7	10	12	15	21	26
1000	1	3	5	8	10	12	16	21
1200	1	3	4	6	8	10	14	17
1400	1	2	4	5	7	8	12	15
1600	0	2	3	5	6	7	10	13
1800	0	2	3	4	5	6	9	10
2000	0	1	2	4	5	6	8	10
2200	0	1	2	3	4	5	7	9
2400	0	1	2	3	4	5	7	8

Figure T-4(a): Number of Engagement Arenas, Delay Time TD = 1 sec, Refire Time = 1 sec.

Attacker Altitude (Ft)

Attacker Velocity (ft/sec)	100	200	300	400	500	600	800	1000
600	0	1	1	2	3	4	5	7
800	0	0	1	2	[illegible]	3	4	5
1000	0	0	1	1	2	2	3	4
1200	0	0	0	1	1	[illegible]	2	3
1400	0	0	0	1	1	1	2	3
1600	0	0	0	1	1	1	2	2
1800	0	0	0	0	1	1	1	2
2000	0	0	0	0	1	1	1	2
2200	0	[illegible]	0	0		1	1	1
2400	0	0	0	0	0	1	1	1

Figure T-4(b): Number of Engagement Arenas, Delay Time TD=5 sec, Refire Time TBS = 5 secs.

Attacker Altitude (Ft)

Attacker Velocity (ft/sec)	100	200	300	400	500	600	800	1000
600	0	0	0	1	1	2	2	3
800	0	0	0	1	1	1	2	2
1000	0	0	0	0	1	1	1	2
1200	0	0	0	0	0	1	1	1
1400	0	0	0	0	0	0	1	1
1600	0	0	0	0	0	0	1	1
1800	0	0	0	0	0	0	0	1
2000	0	0	0	0	0	0	0	1
2200	0	0	0	0	0	0	0	0
2400	0	0	0	0	0	0	0	0

Figure T-4(c): Number of Engagement Arenas, Delay Time TD=10 sec, Refire Time TBS=10 secs.

The previous development has assumed an unlimited number of interceptors fired from a single launcher. This provided NOB, the maximum number of engagement arenas which could occur along the slant range SR. To extend this to the case of multiple launchers, it will be assumed that all such launchers are fired in a salvo, every TBS seconds. This provides an apparent restriction on the firing doctrine. However computation of NOB for realistic parameters shows that if some other firing doctrine were used, perhaps based on observations of the outcomes of previous engagement arenas, that the defense would simply not have sufficient time to react. Additionally, the modeling of a continuous firing doctrine, with its corollary of a continuous encounter along the slant range, is beyond the scope of the present attack geometry model.

If all launchers at a site are fired simultaneously, then the maximum number of encounters possible is the minimum of the NOB determined by the attack geometry model and the number of interceptors which could be fired from a single launcher. Symbolically,

$$\text{NOB} \longrightarrow \min(\text{NOB}, I)$$

Similarly, if all interceptors were perfectly reliable, then the number which engage an attacker in each arena along the slant range is simply the number of launchers L.

Hence the DSM outcome for bomber-delivered weapons has been reduced to determining the results of NOB independent encounters, each of which has L interceptors and the number of attackers which survive the previous encounter. However, since the numbers involved are generally small, the outcome must explicitly consider the uncertainties associated with the interceptor availability, reliability, and PSSK, and the attacker reliability (since he is assumed to allocate sufficient warheads for high confidence DSM). The overall encounter model is pictured in Figure T-5.

4. <u>The Engagement Model</u>

The method used to compute the DSM confidence will be to compute, for each engagement arena, a Markov transition matrix T which gives the distribution of surviving attackers as a function of the number of interceptors in the arena and the maximum number of attackers which could have survived all previous arenas. The following definitions are required.

$\overline{x}(j)$ = An N+1 dimensional vector whose i-th comparent $x_i(j)$ is the probability that exactly i-1 attackers survive the j-th engagement arena.

$\overline{b}(M,P)$ = An M+1 dimensional vector whose i-th component b_i is the i-th term of a binmial distribution of M objects with probability of success P. (Hence b_1 is the probability of zero successes, etc.)

$T(j) = \left[\overline{T}^{o}(j) \vdots \overline{T}^{1}(j) \text{ --- } \overline{T}^{N}(j)\right]$ is the N+1 x N+1 state transition matrix for engagement arena j, where $\overline{T}^{i}(j)$ is an N+1 dimensional vector whose k-th comparent, $T_K^i(j)$, gives the probability of exactly k survivors, given i attackers.

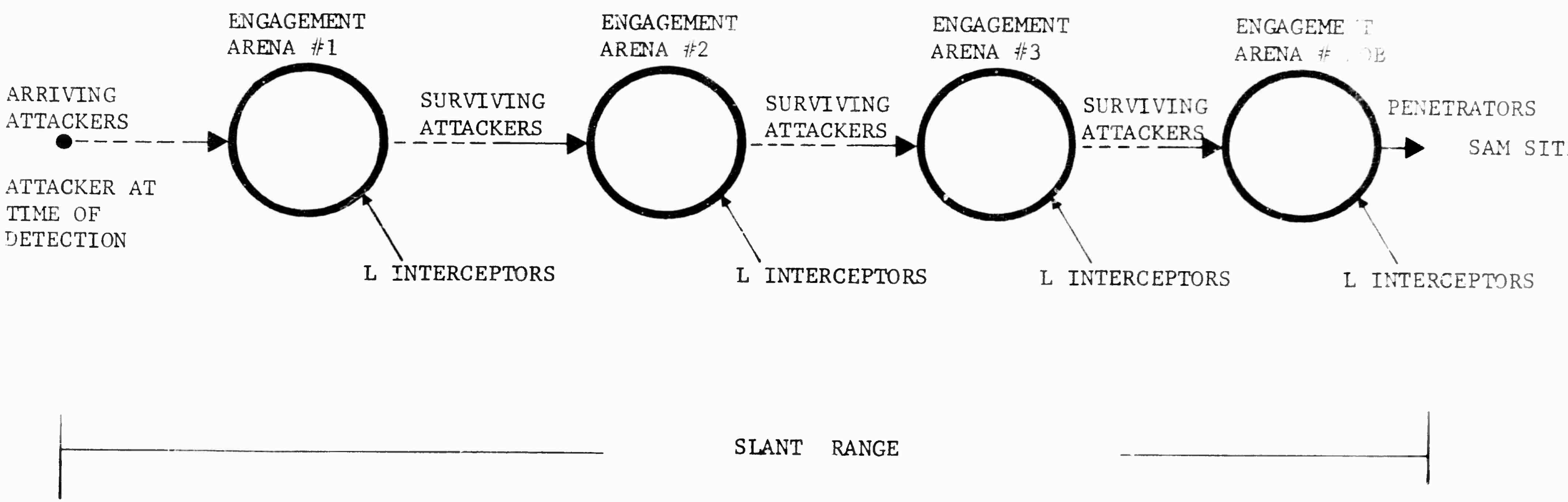


FIGURE T-5. Conceptual Encounter Model

With these definitions, the results of a single engagement arena are obtained by computing

$$\overline{X}(j) = T(j)\,\overline{X}\,(j-1) \tag{T-11}$$

Hence for some initial number of attackers $\overline{X}(0)$, the distribution of surviving attackers, which penetrate to the SAM site itself, is obtained by applying (T-11) recursively NOB times, obtaining $\overline{X}(NOB)$. It is assumed here that the first penetrator kills the defense. Hence a high confidence of defense kill is equivalent to requiring the probability of zero penetrators to be very small. But the probability of zero penetrators is given by $X_1(NOB)$. Hence the high confidence requirement of (T-4) is equivalent to the condition

$$X_1(NOB) \leq .01 \cong 0. \tag{T-12}$$

This leads to a mathematical programming problem which must be solved, for each bomber-delivered weapon against each SAM type, to obtain the smallest number of attackers required for high confidence DSM.

minimize N (Problem T-1)

subject to the conditions

$$X_1(NOB) \leq .01$$

$$\overline{X}(j) = T(j)\,\overline{X}\,(j-1)$$

$$j = 1,\ldots NOB$$

Since $X_1(NOB)$ is a monotone decreasing function of the number of attackers N, Problem T-1 is solved in AEM by a Newton-Raphson based technique.

The remaining issues to be discussed are the computation of $T(j)$ and the determination of $\overline{X}(0)$. The latter question can easily be dealt with, since the distribution of survivors of N attackers after 0 encounters is simply the distribution of arrivers, which is a function of the non-reprogrammable reliability. Hence

$$\overline{X}(0) = \overline{b}\ (N,R) \qquad \text{(T-13)}$$

The computation of T is somewhat more difficult, and is done by computing each column $\overline{T}^i(j)$ separately. The first observation is that $\overline{T}(j)$ is upper diagonal, since the probability of, for example, 6 survivors given 3 attackers is zero. Consequently T has the form

$$T = \begin{bmatrix} T^0_1 & T^1_0 & T^2_0 & \text{-------} & T^N_0 \\ 0 & T^1_1 & T^2_1 & \text{-------} & T^N_1 \\ 0 & 0 & T^2_2 & \text{-------} & T^N_2 \\ & & & \ddots & \vdots \\ & \text{(all zeros)} & & & T^N_N \end{bmatrix} \qquad \text{(T-14)}$$

The computation of each column $\overline{T}^i$ requires consideration of the relationship between the number of attackers i and the number of defenders L. A random independent encounter between attackers and defenders will be assumed, with the defenders distributed as evenly as possible among the attackers. Two cases are possible.

a. $\underline{i \geq L}$ In this case there are more attackers than defenders. Hence each attacker which encounters a defender meets exactly one, and some attackers meet none. This results in the kind of computation shown in Figure T-6 for the case of 6 attackers and 3 interceptors.

Possible Number of Survivors	Probability
0	0
1	0
2	0
3	Prob (3 kills)
4	Prob (2 kills)
5	Prob (1 kill)
6	Prob (0 kills)

Figure T-6: Distribution of Outcomes

In general, then

$$T_k^i(j) = 0 \qquad k = 0,1,\ldots i-L$$
$$= \bar{b}_\ell^*(L,PK) \qquad k = i-L+\ell,\ldots i+1,\ \ell = 1,\ldots L+1$$
$$= 0 \qquad k > i+1$$

where $\bar{b}^*(L,PK)$ denotes the vector $\bar{b}(L,PK)$ taken in reverse order.

b. $i < L$ In this case the number of defenders exceeds the number of attackers. Consequently, let the L defenders be distributed as evenly as possible among the attackers, so that i_1 attackers are engaged by L_1 interceptors, and i_2 attackers by $L_1 + 1$ interceptors. In this case, $\bar{T}^i$ is given by

$$T_k^i(j) = \sum_{\ell=1}^{k} b_\ell(L_1, PK1) \cdot b_\ell^*(L_2, PK2) \quad (k=0,\ldots i+1) \qquad \text{(T-16)}$$

$$= 0 \qquad k > i+1$$

where $PK1 = 1.-(1-PK)^{L_1}$ and $PK2 = 1. - (1-PK)^{L_2}$.

This computation of T has assumed perfectly reliable interceptors. In the case of the in-flight reliability R_S, this can be relaxed by letting $PK \rightarrow PK \cdot R_S$. The treatment of the interceptor availability requires some discussion. Classically in AEM the treatment of reprogrammable reliabilities has been to reduce the stockpile to the expected number of available interceptors. However this is not appropriate here, because even if an interceptor can be replaced, it requires TBS seconds to reload the launcher. But by definition, this means that the reloaded launcher results in an interceptor which appears in the next encounter arena. Consequently, with respect to the j-th encounter arena, the fact that an interceptor was not launchable is a non-reprogrammable uncertainty that L interceptors will in fact arrive. Hence the net PK is assumed to be computed by

$$PK \rightarrow PK \cdot R_S \cdot A_S$$

In summary, the computation of the high confidence DSM price against a single SAM site consists of the following steps:

1. Determining the number of possible engagement arenas based on the attack geometry and the number of interceptors/launcher.
2. Computing the results of a single engagement arena, expressed in the form of a transition matrix, based on the range of possible surviving attackers and the number of launchers per site.

c. Propagating the distribution of arriving attackers (based on the non-reprogrammable reliability), through the engagement arenas to determine the probability of zero penetrators to the SAM site.

d. Iterating on this procedure to find the minimum number of attackers for which the probability of zero penetrators is less than .01.

5. DSM with ICBM's/SLBM's

In this case, since the defense is assumed to be ineffective, the only consideration is the non-reprogrammable reliability R. Hence N is computed by solving

$$(1.-R)^N \leq .01$$

whence

$$N = \left[\log(.01)/\log(1.-R) + .99\right] \qquad \text{(T-17)}$$

6. Fixed Price DSM

If the analyst specifies a DSM price D in reliable RV's, then as in the case of ICBM's/SLBM's, the only consideration is the requirement to allocate sufficient numbers to have a high confidence of D arrivers. Hence the problem is to

minimize N (Problem T-2)

subject to the condition

$$\sum_{i=0}^{D-1} b_{i+1}(N,R) \leq .01$$

This is solved iteratively starting with the N computed in (T-17).

7. Multi-Site Suppression

The previous discussion has been concerned with high confidence suppression against a single site, achieved by determining the minimum number of warheads N_k^i of type k which must be allocated against a SAM site of type i so that, using the notation of (T-4),

$$P_i(0|N_k^i) \geq .99 \cong 1.$$

This is necessary so that the probability of kill function for the weapon used to attack the target itself does not require modification because of surviving defenders. However, if multiple SAM sites are present, then an additional requirement is necessary.

Let S_i = The number of SAM sites of type i protecting a given target.

N_k^i = The number of warheads of type k required for high confidence defense suppression against a single site of type i .

It will be assumed that all DSM against a single target will be done with a single weapon type. Because of the assumed independence of the individual sites, then,

$$P(0|N_k) = P_1(0|N_k^1)^{S_1} \cdot P_2(0|N_k^2)^{S_2} \cdot P_3(0|N_k^3)^{S_3} \qquad \text{(T-18)}$$

Hence if $P(0|N_k) \geq .99$, it is not sufficient to have $P_i(0|N_k^i) = .99$. Rather, the confidence of successful DSM against each individual site must be increased to have a sufficiently high confidence of suppressing all sites. (For example, if $S_1+S_2+S_3 = 50$ and $P_i(0|N_k^i) = .99$ for all i , then $P(0|N_k)$ is only .605).

Unfortunately, the amount by which the confidence must be increased at each individual site is a function of (S_1, S_2, S_3), whereas previous computations have considered only the single site encounter. Computing and storing the data required to express (N_k^1, N_k^2, N_k^3) for each weapon-target combination (as opposed to each SAM site type) is prohibitive. Consequently an approximate technique has been developed which permits computing and storing only the single site prices (N_k^i), but adjusting these for the target of interest when required. To develop this approximation, it will be assumed that $P_i(0|N_k^i) \cong .99$ for all i . Hence the problem of adjusting (N_k^i) can be viewed in 2 steps.

a. Find the ϵ for which

$$(.99 + \epsilon)^{S_1} \cdot (.99 + \epsilon)^{S_2} \cdot (.99 + \epsilon)^{S_3} \geq .99 \qquad \text{(T-19)}$$

b. For that ϵ , find ΔN_k^i so that

$$P\,(0|N_k^i + \Delta N_k^i) \geq .99 + \epsilon \qquad \text{(T-20)}$$

The first step is solved directly:

$$\epsilon = .99^{\frac{1}{S_1+S_2+S_3}} - .99 \qquad \text{(T-21)}$$

To solve the second, consider Figure T-6, which generically illustrates the relationship used to determine N_k^i.

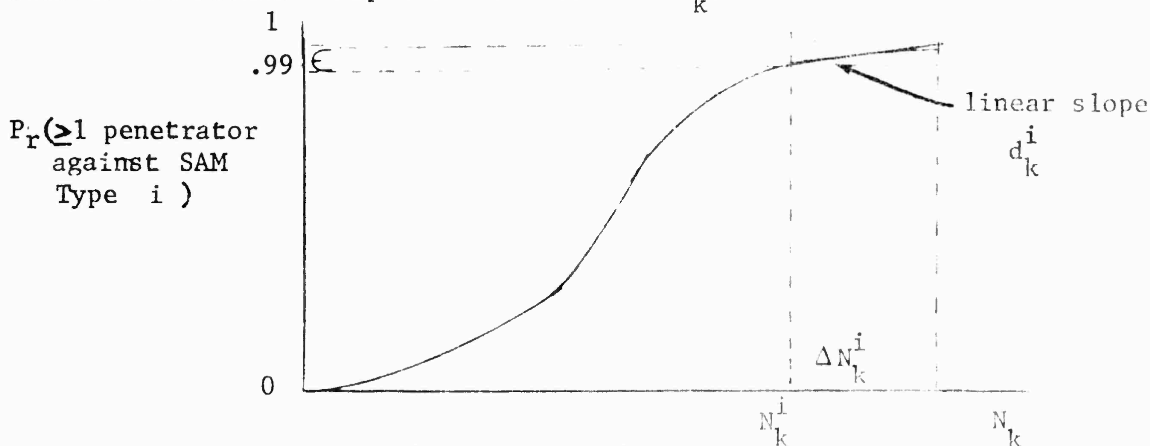


Figure T-6: DSM Confidence Curve

At high confidence regions, the curve is very flat. Consequently a linear estimate of the slope d_k^i deyond the N_k^i point can be used to predict ΔN_k^i. Furthermore, d_k^i can be easily computed at the same time as N_k^i, and, since N_k^i is integer, both N_k^i and d_k^i can be stored in the same computer storage locations. ΔN_k^i is then given by

$$\Delta N_k^i = \left[\epsilon / d_k^i + .99\right] \qquad \text{(T-22)}$$

Consequently the true high confidence DSM price for a given target can be readily computed as

$$N_k = \sum_{i=1}^{3} N_k^i + \Delta N_k^i \qquad \text{(T-23)}$$

8. <u>Inclusion of DSM In The Strategy Generation Process</u>

Once the just described process has developed appropriate defense suppression prices it is necessary to properly account for the DSM option in deciding how to attack a specific target. In AEM the effect is very conveniently dealt with by appropriate modification of the Lagrangian strategy generation process. The Lagrangian strategy generation scheme has been well explored elsewhere in this report. Suffice it to say, at any given stage where a feasible, but non-optimal allocation exists, an improved strategy (if one exists) for a given target can be determined by finding that number of attackers, N_i, of weapon type i, that maximizes H, where

$$H = V \cdot PK(N_i) - N_i \lambda_i \qquad \text{(T-24)}$$

This equation simply allows for balancing the payoff $(V \cdot PK(N_i))$ in attacking the target with the Lagrangian "cost" $(N_i \lambda_i)$.

That value of N_i which maximizes the difference between payoff and cost is the optimal attack level for weapon type i .

As was indicated in equation (T-3) the proper Lagrangian when DSM is required includes an extra DSM "cost", namely $D_K \lambda_K$. This DSM cost is simply that minimal Lagrangian cost required to pay the firm DSM requirement on the target if weapon type i requires such payment. The DSM price, D_K, is known for all weapon types and the Lagrangian multiplier λ_K is known at each stage of the strategy generation process. Therefore, the best DSM weapon on the target is that one which minimizes $D_K \lambda_K$.

These considerations are allowed for in AEM as follows. As each target has a strategy generated for it a check is made to determine if the attacking weapon under consideration requires DSM. If it does, the appropriate DSM Lagrangian price, $D_K \lambda_K$, is added to the weapon price, $N_i \lambda_i$, which maximizes the overall Lagrangian as indicated by equation (T-3). As each weapon type is considered for the target attack an appropriate Lagrangian, H_i, is computed. The best attacking weapon type on the target is therefore that weapon type which results in a maximal H_i, where each H_i has the appropriate DSM cost included.

This process is included even in the more complex hedging strategy generation steps. The only impact, essentially, is to require that a DSM cost be included when certain weapons attack certain targets.

As each selected strategy is inserted into the LP a check is made in order to determine if the chosen attacking weapon requires a companion DSM. If it does the D_K for the correct DSM weapon is inserted into the strategy. The only complexity in the process involves internal AEM bookkeeping to assure that the DSM weapons are properly inserted into the LP, while at the same time those weapons do not take any credit for damage to the target itself.

8. The Impact of Misestimates

If misestimate data is being considered, then even though the allocator is assumed to generate his war plan based on high confidence DSM strategies, the actual confidence of the result may be different. For a given target strategy, define

VD = The expected value destroyed (against the target) based on an estimated high confidence DSM.

$P(0|N_h) \cong 1.$ = The estimated confidence of defense suppression.

$P^*(0|N_h)$ = The actual confidence of defense suppression.

VD* = The actual expected value destroyed.

Then two distinct and drastically different cases occur:

a. $P^*(0|N_h \geq P(0|N_h)$. In this case the attacker allocated more weapons to DSM than would have been required had he known the true situation. However, the confidence of defense suppression remains high. Consequently the warheads allocated against the target were not perturbed, the target damage function remains valid, and VD* = VD.

b. $P^*(0|N_h) < P(0|N_h)$. This is the situation where there is some probability that the defense in fact survived the DSM attack.

Hence

$$VD^* = P^*(0|N_h) \cdot VD + (1 - P^*(0|N_h)) \cdot VD(S) \quad (T-25)$$

where VD(S) denotes the expected value destroyed for the strategy if indeed the defense survived. But the determination of VD(S) is precisely a problem discussed in detail in reference 33, and the inability to compute it in a realistic manner compatible with AEM requirements was one of the principal reasons for modeling only high-confidence DSM. (The proper computation of VD(S) is even more complicated when special purpose target data bases such as the Q-95 or q/a data bases are used, since these assume an optimal patterning of the arriving warheads). Consequently the assumption has been made here that VD(S) = 0, that is, if the defense survives, no target damage results. This is offense conservative, and also probably realistic for the case when a few bombers are allocated against a target, since in this case a surviving defense is likely to nullify the strategy. Consequently (T-25) is replaced by

$$VD^* = P^*(0|N_h) \cdot VD \quad (T-26)$$

U. LINEAR PROGRAMMING VIA GENERALIZED UPPER BOUNDING

1. General

Central to the AEM allocation procedure is the use of a linear program (LP) to select, from among a candidate set of strategies, that subset which maximizes the total expected value destroyed and yet does not violate any weapon, target, or hedging constraints. The LP is used in an iterative fashion on even a single strike game. Since multi-strike games are solved by numerous repetitions of special one-strike games, the number of times an LP may have to be solved in a single AEM run may be large. Consequently the technique used to solve the LP must be extremely rapid. Another requirement is that the amount of core storage required to execute the LP be minimal, so that the maximum amount is available for modeling logic.

In previous versions of the AEM, a standard revised simplex code was used. However, with increased development, in particular the inclusion of separate missile and bomber island defenses, and expanded hedging capabilities, the old revised simplex code would not accommodate the significant number of additional constraints which were required. One solution would have been to simply increase the dimensions of the revised simplex code. This, however, would have caused a significant increase in computer core requirements, and also a disproportionate increase in the time required to solve a single LP. This increase is caused by the additional operations which must be performed to maintain numerical accuracy with a large basis inverse. A second resolution, which was adopted, lies in a new development in advanced LP theory called Generalized Upper Bounding (GUB).

GUB is basically a mathematical theory for dealing with LP problems in which some fraction of the constraint rows have a special structure. This structure must be of the form where a set of rows have the characteristic that one row has non-zero entries in certain columns and all other rows have 0 entries in those columns but non-zero entries in other columns. This structure creates a staircase appearance to the tableau. For example, consider an allocation problem involving three weapon types and three target classes. Then a candidate constraint tableau, expressed in the normal AEM format (see Figure J-1) might be the following:

2			7			1			≤	13	(Weapon Type 1 Constraint)
	3			4			5		≤	6	(Weapon Type 2 Constraint)
		5			2			3	≤	8	(Weapon Type 3 Constraint)
1	1	1							≤	5	(Target Class 1 Constraint)
			1	1	1				≤	3	(Target Class 2 Constraint)
						1	1	1	≤	2	(Target Class 3 Constraint)

Figure U-1. GUB Structure

It happens that when there are such constraint rows the revised simplex algorithm can be adapted so that <u>the special staircase rows need not be kept in the tableau</u>. This results in a significant space saving if large numbers of such rows exist.

In AEM the previous LP could accept up to 80 constraint rows and 50 of them were of this special GUB structure. (The 50 represent the target constraints and they always are of the staircase form.)

The crucial storage is taken up in a matrix of size 80 x 80 (the basis inverse). With GUB, a matrix of size 30 x 30 solves the same problems. This represents an improvement in storage by a factor of 7. Alternatively, for the same storage as the old LP, much larger problems can be solved.

GUB also has computer time advantages for a given problem size. The reasons for these time advantages are complex but they basically result from some simplified bookkeeping steps that eliminate unnecessary operations that occur in standard LP pivots, and from a diminished requirement to maintain GUB numerical accuracy because of the reduced dimensions of the basic inverse.

Implementation of GUB into AEM could have been quite costly except for one basic fact. Under a separate government contract, a GUB subroutine, and all associated routines, have been developed, thoroughly tested, and incorporated into a strategic model OSAGE (reference 22), whose LP is similar in structure to that of a one-strike AEM LP. Consequently, the existing GUB program was resized and meshed into AEM in the current location of the LP program.

Adaptation of GUB into AEM has resulted in a vastly more flexible LP structure for larger problems. Growth in target classes will be of far less importance and many resultant benefits will accrue. The running time for current sized problems will most likely drop. More importantly, growth in problem size will not result in nearly as fast a growth in time.

Commercial GUB codes have been compared with non-GUB codes of comparable capability, and running time improvements by factors of five to ten have been common (see reference 23). Additionally, the GUB included in AEM has been modified to a special list-processing structure (reference 24), which increases its speed even more.

The remainder of this section is divided into three divisions. The first motivates the GUB algorithm by directly considering the revised simplex method for problems of GUB structure. The second division is a compact description of the classical GUB algorithm taken from reference 24, which generally follows the terminology and notation of the original Dantzig-Van Slyke paper (reference 25). Other developments and discussions will be found in references 19 and 26-31. The third division briefly defines and discusses the use of list processing. The complete definition of the list-processing GUB formulation, which is too lengthy and cumbersome to include here, is given in reference 24.

2. GUB Motivation

This section provides a motivation for GUB by considering a transformation of the basis inverse of the standard revised simplex method which is possible for any problem of GUB structure.

Assume an LP problem in standard revised simplex form.

$$\begin{aligned} &\text{Maximize } Z \\ &\text{Subject to the constraints } \bar{A}X = b \\ &\qquad\qquad\qquad\qquad X \geq 0 \end{aligned} \qquad \text{(U-1)}$$

Let $\bar{B}$ be the current basis, and partition

$\bar{A} = (\bar{B} \vdots R)$ and $X = (X^{\bar{B}} \vdots X^R)$, so that

$$\bar{B}X^B + RX^R = b \Rightarrow X^B + \bar{B}^{-1}RX^R = \bar{B}^{-1}b \qquad (U-2)$$

Hence if the inverse of $\bar{B}$ is known, the current solution X^B can easily be determined from (U-2). The revised simplex method uses this fact to advantage, and defines all of the standard simplex operations in terms of operations with $\bar{B}^{-1}$. (See reference 32). Consequently the crucial point is to be able to compute and store $\bar{B}^{-1}$.

Suppose now that A has GUB structure (as in Figure U-1), with L special staircase (or GUB) rows, and M non-staircase (non-GUB) rows. Furthermore, for a given staircase row, let all the columns with an entry in that row be defined as belonging to the same GUB set. (Then there are L GUB sets). It can be shown that at least one column from each GUB set must be in the basis (and hence in $\bar{B}$). Consequently arbitrarily select one such column for each GUB set, and call it the <u>key</u> column for that set. Since the key columns are in the basis, $\bar{B}$ can be re-arranged as follows:

$$\bar{B} = \left(\begin{array}{c|c} A^{\bar{k}} & A^k \\ \hline c & I \end{array}\right) \begin{array}{l} M \text{ non-GUB rows} \\ L \text{ GUB rows} \end{array}$$

where A^k is the MXL matrix which represents the non-GUB row entries in the key columns, $A^{\bar{k}}$ is the MXM matrix which represents the non-GUB row entries in the non-key columns, I is an LXL identity matrix (corresponding to the GUB row entries of the key columns), and c is an LXM matrix of zeros and ones corresponding to the GUB row entries of the non-key columns.

Now consider a transformation $B^* = \bar{B}E_t$, where E_t is the elementary transformation matrix which causes

$$B^* = \bar{B}E_t = \left(\begin{array}{c|c} B & A^k \\ \hline 0 & I \end{array}\right) \begin{array}{l} M \text{ rows} \\ L \text{ rows} \end{array}$$

Using the inverse formula for partitioned matrices (see reference 32),

$$B^{*-1} = \left(\begin{array}{c|c} B^{-1} & -B^{-1}A^k \\ \hline 0 & I \end{array}\right)$$

But $\bar{B}^{-1} = E_t B^{*-1}$, and $E_t = \left(\begin{array}{c|c} I & 0 \\ \hline -c & I \end{array}\right)$

Combining these results gives

$$\bar{B}^{-1} = \left(\begin{array}{c|c} B^{-1} & -B^{-1}A^k \\ \hline -cB^{-1} & cB^{-1}A^k+I \end{array}\right)$$

Hence for any given basis $\bar{B}$, $\bar{B}^{-1}$ can be determined by knowing only

a. The set of key columns

b. B^{-1}, which is of dimension MXM.

c. The matrix c.

Thus all the steps of the revised simplex, which must be performed with the elements of $\bar{B}^{-1}$, can be done, by appropriate redefinition with the elements of B^{-1}, A^i, and c. The GUB algorithm of Dantzig and Van Slyke formalizes the steps required to carry out the revised simplex method using these quantities.

3. Classical Generalized Upper Bounding

The generalized upper bounding (GUB) algorithm is a revised simplex based procedure which applies to linear programming problems of the form

$$\text{maximize } Z \qquad \text{(GUB)}$$

subject to the constraints

$$z-(\bar{c}, \bar{x}) = 0$$
$$A\bar{x} = \bar{b}$$
$$(\bar{e}^k, \bar{x}_k) = 1 \quad k = 1,\ldots L$$

where $\bar{x}$ and $\bar{c}$ are N x 1 column vectors, $\bar{b}$ is an M-1 x 1 column vector, A is an M-1 x N matrix, $\bar{x} = (\bar{x}_0 \vdots \bar{x}_1 \vdots \ldots \bar{x}_L)$ and $\bar{e}^k$ is a column vector, of dimensions equal to $\bar{x}_k$, with all elements equal to unity. (By appropriate scaling, any set of linear constraints which has L constraints of the form

$$(P\bar{e}^k, \bar{x}_k) = d$$

where P is a diagonal matrix and d is scalar, can be reduced to GUB form.) It is also assumed that the system GUB is of full row rank. The k - th GUB set of variables (columns) S_k will refer to those variables (columns) indexed by $\bar{x}_k$. (k = 0,1,...L).

Direct application of the revised simplex procedure to problem GUB requires a basis inverse of dimensions M+L x M+L. For large problems this dimensionality is prohibitive, both because of computation time and potential numerical accuracy problems. The GUB algorithm uses the structure of GUB to define an equivalent system, which uses an inverse of dimensions M x M. All information required to use the revised simplex procedure is available from the smaller inverse plus some additional data. Since in large problems, generally $M \gg L$, the savings in both computer storage and computation time are significant.

The following results apply to GUB structured problems.

<u>LEMMA 1</u>: At least one column from each set S_k is basic ($k = 0,1,\dots L$).

<u>LEMMA 2</u>: The number of sets containing two or more basic columns is at most $M - 1$.

Sets S_k containing two or more basic columns are called <u>essential</u>. Other sets are called <u>non-essential</u>.

Some additional definitions are required. The $M + L \times 1$ column vectors of GUB will be denoted by $\bar{A}^j$ ($j = 0,1,\dots N$), where $\bar{A}^0$ is the vector $(Z,0,0,\dots)'$. A^j denotes the $M \times 1$ column vector identical to the first M components of $\bar{A}^j$. The vector b denotes $(0 \vdots \bar{b})'$. A feasible basis to GUB is $(\bar{A}^{j_1} \vdots \dots \bar{A}^{j_{M+L}})$. Furthermore, for each set S_l ($l = 1,\dots L$), select one variable $x_{k(l)}$ to be the <u>key variable</u>. $A^{k(l)}$ is the <u>key column</u>. (S_0 has no key variable.)

Then, if $A^j \in S_l$, let

$$D^{k(l)} = A^{k(l)}$$

$$D^j = A^j - A^{k(l)} \qquad j \neq k(l)$$

$$d = b - \sum_{l=1}^{L} D^{k(l)}$$

$B = \{ D^j \mid A^j$ is basic and not key $\}$

$\bar{\pi} = (B^{-1})_1$

$\bar{\mu}_1 = -\bar{\pi} A^{k(1)} \quad (1 = 1, \ldots L)$

A^s = current entering column

σ = index for which $A^s \in S_\sigma$

A^{j_r} = current leaving column

ρ = index for which $A^{j_r} \in S_\rho$

A^{*s} = vector for which

$$\bar{A}^s = \sum_{i=1}^{M+L} A_i^{*s} \bar{A}^{j_i}$$

$\bar{D}^s = B^{-1} D^s$

b^* = current value of basic variables

η_i = column number in A corresponding to the i - th column of the working basis B

ν_i = Column number of the key column corresponding to η_i.

Using these definitions, the following results hold.

<u>LEMMA 3</u>: $(\bar{\pi} \mid \bar{\mu})$ is a set of prices for GUB.

<u>LEMMA 4</u>: A^s is determined by solving

$$\min_{\bar{A}^j \in A} \Delta_j = (\bar{\pi}, A^j) + \mu_1 \text{, for } \bar{A}^j \in S_1.$$

LEMMA 5:

$$A_i^{*s} = 1 - \sum_{\nu_t = k(\sigma)} \bar{D}_t^s \quad \text{if } A^{j_i} = A^{k(\sigma)}$$

$$= \bar{D}_t^s \quad \text{if } A^{j_i} = A^{\eta_t} \text{ for some } t.$$

$$= - \sum_{\nu_t = j_i} \bar{D}_t^s \quad \text{if } A^{j_i} = A^{\nu_t} \text{ for some } t.$$

and $$b_i^* = 1 - \sum_{\nu_t = j_i} \bar{d}_t \quad \text{if } A^{j_i} \text{ is key}$$

$$= \bar{d}_t \quad \text{if } A^{j_i} = A^{\eta_t} \text{ for some } t.$$

Using Lemma 5, the computation of the leaving column is done in the usual manner, that is, A^{j_r} is the column which solves

$$\underset{A_i^{*s} > 0}{\text{minimize}} \ \frac{b_i^*}{A_i^{*s}} \qquad i = 1,\ldots M+L$$

where s is determined by Lemma 4.

Given the entering and leaving columns, all quantities are updated as follows. (Only one procedure is applicable at any pivot.)

UPDATING PROCEDURES

A. If S_σ is not essential and $A^{j_r} \in S_\sigma$, then

1. A^S replaces A^{j_r} as the key column in S_σ
2. $\bar{d} = \bar{d} - \bar{D}^S$
3. B remains unchanged.

B. If A^{j_r} is not key, then

1. Update B^{-1} by pivoting on the column $\bar{D}^S$ on the row which $A^{j_r} = A^{k(\rho)}$ occupies in the working basis.

2. $\bar{d} = P\bar{d}$, where P is the pivot matrix.

C. If $A^{j_r} = A^{k(\rho)}$

1. Change the key variable in S_ρ.

2. Transform B^{-1} and $\bar{d}$ to consider the new key column.

3. Update as in procedure B.

Based on these definitions, a macro-definition of the GUB algorithm can be stated.

GENERALIZED UPPER BOUNDING ALGORITHM

Step 1: Enter with B^{-1}, $\bar{d}$, K

Step 2: Price columns using $(B^{-1})_1$, giving min $\Delta_j = \Delta_S$. If Δ_S is positive, go to step 7. If not, assume $\bar{A}^S \in S_\tau$.

Step 3: Compute $\bar{D}^S$, A^{*S}, b^*, A^{j_r}, where $\bar{A}^{j_r} \in S_\rho$. If $\bar{A}^{j_r}$ is key, go to step 5.

Step 4: Pivot on $\bar{D}^S$ in the row corresponding to D^{j_r} in B^{-1} and update $\bar{d}$. Go to Step 1.

Step 5: If S_ρ is essential, go to step 7. Otherwise, make $\bar{A}^S$ key instead of $\bar{A}^{j_r}$. Update $\bar{d}$ and go to step 1.

Step 6: Switch key columns in S_ρ. Update B^{-1} and $\bar{d}$. Go to step 4.

Step 7: Current solution is optimal. STOP

[illegible]. List Processing GUB

Discussions of GUB available in the literature are concerned with a statement of the algorithm, and a mathematical definition of all required elements, but not with the computational implementation of GUB. Computational results with a straightforward implementation of GUB on test problems (see reference 24) showed that is GUB was to be successful in large problems, the implementation must be improved considerably. (The "classical GUB" implementation was based on straight forward coding of the mathematics in the original Dantzig and Van Slyke paper).

The basic method used is to re-structure the operations of the GUB algorithm in a list-processing format. This requires the definition of a number of special sets, lists and pointers. These are all used to minimize data-handling time and virtually eliminate lengthy search procedures and unnecessary operations. A minimum amount of additional high-speed storage is required, but this disadvantage is more than overcome by the reduction in execution time. The final list-processing definitions and technique are the results of extensive experimentation and execution time analysis of the various components of GUB.

The fundamental quantities used in the revised GUB algorithm are the list, the ordered list, and the pointer.

Definition 1

Let $P = \{ p_1, p_2, \dots p_n \}$ and Q be two sets. Then a list y is defined to be a vector $\bar{y}$ whose components are the elements of

$$\pounds_{\bar{y}}(P,Q) = \{i \mid p_i \in Q\}$$

Definition 2:

Let $\bar{V}$ be an n x 1 vector. Then an ordered list Z is defined as a vector $\bar{Z}$ whose components are the elements of the set

$$\pounds(\bar{V}) = \{i \mid V_i > V_{i+1}, \quad i = 1,\ldots n-1\} .$$

Definition 3:

Let $A = \{a_1, a_2, \ldots a_n\}$ and $B = \bigcup_j B_j$ $\quad (\bigcap_j B_j = \emptyset)$ be two sets. Then a pointer y is defined to be a vector $\bar{y}$ whose components define a 1 - 1 mapping $P_{\bar{y}}(A,B)$ from A into B, that is,

$$P_{\bar{y}}(A,B) = \{(i, y_i = j) \mid a_i \to B_j, \quad i = 1,2,\ldots n\} .$$

Definition 4:

(a) The notation y = £, where £ is a list, ordered list, or pointer, will denote that $\bar{y}$ is the vector associated with £.

(b) If $\bar{y} = (V_1, V_2, \ldots V_n)$, then

(1) C(y) = n, where C denotes the cardinality.

(2) $y_i = V_i$

(3) $y^{-1}(i)$ = that j for which $y_j = i$

(4) $i \in y \Rightarrow V_j = i$ for some j.

To clarify further these definitions, consider the following example.

Let $A = \{1,2,3,4,5,6\}$

$B = \{2,8\} \cup \{4,9\} \cup \{6\} = \{2,8,4,9,6\}$

$$D = \{2, 4, 6\}$$

$$\overline{V} = \left[\text{[illegible]}, 8, 47 \right]'$$

Then C(A) = 6, C(B) = 5, and C(D) = 3. Furthermore,

$$x = £_{\overline{x}} (B,A) = \left[1, 3, 5 \right]' \text{ is a \underline{list}.}$$

$$y = P_{\overline{y}} (D,B) = \left[1, 2, 3 \right]' \text{ is a \underline{pointer}.}$$

$$Z = £(\overline{V}) = \left[4, 2, 1, 3 \right]' \text{ is an \underline{ordered list}.}$$

C(x) = 3, C(y) = 3, and C(Z) = 4. Also $x^{-1}(3) = 2$, and $3 \in y$, whereas $4 \notin y$.

Using these concepts, a set of lists/pointers appropriate to GUB has been defined, and the algorithm restructured in terms of them. This restructured algorithm is the one which has been incorporated into AEM.

APPENDIX A: DERIVATION OF KILL PROBABILITIES FOR PERFECTLY RELIABLE WEAPONS VERSUS UNDEFENDED TARGETS

In the discussion of this process, it is illuminating to carry along two examples:

a) LR/CEP = 1, $(L/D)^2 = .9$

b) LR/CEP = ∞, $(L/D)^2 = .2$

NOTATION:

LR = lethal radius
CEP = circular error probable
L = diameter of lethal circle = 2LR
D = target diameter

Case (a) produces a kill probability function which has essentially no straight line portion, whereas case (b) yields a function with an easily recognizable linear part.

Rand Report #RM-2743 contains graphs of $P_k(n)$ ("kill probability" ... "expected coverage by n weapons") versus $n(L/D)^2$ for selected values of the ratio LR/CEP. Visual inspection of these graphs produced $P_k(n)$ data for each pair of LR/CEP and $(L/D)^2$ values. Specifically, for the two above cases, the data obtained were:

Case (a):

n_i	$P_k(n_i)$
1	.39
2	.64
3	.77
4	.85
7	.93

Case (b):

n_i	$P_k(n_i)$
1	.2
2	.4
3	.6
4	.77
5	.87
6	.96
7	.999

It was not possible to read the same number of data points for each LR/CEP and $(L/D)^2$ pair. The number of points in each set varied from as few as 3 to as many as 15 for some curves. (This fact is taken into account in the fitting process later on.)

Each data set was input to a computer program which produced values for the four parameters (T, p, x_T and a) of the approximating function:

$$P_K(n) = \begin{cases} an & (n \le x_T) \\ 1-(1-p)^{n-T} & (n > x_T) \end{cases} \qquad (1)$$

Before discussing the approximation method, it will be useful to derive the least squares' estimate for the slope, c, of a line passing through the origin and having equation:

$$y = cx. \qquad (2)$$

Given the set of points (x_i, y_i) $i = 1,\ldots,m$, the function to be minimized is

$$S = \sum_{i=1}^{m} (y_i - cx_i)^2$$

$$\frac{dS}{dc} = 2 \sum_{i=1}^{m} (y_i - cx_i)(-x_i)$$

$$= -2 \left(\sum_{i=1}^{m} x_i y_i - c \sum_{i=1}^{m} x_i^2 \right)$$

For a minimum:

$$\sum_{i=1}^{m} x_i y_i - c \sum_{i=1}^{m} x_i^2 = 0$$

$$c = \frac{\sum_{i=1}^{m} x_i y_i}{\sum_{i=1}^{m} x_i^2} \qquad (3)$$

Note that the linear part of equation (1) is of the form (2). The non-linear part may be transformed so that it is also of the form (2):

$$P_K(n) = 1 - (1-p)^{n-T}$$

$$(1-p)^{n-T} = 1 - P_K(n)$$

$$(n-T)\ \ln(1-p) = \ln\left[1 - P_K(n)\right] \qquad (4)$$

Defining

$$x = n-T$$
$$y = \ln\left[1 - P_K(n)\right] \qquad (5)$$
$$c = \ln(1-p)$$

it is clear that equation (4) is equivalent to (2).

Thus, for a fixed value of the parameter T, the $\left[n_i,\ P_K(n_i)\right]$ pairs may be transformed to (x_i, y_i) pairs by (5), and a slope, c, may be estimated using (3). This may be converted to an estimate of the parameter p, since

$$c = \ln(1-p)$$

$$e^c = 1-p$$

$$p = 1 - e^c \qquad (6)$$

The process involved in producing the parameters of equation (1) for a given data set may be described as follows:

1) Assuming initially that T = 0 (i.e., that all of the given data was non-linear), equations (5), (3) and (6) were used to produce an approximate value for p.

2) The sum of squares in the n, $P_K(n)$ plane was computed as a measure of goodness-of-fit:

$$S_o = \sum_{i=1}^{m} \left[1 - (1-p)^{n_i} - P_K(n_i) \right]^2 \qquad (7)$$

3) The data set was then divided successively into two subsets, the first of which contained the 1st k points $[n_1, P_K(n_1)], \ldots, [n_k, P_K(n_k)]$. The values of k assigned were $k = 1, \ldots, m-1$.

4) For each value of k, the slope parameter, a, of equation (1) was estimated by fitting a straight line to the first k points according to

$$a_k = \frac{\sum_{i=1}^{k} n_i P_K(n_i)}{\sum_{i=1}^{k} n_i^2} \qquad (8)$$

which is similar to equation (3).

5) At th[illegible] point, an iterative process wa[illegible] [illegible] the parameters T, [illegible] and x_T of a curve with the f[illegible] [illegible]ies:

a) [illegible] it [illegible] the m-k remain[illegible]

b) [illegible] a line passing [illegible]

[illegible] a_k.

The iteration was begun by selecting a value [illegible] T (namely $T_o = k/2$) and estimating p usi[illegible] (6), where

$$= \frac{\sum_{i=k+1}^{m} x_i y_i}{\sum_{i=k+1}^{m} x_i^2} \tag{9}$$

gives the modification of equation (3) necessary to exclude the linear data points.

6) To produce the point of tangency, x_T and the next value of the parameter T, it is necessary that two conditions are satisfied:

$$\text{a)}\quad a_k\, x_T = 1-(1-p)^{x_T-T} \tag{10}$$

$$\text{b)}\quad a_k = -(1-p)^{x_T-T} \ln(1-p) \tag{11}$$

Equation (10) is the requirement that the ordinates of the linear and nonlinear portions be equal at the tangent point, while (11) requires the slopes to be equal there. From (11) and (5):

$$-(1-p)^{x_T-T} = \frac{a_k}{\ln(1-p)} = \frac{a_k}{c} \tag{12}$$

Thus, from (10):

$$a_k x_T = 1 + \frac{a_k}{c}$$

$$x_T = \frac{1}{a_k} + \frac{1}{c} \qquad (13)$$

That is, the tangent point is the sum of the reciprocals of the linear and nonlinear slope estimates. And, from (12):

$$(x_T - T)\ln(1-p) = \ln\left[-\frac{a_k}{c}\right]$$

$$x_T - T = \frac{\ln\left[-\frac{a_k}{c}\right]}{c}$$

$$T = x_T - \frac{\ln\left[-\frac{a_k}{c}\right]}{c} \qquad (14)$$

Thus, use of equations (13) and (14) produce values of x_T and T which are compatible with a_k and the current p value, and satisfy the requirements stated in step 5. Experience has shown this iteration to be convergent, and with ever-decreasing sums of squares in the log plane - that is,

$$s = \sum_{i=k+1}^{m} \left(\ln\left[1-P_K(n_i)\right] - c\left[n_i - T\right]\right)^2 \qquad (15)$$

decreases monotonely to a limit.

7) The goodness-of-fit measure of interest, however, is in the real plane. Thus, at each iteration, the current sum of squared deviations, S, is computed for the $P_K(n_i)$ values in the given data

set. The deviations are measured from the current nonlinear curve for values of n_i greater than the current tangent point, and from the linear part otherwise. The parameters of the curve yielding the minimum value of S encountered, are saved. Denoting by S_k the value of this minimum for each value of $k = 1,\ldots,m-1$, then the overall best fit is chosen to be that line-curve combination corresponding to

$$S_{min} = \min \left\{ S_o, S_1, \ldots, S_{m-1} \right\}. \tag{16}$$

For cases where $S_{min} = S_o$, the parameter x_T was set to zero (so that $T = x_T = 0$) and the slope set equal to the derivative of $P_K(n)$ at the origin:

$$\begin{aligned} a &= P_K'(n = 0) \\ &= -(1-p)^o \ln(1-p) \\ &= -\ln(1-p) \\ &= -c \end{aligned} \tag{17}$$

It is interesting to note that with this definition of the slope, equation (13) remains valid, yielding

$$x_T = \frac{1}{a_o} + \frac{1}{c} = -\frac{1}{c} + \frac{1}{c} = 0.$$

Application of the above procedure yielded the following parameter values for the two sample data sets:

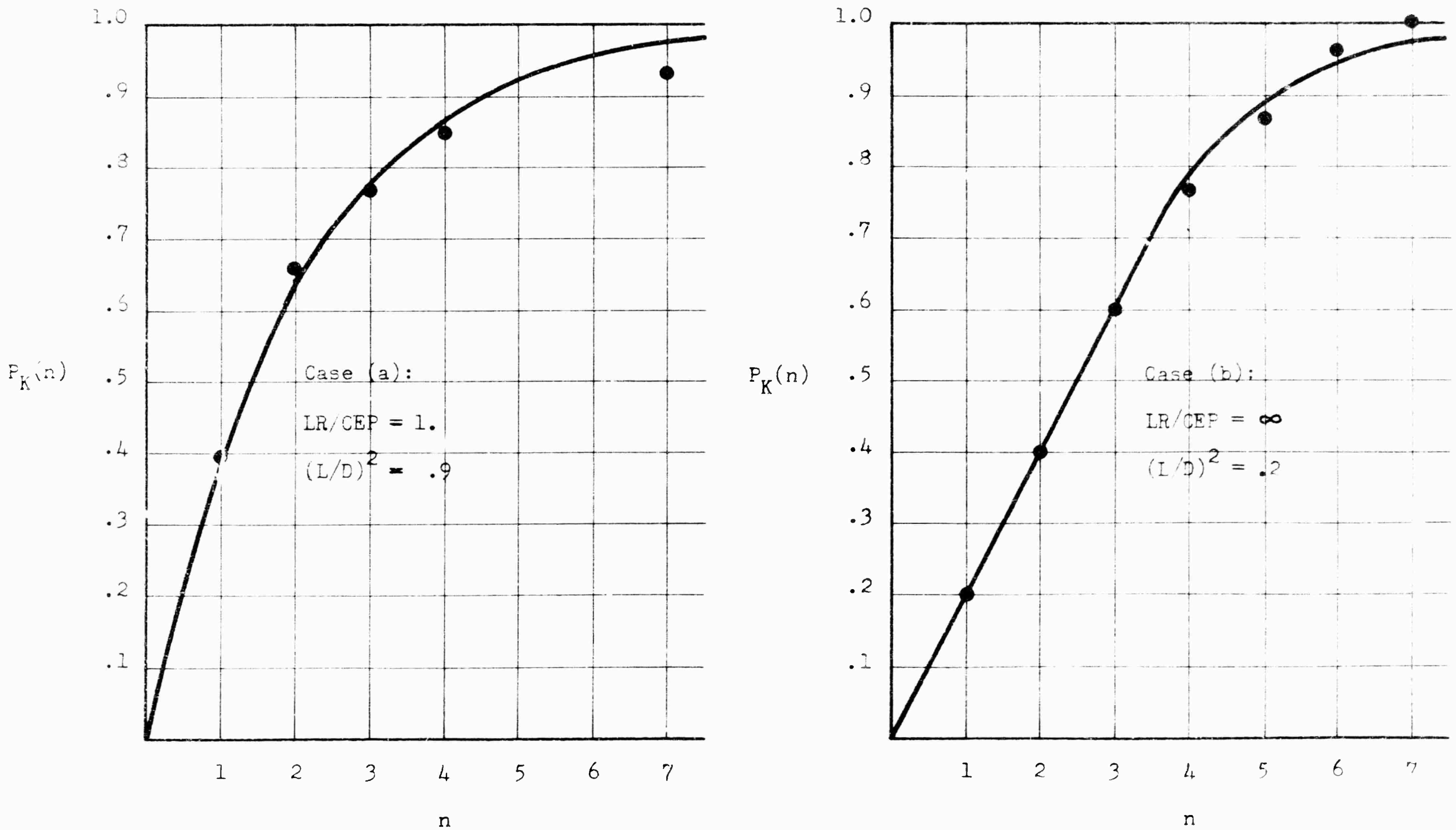
a) $T = .086$

$p = .404$

$x_T = .634$

$a = .390$

FIGURE 1 EXAMPLES OF THE INITIAL FIT



LR/CEP	$(L/D)^2$	T	p	m
[illegible]	.01	.484	[illegible]	15
	[illegible]	.002	[illegible]	[illegible]
	.1	0	.073	5
	.2	.116	.111	5
	.4	.191	.140	5
	.6	0	.138	5
	.8	.058	.148	4
1	.1	0	.102	7
	1.	0	.384	4
	1.2	0	.406	4
	.01	0	.012	12
	.05	0	.058	12
	.25	.036	.195	5
	.5	.016	.291	5
	.75	.020	.343	5
	1.	0	.390	4
	1.	.076	.408	5
	.075	0	.078	5
	.1	.106	.101	5
	.2	0	.164	5
	.4	.019	.259	5
	.6	.084	.347	5
	.8	.083	.384	5
	.9	.086	.404	5
2	.1	.842	.149	8
	1.	0	.668	4
	1.2	0	.661	3

LR/CEP	$(L/D)^2$	T	p	m
2	.01	9.019	.016	12
	.05	1.804	.079	12
	.1	.545	.135	5
	.2	.390	.260	5
	.4	.104	.414	5
	.6	.154	.572	4
	.8	0	.617	4
	.9	.190	.688	4
3	.1	2.290	.217	10
	1.	.430	.191	3
	1.	.354	.882	3
	1.2	.759	.998	3
	.01	24.904	.026	15
	.05	4.981	.122	15
	.4	.252	.495	3
	.6	.200	.628	3
	.8	.258	.765	3
	.9	0	.736	3
∞	.1	6.013	.532	12
	.2	1.794	.506	7
	.4	.945	.708	4
	.6	.675	.910	4
	.8	.512	.961	3
	.01	62.458	.080	12
	.05	12.492	.340	12
	.4	.926	.701	4
	.6	.674	.910	4
	.8	.506	.960	3

FIGURE 2 RESULTS OF INITIAL FITTING PROCESS

(m = No. of Pts. in Each Input Data Set)

b) $T = 1.794$

$p = .506$

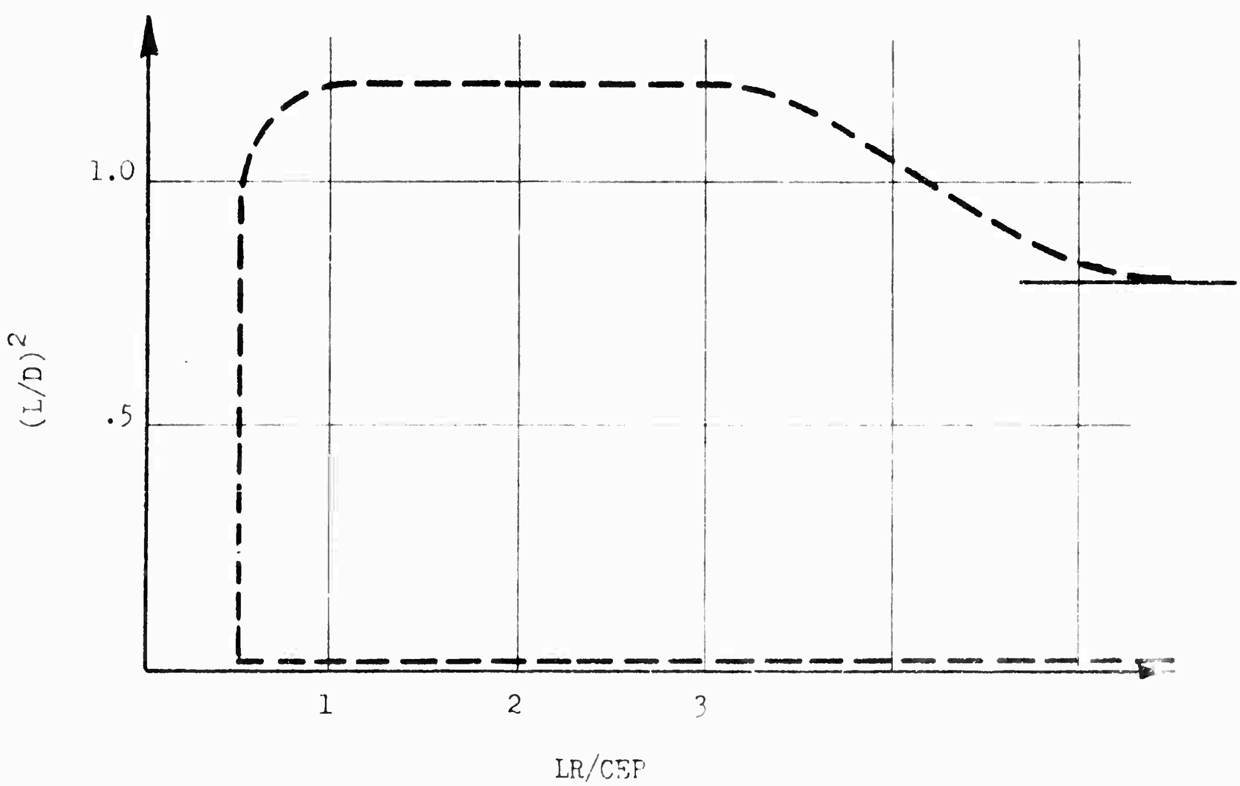
$x_T = 3.581$

$a = .200$

Graphs of the initial data sets and the fitted $P_K(n)$ functions for these two cases are given in Figure 1.

In general, any two of the four parameters (T, p, x_T and a) are sufficient to specify uniquely the complete $P_K(n)$ function. Within the Arsenal Exchange Model, it is particularly convenient to use the T and p values. Therefore, the emphasis in the analysis to follow will be on these two of the four parameters.

Figure 2 tabulates (according to LR/CEP and $(L/D)^2$ values) the number of points in each data set read from the Rand curves and input to the computer program described above, and the resulting T and p estimates. Inspection of this table shows that the region of data collection may be sketched as follows:



The next task was to define the parameters of the chosen $P_K(n)$ [illegible] (equation (1)) in terms of the LR/CEP and $(L/D)^2$ [illegible] the tabulated data, some asymptotic [illegible] the parameter p. It is known that for a point target, the single shot kill probability is given by

$$P_{SSK} = 1 - \left(\frac{1}{2}\right)^{\left(\frac{LR}{CEP}\right)^2} \tag{18}$$

Assuming that the parameter T is zero for a point target in the formulation of kill probability according to equation (1), it follows that

$$\begin{aligned} P_K(1) &= 1 - (1-p)^{1-0} \\ &= p \end{aligned} \tag{19}$$

That is, the p parameter may be interpreted as single shot probability of kill when the target is small. Thus, $p \to P_{SSK}$ as $D \to 0$. Or, stating the same result in a more useful form:

$$p \to 1 - \frac{1}{2}^{\left(\frac{LR}{CEP}\right)^2} \quad \text{as} \left(\frac{L}{D}\right)^2 \to \infty \tag{20}$$

for a fixed value of LR/CEP.

Thus the task of relating the parameters of equation (1) to the LR/CEP and $(L/D)^2$ ratios was initiated by fitting the tabulated values of the p parameter (for constant LR/CEP) using a function of $(L/D)^2$ which approaches P_{SSK} in accordance with property (20).

After several unsuccessful attempts with various infinite domain functional forms, the following (unlikely) form was selected and proved satisfactory:

Let $x = (L/D)^2$ (21)

$$p(x) = P_{SSK}\left[\frac{x}{C}\left(1-\left[1-\frac{x}{C}\right]^{B}\right)+\left(1-\frac{x}{C}\right)\left(\frac{x}{C}\right)^{\frac{1}{B}}\right]$$

for $0 \leq x \leq C$. (22)

Also,

$p(x) = P_{SSK}$ for $x \geq C$. (23)

The function given by equation (22) increases monotonically with increasing x, from a minimum of $p(0) = 0$ to a maximum of $p(C) = P_{SSK}$. It also has the property that $p'(C) = 0$ if $B > 1$. The interpretation of B and C is that they are the parameters of the family, with B determining the shape of the initial portion, and C the point at which the limiting asymptote (P_{SSK}) is achieved.

Ordinarily, both B and C must be estimated from the tabulated p values. However, in the CEP = 0 case (i.e., LR/CEP = ∞) the value

$C = C_{\infty} = 1$ (24)

may be assigned a priori, since perfect single shot kill is achieved if and only if $D \leq L$ (i.e., if and only if $(L/D)^2 = x \geq 1$).

Therefore, for LR/CEP = ∞ :

$$p(x) = x\left(1-\left[1-x\right]^{B}\right)+(1-x)\,x^{\frac{1}{B}}$$

for $0 \leq x \leq 1$ (25)

and

[illegible] for x [illegible]. (26)

[illegible] (for finite values of LR/CEP, [illegible] in equation [illegible] for LR/CEP = ∞, was done by least squares. The methodology is discussed in some detail below. The results are as follows:

LR/CEP	B	C
.5	2.169	.632
1.0	1.845	1.548
2.0	1.640	1.727
3.0	1.613	1.320
∞	2.731	(1.000)

(27)

The curves specified by these values of the parameters, in accordance with equations (18), (21), (22) and (23), are plotted in Figures 3 through 7 inclusive. Also, appearing in these figures are the data points to which the curves were fit, namely the tabulated p values of Figure 2. Each point is labeled with the corresponding value of m given in Figure 2, the number of data points originally read from the Rand curves and input to the computer program (described above) which produced each plotted value of p. These labels were used as relative weights for the data points in the curve-fit process.

Prior to a description of this process, it is convenient to discuss a numerical procedure for solving the simultaneous equations:

$$F_1(u,v) = 0$$

$$F_2(u,v) = 0 \qquad (28)$$

Conceptually, think of F_1 and F_2 as surfaces as in the following sketch:

FIGURE 3

(LR/CEP = .5)

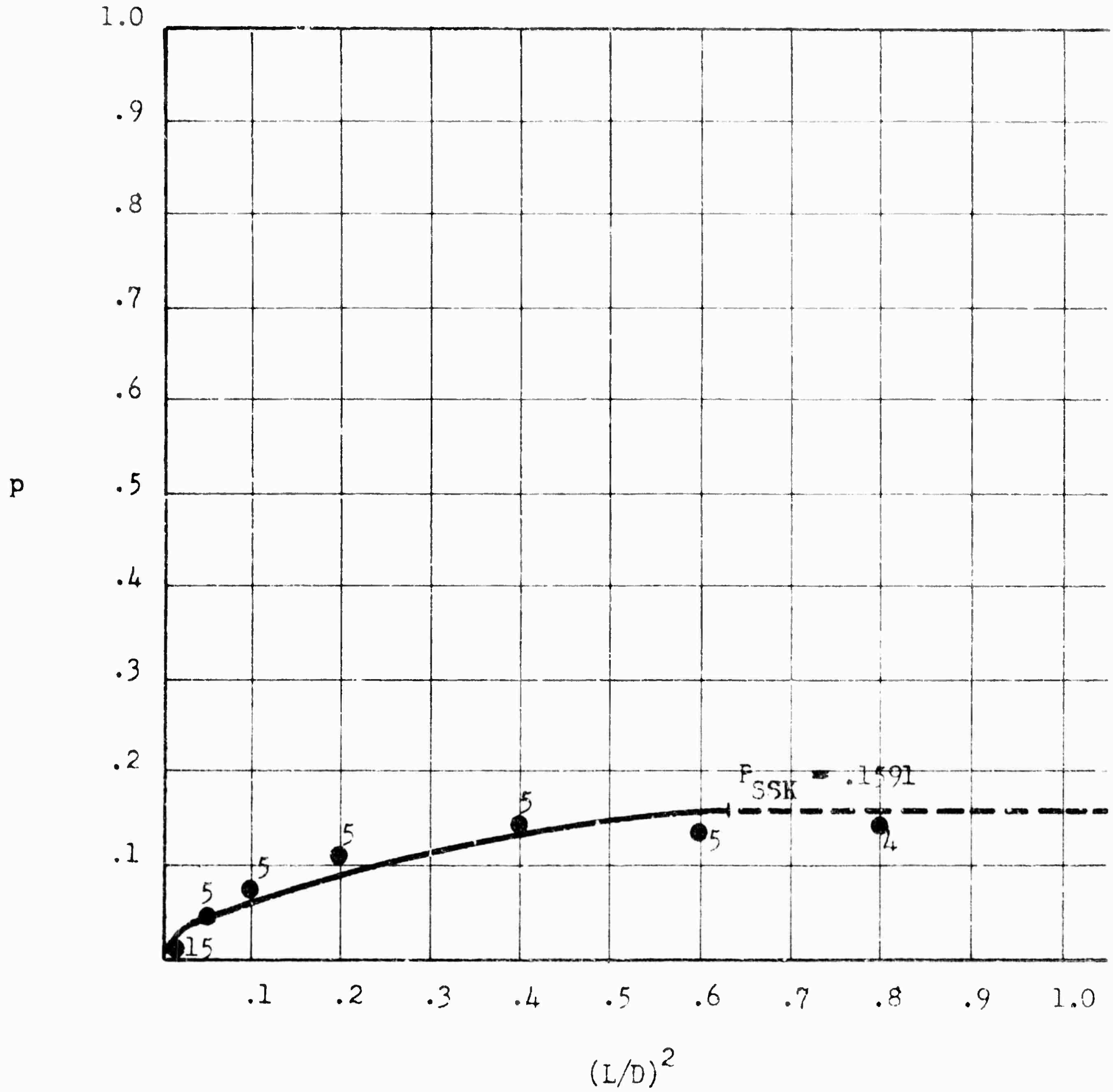


FIGURE 4

(LR/CEP = 1.0)

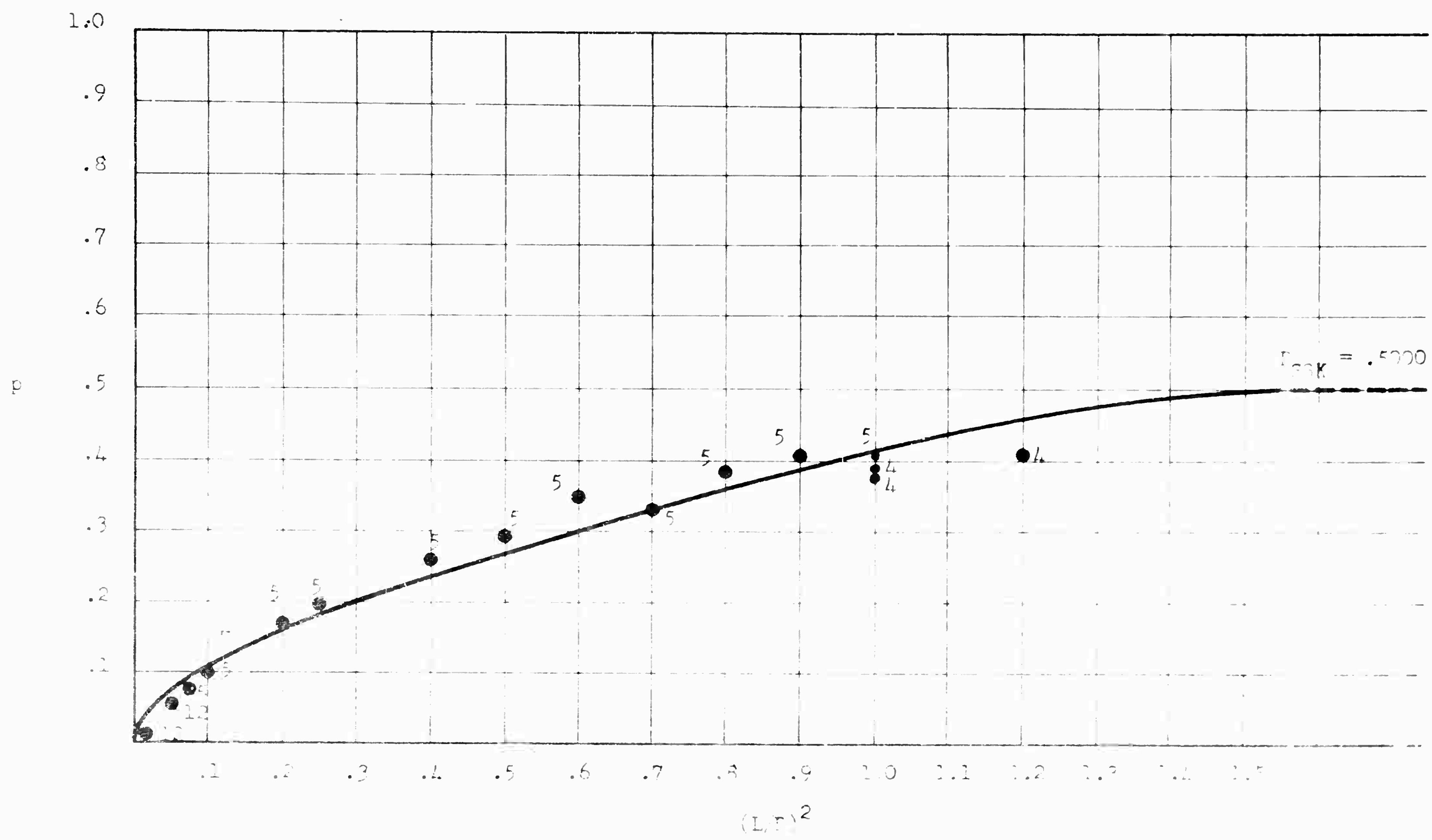


FIGURE 5

(LR/CEP = 2.0)

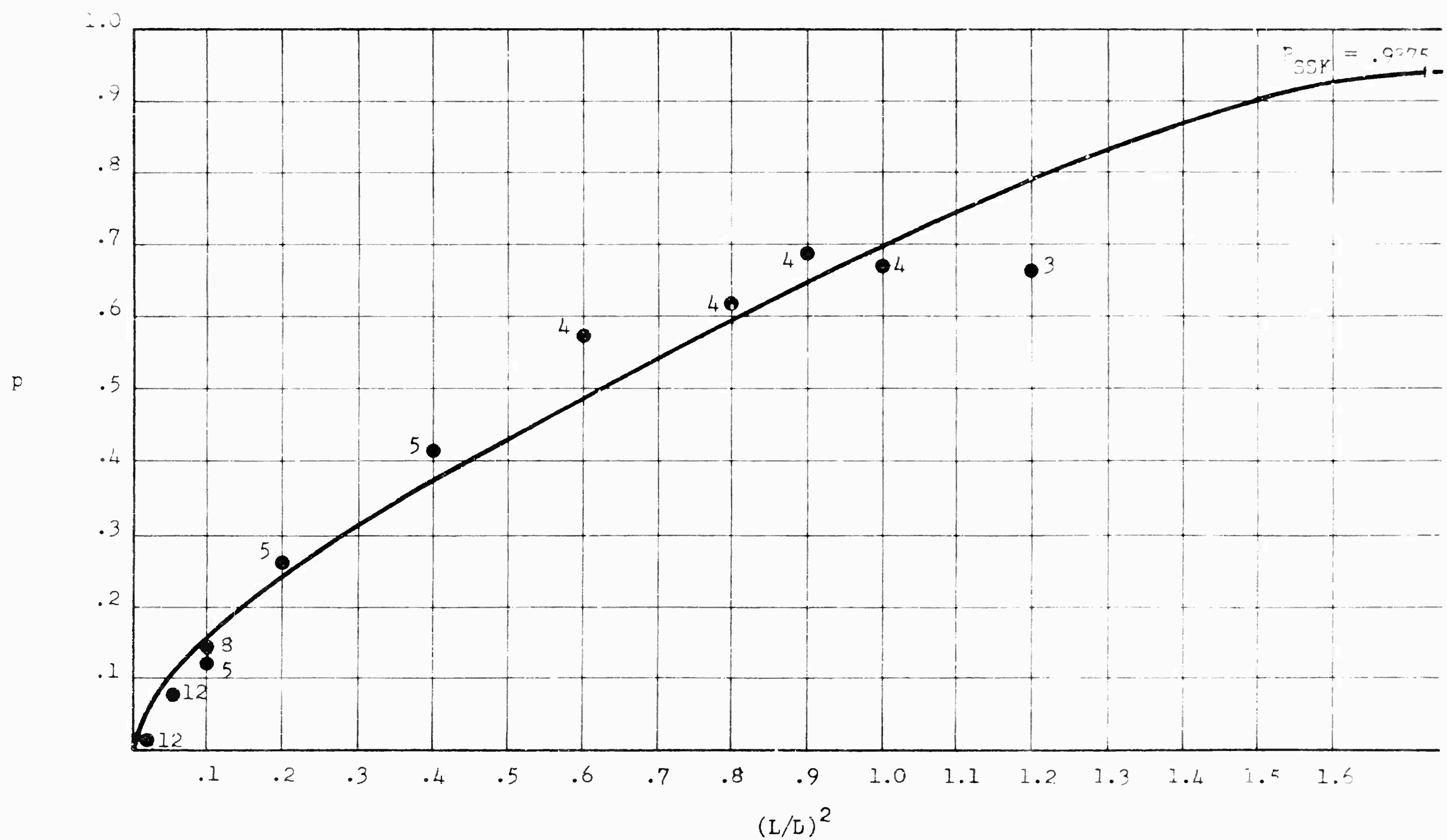


FIGURE 6

(LR/CEP = 3.0)

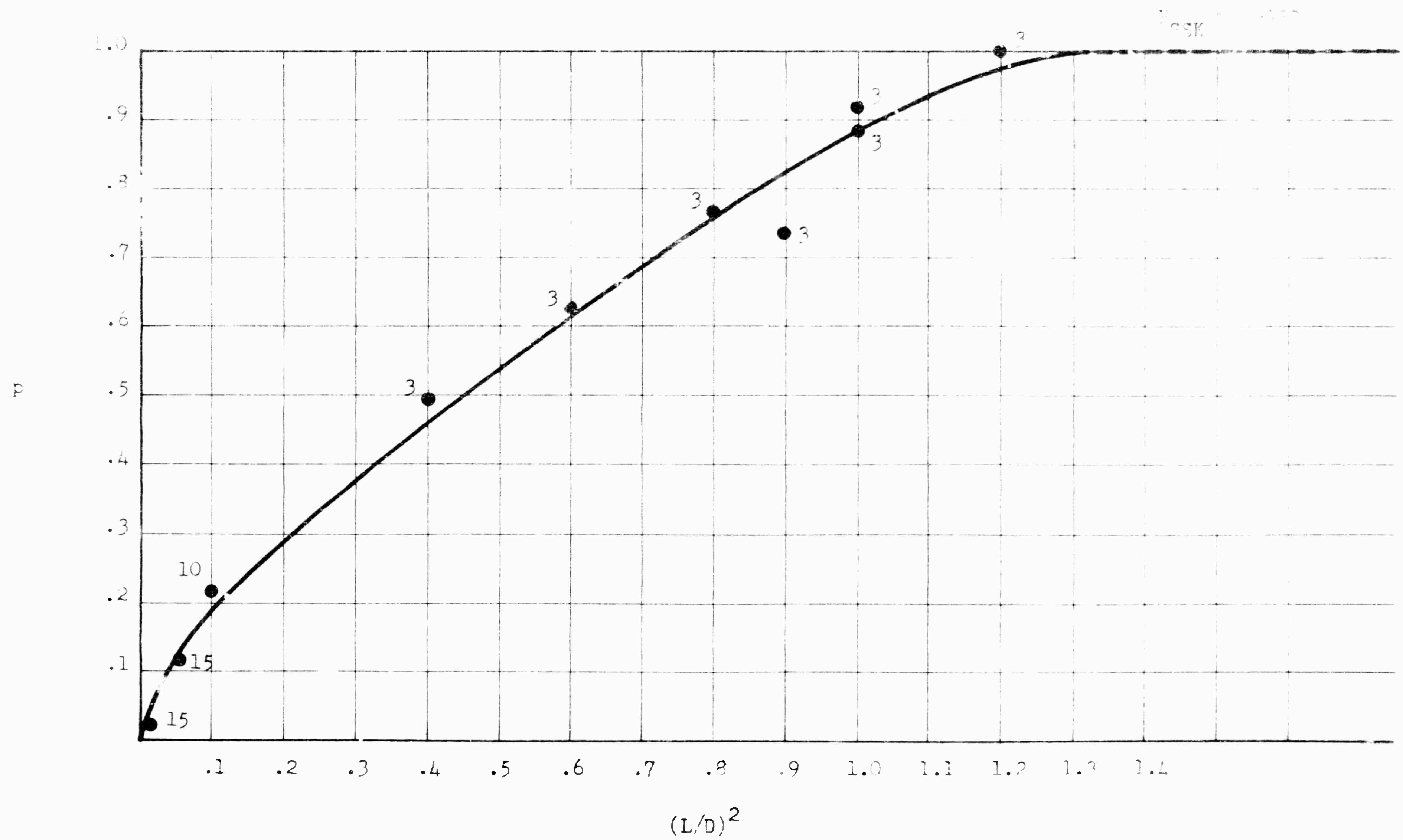
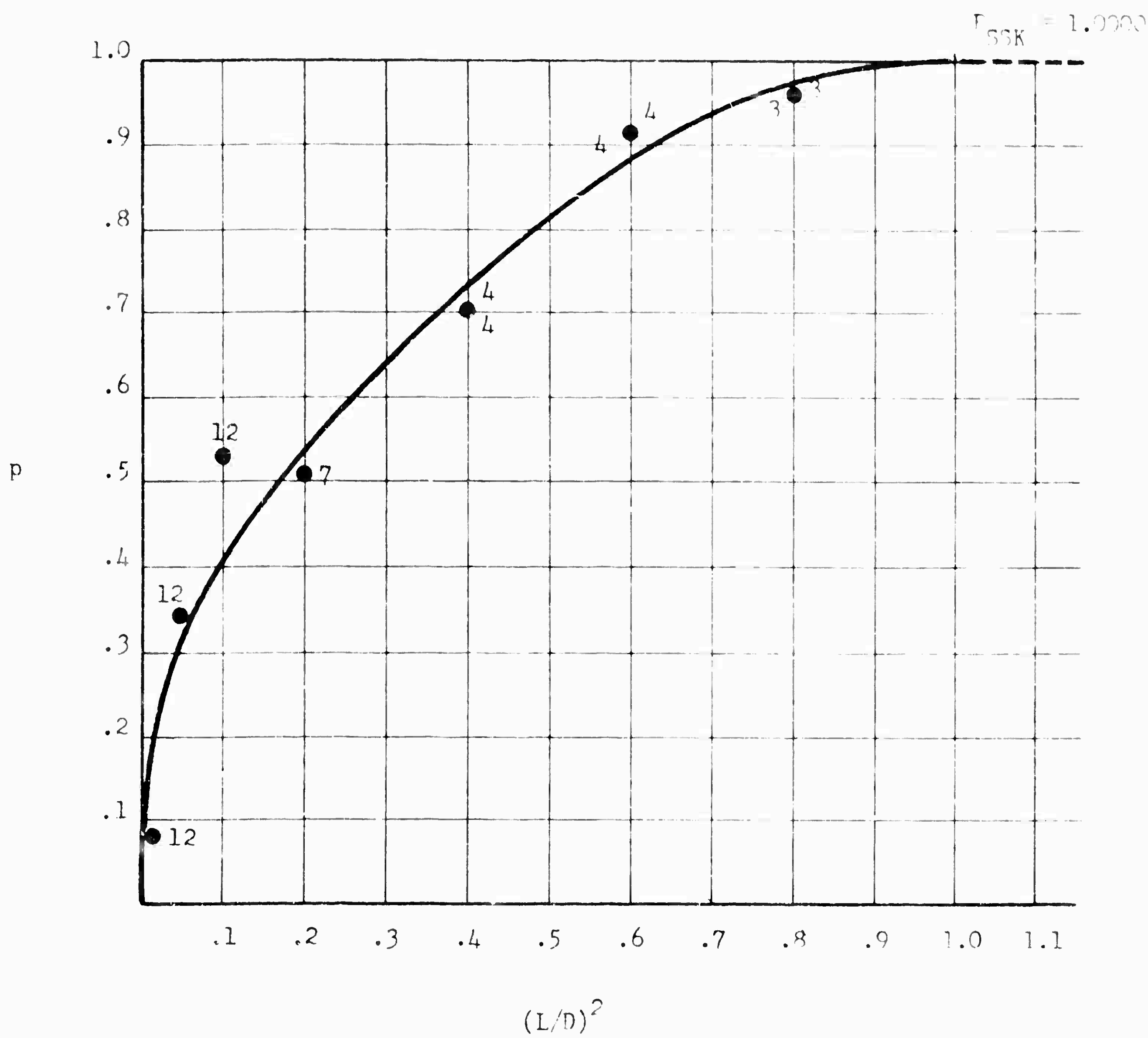
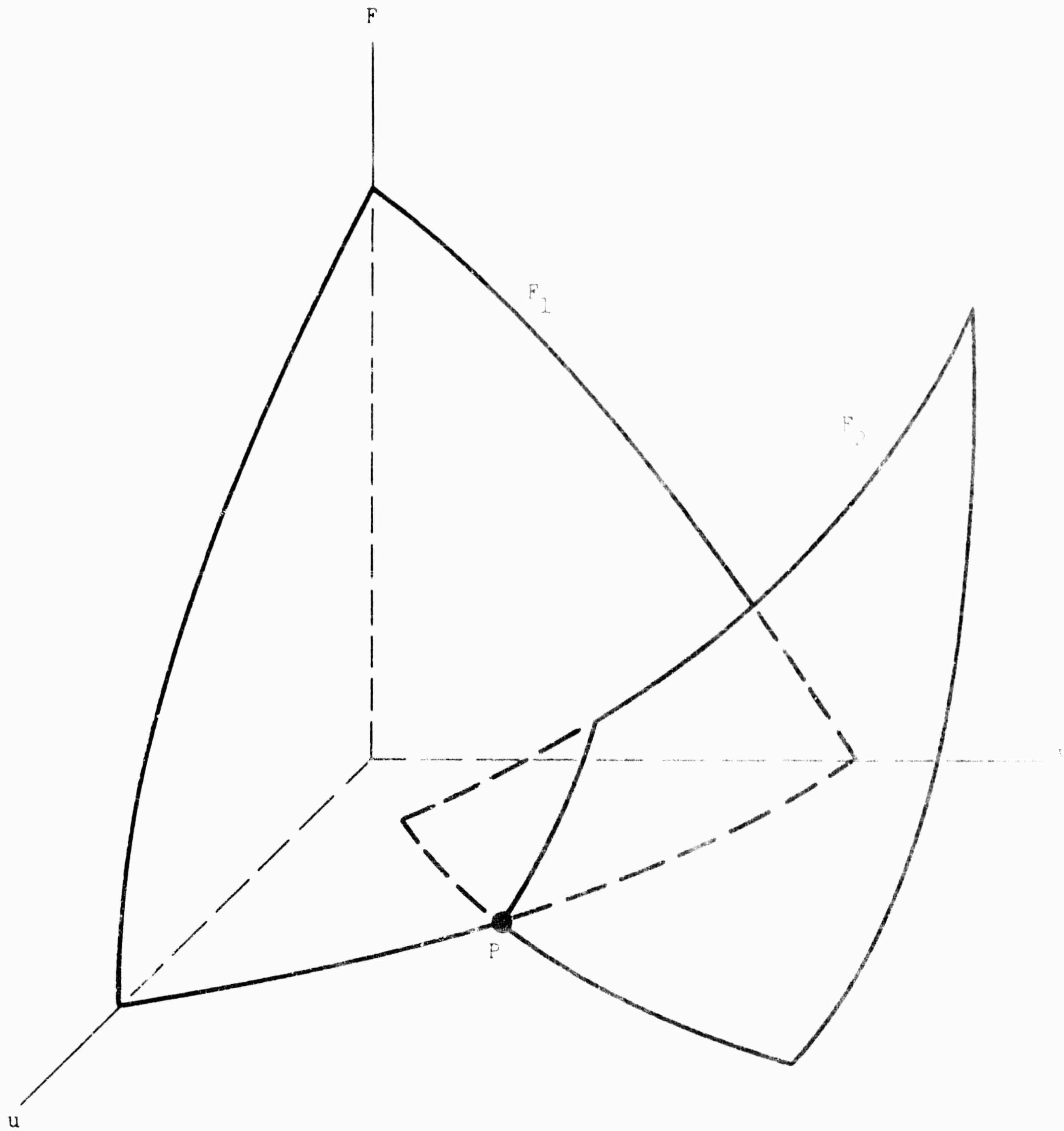


FIGURE 7

(LR/CEF = ∞)




F
F1
F2
P
u

If circumstances are as "nice" as those pictured, then the solution of equations(28) is given by the u and v coordinates of the point P.

It is also intuitively clear that in this case the planes tangent to F_1 and F_2 (at P) and the u, v plane all intersect at the point P, and that this point is uniquely determined by the intersection.

This fact suggests an iterative procedure which might converge to the solution point P. Starting with an initial guess $(u = u_o,\ v = v_o)$, find the tangent planes at the points $F_1(u_o, v_o)$ and $F_2(u_o, v_o)$. Then find the intersection of these tangent planes with the u, v plane, thus determing a new point (u_1, v_1) in the iteration.

The equation of the plane tangent to the surface F_i at the point (u_j, v_j) is:

$$T_i(u,v) = F_i(u_j,v_j) + F_{i_u}(u_j,v_j)\,(u - u_j) + F_{i_v}(u_j,v_j)\,(v - v_j) \tag{29}$$

where

$$F_{i_u}(u_j,v_j) = \left.\frac{\delta F_i}{\delta u}\right|_{\substack{u = u_j \\ v = v_j}} \qquad F_{i_v}(u_j,v_j) = \left.\frac{\delta F_i}{\delta v}\right|_{\substack{u = u_j \\ v = v_j}} \tag{30}$$

$i = 1, 2.$

The next point (u_{j+1}, v_{j+1}) is obtained by solving

$$F_1(u_{j+1}, v_{j+1}) = 0$$

$$F_2(u_{j+1}, v_{j+1}) = 0 \qquad (31)$$

In simplified notation, equations (31) are:

$$F_1 + F_{1_u}(u_{j+1} - u_j) + F_{1_v}(v_{j+1} - v_j) = 0$$

$$F_2 + F_{2_u}(u_{j+1} - u_j) + F_{2_v}(v_{j+1} - v_j) = 0$$

or

$$\begin{bmatrix} F_{1_u} & F_{1_v} \\ F_{2_u} & F_{2_v} \end{bmatrix} \begin{bmatrix} u_{j+1} - u_j \\ v_{j+1} - v_j \end{bmatrix} = \begin{bmatrix} -F_1 \\ -F_2 \end{bmatrix} \qquad (32)$$

The solution of equations (32) are:

$$u_{j+1} - u_j = \frac{\begin{vmatrix} -F_1 & F_{1_v} \\ -F_2 & F_{2_v} \end{vmatrix}}{\begin{vmatrix} F_{1_u} & F_{1_v} \\ F_{2_u} & F_{2_v} \end{vmatrix}} = \frac{F_2 F_{1_v} - F_1 F_{2_v}}{F_{1_u} F_{2_v} - F_{2_u} F_{1_v}}$$

and

$$v_{j+1} - v_j = \frac{\begin{vmatrix} F_{1_u} & -F_1 \\ F_{2_u} & -F_2 \end{vmatrix}}{\begin{vmatrix} F_{1_u} & F_{1_v} \\ F_{2_u} & F_{2_v} \end{vmatrix}} = \frac{F_1 F_{2_u} - F_2 F_{1_u}}{F_{1_u} F_{2_v} - F_{2_u} F_{1_v}}$$

Therefore, the iteration equations are:

$$u_{j+1} = u_j + \frac{F_2 F_{1_v} - F_1 F_{2_v}}{F_{1_u} F_{2_v} - F_{2_u} F_{1_v}}$$

$$v_{j+1} = v_j + \frac{F_1 F_{2_u} - F_2 F_{1_u}}{F_{1_u} F_{2_v} - F_{2_u} F_{1_v}} \qquad (33)$$

where the evaluations of the function F_i and their partial derivatives are to be at the point (u_j, v_j).

The one-dimensional Newton-Raphson iterative procedure may be obtained from equations (33) by setting

$$F_1 (u,v) = F_1 (u)$$

$$F_2 (u,v) = v. \qquad (34)$$

In this case

$$u_{j+1} = u_j + \frac{0 - F_1(u_j)}{F_1'(u_j) - 0}$$

$$u_{j+1} = u_j - \frac{F_1(u_j)}{F_1'(u_j)} \qquad (35)$$

and

$$v_{j+1} = v_j + \frac{0 - v_j F_1'(u_j)}{F_1'(u_j) - 0}$$

$$= v_j - v_j$$

$$= 0 \qquad (36)$$

Equation (35) is the Newton-Raphson [illegible] (36) is a quickly-convergent method of [illegible]

The above discussion does [illegible] convergence of (33) or (35). Rather, these equations [illegible] indiscriminately in the analysis to follow, with only two questions [illegible]:

1) Did the process converge to a numerical [illegible]?

2) Did the resulting parameters specify a curve which fit the data visually?

The answers to both of these have been yes in every case.

Returning to the problem of fitting the parameter p as a function of $(L/D)^2$, the sum of squares for both (22) and (25) is given by:

$$S = \sum_{i=1}^{m} w_i (p_i - p(x_i))^2 \qquad (37)$$

where, for fixed LR/CEP, x_i, p_i and w_i are (respectively) the values of $(L/D)^2$, p and m tabulated in Figure 2. (m is the number of points tabulated).

For finite LR/CEP, p(x) is given by (22). In this case

$$\frac{\partial S}{\partial B} = 0$$

$$\frac{\partial S}{\partial C} = 0 \qquad [illegible]$$

are conditions for the least squares solution, [illegible] of the form (28) with u = B and v = C. Thus, equations [illegible] were applied.

For infinite LR/CEP, equation (25) yields p(x), in which case the condition for minimum S is

$$\frac{\partial S}{\partial B} = 0 \tag{39}$$

which is of the form $F_1(u) = C$ with $u = B$. In this case equation (35) was applicable.

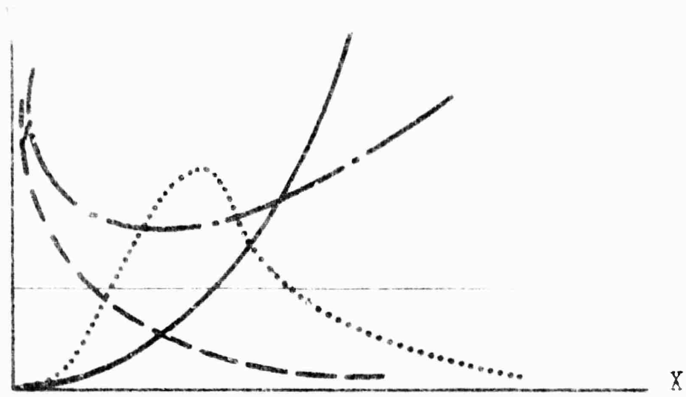
The starting values used were B = C = 2 in the finite LR/CEP cases, and B = 1 in the LR/CEP = ∞ case. In this latter instance, the initial weighted fit seemed too heavily influenced by the point at $(L/D)^2 = .1$ (see Figure 7). Therefore, another curve fit was performed with $w_i = 1$, $(i = 1,\ldots,n)$, which produced a more satisfactory description of the data. The unweighted treatment resulted in the curve plotted in Figure 7, which has as parameter B = 2.731.

The preceding discussion is an outline of the techniques used to fit p(x), but does not give all details. Needless to say, the expressions for the derivatives required in minimizing (37) are tedious, and would occupy undue space if presented here. However, enough of the theory has been given to enable understanding of the results (and their duplication, if desired).

The next step in the analysis was to relate the B and C parameters to the LR/CEP ratio. The basic data is that given in (27). A function playing an important role in this task was

$$y = ab^x x^c. \tag{40}$$

Depending on the values assumed by its parameters, the function of equation (40) may graph in any of the following fashions:



Thus, (40) defines a rather general family of functions. To estimate values of a, b and c which specify a curve approximating a set of known points $\{ (x_i, y_i) \}$ $i = 1, \ldots, n$, first note that

$$\ln y = \ln a + x \ln b + c \ln x \tag{41}$$

i.e. that

$$Y = a_o + a_1 X_1 + a_2 X_2 \tag{42}$$

where we have substituted

$$\begin{aligned} Y &= \ln y \\ X_1 &= x \\ X_2 &= \ln x \\ a_o &= \ln a \\ a_1 &= \ln b \\ a_2 &= c \end{aligned} \tag{43}$$

The approach is to derive least squares estimates for (a_o, a_1 and a_2 in (42), and convert them to estimates of a, b and c via the equations

$$a = e^{a_o}$$
$$b = e^{a_1} \qquad (44)$$
$$c = a_2$$

The sum of squares, (with weights w_i for more generality), is given by:

$$S = \sum_{1=1}^{n} w_i \left[Y_i - (a_o + a_1 X_{1i} + a_2 X_{2i}) \right]^2 \qquad (45)$$

$$\frac{\partial S}{\partial a_o} = -2 \sum_{i=1}^{n} w_i \left[Y_i - (a_o + a_1 X_{1i} + a_2 X_{2i}) \right]$$

$$\frac{\partial S}{\partial a_1} = -2 \sum_{i=1}^{n} w_i X_{1i} \left[Y_i - (a_o + a_1 X_{1i} + a_2 X_{2i}) \right]$$

$$\frac{\partial S}{\partial a_2} = -2 \sum_{i=1}^{n} w_i X_{2i} \left[Y_i - (a_o + a_1 X_{1i} + a_2 X_{2i}) \right]$$

For a minimum, the partial derivatives are zero, so that

$$\sum_{i=1}^{n} w_i Y_i = a_o \sum_{i=1}^{n} w_i + a_1 \sum_{i=1}^{n} w_i X_{1i} + a_2 \sum_{i=1}^{n} w_i X_{2i}$$

$$\sum_{i=1}^{n} w_i X_{1i} Y_i = a_o \sum_{i=1}^{n} w_i X_{1i} + a_1 \sum_{i=1}^{n} w_i X_{1i}^2 + a_2 \sum_{i=1}^{n} w_i X_{1i} X_{2i}$$

$$\sum_{i=1}^{n} w_i X_{2i} Y_i = a_o \sum_{i=1}^{n} w_i X_{2i} + a_1 \sum_{i=1}^{n} w_i X_{1i} X_{2i} + a_2 \sum_{i=1}^{n} w_i X_{2i}^2$$

IE that

$$\begin{bmatrix} \sum w & \sum wX_1 & \sum wX_2 \\ \sum wX_1 & \sum wX_1^2 & \sum wX_1X_2 \\ \sum wX_2 & \sum wX_1X_2 & \sum wX_2^2 \end{bmatrix} \begin{bmatrix} a_o \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum wY \\ \sum wX_1Y \\ \sum wX_2Y \end{bmatrix} \qquad (46)$$

Thus the complete procedure used to fit a curve of the form (40) may be outlined as follows:

(1) Perform the transformations on the given x_i and y_i values indicated by equations (43) to produce corresponding values of X_{1i}, X_{2i}, and Y_i.

(2) Compute the weighted sums (over i = 1,...,n) required as entries in the matrices of coefficients and constant terms in (46).

(3) Solve the linear system (46) for a_o, a_1, and a_2.

(4) Use (44) to obtain a, b and c estimates.

Ordinarily, the estimates obtained by this process would not coincide with the actual least squares estimates of a, b and c. In our applications, however, the number of data points was invariably equal to the number of parameters estimated (I.E. n=3), and in this case the above methodology produces a curve which exactly fits the three data points.

Figures 8 and 9 give plots of the B and C data tabulated at (27), and functions relating these parameters to the LR/CEP ratios. These functions were obtained by the following procedure:

For the C parameter, a curve of the form (40) was fit (by the process described above) to the first three data points (I.E. LR/CEP = .5, 1.0, 2.0), yielding

$$y = C_1 \, C_2^{\,x} \, x^{C_3} \qquad (47)$$

where:

$y = C$

$x = LR/CEP$

$C_1 = 7.45995$ (48)

$C_2 = .207535$

$C_3 = 2.42620$

Then, to insure correct asymptotic behavior, the 2nd, 3rd and 4th points were transformed slightly and also fit by a (40) curve, namely

$$y = C_4 \, C_5^{\,x} \, x^{C_6} \qquad (49)$$

where, in this case

$y = C-1$

$x = LR/CEP$

$C_4 = 5.39243$ (50)

$C_5 = .0930345$

$C_6 = 3.$[illegible]323

FIGURE 8

B PARAMETER VS. LR/CEP

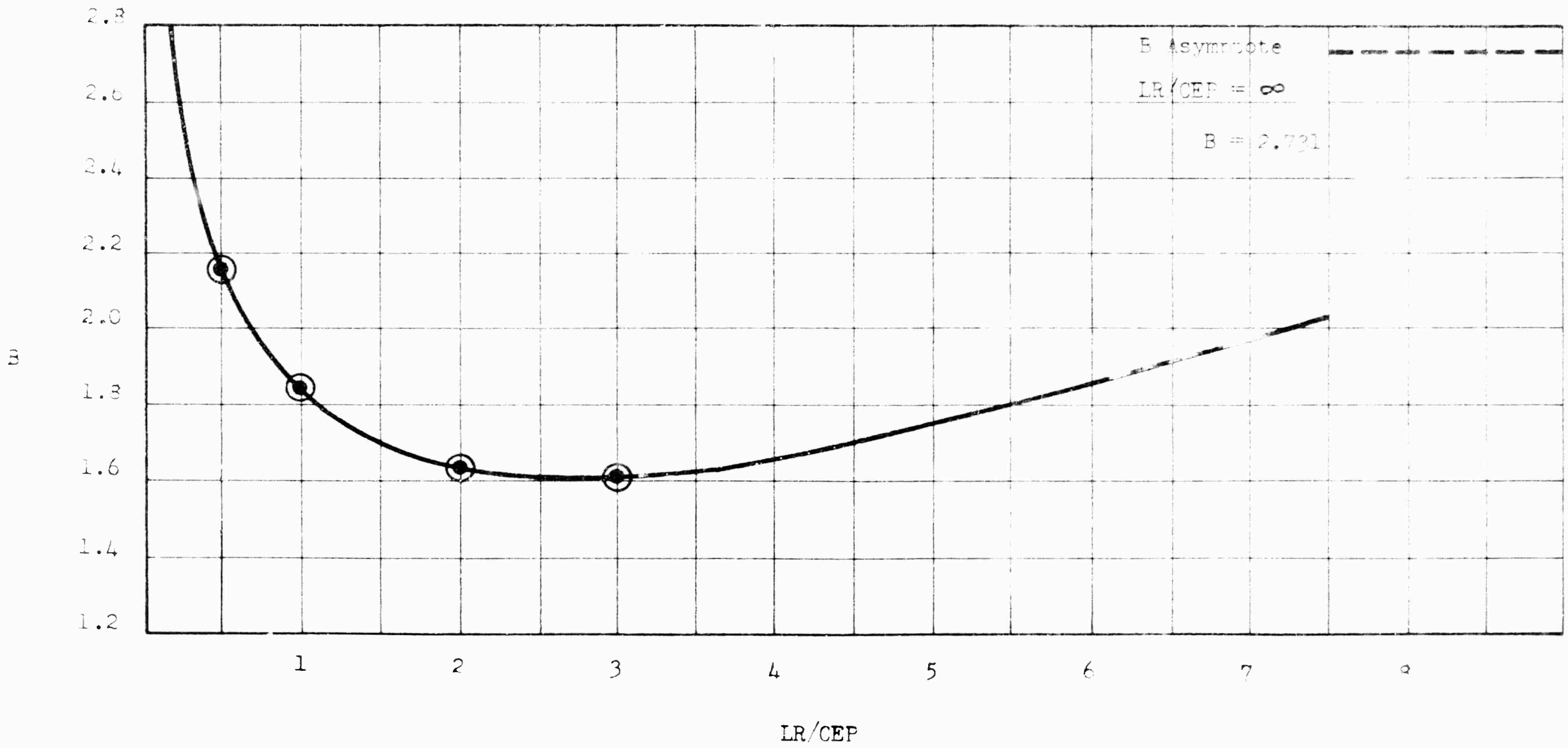
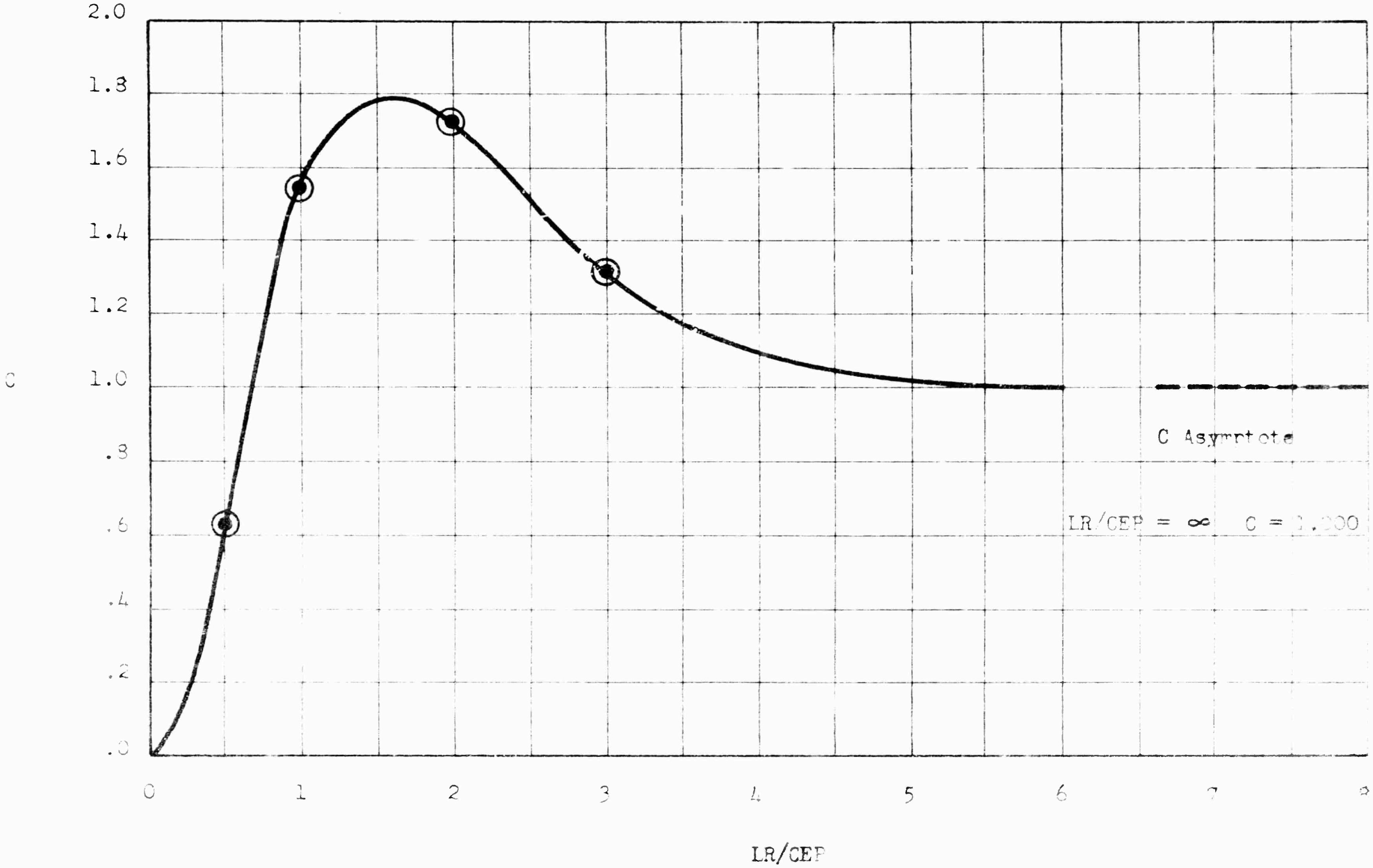


FIGURE 9

C PARAMETER VS. LR/CEP



Both (47) and (49) were compared in the overlap region [illegible] and a smoother overall fit was obtained (visually) using equation [illegible] there.

Thus, the following functional description of C was decided [illegible] is that plotted in Figure 9:

$$C = \begin{cases} C_1 \; C_2^{\,x} \; x^{C_3} & \text{for } 0 < x \leq 1 \\ 1 + C_4 \; C_5^{\,x} \; x^{C_6} & \text{for } 1 \leq x < \infty \end{cases} \qquad (51)$$

where x = LR/CEP, and the constants are given in (48) and (50).

The B parameter was treated similarly. The fit of equation (40) to the first three data points produced the expression:

$$y = C_7 \; C_8^{\,x} \; x^{C_9} \qquad (52)$$

where

$$\begin{aligned} y &= B \\ x &= LR/CEP \\ C_7 &= 1.68900 \\ C_8 &= 1.09218 \\ C_9 &= -.296980 \end{aligned} \qquad (53)$$

Again, a transformation of the 2nd, 3rd and 4th data points (LR/CEP = 1., 2., 3.) was necessary to insure that the fitted curve approached the proper asymptote. In this case the resulting function was:

$$y = C_{11} \; C_{12}^{\,x} \; x^{C_{13}} \qquad (54)$$

where

$$y = C_{10} - B$$
$$x = LR/CEP$$
$$C_{10} = B_{\infty} = 2.73109 \qquad (55)$$
$$C_{11} = 1.17109$$
$$C_{12} = .790655$$
$$C_{13} = .638752$$

As before, equations (52) and (54) were compared visually in the region of overlap ($1 \leq LR/CEP \leq 2$). In this case equation (52) proved more satisfactory. Thus, the following functional description of B is plotted in Figure 8:

$$B = \begin{cases} C_7 \, C_8^{\,x} \, x^{C_9} & \text{for } 0 < x \leq 2 \\ C_{10} - C_{11} \, C_{12}^{\,x} \, x^{C_{13}} & \text{for } 2 \leq x < \infty \end{cases} \qquad (56)$$

where x = LR/CEP, and the constants are those in (53) and (55). This completes the analysis performed to relate the parameter n in equation (1) to the two ratios LR/CEP and $(L/D)^2$. The significant equations are (56), (51), (18), (22) and (23).

As was mentioned earlier, $P_K(n)$ is uniquely determined by any two of the four parameters in equation (1). Note that, if there is any linear portion of the P_K function at all, its slope must be

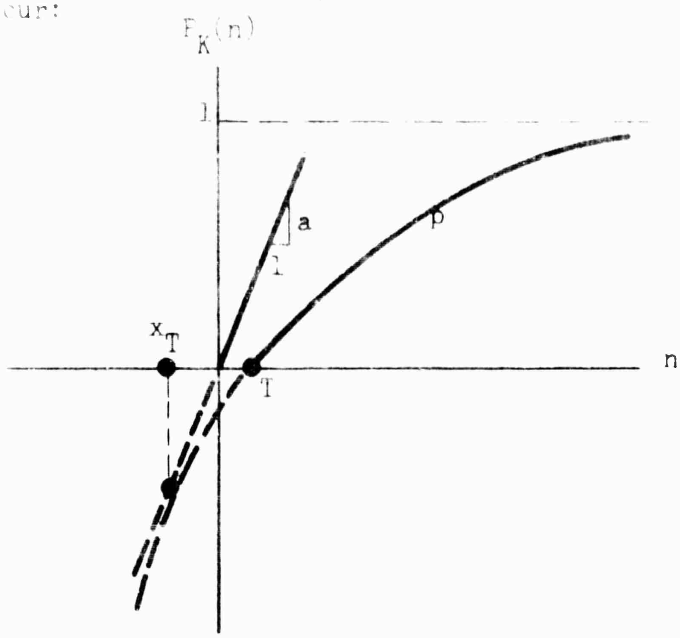
$$a = (L/D)^2 \qquad (57)$$

from physical considerations.

Thus, defining

$$c = -\ln(1-p) \tag{58}$$

for a value of p produced according to the process developed above, and using (57), the remaining two parameters (x_T and T) may be solved for directly by equations (13) and (14). Caution must be exercised, however, to insure that the mathematical possibility illustrated in the following sketch, does not occur:



That is, negative tangent points are not permitted.

From (13) and (58), the condition that $x_T \geq 0$ is equivalent to the condition $\quad a \leq -\ln(1-p)$. (59)

If, for a given pair of a and p values, condition (59) is not satisfied, then $P_K(n)$ has no linear portion and

$$x_T = 0$$
$$T = 0 \tag{60}$$

are the correct values for the remaining parameters of (1).

FIGURE 10

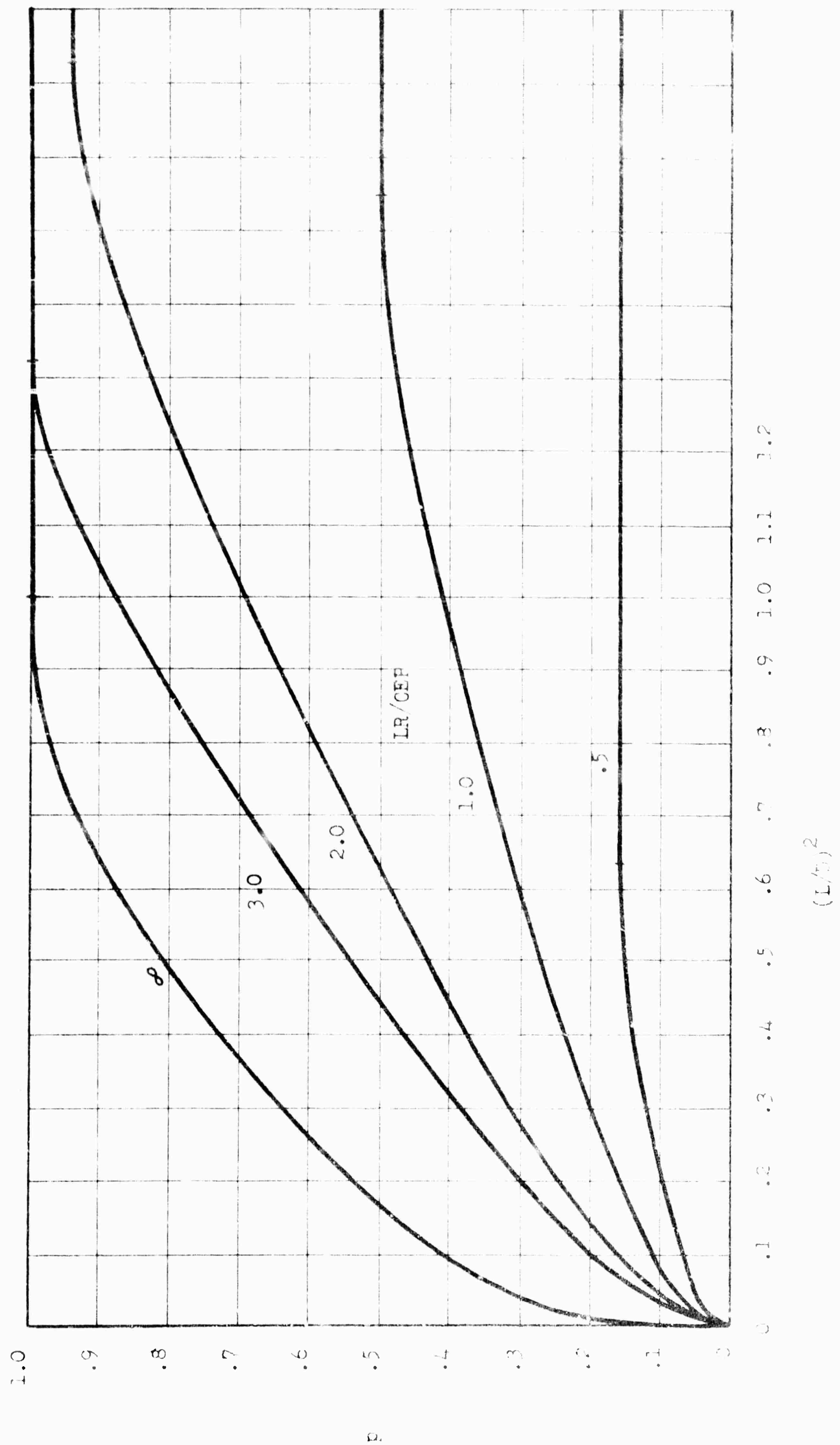


FIGURE 11

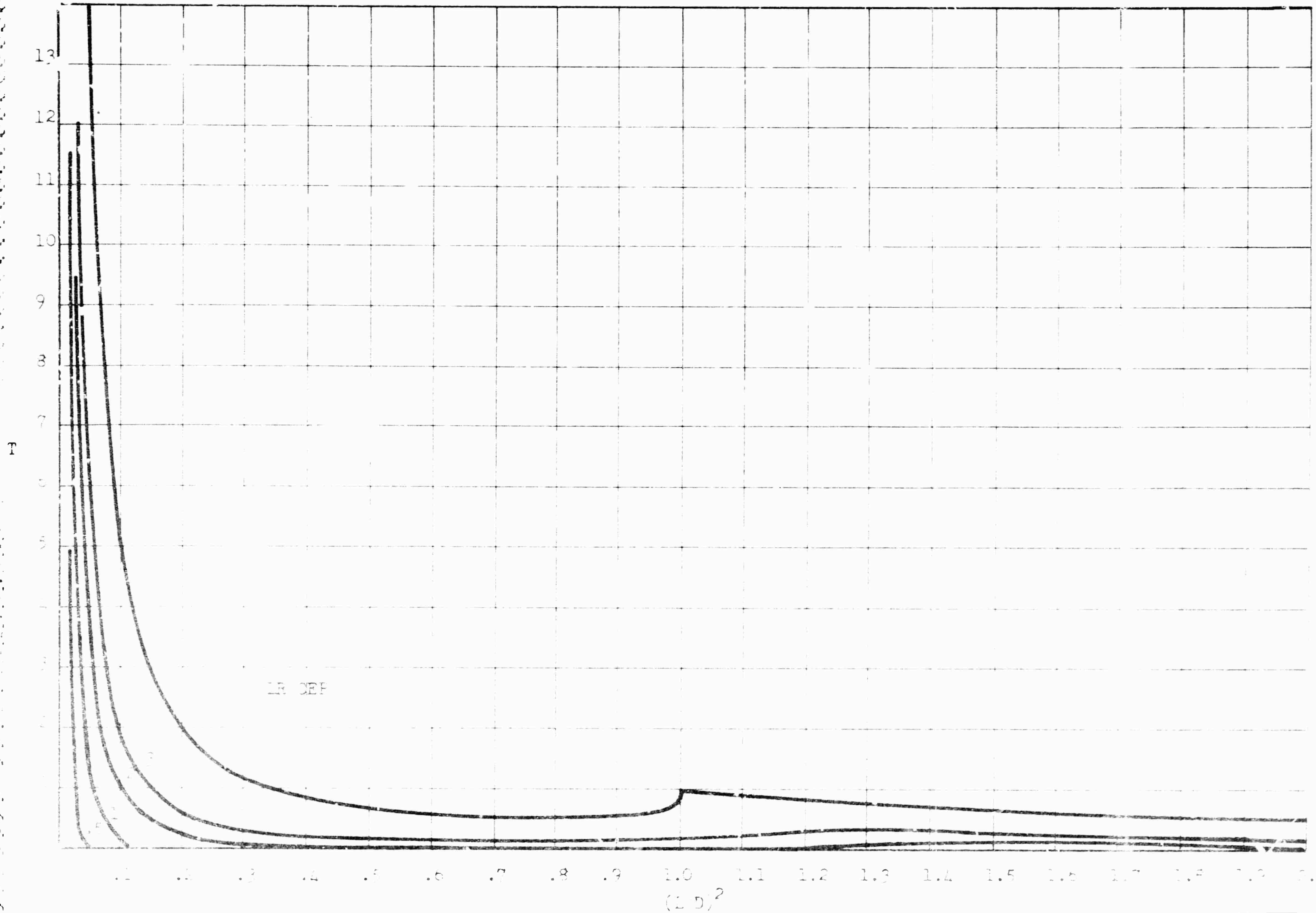


Figure 10 summarizes the analytical description of p as a function of LR/CEP and $(L/D)^2$. Figure 11 depicts the corresponding results for T, obtained in accordance with the above procedure.

Spot checks on the validity of the analysis are possible. Consider the two sample cases of Figure 1:

<u>Case (a)</u> $LR/CEP = 1.$, $(L/D)^2 = .9$

Figures 10 and 11 yield p = .39 and T = 0. Thus, in accordance with equation (1), the following tabular comparison results:

n_i	$P_K(n_i)$ (RAND)	$P_K(n_i)$ (SMOOTHED)
1	.39	.390
2	.64	.628
3	.77	.773
4	.85	.862
7	.93	.969

<u>Case (b)</u> $LR/CEP = \infty$, $(L/D)^2 = .2$

In this case, p = .535 and T = 1.95 (from Figures 10 and 11). Also, a = .2 (equation (57)). From (58), c = ln(.465) = - .766. Since condition (59) is satisfied, equation (13) yields $x_T = \frac{1}{.2} - \frac{1}{.766} = 5 - 1.3 = 3.7$.

Thus, the comparison is:

n_i	$P_K(n_i)$ (RAND)	$P_K(n_i)$ (SMOOTHED)
1	.20	.200
2	.40	.400
3	.60	.600
4	.77	.792
5	.87	.903
6	.96	.955
7	.999	.979

These results are plotted in Figure 12.

Three additional graphs are presented as evidence that the analysis produces proper trending in the kill probabilities as CEP improves. In Figures 13, 14 and 15, the kill probabilities for one, two and ten weapons (respectively) are plotted vs. LR/CEP for constant values of $(L/D)^2$. It is clear from Figure 13 that for $(L/D)^2 \leq 1$, $P_K(1) \rightarrow (L/D)^2$ as CEP $\rightarrow 0$, an intuitively pleasant result.

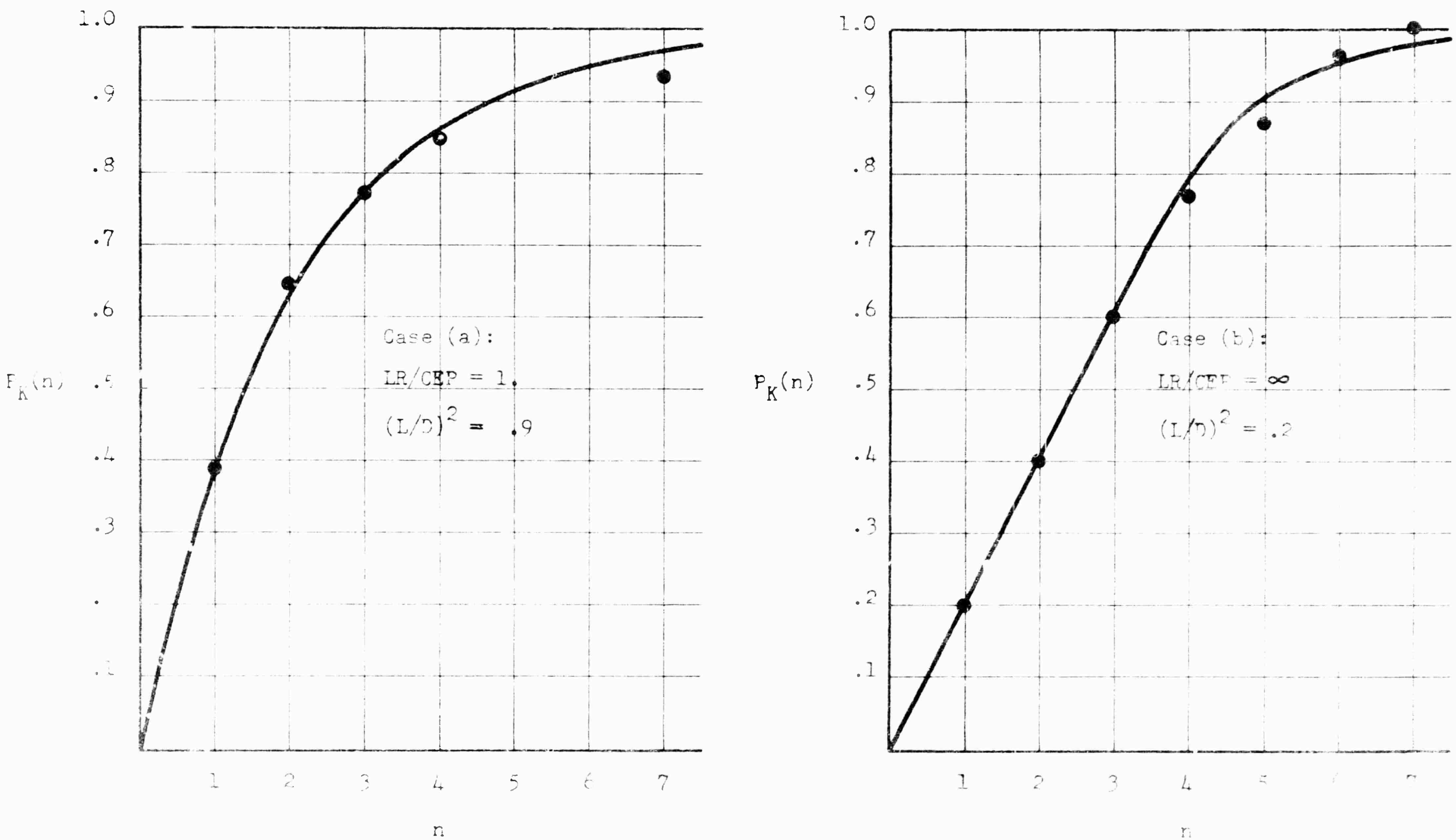


FIGURE 12 EXAMPLES OF THE FINAL FIT

FIGURE 13

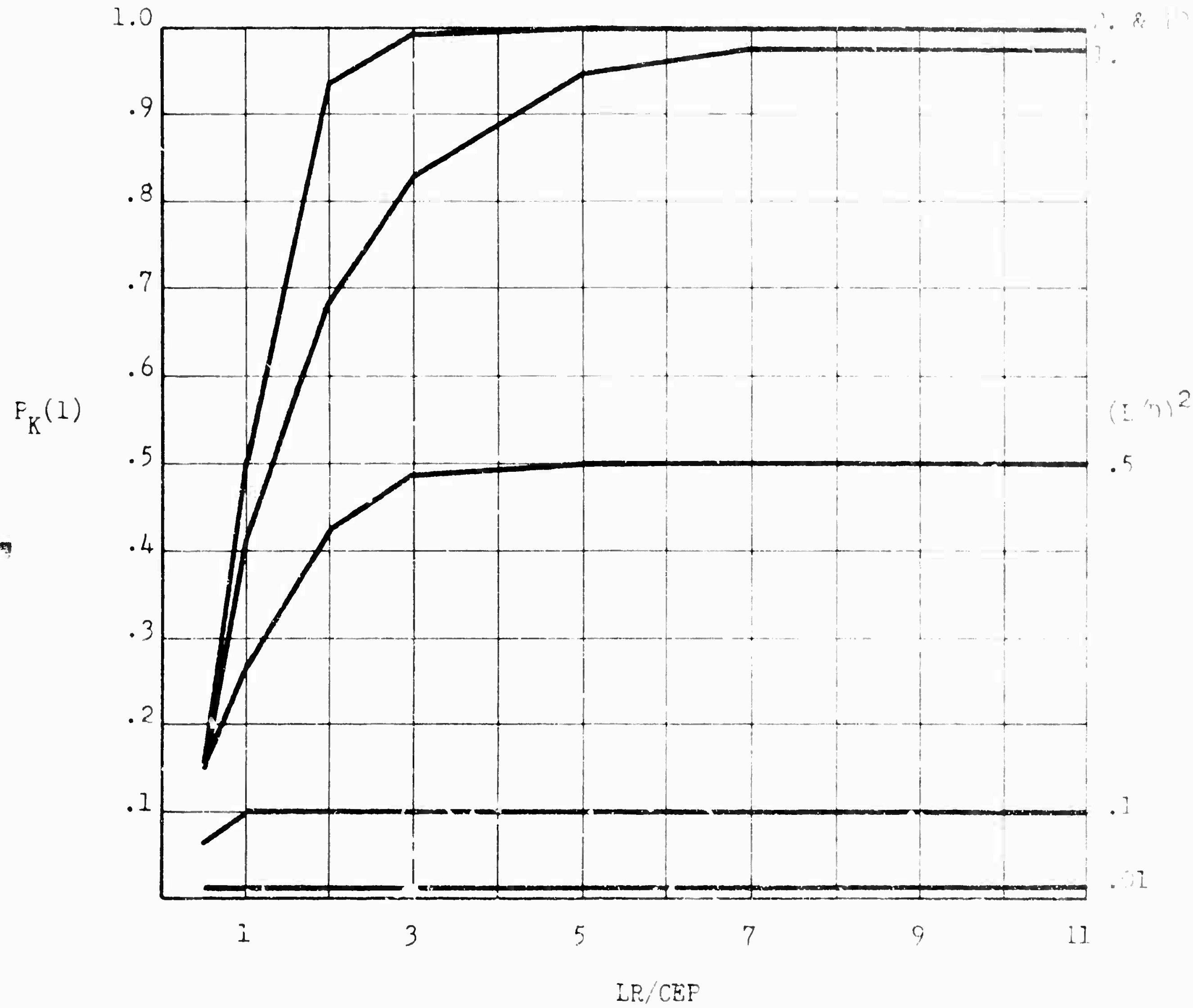


FIGURE 14

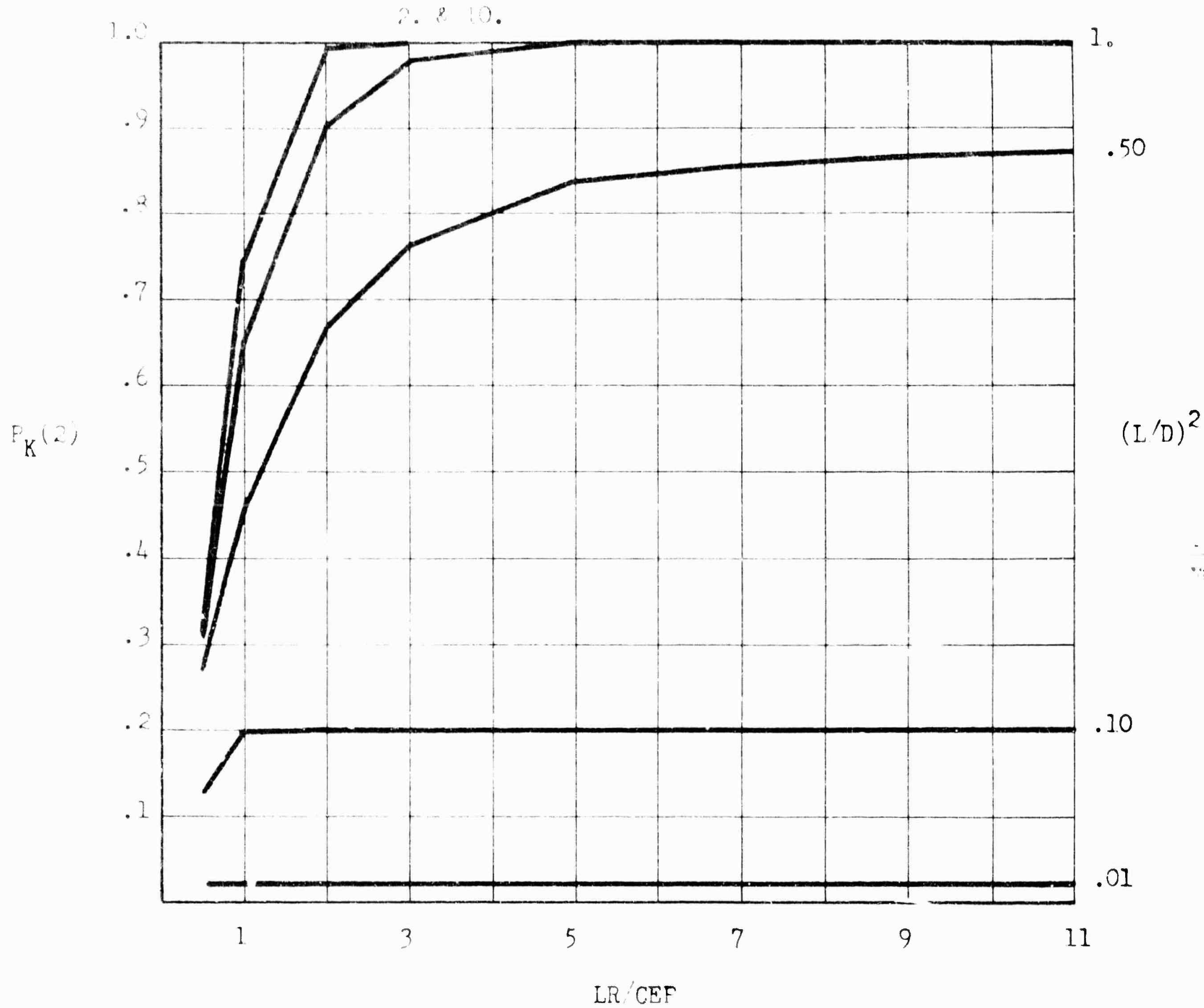
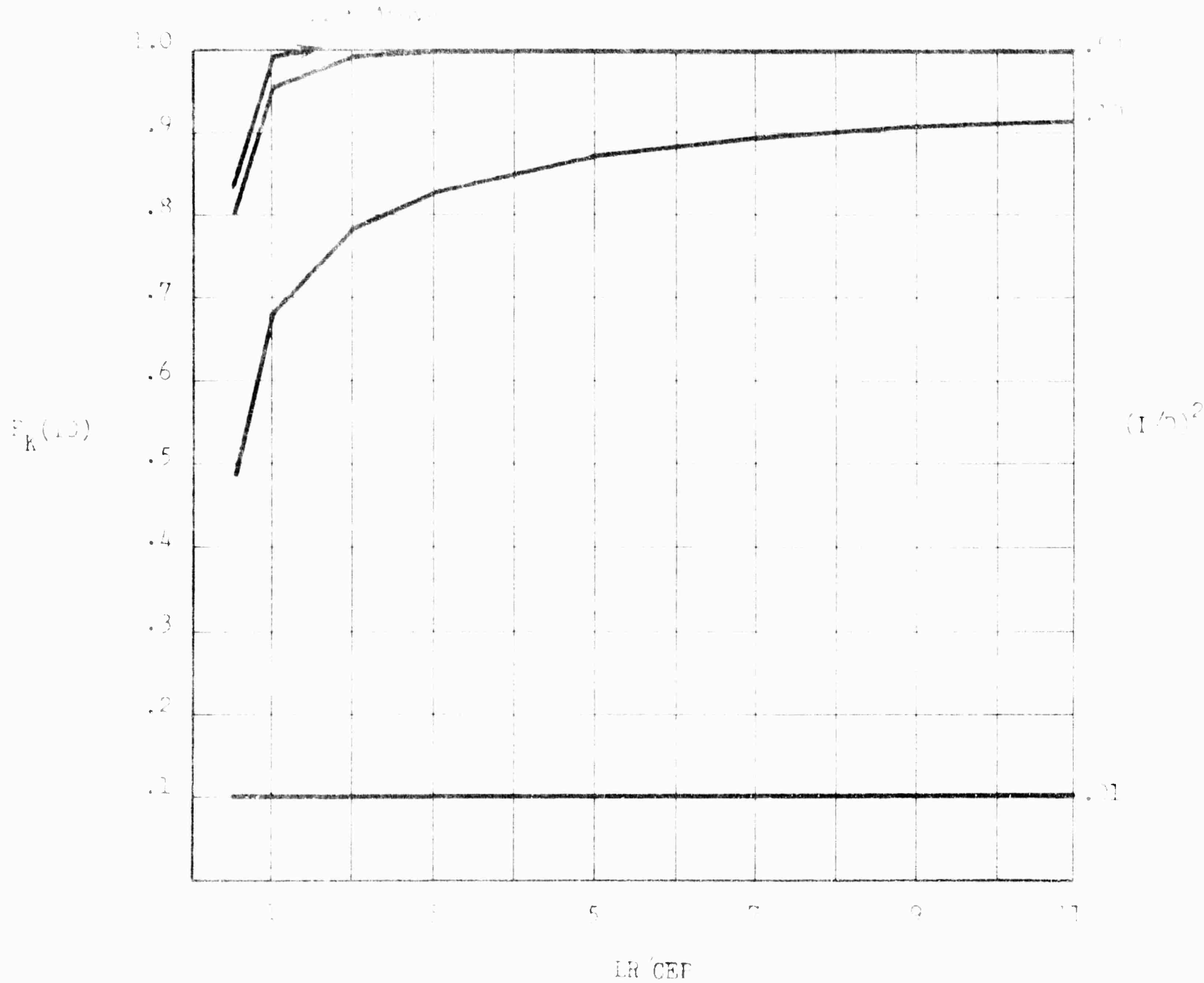


FIGURE 15



APPENDIX B: COMPARISON OF THREE METHODS OF PRODUCING KILL PROBABILITIES FOR THE IMPERFECT RELIABILITY, NO DEFENSE CASE

In this section, the results of three runs of the IBM 1130 program are presented. The input data was that tabulated here:

VARIABLE	VALUE INPUT		
	Case 1	Case 2	Case 3
Reliability	.9	.9	.9
CEP (n. mi)	.5	.5	.5
Yield (MT)	1.0	1.0	1.0
No. of Decoys	2	2	2
Hardness (psi)	7.0	7.0	7.0
Area (n. mi.^{2})	50.0	50.0	50.0
Prob. (acquis.)	.0000	.0001	.0001
Prob. (discrim.)	.875	.875	.875
Prob. (intercept)	.9	.9	.9
No. of AMM's	15	15	15
NDØC	1	-1	1

The only differences in the three cases are the NDØC and probability of acquisition inputs.

In all cases, the initial values produced for the parameters of equation (B-12) were (for perfectly reliable weapons against an undefended target):

$$= .81974148$$

$$T = .42907249 \qquad (1)$$

$$W_T = 1.0184080$$

$$a = .62419808$$

In case 1, since $p_A = 0$, the program computed modified values for these parameters directly via the short-cut method described in the "Reliability" section above, yielding:

$$p = .73776722$$

$$T = .38433289 \qquad (2)$$

$$W_T = 1.0329694$$

$$a = .56177818$$

In case 2, since $p_A > 0$ and NDØC = -1 were input, the program produced several $P_K^*(W)$ values according to equation (B-18), the basic simultaneous strike case equation.

In case 3, since $p_A > 0$ and NDØC = 1 were specified, sequential strike was assumed and equations (B-13, (B-14), and (B-15) were evaluated to produce the $P_K^*(W)$ values.

Due to the small acquisition probability, the influence of defense on the kill probabilities resulting from cases 2 and 3 is negligible. Thus, these values provide a standard against which the short-cut CØMPK methodology (case 1) may be evaluated.

The kill probabilities for case 1, $P_K'(W)$, are obtained by using equation (B-12) with (2) as values for the parameters.

Thus, the desired comparison is provided by the following tabulation of program output:

W	CASE 1 $P'_K(W)$	CASE 2 $P^*_K(W)$	CASE 3 $P^{**}_K(W)$
1	.5617781	.5617220	.5617276
2	.8849750	.8673969	.8674255
3	.9698367	.9634487	.9634659
4	.9920902	.9902347	.9902421
5	.9979258	.9974203	.9974231
6	.9994561	.9993211	.9993224

APPENDIX C: PRIM-READ ASSIGNMENT DOCTRINE

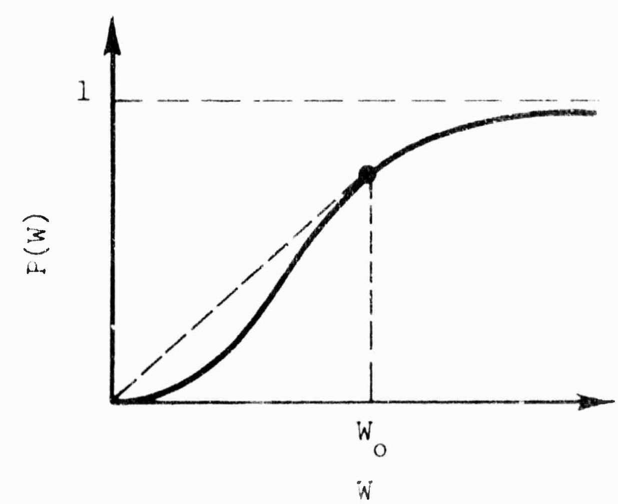
Assuming a uniform (not necessarily integer) allocation of the N_E [illegible] reserved for whds, the most favorable result achievable by the defense against a salvo of W whds is represented by the equation

$$P(W) = 1 - \left[1 - (1 - p_I)^{\frac{N_E}{W}} \right]^{W}, \qquad (1)$$

(arrow to p_I: prob. of intercept)

where P(W) is the probability of at least one penetrating whd. However, in the case where the defense does not know the extent of the attack (sequential case), it would be desirable to devise an interceptor assignment policy which is in some sense optimal.

Equation (1) typically gives rise to an S-shaped curve like that in the ff. sketch:



For any size salvo, the offense's return per whd is given by P(W)/W, assuming that the defense allocates uniformly. If the defense chooses any other allocation, the return/whd is greater.

The maximum return/whd attainable by the offense against an omniscient defense is represented by W_o in the sketch, the point at which $P(W)/W$ is maximized. The Prim-Read firing doctrine is based on the assumption that, in a sequential attack, the offense will try to achieve at least $P(W_o)/W_o$. The defense's approach is to let the offense have the maximum average return achievable against an omniscient defense (but no more than this) until defense exhaustion occurs. Thus the defense wants to allocate such that $P(W)$ becomes linear (with slope $c = P(W_o)/W_o$ up to the point of defense exhaustion. Or, stating the defensive problem mathematically, choose n_j such that

$$\frac{P(W)}{W} = \left[1 - \prod_{j=1}^{W} \left[1 - (1 - p_I)^{n_j}\right]\right] / W$$

$$= c$$

where n_j is the number of AMM's assigned to the $j^{\underline{th}}$ whd. Thus,

$$\prod_{j=1}^{W} \left[1 - (1 - p_I)^{n_j}\right] = 1 - cW.$$

The n_j's are found as follows:

$W = 1$:

$$1 - (1 - p_I)^{n_1} = 1 - c$$

W = 2:

$$\left[1 - (1 - p_I)^{n_1}\right]\left[1 - (1 - p_I)^{n_2}\right] = 1 - 2c$$

$$\left[1 - c\right]\left[1 - (1 - p_I)^{n_2}\right] = 1 - 2c$$

$$1 - (1 - p_I)^{n_2} = \frac{1 - 2c}{1 - c}$$

W = 3:

$$\left[1 - (1 - p_I)^{n_1}\right]\left[1 - (1 - p_I)^{n_2}\right]\left[1 - (1 - p_I)^{n_3}\right] = 1 - 3c$$

$$\left[1 - c\right]\left[\frac{1 - 2c}{1 - c}\right]\left[1 - (1 - p_I)^{n_3}\right] = 1 - 3c$$

$$1 - (1 - p_I)^{n_3} = \frac{1 - 3c}{1 - 2c}$$

In general,

$$1 - (1 - p_I)^{n_j} = \frac{1 - jc}{1 - (j-1)c}$$

$$\frac{c}{1 - (j-1)c} = (1 - p_I)^{n_j}$$

$$n_j = \frac{\ln c - \ln\left[1 - (j-1)c\right]}{\ln (1 - p_I)} \qquad (2)$$

In COMPK, the slope $\underline{c}$ is found by a numerical process, and then equation (2) is used to produce the n_j's up to the point of defense exhaustion.

Besides this formulation of the Prim-Read doctrine, another one (the one in current usage) that allows for non-perfect acquisition probabilities has been developed. This other formulation has the following characteristics:

1. The probability of intercept used in determining the n_j's is modified to take probability of acquisition into account. (I.E. $p_I' = p_A \cdot p_I$ is used in equation (2)).

2. Also, the n_j's so determined are modified to give an <u>integer</u> AMM assignment.

REFERENCES

1. Everett, H., "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," Operations Research, 11, No. 3 (May-June 1963), 399-417.

2. Gomory, R. E., "An Algorithm for Integer Solutions to Linear Programs," Recent Advances in Mathematical Programming, McGraw-Hill, New York, 1963.

3. Gilmore, P. C. and Gomory, R. E., "A Linear Programming Approach to the Cutting Stock Problem-Part II," Operations Research, 11, No. 6 (Nov.-Dec. 1963), 867.

4. Brooks, R. and Geoffrion, A., Finding Everett's Lagrange Multipliers By Linear Programming, P-3315, The Rand Corporation, Santa Monica, California, Feb., 1966.

5. Dantzig, G., Linear Programming and Extensions, Princeton University Press, 1963.

6. R. J. Galiano, Defense Models III, Preferential Defense, Lambda Corporation, March 1967.

7. Von Neumann and Morgenstern, Theory of Games and Economic Behavior, Princeton University Press, 1953.

8. A. I. Penn, Defense Models VIII. A Lagrangian Approach to Constrained Matrix Games, Lambda Corporation, September 1967.

9. G. W. Brown, Some Notes on Computation of Games Solutions, The Rand Corporation, April 1949.

10. R. J. Galiano, Defense Models VI, Strategies of Preferential Defense in Combination With Fixed Terminal Defense, Lambda Corporation, March 1968.

11. G. E. Pugh, Lagrange Multipliers and the Optimal Allocation of Defense Resources, JORSA, V. 12, No. 4, JUL.-AUG. 1964, pp. 543-567.

REFERENCES (Cont.)

12. L. R. Ford, Jr. and D. R. Fulkerson, Flows In Networks, Princeton University Press 1962.

13. J. F. Bozovich, An Investigation of a Non-Linear Programming Application Within AEM, Martin Marietta Corp., Denver Division, Denver, Colo., Dec. 1970.

14. Shareshian, R., Branch and Bound Mixed Integer Programming (BBMIP), IBM S/360, General Program Library, 360 D-15.2.005, New York Scientific Center, April 1967.

15. Land, A. H. and Doig, A. G., "An Automatic Method of Solving Discrete Programming Problems," Econometrica, Vol. 28, No. 2 (July 1960), 497-520.

16. Geoffrion, A. M., "An Improved Implicit Enumeration Approach For Integer Programming," Operations Research, Vol. 17, No. 3 (May-June 1969), 437-454.

17. Geoffrion. A. M., "Integer Programming By Implicit Enumeration and Balas' Method," The Rand Corporation, RM-4783-PR, Santa Monica, Calif., Feb. 1966.

18. Benders, J. F., "Partitioning Procedures For Solving Mixed Variables Programming Problems," Numerische Mathematik, 4, 1962, pp 238-252.

19. Lasdon, L. S., Optimization Theory For Large Systems, The MacMillan Co., New York, New York, 1970.

20. Ford, L. R. and Jago, W. H., "A Subtractive Area Defense Model," General Research Corp., Report IMR-995, Santa Barbara, Calif., 1968.

21. Groover, P. L., "A Report On: Task 2.1.1 Nth Country Improvements", Contract DCA 100-72-C-0028, Martin Marietta Corporation, Denver Division, Denver, Colorado, January 1973.

22. Battilega, John A. and Wm. L. Cotsworth , OSAGE Model Description and User's Manual (Version 1), Martin Marietta Corporation, Denver, Colorado, 1972.

23. Hirshfeld, Davis S, "Generalized Upper Bounding" paper presented to the SHARE Mathematical Programming Project, Rockville, Maryland, 1970.

24. Battilega, J. A., Computational Improvements to Bender's Decomposition for Generalized Fixed Charge Problems, PhD Dissertation, Oregon State University, Corvallis, Oregon, 1973.

25. Dantzig, George B, and R. M. Van Slyke, "Generalized Upper Bounding Techniques", Journal of Computer and System Sciences, Vol 1, pp 213-226, 1967.

26. Orchard-Hays, William, Advanced Linear Programming and Computing Techniques. New York, McGraw-Hill, 1968.

27. Beale, E.M.L., "Mathematical Programming Algorithms", in: Progress in Operations Research, ed. by Julius Aronofsky, Vol 3, New York, Wiley, 1969.

28. Sakarovitch, M., and R. Saigal, "An Extension of Generalized Upper Bounding Techniques for Structured Linear Programs", Journal for the Society of Industrial and Applied Mathematics, Vol 15, pp 906-914, 1967.

29. Hartman, James K., and Lean Lasdon, "A Generalized Upper Bounding Method for Doubly Coupled Linear Programs," Naval Research Logistics Quarterly, Vol 17, pp 411-429, 1970.

30. Grigoriadis, Michael D., "A Dual Generalized Upper Bounding Technique," Management Science, Vol 17, pp 269-284, 1971.

31. Lasdon, L.S., and Terjung, R.C., "An Efficient Algorithm for Multi-Item Scheduling," Operations Research, Vol 19, pp 946-969, 1971.

32. Simmonard, Michel, Linear Programming, Englewood Cliffs, New Jersey, Prentice-Hall, 1966.

33. Groover, P.L. "A Report on: Task (a-2): Defense Suppression", Contract DCA 100-71-C-0032, Martin Marietta Corporation, Denver, Colorado, March, 1972.

34. Naval Operations Analysis, Naval Institute Press, Annapolis, Maryland, 1968.

35. Bean, B.R., and G. D. Thayer, "Models of the Atmospheric Radio Refractive Index," Proceedings of the Institute of Radio Engineers, pp 740-753, 1959.