UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER P-1111	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) REVISED OPTSA MODEL Volume 1: Methodology		5. TYPE OF REPORT & PERIOD COVERED Final
		6. PERFORMING ORG. REPORT NUMBER P-1111
7. AUTHOR(s) Lowell Bruce Anderson, Jerome Bracken, Eleanor L. Schwartz		8. CONTRACT OR GRANT NUMBER(s) DAHC15 73 C 0200
9. PERFORMING ORGANIZATION NAME AND ADDRESS Institute for Defense Analyses, Program Analysis Division, 400 Army-Navy Drive, Arlington, Va. 22202		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS T-229
11. CONTROLLING OFFICE NAME AND ADDRESS Weapons Systems Evaluation Group, 400 Army-Navy Drive, Arlington, Va. 22202		12. REPORT DATE June 1975
		13. NUMBER OF PAGES 99
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Further requests for this document must be approved by Director, Weapons Systems Evaluation Group, Arlington, Virginia 22202.

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

18. SUPPLEMENTARY NOTES

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

Aircraft Allocation to Missions, Multi-Stage Game, Game Theory, General Purpose Forces, Tactical Air Forces

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

This paper describes and documents an improved version of the optimal sortie allocation model (OPTSA) previously presented in IDA Papers P-992 and P-993, published in December 1973. OPTSA is a model for computing allocations of general purpose aircraft to combat air support, airbase attack, and intercept missions. The mathematical problem is a two-side, zero-sum, multi-stage game with simultaneous moves at each

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. continued

stage. The revised OPTSA model includes a substantially improved game-solving procedure and a more detailed simulation of warfare between the opposing sides.

PAPER P-1111

REVISED OPTSA MODEL

Volume 1: Methodology

Lowell Bruce Anderson
Jerome Bracken
Eleanor L. Schwartz

September 1975

IDA

INSTITUTE FOR DEFENSE ANALYSES
PROGRAM ANALYSIS DIVISION

IDA Log No. HQ 75-17205
Copy 28 of 67 copies

PAPER P-1111

REVISED OPTSA MODEL

Volume 1: Methodology

Lowell Bruce Anderson
Jerome Bracken
Eleanor L. Schwartz

September 1975

INSTITUTE FOR DEFENSE ANALYSES
PROGRAM ANALYSIS DIVISION
400 Army-Navy Drive, Arlington, Virginia 22202

Contract DAHC15 73 C 0200
Task Order T-229

CONTENTS OF VOLUME 1

FIGURES

TABLE

PREFACE

One widely used philosophy of simulating warfare that involves tactical aircraft has as its central concept the assignment of aircraft to missions on a game-theoretic basis. Alternative philosophies of simulation attempt either to model the decision processes of a commander or simply to require the analyst to supply mission assignments as input. Since results are very sensitive to mission assignment, this issue is important. Recently, game-theoretic models that go well beyond those discussed in the operations research literature in the 1950s and '60s have been implemented on computers.

Computer-implemented models that should be noted are the TAC CONTENDER model developed by the U.S. Air Force (Ref. [3]), the DYGAM procedure developed by Control Analysis Corporation (Refs. [7] and [8]), the sequential-game procedures included in the Lulejian ground-air model [11], and the OPTSA model treated in this paper. Also, KETRON, Inc., is currently developing a model for the Arms Control and Disarmament Agency.

The present paper describes a new OPTSA model, which is a revised version of the OPTSA II model described in References [4] and [9]. The revisions focus on (1) speeding up the game-solving procedure, which enables an order-of-magnitude reduction in computation time, and (2) elaborating the assessment procedure, which enables additional aspects of tactical air warfare to be represented.

It is appropriate to note the contributions of several individuals to our work. The development of TAC CONTENDER motivated all the other recent models. Lt. Gen. Glenn Kent, USAF(Ret.), was the principal proponent; Louis Finch, Leon Goodson, and Scott Meyer developed the model. Our colleagues--Jerry Blankenship, James Falk, and Alan Karr--made significant contributions to the development and understanding of the whole class of sequential game models. James Falk, Louis Finch, and Frederic Miercort reviewed this paper.

Finally, it should be noted that the revised OPTSA model has been used in two studies. An interim version was used in a study that examined quantity-quality trade-offs of general-purpose aircraft [6]. The version documented in this paper was used in a study analyzing the cost and effectiveness of NATO aircraft shelters and air defenses [5].

Chapter I

FEATURES OF THE REVISED OPTSA MODEL

OPTSA determines percentage assignments of general-purpose aircraft to missions by period, where assessments of occurrences during the war are performed for certain numbers of days within each period.[1] The overall model is a zero-sum, two-person sequential game, with simultaneous moves each day. There are two input lists of feasible assignments of Blue and Red general-purpose aircraft to missions; the solution to the game provides strategies for choosing assignments from these lists to optimize a desired measure of effectiveness. A choice of aircraft allocation is made by each side from its list of assignments at the beginning of each period. The choice may be made in a randomized manner and can depend on what choices each side has made in previous periods. However, once an assignment is chosen, it must be played all the days in the period. The number and extent of the periods is the same for both Blue and Red.

The measure of effectiveness, which becomes a payoff entry in the game, is found by fighting a ground-air war. The three missions that the aircraft may fly are combat air support (CAS), airbase attack (ABA), and intercept (INT). Aircraft may be of four types: general-purpose (GP), special-purpose (SP) CAS, SP-ABA, and SP-INT. The model optimally assigns only the GP

---

[1]Hereinafter, "OPTSA" refers to the model originally called OPTSA II. "Day" is used throughout; however, no specifically daily features (e.g., light and dark) are simulated. A day can represent any kind of subperiod, but each subperiod must have the same characteristics.

aircraft; assignments of the SP aircraft are fixed. Aircraft assigned to INT destroy enemy CAS and ABA attacking aircraft before they perform their attack missions. Aircraft assigned to ABA destroy enemy aircraft on the ground. Therefore, there is a trade-off between assigning one's aircraft to attack the enemy battlefield or airbase versus preventing the enemy aircraft from attacking. There are three types of ground units on each side, differentiated only by firepower per unit. Total firepower (consisting of firepower contributed by ground units plus firepower contributed by CAS) is used in forming force ratios for calculating casualties to ground forces and FEBA movement.

In the original model, three measures of effectiveness (MOEs) could be used:

(1) FEBA position.

(2) Cumulative Blue minus Red total (ground plus air) firepower.

(3) Cumulative Blue minus Red air firepower.

These MOEs are all available in the revised model. In addition, there are two new MOEs, explained in detail in Chapter III of this volume (below):

(4) Weighted sum of surviving Blue minus surviving Red aircraft (by type)--using an input set of weights.

(5) Generalized air measure, involving the difference between weighted sums of Blue and Red cumulative air firepower, surviving aircraft (by type), and levels of quick reaction alert (QRA) aircraft--using an input set of weights.

Though all these MOEs can be calculated for any specified day of the war, the last day of the war is usually used. As mentioned in the original model description [4], the same input data optimized on different MOEs can yield widely different optimal strategies.

Table 1 summarizes the important differences between the game structures of the original and the revised models. The nature of adaptive, nonadaptive, and behavioral games is described in References [4] and [9]; the original model's solution procedure, in Reference [4]. The solution procedure used in the revised model is described in detail in Chapter II of this volume (below). A main feature of behavioral games is that the game value of a matrix game at stage k becomes a payoff entry in a matrix game at stage k-1. The most interesting difference between the old and new models is that, in the former, all payoff entries are computed and then all matrix games are solved; in the latter, only those matrix games needed to generate a payoff entry for a previous-stage matrix game are solved; a payoff entry for a matrix game at any stage is generated only if there is a reasonable possibility that the corresponding strategies are active in the solution, and the computation of payoff entries and solution of matrix games are intermingled. Also, when a payoff entry must be computed, the number of daily assessments performed is kept as small as possible by computing assessments period by period.

The reason for developing the new game-solving method is that the computer time of OPTSA's assessment routine is practically all spent in computing payoff entries; hence, the running time of the model is essentially equal to the number of daily assessments computed times the time per daily assessment. Increasing the complexity and richness of the assessment routine results in a somewhat longer time per daily assessment. Therefore, to prevent the total running time from increasing, the number of daily assessments computed is reduced. It is not clear exactly how these two opposing factors together affect the running time. (Some of the computational experience with the model to date is discussed in Chapter IV, below.)

Table 1. REVISIONS TO OPTSA GAME STRUCTURE

Original Model	Revised Model
All payoff entries computed, regardless of whether they are needed to solve the game.	Most payoff entries not needed to determine a game solution are not computed.
Combat simulation in first period repeated for computation of all payoff entries having the same first-period allocation.	Assessments performed only for period in which a game is being solved; results from previous periods are stored for later use.
One subroutine "GAME" to set up games and ask for appropriate payoff entries and one subroutine "SIMPLEX" to solve an arbitrary matrix game.	Three subroutines "SIMPL1," "SIMPL2," "SIMPL3," which set up, ask for payoff entries to, and solve a game at stage 1, 2, or 3 (resp.).
Must be exactly three decision periods.	There can be one, two, or three decision periods: a two-stage game is solved by using routines SIMPL2 and SIMPL3; a one-stage game, by SIMPL3 alone.
Notation seemed to allow Red and Blue to make allocation decisions on different days.	Notation reflects fact that, to preserve theoretical validity of model, Blue and Red must make allocation decisions simultaneously at the beginning of each period.
Extremely long computer printout.	Variety of printout options available, including optimal second- or third-period *strategies* only, without payoff matrices.
Long running time.	Somewhat shorter running time (much shorter game solution, but expanded simulation).
Three MOEs available for payoff entries.	Five basic MOEs; by varying a set of input weights, a wide variety of combat measures can be used.

Figure 1 lists the new features of the assessment routine, which are incorporated into each daily assessment. SAMs and quick-reaction-alert (QRA) aircraft have been introduced; more detailed and accurate attrition equations have been developed; and aircraft shelters can be destroyed by being bombed from the air. There are two alternative ways of determining attrition in the air-to-air war. In the air-to-ground interaction, four different methods of ABA are possible for each side. These and the many other added features allow a much more accurate combat simulation. However, the decision variables in the model remain the proportions of GP aircraft assigned to the three missions; the other data are all input. (The algebra of the assessment procedure is described in the appendix to this volume; the revisions and additions to it are explained in detail, with somewhat different notation, in Chapter III, below.)

New features of the OPTSA assessment routine are listed in the order in which they occur in the assessment routine. Unless specifically noted, they apply for both sides (i.e., there can be separate numbers for Blue and Red).

Setup at Beginning of Day

New inventory of aircraft shelters computed at beginning of each day.

Quick Reaction Alert (QRA) aircraft.

A portion of the GP aircraft need not be assigned to missions but may sit home on the airbase, if the proportions in an allocation sum to less than 1.

Sortie rates by aircraft type (GP/SP) and mission.

Sortie rates can change once during war on an input day.

Air-To-Air Interaction

A number of identical air-to-air combat regions can be played, each side.

Two alternative methods of determining attrition--*not* for each side individually (i.e., a method is chosen and that method is used to compute casualties to both sides).

Both detection and kill parameters are used. They can vary with shooter and target type (GP/SP) and mission; averaging occurs only over target type.

Binomial (not exponential) attrition equations are used (Equation (17´), Ref. [10]).

The following quantities are computed in the second attrition method: attacking aircraft engaged by interceptors, attacking aircraft that fly back to their own side, aircraft that do not fly that day.

Ground-To-Air Interaction

A ground-to-air interaction is played.

Four input parameters for each side representing proportions of GP-CAS, GP-ABA, SP-CAS, and SP-ABA aircraft (resp.) destroyed by enemy SAMs.

Planes on INT are not vulnerable to SAMs.

(continued on next page)

Figure 1. BRIEF LISTING OF NEW FEATURES OF OPTSA ASSESSMENT ROUTINE

Air-To-Ground Interaction (ABA)

An input number of notional but identical airbases can be played, each side.

An input fraction of aircraft shelters hit by a munition are destroyed.

The "shell game" is used throughout (i.e., an occupied aircraft shelter is indistinguishable from an empty one).

QRA aircraft are sheltered before anything else.

Option exists not to shelter Red SP-ABA aircraft.

Other aircraft are sheltered proportionally by kind of aircraft.

Only an input fraction of the aircraft are on base. The remaining aircraft are out flying missions. However, the assessment procedure first assigns aircraft to shelters, then reduces both the sheltered (excluding QRA) and nonsheltered aircraft by the input fraction. Therefore, there are practically always some empty shelters.

The fraction of aircraft on base can change when the sortie rates do.

Each ABA sortie makes an input number of passes. The number can be different for GP- versus SP-ABA aircraft.

Each side has four different modes of attacking the enemy airbase (unlike the air-to-air interaction, one side can use one mode and the other another):

(1) Point fire, nonsheltered aircraft on parking areas; attackers shoot at what they detect.

(2) Point fire, nonsheltered aircraft on parking areas; attackers allocated in advance (by an internal optimization) to sheltered or nonsheltered aircraft.

(3) Point fire, sheltered and nonsheltered aircraft on parking areas; attackers shoot at what they detect.

(4) Area fire. An internal optimization is done to determine the proportion of attack planes that load up with anti-shelter vs. anti-nonsheltered-aircraft munitions.

In the point-fire attack modes, binomial (not exponential) attrition equations are used. Separate parameters for detection and kill are input to the model. The input parameters depend on shooter type (GP/SP) and target type

(concluded on next page)

Figure 1 (continued)

(sheltered/nonsheltered). Average detection and kill parameters are formed by averaging over shooter type and used as input to the attrition equations.

The FEBA advance function is forced to be logarithmically symmetrical. If the force ratio x is less than 1, the value F(1/x) is computed by interpolation and F(x) is set equal to -F(1/x); therefore, the function values of abscissa breakpoints less than 1 are ignored.

There are separate variables for Red and Blue to indicate whether all or none of the ground casualties are replaced.

Figure 1 (concluded)

Chapter II

GAME-SOLVING PROCEDURE

The running time of OPTSA is essentially proportional to the number of *daily assessments* that are computed--as practically all of the time is spent in calculating payoffs, rather than in simplex operations. Therefore, the objective of revising the game-solving procedure is to minimize (or, at least, reduce) the number of daily assessments that must be calculated to solve the game. Two avenues of approach have been used. First, in order to solve the game, it is not necessary that all payoff entries of a matrix game be known. The new procedure generates columns of payoff entries as they are needed, finds intermediate solutions, and stops when an intermediate solution is also a solution for the whole game (the method is proved in Section B and illustrated in Section C of this chapter). Second, assessments are computed period by period--storing assessment results at the last day of each period and using them later (as is explained in Section D). Note that the first improvement is unrelated to the fact that OPTSA is a multistage behavioral game, while the second is directly related to the order in which payoff entries are computed in a staged game. The first improvement can be utilized in a staged game in such a way that each first-period payoff entry that need not be computed eliminates the need for solving many games at the latter stages. This is the major factor in reducing the amount of computation in OPTSA.

A. SUMMARY OF PROCEDURE OF THE ORIGINAL OPTSA MODEL

Consider a three-period game representing a war of D days with d_1, d_2, and d_3 days (resp.) in each period, with a list of n possible aircraft allocations in each period for each side. Therefore, there are n^2 possible allocation-choice pairs for each period (a "choice pair" meaning a choice of allocation by each side), and $(n^2)^3 = n^6$ different possible combinations of allocations for the three-period war. For each of these allocation combinations, the assessment routine is performed to find a specified payoff measure (e.g., FEBA position at the end of day D). Therefore, n^6 payoffs are computed. These payoffs are organized into $n^4 = (n^2)^2$ $n \times n$ matrix games--one game for each combination of first- and second-period allocations by each side. These games are solved by the simplex method to determine optimal third-period (max-min) Blue and Red strategies, given the corresponding allocations made the first and second periods. The resulting n^4 game values become the payoffs for n^2 $n \times n$ second-period games--which are solved to give the optimal second-period strategies, given the allocations made the first period. The n^2 resulting game values are the payoffs for one first-period game, which is solved to determine the optimal strategies for Red and Blue for choosing a first-period allocation. Therefore, n^4+n^2+1 games are solved--each game $n \times n$. The assessment routine, which makes D daily assessments for each campaign, is called n^6 times (i.e., n^6D daily assessments are performed).

The original model was unable to play two-period wars; but by using the analogous procedure, a two-period war would require n^4D daily assessments.

B. REVISED METHOD OF SOLUTION OF MATRIX GAMES

Let us consider first a new procedure for solving a (one-stage) $n \times n$ matrix game, paying attention to how many payoff entries a_{ij} (both i and j=1 to n) need to be computed. Blue is the row player; Red, the column player; and a_{ij}, the payoff to Blue when Blue chooses pure strategy i and Red chooses pure strategy j. Let Red start by choosing (arbitrarily) pure strategy 1. Against this declared Red strategy, Blue's best strategy is to choose strategy i_1, where $a_{i_1,1}$ is the largest entry in column 1. Compare all the payoff entries a_{i1}, i=1 to n; and take i_1, the argument of the largest, as Blue's initial strategy. It is possible that $a_{i_1,1}$ is actually a saddle point of the whole game--which can be tested by computing all the entries $a_{i_1,j}$, j=2 to n. If $a_{i_1,1}$--which we already know is equal to or larger than a_{i1} for i=1 to n--is also equal to or smaller than $a_{i_1,j}$ for j=2 to n, then $a_{i_1,1}$ is a saddle point of the whole game. Thus, the optimal strategies and value have been found by determining only 2n-1 payoff entries, instead of the full n^2.

What if $a_{i_1,1}$ is not a saddle point? Then there is at least one j where $a_{i_1,1} > a_{i_1,j}$. Let us call the j such that $a_{i_1,j}$ is the *smallest* j_2. Let j_2 enter as a new constraint; that is, allow Red to play *two* pure strategies, j_1=1 and j_2, which was just found. What are the max-min strategies for Blue and Red for this new game? They are found by using the dual simplex method of linear programming. Set up the first part of the procedure (when the payoff entries a_{ij} of column 1 were computed) as the two-constraint linear programming problem (LP):

maximize σ

$$\text{s.t.} \quad \sigma \leq a_{11}x_1 + a_{21}x_2 + a_{31}x_3 + \cdots + a_{n1}x_n ,$$

$$\sum_{i=1}^{n} x_i = 1 .$$

The x_i form the Blue randomized strategy, and the first constraint contains the payoffs for column 1 (i.e., constraints correspond to Red pure strategies). In the computer program, this LP is put into standard form at optimality.[1] The variable x_{i_1} is basic at 1.0; the other x_i are zero; and $\sigma = a_{i_1,1}$. The dual variables give $y_1 = 1.0$ as Red's strategy.

Now let Red also be able to play pure strategy j_2. Add the constraint

$$\sigma \leq a_{1,j_2}x_1 + a_{2,j_2}x_2 + a_{3,j_2}x_3 + \ldots = a_{n,j_2}x_n$$

$$(\text{i.e., } \sigma \leq \sum_{i=1}^{n} a_{ij}x_i \quad j=j_2) \ .$$

to the above LP. To do so, the payoff entries a_{i,j_2}, i=1 to n, must be computed. The entries form a column (j_2) in the game matrix but become a row of the LP. When this new constraint is pivoted into the previous LP, which was optimal and in standard form, the solution to the previous LP becomes infeasible for the new LP, as the new constraint is violated. The dual simplex method is used to reoptimize and find a solution $\vec{x}^*$ and game value σ^* to this three-constraint LP. The dual variables corresponding to the constraints j=1 and $j=j_2$ yield Red's probabilities for j=1 and $j=j_2$ in the optimal Red strategy. No other Red pure strategy is used.

The original basic variable x_{i_1} generally (but not necessarily) stays basic. It is possible that it remains the only active Blue strategy and that only the Red strategy changes (e.g., if there is a saddle point at a_{i_1,j_2}). Let I be the set $\{i | x_i^* > 0\}$.

Does this new Blue strategy violate any constraints corresponding to Red strategies that were not considered in finding the Blue strategy? More precisely, if the game is formulated as a linear program (as above), we have

[1]This statement is not precisely true; the computer program adds an input value to all payoff entries to make them positive and then solves the LP min $\sum u_i$ s.t. $1 \leq \sum a_{ij}u_i$ for $j \varepsilon J$, where $u_i = x_i/\sigma$.

maximize σ (the game value)

$$\text{s.t.} \quad \sigma \leq \sum_{i=1}^{n} a_{ij} x_i \qquad j=1,2,\ldots,n \ ,$$

$$\sum_{i=1}^{n} = 1 \ .$$

We have found a solution vector $\vec{x}^*$ to the relaxed problem

maximize σ

$$\text{s.t.} \quad \sigma \leq \sum_{i=1}^{n} a_{ij} x_i \qquad j \varepsilon J \ ,$$

$$\sum_{i=1}^{n} = 1 \ ,$$

where J is a subset of $\{1,2,\ldots,n\}$. For the three-constraint LP (above), $J = \{1,j_2\}$. If $\vec{x}^*$ also satisfies the $n-o(J)$ constraints

$$\sigma \leq \sum_{i=1}^{n} a_{ij} x_i^* \qquad j \varepsilon J' = \{1,2,\ldots,n\}-J \ ,$$

it will be an optimal solution to the original game. The optimal value of σ^* for the relaxed problem is known. Compute the payoffs a_{ij} for $i \ \varepsilon \ I = \{i | x_i^* \neq 0\}$ and $j \ \varepsilon \ J'$;[1] and compute for each $j \ \varepsilon \ J'$ the quantity $\sum_{i \varepsilon I} a_{ij} x_i^* - \sigma^*$. Since $x_i^* = 0$ for $i \notin I$, this sum is indeed the right-hand side of the constraint for j in the full-game problem with the current $\vec{x}^*$. If this sum is nonnegative for each j, the solution $\vec{x}^*$ to the relaxed problem is the optimal Blue strategy for the whole game. The dual variables provide Red's optimal solution. This procedure has involved computing no more than $o(I)o(J')$ new payoff entries.

If for some $j \ \varepsilon \ J'$, $\sum_{i \varepsilon I} a_{ij} x_i^* - \sigma^* < 0$, choose the j for which this quantity is most negative to enter as a new constraint. In the example above, call this fourth constraint j_3,

[1]The program keeps track of payoff entries that were computed at a prior step and does not recompute them.

allowing Red to choose among the three pure strategies $j_1=1$, j_2, and j_3. Compute the payoff entries in column j_3, and pivot the new constraint into the tableau. Since the current solution is now infeasible, use the dual simplex method to reoptimize and find the new game value and max-min Blue and Red strategies. Then test Blue's solution to see whether it violates any of the remaining constraints--continuing as above until an optimal solution to the whole game is found. Termination must occur, as eventually J' will be empty. Usually, many fewer than n^2 payoff entries need to be computed. Figure 2 is a flowchart of the procedure for solving the one-stage matrix game.

The number of rows (Blue pure strategies) r that have to be computed varies between r* (the number of active Blue strategies in the solution to the game) and n. Similarly, c (the number of columns that have to be computed) varies between c* (the number of active Red strategies in the solution to the game) and n. Experience with OPTSA so far indicates that r* and c* never exceed 3 and rarely exceed 2; and, in general, $r = r^*$ to r^*+2 and $c = c^*+1$ or c^*+2--regardless of n. Given r and c, the number of payoff entries that have to be computed is $(r+c)n-rc$.

Note that, in all this discussion, the initial choice of pure strategy 1 as the first Red pure strategy to try was completely arbitrary. If there is a strong possibility that Red pure strategy 1 will not be active in the solution but that another pure strategy on the Red list will be, then (if this other strategy is chosen first) a column of unnecessary computation might be saved. Therefore, the computer program allows an input pure strategy from the Red list to be tried first. (Unless specifically indicated otherwise, pure strategy 1 will be used.) The input pure strategy can be different for matrix games occurring at different stages of the behavioral game.

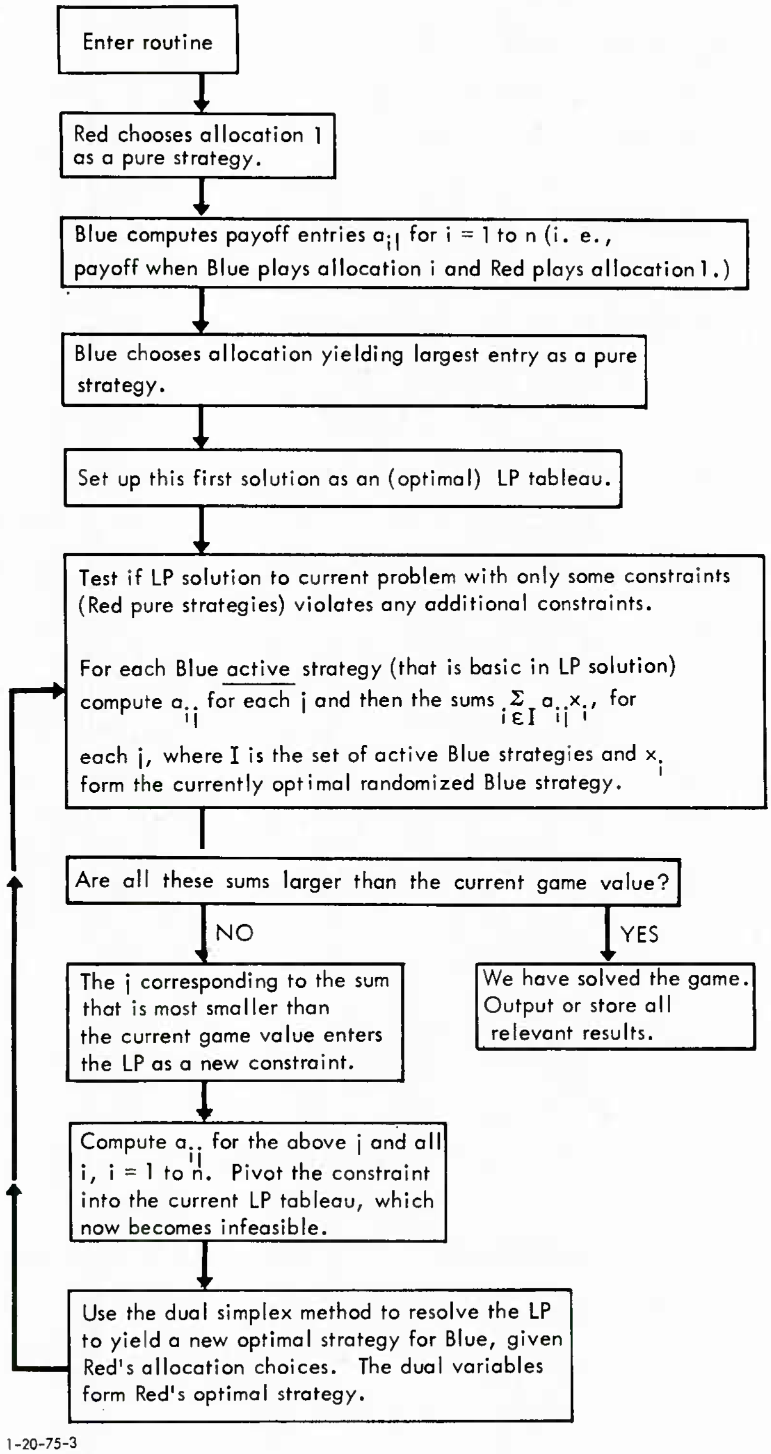

Figure 2. ONE-STAGE GAME-SOLVING PROCEDURE

C. EXAMPLES OF THE REVISED SOLUTION PROCEDURE FOR MATRIX GAMES

To clarify the new matrix game-solution procedure, this section presents two examples--showing how the calculations proceed and exactly which payoff entries must be calculated. The first example is a 3 x 3 matrix game with a saddle point. The second has four pure strategies for Blue and six for Red (i.e., a 4 x 6 payoff matrix). Though the formulas for running time that are derived in Chapter IV (below) assume square payoff matrices (as will usually be the case in actual practice), the method still works for nonsquare payoff matrices. In both examples, the effects on the amount of computation--with, first, Red picking pure strategy 1 to start and, second, an intelligent guess by Red for a starting pure strategy--are illustrated.

These examples are somewhat typical of payoff matrices encountered in OPTSA, where (for one side or the other) one pure strategy dominates most or all of the others and is found early on. Then it is easy for the other side to optimize against it.

The first example has the following payoff matrix:

	R1	R2	R3
B1	1.	3.	2.
B2	4.	6.	1.
B3	5.	4.	3.

By looking at the whole payoff matrix, it is evident that element (B3,R3)=3 is a saddle point. However, if no payoff entries are known at the outset, how can the solution be found? Each step in the following sequence represents the operations performed by one of the game-solving subroutines in the computer program.

Step 1 - Initial Setup. Arbitrarily, Red chooses pure strategy 1 to be tried first. Column R1 of the matrix must be computed--resulting in the following (a circle around a payoff entry indicates that it has been computed):

	R1	R2	R3
B1	1.	3.	2.
B2	4.	6.	1.
B3	5.	4.	3.

Blue solves the relaxed game or subgame:[1]

	R1
B1	1.
B2	4.
B3	5.

This is equivalent to finding the largest element in column R1. This is element (B3,R1)=5. Therefore, Blue's solution to the first relaxed game is pure strategy B3, with the game value equal to 5.

Test whether strategy B3 is a solution to the *whole* game--by computing row B3 of the matrix, yielding--

	R1	R2	R3
B1	1.	3.	2.
B2	4.	6.	1.
B3	5.	4.	3.

Is the current game value (5) less than or equal to each element in row B3? No. The smallest element in row B3 is 3, corresponding to Red pure strategy R3.

Step 2. Red pure strategy R3 (as well as R1) must be considered as a possibility for Red. Column R3 of the payoff matrix is computed, yielding--

---

[1]The terms "relaxed game" and "subgame" are synonymous. The latter term reflects the fact that the payoff matrix for the subgame is a submatrix of the whole game matrix; the former, that solving the subgame involves a relaxed LP (i.e., an LP with fewer constraints than the whole-game LP).

	R1	R2	R3
B1	1.	3.	2.
B2	4.	6.	1.
B3	5.	4.	3. .

Blue solves the subgame:

	R1	R3
B1	1	2
B2	4	1
B3	5	3 .

In the program this is done by starting at the (now infeasible) Blue solution for the *previous* relaxed game:

	R1
B1	1
B2	4
B3	5

and using the dual simplex method. Here, it is evident from inspection that element (B3,R3)=3 is a saddle point of the subgame. Thus, Blue pure strategy B3 remains the optimal solution, but the Red strategy changes--corresponding to a change of basic slack variables in the LP. The game value is now 3--not 5--which, not surprisingly, is worse for Blue, since a new option for Red was added.

Test whether strategy B3 is a solution to the whole game. Since row B3 has already been computed, all that now need be done is to check whether 3 (the new game value) is less than or equal to each element in row B3. This is the case; hence, the solution to the game is the saddle point (B3,R3), with game value 3--which can be stored or output. By looking back at the first matrix in Step 2, we see that only 7--not the full 9--payoff entries needed to be computed.

Matrices like this, which have a saddle point, are extremely common in OPTSA. The first Red strategy tried is not good; however, the correct Blue strategy is found immediately, and then it is easy to find the correct Red strategy. With a good guess of the first Red strategy, the computation can be reduced further. In this example, if R3 rather than R1 were tried first, the following situation would have resulted:

	R1	R2	R3
B1	1.	3.	2.
B2	4.	6.	1.
B3	5.	4.	3.

Payoff entries (B1,R1) and (B2,R1) did not need to be found; hence, only 5--not 7--entries were computed.

The second example (a nonsquare game) has the 4 x 6 payoff matrix (shown below). There is no saddle point. In the series of steps presented, the actual solution of the subgames is non-trivial; it is omitted here. In the computer program, it is done by the dual simplex method.

	R1	R2	R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

Let probability vectors (b_1,b_2,b_3,b_4) and $(r_1,r_2,r_3,r_4,r_5,r_6)$ denote the optimal Blue and Red randomized strategies for a subgame solved at some step; G, the value of the subgame. As a check, the game was solved first by the regular simplex method--to yield $G = 4\frac{2}{3}$; $\vec{b}^* = (0,0,\frac{1}{3},\frac{2}{3})$; and $\vec{r}^* = (0,0,\frac{2}{3},\frac{1}{3},0,0)$.

Step 1 - Initial Setup. In the absence of any input first guess, try Red pure strategy R1. Compute column R1, yielding--

	R1	R2	R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

The largest element in column R1 is (B4,R1) = 6. Pure strategy B4 is Blue's solution to the first subgame; G = 6.

Test whether B4 is the solution to the whole game. Compute row B4, yielding--

	R1	R2	R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

Is $6 \leq$ element (B4,Ri) for i=1 to 6? No, elements (B4,R3) and (B4,R4) are both less than 6. The smallest of these is element (B4,R4)=4. Hence, Red should also consider pure strategy R4.

Step 2. Compute column R4, yielding--

	R1	R2	R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

Blue must solve the subgame:

	R1	R4
B1	5	2
B2	5	1
B3	3	6
B4	6	4

All payoff entries to this subgame are known. The subgame solution is $b_3 = 0.4$; $b_4 = 0.6$; game value $G = 3b_3 + 6b_4 =$

$6b_3 + 4b_4 = 4.8$. B3 and B4 are the active Blue pure strategies-- B_3 has entered.

Test whether this solution is optimal to the whole game. To do so, the values in row B3 and B4 are needed. Compute row B3, yielding--

	R1	R2	R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

Then test whether the current game value, 4.8, is less than or equal to [0.4 x element (B3,Ri)] + [0.6 x element (B4,Ri)], for i=1 to 6. This is already the case with R1 and R4 as they were in the subgame. For the other columns, construct the following table:

Column i	Entry (B3,Ri)	Entry (B4,Ri)	0.4(B3,Ri) + 0.6(B4,Ri)
R1	(already satisfied)		(4.8)
R2	3	10	7.2
R3	4	5	4.6
R4	(already satisfied)		(4.8)
R5	1	8	5.2
R6	7	6	6.4

4.8 is *not* less than or equal to 4.6, the value from column R3. Therefore, Red pure strategy R3 should also be considered.

Step 3. The procedure is exactly analogous to Step 2 and uses the same section of the computer code.

Compute column R3, yielding--

	R1	R2	R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

Blue must solve the subgame:

	R1	R3	R4
B1	5	1	2
B2	5	3	1
B3	3	4	6
B4	6	5	4

The optimal solution is $b_3 = \frac{1}{3}$; $b_4 = \frac{2}{3}$; game value $G = \frac{1}{3} \times 4 + \frac{2}{3} \times 5 = \frac{1}{3} \times 6 + \frac{2}{3} \times 4 = 4\frac{2}{3}$. The Blue active strategies remain B3 and B4; however, the Red active strategies change from R1 and R4 to R3 and R4.

Since rows B3 and B4 have already been computed and there are no additional Blue active strategies, no new payoff computation need be done at this point. Test: is $G = 4\frac{2}{3} \leq$ $\left(\frac{1}{3} \text{ x element } (B3,Ri)\right) + \left(\frac{2}{3} \text{ x element } (B4,Ri)\right)$ for i=1 to 6? Since columns R1, R3, and R4 were in the subgame, the inequality will be satisfied for them. The inequality is strict for R1 (as R1 is not active in the solution). Equality holds for R3 and R4, the active Red pure strategies. For other columns, construct the following table:

Column i	Entry (B3,Ri)	Entry (B3,Ri)	$\frac{1}{3}$(B3,Ri) + $\frac{2}{3}$(B4,Ri)
R1	(already satisfied)		(5)
R2	3	10	$7\frac{2}{3}$
R3	(already satisfied)		$(4\frac{2}{3})$
R4	(already satisfied)		$(4\frac{2}{3})$
R5	1	8	$5\frac{2}{3}$
R6	7	6	$6\frac{1}{3}$

Since $4\frac{2}{3}$ *is* less than or equal to all the numbers in the last column, the solution to the current relaxed game is also the solution to the whole game. In the computer program, the dual variables can be obtained from the simplex tableau to yield Red's optimal strategy. Here a little bit of calculation shows it to be $r_3 = \frac{2}{3}$ and $r_4 = \frac{1}{3}$. Therefore, the solution to the game is $\vec{b}^* = (0,0,\frac{1}{3},\frac{2}{3})$ and $\vec{r}^* = (0,0,\frac{2}{3},\frac{1}{3},0,0)$, with game value $G = 4\frac{2}{3}$. The game matrix with all computed entries circled is shown below. Rows B3 and B4 and columns R1, R4, R3 were computed; only 18 (instead of the full 24) payoff entries were computed.

	R1	R2	R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

Though payoff column R1 was computed, it was not active in the final solution. Suppose it was guessed that since, say, R3 would be active in the final solution, it was tried first. The series of computations would have been as follows:

Step 1: Compute column R3, where the largest element is 5 (in row B4). Compute row B4. 5 is not a saddle point. The smallest element in row B4 is 4 (in column R4).

Step 2: Compute column R4, and solve the subgame for Blue. The solution involves B3 and B4.

This solution is optimal for the whole game. Therefore, only the circled payoff entries (below) have been computed (i.e., only 16 of the 24 entries have been necessary). This is a minimal number--as B3, B4, R3, and R4 are all active in the solution to the whole game. However, the good first guess of R3 (rather than R1) has eliminated the results for computation of two entries. With larger matrices, a good first guess can result in much more substantial savings.

	R1	R2	(first guess) R3	R4	R5	R6
B1	5.	8.	1.	2.	4.	5.
B2	5.	11.	3.	1.	1.	7.
B3	3.	3.	4.	6.	1.	7.
B4	6.	10.	5.	4.	8.	6.

Note, however, that if R4 were the input first guess (even though R4 is active in the final solution), column R5 would have to be computed along the way. Since Red pure strategy R5 is not active in the final solution, there has still been unnecessary payoff computation. The point is that, while a good first guess sometimes will save considerable computation, what seems like a reasonable first guess often will not.

D. PROCEDURE FOR COMPUTING ASSESSMENTS PERIOD BY PERIOD IN MULTIPERIOD GAMES

This second revision to shorten the running time of OPTSA operates independently of the game-solving procedure described in Section C (above). It is best illustrated with an algebraic example.

Suppose that there are three periods of lengths d_1, d_2, and d_3 days, where $d_1 + d_2 + d_3 = D$, the number of days in the war.

Suppose that allocations for Blue and Red have been made for the first two periods; and, given this, we wish to find the optimal third-period strategies. To do so, we must solve a third-stage game, which involves computing a number of payoff entries. But in the computation of each entry, the situation at the end of day d_1+d_2 is the same. Therefore, we can compute the outcome of the war at the end of day d_1+d_2 and store the result (the aircraft inventory, shelter inventory, firepower, ground-force inventory, FEBA position). For each third-period allocation-choice pair, we can--starting with the situation at the end of the day d_1+d_2--fight the war for the third period only and find the payoff entry. When all the necessary entries are found, the game is solved to find a second-period payoff entry. Similarly, when the second-period allocation varies and the first-period allocation remains fixed, we can compute and store the outcome of the war at the end of day d_1 and start from there in computing payoff entries. This procedure eliminates the duplication involved in recomputing the first period for each payoff entry. (The procedure in two-period games is entirely analogous.)

If this revision alone were implemented in the original OPTSA model without the method of Section C (above), assessments required would be reduced from n^6D to $n^2d_1+n^4d_2+n^6d_3$--a considerable savings. For a two-period war, the reduction would be from n^4D to $n^2d_1+n^4d_2$ daily assessments.

The particular results that must be stored at the end of the period are--

- Blue and Red division inventory, by type, at beginning of last day of period.
- Blue and Red aircraft inventory, by type, at beginning of last day of period.
- Blue and Red shelter inventory, by type, at beginning of last day of period.
- Blue and Red divisions destroyed, ty type, on last day of period.

- Blue and Red aircraft destroyed, by type, on last day of period.
- Blue and Red shelters destroyed, by type, on last day of period.
- Blue and Red cumulative total firepower delivered, up to and including last day of period.
- Blue and Red cumulative air firepower delivered, up to and including last day of period.
- FEBA position at end of last day of period.

These variables are all set up in COMMON arrays in the computer program. The arrays show the values of the variables on every day.

E. PROCEDURE OF THE REVISED OPTSA MODEL

The revised game-solving procedure of OPTSA is described for a three-period war. Since a separate subroutine for computing payoff entries and solving a game is required for each stage, the number of subroutines increases with the number of periods in the war. Hence, some programming revisions are necessary to extend the model to process wars of four or more periods. The program can process wars of one or two periods by using part of the procedure for a three-period war. More specifically, a two-period war is treated as the last two periods of a three-period war with an arbitrary first-period allocation by both sides. Similarly, a one-period war is treated as the last period of a three-period war.

Figure 3 is a flowchart of the model for a three-period war. As in the original OPTSA model, a payoff entry for a game at stage k is the *value* of a game at stage k+1. Hence, to generate a payoff entry for a stage-k game requires solving a game at stage k+1--which in turn requires finding payoff entries for this stage k+1 game, which means solving a stage k+2 game for each stage k+1 payoff entry, and so forth--until the last stage, where payoff entries are actual war outcomes. (This is illustrated by diagrams in Chapter IV of Reference [4].)

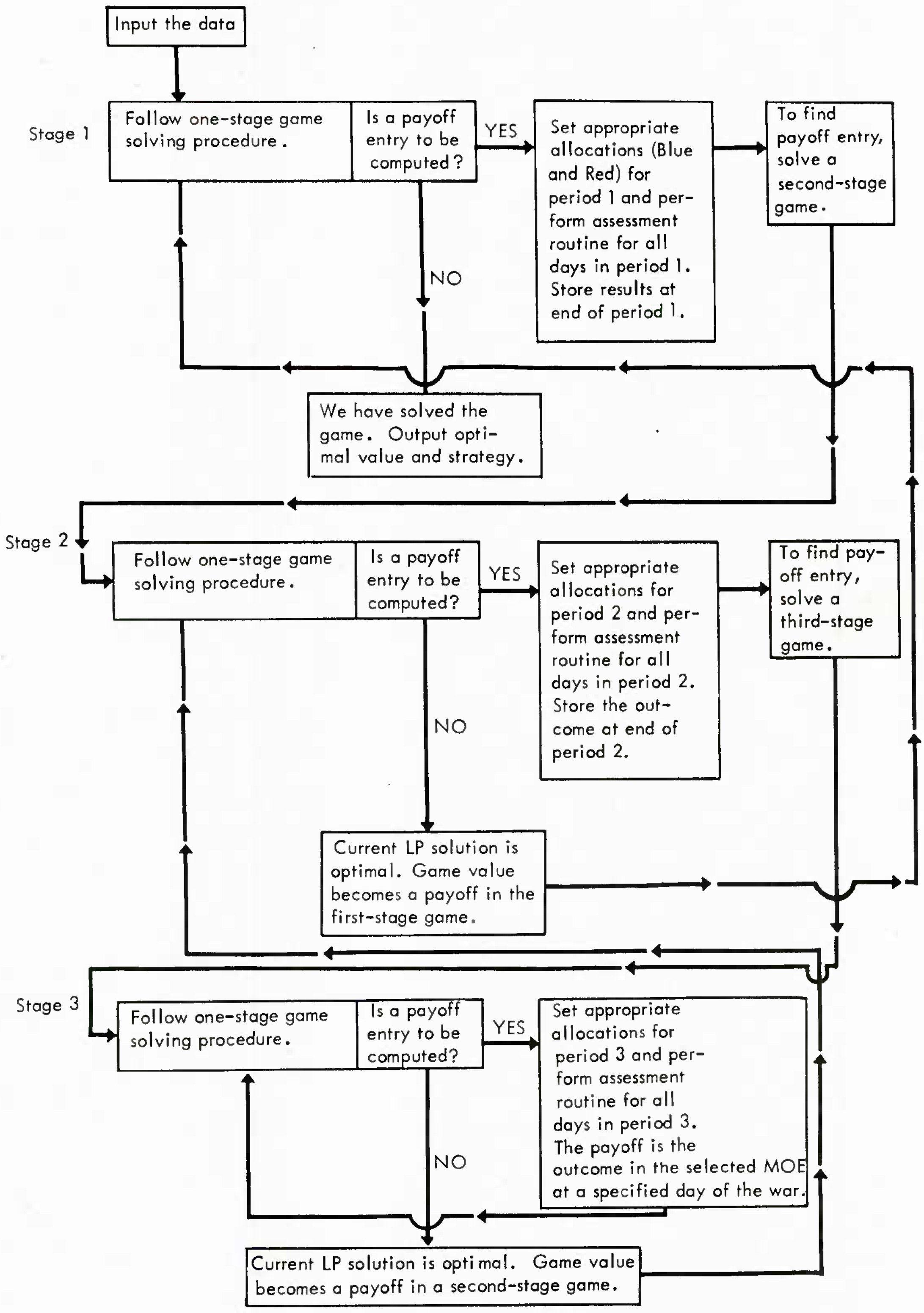

Figure 3. PROCEDURE FOR A THREE-PERIOD WAR USING THE REVISED OPTSA MODEL

The game begins at stage 1. Red arbitrarily chooses allocation 1 (or, if input, another allocation from the list) for the first period; and, in order for Blue to choose his best strategy, the first-period payoff entries must be computed for each of Blue's allocation choices. First, compute the entry where Blue plays allocation 1 the first period. The assessment routine is fought for the first period, and the result at the end of day d_1 is stored. The payoff entry is the value of a stage-two game in which Blue and Red play allocation 1 in period 1 *and play optimally from then on.* To determine this value, call the second-stage game-solving routine, which will assess the payoff of the war for various second-period strategies and determine the best one. To compute payoff entries, the second-stage game-solving routine will assess the outcome at the end of day d_1+d_2 by using the appropriate second-period allocation, storing the outcome, and then calling the third-stage game-solving routine to solve a game whose value is the payoff entry for the second-stage game. Payoff entries for the third-stage games are merely the outcomes of the war in a specified payoff measure (e.g., FEBA position at the end of the war) when appropriate combinations of first-, second-, and third-period aircraft allocations are used.

Each third-stage game is solved by using the one-stage game-solving routine (which only generates payoff entries when necessary), and its game value becomes the appropriate payoff entry for the second-stage game. The second-stage game is then solved--computing more payoff entries (i.e., solving more third-stage games) as they are needed. The value of the second-stage game then becomes the first-period payoff entry when Blue and Red both play allocation 1 the first period. The optimal second- and third-period strategies can be printed if desired. Then go through the whole procedure again, to compute the first-stage game-payoff entry when Blue plays allocation 2 the first period and Red plays allocation 1. Continue to solve the first-stage game by the one-stage game-solving method--going through the whole procedure (above) to compute each needed first-stage payoff entry. When the first-stage game is solved, the game value and optimal first- and second-period strategies can be output. Third-period strategies for all third-stage games solved can also be output.

In a two-period war, the procedure is very similar. Two one-stage game-solving routines are used. Payoff entries for second-stage games are war outcomes (e.g., FEBA position). The values of second-stage games become payoff entries for the first-stage game; payoff entries are computed only as needed.

Chapter III

ASSESSMENT PROCEDURE

This chapter describes the changes made to the assessment procedure of OPTSA, to incorporate more detail in some activities and to improve the mechanism for representing other activities. This description is methodologically oriented. A detailed documentation of the entire new assessment procedure is included as an appendix to this volume (below). Throughout this chapter, the assessment procedure of the original version of OPTSA is contrasted with the assessment procedure of the revised version of OPTSA.

A. QRA AIRCRAFT

The original version of OPTSA could not play Quick Reaction Alert (QRA) aircraft (except by assuming that all QRA aircraft are in sanctuary as a justification for not playing them). OPTSA can now play QRA aircraft, as follows: An input number of Blue GP aircraft are assumed not to fly any missions on each day and, instead, to stay on the airbase. For example, if there are to be 100 Blue QRA aircraft, then each day 100 of the Blue GP aircraft do not fly any missions--provided that there are at least 100 Blue GP aircraft. If there are less than 100 Blue GP aircraft, then none of these aircraft fly any missions. (The same holds for Red.)

Note that there is no explicit assumption that the initial QRA aircraft actually are GP aircraft. For example, suppose that Blue has 1,000 GP aircraft and 200 SP-ABA aircraft--100 of which are designated to be QRA. Then the inputs to OPTSA should be that Blue has 100 SP-ABA aircraft and 1,100 GP aircraft--100 of which are QRA. The assumption that OPTSA makes is that QRA losses due to enemy ABAs are replaced with GP aircraft.

If there is a shortage of shelters, all QRA aircraft are assumed to be sheltered before other aircraft are allowed to be sheltered.

B. AIR-TO-AIR MODIFICATIONS

1. Sortie Rates

The original version of OPTSA did not play sortie rates (i.e., it assumed essentially that all aircraft always had a sortie rate of 1.0). The new version of OPTSA allows different types of aircraft on different missions to have different sortie rates, and these sortie rates can change once during the war (the day on which the change occurs need not be the same for the two sides). Playing sortie rates (other than 1.0) is a basic change that affects more than just the air-to-air combat (e.g., a higher ABA sortie rate will result in more air-to-ground kills). Since this is a basic change, and since this change affects air-to-air combat significantly, this section describes how OPTSA plays sortie rates.

There are a total of 24 different sortie rates that can now be input into OPTSA (i.e., 12 for Blue and 12 for Red). The day on which Blue first begins to use his sustained sortie rates is also input, and six of the Blue sortie rates are surge rates that apply up to that day; the remaining six (sustained) sortie rates apply from that day on. The six rates apply to--

(1) GP aircraft on CAS missions;
(2) GP aircraft on ABA missions;
(3) GP aircraft on INT missions;
(4) SP-CAS aircraft;
(5) SP-ABA aircraft; and

(6) SP-INT aircraft.

(The same structure also holds for Red.)

Air-to-air attrition in the original version of OPTSA was calculated as follows: if there were S shooters, T targets, and an air-to-air probability of kill of k, then the number of targets killed was given by

$$\dot{T} = T(1-\exp[-\frac{Sk}{T}]) \ .$$

An intermediate version of OPTSA was constructed, in which

$$\dot{T} = f(S,T,d,k) \ , \qquad (1)$$

where d is the probability of detection and

$$f(S,T,d,k) = T(1-\exp[-\frac{Sk}{T}(1-\exp[-dT])]) \ .$$

In the new version of OPTSA, the functional form of f has been changed and new functions have been added as options. (These functional changes are discussed in Subsection 3 of this section, below.) For the time being, let f´ denote one of the new attrition functions in OPTSA. Then instead of computing

$$\dot{T} = f'(S,T,d,k) \ ,$$

OPTSA now computes that the number of target sorties killed $\dot{T}_s$ is given by

$$\dot{T}_s = f'(r_t T, r_s S, d, k) \ , \qquad (2)$$

where r_s is the sortie rate of the shooters and r_t is the sortie rate of the targets; and then the number of targets killed is computed by

$$\dot{T} = \begin{cases} \dot{T}_s, & \text{if } r_t \leq 1 \ ; \\ \dfrac{T_s}{r_t}, & \text{if } r_t > 1 \ . \end{cases} \qquad (3)$$

That is, one aircraft with a sortie rate of 2.0 is treated as though it were two aircraft; and it is assumed that, if the sortie rate is less than 1.0, then a sortie killed results in an aircraft killed--while if the sortie rate is greater than 1.0, then the percentage of sorties killed equals the percentage of aircraft killed.

This method for considering sortie rates other than 1.0 is the same as is used in IDAGAM I (see Refs. [1] and [2]) and appears to give reasonable results. However, it cannot be theoretically justified for sortie rates greater than 1.0, because OPTSA (and IDAGAM I) assesses attrition once per day, while multiple sorties per day imply that attrition can occur on the first sortie (which would affect the outcome one way) or on later sorties (which would affect the outcome a different way). If further research indicates that this variance in outcome is significant, then attrition should be assessed more frequently than once per day. For the time being, note that, if the number of sorties killed as given by Equation (2) is correct, then (for $r_t > 1$) Equation (3) would give a lower bound on the number of aircraft killed--because, if less than $\dot{T}_s/r_t$ aircraft are killed, then the number of sorties killed would be less than $\dot{T}_s$, even if all aircraft are killed on their first sortie. On the other hand, for $r_t > 1$, Equation (2) might overestimate the number of sorties killed--because some aircraft on both sides would be killed on their first sortie; and so there would be, on the average, less than $r_t T_t$ targets and less than $r_s S_s$ shooters.

2. Air-To-Air Probabilities of Kill

In the original version of OPTSA, the air-to-air probabilities of kill could be only partially a function of mission and of shooting type of aircraft--and not a function of the type of target. This limitation could be significant, because there can be a considerable difference between the capability of interceptors to kill GP-ABA aircraft and their capability to kill SP-ABA aircraft. Accordingly, 32 probabilities of kill are now input: Red GP-INT killing Blue GP-CAS, GP-ABA, SP-CAS, and SP-ABA (and vice versa); Red SP-INT versus the same four types of attackers; and the same for Blue interceptors of each type versus the four types of Red attackers.

3. Air-To-Air Attrition Equations

To determine air-to-air attrition, the original version of OPTSA used a one-parameter exponential attrition equation. The intermediate version of OPTSA averaged out the probability of kill of the various types of aircraft and used Equation (1) with the average probability of kill. Equation (1) is an approximation to another equation that can be derived from basic assumptions (as in Ref. [10]). In the new version of OPTSA, Equation (1)--which is a homogeneous, exponential equation--has been replaced with a heterogeneous, binomial equation.[1]

For example, let

I_1 = number of Blue GP-INT aircraft;
I_2 = number of Blue SP-INT aircraft;
A_1 = number of Red GP-CAS aircraft;
A_2 = number of Red GP-ABA aircraft;
A_3 = number of Red SP-CAS aircraft; and
A_4 = number of Red SP-ABA aircraft.

Let r_{11}, r_{12}, r_{21}, r_{22}, r_{23}, and r_{24} be sortie rates corresponding (resp.) to the six categories above. Let d_{ij} be the probability that a type-i interceptor (i=1,2) detects a type-j attacker (j=1,2,3,4), given that they are both in the same air combat area; and let k_{ij} be the same for the probability of kill, given detection. Finally, let N be the number of air combat areas (i.e., if N = 2, one-half of the attackers are assumed to be attacking through each of the two areas and

---

[1]In terms of Reference [10], we have replaced an exponential approximation to Equation (18) with Equation (17´) of that reference. If any binomial attrition expression would involve exponentiating a negative number, zero is used instead.

one-half of the interceptors are defending in each area). Then the number of type-j attacking sorties killed, A_{js}, is now computed in OPTSA by the equation

$$\dot{A}_{js} = r_{2j}A_j\left(1 - \prod_{i=1}^{2}\left[1 - \frac{k_{ij}}{\bar{A}_s/N}\left(1-(1-\bar{d}_i)^{\bar{A}_s/N}\right)\right]^{r_{1i}I_i/N}\right)t \quad , \quad (4)$$

where $\bar{A}_s = \sum_{j=1}^{4} r_{2j}A_j$ and $\bar{d}_i = \dfrac{\sum_{j=1}^{4} d_{ij}r_{2j}A_j}{\bar{A}_s}$. As discussed in Subsection 1 of this section (above), the number of type-j attacking aircraft killed, A_{ij}, is then given by

$$\dot{A}_j = \begin{cases} \dot{A}_{js}, & \text{if } r_{2j} \le 1 \; ; \\ \dfrac{\dot{A}_{js}}{r_{2j}}, & \text{if } r_{2j} > 1 \; . \end{cases}$$

(Equations for Red interceptors killing Blue attackers are completely analogous.)

Analogous equations are also available to calculate the number of interceptors killed by attackers; however, as an option, an alternative method for calculating the number of interceptors killed by attackers has been added to OPTSA. This new method assumes that aircraft on CAS and ABA missions are not interested in engaging enemy interceptors and will do so only if they are engaged by them. If an attacking aircraft is not engaged by an interceptor, it never attacks an interceptor but always continues on toward its primary target. An input percent of the attackers (which can depend on the type of aircraft--GP or SP--and on mission) that are engaged by enemy interceptors are assumed to jettison their ordnance and engage the interceptors; the remainder are assumed to keep their ordnance--not shoot at the interceptors--and, if they are not killed by the interceptors, to continue toward their primary target.[1]

For example, let I_1, I_2, $A_1,\ldots,A_4$, $r_{11},\ldots,r_{24}$, N, $\bar{A}_s$, and $\bar{d}_i$ be as defined above; and let k_{ji} be the probability that a type-j attacker kills a type-i interceptor, given that he is engaged by enemy interceptors. Then, if the new method for computing air attrition is selected, the number of attacker sorties engaged by type j, A^e_{js}, is given by

$$A^e_{js} = r_{2j}A_j\left(1 - \prod_{i=1}^{2}\left[1 - \frac{1}{\bar{A}_s/N}\left(1-(1-\bar{d}_i)^{\bar{A}_s/N}\right)\right]^{r_{1i}I_i/N}\right) \qquad (5)$$

and the number of type-i interceptor sorties killed, $\dot{I}_{is}$, is given by

$$\dot{I}_{is} = \sum_{j=1}^{4} A^e_{js}(1-\alpha_j)k'_{jk}p_i \ , \qquad (6)$$

where α_j = the fraction of attackers which, when engaged by enemy interceptors, continue toward their primary target, and

$$p_i = \frac{r_{1i}I_i\bar{d}_i}{\sum_{i'} r_{1i'}I_{i'}\bar{d}_{i'}} \ .$$ [2]

The number of attacking aircraft sorties that go on toward their primary target, A^a_{js}, is given by

$$A^a_{js} = r_{2s}A_j - A^e_{js} + \alpha_j(A^e_{js}-\dot{A}_{js}) \ . \qquad (7)$$

The attacking aircraft that were engaged and that jettisoned their ordnance--but that were not killed by interceptors--are assumed to return to their home airbase.

---

[1]If equations analogous to Equation (4) are used to calculate the number of interceptors killed by attackers, then all attacker sorties not killed by interceptors are assumed to continue toward their primary target.

[2]This is the correct version of the erroneous formula given on page B-8 of Reference [5].

C. GROUND-TO-AIR MODIFICATIONS

1. Order of Calculations

In the original version of OPTSA, ground-to-air attrition (due to SAMs and AAA) was not played. In the intermediate version, this attrition was played and was assessed before air-to-air attrition (due to interceptors). Inverting this order, the new version of OPTSA now first assesses attrition to attackers (CAS and ABA) caused by interceptors and then assesses attrition caused by SAMs.

2. Different Rates of Attrition to CAS and ABA Aircraft Due to SAMs

In the intermediate version of OPTSA, one attrition rate that gave the fraction of Blue attacking aircraft (both CAS and ABA) that suffered attrition to Red SAMs (and AAA) was input. Now, four different fractions can be input, so that the attrition rate of attack aircraft to enemy SAMs can depend on the mission (CAS or ABA) that the attack aircraft is flying and on whether the aircraft is GP, SP-CAS, or SP-ABA. (The same structure holds for Red attackers and Blue SAMs.) This structure allows the attrition caused by SAMs to ABA aircraft, which have to penetrate farther into enemy territory, to be higher than that to CAS aircraft. Also, it could allow heavy, slow, SP-ABA aircraft to be more easily shot down by SAMs.

D. AIR-TO-GROUND MODIFICATIONS

1. Red SP-ABA Aircraft and Shelters

If there were more aircraft than shelters, the original version of OPTSA sheltered aircraft by type proportionately to the number of aircraft of that type; and if there were enough shelters, then all aircraft were sheltered. The new version still does so for Blue (except for the preference given to QRA aircraft) and can do it for Red. An option has been added, however, to allow *no* Red SP-ABA aircraft to be sheltered. Other types of Red aircraft are then sheltered proportionally (by type), and they fill up the available Red shelters.

2. Fraction of Aircraft on Ground

The original version of OPTSA assumed that all aircraft on each side were on the ground when the bombs from enemy airbase attackers hit the airbase. A simple improvement is to assume that a fixed percentage of the aircraft are out flying their missions when the attack occurs. (This percentage could depend on an average sortie rate, on average flying-time per sortie, and on the length of a "working" day.) OPTSA now allows such a fixed percentage (for each side), which is used in the following manner: If there are 125 aircraft and 200 shelters on an air-base and if an average of 80 percent of the aircraft are assumed to be on the ground at any one time, then OPTSA treats this case as if there were 100 aircraft and 200 shelters. If there are 400 aircraft and 200 shelters and 80 percent of the aircraft are assumed to be on the ground, then OPTSA "assigns" 200 aircraft (other than Red SP-ABA) to the 200 shelters; and the other 200 aircraft cannot use the shelters (that day). Thus, when the attack occurs, OPTSA assumes that there are 160 aircraft in the open, 160 aircraft in shelters, and 40 empty shelters. (Note that OPTSA assumes that, if an aircraft is assigned a shelter and the attack occurs while that aircraft is on the ground, then that aircraft is always in its shelter--not outside the shelter--being rearmed, refueled, repaired, etc. On the other hand, OPTSA assumes that an aircraft not assigned a shelter cannot use a shelter left empty by an aircraft out flying its mission--as the example above indicates.)

3. Shell Game

In the original version of OPTSA, if there were 100 aircraft and 200 shelters on a notional airbase, then the number of targets for enemy airbase attackers was 100 (i.e., the airbase attackers knew which shelters were occupied and which were not; and they attacked only occupied shelters). In the new version, occupied shelters are indistinguishable from empty shelters. Thus, if there are 125 aircraft (all of which could fit into shelters) and 80 percent of them (100 aircraft) are on the ground, and if there are 200 shelters, then the attackers would have 200 targets. And if x percent of these shelters are successfully hit (say, 10 percent or 20 shelters), then it is assumed that x percent of the aircraft (or 10 aircraft) are destroyed.

4. Number of Passes per ABA Sortie

The original version of OPTSA used a one-parameter exponential attrition equation to compute the attrition of nonsheltered aircraft due to enemy airbase attackers. The equation used by the intermediate version of OPTSA to compute this attrition is

$$\dot{T}_n = f(T_n, S, d_n, k_n) = T_n\left(1-\exp\left[-\frac{qSk_n}{T_n}(1-\exp[-d_nT_n])\right]\right),$$

where T_n is the number of nonsheltered aircraft, S is the total number of airbase attackers that have penetrated the interceptor and SAM barriers, q is the fraction of attackers that attack nonsheltered aircraft, and d_n and k_n are weighted averages of the probabilities of detection and kill of a nonsheltered aircraft. The new version of OPTSA has changed the functional form of f for this attrition calculation and has added new functional forms as options (these functions are discussed in Subsections 5 and 6 of this section, below). In the new functional form, the number of attackers is not qS, but pr_sqS, where r_s is the average sortie rate for airbase attackers (as discussed in Section B.1 of this chapter, above) and p is the number of passes per ABA sortie. Corresponding changes were made in the calculation of attrition of shelters.

The reason for adding a factor to account for the number of passes per sortie is as follows. If

$$\dot{T}_n = f(T_n, r_s qS, d_n, k_n) \ ,$$

with the same form for f as in Equation (4), then each of the $r_s qS$ attacking sorties can kill at most one target, even if $d_n = k_n = 1.0$. Yet it is possible for one sortie (1) to make p passes over the airbase and to drop 1/p of its ordnance on each pass, (2) to make one pass but shoot at p targets on that pass and drop 1/p of its ordnance on each target, or (3) to have any combination of passes and targets per pass that results in p targets. Based on the fraction of ordnance dropped on each target, the value for k_n could change, but, since the change in k_n may not be linear (and even if it were, $\dot{T}_n$ is not a linear function of k_n), there may be a significant difference (if $p > 1$) between

$$\dot{T}_n = f(T_n, r_s qS, d_n, k_n)$$

and

$$\dot{T}_n = f\big(T_n, p r_s qS, d_n, k_n(p)\big) \ ,$$

where $k_n(p)$ is the appropriate probability of kill per pass for an aircraft dropping 1/p of its ordnance per pass. OPTSA now uses the latter expression (above), where the form of f is as discussed in Subsection 5 of this section (below), p must be input, and $k_n(p)$ must be input in place of k_n (k_n is not automatically computed in OPTSA as a function of p). This is the same structure as used in IDAGAM I for multiple passes per aircraft.

5. Revised ABA Air-To-Ground Attrition Equation

The ABA air-to-ground attrition equation was revised in the new version of OPTSA along the same lines as the revisions to the air-to-air attrition equations. That is, the exponential form of the equation was replaced with the binomial form and the number of airbases on each side was included in the air-to-ground equations in the same manner as the number of air-combat areas was included in the air-to-air equation.[1] Just as in the original version of OPTSA, the air-to-ground equations are homogeneous equations that use weighted averages of the probabilities of detection and of kill, each of which is input as a function of type of aircraft (GP or SP-ABA).

In addition, two more changes to the air-to-ground attrition equations were made. First, a procedure was added that allows an ABA aircraft that is shooting at a particular nonsheltered enemy aircraft on the ground to kill other nonsheltered aircraft in the same general area. Specifically, suppose that each airbase has M different areas on which nonsheltered aircraft can be parked and that, if an airbase attacker shoots at an aircraft on a parking area, then the probability that he kills any particular nonsheltered aircraft on that parking area is k_n, independent of the rest of the attrition process. If there are B airbases and there are more nonsheltered aircraft on each airbase than there are parking areas, assume that there are $(T_n/B)/M$ nonsheltered aircraft on each parking area (i.e., the nonsheltered aircraft are uniformly distributed on the B airbases and on the M parking areas contained in each airbase). If there are fewer nonsheltered aircraft than there are parking areas, assume that there is one nonsheltered aircraft on each of T_n/B parking areas and none on the others. Based on these assumptions and changes, the new version of OPTSA computes the number of nonsheltered aircraft killed as

$$\dot{T}_n = T_n\left(1-\left[1 - \frac{k_n}{\min(M,T_n/B)}\left(1-[1-d_n]^{T_n/B}\right)\right]^{pr_s qS/B}\right) . \quad (8)$$

[1]As a result, d_n is no longer defined as the probability that an airbase attacker detects a particular nonsheltered aircraft on any airbase in the theater, but is now defined as the probability that an airbase attacker who is attacking a particular airbase detects a particular nonsheltered aircraft on that airbase.

Note that, if it is desired not to play these parking areas, then this can easily be done by inputting a very high value for M--which is equivalent to assuming that at most one nonsheltered aircraft can be killed per pass. It is assumed here than at most one sheltered aircraft can be killed per pass.

The second change to the air-to-ground attrition process is as follows: The original version of OPTSA computed the value for q (the fraction of ABA aircraft that attack nonsheltered aircraft) by making it proportional to the number of targets. That is, since attackers could distinguish occupied shelters from the unoccupied shelters in the original version, q equaled the number of nonsheltered aircraft divided by the number of aircraft on the ground. The new version of OPTSA now computes a value for q that maximizes the total number of aircraft killed on the ground each day. The total number of aircraft killed on the ground, $\dot{T}$, is given by

$$\dot{T} = T_n\left(1-\left[1 - \frac{k_n}{\min(M,T_n/B)}\left(1-[1-d_n]^{T_n/B}\right)\right]^{pr_s qS/B}\right)$$

$$+ T_s\left(1-\left[1 - \frac{k_s}{H/B}\left(1-[1-d_s]^{H/B}\right)\right]^{pr_s(1-q)S/B}\right) , \qquad (9)$$

where T_n, p, r_s, q, S, B, M, d_n, and k_n are as defined above; T_s is the total number of sheltered aircraft; H is the total number of shelters; d_s is the probability that an aircraft attacking a particular airbase detects a particular shelter on that airbase; and k_s is the probability of killing an aircraft in a shelter, given that the shelter is attacked.[1] The value for q that maximizes $\dot{T}$ such that $0 \le q \le 1$ is as follows: Let

$$K_n' = 1 - \frac{k_n}{\min(M,T_n/B)}\left(1-[1-d_n]^{T_n/B}\right) ;$$

---

[1] Note that d_n, k_n, d_s, and k_s are actually weighted averages of these probabilities.

$$K_n = [K_n']^{pr_s S/B} \quad ;$$

$$K_s = \left[1 - \frac{k_n}{H/B}\left(1-[1-d_s]^{H/B}\right)\right]^{pr_s S/B} \quad ;$$

$$K = \frac{T_n K_n \log (K_n)}{T_s \log (K_s)} \text{ ; and}$$

$$q_o = 1 - \frac{\log K}{\log K_s + \log K_n} .$$

Then

$$q = \begin{cases} 0, & \text{if } q_o < 0 \text{ ;} \\ q_o, & \text{if } 0 \leq q_o \leq 1 \text{ ;} \\ 1, & \text{if } q_o > 1 . \end{cases}$$

Note that this maximization assumes essentially that occupied and empty shelters are indistinguishable to an attacker.

6. New ABA Air-To-Ground Attrition Equations

In addition to the revised version of the ABA air-to-ground attrition equation given above, three new air-to-ground attrition equations have been added as options to OPTSA. These three options are (1) not to allocate airbase attackers optimally between attacking sheltered and nonsheltered aircraft, but (instead) to assume that each attacker will attack nonsheltered aircraft if he detects one (and he will attempt to attack sheltered aircraft only if he does not detect any nonsheltered aircraft); (2) to assume that both sheltered and nonsheltered aircraft are located in the same "parking areas," so that, if a particular area is attacked, then any aircraft--sheltered or not--on that area might be killed; (3) to assume that area fire is used by all airbase attackers. The equations for these options are described in turn (below).

The total number of passes made by airbase attackers at a particular airbase is $pr_s S/B$. If, on each pass, an attacker attempts to kill a sheltered aircraft only if he does not detect a nonsheltered aircraft, then the number of sheltered aircraft killed, $\dot{T}_s$, is given by

$$\dot{T}_s = T_s\left(1-\left[1 - \frac{k_s}{H/B}\left(1-[1-d_s]^{H/B}\right)(1-d_n)^{T_n/B}\right]^{pr_sS/B}\right) \quad (10)$$

where the $(1-d_n)^{T_n/B}$ factor accounts for the probability that none of the nonsheltered aircraft on the base is detected on a particular pass. The equation for the number of nonsheltered aircraft killed is the same as Equation (8) except that, instead of assuming that q of the pr_sS/B passes attempt to detect and kill nonsheltered aircraft, all passes attempt to do so. Thus,

$$\dot{T}_n = T_n\left(1-\left[1 - \frac{k_n}{\min(M,T_n/B)}\left(1-[1-d_n]^{T_n/B}\right)\right]^{pr_sS/B}\right). \quad (11)$$

This option assumes a simpler command-and-control procedure (i.e., attack nonsheltered aircraft if you see one; otherwise, attack a shelter) than the making of optimal allocations based on the number of sheltered and nonsheltered aircraft. Further, even though it is not "optimal," it might result in more aircraft killed on the ground, because each attacker has a chance to detect each target--sheltered or not. The equations in the previous subsection assume that, once allocated to one type of target, an attacker cannot detect and attack the other type of target, even if he detects no targets of the type he is allocated against.

The second new air-to-ground attrition option also assumes that attackers are not assigned in advance to attack either sheltered or nonsheltered aircraft. A weighted average of the detection probabilities is used; and, if an attacker attacks an aircraft (sheltered or not) on a particular parking area, then he has a probability of k_s of killing any particular sheltered aircraft on that parking area and a probability of k_n of killing any particular nonsheltered aircraft on that area. The equations used are

$$\dot{T}_n = T_n\left(1-\left[1 - \frac{k_n}{\min(M,T/B)}\left(1-\left[1 - \frac{d_nT_n + d_sH}{T}\right]^{T/B}\right)\right]^{pr_sS/B}\right) \quad (12)$$

and

$$\dot{T}_s = T_s\left(1-\left[1 - \frac{k_s}{\min(M,T/B)}\left(1-\left[1 - \frac{d_nT_n + d_sH}{T}\right]^{T/B}\right)\right]^{pr_sS/B}\right), \quad (13)$$

where T = total number of targets = T_n + H. Here, the mix of air munitions is not allowed to depend upon the number of enemy sheltered and nonsheltered aircraft.

The two options (above) used no new inputs; the previous inputs were just used in different ways. The last option (area fire) requires new inputs.

Area fire can be either perfectly coordinated, so that no two lethal areas overlap; perfectly uncoordinated, so that the center of the lethal areas are independently distributed over the area attacked; or somewhere in between. The area-fire option in OPTSA allows both extremes--and any linear mix of these extremes--as follows: Let α be an input such that, if α = 1.0, then the area fire is perfectly coordinated while, if α = 0.0, then the area fire is perfectly uncoordinated. The other new inputs are

- b = area of a typical airbase on which aircraft might be located;
- a_n = lethal area covered by one pass of an aircraft dropping "anti-nonsheltered" munitions against nonsheltered aircraft (this is taken as a weighted average by aircraft type GP or SP-ABA);
- a_s = lethal area covered by one pass of an aircraft dropping "anti-shelter" munitions against shelters (which is also a weighted average);

k_{ns} = a reduction factor applied to a_n when "anti-shelter" munitions are dropped on shelters; and

k_{sn} = an expansion (or reduction) factor applied to a_s when "anti-shelter" munitions are dropped on nonsheltered aircraft.

Note that it is assumed here that the targets (say, nonsheltered aircraft) are points and that the bombs have a lethal area (say, of a_n). At this level of detail, this assumption is equivalent to assuming that bombs are points and that targets have a vulnerable area (which, in the above formulation, would be a_n for nonsheltered aircraft). The equations used are

$$\dot{T}_n = T_n \min\left\{1.0, \frac{\alpha(A_n a_n + A_s a_s k_{sn})}{b} + (1-[1-x_n]^{A_n}[1-s_{sn}]^{A_s})\right\} \quad (14)$$

and

$$\dot{T}_s = T_s \min\left\{1.0, \frac{\alpha(A_n a_n k_{ns} + A_s a_s)}{b} + (1-[1-x_{ns}]^{A_n}[1-x_s]^{A_s})\right\} \quad ,(15)$$

where $A_n = qpr_s S/B$; $A_s = (1-q)pr_s S/B$;

$$x_n = \begin{cases} 0.0, & \text{if } \frac{(1-\alpha)a_n}{b} \leq 0.0; \\ 1.0, & \text{if } \frac{(1-\alpha)a_n}{b} \geq 1.0; \\ \frac{(1-\alpha)a_n}{b}, & \text{otherwise;} \end{cases}$$

$$x_{sn} = \begin{cases} 0.0, & \text{if } \frac{(1-\alpha)a_s k_{sn}}{b} \leq 0.0; \\ 1.0, & \text{if } \frac{(1-\alpha)a_s k_{sn}}{b} \geq 0.0; \\ \frac{(1-\alpha)a_s k_{sn}}{b}, & \text{otherwise;} \end{cases}$$

$$x_s = \begin{cases} 0.0, & \text{if } \dfrac{(1-\alpha)a_s}{b} \le 0.0; \\ 1.0, & \text{if } \dfrac{(1-\alpha)a_s}{b} \ge 1.0; \\ \dfrac{(1-\alpha)a_s}{b}, & \text{otherwise; and} \end{cases}$$

$$x_{ns} = \begin{cases} 0.0, & \text{if } \dfrac{(1-\alpha)a_n k_{ns}}{b} \le 0.0; \\ 1.0, & \text{if } \dfrac{(1-\alpha)a_n k_{ns}}{b} \ge 1.0; \\ \dfrac{(1-\alpha)a_n k_{ns}}{b}, & \text{otherwise.} \end{cases}$$

7. Destruction of Shelters

In the original version of OPTSA, shelters could not be destroyed. The new version of OPTSA assumes that an input fraction (one for each side) of the shelters that are hit (in such a way that aircraft inside of them would be destroyed) are themselves destroyed. The rest of the shelters (if any) that are hit are assumed to be reusable the next day.

8. Proportion of Aircraft Killed on the Ground

The original version of OPTSA assumed that if there were 10 GP, 10 SP-CAS, 10 SP-ABA, and 10 SP-INT aircraft on an airbase at the beginning of a day and that eight aircraft were killed on that airbase, then there would be two aircraft of each type killed on the airbase--no matter what happened in the air-to-air or the ground-to-air combat that day. The new version of OPTSA assumes that if there are 10 aircraft of each type on the airbase at the beginning of the day and that five SP-CAS and five SP-ABA aircraft are killed in air-to-air and ground-to-air combat that day (and no GP or SP-INT aircraft are killed) so that, after air-to-air and ground-to-air combat is assessed, there are 10 GP, five SP-CAS, five SP-ABA, and 10 SP-INT aircraft, and if eight aircraft are killed on the airbase, then three GP, one SP-CAS, one SP-ABA, and three SP-INT aircraft would be killed on the airbase.

E. FEBA MOVEMENT FUNCTION

Let x be the force ratio given by the Red (ground plus air) firepower divided by the Blue (ground plus air) firepower; and let f be a function mapping this force ratio into the movement of the FEBA. Then the input data (abscissa and ordinate breakpoints) for OPTSA used so far have had the property that $f(1/x) = -f(x)$. However, this property is not preserved by linear interpolation. The new version of OPTSA explicitly requires that it be preserved. Therefore, $f(x)$ for $x > 1$ is input as before; but OPTSA now calculates $f(x)$ for $x < 1$ by $f(x) = -f(1/x)$, and OPTSA ignores any inputs for the function f that affect FEBA movement for $x < 1$.

F. NEW MEASURES OF EFFECTIVENESS

1. Weighted Number of Surviving Aircraft

There are two new optional measures of effectiveness that are now available for use as the objective function for calculating the payoffs of the game that OPTSA plays. The first new measure of effectiveness is the weighted number of surviving aircraft, where the weights are both by side and by type of aircraft. Let B_g, B_c, B_a, and B_i be the number of Blue surviving aircraft of type GP, SP-CAS, SP-ABA, and SP-INT (resp.); and let R_g, R_c, R_a, and R_i be the same for Red. Then OPTSA now accepts the eight inputs α^b, β^b, γ^b, δ^b, α^r, β^r, γ^r, and δ^r; and OPTSA can calculate the payoff to Blue as

$$\alpha^b B_g + \beta^b B_b + \gamma^b B_a + \delta^b B_i - \alpha^r R_g - \beta^r R_c - \gamma^r R_a - \delta^r R_i \ .$$

2. Comprehensive Air Measure

The second new measure of effectiveness is a comprehensive air measure that includes (1) firepower delivered on CAS, (2) surviving aircraft, and (3) QRA levels. Let B_q be the desired number of aircraft that Blue wishes to keep on QRA; and let R_q be the same for Red. Note that, if B_g (as defined in the previous subsection) is greater than B_q, then B_g-B_q is the actual number of surviving Blue GP aircraft that can fly missions and Blue has enough surviving aircraft to meet his desired QRA level; if B_g is less than B_q, then Blue has no surviving GP aircraft that can fly missions and Blue is short B_q-B_g QRA aircraft. Thus, in this new measure, the number of surviving Blue GP aircraft is taken to be $\max\{B_g-B_q,0\}$, not B_g, and the Blue QRA shortfall is taken to be $\max\{B_q-B_g,0\}$.

Let B_f be the amount of Blue firepower delivered on all CAS missions; and let R_f be the same for Red. Then, in addition to the eight inputs given in the previous subsection, OPTSA can also accept the four inputs ε^b, ε^r, M^b, and M^r and can calculate the payoff to Blue as

$$\alpha^b(\max\{B_g-B_q,0\}) + \beta^b B_c + \gamma^b B_a + \delta^b B_i + \varepsilon^b B_f - M^b(\max\{B_q-B_g,0\})$$

$$-\alpha^r(\max\{R_g-R_q,0\}) - \beta^r R_c - \gamma^r R_a - \delta^r R_i - \varepsilon^r R_f + M^r(\max\{R_q-R_g,0\}).$$

Chapter IV

COMPUTATIONAL CONSIDERATIONS

To estimate the central processor (CP) time used by OPTSA, it suffices to determine the number of daily assessments fought and the time per daily assessment. As before, let there be D days in the war; three periods with d_1, d_2, and d_3 days in each; and n allocation choices per period per side.

The number of first-period payoff entries that must be computed is $n_1 = (r_1+c_1)n - r_1c_1$, where r_1 and c_1 are the number of rows and columns in the first-period game matrix whose entries must be computed to solve the game. Therefore, n_1 second-stage games must be solved. Each of these n_1 games might have a different number of payoff entries that need to be computed, but let n_2 be the average number. Then, on the average, n_1n_2 third-stage games must be solved. Thus, $1+n_1+n_1n_2$ games in all must be solved. On the average, let each third-stage game require computation of n_3 payoff entries for solution. Then $n_1n_2n_3$ war outcomes at the end of day D are found. The number of daily assessments computed is then $n_1d_1+n_1n_2d_2+n_1n_2n_3d_3$, on the average, because the model does not fight D days to find each of the $n_1n_2n_3$ war outcomes but stores results at the end of each period (as described in Sec. D of Ch. II, above). For a two-period war, the model solves $1+n_1$ games, computes n_1n_2 war outcomes, and computes $n_1d_1+n_1n_2d_2$ daily assessments.

What are reasonable estimates of n_1, n_2, and n_3, given n? The computational experience so far indicates that there is a wide variance in the number of rows and columns whose entries need to be computed to find the solution to a one-stage game.

The minimum, of course, is one row and one column (or 2n-1 payoff entries per game); the maximum is n^2 payoff entries. A good first guess of a Red strategy can eliminate some computation; however, it is often hard, *a priori*, to know whether a particular pure strategy will be active in the final solution. If it is not, at least one unnecessary column of payoff entries must be computed.

Experience with the two-period war suggests that for a fairly wide range of n (around 6 to 13), $n_2 = 3n - 2$ (two columns and one row, implying a saddle point) and $n_1 = 4n - 4$ (two columns and two rows) are reasonable estimates. In this case, the number of daily assessments is $n_1d_1 + n_1n_2d_2$--or $(3n-2)d_1 + (3n-2)(4n-4)d_2$. The computational experience with the three-period war has been more limited; but (with n = 7) running times from 65 to 150 CP seconds have been encountered. 3n - 2 (one row and two columns) still seems to be a good estimate of n_3, the number of *last*-period payoff entries required to solve a last- (third-) stage game, for n in a fairly wide range. It is difficult at this point to say anything about n_1 and n_2. However, it should be kept in mind that small shifts in the input data can and generally do dramatically affect the number of payoff entries required.[1] Note that in both two- and three-period wars there is a substantial reduction from the original version of the model, which calculated n^6D daily campaigns for the three-period war and n^4D for the two-period war.

Though the running time per daily campaign is constant throughout any particular game, it depends on the input data. The determining factor is whether or not SP aircraft are played. Several experiments were performed to determine timings. With one kind of Blue and Red division and only GP aircraft, 0.015 seconds (on the average) are required to compute one daily

[1] This is an interesting feature of the revised OPTSA model, as the running time of most models is independent of the data.

campaign. With three kinds of Blue and Red divisions and four kinds of Blue and Red aircraft, the average CP time per daily campaign rises to 0.027 seconds. Other input quantities do not seem to affect greatly the running time per daily campaign.

If the periods are equally spaced (i.e., $d_1 = d_2 = d_3$; or $d_1 = d_2$ for the two-period war), the CP time for the game is proportional to the number of days in the war. For a given value of D, the shorter the first periods and the longer the last periods, the longer the game will take to play (and vice versa). In general, larger values of D require longer running time; but, because of the period-spacing factor, the running time is not strictly monotone with D.

An estimate of the running time for a two-period war with six strategies per period per side (n = 6) can be obtained as follows: Let D = 30; $d_1 = 10$; and $d_2 = 20$. SP aircraft are played. The number of daily campaigns is

$$\begin{aligned} n_1 d_1 + n_1 n_2 d_2 &= (3n-2)d_1 + (3n-2)(4n-4)d_2 \\ &= (18-2)10 + (18-2)(24-4)20 \\ &= 160 + 6{,}400 \\ &= 6{,}560 \text{ daily campaigns at } 0.027 \text{ CP seconds each} \\ &= 177 \text{ CP seconds.} \end{aligned}$$

Actual running-times for games with these data have ranged from 150 to 185 CP seconds. With the original model,[1] using the improved assessment routine, $6^4 \times 30$ (or 38,880) daily campaigns at a total of 1,050 CP seconds would be required. The new game-solving method results in an 85-percent time savings.

Experience with three-period games has been limited, but running times of about 700 CP seconds have been encountered for 30-day wars. Since the original game-solving method used with

[1]The original version of OPTSA (documented in Reference [4]) could not play two-period wars; however, an intermediate version of the model (using the old game-solving and assessment procedures) did.

the new assessment routine would have taken (6^6 x 30 x 0.027 =) 37,800 CP seconds, there has been an improvement of 98 percent. We would also like to see, however, how the two opposing factors of shorter game-solving method and longer assessment time affect the actual running-time. A timing estimate of 0.003 CP seconds per daily campaign has been obtained for the original OPTSA model. Therefore, since the original model with the old assessment routine would take approximately (6^6 x 30 x 0.003 =) 4,199 CP seconds, the new model still appears to take considerably shorter time. For a two-period war, the original model would have taken (6^4 x 30 x 0.003 =) 117 CP seconds--comparable to the time of the new model.

The number n of pure strategies per period per side is obviously an important factor in the running time, which rises roughly as does n^2 for two-period or n^3 for three-period wars (as compared to n^4 or n^6 in the original model). Too few pure strategies do not allow a realistic range of allocations. The set of six allocations consisting of all combinations of all or half the GP aircraft to missions seems to work quite well. (The strategies are listed in the example in Vol. 2, Ch. II, Sec. C.) Excursions done on increasing the number of pure strategies did not dramatically affect the game value [5, Ch. I]. The model is currently dimensioned to hold up to 11 pure strategies per side (e.g., 10 "normal" ones and one that it might be interesting to try) and can easily be redimensioned to hold up to 20, if sufficient computer-core storage is available.

Chapter V

LIMITATIONS OF THE REVISED OPTSA MODEL

There are three types of limitations to the revised OPTSA model: limitations to the game-solving procedure (given the current assessment procedure), limitations to the assessment procedure (given the current game-solving procedure), and limitations that affect both the game-solving and assessment procedures. Some limitations of each type are listed below. Within each type the limitations are listed in order of importance.

A. LIMITATIONS OF THE GAME-SOLVING PROCEDURE

The limitations of the game-solving procedure are as follows:

(1) Limited Number of Decision Periods. OPTSA can consider only two- or three-period wars (with a variable number of days in each period). Therefore, if the war being modeled lasts longer than three days, OPTSA cannot optimize the allocation of aircraft to missions for each day of the war. Theoretically, a side may be able to achieve a much better payoff with a strategy that changes on each day of the war than it can achieve with the strategy that is optimal among the set of strategies that do not change within each period. It should also be noted that the periods must begin and end on the same days for each side (Blue cannot change strategy on day 10 and Red on day 15 of a 30-day war), and that a mixed strategy is played as a mixture of period-long pure strategies. For example, if the first period lasts 10 days, OPTSA can play that GP aircraft fly CAS on all of the first

ten days with probability 1/2, and fly ABA on all of those days with probability 1/2. But OPTSA cannot consider strategies that allow GP aircraft to fly CAS with probability 1/2 and to fly ABA with probability 1/2 on each of the ten days independently of what was flown on the other days.

(2) Limited Strategy Space. The pure strategies considered by OPTSA must be input and must be fractional assignments of sorties to missions. Theoretically, it is possible that the optimal Blue strategy is to fly 3/8ths of its GP aircraft on CAS and 5/8ths on ABA, yet if this strategy is not on the input list, OPTSA cannot consider it and will only select the best of strategies that are on the input list. Also, given the first limitation above, it might be reasonable to play that the Blue commander tells 1/2 of his GP aircraft to fly CAS throughout a period and 1/2 to fly ABA throughout that period. However, OPTSA cannot consider this strategy if the period is longer than one day and if the CAS attrition differs from the ABA attrition. For example, suppose that Blue has 2,000 GP aircraft with a sortie rate of one and that aircraft suffer two-percent attrition on CAS each day and ten-percent attrition on ABA each day. Then, if 1,000 aircraft were told to fly CAS each day and 1,000 were told to fly ABA each day, on the second day there would be 980 aircraft flying CAS and 900 flying ABA, and on the third day there would be 960.4 aircraft flying CAS and 810 flying ABA, and so on. OPTSA cannot play this strategy; it can play that 1/2 of the sorties fly CAS each day and 1/2 fly ABA each day, which results in 940 aircraft flying each mission on day 2 and 883.6 aircraft flying each mission on day 3, and so on. Finally, it should be noted that the set of allowable strategies can be different for each side but this set must be the same for all of the periods played. For example, Red cannot be required to fly all ABA missions during period one simply by inputting only that strategy only for period one.

B. LIMITATIONS OF THE ASSESSMENT PROCEDURE

The limitations of the assessment procedure are as follows:

(1) Notional Types of Aircraft (Capabilities). OPTSA plays only four types of aircraft which are differentiated according to allowable mission assignments. It cannot play different types of aircraft which have the same allowable mission assignments, but have much different air-to-air or air-to-ground capabilities. For example, both an F-4 and an F-104 can be considered as GP aircraft, yet they have much different capabilities. Using average parameters to reflect an average capability as OPTSA requires (rather than using parameters that reflect the capabilities of each type of aircraft) might lead to significantly different results in all aspects of air combat. Using average parameters also makes preparing and changing inputs difficult and makes priority sheltering of a particular type of aircraft generally impossible. Playing various types of aircraft according to capabilities within the four basic mission assignment types currently played by OPTSA would require only changing the assessment procedure. (The optimization posture would only have to be changed if each different type of aircraft could be assigned differently, as will be discussed below.) However, adding new types of aircraft within the four mission assignment types might significantly increase the computer running time of OPTSA.

(2) SAMs. OPTSA plays ground-to-air attrition only by an input attrition rate. SAMs and AAA could be played directly. Various numbers of different types of SAMs and AAA could be played at various locations (barriers, area defenses, point defenses, etc.) and SAMs could either kill or suppress attacking aircraft.

(3) Constant Air Firepower. Each unit of air firepower delivered (no matter on which day it is delivered or how much

was delivered that day) counts the same in contributing towards the MOEs which use air firepower delivered. However, the second unit of firepower delivered on a day might not be worth as much as the first unit on that day, and the third not worth as much as the second, and so on. Similarly, firepower delivered on day 2 may not be worth as much as the same amount delivered on day 1, and firepower delivered on day 3 might not be worth as much as the same amount delivered on day 2, and so on. OPTSA cannot play this discounting of firepower. Also, OPTSA must play fixed length wars where the length of the war must be determined in advance of the combat. A major advantage of being able to discount firepower by day is that it allows random length wars to be played where only the probability distribution of the length of the war need be known in advance. For example, suppose the value of a unit of firepower deliverd on day t is assumed to be V_t if the war has not ended before day t. Let

P_t = the probability that the length of the war is $\geq$ t.

Values for P_t would have to be found, but since $P_t \geq P_{t+1}$, reasonable guesses could be made. And guessing values for P_t is generally better, and always at least as good, than being able to play only fixed length wars. (Playing a 30-day war, for example, is equivalent to assuming that $P_t = 1$ for $t \leq 30$ and $P_t = 0$ for $t > 30$.) If OPTSA could discount firepower, then the value of a unit of firepower on day t could be input as P_tV_t instead of as some constant, and by this device OPTSA could play random length wars instead of only being able to play fixed length wars.

(4) Air Munitions. Different air munitions can have different firepower values (and can result in different P_k's for ABA missions) and might be available in different quantities. And the rate of consumption of air munitions could depend on the numbers and types of missions flown on each day. OPTSA cannot

address this problem and must assume that the same notional loads of air munitions are carried throughout the war.

(5) Shelters Independent of FEBA Position. Aircraft can move back and forth as the FEBA moves back and forth (if there are sufficiently many suitable airbases). But shelters are essentially fixed in one position on the ground and can become too close or too far from the FEBA to be useful. OPTSA does not play that shelters remain in a fixed position as the FEBA moves. As OPTSA stands now, this limitation is not significant if there is relatively heavy air-to-air attrition. However, if several locations for airbases and corresponding considerations were played, then this limitation could be significant.

(6) Force Ratios. OPTSA uses the standard form of the force ratio for calculating losses to ground units. This ratio is of the form "air plus ground" divided by "air plus ground". While this form is standard, it is not symmetric in terms of shooters and targets, and this form can lead to certain anomalies. A nonstandard force ratio, as described in Reference [1], should at least be available as an option.

(7) Changeable Parameters. OPTSA can only change two parameters (sortie rates and fraction of day on ground) during the course of the war, and these parameters can only be changed once during a war. A more general structure would be to allow all parameters to be changed as often as desired.

C. LIMITATIONS WHICH AFFECT BOTH PROCEDURES

Those limitations which affect both procedures are as follows:

(1) One Notional Location for Airbases. OPTSA cannot play that some airbases are closer to the FEBA (and hence closer to the enemy's airbases) than others, and this is perhaps the most

critical limitation of OPTSA. If an aircraft can fly an ABA mission from a particular airbase, then it can move farther to the rear and fly a CAS mission with the same amount of fuel, and aircraft based farther to the rear are less vulnerable to enemy ABA missions for two reasons. First, enemy ABA aircraft have to fly over more friendly territory to reach rearward airbases and so are more vulnerable to detection and kill by more friendly interceptors, SAMs, and AAA. Second, and more importantly, to reach rearward airbases enemy aircraft have to carry more fuel and less payload, and so are not as effective when they reach the rearward airbases (if they can carry enough fuel to reach them at all). Thus, the effectiveness of aircraft is inherently related to the basing of the aircraft. The commander must choose where his aircraft are to be based and, for ABA missions, how deep into enemy territory his aircraft are to fly. These are interrelated decisions. OPTSA cannot optimize the basing of aircraft or how far into enemy territory these aircraft are to fly. Even for a fixed basing of aircraft, OPTSA cannot calculate an optimal allocation which considers significant differences in distances between aircraft and the FEBA (if the particular basing has these differences). To consider this limitation would require changing both the game-solving and the assessment procedures.

(2) Deterministic Attrition Processes. One strength of OPTSA is the way it handles mixed (stochastic) strategies. But this strength is somewhat mitigated by the fact that it requires deterministic attrition equations. Stochastic considerations are significant when there is a possibility of a shortage of aircraft,[1] and one reason for optimizing over aircraft missions is because there is a possible shortage. (For example, if a commander could replace all his losses on a one-for-one basis,

---

[1]Many runs of OPTSA have shown that at least one side soon has a severe shortage of aircraft due to heavy attrition.

but could not bring in new aircraft except as replacements, then all aircraft would probably always be sent on CAS for ground-related MOEs.) Thus, a significant reason for optimizing aircraft assignments implies a need to play stochastic attrition processes. A stochastic model might also handle other important stochastic phenomena, such as weather. The limitation that OPTSA must play deterministic attrition processes may be second in significance only to its not being able to play several locations for airbases on each side.

(3) Notional Types of Aircraft (Allocations). In the previous section, the limitation of only being able to play different capabilities for four notional types of aircraft was discussed. If OPTSA were able to play more than four types of aircraft in the assessment procedure, then the next step would be to allocate different types of aircraft to missions according to their individual capabilities. For example, if OPTSA could play ten types of aircraft, then the allocation of each of these ten types could be optimized. The significant difference in capabilities among the various (actual) types of aircraft makes this an important limitation of OPTSA.

(4) Missions for Aircraft. OPTSA can only play three missions for aircraft (CAS, ABA, and INT). There are several other possible missions that are related to the air war. INT can be divided into Battlefield Defense and Airbase Defense. There are SAM suppression missions and there are escort missions for all of the attacking missions. Further, it may be optimal under some circumstances for aircraft to stay on the ground and not fly any missions on some days, or to take off on warning of an attack in order not to be caught on the ground (even if this means that the aircraft cannot fly other missions that day). Since CAS is a surrogate for all air attacks on ground units, not being able to play interdiction missions is not a significant limitation of OPTSA.

(5) Attacking Shelters Versus Attacking Nonsheltered Aircraft. OPTSA allows two options to determine whether an ABA attacker shoots at a shelter or at a nonsheltered aircraft (and perhaps a better method to determine this decision can be found). But whatever method is used, both the defender and the attacker have an allocation problem to solve. The defender must decide how many aircraft he should shelter and how many he should leave out in the open (OPTSA now assumes that he shelters as many of the aircraft as he can). The attacker must decide whether to allocate (in some sense) his aircraft against shelters or against nonsheltered aircraft. These are the only allocation decisions in OPTSA that the model does not optimize. Since they are not optimized, it is theoretically possible, for example, for Blue to do worse with more shelters than he does with slightly fewer shelters (all else being equal). This type of anomaly can be fixed for all possible cases only by optimizing these allocation decisions.

REFERENCES

[1] Anderson, L. B., J. Bracken, J. G. Healy, M. J. Hutzler, and E. P. Kerlin. *IDA Ground-Air Model I (IDAGAM I).* IDA Report R-199. Arlington, Va.: Institute for Defense Analyses, May 1974.

[2] Anderson, L. B., D. Bennett, and M. J. Hutzler. *Modifications to IDAGAM I.* IDA Paper P-1034. Arlington, Va.: Institute for Defense Analyses, June 1974.

[3] Assistant Chief of Staff, Studies and Analysis, U.S. Air Force. *Methodology for Use in Measuring the Effectiveness of General Purpose Forces (An Algorithm for Approximating the Game Theoretic Value of N-Staged Games), SABER GRAND (ALPHA).* March 1971.

[4] Bracken, J. *Two Optimal Sortie Allocation Models.* IDA Paper P-992. Arlington, Va.: Institute for Defense Analyses, December 1973.

[5] Bracken, J., J. F. Dinwiddie (WSEG), C. R. Gentzel (WSEG), L. B. Anderson, E. L. Schwartz, R. J. Shanker, *A Theater-Level Analysis of NATO Aircraft Shelters and Air Defense Attrition Rates* (U). IDA Study S-447 (also WSEG Report 264), SECRET. Arlington, Va.: Institute for Defense Analyses, May 1975.

[6] Bracken, J., A. F. Karr, E. L. Schwartz, R. J. Shanker, H. E. Strickland, and M. J. Zusman. *Quantity-Quality Tradeoffs.* IDA Study S-443. Arlington, Va.: Institute for Defense Analyses, January 1975.

[7] Control Analysis Corporation. *Approximate Solutions of Multi-Stage Network Games.* Palo Alto, Calif.: Control Analysis Corporation, August 1974.

[8] ——— (contributors: Z. F. Lansdowne, G. B. Dantzig, R. P. Harvey, and R. D. McKnight). *Development of an Algorithm to Save Multi-Stage Games.* Palo Alto, Calif.: Control Analysis Corporation, May 1973.

[9] Karr, A. F. *Adaptive and Behavioral Strategies in a Two-Person, Two-Move, Two-Action Zero-Sum Games.* IDA Paper

P-993. Arlington, Va.: Institute for Defense Analyses, December 1973.

[10] ———. *On a Class of Binomial Attrition Processes.* IDA Paper P-1031. Arlington, Va.: Institute for Defense Analyses, June 1974.

[11] Lulejian and Associates. *The Lulejian-I Theater-Level Model.* WSEG Report 259. Torrance Calif.: Lulejian and Associates, October 1974.

APPENDIX

DOCUMENTATION OF ASSESSMENT PROCEDURE

Appendix

DOCUMENTATION OF ASSESSMENT PROCEDURE

This appendix gives an algebraic description of the assessment routine. The purpose of the routine is--with a given combination of allocations of GP aircraft to missions (for each side, over all the periods)--to fight a campaign to compute a measure of effectiveness that becomes a payoff entry in a final-stage game corresponding to the allocations. There are three different kinds of inputs to the assessment procedure. First, there is the fixed set of effectiveness parameters (including firepower per division; sortie rates; firepower per CAS sortie; probabilities, by aircraft type, of detection and kill by attack and defense aircraft in the INT interaction; proportions of aircraft destroyed by enemy SAMs; probability of detection and kill of sheltered and non-sheltered aircraft in the ABA interaction; FEBA advance as a function of force ratio; and divisional casualties as a function of force ratio). These are input once at the beginning of the program. Second, there are the time-varying numbers of Blue and Red divisions, by type; aircraft, by type; and shelters. At the beginning of the program, the number of days in the war and streams of Blue and Red divisions and aircraft *additions* occurring on each day of the war (including day 1) are specified. However, the *inventories* of divisions, aircraft, and shelters computed at the beginning of each day are what actually affect the results. Finally, of course, are the allocations of Blue and Red GP aircraft. The game-solving routine determines the choice of allocation from lists of feasible allocations input at the beginning of the program.[1]

The assessment routine is set up as a large DO loop (indexed by day of the war) and is performed period by period; that is, the lower and upper limits of the day index are the first and last days of some period in the war, determined by the game-solving routine. Each day the assessment proceeds as follows: First, the starting division inventory is computed as the starting division inventory of the previous day, minus destruction the previous day, plus new divisions arriving for the present day. The ground firepower is computed from the division inventory and ground-firepower scores (the sum of divisions times firepower scores). Starting aircraft inventory is computed in the same way as starting division inventory. Aircraft allocations to missions are computed from fractional allocations times inventories for the GP aircraft and from total allocations for the SP aircraft. Total aircraft assigned to missions are computed. The air-to-air assessment is then performed with the aircraft assigned to INT encountering the aircraft assigned to CAS and ABA. An input proportion of the attackers remaining after the air-to-air assessment are destroyed by enemy SAMs. The aircraft still remaining are computed for use in the air-to-ground (ABA) computation. Inventories of sheltered and non-sheltered aircraft (by type) are established, and the number of aircraft and shelters destroyed are computed by using one of four possible attack modes (the choice of attack mode for each side is input). The total number of aircraft killed (by type) is computed as the number of aircraft of each type destroyed air-to-ground (sheltered plus nonsheltered), plus the aircraft of each type destroyed in the air-to-air interaction, plus the aircraft of each type destroyed by enemy SAMs. The air firepower

[1]If the fractional allocations sum to less than 1, the GP aircraft not assigned to missions are computed. They are still vulnerable to enemy ABA.

is computed in the same way as ground firepower (sum of successful CAS sorties times firepower scores).

The measures of effectiveness and ground-forces destruction are then computed by using the firepowers and inventories. Total firepower is computed as ground-plus-air firepower. FEBA position is computed as the previous FEBA position plus or minus the movement. Division destruction is computed as division inventory times percent casualties, assessed on the basis of force ratio. Finally, cumulative total firepower and air firepower are computed. To save computer-storage space, the two new MOEs are computed in the game-solving (rather than the assessment) routine.

A. DEFINITIONS OF INPUT QUANTITIES[1]

${}^{B}D_{\ell_B t}$ = Blue type-ℓ_B divisions on day t.

${}^{R}D_{\ell_R t}$ = Red type-ℓ_R divisions on day t.

x_{mt} = Blue type-m aircraft on day t (m=1 - GP aircraft; m=2 - CAS aircraft; m=3 - ABA aircraft; m=4 - INT aircraft).

y_{nt} = Red type-n aircraft on day t (n=1 - GP aircraft; n=2 - CAS aircraft; n=3 - ABA aircraft; n=4 - INT aircraft).

Q_B' = desired number of Blue QRA aircraft.

Q_R' = desired number of Red QRA aircraft.

S_t^B = number of Blue aircraft shelters at beginning of day t.

S_t^R = number of Red aircraft shelters at beginning of day t.

${}^{B}f_{\ell_B}$ = firepower per Blue type-ℓ_B division.

${}^{R}f_{\ell_R}$ = firepower per Red type-ℓ_R division.

---

[1]In order of use in the assessment routine.

${}^{B}C_{m}$ = firepower per Blue CAS sortie performed by type-m aircraft.

${}^{R}C_{n}$ = firepower per Red CAS sortie performed by type-n aircraft.

${}^{B}t_{\xi}$ = day on which Blue sortie rates are to change.

${}^{R}t_{\xi}$ = day on which Red sortie rates are to change.

The variables for the air-to-air war are indexed by i, j, and k (as appropriate): i stands for type of aircraft (1 - GP; 2 - SP--the kind is specified by variable j); j stands for mission (1 - CAS; 2 - ABA; 3 - INT); and k stands for type and mission of an attack aircraft (1 - GP-CAS; 2 - GP-ABA; 3 - SP-CAS; 4 - SP-ABA).

${}^{B}\xi^{1}_{ij}$ = Blue sortie rates before day ${}^{B}t_{\xi}$, by aircraft type and mission (Blue surge sortie rates).

${}^{B}\xi^{2}_{ij}$ = Blue sortie rates on or after day ${}^{B}t_{\xi}$, by aircraft type and mission (Blue sustained sortie rates).

${}^{R}\xi^{1}_{ij}$ = Red sortie rates before ${}^{R}t_{\xi}$, by aircraft type and mission (Red surge sortie rates).

${}^{R}\xi^{2}_{ij}$ = Red sortie rates on or after day ${}^{R}t_{\xi}$, by aircraft type and mission (Red sustained sortie rates).

A_{B} = number of identical notional air-to-air combat regions on Blue side of FEBA.

A_{R} = number of identical notional air-to-air combat regions on Red side of FEBA.

${}^{B}\alpha_{ij}$ = proportion of engaged Blue attack aircraft that keep their ordnance and fly on to their missions, if second method of air-to-air attrition is used (j=1 or 2--i.e., attack missions [CAS or ABA] only).

${}^{R}\alpha_{ij}$ = proportion of engaged Red attack aircraft that keep their ordnance and fly on to their missions.

There are 64 air-to-air effectiveness parameters (i.e., 32 detection and 32 kill) over type of Red and Blue interceptor (i=1,2) and attacker (k=1,2,3,4). The detection parameters are probabilities that a *particular* aircraft of a given type detects (and engages) a particular enemy aircraft of a given type. The kill parameters give probabilities of kill, given detection. They are inputs to a binomial attrition equation (Ref. [10]).

${}^{BI}d_{ik}$ = probability that Blue type-i interceptor detects Red type-k attacker (Blue side of FEBA).

${}^{BA}d_{ki}$ = probability that Blue type-k attacker detects Red type-i interceptor (Red side of FEBA).

${}^{RI}d_{ik}$ = probability that Red type-i interceptor detects Blue type-k attacker (Red side of FEBA).

${}^{RA}d_{ki}$ = probability that Red type-k attacker detects Blue type-i interceptor (Blue side of FEBA).

${}^{BI}\kappa_{ik}$ = probability (given detection) that Blue type-i interceptor kills Red type-k attacker (Blue side of FEBA).

${}^{BA}\kappa_{ki}$ = probability (given detection) that Blue type-k attacker kills Red type-i interceptor (Red side of FEBA).

${}^{RI}\kappa_{ik}$ = probability (given detection) that Red type-i interceptor kills Blue type-k attacker (Red side of FEBA).

${}^{RA}\kappa_{ki}$ = probability (given detection) that Red type-k attacker kills Blue type-i interceptor (Blue side of FEBA).

If the second method of air-to-air attrition is used, the parameters for attackers detecting enemy interceptors are not needed--as an attacker shoots at an interceptor only if the attacker is engaged.

${}^{B}\sigma_{ij}$ = proportion of Red attack aircraft, by type and attack mission (j=1 - CAS; j=2 - ABA) destroyed by Blue SAMs.

${}^{R}\sigma_{ij}$ = proportion of Blue attack aircraft, by type and attack mission (j=1 - CAS; j=2 - ABA) destroyed by Red SAMs.

${}^{B}\phi^{1}$ = fraction of Blue non-QRA aircraft on the airbase (the remainder are out flying missions) before day ${}^{B}t_{\xi}$.

${}^{B}\phi^{2}$ = fraction of Blue non-QRA aircraft on the airbase on or after day ${}^{B}t_{\xi}$.

${}^{R}\phi^{1}$ = fraction of Red non-QRA aircraft on the airbase before day ${}^{R}t_{\xi}$.

${}^{R}\phi^{2}$ = fraction of Red non-QRA aircraft on the airbase on or after day ${}^{R}t_{\xi}$.

γ_B = fraction of Blue shelters hit that are destroyed.

γ_R = fraction of Red shelters hit that are destroyed.

${}^{B}\pi_{i}$ = number of passes per Blue ABA sortie, by type of plane (1 - GP; 2 - SP-ABA).

${}^{R}\pi_{i}$ = number of passes per Red ABA sortie, by type of plane (1 - GP; 2 - SP-ABA).

$\mathcal{B}_B$ = number of Blue identical notional airbases.

$\mathcal{B}_R$ = number of Red identical notional airbases.

M_B = number of Blue parking areas for aircraft (nonsheltered aircraft only for Red attack modes 1 and 2; sheltered and nonsheltered aircraft for Red attack mode 3), per notional airbase.

M_R = number of Red parking areas for aircraft (nonsheltered aircraft only for Blue attack modes 1 and 2; sheltered and nonsheltered aircraft for Blue attack mode 3), per notional airbase.

For the point-fire attack modes (1, 2, and 3), there are 16 input effectiveness parameters.

${}^{B}d_{i}^{S}$ = probability that Blue type-i attack pass (1 - GP; 2 - SP) detects Red shelter.

${}^{B}d_{i}^{N}$ = probability that Blue type-i attack pass (1 - GP; 2 - SP) detects Red nonsheltered aircraft.

${}^{B}k_{i}^{S}$ = probability that Blue type-i attack pass (1 - GP; 2 - SP) hits Red shelter.

${}^{B}k_i^N$ = probability that Blue type-i attack pass (1 - GP; 2 - SP) kills Red nonsheltered aircraft.

${}^{R}d_i^S$ = probability that Red type-i attack pass (1 - GP; 2 - SP) detects Blue shelter.

${}^{R}d_i^N$ = probability that Red type-i attack pass (1 - GP; 2 - SP) detects Blue nonsheltered aircraft.

${}^{R}k_i^S$ = probability that Red type-i attack pass (1 - GP; 2 - SP) hits Blue shelter.

${}^{R}k_i^N$ = probability that Red type-i attack pass (1 - GP; 2 - SP) kills Blue nonsheltered aircraft.

If the area fire-attack mode (mode 4 explained in Section D.6 of Chapter III, above) is to be used, then inputs are needed for the appropriate airbases; if both sides use mode 4, 20 inputs are required. A "B" on a variable refers to events taking place at the Blue airbases (including the effectiveness of Red munitions on Blue planes). (The notation for Red is analogous.)

ω_B = overlap factor (between 0 and 1 for Red munitions at Blue airbase.

b_B = area (in square meters) of a typical airbase on which Blue aircraft might be located.

${}^{B}a_n^1, {}^{B}a_n^2$ = lethal area covered by one pass of a Red GP- or SP-ABA aircraft (resp.) dropping "anti-nonsheltered" munitions against nonsheltered aircraft.

${}^{B}a_s^1, {}^{B}a_s^2$ = lethal area covered by one pass of a Red GP- or SP-ABA aircraft (resp.) dropping "anti-shelter" munitions against shelters.

${}^{B}k_{ns}^1, {}^{B}k_{ns}^2$ = a reduction factor applied to ${}^{B}a_n^1$ or ${}^{B}a_n^2$ (resp.) when "anti-nonsheltered" munitions are dropped on shelters.

${}^{B}k_{sn}^1, {}^{B}k_{sn}^2$ = an expansion (or reduction) factor applied to ${}^{B}a_s^1$ or ${}^{B}a_s^2$ (resp.) when "anti-shelter" munitions are dropped on nonsheltered aircraft.

ω_R = overlap factor (between 0 and 1) for Blue munitions at Red airbase.

b_R = area of a typical airbase on which Red aircraft might be located.

${}^{R}a_n^1, {}^{R}a_n^2$ = lethal area covered by one pass of a Blue GP- or SP-ABA aircraft (resp.) dropping "anti-nonsheltered" munitions against nonsheltered aircraft.

${}^{R}a_s^1, {}^{R}a_s^2$ = lethal area covered by one pass of Blue GP- or SP-ABA aircraft (resp.) dropping "anti-shelter" munitions against shelters.

${}^{R}k_{ns}^1, {}^{R}k_{ns}^2$ = a reduction factor applied to ${}^{R}a_n^1$ or ${}^{R}a_n^2$ (resp.) when "anti-nonsheltered" munitions are dropped on shelters.

${}^{R}k_{sn}^1, {}^{R}k_{sn}^2$ = an expansion (or reduction) factor applied to ${}^{R}a_s^1$ and ${}^{R}a_s^2$ (resp.) when "anti-shelter" munitions are dropped on nonsheltered aircraft.

$F(\cdot)$ = function for FEBA advance per subperiod as a function of the ratio of Blue firepower to Red firepower.

$g^B(\cdot)$ = function for percent Blue division destruction per subperiod as a function of the ratio of Blue firepower to Red firepower.

$g^R(\cdot)$ = function for percent Red division destruction per subperiod as a function of the ratio of Blue firepower to Red firepower.

$\beta_{1t}, \beta_{2t}, \beta_{3t}$ = fraction of Blue GP aircraft assigned to CAS, ABA, INT on day t, $\sum_i \beta_{it} \le 1.0$ (usually equal).

$\rho_{1t}, \rho_{2t}, \rho_{3t}$ = fraction of Red GP aircraft assigned to CAS, ABA, INT on day t, $\sum_i \rho_{it} \le 1.0$ (usually equal).

If measure of effectiveness 4 (surviving aircraft) or 5 (generalized air measure, including penalty for QRA deficiency) is to be used, the following weights must be input:

${}^{B}w^C$ = weight for cumulative Blue air firepower delivered.

${}^{B}w_j^S$ = weights for Blue SP surviving aircraft, mission j (j=1 to 3).

${}^{B}w_i^Q$ = for MOE 4: weight for surviving Blue GP aircraft (i=1 only); for MOE 5: i = 1, weight for GP aircraft surviving minus desired QRA and, i = 2, weight for desired QRA minus actual QRA.

${}^{R}w^C$ = weight for cumulative Red air firepower delivered.

${}^{R}w_j^S$ = weights for Red SP surviving aircraft, mission j (j=1 to 3).

${}^{R}w_i^Q$ = for MOE 4: weight for surviving Red GP aircraft (i=1 only); for MOE 5: i = 1, weight for GP aircraft surviving minus desired QRA and i = 2, weight for desired QRA minus actual QRA.

B. PROCEDURE AND DEFINITIONS OF COMPUTED QUANTITIES

The following 24-step procedure is performed for each day:

(1) Compute starting division inventory for day t as starting division inventory for day t-1, minus division destruction during day t-1, plus divisions added for day t:

$$ {}^{B}D_{\ell_B t} = {}^{B}D_{\ell_B t-1} - {}^{B}D^{d}_{\ell_B t-1} + {}^{B}D^{a}_{\ell_B t} \qquad \text{for all } \ell_B \text{ ;} $$

$$ {}^{R}D_{\ell_R t} = {}^{R}D_{\ell_R t-1} - {}^{R}D^{d}_{\ell_R t-1} + {}^{R}D^{a}_{\ell_R t} \qquad \text{for all } \ell_R \text{ ,} $$

where the superscript d denotes destruction and the superscript a denotes addition. (The number of kinds of Blue and Red divisions are input.)

(2) Compute ground firepower for day t as the sum of divisions times their firepower scores:

$$ G_t^B = \sum_{\ell_B} {}^{B}D_{\ell_B t} \times {}^{B}f_{\ell_B} \text{ ;} $$

$$ B_t^R = \sum_{\ell_R} {}^{R}D_{\ell_R t} \times {}^{R}f_{\ell_R} \text{ .} $$

(3) Compute shelter inventory for day t as starting shelter inventory of previous day minus shelters destroyed on previous day:

$$S_t^B = S_{t-1}^B - \dot{S}_{t-1}^B$$

$$S_t^R = S_{t-1}^R - \dot{S}_{t-1}^R$$

(4) Compute starting aircraft inventory for subperiod t as starting aircraft inventory for subperiod t-1, minus aircraft destruction during subperiod t-1 plus aircraft added for subperiod t:

$$x_{mt} = x_{m,t-1} - x_{m,t-1}^d + x_{mt}^a \qquad \text{for all } m \ ;$$

$$y_{nt} = y_{n,t-1} - y_{n,t-1}^d + y_{nt}^a \qquad \text{for all } n \ ,$$

where the superscript d denotes destruction and the superscript a denotes addition. (The number of kinds of Blue and Red aircraft are input. However, most of the assessment routine assumes that all four kinds of aircraft are played, and the number of aircraft is automatically set to zero for all kinds of aircraft not specifically input.)

(5) QRA.

Some of the *GP* aircraft are set aside as QRA:

$$Q_B = \begin{cases} Q_B', & \text{if } Q_B' \le x_{1t} \ ; \\ x_{1t}, & \text{if } Q_B' > x_{1t} \ ; \end{cases}$$

$$Q_R = \begin{cases} Q_R', & \text{if } Q_R' \le y_{1t} \ ; \\ y_{1t}, & \text{if } Q_R' > y_{1t} \ . \end{cases}$$

The remaining GP aircraft are assignable to missions.

(6a) Compute aircraft assignments for day t:

$x_{11} = \beta_{1t}(x_{1t}-Q_B)$ (Blue GP aircraft to CAS)

$x_{12} = \beta_{2t}(x_{1t}-Q_B)$ (Blue GP aircraft to ABA)

$x_{13} = \beta_{3t}(x_{1t}-Q_B)$ (Blue GP aircraft to INT)

$x_{21} = x_{2t}$ (Blue CAS aircraft to CAS)

$x_{22} = x_{3t}$ (Blue ABA aircraft to ABA)

$x_{23} = x_{4t}$ (Blue INT aircraft to INT)

$y_{11} = \rho_{1t}(y_{1t}-Q_R)$ (Red GP aircraft to CAS)

$y_{12} = \rho_{2t}(y_{1t}-Q_R)$ (Red GP aircraft to ABA)

$y_{13} = \rho_{3t}(y_{1t}-Q_R)$ (Red GP aircraft to INT)

$y_{21} = y_{2t}$ (Red CAS aircraft to CAS)

$y_{22} = y_{3t}$ (Red ABA aircraft to ABA)

$y_{23} = y_{4t}$ (Red INT aircraft to INT)

(6b) Compute GP aircraft not flying missions:

$$\tilde{x}_1 = (x_{1t}-Q_B)(1-\sum_{j=1}^{3} \beta_{jt})$$

$$\tilde{y}_1 = (t_{1t}-Q_R)(1-\sum_{j=1}^{3} \rho_{jt})$$

(7) Sorties.

(a) Find appropriate sortie rate:

$$\left.\begin{array}{l} B_{\xi_{ij}} = \begin{cases} B_{\xi_{ij}^1}, & \text{if } t < {}^{B}t_{\xi} \\ B_{\xi_{ij}^2}, & \text{if } t \geq {}^{B}t_{\xi} \end{cases} \\ \\ R_{\xi_{ij}} = \begin{cases} R_{\xi_{ij}^1}, & \text{if } t < {}^{R}t_{\xi} \\ R_{\xi_{ij}^2}, & \text{if } t \geq {}^{R}t_{\xi} \end{cases} \end{array}\right\} \begin{array}{l} i=1,2 \text{ (aircraft type: GP or SP)};\\ j=1,2,3 \text{ (mission)} \end{array}$$

At this point in the computer program, the fraction of aircraft on base is also computed--to be used in Steps (13) and (15):

$$B_\phi = \begin{cases} B_{\phi 1}, & \text{if } t < {}^{B}t_\xi \ ; \\ B_{\phi 2}, & \text{if } t \geq {}^{B}t_\xi \ ; \end{cases}$$

$$R_\phi = \begin{cases} R_{\phi 1}, & \text{if } t < {}^{R}t_\xi \ ; \\ R_{\phi 2}, & \text{if } t \geq {}^{R}t_\xi \ . \end{cases}$$

(b) Compute number of sorties, by aircraft type and mission:

$$u_{ij} = {}^{B}\xi_{ij} \times x_{ij} \quad \text{(Blue sorties)}$$

$$v_{ij} = {}^{R}\xi_{ij} \times y_{ij} \quad \text{(Red sorties)}$$

(c) Compute number of aircraft not flying (if sortie rate is less than one), by aircraft type and mission:

$$x_{ij}^{NF} = \begin{cases} x_{ij}(1-{}^{B}\xi_{ij}), & \text{if } {}^{B}\xi_{ij} < 1.0 \ ; \\ 0.0, & \text{otherwise.} \end{cases}$$

$$y_{ij}^{NF} = \begin{cases} y_{ij}(1-{}^{R}\xi_{ij}), & \text{if } {}^{R}\xi_{ij} < 1.0 \ ; \\ 0.0, & \text{otherwise.} \end{cases}$$

For example, if the sortie rate for Red SP-ABA aircraft is 0.8, 20 percent of these aircraft stay on the ground; thus, $v_{22} = 0.8y_{22}$ and $y_{22}^{NF} = (1.0-0.8)y_{22} = 0.2y_{22}$.

(d) Compute total sorties, used as shooters and targets in attrition equations:

$U_A = u_{11} + u_{12} + u_{21} + u_{22}$ (Blue attack [CAS and ABA] sorties)

$U_I = u_{13} + u_{23}$ (Blue intercept sorties)

$V_A = v_{11} + v_{12} + v_{21} + v_{22}$ (Red attack [CAS and ABA] sorties)

$V_I = v_{13} + v_{23}$ (Red intercept sorties)

(8) **Average detection parameters for air-to-air interaction** (average is taken over target type and is, therefore, a function of shooter type):

$$ {}^{BI}\bar{d}_i = \left(\sum_{k=1}^{4} {}^{BI}d_{ik} v_{i'j'} \right) \Big/ V_A $$ (Probability that a Blue type-i INT detects a Red attacker target)

$$ {}^{BA}\bar{d}_k = \left(\sum_{i=1}^{2} {}^{BA}d_{ki} v_{i3} \right) \Big/ V_I $$ (Probability that a Blue type-k attacker detects a Red interceptor)

$$ {}^{RI}\bar{d}_i = \left(\sum_{k=1}^{4} {}^{RI}d_{ik} u_{i'j'} \right) \Big/ U_A $$ (Probability that a Red type-i INT detects a Blue attacker target)

$$ {}^{RA}\bar{d}_k = \left(\sum_{i=1}^{2} {}^{RA}d_{ki} u_{i3} \right) \Big/ U_I $$ (Probability that a Red type-k attacker detects a Blue interceptor)

The index k for kind of attacker is computed as $j' + 2(i'-1)$, where i' is the type of attack aircraft (1 - GP; 2 - SP)--not to be confused with the type of interceptor i--and j'=1 or 2 (i.e., attack mission CAS or ABA). ${}^{BA}\bar{d}_k$ and ${}^{RA}\bar{d}_k$ are not computed if the second method of attrition is used.

(9) **Attrition from the first method of air-to-air interaction.** Quantities ${}^{AA}\dot{u}_{ij}$ and ${}^{AA}\dot{v}_{ij}$ (i=1,2; j=1,2,3) are found. If the second method of attrition is to be used, perform Step (10) instead of this step.

On the Blue side of the FEBA, Blue interceptors oppose Red attackers. A two-sided heterogeneous binominal equation is used:

$$ {}^{AA}\dot{v}_{i'j'} = v_{i'j'} \left(1 - \prod_{i=1}^{2} \left[1 - \frac{{}^{BI}\kappa_{ik}}{(V_A/A_B)} \left(1 - [1 - {}^{BI}\bar{d}_i]^{(V_A/A_B)} \right) \right]^{(u_{i3}/A_B)} \right) $$

(Red attackers killed, where i'=1,2; j'=1,2; (attack missions--CAS and ABA) only; and the index k is defined as $j'+2(i'-1)$, the kind of attacker, as before.)

$AA_{\dot{u}_{i3}}$

$$= u_{i3}\left(1 - \prod_{k=1}^{4}\left[1 - \frac{{}^{RA}\kappa_{ki}}{(U_I/A_B)}\left(1-[1-{}^{RA}\bar{d}_k]^{(U_I/A_B)}\right)\right]^{(v_{i'j'}/A_B)}\right)$$

(Blue interceptors killed: i=1 - GP; 2 - SP)

Note that in these equations the inside product is taken over *shooter* type, whether attacker or interceptor. On the other side of the FEBA, Red interceptors oppose Blue attackers.

$AA_{\dot{u}_{i'j'}}$

$$= u_{i'j'}\left(1 - \prod_{i=1}^{2}\left[1 - \frac{{}^{RI}\kappa_{ik}}{(U_A/A_R)}\left(1-[1-{}^{RI}\bar{d}_i]^{(U_A/A_R)}\right)\right]^{(v_{i3}/A_R)}\right)$$

(Blue attackers killed: i´=1,2; j´=1,2; k=j´+2(i´-1))

$AA_{\dot{v}_{i3}}$

$$= v_{i3}\left(1 - \prod_{k=1}^{4}\left[1 - \frac{{}^{BA}\kappa_{ki}}{(V_I/A_R)}\left(1-[1-{}^{BA}\bar{d}_k]^{(V_I/A_R)}\right)\right]^{(u_{i'j'}/A_R)}\right)$$

(Red interceptors killed: i=1,2)

(10) **Attrition from the second method of air-to-air interaction.**

The rationale for this method is explained in Chapter III, Section B (above). The attritions to attackers are as in Step (9) above, but attritions to interceptors are found differently.

On the Blue side of the FEBA, the number of Red attackers killed air-to-air, by type and mission, is

$$AA_{\dot{v}_{i'j'}}$$

$$= v_{i'j'}\left(1 - \prod_{i=1}^{2}\left[1 - \frac{{}^{BI}\kappa_{ik}}{(V_A/A_B)}\left(1-[1-{}^{BI}\bar{d}_i]^{(V_A/A_B)}\right)\right]^{(u_{i3}/A_B)}\right)$$

$$i=1,2;\ j=1,2 \ .$$

The number of Red attackers engaged, by type and mission, is

$$v^e_{i'j'}$$

$$= v_{i'j'}\left(1 - \prod_{i=i}^{2}\left[1 - \frac{1}{(V_A/A_B)}\left(1-[1-{}^{BI}\bar{d}_i]^{(V_A/A_B)}\right)\right]^{(u_{i3}/A_B)}\right).$$

Essentially, the kill parameters have been replaced by 1.0 to determine engagements. $(1-{}^{R}\alpha_{i'j'})$ of the engaged attack sorties jettison their ordnance and fight back at the Blue intercept sorties that have engaged them. Of the Blue intercept sorties that have engaged Red attack sorties, the proportion

$$^{B}p_1 = \frac{u_{13}\,{}^{B}\bar{d}_1}{u_{13}\,{}^{B}\bar{d}_1 + u_{23}\,{}^{B}\bar{d}_2}$$

are GP; the proportion

$$^{B}p_2 = \frac{u_{23}\,{}^{B}\bar{d}_2}{u_{13}\,{}^{B}\bar{d}_1 + u_{23}\,{}^{B}\bar{d}_2}$$

are SP-INT.

The number of Blue type-i INT sorties killed is therefore

$$AA_{\dot{u}_{i3}} = \sum_{i'=1}^{2}\sum_{j'=1}^{2}\left((1-{}^{R}\alpha_{i'j'})v^e_{i'j'}\right){}^{RA}\kappa_{ki}\,{}^{B}p_i \qquad i=1,2 \ ,$$

where $(1-{}^{R}\alpha_{i'j'})v^{e}_{i'j'}$ = number of Red shooters;

${}^{RA}\kappa_{ki}$ = kill parameter of Red shooters against type-i targets; and

${}^{B}p_{i}$ = proportion of type-i targets.

On the Red side of the FEBA,

$$
{}^{AA}\dot{u}_{i'j'}
$$

$$
= u_{i'j'}\left(1 - \prod_{i=1}^{2}\left[1 - \frac{{}^{RI}\kappa_{ik}}{(U_A/A_R)}\left(1-[1-{}^{RI}\bar{d}_i]^{(U_A/A_R)}\right)\right]^{(v_{i3}/A_R)}\right);
$$

(i´=1,2; j´=1,2 [Blue CAS,ABA];
k = kind of Blue attacker = j´+2(i´-1)--
Blue attackers killed)

$$
u^{e}_{i'j'}
$$

$$
= u_{i'j'}\left(1 - \prod_{i=1}^{2}\left[1 - \frac{1}{(U_A/A_R)}\left(1-[1-{}^{RI}\bar{d}_i]^{(U_A/A_R)}\right)\right]^{(v_{i3}/A_R)}\right);
$$

(Blue attackers engaged)

and

$$
{}^{AA}\dot{v}_{i3} = \sum_{i'=1}^{2}\sum_{j'=1}^{2}(1-{}^{B}\alpha_{i'j'})u^{e}_{i'j'}\,{}^{BA}\kappa_{ki}\,{}^{R}p_{i} \qquad i=1,2 ,
$$

(Red interceptors killed)

where $${}^{R}p_{1} = \frac{v_{13}{}^{R}\bar{d}_{1}}{v_{13}{}^{R}\bar{d}_{1} + v_{23}{}^{R}\bar{d}_{2}} \text{ ; and } {}^{R}p_{2} = \frac{v_{23}{}^{R}\bar{d}_{2}}{v_{13}{}^{R}\bar{d}_{1} + v_{23}{}^{R}\bar{d}_{2}} .$$

(11) Subtract out losses from air-to-air interaction.

(a) If the first method of attrition is used,

$$u_{ij}^{new} = u_{ij}^{old} - {}^{AA}\dot{u}_{ij} \qquad i=1,2;\ j=1,3 \text{ (Blue sorties)}$$

$$v_{ij}^{new} = v_{ij}^{old} - {}^{AA}\dot{v}_{ij} \qquad i=1,2;\ j=1,3 \text{ (Red sorties)}$$

(b) If the second method of attrition is used, first define

$$u_{ij}^{FB} = (1-{}^{B}\alpha_{ij})(u_{ij}^{e}-{}^{AA}\dot{u}_{ij})$$ (Blue engaged attackers that jettisoned their ordnance and survived to fly back to Blue airbase)

$i=1,2;\ j=1,2$ (attack missions [CAS and ABA] only; variables u_{i3}^{FB} exist but are set to zero);

and

$$v_{ij}^{FB} = (1-{}^{R}\alpha_{ij})(v_{ij}^{e}-{}^{AA}\dot{v}_{ij}) \qquad v_{i3}^{FB}=0;\ i-1,2$$

(Red engaged attackers that fly back to Red airbase).

Then compute the number of sorties that can go on to deliver ordnance.

$$u_{ij}^{new} = u_{ij}^{old} - {}^{AA}\dot{u}_{ij} - u_{ij}^{FB} \qquad i=1,2;\ j=1,3 \text{ (Blue)}$$

$$v_{ij}^{new} = v_{ij}^{old} - {}^{AA}\dot{v}_{ij} - v_{ij}^{FB} \qquad i=1,2;\ j=1,3 \text{ (Red)}$$

(c) To convert a number of sorties to a number of aircraft, divide the number of sorties by the maximum of 1.0 and the appropriate sortie rate. With sortie rates less than 1.0, it is extremely important that the operations be performed exactly as follows:

$${}^{AA}\dot{x}_{ij} = {}^{AA}\dot{u}_{ij}/\max\{1.0,{}^{B}\xi_{ij}\} \quad \text{(Blue aircraft killed)}$$

$${}^{AA}\dot{y}_{ij} = {}^{AA}\dot{v}_{ij}/\max\{1.0,{}^{R}\xi_{ij}\} \quad \text{(Red aircraft killed)}$$

$$x_{ij}^{FB} = u_{ij}^{FB}/\max\{1.0, {}^{B}\xi_{ij}\}$$ (Blue aircraft that fly back)

$$y_{ij}^{FB} = v_{ij}/\max\{1.0, {}^{R}\xi_{ij}\}$$ (Red aircraft that fly back)

(d) Compute the number of Blue and Red aircraft remaining to perform their mission.

$$x_{ij}^{new} = x_{ij}^{old} - x_{ij}^{NF} - x_{ij}^{FB} - {}^{AA}\dot{x}_{ij} \qquad i=1,2;\ j=1,3$$

$$y_{ij}^{new} = y_{ij}^{old} - y_{ij}^{NF} - y_{ij}^{FB} - {}^{AA}\dot{y}_{ij} \qquad i=1,2;\ j=1,3$$

(12) **Ground-to-air interaction.**

First, compute sorties; then, aircraft lost to enemy SAMs:

(a)
$$\left.\begin{aligned} {}^{GA}\dot{u}_{ij} &= {}^{R}\sigma_{ij}u_{ij} \\ {}^{GA}\dot{v}_{ij} &= {}^{B}\sigma_{ij}v_{ij} \end{aligned}\right\} \begin{aligned} &i=1,2; \\ &j=1,2 \text{ (attack missions only)} \end{aligned}$$

(b) Convert sorties to aircraft:

$${}^{GA}\dot{x}_{ij} = {}^{GA}\dot{u}_{ij}/\max\{1.0, {}^{B}\xi_{ij}\}$$

$${}^{GA}\dot{y}_{ij} = {}^{GA}\dot{v}_{ij}/\max\{1.0, {}^{B}\xi_{ij}\}$$

(c) Subtract out losses:

$$u_{ij}^{new} = u_{ij}^{old} - {}^{GA}\dot{u}_{ij}$$ (Blue sorties remaining to perform CAS and ABA)

$$v_{ij}^{new} = v_{ij}^{old} - {}^{GA}\dot{v}_{ij}$$ (Red sorties remaining to perform CAS and ABA)

$$x_{ij}^{new} = x_{ij}^{old} - {}^{GA}\dot{x}_{ij}$$ (Blue aircraft remaining)

$$y_{ij}^{new} = y_{ij}^{old} - {}^{GA}\dot{y}_{ij}$$ (Red aircraft remaining)

(13) **ABA--Blue airbases.**

There are an input number β_B of *identical* notional airbases. The total population of aircraft computed at the start are assumed to be uniformly spread among the airbases.

The indexing is done by *kind* of Blue aircraft m (m=1 - GP; 2 - SP-CAS; 3 - SP-ABA; 4 - SP-INT). First, populations of sheltered and nonsheltered aircraft (by kind), the number of unoccupied shelters, and the number of Red attackers (GP- and SP-ABA) are established.

(a) Compute initial Blue aircraft inventory by kind (not counting QRA):

$$B_1 = \sum_{j=1}^{3} \left[x_{ij} + x_{ij}^{NF} + x_{ij}^{FB} \right] + \tilde{x}_1 \quad \text{(GP)}$$

$$B_m = x_{2,m-1} + x_{2,m-1}^{NF} + x_{2,m-1}^{FB} \qquad m=2,3,4 \text{ (SP-CAS, SP-ABA, SP-INT)}$$

$$B' = B_1 + B_2 + B_3 + B_4$$

(b) Shelter Blue QRA aircraft:

$$Q_B^S = \begin{cases} Q_B, & \text{if } Q_B \leq S_t^B ; \\ S_t^B, & \text{otherwise.} \end{cases}$$

$$Q_B^{NS} = Q_B - Q_B^S$$

$$S_t'^B = S_t^B - Q_B^S \quad \text{(Blue shelters available for non-QRA aircraft)}$$

(c) Shelter remaining aircraft proportionally, by kind kind. Reduce by the fraction $^B\phi$ of aircraft on base:

$$\left.\begin{aligned} B_m^S &= \min\{B', S_t{}^B\} \times \frac{B_m}{B'} \\ B_m^{NS} &= (B_m - B_m^S) \times {}^B\phi \\ B_m^{S^{new}} &= B_n^{S^{old}} \times {}^B\phi \end{aligned}\right\} \quad m=1,2,3,4$$

Add in QRA to GP aircraft population:

$$B_1^S = B_1^S + Q_B^S \; ;$$

$$B_1^{NS} = B_1^{NS} + Q_B^{NS} \; .$$

Find total inventories:

$$B^S = \sum_{m=1}^{4} B_m^S \; ;$$

$$B^{NS} = \sum_{m=1}^{4} B_m^{NS} \; ;$$

$$B = B^S + B^{NS} \; .$$

Therefore, there are a number of sheltered aircraft, a number of nonsheltered aircraft, and a number of shelters (hence, a number of unoccupied shelters).

(d) Compute the number of Red attackers:

$$R_1 = v_{12}\,{}^{R}\pi_1 \quad \text{(GP attack passes)}$$

$$R_2 = v_{22}\,{}^{R}\pi_2 \quad \text{(SP attack passes)}$$

$$R = R_1 + R_2$$

(Recall that v_{12} = Red ABA sorties surviving interceptors and SAMs.)

(e) Compute the average Red effectiveness parameters for the point-fire equations:

$${}^{R}\bar{d}^{S} = \left(R_1\,{}^{R}d_1^S + R_2\,{}^{R}d_2^S\right)/R$$

$${}^{R}\bar{k}^{S} = \left(R_1\,{}^{R}k_1^S + R_2\,{}^{R}k_2^S\right)/R$$

$${}^{R}\bar{d}^{N} = \left(R_1\,{}^{R}d_1^N + R_2\,{}^{R}d_2^N\right)/R$$

$${}^{R}\bar{k}^{N} = \left(R_1\,{}^{R}k_1^N + R_2\,{}^{R}k_2^N\right)/R$$

(14) **Attrition to Blue from ABA.**

First, the desired Red attack mode is input. In modes 2 and 4, Red must make a decision on the proportion of its attack passes to use against shelters, in mode 2, or to load with anti-shelter munitions, in mode 4; the remaining passes attack nonsheltered aircraft, or load with anti-nonsheltered munitions. In modes 1 and 3, a prior allocation of passes is not made. In all cases, the inputs to the attrition equations are B^S, B^{NS}, and S_t^B (the number of Blue sheltered and nonsheltered aircraft) and R_1, R_2, and $R=R_1+R_2$ (the number of GP, SP, and total Red attack passes), plus the point- or area-fire effectiveness parameters. The outputs are $\dot{B}^S$, $\dot{B}^{NS}$, and $\dot{S}_t^B$ (the numbers of Blue sheltered aircraft, nonsheltered aircraft, and shelters destroyed on day t).[1]

(a) **Red Attack Mode 1.** First, let

$$f = 1 - \left[1 - \frac{R_k^S}{S_t^B/\mathcal{B}_B}\left(1-[1-R_d^S]^{S_t^B/\mathcal{B}_B}\right)\left(1-R_d^N\right)^{B^{NS}/\mathcal{B}_B}\right]^{R/\mathcal{B}_B}$$

The number of Blue shelters *hit* is

$$\dot{S}_t^{B'} = fS_t^B .$$

Of the S_t^B shelters, B^S are occupied; hence the number of Blue aircraft killed is

$$\dot{B}^S = \frac{B^S}{S_t^B} \times \dot{S}_t^{B'} = fB^S ,$$

since, if a shelter is hit, an aircraft in it is destroyed.

---

[1]In Sections D.5 and D.6 of Chapter III (above), the various attack modes and corresponding attrition equations are described in some detail, but in the following order: the basic mode described first is mode 2, then the point-fire modes 1 and 3—which differ only in that, in mode 3, both sheltered and nonsheltered aircraft are located on parking areas while, in mode 1, only nonsheltered aircraft are located on parking areas--are described. Mode 4 (area fire) is described last.

The number of Blue shelters destroyed is an input fraction of the number of hit shelters:

$$\dot{S}_t^B = \gamma_B \dot{S}_t^{B'} = \gamma_B f S_t^B \ .$$

The number of nonsheltered aircraft killed is

$$\dot{B}^{NS}$$

$$= B^{NS}\left(1 - \left[1 - \frac{R_{\overline{k}}N}{\min(M_B, B^{NS}/\mathcal{B}_B)}\left(1 - [1 - R_{\overline{d}}N]^{B^{NS}/\mathcal{B}_B}\right)\right]^{R/\mathcal{B}_B}\right) .$$

(b) Red Attack Mode 2. Let q be the proportion of Red attack passes[1] that attack Blue shelters; and let 1 - q be the proportion that attack nonsheltered aircraft.[2]. Given q, let

$$f = 1 - \left[1 - \frac{R_{\overline{k}}S}{S_t^B/\mathcal{B}_B}\left(1 - [1 - R_{\overline{d}}S]^{S_t^B/\mathcal{B}_B}\right)\right]^{qR/\mathcal{B}_B}$$

$$\dot{B}^S = fB^S$$

$$\dot{S}_t^B = \gamma_B f S_t^B$$

$$\dot{B}^{NS}$$

$$= B^{NS}\left(1 - \left[1 - \frac{R_{\overline{k}}N}{\min(M_B, B^{NS}/\mathcal{B}_B)}\left(1 - [1 - R_{\overline{d}}N]^{B^{NS}/\mathcal{B}_B}\right)\right]^{(1-q)R/\mathcal{B}_B}\right) .$$

The q that maximizes $\dot{B}^S + \dot{B}^{NS}$ is found as follows:

---

[1]GP and SP passes are divided in the same proportion.

[2]This is the notation used in the computer program. In Section D.5 of Chapter III of this volume (above), q is the proportion attacking nonsheltered aircraft.

Let

$$K_N' = 1 - \frac{R_{\overline{k}}^N}{\min(M_B, B^{NS}/\mathcal{B}_B)}\left(1-[1-R_{\overline{d}}^N]^{(B^{NS}/\mathcal{B}_B)}\right)$$

$$K_N = K_N'^{(R/\mathcal{B}_B)}$$

$$K_S = \left[1 - \frac{R_{\overline{k}}^S}{(S_t^B/\mathcal{B}_B)}\left(1-[1-R_{\overline{d}}^S]^{(S_t^B/\mathcal{B}_B)}\right)\right]^{(R/\mathcal{B}_B)}$$

$$K = \frac{B^{NS} \times K_N \times \ln(K_N)}{B^S \times \ln(K_S)}$$

$$q_0 = \frac{\ln K}{\ln K_N + \ln K_S}$$

$$q = \begin{cases} 0, & \text{if } q_0 < 0 ; \\ q_0, & \text{if } q_0 \in [0,1] ; \\ 1, & \text{if } q_0 > 1 . \end{cases}$$

(c) **Red Attack Mode 3.** Let $B' = B^{NS} + S_t^B$, the total number of targets:

$$g = 1 - \left[1 - \frac{R_{\overline{d}}^N \times B^{NS} + R_{\overline{d}}^S \times S_t^B}{B'}\right]^{B'/\mathcal{B}_B}$$

$$f = 1 - \left[1 - \frac{R_{\overline{k}}^S\, g}{\min(M_B, B'/\mathcal{B}_B)}\right]^{R/\mathcal{B}_B}$$

Then $\dot{B}^S = fB^S$; $\dot{S}_t^B = \gamma_B f S_t^B$; and

$$\dot{B}^{NS} = B^{NS}\left(1 - \left[1 - \frac{R_{\overline{k}}^{NS}\, g}{\min(M_B, B'/\mathcal{B}_B)}\right]^{R/\mathcal{B}_B}\right).$$

(d) Red Attack Mode 4 requires the area-fire inputs for the airbase. First, average over shooter type:

$$^{B}a_s = (R_1 \times {}^{B}a_s^1 + R_2 \times {}^{B}a_s^2)/R \; ;$$

$$^{B}a_n = (R_1 \times {}^{B}a_n^1 + R_2 \times {}^{B}a_n^2)/R \; ;$$

$$^{B}k_{ns} = (R_1 \times {}^{B}k_{ns}^1 + R_2 \times {}^{B}k_{ns}^2)/R \; ;$$

$$^{B}k_{sn} = (R_1 \times {}^{B}k_{sn}^1 + R_2 \times {}^{B}k_{sn}^2)/R \; .$$

Let q be the proportion of Red attack passes that load with munitions against shelters,[1] and let

$$R^S = \frac{qR_1 + qR_2}{\mathcal{B}_B} = qR/\mathcal{B}_B$$

$$R^{NS} = \frac{(1-q)R_1 + (1-q)R_2}{\mathcal{B}_B} = (1-q)R/\mathcal{B}_B \; .$$

Define quantities[2]

$$^{B}x_n = \begin{cases} 0.0, & \text{if } (1-\omega_B)\,^{B}a_n/b_B < 0 \; ; \\ 1.0, & \text{if } (1-\omega_B)\,^{B}a_n/b_B > 1 \; ; \\ (1-\omega_B)\,^{B}a_n/b_B, & \text{otherwise.} \end{cases}$$

$$^{B}x_{sn} = \begin{cases} 0.0, & \text{if } (1-\omega_B)\,^{B}a_s\,^{B}k_{sn}/b_B < 0 \; ; \\ 1.0, & \text{if } (1-\omega_B)\,^{B}a_s\,^{B}k_{sn}/b_B > 1 \; ; \\ (1-\omega_B)\,^{B}a_s\,^{B}k_{sn}/b_B, & \text{otherwise.} \end{cases}$$

---

[1]This notation is compatible with the computer program. In Chapter III (above), q is the proportion loading with munitions against nonsheltered aircraft. GP and SP passes are divided in the same proportion.

[2]The letter "x" used here is not to be confused with the prior use of x as Blue aircraft levels.

$$
{}^{B}x_{s} = \begin{cases} 0.0, & \text{if } (1-\omega_B)\,{}^{B}a_s/b_B < 0\ ; \\ 1.0, & \text{if } (1-\omega_B)\,{}^{B}a_s/b_B > 1\ ; \\ (1-\omega_B)\,{}^{B}a_s/b_B, & \text{otherwise.} \end{cases}
$$

$$
{}^{B}x_{ns} = \begin{cases} 0.0, & \text{if } (1-\omega_B)\,{}^{B}a_n\,{}^{B}k_{ns}/b_B < 0\ ; \\ 1.0, & \text{if } (1-\omega_B)\,{}^{B}a_n\,{}^{B}k_{ns}/b_B > 1\ ; \\ (1-\omega_B)\,{}^{B}a_n\,{}^{B}k_{ns}/b_B), & \text{otherwise.} \end{cases}
$$

$$
\text{Let } f = \min\left\{1.0,\ \frac{\omega_B(R^{NS}x\,{}^{B}a_n\,x\,{}^{B}k_{ns}+R^{S}x\,{}^{B}a_s)}{b_B} + \left(1-\left[1-{}^{B}x_{ns}\right]^{R^{NS}}\left[1-{}^{B}x_s\right]^{R^{S}}\right)\right\}.
$$

Then $\dot{B}^S = fB^S$; $\dot{S}_t^B = \gamma_B f S_t^B$; and

$$
\dot{B}^{NS} = B^{NS}\min\left\{1.0,\ \frac{\omega_B(R^{NS}x\,{}^{B}a_n+R^{S}x\,{}^{B}a_s\,x\,{}^{B}k_{sn})}{b_B} + \left(1-\left[1-{}^{B}x_n\right]^{R^{NS}}\left[1-{}^{B}x_{sn}\right]^{R^{S}}\right)\right\}.
$$

The q that maximizes $\dot{B}^S + \dot{B}^{NS}$ is found using Newton's method;[1] however, before Newton's method is used, checks are used to determine whether the optimal q is, as is often the case, 0 or 1.

---

[1]Actually, the function maximized is the sum of the second terms within the braces. It seems that these terms are practically always less than 1.0; hence, the maximization performed is nearly equivalent to maximizing $\dot{B}^S + \dot{B}^{NS}$.

If, after 100 iterations of Newton's method, successive estimates are further than an input epsilon apart, the program stops (see Section D of Ch. I in Vol. 2).

(15) ABA--Red Airbases.

There are B_R identical notional Red airbases. The only difference from the procedure for Blue is that there is an option for no Red SP-ABA aircraft to be sheltered. This results in the following procedure for setting up populations of aircraft--Step (13), above.

$$R_1 = \sum_{j=1}^{3} (y_{1j}+y_{1j}^{NF}+y_{1j}^{FB}) + \tilde{y}_1 \;;$$

$$R_n = y_{2,n-1} + y_{2,n-1}^{NF} + y_{2,n-1}^{FB} \qquad n=2 \text{ to } 4 \;;$$

$$R' = \begin{cases} R_1 + R_2 + R_3 + R_4, & \text{if Red SP-ABA aircraft are to be sheltered;} \\ R_1 + R_2 + R_4, & \text{otherwise;} \end{cases}$$

$$Q_R^S = \begin{cases} Q_R, & \text{if } Q_R - S_t^R \;; \\ S_t^R, & \text{otherwise;} \end{cases}$$

$$Q_R^{NS} = Q_R - Q_R^S; \text{ and}$$

$$S_t^{\prime R} = S_t^R - Q_R^S \;.$$

Shelter remaining Red non-SP-ABA aircraft proportionally, by kind. Reduce by fraction ${}^R\phi$ of aircraft on base:

$$R_n^S = \min\{R', S_t^{\prime R}\} \times \frac{R_n}{R'} \qquad n=1 \text{ to } 4 \;.$$

If Red SP-ABA are not to be sheltered, $R_3^S = 0$.

$$\left.\begin{aligned} R_n^{NS} &= (R_n - R_n^S)\,{}^R\phi \\ R_n^{S^{new}} &= R_n^{S^{old}} \times {}^R\phi \end{aligned}\right\} \qquad n=1 \text{ to } 4$$

$$R_1^S = R_1^S + Q_R^S \;;$$

$$R_1^{NS} = R_1^{NS} + Q_R^{NS} \;;$$

$$R^S = \sum_{n=1}^{4} R_n^S \;;$$

$$R^{NS} = \sum_{n=1}^{4} R_n^{NS} ;$$

$$R = R^S + R^{NS} .$$

Define Blue attack passes:

$$B_1 = u_{12} \times {}^B\pi_1 ;$$

$$B_2 = u_{22} \times {}^B\pi_2 ;$$

$$B = B_1 + B_2 ;$$

and average Blue effectiveness parameters for point-fire equations:

$${}^B\bar{d}^S = (B_1 \times {}^B d_1^S + B_2 \times {}^B d_2^S)/B ;$$

$${}^B\bar{k}^S = (B_1 \times {}^B k_1^S + B_2 \times {}^B k_2^S)/B ;$$

$${}^B\bar{d}^N = (B_1 \times {}^B d_1^N + B_2 \times {}^B d_2^N)/B ;$$

$${}^B\bar{k}^N = (B_1 \times {}^B k_1^N + B_2 \times {}^B k_2^N)/B .$$

(16) Attrition equations for Red are *exactly* the same, ***mutatis mutandis***, as for Blue--using the input Blue attack mode.

(a) **Blue Attack Mode 1.**

$$f = 1 - \left(1 - \frac{{}^B\bar{k}^S}{S_t^R/\mathcal{B}_R}\left(1 - \left[1 - {}^B\bar{d}^S\right]^{S_t^R/\mathcal{B}_R}\right)\left(1 - {}^B\bar{d}^N\right)^{R^{NS}/\mathcal{B}_R}\right)^{B/\mathcal{B}_R}$$

$\dot{S}_t^{R'} = fS_t^R$ (Red shelters hit)

$\dot{R}^S = fR^S$ (Red sheltered aircraft killed)

$\dot{S}_t^R = \gamma_R f S_t^R$ (Red shelters destroyed)

$$\dot{R}^{NS} = R^{NS}\left(1 - \left[1 = \frac{B_{\bar{k}}^{N}}{\min(\mathcal{M}_R, R^{NS}/\mathcal{B}_R)}\left(1 - \left[1 - B_{\bar{d}}^{N}\right]^{R^{NS}/\mathcal{B}_R}\right)\right]^{B/\mathcal{B}_R}\right).$$

(Red nonsheltered aircraft destroyed)

(b) **Blue Attack Mode 2.** Let q be the proportion of Blue attack passes that attack Red shelters:

$$f = 1 - \left[1 - \frac{B_{\bar{k}}^{S}}{S_t^R/\mathcal{B}_R}\left(1 - \left[1 - B_{\bar{d}}^{S}\right]^{S_t^R/\mathcal{B}_R}\right)\right]^{qB/\mathcal{B}_R} ;$$

$$\dot{R}^{S} = fR^{S} ;$$

$$\dot{S}_t^{R} = \gamma_R f S_t^R ;$$

$$\dot{R}^{NS} = R^{NS}\left(1 - \left[1 - \frac{B_{\bar{k}}^{N}}{\min(\mathcal{M}_R, R^{NS}/\mathcal{B}_R)}\left(1 - \left[1 - B_{\bar{d}}^{N}\right]^{R^{NS}/\mathcal{B}_R}\right)\right]^{(1-q)B/\mathcal{B}_R}\right).$$

The q that maximizes $\dot{R}^S + \dot{R}^{NS}$ is found in a manner exactly analogous to the method in Step (14b), *mutatis mutandis.*

(c) **Blue Attack Mode 3.** Let $R' = R^{NS} + S_t^R$, the total number of targets:

$$g = 1 - \left[1 - \frac{B_{\bar{d}}^{N} \times R^{NS} + B_{\bar{d}}^{S} \times S_t^R}{R'}\right]^{R'/\mathcal{B}_R} ;$$

$$f = 1 - \left[1 - \frac{B_{\bar{k}}^{S} g}{\min(\mathcal{M}_R, R'/\mathcal{B}_R)}\right]^{B/\mathcal{B}_R} ;$$

$$\dot{R}^{S} = fR^{S} ;$$

$$\dot{S}_t^{R} = \gamma_R f S_t^R ;$$

$$\dot{R}^{NS} = R^{NS}\left(1-\left[1 - \frac{B\bar{k}N_g}{\min(M_R, R'/\mathcal{B}_R)}\right]^{B/\mathcal{B}_R}\right).$$

(d) **Blue Attack Mode 4.** As in Step (14d), first average over shooter type:

$$^{R}a_s = (B_1 \times {}^{R}a_s^1 + B_2 \times {}^{R}a_s^2)/B \ ;$$

$$^{R}a_n = (B_1 \times {}^{R}a_n^1 + B_2 \times {}^{R}a_n^2)/B \ ;$$

$$^{R}k_{ns} = (B_1 \times {}^{R}k_{ns}^1 + B_2 \times {}^{R}k_{ns}^2)/B \ ;$$

$$^{R}k_{sn} = (B_1 \times {}^{R}k_{sn}^1 + B_2 \times {}^{R}k_{sn}^2)/B \ .$$

Let q be the proportion of Blue attack passes that load with munitions against shelters; and let $B^S = \frac{qB_1 + qB_2}{\mathcal{B}_R} = \frac{qB}{\mathcal{B}_R}$, and $B^{NS} = \frac{(1-q)B}{\mathcal{B}_R}$. Define quantities

$$^{R}x_n = \begin{cases} 0.0, & \text{if } (1-\omega_R) \times {}^{R}a_n/b_R < 0 \ ; \\ 1.0, & \text{if } (1-\omega_R) \times {}^{R}a_n/b_R > 1 \ ; \\ (1-\omega_R) \times {}^{R}a_n/b_R, & \text{otherwise;} \end{cases}$$

$$^{R}x_{sn} = \begin{cases} 0.0, & \text{if } (1-\omega_R) \times {}^{R}a_s \times {}^{R}k_{sn}/b_R < 0 \ ; \\ 1.0, & \text{if } (1-\omega_R) \times {}^{R}a_s \times {}^{R}k_{sn}/b_R > 1 \ ; \\ (1-\omega_R) \times {}^{R}a_s \times {}^{R}k_{sn}/b_R, & \text{otherwise;} \end{cases}$$

$$
{}^{R}x_{s} = \begin{cases} 0.0, & \text{if } (1-\omega_R) \times {}^{R}a_s/b_R < 0 \text{ ;} \\ 1.0, & \text{if } (1-\omega_R) \times {}^{R}a_s/b_R > 1 \text{ ;} \\ (1-\omega_R) \times {}^{R}a_s/b_R, & \text{otherwise;} \end{cases}
$$

$$
{}^{R}x_{ns} = \begin{cases} 0.0, & \text{if } (1-\omega_R) \times {}^{R}a_n \times {}^{R}k_{ns}/b_R < 0 \text{ ;} \\ 1.0, & \text{if } (1-\omega_R) \times {}^{R}a_n \times {}^{R}k_{ns}/b_R > 1 \text{ ;} \\ (1-\omega_R) \times {}^{R}a_n \times {}^{R}k_{ns}/b_R, & \text{otherwise;} \end{cases}
$$

$$
\text{Let } f = \min\left\{1.0,\ \frac{\omega_R(B^{NS} x\, {}^{R}a_n x\, {}^{R}k_{ns} + B^{S} x\, {}^{R}a_s)}{b_R} + (1-[1 - {}^{R}x_{ns}]^{B^{NS}}[1 - {}^{R}x_s]^{B^{S}})\right\} .
$$

Then $\dot{R}^{S} = fR^{S}$; $\dot{S}_{t}^{R} = \gamma_R f S_{t}^{R}$; and

$$
\dot{R}^{NS} = R^{NS} \times \min\left\{1.0,\ \frac{\omega_R(B^{NS} x\, {}^{R}a_n + B^{S} x\, {}^{R}a_s x\, {}^{R}k_{sn})}{b_R} + (1-[1 - {}^{R}x_n]^{B^{NS}}[1 - {}^{R}x_{sn}]^{B^{S}})\right\} .
$$

The appropriate q is found by using Newton's method.

(17) **Total aircraft destroyed for the day.**

The quantities for sheltered and nonsheltered aircraft killed on the airbase are apportioned by kind of aircraft. Then aircraft killed in the air-to-air and ground-to-air interactions are added.

(a) Blue aircraft destroyed:

$$x^d_{mt} = \dot{B}^S \times B^S_m/B^S + \dot{B}^{NS} \times B^{NS}_m/B^{NS} \qquad m=1 \text{ to } 4 \;;$$

$$x^d_{1t} = x^{d'}_{1t} + \sum_{j=1}^{3} ({}^{AA}\dot{x}_{1j} + {}^{GA}\dot{x}_{1j}) \qquad (GP) \;;$$

$$x^d_{mt} = x^{d'}_{mt} + {}^{AA}\dot{x}_{2,m-1} + {}^{GA}\dot{x}_{2,m-1} \qquad m=2 \text{ to } 4 \text{ (SP-CAS, SP-ABA, SP-INT)} \,.$$

(b) Red aircraft destroyed:

$$y^{d'}_{nt} = \dot{R}^S \times R^S_n/R^S + \dot{R}^{NS} \times R^{NS}_n/R^{NS} \qquad n=1 \text{ to } 4 \;;$$

$$y^d_{1t} = y^{d'}_{1t} + \sum_{j=1}^{3} ({}^{AA}\dot{y}_{1j} + {}^{GA}\dot{y}_{1j}) \qquad (GP) \;;$$

$$y^d_{nt} = y^{d'}_{nt} + {}^{AA}\dot{y}_{2,n-1} + {}^{GA}\dot{y}_{2,n-1} \qquad n=2 \text{ to } 4 \text{ (SP-CAS, SP-ABA, SP-INT)} \,.$$

(18) Compute air firepower for day t as the sum of successful CAS sorties times firepower score per sortie:

$$A^B_t = u_{11}\,{}^BC_1 + u_{21}\,{}^BC_2 \;;$$

$$A^R_t = v_{11}\,{}^RC_1 + v_{21}\,{}^RC_2 \,.$$

(19) Compute total firepower for subperiod t as the sum of ground-plus-air firepower:

$$T^B_t = G^B_t + A^B_t \;;$$

$$T^R_t = G^R_t + A^R_t \,.$$

(20a) FEBA advance for day:

$$\Delta\ \text{FEBA} = \begin{cases} F\left(\dfrac{T^B_t}{T^R_t}\right), & \text{if } T^B_t \geq T^R_t \;; \\[2ex] -F\left(\dfrac{T^R_t}{T^B_t}\right), & \text{if } T^B_t < T^R_t \,. \end{cases}$$

(20b) FEBA position at end of day:

$$FEBA_t = FEBA_{t-1} + \Delta\ FEBA \ .$$

(21) **Division destruction.**

If all Blue casualties are replaced,

$$^{B}D^{d}_{\ell_B t} = 0.0 \qquad \text{all } \ell_B \ .$$

If no Blue casualties are replaced,

$$^{B}D^{d}_{\ell_B t} = {}^{B}D_{\ell_B t} \times g^B\left(\frac{T^B_t}{T^R_t}\right) \qquad \text{all } \ell_B \ .$$

If all Red casualties are replaced,

$$^{R}D^{d}_{\ell_R t} = 0.0 \qquad \text{all } \ell_R \ .$$

If no Red casualties are replaced

$$^{R}D^{d}_{\ell_R t} = {}^{R}D_{\ell_R t} \times g^R\left(\frac{T^B_t}{T^R_t}\right) \qquad \text{all } \ell_R \ .$$

(22) Compute cumulative total firepower by day t:

$$CT^B_t = CT^B_{t-1} + T^B_t \ ;$$

$$CT^R_t = CT^R_{t-1} + T^B_t \ .$$

(23) Compute cumulative air firepower by day t:

$$CA^B_t = CA^B_{t-1} + A^B_t \ ;$$

$$CT^R_t = CT^R_{t-1} + T^B_t \ .$$

(24) The additional MOEs are not computed unless specifically called for--and then only for *one* input day t (usually the last day of the war).

(a) MOE4--surviving aircraft, weighted by type:

$$MOE = {}^{B}W_1^Q(x_{1t}-x_{1t}^d) + \sum_{m=2}^{4} {}^{B}W_{m-1}^S(x_{mt}-x_{mt}^d)$$

$$- {}^{R}W_1^Q(y_{1t}-y_{1t}^d) - \sum_{n=2}^{4} {}^{R}W_{n-1}^S(y_{nt}-y_{nt}^d) \ .$$

(b) MOE5--comprehensive air measure--is explained in Chapter III, Section F.2 (above).

$$MOE = {}^{B}W^C CA_t^B + {}^{B}W_1^Q(x_{1t}-x_{1t}^d-Q_B')^+$$

$$+ \sum_{m=2}^{4} {}^{B}W_{m-1}^S(x_{mt}-x_{mt}^d) + {}^{B}W_2^Q(x_{1t}-x_{1t}^d-Q_B')^-$$

$$- {}^{R}W^C CA_t^R - {}^{R}W_1^Q(y_{1t}-y_{1t}^d-Q_R')^+$$

$$- \sum_{n=2}^{4} {}^{R}W_{n-1}^S(y_{nt}-y_{nt}^d) - {}^{R}W_2^Q(y_{1t}-y_{1t}^d-Q_R')^- \ .$$

The symbol $(z)^+$ means the positive part of expression z--i.e.,

$$(z)^+ = \begin{cases} z, & \text{if } z \geq 0 \ ; \\ 0, & \text{if } z < 0 \ . \end{cases}$$

Similarly,

$$(z)^- = \begin{cases} 0, & \text{if } z > 0 \ ; \\ z, & \text{if } z \leq 0 \ . \end{cases}$$