AD-A094 814   AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH   SCHOO--ETC   F/G 9/2
A METHODOLOGY FOR VALIDATION OF COMPLEX MULTI-VARIABLE MILITARY--ETC(U)
DEC 80   C S GHELBER, C A HALEY

UNCLASSIFIED   AFIT/GOR/MA/80D-3   NL

1 OF 2

AD A
094814

UNITED STATES AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY
Wright-Patterson Air Force Base,Ohio

AFIT/GOR/MA/80D-3

1

A METHODOLOGY FOR VALIDATION OF COMPLEX MULTI-VARIABLE MILITARY COMPUTERIZED MODELS.

THESIS

Craig S. Ghelber   Charles A. Haley
Captain   USAF   Captain   USAF

AFIT/GOR/MA/80D-3

Approved for public release; distribution unlimited

AFIT/GOR/MA/80D-3

A METHODOLOGY FOR VALIDATION OF COMPLEX MULTI-VARIABLE MILITARY COMPUTERIZED MODELS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Craig S. Ghelber
Captain USAF
Graduate Strategic
and Tactical Sciences

Charles A. Haley
Captain USAF
Graduate Operations
Research

December 1980

Approved for public release; distribution unlimited

Preface

The objectives of this research effort were to develop a methodology for the validation of complex simulation models and to apply it to an existing Department of Defense model. Few existing large scale models have been subjected to rigorous validation procedures, yet decisions involving multi-million dollar expenditures are often based on these models. Even though the requirement for validation is apparent, existing procedures provide little practical guidance.

We wish to thank Major Kenneth Melendez, our advisor, for his insight and patient support in this thesis effort. Also our appreciation is extended to our readers, Lt Col Charles W. McNichols and Captain Brian Woodruff, for their assistance on the subjects of experimental design and statistical testing. We further gratefully acknowledge the assistance provided by Captains John Fox and Greg Smith of Air Force Studies and Analysis. Their knowledge of the Interceptor War Game Model contributed significantly to the implementation and understanding of the model.

We sincerely hope that this research effort serves as a catalyst for a renewed emphasis on the importance of model validation.

Craig S. Ghelber
Charles A. Haley

Contents

Contents

List of Figures

List of Tables

Abstract

Computerized simulation models that are characterized by multi-variables and minimal or nonexistent real world supporting data are often used without being properly validated. The towards-validation methodology is introduced as a four-phase approach for validating these complex models and is defined as: The documented evidence that a computerized model can provide users verifiable insight, within the model's domain of application, for the purpose of formulating analytical or decision-making inferences.

Towards-validation begins with the conceptual phase of model development. Next, the verification phase examines the mechanical validity of model design. The third phase, credibility, is concerned with both intuitive and statistical appeal. The final phase deals with confidence building and documentation.

To illustrate the application of towards-validation, the Headquarters Air Force version of the Interceptor War Game Model is examined. Results are documented in a user's guide.

A METHODOLOGY FOR VALIDATION OF COMPLEX
MULTI-VARIABLE MILITARY COMPUTERIZED MODELS

I. INTRODUCTION

Military organizations, due to the magnitude of national defense and the nature of economic restraint, have evolved as leaders in the development and use of complex modeling techniques. The "Catalog of War Gaming and Military Simulation Models" lists more than 140 models that are in general use throughout the Department of Defense (DOD). Proponents include: Assistant Secretary of Defense, Office of Program Analysis and Evaluation; Organization of the Joint Chiefs of Staff, Logistics Directorate and the Studies, Analysis, and Gaming Agency; Defense Intelligence Agency; and Defense Nuclear Agency, to mention just a few (Ref 1).

Simulation results are often instrumental in influencing decisions on multi-million dollar expenditures as well as national defense policies. With this level of usage, it is obvious that computerized models have become an important tool for analysts and decision-makers in DOD. Therefore, it is imperative that users understand the proper application and fully comprehend the need for validation of military models.

Purpose of Simulation

Computer modeling can be defined as a means whereby a system in the real world can be conceptually represented by a simulation computer language for the purpose of providing valuable information to users. Users are generally analysts concerned with inferences based on input/output values, or decision-makers who dictate policy based on simulation results. Furthermore, simulation is the process of exercising the model for the purpose of:

1. Evaluation: determining how good a system performs in an absolute sense.

2. Comparison: comparing competitive or proposed policies or procedures.

3. Prediction: estimating the performance of a system.

4. Sensitivity analysis: determining the variables that most significantly affect the system.

5. Optimization: determining the level of variables that produces the best system outcome (Ref 11:59).

Overview

Often, those involved in the intricacies of model development and use overlook the design purpose and limitations of the model. Due to the high uncertainty of reality, care must be taken by users to avoid interpreting results as a prediction of what will actually occur in a real world situation. To enhance the capability of the user to draw inferences or make decisions, an acceptable level of confidence in simulation results must be insured by means of a validation process. However, an indepth review of existing literature indicates that theoretical approaches to model validation either rely on an extensive data base from actual field testing, or are limited in application to models with a small number of variables.

The purpose of this research effort is to present a methodology for the validation of complex, multi-variable, computerized models. Chapter II outlines the terminology and the present literature on the subject. A framework for a practical approach to validation is presented in Chapter III. The remaining chapters are dedicated to the application of this methodology to an existing DOD model.

II. REVIEW OF EXISTING LITERATURE

Terminology

In order to evaluate validation procedures presented by other authors, it is important to standardize terminology. One such set of definitions was presented in 1974 by the Society for Computer Simulation (SCS) through its Technical Committee on Model Credibility (Ref 10):

Reality: An entity, situation, or system which has been selected for analysis.

Conceptual Model: Verbal description, equations, governing relationships, or "natural laws" that purport to describe reality.

Computerized Model: An operational computer program which implements a conceptual model.

Model Verification: Substantiation that a computerized model represents a conceptual model within specific limits of accuracy.

Model Validation: Substantiation that a computerized model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model.

Computer Simulation: Exercise of a tested and [validated] computerized model to gain insight about reality.

Although this list of definitions is not all-inclusive, it does provide a common basis for communication. The following sections of this chapter outline several noteworthy contributions made to the subject of model validation.

Hermann (Ref 4:220-231)

In early 1967 Charles F. Hermann developed one of the first structured approaches to model validation. He proposed the following five criteria:

Internal Validity. Internal validity is concerned with the variance between replications of a simulation. Identical initial parameters are used in consecutive simulation runs. If a variance is noted that can be attributed to extraneous factors, then internal validity is low.

Face Validity. This test is accomplished by having someone familiar with the real world system make a purely subjective analysis of simulation results.

Variable-Parameter Validity. There are two primary features of Hermann's variable-parameter criterion: comparisions between the simulation's parameters with the corresponding values in the real world; and a sensitivity analysis of input parameters.

Event Validity. Event validity addresses the issue of isomorphism. This test attempts to measure how accurately the elements of the model must simulate the detailed aspects of the real world system.

Hypothesis Validity. This final criterion is concerned with the validity of hypothesized or empirically derived relationships used in the model.

Hermann's approach, although developed in the context of international political models, does contain two points of particular interest. First, his face validation (which is a part of most validation schemes) is an important initial step. Irregularities and inconsistencies can be identified in the early stages of model validation, saving the time and the expense of lengthy statistical testing. Face validation is often the only form of validation used on complex military models. Next, he states in his report: "validity is always a matter of degree," and his approach "helps build confidence in the validity of the model." These statements are fundamental to the approach detailed in Chapter III.

Unfortunately, Hermann fails to address specific procedures for implementation of his approach, particularly with respect to complex multi-variable models. For example, the variable-parameter criterion is not applicable if real world data is nonexistent. This is the case in war-gaming or system proposal models. Likewise, sensitivity analysis may not be practical for models that have large numbers of input variables or that require several hours of computer time for each simulation run. (These problems will be addressed in Chapter III.)

Naylor and Finger (Ref 8)

In the existing literature the verification/validation process most often quoted was presented in late 1967 by Thomas H. Naylor and J. M. Finger. Their three-stage approach incorporates the philosophies of rationalism, empiricism, and positive economics. [See Naylor and Finger (Ref 8) or Shannon (Ref 11:212-215) for a discussion of these terms].

Their first stage is to identify the processes that form the foundation on which the entire model is structured. These processes are then examined, based on prior knowledge and face validation, to formulate a set of postulates on which the model is built. Stage two is concerned with the formal testing of the postulates identified in the first stage. Statistical estimation and hypotheses testing are the primary tools for this step. The third stage attempts to test the model's ability to predict the real system. Ideally this is accomplished by statistically testing the simulated output with real world data.

Naylor and Finger make the point that for simple models, the first two stages can be skipped with a minimum of risk. However, for complex models with a large number of variables (some stochastic), too much is lost by not accomplishing all stages.

Again there are severe limitations in the use of this procedure. The first two stages can be interpreted as a verification process of the conceptual model. However, without real world data to accomplish the third stage, a critical shortcoming exists. Nevertheless, the strength of their process lies in the examination of the foundation elements of the model and the building of confidence as the model is developed.

It is obvious from the methodologies presented by Hermann, and Naylor and Finger that 1967 was a very productive year for the advancement of validation theory. This was due partially to the development of a new generation of large capacity, high speed computers, and attempts by industry and the military to model complex systems. Hence, the necessity arose for a more standard and useful set of procedures for validation. A step in this direction was taken in November 1967 by Fishman and Kiviat when they segmented verification and validation into separate disciplines aimed at building confidence and credibility, respectively, in a model's response (Ref 3:v).

Schlesinger (Ref 9:927-933)

In 1974 a noteworthy procedure was advanced that specified a standard for model verification and validation. S. Schlesinger, et. al., determined there was a need for procedures and standards that would provide a credible assessment of a model's ability to generate appropriate as well as reasonable data.

The first of their four steps requires that the computerized model be analyzed to insure it accurately represents the conceptual model. Checking numerical techniques, logical flow, and general completeness of the model is emphasized.

Next, the reasonableness of the model, which is analogous to face validation, is examined. Reasonableness is characterized by continuity, consistency, and degeneracy. Continuity insures that appropriate changes in input values do not cause extraordinary changes in output. Consistency requires that similar input data generate similar output results. Degeneracy examines the extreme values of parameters to insure that model logic remains intact.

Their third step is a validation process. Similar to procedures previously discussed, quantitative measures are used to determine deviations between simulated and actual data.

Finally, Schlesinger stresses that a model should not be used outside its "domain of applicability." Furthermore, once a model is certified [verified/validated] any new assumptions or changes must be recertified and the domain of applicability redefined. In a concluding comment he emphasizes that experimentation should only be conducted on certified/validated models.

Tytula (Ref 12)

Thomas P. Tytula in 1978, while attempting a validation of a missile system simulation model, organized the work of other authors into four general categories: judgmental comparisons, hypothesis testing, sensitivity analysis, and indices of performance.

Judgmental Comparisons. Tytula describes judgmental comparisons as the process of visually examining the model for logical flow. This includes a graphical analysis of common properties of the real system and the model, and a face validation by people familiar with the actual process. The inability to quantify this judgment, however, is a significant drawback.

Hypothesis Testing. He next points out that in an attempt to quantify the validation process many authors employ statistical hypothesis testing. There are two drawbacks, however. First, the strength of hypothesis testing is in rejecting that which was set out to be proven. Unfortunately, it is usually desired to show acceptance of a hypothesis. The second drawback concerns the misuse of statistics due to underlying assumptions like independence and normality.

Sensitivity Analysis. The intent of performing a sensitivity analysis is to determine the range of model parameters for which output remains valid. Furthermore, confidence can be enhanced if it can be reasonably assumed that actual parameter values will not be outside the range tested. However, due to the problems associated with the time and cost of gathering this data, sensitivity analysis is infrequently used.

Performance Indices. Several authors, including Naylor and Finger, have proposed performance indices that profess to quantify the agreement between simulated and real world data. Generally, these indices are based on the square of the difference between expected and observed data. The obvious problem with this technique is determining at what level validity is proved or disproved. Its strength is in ranking alternatives.

Tytula concludes from his research that all validation methodologies have certain pitfalls:

> ... The most important of these shortcomings are the inability to handle the autocorrelation of the simulation output variables, concentration on the wrong issue, and difficulty in transforming the measure of disagreement between simulated and actual results into some meaningful set of consequences.

To emphasize Tytula's skepticism, Richard Van Horn points out, "This method of testing suffers from the standard problems of empirical research: (1) too small samples due to the high cost of data, (2) too aggregate data, and (3) data whose own validity is questionable" (Ref 13:257).

Comment

The purpose of presenting the above methodologies is to outline the chronological growth of the theory of validation since 1967. The authors quoted by no means represent all existing work on the subject. Appendix B lists some additional reference material.

Earlier approaches to validation professed theoretical procedures for assuring agreement between simulation results and the real world system. As the complexity of simulation models increased, predictably, the complexity of validation increased. The definition of validation was then redefined as: "The process of building an acceptable level of confidence that an inference about a simulation is a valid inference for the actual process" (Ref 13:233). All too often, however, this new complex problem has been handled by not validating.

Chapter III will present a framework for a validation procedure which addresses a class of models common to military applications. However, departing slightly from classical approaches, this procedure incorporates the philosophy that: "Nothing will ever be attempted if all possible objections must be first overcome" (Samuel Johnson).

III. METHODOLOGY

Contrary to popular belief, most models do not have absolute replication of the real world system as a purpose. Many models provide users alternatives in the decision-making process. As Van Horn states: "The validity requirement is that the simulation aid its users in such ways as to detect useful alternative means of handling a problem" (Ref 13:249).

Experimentation with a validated model should ensure that a decision-maker can make well informed decisions without costly field testing of the actual system. A simulation model of a complex process, however, is only an estimation of the real world system. Thus, absolute validity should be measured only by the degree to which the model performs an intended purpose.

Towards-Validation

There is no such thing as *the* appropriate validation procedure; "validation is problem-dependent" (Ref 13:257). A "checklist" approach useful for one model may not be applicable to others. However, if a procedure can be tailored to a sufficiently restricted class of models, one methodology, with only minor problem-dependent changes, might apply. To this end "towards-validation" is presented as a new concept defined as: The documented evidence that a computerized model can provide users verifiable insight, within the model's domain of application, for the purpose of formulating analytical or decision-making inferences.

The process of towards-validation is achieved by a four-phase approach:

1. Conceptual
2. Verification
3.

Credibility

4. Confidence

Figure 1 pictorially demonstrates these concepts as they apply to the notion, development, and application stages of a model. A basic premise is that towards-validation is a wholistic approach to computerized modeling. It begins with problem definition and continues through implementation.

Many military oriented models have common characteristics that lend themselves to the use of towards-validation:

1. The requirement to compare alternative information for policy decisions.
2. Limited or nonexistent supporting data from the real world system.
3. Physical processes that require numerous variables to adequately describe their complexity.
4. Separable subsystems whereby variables can be partitioned into convenient groups.

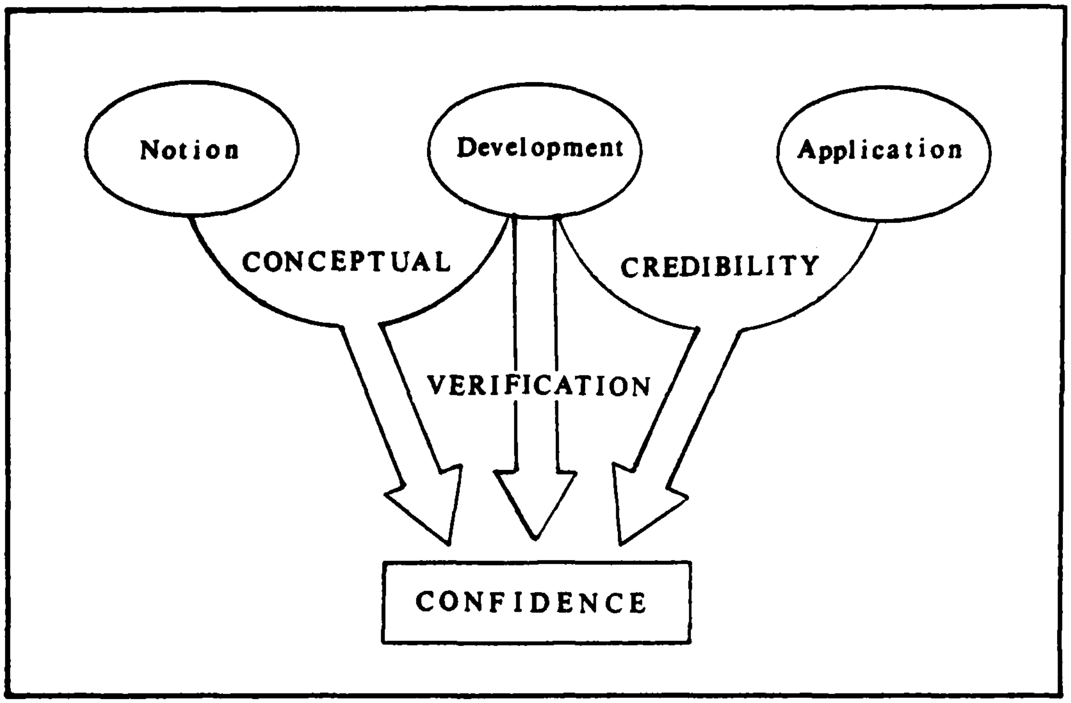


Figure 1. Towards-Validation

Conceptual

The conceptual phase of towards-validation deals with the early stages of model development and contains the following basic elements:

1. A formal written statement of the intended application of the model.

2. Specification of the degree of accuracy desired.

3. Description of assumptions and limitations.

4. Structural model or framework for design development.

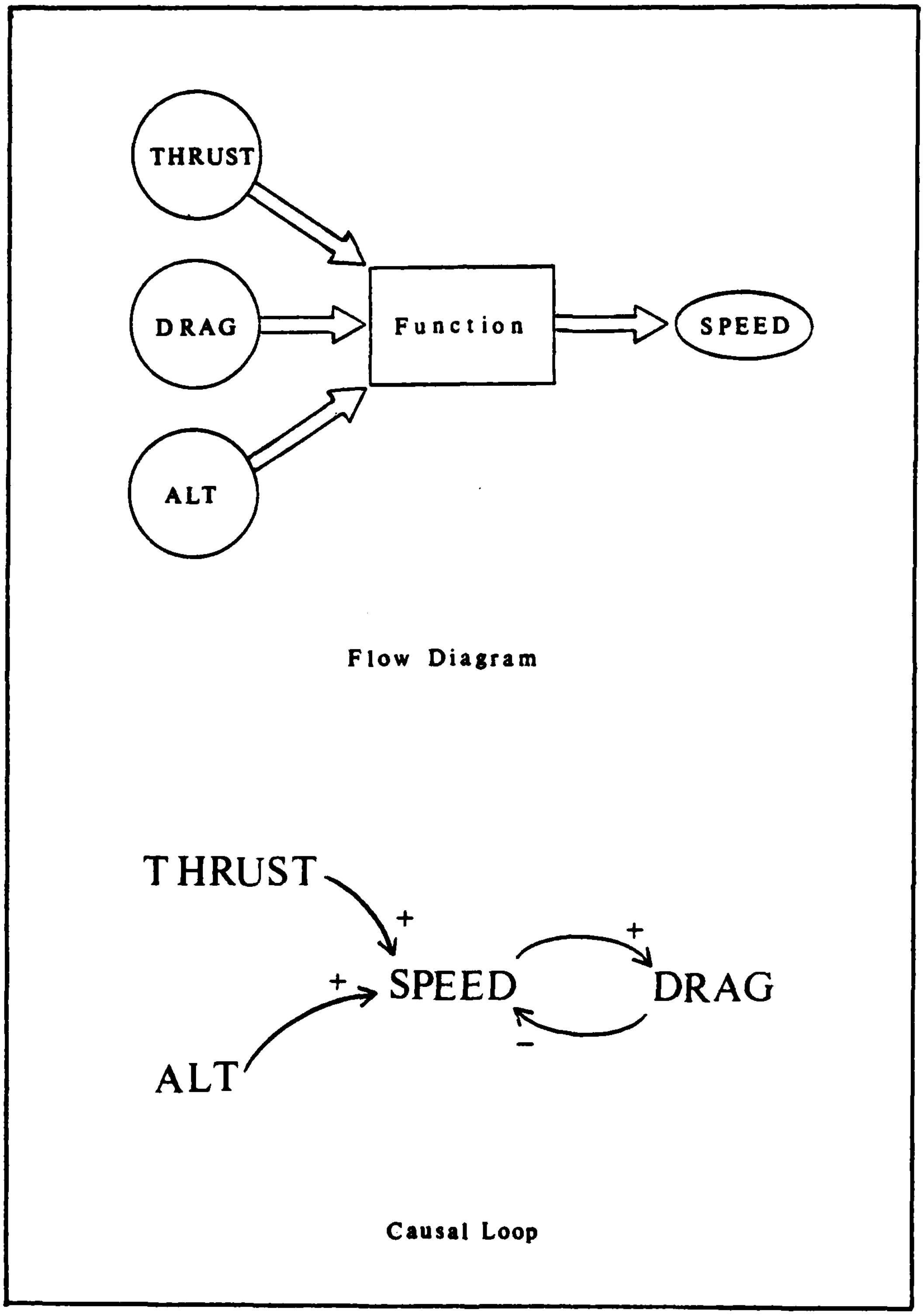
A formal written statement will proviJe guidance to the model designer and should define the domain of application as described by those intending to use the model. As a minimum it should include: (1) a well-defined statement of the intended application, (2) the level of usage, and (3) any specific guidance provided by prospective users. If the task is to validate an existing model, this process is only slightly modified, and should include a list of present and past projects, plus the level of reliance decision-makers place on simulation results.

In order to place the proper emphasis on the labors of model development and to relate cost to time and effort, the desired degree of accuracy must be specified. Guidance can come directly from decision-makers or can be implied from the intended application. If comparison of alternatives is the goal of the model, rather than replication of physical processes, then less accuracy may be required. The range of accuracy is generally specified in terms of statistical confidence or decision criteria. Often, for complex military models, supporting data sample sizes are too small for significant statistical testing. Decision criteria or analytical insight must then be relied on.

Defining assumptions and limitations may be the most important part of the conceptual phase of towards-validation. Computerized models cannot simulate all phases of even limited real world systems. Often, models are limited in scope by scenario assumptions. For example, it might be assumed in a war-gaming model that only the air-to-air portion of the war is being studied and ground threats do not affect the battle. Assumptions concerning human interaction, and command and control can further limit the domain of applicability of a model. Throughout the validation process, limitations and assumptions will surface that require a reevaluation of the intended application of the model.

Finally, for the structural model step the model designer must identify the dependent or output variable(s). These are the variables that provide the user with decision-making alternatives. The importance of this step lies in the need for user understanding of the flow and basic interaction of model variables. A structural model can then be designed for visual reenforcement of the conceptual model. For example, a fighter aircraft might be modeled to analyze aircraft performance. The dependent variable, maximum speed, is a function of decision variables: thrust, drag, and altitude. Figure 2 demonstrates two structural model techniques commonly used. Simple diagrams provide users an intuitive feel for the intricacies of the model without overwhelming them with computer code or physical laws.

It is important at this point, to emphasize that the four phases of towards-validation are iterative (Fig. 3). Following the completion of each phase, the previous steps



Flow Diagram

Causal Loop

Figure 2. Structural Models

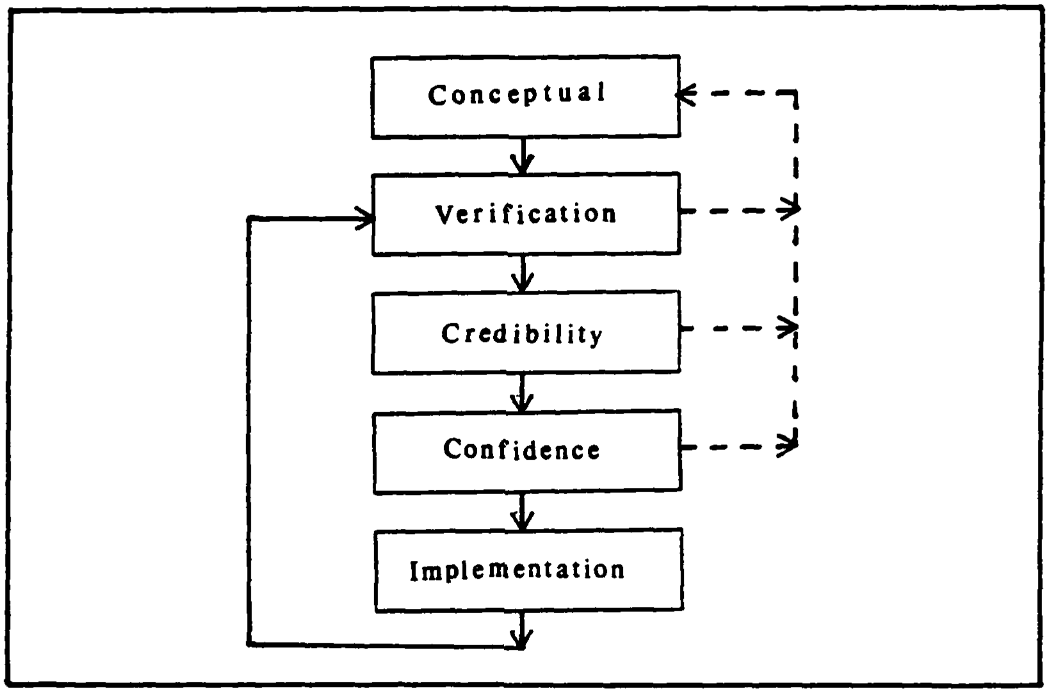


Figure 3. Iterative Process

should be reexamined for completeness and consistency.

Verification

Verification in this context is similar to classical approaches. It is concerned with the mechanical validity of model design. Four steps are suggested:

1. Structured walk-through;
2. Verification of technical physical processes;
3. Simulation of predictable states;
4. Testing of stochastic events.

Classical approaches to a structured walk-through involve hand calculating and manually tracking data through the model. This process can build confidence in the mechanical structure. There are three additional benefits to be gained from a structured walk-through: (1) verification of event-path integrity, (2) model familiarization, and (3) identification of physical processes and stochastic events for steps two and four above.

Unfortunately, for complex models, this procedure is time and manpower prohibitive. Therefore, a towards-validation structured walk-through is aimed primarily at the three additional benefits only. Most computerized models are built around a source program that controls flow to and from subroutines. By methodically insuring that subroutines are properly accessed and that expected parameter values will in fact direct calculations appropriately, event-path integrity can be checked. For example, missile launch range for a fighter aircraft might be determined by a subroutine accessed during an air-to-air engagement. Improper values passed to the subroutine will adversely bias results. An extremely important added benefit of a walk-through is the familiarization gained, particularly, for existing models.

Verification of physical processes is accomplished by insuring that the proper equations and relationships are used in developing the model. For example, it might be found that in calculating the collision angle in an aircraft intercept problem, the cosine of an angle is used instead of the sine. Other common errors to look for include mismatched units and unfounded empirical equations.

The next step of the verification phase is simulation of predictable processes. Total predictability may be hard to insure if there are a large number of stochastic events; however, by setting variances equal to zero and probabilities to either zero or one, partial predictability can be assured. The key to this procedure is the careful selection of input data so as to limit the simulation's scope to the process desired. For example, in the model that simulates an air battle, by structuring input data, a fighter aircraft can be placed in an ideal firing position to shoot another aircraft. A predictable outcome should ensue. Other processes can similarly be studied until confidence in the major functions of the model is achieved. Further insight can be gained by inserting print statements into appropriate sections of the computer code in order to track structural flow.

The final step is to test stochastic events. This is accomplished by comparing simulation generated variates with the expected distributions. The most appropriate statistical tests are chi-square goodness of fit test, if 30 or more data points are available, and the Kolmogorov-Smirnov test for fewer data points (Ref 6).

Additional confidence in the mechanical validity of the model can be gained by varying the random number seed for successive simulation runs. If the input parameters are held constant, a small variance in the output would be expected. Any large deviations should be investigated to insure that stochastic events did not cause infeasible results.

Credibility

The third phase, credibility, deals with both the intuitive and the statistical appeal of the model based on:

1. Face validation;
2. Sensitivity analysis.

Face validation is accomplished by having someone familiar with the real world system make a purely subjective analysis of the results. Two approaches are suggested. First, the expert can create the scenario; then compare simulation results with his expected results. The second approach is to inform the expert of the scenario and input data, and to elicit his predicted outcome.

To analyze the results of a face validation, it is important to remember that only the feasibility of the simulation is being tested. After several exchanges between model and expert, conclusions about face validity can be drawn. If the expert is not in agreement with the simulation results, the criteria established in the conceptual phase will have to be reviewed before implying negative confidence in the model.

The next step in the credibility phase is sensitivity analysis. To maximize the amount of information gained from this analysis, an experimental design should be developed to systematically specify input values for simulation runs. The effect an input variable has on the output of a simulation can be represented graphically (Fig. 4) by plotting the dependent variable as it varies through its range of values. This is called the response surface--the output value is called the response. The purpose of this analysis is to explore the response surface to insure that the range of input values does not produce discontinuities.

To develop this technique, first consider a model with two input variables. One method of experimental design, the one-factor-at-a-time approach, would suggest four simulation runs, one at the high and low values of each variable, varying only one factor at a time. In addition, statistical testing requires multiple data points; therefore, the entire experiment must be repeated an appropriate number of times. There are two obvious drawbacks to this procedure. First, the cost associated with running multi-variable models makes it cost-prohibitive for the number of runs required. Second, this experiment does not account for effects caused by varying more than one variable at a time.

Hunter and Naylor suggest two experiments that address these limitations: the full factorial and fractional factorial designs (Ref 5:43). The full factorial design solves the problem of multiple interactions by testing all combinations. The number of runs required can be calculated by evaluating r^k, where k is the number of variables and r

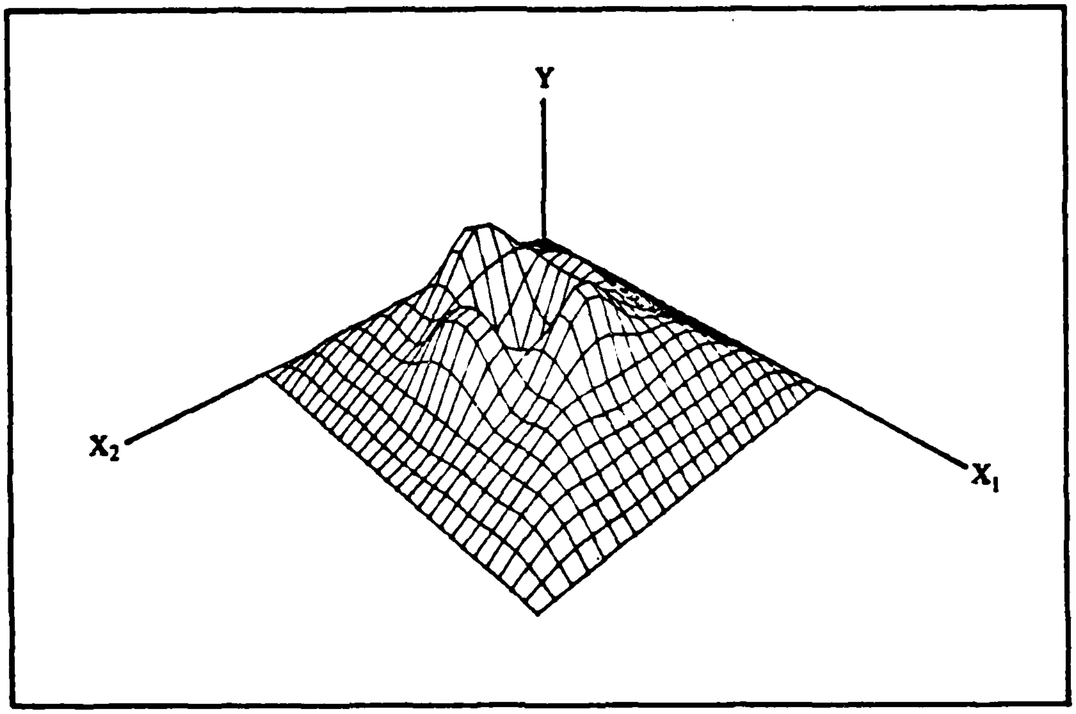


Figure 4. Response Surface

is the number of levels at which each variable is tested. Although the interaction problem would be solved with this design, the number of runs required can still be excessive. For example, a 2^{10} experiment would require 1024 runs--cost-prohibitive for most models.

With the fractional factorial design, the number of runs can be reduced significantly by eliminating undesired interactions. It might be determined, in a four variable experiment, for example, that only two-factor interactions need be considered. A fractional factorial design (Fig. 5) suggest 8 simulation runs, whereas a full factorial needs 16. The drawback is that accuracy is lost when interactions

A 2^{4-1} FRACTIONAL FACTORIAL DESIGN

	Design Matrix				
Run	A	B	C	D	Interaction Tested
1	−	−	−	−	ABCD
2	+	−	−	+	BC
3	−	+	−	+	AC
4	+	+	−	−	CD
5	−	−	+	+	AB
6	+	−	+	−	BD
7	−	+	+	−	AD
8	+	+	+	+	I

A,B,C,D. . . Decision Variables

+. . . High Value

−. . . Low Value

Figure 5. Experimental Design

are eliminated. Therefore, cost, time and accuracy considerations need to be carefully weighed in order to optimize information gained from an experimental design. For a technical discussion of the fractional factorial design see Hunter and Naylor (Ref 5:43-54).

So far only two levels of response have been addressed for input variables. Performing a sensitivity analysis by examining only the high and low values of a variable is adequate if the output response remains linear as the parameter is varied (Fig. 6a). Fortunately, as long as the curve can be approximated by a linear curve, little accuracy is lost (Fig. 6b).

For models with a large number of variables, the above techniques may not sufficiently reduce the required runs. Therefore, screening techniques must be introduced to minimize the number of variables considered.

Screening is the key to a successful experimental design for towards-validation. The first step is to confer with individuals familiar with the real world system. They can identify the input variables that have, according to their experience, the least effect on the overall system. One of two techniques can then be used. First, the remaining variables can be partitioned into groups where variables of each group are independent of the other groups. Military models that are built around weapon subsystems generally lend themselves to this approach. The number of simulation runs is significantly reduced by then developing an experimental design for each group individually. For example, a full factorial design for a 20 variable model would take 2^{20} or 1,048,576 runs. However, if this problem could be divided into five separate 2^{4} experiments, only 80 runs are needed. If this number of runs is still too large, additional variables can be screened until budget or time

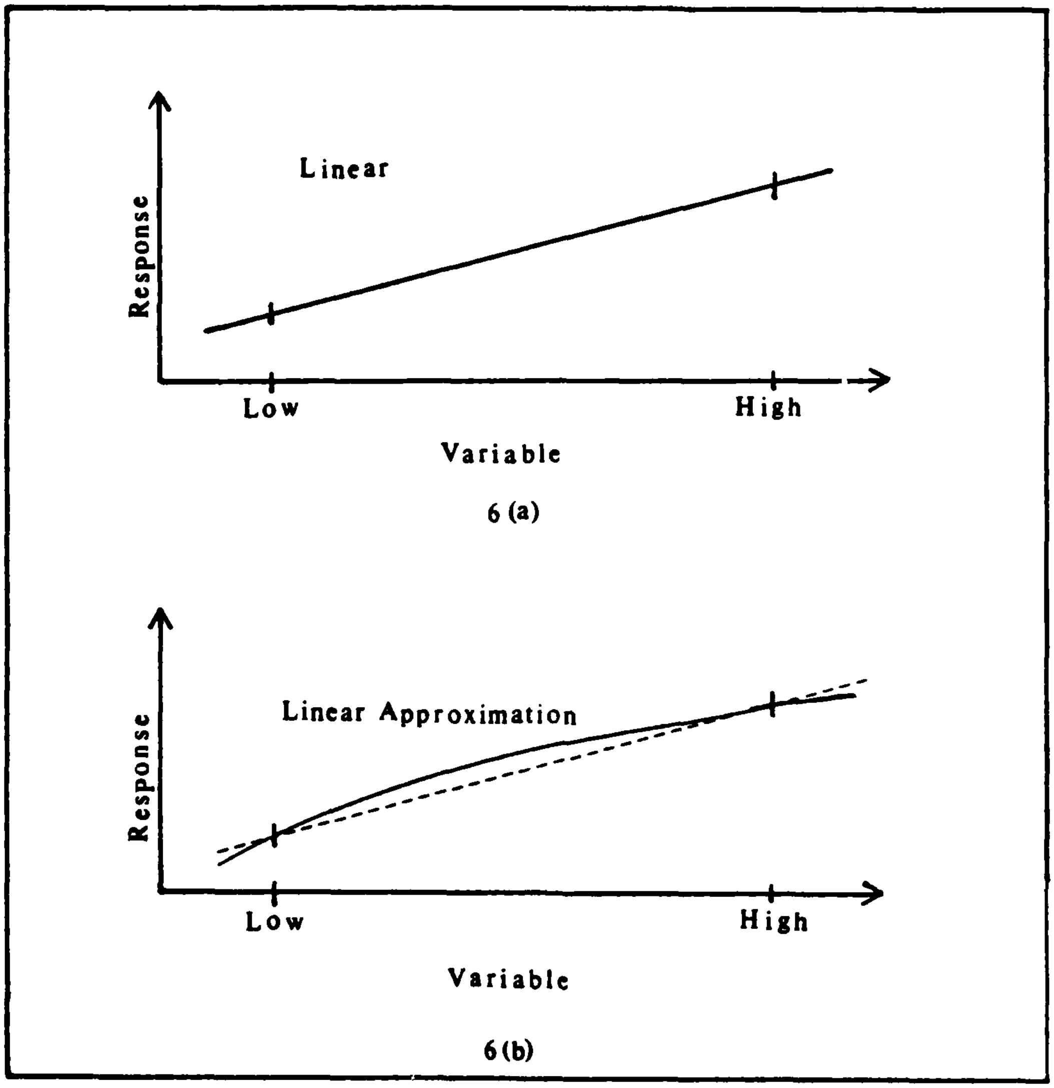


Figure 6. Linearity Assumption

restraints are satisfied. After variables are eliminated from consideration, use their expected values for all sensitivity analysis runs.

The second screening technique, which is not as easily applied, is to use variables for the experimental design that are a function of other variables. For example, since maximum aircraft speed is a function of thrust, drag and altitude, the response of these three factors can be accounted for by the variability of maximum speed alone.

The response surface generated by the simulation runs of an experimental design can be analyzed by use of statistical analysis of variance (ANOVA), or regression analysis. Most good statistical textbooks describe procedures for these statistical test.

As is often the case, however, "you do not get something for nothing." With the use of these screening techniques, not only is accuracy lost, but extreme care must be taken when comparing significance levels across group boundaries. Therefore, to justify the use of these techniques, the benefits gained must be examined.

Due to the independence assumption between groups, we can determine the variables that most significantly affect model response. They are called the driving variables. Driving variables are important to decision-makers because they control the response of the system and, therefore, require the most attention in data collection and parameter range specification. If the driving variables can be controlled, then the problem of optimization is also significantly reduced.

To summarize this expermental design--it allows the analyst to develop a series of short experiments, by use of screening techniques, which will identify an ordinal grouping of variables according to statistical significance. The ultimate purpose of this rather lengthy process is to identify the driving variables or, in other words, those parameters whose variability most significantly affect the model response. To a decision-maker this distinction is invaluable for the interpretation and use of a model and the confidence he places in its results.

Confidence

The final phase of towards-validation is the confidence phase. Confidence building is a process that begins with the first step in the conceptual phase. As the process of towards-validation proceeds, intuitive appeal or lack of appeal develops for the user. If real world data from field tests exists, it is a relatively simple task to run statistical tests of hypotheses that can quantitatively establish confidence. However, as stated earlier, towards-validation is applicable to models that have little or no data from the real world system. Three steps are suggested for enhancing confidence:

1. Statistical comparison of modified simulation runs with related data.

2. Examination of the cost-benefit of increasing information.

3. Full documentation of the towards-validation process.

Although real world data for the system being modeled may not exist, often systems that are already in use can provide data that will be useful in model testing and confidence building. For example, a model might be built to compare the capabilities of six air-to-air missiles that are proposed by six separate contractors. In order to save development costs, engineering specifications are often run through a simulation and only the top scoring designs are contracted to build prototypes for field testing. To establish confidence that the model will in fact be capable of comparing the six missiles, existing missiles can be simulated that have known capabilities and available data. Results can then be statistically analyzed to help establish confidence.

Military training exercises are another possible source of data, particularly for war-gaming models. However, limitations that artificially constrain training exercises must be considered and carefully included in the simulation input.

To this point in the towards-validation process, efforts have been made to limit the number of runs required to analyze the various aspects of the model. However, in the final analysis it might be determined that either the model falls short of the desired level of accuracy, or that the validation process falls short of providing the desired level of confidence. In either case the cost of seeking additional information must be weighed against the amount of information gained (Fig. 7). These costs can be measured by

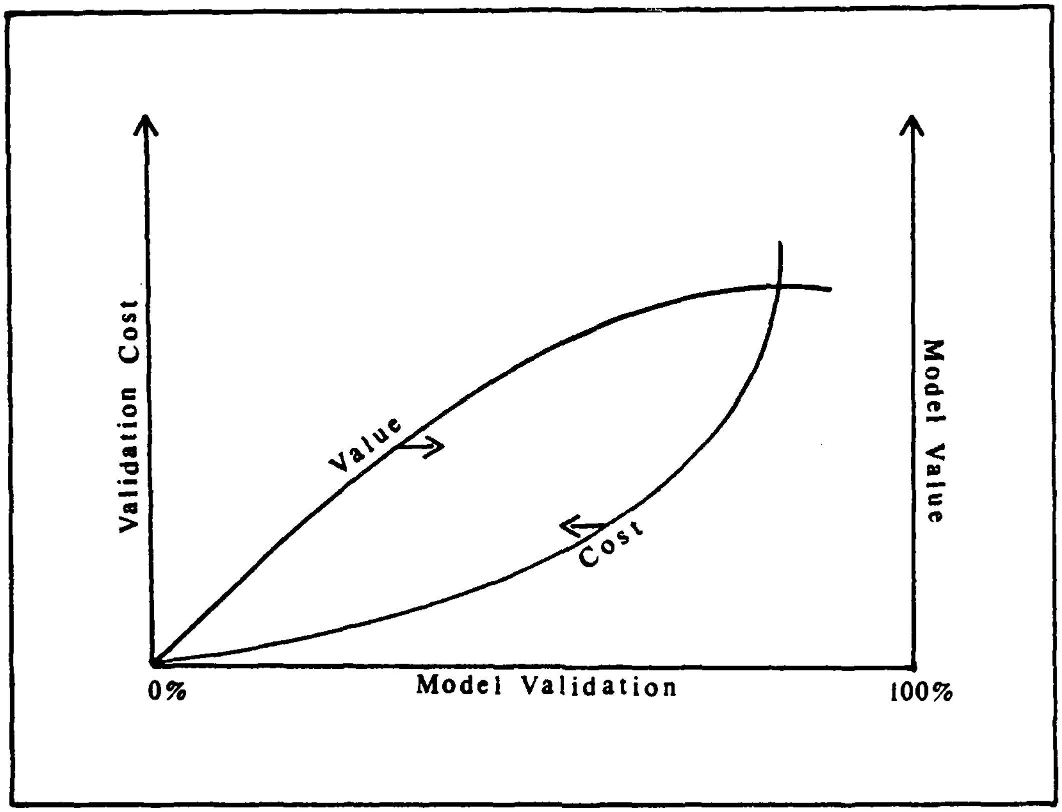


Figure 7. Cost of Additional Information

the additional number of simulation runs required to achieve a desired statistical confidence or by the cost required to rewrite sections of the model. At some point it will not be cost-effective to increase information.

The final step in the towards-validation process is documentation. A step-by-step description of all procedures, results, and analysis should be incorporated in the model's user's guide. This will insure that decision-makers, present and future, will have available the tools for establishing confidence in the model.

Comment

The problem of validating models is one that has been addressed by countless authors. The literature is filled with theoretical and idealistic approaches. However, most of these approaches fail when applied to complex, multivariable models that have little or no real world supporting data. Towards-validation addresses these problems and suggests a methodology for providing users the insight required for the decision-making process.

Unfortunately, towards-validation is not a "cure-all" for all validation problems. Several limitations reduce its applicability. These limitations include:

1. Variables must be able to be sufficiently screened in order to apply a workable experimental design.

2. Only portions of a model can realistically be analyzed if several hours of computer time are required for each simulation run.

3. Special consideration must be given models that have multiple response variables.

It is the responsibility of those tasked to perform the validation process to establish their own procedures using the guidelines presented. Creativity is the key. If, for example, the number of runs required is excessive, by carefully choosing the data base, the simulation runs from the sensitivity analysis can also be used for face validation and the confidence phase.

In conclusion, it is important to note that authors of validation schemes have identified one common need--the need for a substantiation process to develop confidence in the use of computerized models. If a model passes only some of the towards-validation requirements, or if due to budget and time constraints not all the steps were performed, the quality of information still exceeds that of no validation at all.

IV. THE MODEL

Introduction

In order to demonstrate the application of towards-validation, an existing DOD model was chosen. The Interceptor War Game Model was created by William R. Fischer, North American Aerospace Defense Command, Plans Division (NORAD/XPYA). The model is also known as the NORAD Air Defense Simulation Program or the Fischer Model. The only existing support documentation available includes: NORAD Technical Memorandum 75-5; "NORAD Air Defense Simulation Program (Fischer Model Methodology)," written by Fischer in 1975; and Staff Note 78/2, "A Brief Description and User's Guide to the Fischer Model," written by E. J. Edmund of the Canadian Air Opertional Research Directorate. However, since the three primary users (NORAD, Canada, and Headquarters Air Force) have developed somewhat different requirements, three versions of the model have evolved. As a result, no existing documentation accurately applies to any version. Furthermore, no attempt beyond face validation has been made to validate the model.

The version used for this study was provided by Air Force Studies and Analysis, Aerospace Defense Division (AF/SASI).

The Interceptor War Game Model

The Interceptor War Game Model is a general purpose air defense model. There are five principal components:

1. Enemy bombers, carrying gravity bombs and/or air-to-surface missiles (ASM), are categorized into raid classes according to size, radar cross-section, speed, and general defensive capability. The raids are then formed by assigning targets and penetration routes which include: a start time, turn points, altitudes, and speeds.

2. When an incoming raid enters either ground or airborne radar coverage (for a given cross-section and radar range), a raid detection occurs. Delays, representing response time and equipment capabilities, can be entered to slow positive detection.

3. Once a radar detection has been classified as a threat, interceptor aircraft are committed to engage the enemy. The number of interceptors committed is dictated by predetermined tactical strategy.

4. Calculations are made to insure that fuel is available to complete the intercept. If the intercept is possible, a probabilistic engagement is conducted and enemy destruction is determined as a result of multiple Monte Carlo tests. These tests include the interceptor's ability to: get airborne, detect the target, obtain a favorable firing position, and successfully launch armament.

5. Finally, the interceptors are returned to the nearest base where they are refueled and rearmed for future

commits. Again Monte Carlo tests are conducted to determine the probability of successfully readying the aircraft.

The versatility of this model allows for the simulation of a wide variety of scenarios ranging on a continuum from an all-out attack against the North American Continent to a limited attack of one bomber against one target. Four features of the model contribute directly to this versatility:

1. The number of bombers, interceptors and radars is limited only by the capacity of the computer being used.

2. The orbits of the airborne radars, the locations of ground radars and the operating specifications of both can be input by the user.

3. Airfields can be located as desired for either launch or recovery of interceptor aircraft.

4. Lastly, any type interceptor can be simulated by simply specifying the appropriate operating characteristics, such as: speed, range, fuel flow, weapon system capability, and so forth.

Model Uses

The versatility of the Fischer Model can be further exemplified by examining past projects. The Saber Shield/Saber Shield Alpha exercises and the follow-on interceptor study are two of the many projects that the Headquarters Air Force (HAF) version has been used for. In the Saber Shield exercises, the model provided data on several air defense force allocation alternatives, which helped decision-makers formulate Air Force inputs to the Program Objective Memoranda (POM).

The Fischer Model was also instrumental in the follow-on interceptor study. New generation fighter aircraft were simulated in varying scenarios to determine the replacement for the F-106 as the nation's front line air defense interceptor. Ultimately, the F-15 was chosen as the follow-on interceptor, and the data provided by the model proved invaluable.

Nevertheless, due to limitations and assumptions incorporated in the model, there are several classes of problems to which the model is not sufficiently sensitive. For example, a scenario that is dependent upon a roll-back tactic could not be simulated because ground damage is ignored. An extensive list of the limitations and assumptions is presented in Appendix A, " A User's Guide for the Interceptor War Game Model," which incorporates the towards-validation process.

Conversion to the Cyber Computer

All versions of the Interceptor War Game Model are written in the SIMSCRIPT II.5 simulation language. At the time of this research effort, the HAF version was operating on a Honeywell computer. Since the Control Data Cyber (CDC) was the computer available for this study, a conversion was required. Several subtle differences between the Honeywell and CDC SIMSCRIPT II.5 compilers arose:

1. Some word packing in the Preamble required altering due to the difference in word size.

2. Double precision was not required on the CDC.

3. Errors developed on the CDC when subscripted variables assumed a value of zero.

The problem with subscripted variables proved to be an insidious problem. Usually, the discrepancy was the result of a temporary entity being used in the FOR EVERY statement. On the CDC, if no elements in the FOR EVERY search were found, then the value of the temporary entity was changed to zero; whereas, with the Honeywell the variable maintained its old value. Thus, if the temporary entity was subsequently used as a subscript for one of its attributes, the CDC computer stopped execution with a mode error. To illustrate how the problem was corrected, the line of code:

```
FOR EVERY INT IN AC.ON(BASE) DO
```

where INT is the temporary entity, was changed to:

```
DEFINE X AS AN INTEGER VARIABLE
FOR EVERY X IN AC.ON(BASE) DO
LET INT=X
```

An in-depth discussion of input data requirements, operating instructions, and general characteristics of the model is presented in Appendix A.

V. APPLICATION OF METHODOLOGY

One of the main objectives of this research effort was to apply the towards-validation methodology to the Interceptor War Game Model. However, instead of placing the primary emphasis on the actual results, the emphasis was placed on how each step was accomplished and its value in building confidence in the model. This chapter, therefore, focuses on the insights gained during the methodology application, while Appendix A presents the actual results of the towards-validation process. A brief outline of the principal steps of this methodology is provided in Table I.

Conceptual Phase

Generally, the steps of the conceptual phase are accomplished prior to the development stage of a model. However, for an existing model, such as the Interceptor War Game Model, the validation process must be adapted to fit the model's present domain of application. The inputs for the conceptual phase, provided by Air Force Studies and Analysis, included: past projects, the degree of accuracy required for these projects, the identification and command level of users, and the assumptions and limitations derived from past usage. These inputs were then consolidated to formulate the formal written statement of present application. Care was taken to insure that the scope of this statement was broad enough to include the model's

TABLE I

THE TOWARDS-VALIDATION METHODOLOGY

I. Conceptual Phase

   A. Formal written statement of intended model application
   B. Specification of degree of accuracy desired
   C. Description of assumptions and limitations
   D. Structural model or framework for design development

II. Verification Phase

   A. Structured walk-through
   B. Verification of technical physical processes
   C. Simulation of predictable states
   D. Testing of stochastic events

III. Credibility Phase

   A. Face validation
   B. Sensitivity analysis

IV. Confidence phase

   A. Statistical comparison of modified simulation runs
   B. Examination of cost-benefit of increasing information
   C. Full documentation of the towards-validation process

application in past projects, yet narrow enough to allow for a precise validation effort.

The value of this formal written statement becomes apparent when considering the model for possible use on future projects. The insight required to initially determine the model's applicability to a project is gained, for the most part, from this declaration. Thus, prior to starting a project (rather than halfway through it) the suitability of the model can be previewed, saving time and resources.

The formal written statement was also useful in the other phases of towards-validation. For example, during identification of the driving variables, initial screening was accomplished by users knowledgeable in air defense. Their success was enhanced by reviewing the statement concerning desired simulation accuracy, which was included in the formal written statement of present application.

The assumptions and limitations of the model were also reviewed prior to initiating data generation. If the user can ascertain that proposed usage *will not result in a* violation of an assumption or limitation, confidence is built in the model's ability to produce worthwhile simulation results. However, if a limitation or assumption must be violated, the user should either abandon this model or alter the program's code. Such actions will prevent invalid data and subsequent erroneous inferences.

Listing the limitations and assumptions in the user's guide is a convenient method for making this valuable information readily available to users. The need to have a complete listing is illustrated in the Fischer Model by the armed AWACS's (Airborne Warning and Control System) inability to carry more than one type of missile. Existing model documentation does not mention this limitation, and it is highly improbable that even a frequent user would ever notice this coding feature. By having it in a user's guide, even the occasional user would avoid this type of data inconsistency. A listing of 25 limitations and assumptions is presented in Appendix A and was compiled from interviews with present users and from the knowledge gained by accomplishing the validation steps. Five of the limitations were provided by Air Force Studies and Analysis (AF/SASI), and the remainder were found during the towards-validation process.

The last step in the conceptual phase was to identify the dependent variables and to examine the flow and basic interactions of the model. This was accomplished by creating a structural model. Normally, a structural model will be complex for even a small number of variables. As the number of variables increases, the complexity of the structural model also increases, and eventually the benefits that could be realized from this step would be lost. To handle this problem, the 84 input variables in the Fischer Model were grouped into 19 functional categories, based on user knowledge of the real world system. These categories were then traced through the major routines and events of the computerized model, terminating at the dependent variables. Thus, the first-time user, because of a basic understanding of the model's flow, should be better prepared to run the simulation model.

The conceptual phase is perhaps the most useful phase in the towards-validation process. The domain of application, desired accuracy levels, and the assumptions and limitations should be reviewed during each step of towards-validation to insure basic principals are not violated.

Verification Phase

The verification phase is concerned with the mechanical validity of the model's design. To facilitate the accomplishment of this phase, the structured walk-through was designed to compile information required for subsequent steps while verifying event-path integrity. Since the model was already in use, this step also contributed to the initial phases of model familiarization.

The structured walk-through was accomplished by first developing a limited data base that would incorporate the interactions of all events and routines in the computer model. Next, a time-line was drawn to track event scheduling, and forms were established for accumulating a listing of stochastic processes, Monte Carlo events, and technical physical processes. Also charts were developed for identifying events and subroutines by their scheduling or calling event/routine, by a time-line entry, and by the real world function it supports (interceptor, radar, or raid, for example).

The first step of the walk-through was to examine the model set structures and to gain familiarity with defined entities and attributes. Then the computer code associated with the input process was investigated by using the limited data base mentioned above. The code was checked to insure that arrays were adequately dimensioned, alpha character strings were correct, proper values were assigned to the variables, and so forth. During this process, the first events were scheduled and several subroutines were called.

When the START SIMULATION statement was encountered, the first event on the time line was processed. The simulation was performed by hand until all subroutines and events had been examined. To insure that all branching operations functioned properly, the values of branching variables were varied through their feasible ranges.

After completion of the structured walk-through, all technical physical processes had been identified. Technical reports and appropriate textbooks were referenced to derive and verify all of the formulas and spherical geometry applications in the model. The only use of stochastic processes in the model was the generation of uniform variates. These values were used in six Monte Carlo steps that compared the uniform variates with input probabilities to determine the outcome of specified events. Another contribution of the structured walk-through was the identification of significant assumptions and limitations which would otherwise never have been observed. For example, the armed AWACS limitation discussed earlier was first recognized during this step. Even though this process was time consuming, the quality of information gained made the structured walk-through an extremely valuable step for building confidence.

For the simulation of a predictable state step, two methods were available for making the Interceptor War Game Model deterministic. The first method involved redefining the function RANDOM.F (the uniform variate function) so that it returned a constant, either zero or one. Thus, whenever a Monte Carlo step was encountered, either event success or failure was guaranteed. The other method included setting the input probability of an event success at either zero or one so that regardless of the uniform variate obtained from RANDOM.F, event occurrence or failure was again assured. Because the first method required altering the model's computer code and since changing the data was deemed easier, the latter method was chosen.

The next step was to create a scenario for which event outcomes could be calculated manually. These calculations were then used to evaluate simulation results of the same data. For example, the raid path of one bomber and the location of one interceptor, one base, and one ground radar were plotted with the aid of a navigational chart. Calculations, such as the time and location of radar detection and the time required for a bomber to travel a fixed distance, were then computed and compared with simulation results.

Next, the input data was changed to insure that all Monte Carlo events would occur. Since the difference between the cost of one iteration and ten iterations was insignificant, the simulation was performed ten times to further the confidence in the deterministic state of the model. To analyze the data generated by both manual and computer calculations, acceptable difference levels between the two methods were established. The tolerance chosen for this test required that all simulated times and distances be within ± 5 percent of the manually calculated values. These tolerances were easily satisfied.

To investigate other predictable states, the input data was changed to insure a zero probability of occurrence for Monte Carlo events on a one-at-a-time basis. This procedure resulted in a more thorough understanding of the sequential actions involved in each segment of a real world intercept. This further enhanced the user's ability to establish realistic input probabilities for the corresponding model variables. A summary of how these runs were structured is listed in Table II.

Since the predictable state investigation required several simulation runs, the question of whether the possible benefits of these runs would outweigh the cost had to be considered. However, at less than $0.75 per run for ten iterations (when using a binary source deck), cost was not a major factor. Also the sequential nature of the Monte Carlo events allowed the one-at-a-time runs to yield the maximum amount of information possible, rather than requiring a design that would include the investigation of

TABLE II

SUMMARY OF RUNS FOR PREDICTABLE STATES

Action	Results
1. All Monte Carlo variables set to 1.	1. Interceptor was successfully committed and launched, and it successfully detected, converted, and killed the target.
2. All Monte Carlo variables set to 0.	2. Interceptor did not get airborne.
3. Interceptor reliability (REL) set to 0.	3. Same as #2
4a. Probability of interceptor detecting a target being tracked by radar (PD.IN)set to 0.	4a. Target was not detected.
b. Probability of interceptor detecting a target not being tracked by radar (PD.OUT) set to 0.	b. Same as #4a.
c. Both PD.IN and PD.OUT set to 0.	c. Same as #4a.
5a. Probability of interceptor being able to obtain a favorable firing position on a frontal attack (PC.NOSE) set to 0.	5a. Interceptor successfully converted to A stern attack and killed the target.
b. Probability of interceptor being able to obtain a favorable firing position on a stern attack (PC.TAIL) and PC.NOSE set to 0.	b. Interceptor could not obtain a firing position, and target was not killed.

TABLE II (con't)

6a. Probability of the missile destroying the target after being launched on a frontal attack (PK.NOSE) set to 0.	6a. Missile missed target, but interceptor converted to stern and successfully killed target.
b. Probability of the missile destroying the target after being launched on a stern attack (PK.TAIL) and PK.NOSE set to 0.	b. All missiles fired missed the target.
7. Probability of being able to turn the interceptor (PR.TURN) set to 0.	7. Interceptor was not turned.

the probabilistic event interactions.

The last step of the stochastic processes examination was to investigate the variance of the output variables. Using realistic probabilities and a different random number seed for each iteration, a sample of ten simulation runs was obtained. Then the sample mean and variance were examined to insure that no extraneous factors were distorting the output variables.

Credibility Phase

To examine the intuitive and statistical appeal of the model, two steps were accomplished during the credibility phase. The first step, face validation, was limited to examining only the primary features of the model.

Furthermore, all runs from the sensitivity analysis and from the confidence phase were also investigated as part of face validation.

The scenarios used for this face validation step were developed from the data base built during the predictable state investigation. Ten different real world situations, listed in Table III, were simulated and examined by persons familiar with air defense operations. It was determined that the model did respond as expected, according to real world experience. The key to this procedure was to simulate only as many scenarios as was needed to allow the knowledgeable user to feel confident about the feasibility of simulation results. The main purpose of seeking agreement between the expert's intuition and simulation results was the enhancement of the user's confidence in the model's ability to provide usable data. Additionally, the fact that the model had been successfully used in the past contributed to the face validation process and ultimately the user's confidence.

As stated above, the face validation process was continued throughout the sensitivity analysis and the confidence phase. The data bases used in these steps, however, represented complex scenarios; therefore, only the feasibility of output results were face validated.

The sensitivity analysis consisted of a four step process. The objective was to identify those decision variables that most significantly affected the output

TABLE III

FACE VALIDATION COMPUTER SIMULATION RUNS

Situation Examined

1. Weapon selection logic
2. AWAC operations
3. Front/stern attack and reattack logic
4. Intercept geometry
5. Interceptor commit logic
6. Desired kill level logic (used to determine how many interceptors to commit)
7. Logic and event occurrence after the bomber turns, staying in radar coverage.
8. Logic and event occurrence after the bomber turns, exiting radar coverage
9. Logic and event occurrence after bomber turns, causing impossible intercept
10. Logic after bomber turns, maintaining a possible intercept

variables when varied over their expected range of values. The first step of the process required an air defense expert to screen the 84 model variables and to eliminate any variable that would not significantly affect the response surface. Two experts were used, and separate lists of variables were elicited. Discrepancies between the two lists of insignificant variables were resolved so that either full agreement was reached on eliminating a variable or else the variable was retained. Thus, borderline variables were retained, and a greater degree of confidence was placed on the removal of insignificant variables. This first step resulted in the elimination of 64 variables, leaving 20 variables for further screening.

For the Interceptor War Game Model, gathering sufficient data to statistically analyze 20 variables was too costly and time-consuming. Therefore, to further screen the variables, the second step of this analysis involved grouping the remaining 20 variables into five categories. The five categories chosen were radar, aircraft, basing, fire-control system, and command and control. Although alternative methods were available, the categorizing of the variables by real world functions was preferred. This method utilized the independence between the variables of one category and the variables of the other categories. Thus, each category could be individually statistically analyzed, which assumed no interactions existed between the five groups. As a result, more cost-effective data collection resulted.

Developing the experimental design and collecting the data for each variable was the third step in the sensitivity analysis. A $2^{(4-1)}$ fractional factorial design was used for the fire-control system, and command and control categories, while the aircraft category required a $2^{(8-3)}$ design. The remaining categories, due to the small number of variables in each, allowed for full factorial experiments. The use of two levels for each variable in each design was based on the linearity assumption. The designs used for these categories, listed in Table IV, were obtained from

TABLE IV

EXPERIMENTAL DESIGNS

CATEGORY	TYPE DESIGN	NUMBER VARIABLES/RUNS	BLOCKS (See Note)	EFFECTS
Radars	2^2 Full Factorial	2/4	None,a,b,ab	All interactions
Basing	2^2 Full Factorial	2/4	Same as Above	Same as Above
Command And Control	2^{4-1} Fractional Factorial	4/8	None,ab,ac,ad bc,bd,cd,abcd	All main and 2 Factor Interactions
Fire-Control	2^{4-1} Fractional Factorial	4/8	Same as Line Above	Same as Line Above
Aircraft	2^{8-3} Fractional Factorial	8/32	None,abcd,acfg,bdfg cdefg,abefh,adegh, bcegh,abcdefg,efg,bde, ace,abgh,cdgh,beth, adfh,abcdfgh,fgh,bdh, ach,abeg,cdeg,bcef, cdef,eh,abcdeh,acefgh, bdefgh,cdf,abf,adg, bcg.	All Main and These 2 Factor Interactions ae,ah,be,bh, ce,ch,de,dh, ef,eg,eh,fh,gh

Note: Variables to have low values are listed. If a 2 factor interaction was not measurable, independent variables were assigned to the letters for which the interactions were not measurable.

preconstructed tables (Ref 2:625; 7).

Data from the 21st NORAD Region exercise AMALGAM Chief 80-1 was used to create the data base for the fourth step of this analysis. The raid paths, the number of interceptors and their locations, the locations of ground radars, and the orbits of airborne radars were determined from the exercise scenario. To establish the values of the input variables, general knowledge (avoiding classification problems) was used to approximate parameters representative of present day systems, such as the F-106. Values representative of state of the art systems were used as the upper values for the 20 variables in the experimental designs, and values representative of older generation systems were used as the lower values.

A common basis of comparison was established by considering the variables associated with the raid function as constants. Since a factorial design dictates that each run must have its own data base, 56 data bases were constructed for the five experimental designs in this analysis. Five iterations per simulation run produced the responses needed to form the data points used to statistically analyze each category of variables.

After completing all of the necessary computer runs, the Statistical Package for the Social Sciences (SPSS) was employed to obtain regression data on each design. The stepwise regression analysis routine was selected for use instead of the analysis of variance (ANOVA) because of computer core memory requirements and other limitations which reduced the appeal of an ANOVA. The insight needed for determining the significance of variables was provided by the covariance matrix, the various indicators for the goodness of fit of the regression model (R^2 and adjusted R^2 values for instance), the analysis of variance information, and the statistical information about the coefficients of the regression variables.

As in the first step of this sensitivity analysis, the insignificance of a variable was the criterion used for its elimination from further consideration. The F-statistic significance of each variable in the regression model was the principal criterion used to determine significance levels. At a F-statistic significance level of 0.1, a natural break occurred between the variables in each category. As a result, all variables with a value greater than 0.1 were removed. The seven remaining variables were then identified as the model's driving variables (Table V). Furthermore, the covariance matrix for each regression revealed that, as expected, the variables within each category were nearly independent since the absolute value of all covariance values was less than 0.085. In addition this result increased the validator's confidence in the absence of multicollinearity.

A subjective evaluation of the results of this sensitivity analysis indicated that individuals familiar with the real world system were in agreement with the seven

TABLE V

Driving Variables

INT	Number of interceptors
REL	Interceptor reliability
LOAD	Interceptor weapon configuration
PK.NOSE	Probability of missile kill on front attack
PT.TAIL	Probability of missile kill on stern attack
MX.CDY	Scramble delay
RE.CDY	Recommit delay

variables identified as the drivers. This evaluation further built confidence in the conclusions drawn from the sensitivity analysis.

The primary value of sensitivity analysis was to identify those variables that produced the most variability in the response surface. Then, if time and resources are critical, they can be best spent on the data collection for these variables. Furthermore, if future simulation results are determined to be erroneous, the driving variables provide a starting point for searching out possible input data errors. The results of this sensitivity analysis contributed significantly to the validator's confidence in the Fischer Model.

Confidence Phase

The primary purpose of the confidence phase was to enhance the confidence built in the model by the three previous phases. Since real world data did not exist, exercise data was the closest facsimile available. Therefore, the exercise results, target tracks, and AWACS tracks for the AMALGAM Chief 80-1 exercise were again used. To effectively simulate this exercise, two data modifications were required. First, the simulation targets were input so as to originate at their exercise entry points, rather than originating at a base and then flying to the entry point as the exercise targets did. This modification prevented early radar detection and interception, and it kept the target track times the same for both the exercise targets and corresponding simulation targets. The second modification concerned incorporating the exercise targets' cross-sections, speeds, altitudes, and defensive countermeasures into four simulation target classes. This information was subsequently used when determining the probabilities of detection, conversion, and kill for an interceptor engaging a particular target class.

After the data base was completed, the exercise was simulated thirty times. Since the live exercise was not repeated, no statistical test of means or variance could be conducted. However, to contribute to the intuitive appeal of the model, a 90% confidence interval was constructed from the simulation results. The single exercise data point was found, in fact, to lie on the interval. The simulation results were also compared with the exercise results by examining the number of bombers killed, average target penetration, interceptor force attrition rate, intercept success rates, reasons for missed intercepts, and armament success rates. If data could be obtained for an exercise that had been repeated at least three times, a numerical statistical inference could have been made. However, even though a statistical test of hypothesis could not be performed, the comparison of one-time exercise results with simulation results contributed more tɔ the confidence building process than if the step had not been performed.

After completing all towards-validation steps, the overall confidence built in the model was reviewed. Had it been perceived that the model could provide the data needed to address the project under consideration, the model would be ready for use. But if the user was still indecisive about the model's ability to generate valid data, all questionable areas would have to be examined further. Before addressing any vague areas, however, the marginal cost of building the desired confidence--through additional testing or by rewriting parts of the model--must be compared with the added benefits to be gained.

Based on the results of this towards-validation process and face validation inputs received from previous users, it was felt that no further confidence could be built in the Fischer Model without more extensive exercise data or classified inputs for more accurate comparisons. Regardless, confidence building is a never ending process, and every time a model is run, confidence is effected. A full documentation of the towards-validation process is presented in Appendix A.

VI. RECOMMENDATIONS AND CONCLUSIONS

The towards-validation process was developed from a combination of theoretical techniques and tried-and-proven methods. The intent of this research effort was to provide an orderly procedure for the practical application of confidence building tests. However, since the validation process is model dependent, modifications are recommended and encouraged.

Other Possible Approaches

One approach to confidence building, often suggested by other authors, is the development of a simple parallel model. This could be accomplished by rewriting the main routine of the model and by including such simplifying steps as:

1. Making as many input variables constant as practical.
2. Eliminating complicated mathematical conversions such as the spherical coordinate system in the Fischer Model.
3. Eliminating subroutines by hand calculating values and inputting them as constants.

When complete, this new model (for a limited data scenario) should replicate the other model's output--within reasonable tolerances.

Another aspect of validation not previously addressed is the team concept of validation. The time needed to validate a model and the quality of work can be optimized by concentrating the efforts of a team consisting of: computer/model specialists, engineers, experts in the real world system, and decision-makers. If confidence can be gained by each of these specialists in their own area of expertise, then overall confidence is greatly enhanced.

Areas for Further Research

A major shortcoming of all validation procedures is a lack of quantifiable measures for intuitive concepts such as confidence. In the towards-validation process, statistical tests are used to build confidence; however, in the final analysis no method exists for quantifying the level obtained. Work in this area would require creative and original research and would be invaluable to the field of computer simulation.

Since sensitivity analysis is the most time consuming and costly step in this validation process, any procedure that reduces the number of runs required and/or increases accuracy would be beneficial. One such procedure might be a process whereby variables can be grouped by functional areas and quantified, for sensitivity analysis purposes, by a single index. For example, in the Interceptor War Game Model 21 decision variables are required to describe the general performance characteristics of an interceptor. The combination of these variables, each at a specified value, uniquely describes one aircraft. This would suggest that a single index could be calculated (off-line) that is a function of the 21 variables. A F-15, for example, might have an index measure of 16.8 whereas a F-106 might be 11.2. Since the values of the original 21 variables are varied uniquely for each aircraft, the calculated index could by itself be used in the sensitivity analysis with no loss in accuracy, yet a significant reduction in required runs.

A final area for suggested research is a refinement of towards-validation by further applications to other models. At this time there are at least two other models, developed for AFIT master's theses, being validated by the use of towards-validation, however, results are not currently available.

Conclusions

The intent of this research effort was to develop a methodology for validating complex models. As previously stated, validation--in the purest sense--can never be obtained; however, it is the contention of this report that any steps taken to build confidence are steps in the proper direction.

Towards-validation is a process that should begin with model conception and continue through implementation. Once confidence has been obtained in a model's ability to provide desired insights, data can be generated for decision-making purposes. However, the process should not stop there. On every occasion for which the model is run, results should be examined for feasibility. This is particularly important if modifications are made, or if the model is used outside its validated domain of application.

In conclusion, it is hoped that the application of this validation process to the Interceptor War Game Model will provide not only a framework for the use of towards-validation, but also significant insight for future Air Force use of the model.

Bibliography

1. Berg, L/C Donald J. "Catalog of War Gaming and Military Simulation Models, 7th edition." Published catalog. Studies, Analysis, and Gaming Agency (August 1977). (SAGA-180-77).

2. Davies, Owen L. (Editor). The Design and Analysis of Industrial Experiments. New York: Hafner Publishing Co., 1963.

3. Fishman, George S. and Philip J. Kiviat. Digital Computer Simulation: Statistical Considerations. Memorandum RM-5387-PR. Santa Monica, Ca: The Rand Corporation, November 1967.

4. Hermann, Charles F. "Validation Problems in Games and Simulation," Behavioral Science, 12:216-231 (May 1967).

5. Hunter, J. S. and Thomas H. Naylor. "Experimental Design," The Design of Computer Simulation Experiments, edited by Thomas H. Naylor. Durham, N. C.: Duke University Press, 1969.

6. Mendenhall, William and Richard L. Schaeffer. Mathematical Statistics With Applications. North Scituate, Mass: Duxbury Press, 1973.

7. National Bureau of Standards. "Fractional Factorial Experiment Designs for Factors at Two Levels." Issued 1957.

8. Naylor, Thomas H. and J. M. Finger. "Verification of Computer Simulation Models," Management Science, 14(2):B-92 to B-101 (October 1967).

9. Schlesinger, S., et al. "Developing Standard Procedures for Simulation Validation and Verification," Proceedings of the 1974 Summer Computer Simulation Conference, 927-933. LaJolla, Ca: The Simulation Councils, Inc., 1974.

10. -----. "Terminology for Model Credibility," Simulation, 32(3): 103-104 (March 1979).

11. Shannon, Robert F. Systems Simulation: The Art and The Science. Englewood Cliffs, N. J.: Prentice-Hall, Inc., 1975.

12. Tytula, Thomas P. *A Method for Validating Missile System Simulation Models*. Technical Report E-78-11 for U.S. Army Missile Research and Development Command. Redstone Arsenal, AL: USAMRDC, 1978. (Ad A055 689).

13. Van Horn, Richard. "Validation," *The Design of Computer Simulation Experiments*, edited by Thomas H. Naylor. Durham, N.C.: Duke University Press, 1969.

Appendix A

USER'S GUIDE FOR THE INTERCEPTOR WAR GAME MODEL

This user's guide was prepared by Captains Craig S. Ghelber and Charles A. Haley as part of a Masters of Science thesis entitled "A Methodology For Validation of Complex Multi-variable Military Computerized Models." It was presented to the faculty of the Air Force Institute of Technology, Wright-Patterson AFB, Ohio, December 1980.

Preface

The Interceptor War Game Model was created by William R. Fischer of the Plans Division, North American Aerospace Defense Command (NORAD/XP). The model is also known as the NORAD Air Defense Simulation Program or the Fischer Model. The three principal users of this model are NORAD, the Canadian Department of National Defense, and Headquarters Air Force Studies and Analysis (HAF/SA). Due to differences in requirements, three versions have evolved over the past few years. Some documentation does exist, however, no attempt has been made to update or expand it into a usable user's guide. Furthermore, no attempt has ever been made to validate this model--other than face validation. Therefore, the objectives of this document are to provide a practical user's guide for the HAF version and to present insights gained from the application of a validation methodology for the purpose of building confidence in the model's ability to provide usable data.

It is presumed that the reader is familiar with the computer simulation language, SIMSCRIPT II.5, and is familiar with Air Force air defense operations and terminology.

Craig S. Ghelber
Charles A. Haley

I. MODEL DESCRIPTION

General Description

The Interceptor War Game Model is a general purpose computer model that simulates air defense operations anywhere in the world. The model is primarily used as a tool for generating data needed to compare the relative effectiveness of alternative courses of action associated with: interceptor force makeup, force allocation plans, AWACS deployment, and so forth. To illustrate the model's intended application, the following is a list of recent projects in which it played an integral part.

1. In response to a Joint Chiefs of Staff (JCS) request, a study was conducted to investigate the joint United States and Canadian air defense capability. The Interceptor War Game Model simulated both present and proposed force capabilities against a continental threat. From the data generated the relative effectiveness of each force capability was determined, which led to results published in a JCS Report.

2. The model was used in the Saber Shield/Saber Shield Alpha exercises to provide insights needed to compare different air defense force allocation plans proposed for the 1980's. The results were then used by Air Force planners to formulate several of the Air Force's inputs to the Program Objective Memoranda (POM).

3. The Interceptor War Game Model was also

instrumental in the follow-on interceptor study. Candidates to replace the current F-106 fighter interceptor included the F-14, F-15, F-16, and F-18 fighters. The F-15 was eventually chosen, and the Fischer Model inputs were considered invaluable.

4. Exercise Blue Ice was conducted for the Department of Defense in response to a growing need for contingency plans for protecting the North Atlantic sea lanes. The model generated data on land based versus aircraft carrier based defense options.

These four examples of projects for which the Interceptor War Game Model has been used, demonstrate its versatility. It is this versatility that allows the simulation of a wide variety of scenarios ranging on a continuum from an all-out attack against the North American Continent to a limited attack of one bomber against one target. The principal feature of the model that creates such versatility is the manner in which data is input. Specifically, any type of interceptor can be simulated by simply inputting the appropriate operating characteristics, such as: speeds, range, fuel flows, and weapon system capability, for example. Similarly, airfields, for either staging and/or recovery of interceptors, may be located as desired. Also, the orbits of airborne radars, the location of ground radars, and operating characteristics of both are all user inputs. Only the physical capacity of the computer used to run the simulation limits the size of the scenario that can be studied.

The Interceptor War Game Model is written to simulate the following five real world functions:

1. Enemy bombers, carrying gravity bombs and/or air-to-surface missiles (ASM), are categorized into raid classes according to size, radar cross-section, speed, and defensive capability. Raids are then formed by assigning flights of bombers to planner designed penetration routes and targets.

2. The detection of a raid occurs when it enters either ground or airborne radar coverage. Delays representing response times and equipment capabilities may be entered to slow positive detection.

3. Once a radar detection has been identified, interceptors are committed to engage the target. The number of interceptors committed is dictated by predetermined tactical strategy, which will be addressed later.

4. Prior to committing an interceptor, calculations are made to insure that sufficient fuel is available to complete the intercept and to recover at the closest base. If the intercept is possible, a probabilistic engagement is conducted and target destruction is determined by the results of several Monte Carlo tests. These tests include the interceptor's ability to: get airborne, detect a target, obtain a favorable firing position, and successfully launch armament. The Monte Carlo technique is used to insure that only whole numbers of bombers and interceptors appear in the output.

5. Finally, the interceptors are recovered at the nearest base where they are refueled and rearmed if the services are available. The prɔbability of a successful turn is determined by another Monte Carlo test.

The manner in which input variables interact within the context of the above functional areas can be presented (in a simplified manner) pictorially by a structural model. To facilitate simplicity the input variables are grouped into real world functions (Fig. A-1). Then these groupings or categories are traced through the main events and routines of the computer model (Fig. A-2). The structural model terminates at the dependent variable, or as in this case, the bombs dropped.

The versatility of this model, however, is restricted by assumptions and limitations. These restrictions make the Interceptor War Game Model inappropriate for some classes of problems. Table I-A presents a list of many of the assumptions and limitations.

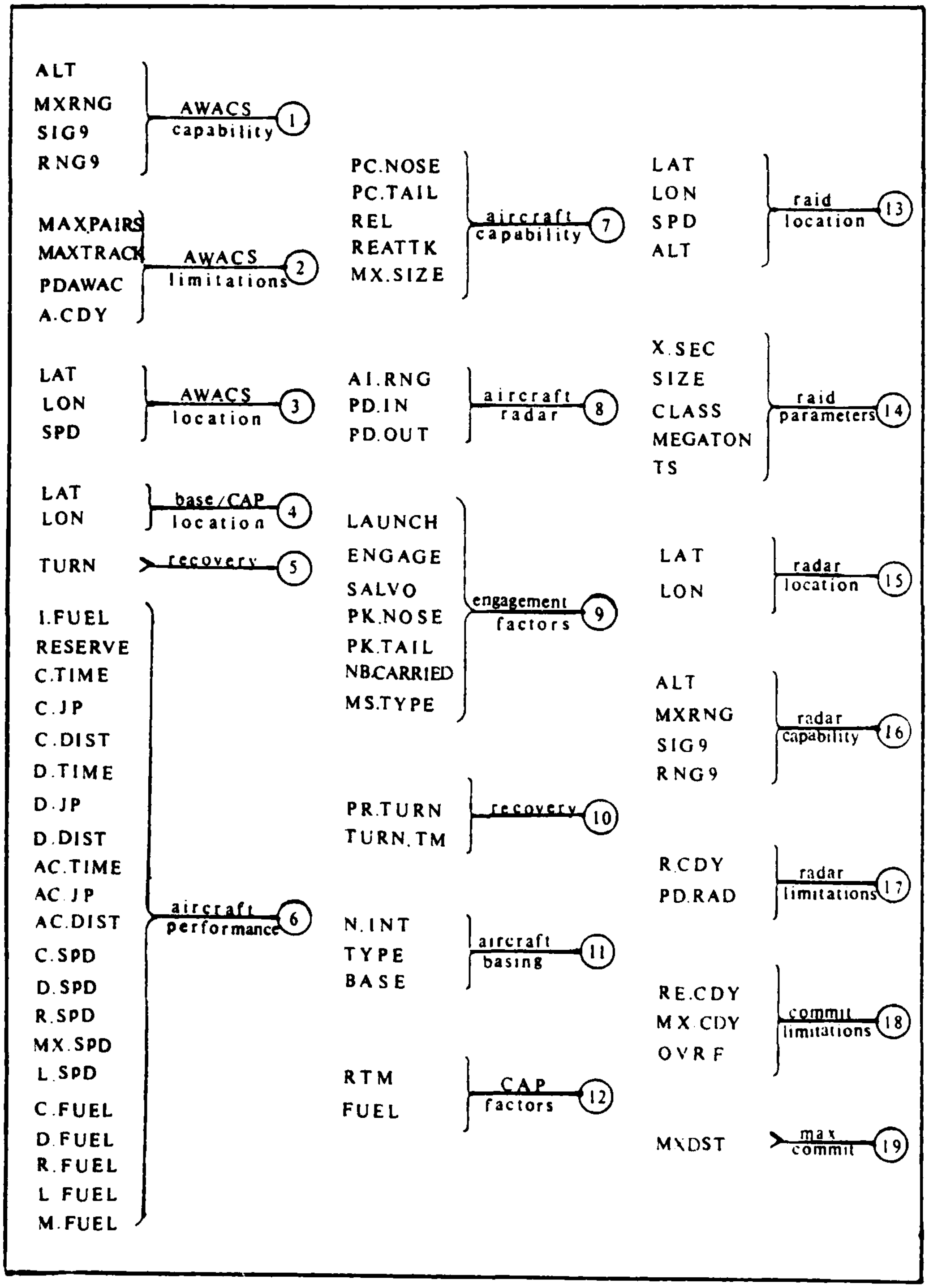


Figure A-1. Variable Categories

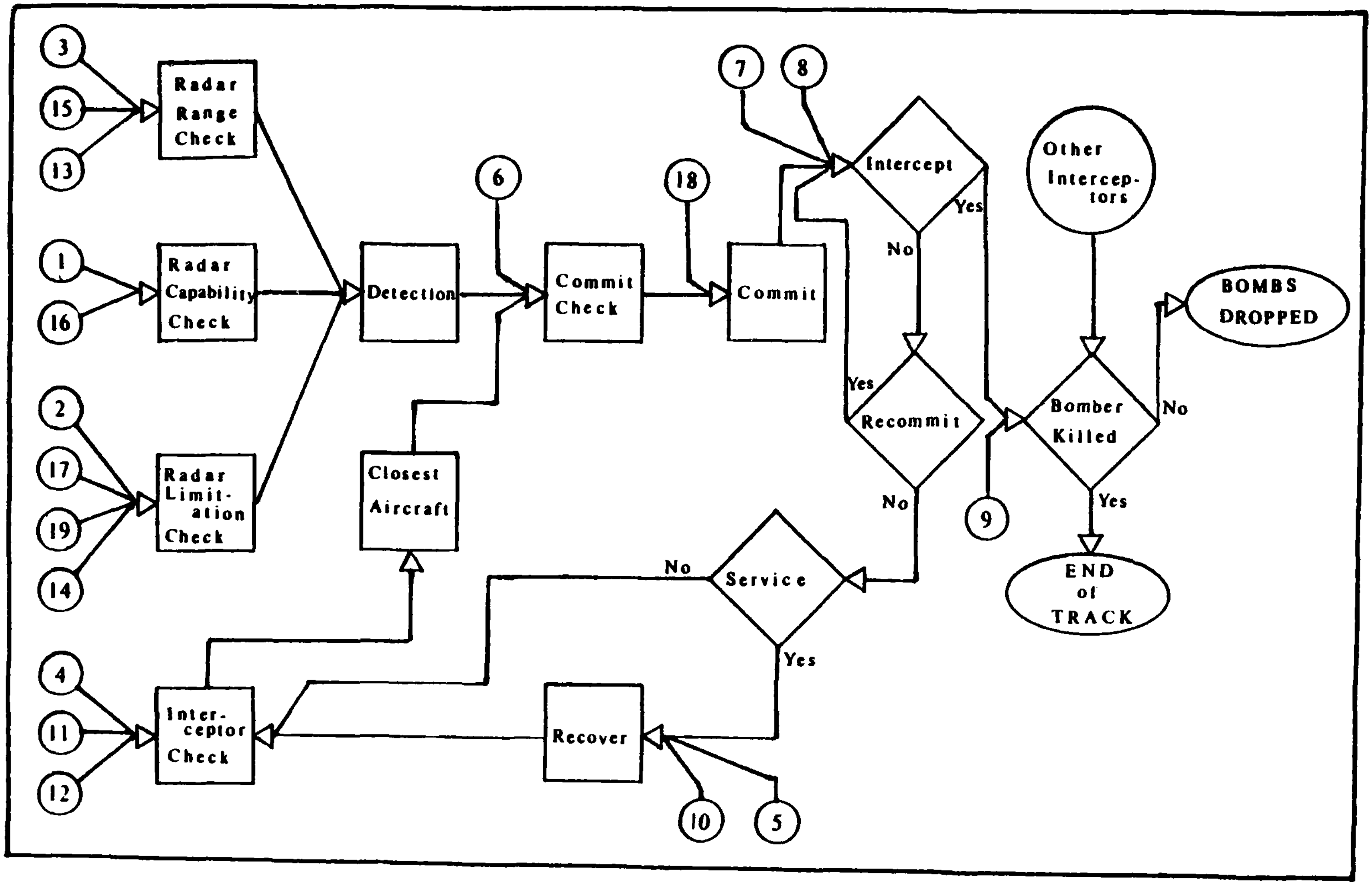


Figure A-2. Structural Model

TABLE I-A

ASSUMPTIONS AND LIMITATIONS

A. Command and control is not considered, however, some delays can be input.

B. Communication jamming can only be simulated by reducing probability of detection or conversion.

C. Enemy tactics to cutoff communication or resupply lines can not be simulated because ground damage is not considered.

D. Electronic counter-measures (ECM) can only be simulated by reducing probabilities of detection, conversion and kill, and by increasing delay times.

E. Surface-to-air missiles (SAM) and other area defense measures are not modeled.

F. Bombers do not have defensive capabilites.

G. Terrain masking is not considered.

H. Pilot and radar operator capabilites are not considered other than as constant input delays.

I. No provisons are included in the model for escalating from a peace-time environment to all-out war.

J. AWACS have 100% reliability and never run out of fuel or get shot down.

K. Armed AWACS can only be armed with one type missile or erroneous data will result.

L. Bomber raids are detected according to the cross-section of only one bomber and all bombers in a raid must have the same characteristics.

M. All aircraft turns and changes of altitude and airspeed are instantaneous.

N. Once an interceptor aborts, the delay to commit another fighter is always constant.

TABLE I-A (Con't)

O. Autonomous operations are not simulated.

P. Strategic orbit points (STOP) can not be remanned once an interceptor has left the STOP.

Q. No altitudes are established for STOPs; therefore, fuel required to transition to and from these points is not considered.

R. If a target turns, all interceptors committed on it are put on STOP and reassessed by routine RECOMIT. No time is lost if the fighters are recommitted, however, it is possible to take an interceptor out of a legitimate firing position.

S. Interceptors are recovered at the nearest designated base regardless of fuel and armament availabiltiy.

T. Turnaround times are constant for each type aircraft regardless of base workload.

U. A 4/3 radius earth is assumed for line-of-sight calculations.

V. Only one flight of interceptors is committed on a raid regardless of circumstances, until an interceptor successfully identifies the raid. Then, if required, more fighters are committed.

W. The number of interceptors, radars, bombers and raids is limited only by the capacity of the computer being used.

X. Once an aircraft aborts an intercept or a scramble, it cannot be fixed and returned to the war.

Y. If an interceptor is committed fcr a scramble and the target is killed before the interceptor gets airborne, that aircraft will not be available for further commitment until it again goes through commit delays.

Methodology

The cornerstone of this model is the raid function, since the program reacts to the actions taken by an incoming raid. Each penetration route is defined by a set of checkpoint data cards. Each of these data cards indicates the latitude and longitude of the checkpoint and the altitude and speed for the raid's next leg. When computer time-line calculations indicate that a checkpoint has been reached, checks are made to ascertain whether the raid will enter, exit, or remain in/out of radar coverage. Also all interceptors are reevaluated, resulting in a new commitment sequence.

Bombers are capable of releasing a gravity bomb or an ASM at any checkpoint except the first or last. As many weapons are released as there are bombers remaining in the raid. Once an ASM is fired, it becomes its own raid with a separate cross-section, speed and altitude, and is also capable of being shot down.

The maximum range for detecting a raid is a function of the radar's maximum detection range, the target's radar cross-section, and the altitudes of the radar and target. However, before the detection process can be completed, an input delay time must pass.

Radars are created by specifying latitude, longitude and operating characteristics. Similarly, AWACS operating characteristics are input; however, the orbits are specified by at least two checkpoints. Furthermore, AWACS can be flown to the orbit point by using another set of navigation checkpoints. The orbit start time can be either a specified or random time. All radars have a 360 degree circular coverage pattern, and in addition, the AWACS has a moving coverage whose antenna height is the aircraft's altitude.

After a raid has been detected, interceptors are committed on that raid (not individual bombers). To determine which interceptors to commit, each uncommitted fighter is examined for intercept feasibility and then time to intercept. The fighter with minimum time to intercept and enough fuel available for recovery is then chosen. The maximum number of fighters in a flight is an input value, and until a raid has been successfully intercepted, only one flight of interceptors is committed. Once the number of bombers in a raid is determined, the appropriate number of fighters to satisfy a predesignated overcommitment ratio is sent to engage the incoming raid.

Interceptors are scrambled from their preassigned bases so as to reach the computed intercept point at the calculated time. Separation on takeoff between aircraft in a flight is 30 seconds, to simulate a three mile in-trail weather departure.

When an interceptor reaches the intercept point, a Monte Carlo test is made to examine aircraft reliability. If the aircraft is found mechanically capable of accomplishing the mission, probabilistic values are tested for detection and conversion. If the interceptor successfully converts to either the front or stern, the armanent which provides the highest probability of kill (PK) for the attack flown is chosen. The maximum number of weapons launched on each firing pass is an input function. Next, the PK is tested against a uniform variate on the interval zero to one to determine whether the bomber was killed or not. Then the interceptor's fuel state is reevaluated, and the aircraft either recovers or continues fighting by reattacking or assuming a STOP for further commitment.

Once an interceptor has either fired all its armament or has reached its recovery fuel state, the recovery logic is entered. The base chosen for landing is simply the closest base identified to handle that specific type fighter. Fuel and armament availability is not considered. Then, if the aircraft passes a Monte Carlo turn-around test, it is readied for further commitment following predetermined delays. Any aircraft that lands because of an aborted intercept or that fails to get airborne is not turned and is dropped from further consideration.

After all the bombers are destroyed or have reached the end of their penetration routes, all interceptors are recovered and the simulation is terminated.

Technical Aspects

Model Structure. To better understand technical aspects, the model's structural properties must be examined.

The HAF version of the Interceptor War Game Model is composed of 12 event notices, 30 subroutines, and the rèquired PREAMBLE and MAIN routines. The PREAMBLE establishes the framework for the model. Its code is well documented with comment cards. The MAIN routine is primarily used to accomplish three taskes:

1. Default values are established and input data is read.

2. The simulation is started and the routine to print results is called.

3. If more than one iteration is specified, the program is prepared for another simulation run by re-initializing variables.

Event notices and subroutines can be divided into six general categories as illustrated in Table II-A. It can be seen that most of these event notices and subroutines relate directly to a real world system and are labeled descriptively.

The sole purpose of several routines is to convert units of measure to radians, while other routines convert the simulation units back to the original units for output. For example, routine AI converts latitude and longitude inputs from a degrees and minutes format to radians, and AO converts the radian results back to degrees and minutes. Similarly, values input as nautical miles (nm) are changed to radians; minutes are changed to days; and knots are converted to radians per day (Table III-A). Radian units

TABLE II-A

EVENTS AND SUBROUTINES

Category	Events	Subroutines
Ground radar	ENRAD	DETRK
	XRAD	GENINT
		RRNG
AWACS	ATURN	DELETE.RAID
	AW.SHOOT	DETRK
	DETCT	LOCATE.AWAC
	XAWAC	RRNG
		SURVEY
Interceptor	INTERCEPT	DELETE.RAID
	LAND	DRKN
	LVSTOP	EKILL
	READY	FILL.PTCM
		NEAREST
		POSSIBLE
		RECOMIT
		RECOVER
		UPDATE
Raid	NXLG	FIND.RAID
		FORECAST.RAID
General	COMIT	FINISHED
		INPUT.SUMMARY
		OUTPUT
Calculations		AI
		ANGLE
		AO
		BOMB.COUNT
		COSDS
		DI
		DIST
		DT
		LATLON
		STP.NM
(Not used)		CALC

TABLE III-A

CONVERSION FACTORS

```
1(nm)      = 1(nm)*(1degree/60nm)*(PIradians/180degrees)
           = PI/10800(radians)

1(knot)    = (1nm/hour)*(24 hours/day)*(PIradians/nm)
           = PI/1440(radians/day)

1(minute)  = (1min)*(1hour/60min)*(1day/24hours)
           = 1/1440(days)
```

are calculated because spherical (or great circle) geometry is used extensively in this model for calculating distances, geographic locations, and computing angles used in solving intercept geometry problems.

One process that does not involve radians is the computation of the radar detection range for a given target. Line-of-sight distance between radar and raid is used and is determined by the equation:

```
RANGE1 = RNG9*[(X.SEC/SIG9)**0.25]
```

RNG9 is the maximum radar range that a target with a cross-section of SIG9 can be detected; and X.SEC is the radar cross-section of the bomber. However, the line-of-sight may be limited by the radar horizon, which is computed by the equation:

RANGE2 = [SQRT(Height Radar)+SQRT(Height Raid)]*1.228

Then the actual detection range becomes the minimum of RANGE1, RANGE2, or the input value for maximum radar detection range (MX.RNG). These equations hold for both ground based and AWACS radars.

The number of interceptors committed on a raid is calculated by first determining the expected number of kills (EK) per interceptor for a front attack. EK is the product of the interceptor's reliability, and the probabilities of detection, conversion, and kill (for the missile with the highest PK).

EK = REL*PD.IN*PC.NOSE*PK.NOSE

If multiple launches are allowed, PK.NOSE is replaced by:

PK(Mult Launch) = 1-(1-PK.NOSE)**N

N is the number of missiles salvoed. For stern attacks the appropriate TAIL values are substituted. Then as each interceptor is committed, the EK value is subtracted from the desired kill level (DESKL), which is the product of the size of the raid and the input overcommitment ratio (OVRF):

DESKL = OVRF * SIZE(RAID)

Interceptors are committed until DESKL is no longer positive. However, care must be taken when establishing an OVRF value. If the attacking force is larger than the defending force, an OVRF of greater than one could cause interceptors to be concentrated on only a few bombers at a time.

Credibility. The credibility of the Interceptor War Game Model was investigated by the application of a formal validation process. Part of this investigation included the examination of predictable states. In order to make the model predictable, all input probabilities were given the value of one to insure the success of every event, and all delays were set to zero. Then a scenario was established for a one bomber raid at 40,000 feet and 480 knots approached a radar site from a range of 300nm. One F-106 interceptor was located near the radar site.

Hand calculations indicated that for specified radar inputs, the bomber should have required 7.5 minutes to reach radar detection--240nm from the radar site. The simulation results indicated that the bomber took 7.54 minutes to reach a detection point 237.81nm from the site. These results were within a pre-selected tolerance level of ±5 percent. Additional investigation of the intercept and engagement processes further verified that simulation results were within hand calculated tolerance limits.

To continue the investigation of predictable states, input probabilities were set to zero on a one-at-a-time basis so that each Monte Carlo event might be examined.

Then each scenario was simulated, and model reaction to system failures was investigated (Table IV-A). It was found that when probabilistic parameters were input at extreme values, simulation results were as expected.

Finally, scenarios were developed to examine:

1. AWACS and ground based radar operations;
2. Commit, attack, and weapon selection logic;
3. The effects of raid turns on intercept geometry and logic.

The output features of the model allowed for easy tracking of these functions; therefore, it was a simple matter to verify that the above logic worked as expected.

The next step of the credibility investigation was a critical examination of simulation results by air defense experts. Many complex scenarios were developed and simulated. Consistently, results proved to be within feasible ranges established by the experts.

Statistical Analysis. Meaningful statistical testing of the Interceptor War Game Model is limited by the fact that real world data does not exist. However, since the primary function of this model is to provide data on alternative actions, sensitivity analysis of the 84 input variables will provide invaluable insights. The objective of this analysis was to determine the driving variables--those variables that most significantly affected the variance of the dependent variable, total bombs dropped,

TABLE IV-A

SUMMARY OF RUNS FOR PREDICTABLE STATES

Action	Results
1. All Monte Carlo variables set to 1.	1. Interceptor was successfully committed and launched, and it successfully detected, converted, and killed the target.
2. All Monte Carlo variables set to 0.	2. Interceptor did not get airborne.
3. Interceptor reliability (REL) set to 0.	3. Same as #2
4a. Probability of interceptor detecting a target being tracked by radar (PD.IN)set to 0.	4a. Target was not detected.
b. Probability of interceptor detecting a target not being tracked by radar (PD.OUT) set to 0.	b. Same as #4a.
c. Both PD.IN and PD.OUT set to 0.	c. Same as #4a.
5a. Probability of interceptor being able to obtain a favorable firing position on a frontal attack (PC.NOSE) set to 0.	5a. Interceptor successfully converted to A stern attack and killed the target.
b. Probability of interceptor being able to obtain a favorable firing position on a stern attack (PC.TAIL) and PC.NOSE set to 0.	b. Interceptor could not obtain a firing position, and target was not killed.

TABLE IV-A (con't)

6a.	Probability of the missile destroying the target after being launched on a frontal attack (PK.NOSE) set to 0.	6a.	Missile missed target, but interceptor converted to stern and successfully killed target.
b.	Probability of the missile destroying the target after being launched on a stern attack (PK.TAIL) and PK.NOSE set to 0.	b.	All missiles fired missed the target.
7.	Probability of being able to turn the interceptor (PR.TURN) set to 0.	7.	Interceptor was not turned.

when varied over their range of possible values.

To facilitate sensitivity analysis an initial screening resulted in the elimination of 64 variables--due to their intuitive insignificance based on the judgment of air defense experts. The remaining 20 variables were then grouped into five categories by real world function, which assumed independence between categories. Thus, each grouping was individually statistically anaylzed--since the independence assumption implied that no interactions existed between groups.

To minimize data problems factorial experimental designs were developed for each of the five categories. A scenario was created to insure that all modeled processes would be tested. Then as each design dictated, the needed data bases were constructed and the model was run to produce the data points required to statistically analyze each experimental design. To establish a common basis for comparison, the same raids were used in all simulation runs.

The Statistical Package for the Social Sciences (SPSS) was used to obtain stepwise regression data on each design. The F-statistic significance of each variable was used to screen the remaining variables. It was found that a natural break existed at a significance of 0.10. As a result seven variables were classified as driving variables (Table V-A). To illustrate the regression information provided by SPSS, Table VI-A presents the results of the weapon system category analysis.

The seven driving variables identified by this sensitivity analysis produce the most variability in the output variable. If time and resources are critical, they might be best spent on the data collection for these variables.

In an attempt to generate other statictical inferences, data from a live exercise, the 21st NORAD Region AMALGAM Chief 80-1, was obtained. A data base that closely represented the exercise was created, and 30 simulation runs were made. A 90% confidence interval for these observations was calculated:

$$\bar{y} \pm t(.05,29)*[S/SQRT(29)]$$

$$25.96 < \bar{y} < 26.84$$

Even though the exercise result of 26 bombers killed falls

AD-A094 814 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 9/2
A METHODOLOGY FOR VALIDATION OF COMPLEX MULTI-VARIABLE MILITARY--ETC(U)
DEC 80 C S GHELBER, C A HALEY

UNCLASSIFIED AFIT/GOR/MA/80D-3 NL

2 OF 2

AD A
094814

END
DATE
FILMED
3-81
DTIC

TABLE V-A

DRIVING VARIABLES

INT	Number of interceptors
REL	Interceptor reliability
LOAD	Interceptor weapon configuration
PK.NOSE	Probability of missile kill on front attack
PT.TAIL	Probability of missile kill on stern attack
MX.CDY	Scramble delay
RE.CDY	Recommit delay

in this interval, no profound statistical conclusion can be drawn, due to an insufficient number of exercise data points.

The next chapter presents the input data card format and information on constructing the data base.

TABLE VI-A

SPSS RESULTS FOR WEAPON SYSTEM CATEGORY

Regression Model: Bombs Dropped = $B_0+B_1(\text{LOAD})+B_2(\text{PK.NOSE})$
$+B_3(\text{PK.TAIL})+B_4(\text{OVRF})$

Regression: Bombs Dropped with LOAD, PK.NOSE, PK.TAIL, and OVRF

Results: Bombs Dropped = 16.91-1.04(LOAD)
-4.08(PK.NOSE)-9.15(PK.TAIL)

Variable	F-Value	Significance	T-Statistic	R Square Contribution
LOAD	20.18	.00	-4.51	.19
PK.NOSE	8.23	.007	-2.88	.07
PK.TAIL	41.47	.00	-6.47	.39
OVRF	1.30	.26*	-1.14	.01

*This indicates that OVRF does not meet a 0.1 significance level.

Goodness of fit indicators:	R Square	.660
	Adjusted R Square	.632
	Residual Degrees of Freedom	36

Note: Correlation coefficients between all variables were 0.0, which indicates the independence of variables and absence of multicollinearity.

II. INPUT DATA CARD SUMMARY

The Interceptor War Game Model is designed to read data from two input files, designated as UNIT 10 and UNIT 12. UNIT 12 is the data base from which UNIT 10 selects, by file name, the specified data required for a particular simulation run. The objectives of this chapter are to present the detailed input card requirements and to illustrate the use of the two input files.

General Characteristics

Most data card entries are read by a free form READ statement; however, a formatted READ is used for key words which are generally found in the first six columns of a card. Free form implies that one or more blank spaces follow each entry. The last column for each card is specified as column 72. To illustrate each type of data card, the format "/KEYWORD X_1 X_2 X_3" will be used. The symbol "/" indicates the start of each card; and when required on formatted READ data, a "b" will be used to indicate a mandatory space.

The units of measure commonly seen in aviation are used for data input. All latitudes and longitudes are input in the form "DD.MM", where DD is degrees and MM is minutes. Positive numbers are for west longitudes and north latitudes and negatives for east and south. Similarly, ranges are in nautical miles, times are in minutes, fuel is measured in pounds, and speed in knots.

Data Cards for UNIT 10

Three types of input cards are associated with UNIT 10: the title card, file cards, and ALL.IN card. A sample Unit 10 file is presented in Attachment 1-A.

Title Card

/Title of data base

The first inpuι data card must be a title card. All 72 columns of the card can be used for a title which is printed at the top of each page of output.

FILE Card

/FILEbb Name Comment

The file card is used to transfer control from UNIT 10 to UNIT 12. "Name" is read free form and should be six or less alpha characters. This is the title given to a block of data cards that are to be used for the simulation run. "Name" appears on a UNIT 12 XNAMEX card.

ALL.IN Card

/ALL.IN Comment

As the last card of this file, ALL.IN signifies the end of the input data requirement.

Data Cards for UNIT 12

The structure of UNIT 12 involves eight blocks of input cards separated by XNAMEX cards. Each block contains the cards relevant to a particular function. The order of the blocks and the order of the cards within each block is usually unimportant; however, there are exceptions which will be noted. A sample UNIT 12 is presented in Attachment 2-A.

XNAMEX Card

/XNAMEX Name Comment

XNAMEX cards must be used as the first and last card of this file and designate the start of a block. As stated above, "Name" is the file specified on UNIT 10.

The first block of data cards specifies the control parameters and consists of eight cards: DELAYS, MXDST, OUTPUT, OVRF, PDAWC, PDRAD, PKLVL, and RANDOM.

DELAYS Card

/DELAYS A.CDY R.CDY RE.CDY MX.CDY Comment

Four inputs are used in the Interceptor War Game Model to represent command and control and operator delays. A.CDY and R.CDY specify the time between target detection and commitment for AWACS and ground radars respectively. RE.CDY is the time delay after an interceptor is placed on STOP before it can be recommitted. Finally, MX.CDY is the time required to get an interceptor airborne after a commitment. These times are input as real numbers; for example, one minute 30 seconds is input as 1.5 minutes.

MXDST Card

/MXDSTb Distance Comment

"Distance" is used to specify a maximum intercept commit range. This prevents the model from scrambling interceptors on raids that are an excessive distance away. The MXDST card can be omitted if the 600 nautical mile default range is adequate.

OUTPUT Card

/OUTPUT X_1 X_2 X_3 X_4 X_5 X_6 X_7 X_8 X_9 X_{10} X_{11} X_{12} Comment

The OUTPUT card fills the array OPTION(X_i) with the 12 values used to control the different output reports. A zero on this card causes the corresponding output report not to be printed, and conversely, a one allows the printing of the report. If all of the reports are desired the OUTPUT card can be deleted, since default values are ones.

OPTION (X_1): Chronological listing of events are printed from routine OUTPUT.

OPTION (X_2): Input data summary printed by INPUT.SUMMARY.

OPTION (X_3): Permits the accumulation of all interceptor commits by raid. This information is gathered in routine OUTPUT, but printed as part of the raid summary; therefore, to obtain this information OPTION (X_4) must be set to one.

OPTION (X_4): Print the raid summary by routine FINISHED. With both OPTION (X_3) and OPTION (X_4) set to one, a by-raid summary of significant events over each raid's penetration route is printed. If only OPTION (X_4) is set, a by-raid summary of only the raid's checkpointn is printed.

OPTION (X_5): With OPTION (X_4) and OPTION (X_5) set to one, the summary of how close a raid came to reaching its target will be printed from routine FINISHED.

OPTION (X_6): Controls the printing by routine FINISHED of random events such as: air aborts, interceptor detections, missile kills, and AWACS and radar detections.

OPTION (X_7): Summary of detonations from routine FINISHED.

OPTION (X_8): Elapsed time is printed during execution by routine OUTPUT. This option is a diagnostic tool and is generally not used.

OPTION (X_9) - (X_{12}): Not used, but a value must appear.

OVRF Card

/OVRFbb Value Comment

The OVRF "value" is the overcommitment ratio and is input as a real number. This factor changes the number of interceptors committed on a raid in order to control the expected kill ratio. Default value is 1.00.

PDAWC Card

/PDAWCb 0.0 T_1 P_2 T_2 ... 1.0 T_n *

The PDAWC card defines the cumulative probability distribution for AWACS detection times. For example, "/PDAWC 0.0 0.5 1.0 1.0 *" indicates a 0.0 probability of detection 0.5 minutes after a target enters radar coverage and a 1.0 probability at 1.0 minute. An asterisk or some other limit character must be placed after the last time entry.

PDRAD Card

/PDRADb 0.0 T_1 P_2 T_2 ... 1.0 T_n *

The PDRAD card is identical to the PDAWC card except the cumulative distribution is for ground based radars.

PKLVL Card

/PKLVLb Value Comment

The PKLVL "value" is used in determining the number of missiles that will be launched on a single pass. If missile probability of kill (PK) is below this "value", the model salvoes enough missiles to obtain the required PK. The default is 0.50.

RANDOM Card

/RANDOM Iterations Seed Comment

"Seed" is the value that is used to initialize the random number stream by setting SEED.V(1) equal to the "seed" value. "Iterations" is the number of complete simulations per computer run. This card can be omitted for default values of one.

The next block of input cards contains the parameters for all armament types carried by interceptors. This set of cards must precede the interceptor specification block. There are three input cards in this block: MTYPE, PKNOSE, and PKTAIL.

MTYPE Card

/MTYPEb Name Comment

The MTYPE card specifies the "name" of a missile type and signals that PKNOSE and PKTAIL cards will immediately follow.

PKNOSE Card

/PKNOSE PK_1 PK_2 ... PK_n

The PKNOSE card indicates, for each raid class from 1 to n, the PK of the missile when it is launched on a frontal attack. "PK_1" to "PK_n" are probabilities between zero and one.

PKTAIL Card

/PKTAIL PK_1 PK_2 ... PK_n

This card is the same as the PKNOSE card except for a stern attack.

The third block contains 15 different cards used to describe each type of interceptor (ITYPE). This block must follow the missile types and must precede the AWACS cards (if fighter-AWACS are used) and base specification cards. The first eleven cards presented are required, but their order can be varied--except for the ITYPE card which must be first. The last four cards: INTERN, LAUNCH, REATTK, and SALVO are optional since default values exist for each.

ITYPE Card

/ITYPEb I.NAME AI.RNG FULL MX.SIZE TURN.TM PR.TURN REL
RESERVE Comment

The ITYPE card specifies the name of an interceptor type and several parameters relevant to all interceptors of this type. This card also signals that a set of parameter cards will follow prior to the next ITYPE card.

I.NAME: six character (or less) label for the interceptor type.

AI.RNG: interceptor's radar range.

FULL: total fuel capacity for this type of interceptor.

MX.SIZE: maximum flight size.

TURN.TM: time required to turn this interceptor type.

PR.TURN: probability used in the Monte Carlo test which determines if an interceptor of this type is turned.

REL: probability used in the Monte Carlo test which determines if an interceptor of this type aborts.

RESERVE: the amount of fuel required after landing.

ACCEL Card

/ACCELb AC.TIME AC.FUEL AC.DIST Comment

The ACCEL card dictates the time (AC.TIME), the fuel (AC.FUEL), and the distance (AC.DIST) required to accelerate from STOP airspeed to attack airspeed.

CRUISE Card

/CRUISE C.TIME C.JP C.DIST Comment

The CRUISE card gives the time (C.TIME), fuel (C.JP), and distance (C.DIST) needed for an interceptor to takeoff and obtain the airspeed used on a cruise profile.

DASH Card

/DASHbb D.TIME D.JP D.DIST Comment

The Dash card gives the time (D.TIME), fuel (D.JP), and distance (D.DIST) required for an interceptor to takeoff and obtain the airspeed used on a dash profile.

The values from these three cards are used to identify the set of interceptors capable of completing a particular intercept. First, the "dash" profile for ground based interceptors or the "accel" profile for airborne interceptors is used to identify an initial set of interceptors. Then the "cruise" profile is used to identify any others interceptor that would require a lower airspeed in order to have enough fuel to complete the intercept.

FLOW Card

/FLOWbb C.FUEL D.FUEL R.FUEL L.FUEL MX.FUEL Comment

The FLOW card indicates the interceptor's fuel flows, in pounds per hour, for the cruise (C.FUEL), dash (D.FUEL), recovery (R.FUEL), loiter (L.FUEL), and maximum speed (MX.FUEL) profiles.

SPEED Card

/SPEEDb C.SPD D.SPD R.SPD L.SPD MX.SPD Comment

The SPEED card indicates the interceptor's cruise (C.SPD), dash (D.SPD), recovery (R.SPD), loiter (L.SPD), and maximum (MX.SPD) speeds. Since the loiter phase is associated with aircraft on STOP, L.SPD is never used in the program because an interceptor on STOP does not move. However, a value must be included on the card.

LOAD Card

/LOADbb NUM_1 $\mathrm{M.NAME}_1$ NUM_2 $\mathrm{M.NAME}_2$ ... NUM_n $\mathrm{M.NAME}_n$

The LOAD card indicates how many missiles (NUM_i) of the missile type (M.NAME) that are to be loaded on all interceptors in this block. This is the reason the ITYPE block must follow the MTYPE block.

PCNOSE Card

/PNOSEb PC_1 PC_2 ... PC_n

The PCNOSE card designates the probability of this interceptor type of obtaining a firing position for a front attack after detecting a particular raid class. Thus, PC_i is the probability of conversion on a raid of class (i), for i = 1 to n.

PCTAIL Card

/PCTAIL PC_1 PC_2 ... PC_n

Similar to a PCNOSE Card, the PCTAIL card designates the probability of conversion (PC_i) for a stern attack on a raid of class (i), for i = 1 to n.

PDIN Card

/PDINbb PD_1 PD_2 ... PD_n

The PDIN card contains the interceptor's probability of detecting (PD_i) a raid of class (i) that is in radar coverage. Probabilities are listed for all raid classes.

PDOUT Card

/PDOUTb PD_1 PD_2 ... PD_n

The PDOUT card indicates the probabilities of detecting the raid classes when the raid is not in radar coverage.

INTERN Card

/INTERN I.FUEL Comment

The INTERN card specifies the amount of internal fuel (I.FUEL) this type of interceptor carries. Only internal fuel is included when determining the interceptors that can complete a dash profile intercept. If no external fuel is carried, the card can be omitted since the default value is a full fuel load.

LAUNCH Card

/LAUNCH Number ENGAGE Comment

The LAUNCH card indicates the "number" of missiles that can be in the air simultaneously. This value is used in determining whether multiple launches can occur. ENGAGE is the maximum number of targets that one interceptor can simultaneously engage. If this card is omitted, then the default values of "number" and ENGAGE are two and one, respectively.

REATTK Card

/REATTK Time Comment

The REATTK card specifies the "time" (in minutes) required for an interceptor to complete the reattack phase. This card is optional since the default vaule is a reattack "time" of three minutes.

SALVO Card

/SALVOb Number Comment

The SALVO card indicates the "number" of missiles that can be fired on one shot. Two missiles per shot is the default value.

The fourth block specifies all the bases. Only one type of data card is needed to input a base.

BASE Card

/BASEbb NAME I.NAME LAT LON TURN CAP

The BASE card is used to identify each base by:

NAME: six character (or less) alpha-numeric designator of the base.

I.NAME: name of the interceptor type located on the base. (This is the reason why the block of bases must follow the block of interceptor types. If two or more interceptor types are desired at one base, as many BASE cards must be input as there are interceptor types. However, the only difference in the BASE cards would be a different I.NAME on each card.)

LAT: latitude of the base.

LON: longitude of the base.

TURN: the services that are available at the base: 1=fuel and armament; 2=fuel only; and 3=no services available.

CAP: indicates that the base is a STOP. (If any character is placed between TURN and column 73, the base becomes a STOP.)

The next block is used to put interceptors on bases, and it must follow the block of interceptor types and the block of bases.

PUT Card

/PUTbbb Number I.NAME ON BASE AT Time LESS Fuel

The PUT card is the only type of card in this block and is used to indicate the "number" of interceptors of type I.NAME that are placed on each BASE. "ON" is an alphanumeric word used to separate I.NAME and BASE. Then the "time" required to get the interceptors on alert status is read, and the "fuel" used to ready the interceptors is subtracted from the total fuel. AT and LESS are also alphanumeric words used to make the card more readable. "Time" and "fuel" have default values of zero so that the phrase "AT Time LESS Fuel" may be omitted. However, no other entry can be made on the card if the phrase is omitted. The "time" and "fuel" values are used primarily for putting aircraft on STOPS.

The sixth block has two input card types which are used to input information about the AWACS and their orbits. If a fighter-AWACS is used, this block must follow the block of interceptor types.

AWACS Card

/AWACSb ID.NUMB ALT MXRNG SIG9 RNG9 MAX.PAIRS MAX.TRACKS
ATYPE MISS.ON

The AWACS card is used to specify the operating characteristics of an AWACS and to signal that a set of "checkpointn cards will follow.

ID.NUMB: identification number of the AWACS--an integer number of five digits or less.

ALT: AWACS orbit altitude, in feet above mean sea level (MSL).

MXRNG: Maximum theoretical radar range of the AWACS.

SIG9: radar cross-section, in square feet, that can be detected at the range RNG9.

RNG9: radar range at which a target of size SIG9 can be detected.

MAX.PAIRS: maximum number of interceptor flights the AWACS can control.

MAX.TRACKS: maximum number of raid tracks that the AWACS can control.

If the AWACS is not a fighter-AWACS, no other information should be placed on the card. For fighter-AWACS, ATYPE must correspond to a name of an interceptor (I.NAME) previously input. MISS.ON is the total number of missiles loaded on the AWACS, and the missile type must be specified on a LOAD card for the interceptor type. Due to a program limitation, only one type of missile should be loaded.

"Checkpoint" cards do not have a key word and are formatted as follows:

```
/bX LAT LON SPD
```

The orbit of an AWACS is defined by a set of "checkpoint cards. Two types of checkpoint are used: navigational and orbit checkpoints. Navigational checkpoints are used to fly the AWACS to its orbit, thus serving as a method for delaying an AWACS entry into the simulation. Once established in an orbit, an AWACS can never leave it. Checkpoints are differentiated by letting the X in column 2 be a one for a navigational checkpoint, or a zero for an orbit checkpoint. LAT and LON are the latitude and longitude of the checkpoint, and SPD is the AWACS speed for the leg that starts at the checkpoint.

The ground radar cards specify the location and operating characteristics of each ground radar by using one input card per radar.

RADAR Card

/RADAR ID.NUMB LAT LON ALT MXRNG SIG9 RNG9

ID.NUMB is the ground radar's identification number, which should be a five digit (or less) integer number.

LAT: radar's latitude.

LON: radar's longitude.

ALT: radar's altitude in feet (msl).

MXRNG: maximum theoretical radar range.

SIG9: cross-section (in square feet) that can be detected at a range RNG9.

RNG9: maximum range at which a target with a cross-section of SIG9 can be detected.

The last block contains all of the raids. Seven input card types can be found in this block, and the order of these cards is significant.

RAIDS Card

/RAIDSb Comment

The RAIDS card indicates the beginning of the raid information.

"Sortie ID" cards do not have key words and must be the first card for each individual raid. The cards are formatted as follows:

/b ID.NUMB SIZE Time X.SEC Comment

ID.NUMB: identification number for the raid (an integer number with at most three digits).

SIZE: number of bombers in the raid.

"Time": start time of the raid.

X.SEC: radar cross-section of one bomber in the raid.

"Checkpoint" cards also do not have key words and are formatted as follows:

/Xbbb LAT LON SPD ALT CLASS Comment

"Checkpoint" cards define the penetration route of the raid. There is no limit on the number of "checkpoints" that may be used. "X" in column one indicates that this checkpoint will be used as the reference checkpoint in all output reports.

LAT: latitude of the checkpoint.

LON: longitude of the checkpoint.

SPD: airspeed for the leg starting at this checkpoint.

ALT: altitude for the leg starting at this checkpoint.

CLASS: raid class that best describes the raid.

LAU Card

/bLAU LAT LON SPD ALT CLASS Comment

Bombers carry either gravity bombs or ASMs or a combination of both. The LAU card indicates an ASM launch point. The other parameters on the card are identical to the parameters on a checkpoint card. The next card after a LAU card must be an ASM card.

ASM Card

/bASM LAT LON SPD ALT X.SEC CLASS Comment

The ASM card indicates an ASM impact point. An ASM is treated like any other raid since it can be detected and killed. The ASM card must immediately follow a LAU card and precede a DGZ card.

LAT: latitude of the impact point.

LON: longitude of the impact point.

SPD: ASM's speed from launch to impact.

ALT: ASM's altitude.

XSEC: ASM's cross-section in square feet.

CLASS: raid class that best describes the ASM.

DGZ Card

/bDGZ MEGATONLocation

The DGZ card indicates either a gravity bomb or an ASM impact point. If the card prior to a DGZ card is an ASM card, then an ASM will impact; otherwise a gravity bomb will impact. MEGATON is the megaton yield of the weapon, and "location" is an alpha-numeric name of the impact point. It can be up to 24 characters in length and starts in the column immediately following the last digit of MEGATON.

END Cards

/bEND LAT LON Comment
/ENDbbb Comment

Two types of END cards are used. The END card that has a blank in column one represents the end of a particular raid. LAT and LON is the latitude and longitude of the final point on the raid's penetration route. The END card that has END in columns 1 through 3 signals the end of all raid information. A final XNAMEX card then completes UNIT 12.

Attachment 1-A

Sample Unit 10 Data Cards

```
/  SOME TITLE FOR DATA BASE
/FILE   CONTRL
/FILE   MISSIL
/FILE   ACFT
/FILE   BASES
/FILE   AWACS
/FILE   RADARS
/FILE   PLACE
/FILE   RAIDS
/ALL.IN
```

Attachment 2-A

Sample Unit 12 Data Cards

```
/XNAMEX  CONTRL   CONTROL PARAMETERS FILE
/DELAYS  0.5  0.5  0.5  4.0
/MXDST   600
/OUTPUT  1 1 1 1 1 1 1 0 0 0 0 0
/OVRF    1.2
/PDAWC   0.0  0.5  1.0  1.0  *
/PDRAD   0.0  0.5  1.0  1.0  *
/PKLVL   0.01
/RANDOM  5  476
/XNAMEX  MISSIL   MISSILE PARAMETERS FOR ACFT-A
/MTYPE   WPNA
/PKNOSE  .35  .30  .05  .00
/PKTAIL  .30  .30  .05  .00
/MTYPE   WPNB
/PKNOSE  .00  .00  .10  .05
/PKTAIL  .50  .40  .10  .00
/XNAMEX  ACFT
/ITYPE   ACFT-A  30  14500  2  30  .80  .80  1200
/ACCEL   2.0  1000  20.0
/CRUISE  8.0  1800  85.0
/DASH    4.0  2400  25.0
/FLOW    60.  300.  50.  50.  300.
/SPEED   500  700  450  500  1100
/LOAD    1 WPNA    1 WPNB
/PDIN    .85  .70  .80  .75
/PDOUT   .40  .30  .20  .15
/PCNOSE  .75  .30  .60  .20
/PCTAIL  .90  .75  .20  .00
/INTERN  12000
/LAUNCH  4  2
/REATTK  2.5
/SALVO   4
/XNAMEX  BASES  ACFT-A BASES
/BASE    NAME-A  ACFT-A  46.57  67.53  1
/BASE    NAME-B  ACFT-A  41.39  70.31  2
/BASE    STOP-1  ACFT-1  40.50  70.30  3  STOP
/XNAMEX  PLACE
/PUT     14  ACFT-A ON NAME-A
/PUT      4  ACFT-A ON NAME-B
/PUT      2  ACFT-A ON STOP-1 AT 45 LESS 2000
/XNAMEX  AWACS
/ASTART  RANDOM
/AWACS   1  29000  350  25  250  50  25
/ 1      42.00  73.00  330
/ 1      42.15  73.10  330
/ 0      41.10  72.10  330
/ 0      41.00  72.25  330
/AWACS   2  29000  350  25  250  50  25
/ 0      43.15  75.50  330
/ 0      43.15  74.05  330
/XNAMEX  RADARS
/RADAR   1000  44.38  67.24  340  240  25  240  SOMEWHERE-A
/RADAR   2000  43.27  65.28  100  240  25  240  SOMEWHERE-B
/XNAMEX  RAIDS
/RAIDS
/      133   1   741.0   25
/      40.19   67.15   330   31000   1
/      40.16   69.49   340   31000   1
/ LAU  40.48   71.14   345   31000   1
/ ASM  40.48   71.00   1000  60000  10  4
/ DGZ   1.0   TARGET-A
/ END   40.48   71.15   345   31000   1
/      134   1   854.0   25
/      40.43   67.17   300   31000   1
/      40.09   69.16   340   31000   1
/      40.15   70.21   410   31000   1
/      40.48   70.31   450   31000   1
/ DGZ   1.0   TARGET-B
/ END   40.48   70.32   450   31000   1
/END
/XNAMEX
```

Appendix B

Additional Reference Material

Validation and Verification References

Arnett, James M. "Toward Validation of Computer Simulation Models in Operational Test and Evaluation." Unpublished MS thesis. School of Engineering, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, December 1979.

Gilmour, Peter. "A General Validation Procedure for Computer Simulation Models," The Australian Computer Journal, 5(3):127-131 (November 1973).

Law, Averill M. "Validation of Simulation Models, I: An Overview and Survey of Real-World Practice," Technical Report No. 78-14, Department of Industrial Engineering, University of Wisconsin (August 1979).

Mihram, G. Arthur. "Some Practical Aspects of Verification and Validation of Simulation Models," Operational Research Quarterly, 23(1):17-29 (March 1972).

Miller, D.R. "Model Validation Through Sensitivity Analysis," Proceedings of the 1974 Summer Computer Simulation Conference, 911-914. LaJolla, Ca: The Simulation Councils, Inc., 1974.

U.S. General Accounting Office, "Guidelines for Model Evaluation," Report PAD-79-17, Washington (1979).

Winter, E. M. et al. "Verification and Validation of Engineering Simulations with Minimal Data," Proceeding of the 1976 Summer Computer Simulation Conference, 718-723. LaJolla, Ca: The Simulation Councils, Inc., 1976.

Statistical References

Box, G.E.P. and J. S. Hunter. "The 2^{k-p} Fractional Factorial Designs, Part I," Technometrics,3(3):311-351 (August 1961).

-----. "The 2^{k-p} Fractional Factorial Designs, Part II," Technometrics, 3(4):449-458 (November 1961).

Garratt, Michael. "Statistical Validation of Simulation Model," Proceedings of the 1974 Summer Computer Simulation Conference, 915-927. LaJolla, Ca: The Simulation Councils Inc., 1974.

Kleijnen, Jack P. C. Statistical Techniques in Simulation, II. New York: Marcel Dekker, Inc., 1975.

McNichols, Charles W. An Introduction To: Applied Multivariate Analysis--Course Notes. Wright-Patterson AFB, Ohio: Air Force Institute of Technology, 1979.

Naylor, Thomas H., et al. The Design of Computer Simulation Experiments, edited by Thomas H. Naylor. Durham, N.C.: Duke University Press, 1969.

Nie, Norman H., et al. SPSS Statistical Package for the Social Sciences Second Edition. New York, NY: McGraw-Hill, Inc., 1975.

Siegel, Sidney. Nonparametric Statistics for the Behavioral Sciences. New York, N.Y.: McGraw-Hill, Inc., 1956.

VITA

Craig S. Ghelber was born on 19 January 1950 in Denver, Colorado. He graduated from high school in Cheyenne, Wyoming in 1968 and attended the United States Air Force Academy, from which he received the degree of Bachelor of Science in Mathematics in June 1972. Upon graduation he entered Undergraduate Pilot Training at Laredo AFB, Texas, and received his wings in May 1973. He served as a T-39 pilot with the 56th Combat Support Group, Nakhon Phanom RTAFB, Thailand until being assigned as a T-33 instructor pilot with the 46th Flying Training Squadron, Peterson AFB, Colorado in June 1974. In October 1975 he was assigned as the aide to the commander of the 21st NORAD/AD, Hancock Field, New York. He then served as a F-106 pilot with the 84th Fighter Interceptor Squadron, Castle AFB, California until entering the School of Engineering, Air Force Institute of Technology in June 1979.

Permanent address: 4701 East 14th Street
Cheyenne, Wyoming 82001

VITA

Charles A. Haley was born on 21 June 1951 in Ponca City, Oklahoma. After graduating from a Ponca City high school in 1969, he attended Oklahoma State University and received a Bachelor of Science Degree in Mathematics and an Air Force Commission through ROTC in 1973. After graduation, he was assigned to Undergraduate Pilot Training at Williams AFB, Arizona. After receiving his wings in June 1974, he served as a T-33 and F-106 pilot with the 84th Fighter Interceptor Squadron at Castle AFB, California until entering the School of Engineering, Air Force Institute of Technology in June 1979.

Permanent address: 1020 South Sixth Street
Ponca City, Oklahoma 74601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GOR/MA/80D-3	2. GOVT ACCESSION NO. AD-A094 814	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A METHODOLOGY FOR VALIDATION OF COMPLEX MULTI-VARIABLE MILITARY COMPUTERIZED MODELS		5. TYPE OF REPORT & PERIOD COVERED MS Thesis
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Craig S. Ghelber, Captain, USAF Charles A. Haley, Captain, USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB, OH 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright Patterson AFB, OH 45433		12. REPORT DATE December 1980
		13. NUMBER OF PAGES 122
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)

Approved for public release; distribution unlimited

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

18. SUPPLEMENTARY NOTES

Approved for public release IAW AFR 190-17

Frederic Lynch, Major, USAF
Director of Information

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

Computer Simulation
Validation
Verification
War Game Models
Modeling
Experimental Design in Validation of Simulation Models

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

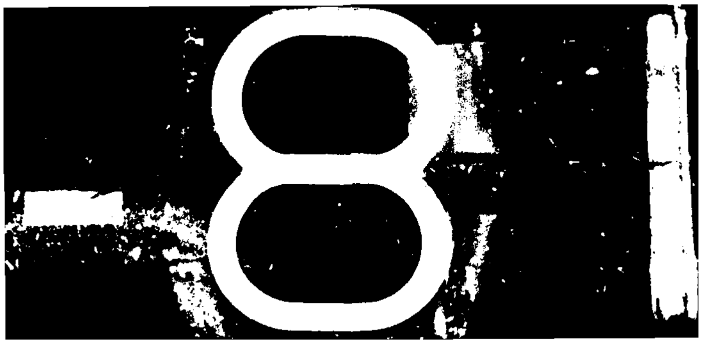
Computerized simulation models that are characterized by multi-variable and minimal or nonexistent real world supporting data are often used without being properly validated. The towards-validation methodology is introduced as a four-phase approach for validating these complex models and is defined as: The documented evidence that a computerized model can provide users verifiable insight, within the model's domain of application, for the purpose of formulating analytical or decision-making inferences.

(Continued on Reverse)

BLOCK 20

Towards-validation begins with the conceptual phase of model development. Next, the verification phase examines the mechanical validity of model design. The third phase, credibility, is concerned with both intuitive and statistical appeal. The final phase deals with confidence building and documentation.

To illustrate the application of towards-validation, the Headquarters Air Force version of the Interceptor War Game Model is examined. Results are documented in a user's guide.


8