1988

A DECISION SUPPORT SYSTEM FOR THE DESIGN OPTIMIZATION OF MULTI-PHASE TRAINER INSTRUCTOR/OPERATOR STATION IN FLIGHT SIMULATORS

Cheickna Sylla
Drexel University

Ramaswamy Ramesh
State University of New York at Buffalo

A DECISION SUPPORT SYSTEM FOR THE DESIGN OPTIMIZATION OF MULTI-PHASE TRAINER INSTRUCTOR/OPERATOR STATION IN FLIGHT SIMULATORS

Cheickna Sylla
Engineering Management
Drexel University

Ramaswamy Ramesh
Management Science and Systems
State University of New York at Buffalo

ABSTRACT

A significant problem encountered in designing flight simulator device training programs is the of lack a decision support system (DSS) providing a unified methodology for establishing appropriate training criteria and optimal training device parameters. This paper presents an efficient DSS for cost-effectiveness analysis for the optimal design of the Instructor Operator Station (IOS) in flight simulators. This IOS can be used to teach a multiple of training phases. This DSS integrates three relatively distinct areas: (1) learning curve modeling, (2) economic analysis, and (3) multi-criteria decision making for the design tradeoffs optimization. The paper presents the methods for deriving the IOS design configurations from an initial broadly defined set of training objectives and the related training taxonomy of training devices from the training expert's opinions. The data requirement for making the design tradeoff decisions and the methods and sources of that data are also proposed in this paper. This DSS is designed to assist engineers and training specialists in their decision making for simulators and other training device design and development projects.

1. INTRODUCTION

The U.S. Army Research Institute (ARI) and its Program Manager for Training Devices (PM TRADE) are developing a large scale computerized system to enable tradeoff studies leading to the design of cost-effective simulators. An important component of this system is the optimization of Simulation-Based Training Systems (OSBATS). In its present, form the development of OSBATS includes five computer subsystems. The first subsystem will determine if simulation should be used to accomplish a given training task. If simulation is required, the second subsystem determines which instructional features should be included in the design configuration; the third subsystem determines what level of fidelity is required; the fourth subsystem determines how to allocate training time to various types of training devices; and finally, the fifth subsystem determines which type of training device configuration for the Instructor/Operators' Station (IOS) is most appropriate to use in multiple training phases.

The principal focus of this paper is concerned with the development of a DSS for the design optimization of the IOS. In addition, the paper presents some of the industrial engineering and information system management issues related to the conceptual framework of the IOS design. It also presents a large scale mixed integer programming formulation of the design problem and briefly elaborates on the data collection procedures being developed for the highly subjective training components of the model. All of the tradeoff components of the optimization design model are being implemented in the IBM PC/AT environment.

The organization of this paper is as follows. We present an overview of the organization and structure of the training program development at ARI in Section 2. We develop a conceptual framework for the design of the IOS in Section 3. The criteria for the IOS design are discussed in Section 4; the design optimization problem of a multiphase IOS is presented in Section 5. We develop the solution methodology and framework for the DSS in Section 6 and present the instruments for data collection in Section 7. The conclusions are presented in Section 8.

2. AN OVERVIEW OF OSBATS

Figure 1 presents a conceptual framework of the OSBATS system. The system supports the dynamic process of the training program development at ARI. Training programs are developed from an initial identification of training goals. These goals are derived from the evolving new technology, new training doctrines, new equipment, new policies and new development in the existing training systems. These goals are refined into training requirements, which are subject to detailed analyses to find the appropriate behavioral skills and knowledge necessary to perform the job tasks in the training process. The training strategy is then developed from these requirements, which stipulate the method and direction of application of the training systems. The training concept is thus formulated, and the training systems are developed within its framework (see Andrews et. al. 1987). The OSBATS decision support facility supports each phase of this developmental task.

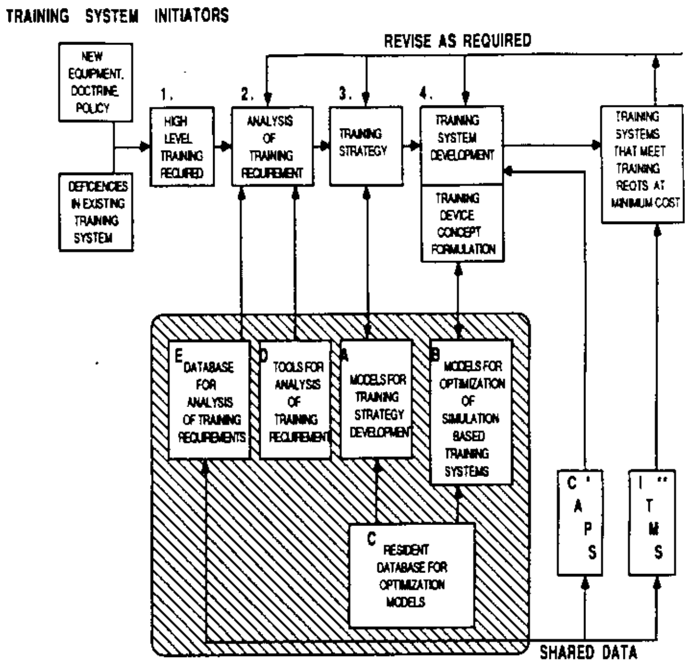

Figure 1: OSBATS Conceptual Framework

The OSBATS system consists of five separate, but interdependent, subsystems or modules tailored to each of the tasks discussed above. These subsystems, denoted as the modules A,B,C,D, and E in Figure 1, provide the tools, data and an operational framework for the developmental process, and are briefly described below.

Module A

This module consists of models for the training strategy development, and provides an analytical framework for producing a training strategy that derives the Training Subsystem training plans.

Module B

This module consists of models for the optimization of the simulation-based training subsystems, and provides an automated decision aid for training device design. The proposed optimization model for configuring the design of the IOS is a component of this module.

Module C

This module consists of the databases necessary to support the optimization of training subsystems and training devices and is designed to provide the internal, and the intermediate fundamental information needed by the training device development models.

Module D

This module consists of the tools for the analysis of training requirements, which examine the input requirements of the optimization models, and select and develop techniques for defining the training requirements necessary to initiate the models.

Module E

This module consists of the databases for the analyses of missions, functions and tasks of the training process, and provides a starting point based upon existing data for the analyst to define training requirements.

There are two groups of potential users of these DSS optimization tools. The first group of potential users are individuals concerned with the design and development of Training Subsystems. They develop training strategies and training subsystem alternatives which lead to effective training plans. Specifically, these users are the training decision makers and developers within the U.S. Army. They are school commanders, unit commanders, personnel at the Army's Training and Doctrine Command, and training developers at the formal schools. Generally, these individuals are key training decision makers and training developers in the military organizations, including NATO.

The second group of potential users are individuals who have the specific task of developing training devices for Training Subsystems. These individuals are training device designers and engineers. They work for PM TRADE and with contractors who build training devices. This group includes the individuals who will use the IOS design tools developed in this research.

The optimization models are run on desk-top personal computers. The goal is to make the systems as inexpensive as possible to accommodate the largest number of users. PM TRADE is currently examining computer systems which will have widespread use throughout the Army and its goal is to "piggy-back" on those systems wherever possible. Use of the computer-based optimization tools will result in decision audit trails for both groups of users.

3. CONCEPTUAL FRAMEWORK FOR IOS DESIGN

The overall architecture is presented as a set of computer-aided design tools to be used to design an Instructor/Operator Station (IOS). These design tools are used to select the instructional features which should be made available to the instructor on the training device to control and perfect the student training performance criteria. Emphasis in this section is placed on defining the conceptual framework used in this design. The framework is designed using taxonomies of training terms concerned with the phases of training, the instructor functions, the types of instructors for the IOS, the teaching strategies, the location of instructor consoles on the IOS, and the many types of features added to the IOS to support the simulator device operation, instruction and management. In addition to the conceptual framework, this section briefly describes the contents of a series of matrices defining the relationship among the various terms and concepts.

In this conceptual framework, a simulation-based training system consists of five major inter-related subsystems, all of which are included in the optimization model of the IOS. These are:

a. student station (crew station mock-up),

b. simulation subsystem (software, environmental simulations),

c. instructional subsystem (instructional features and training exercises/scenarios),

d. instructor/operator station (operating consoles), and

e. utility subsystem (e.g., hydraulics, pneumatics).

The relationship among these subsystems is presented in Figure 2. There are three ways in which the IOS affects the student station. First, there is a direct contact between the instructor and student. Second, the instructor can affect the student through the selection of training exercises (Instructional Subsystem). Third, the instructor can affect the student through environmental simulation (Simulation Subsystem). The Utility Subsystem may also impact the student, but it can only affect the student indirectly through the Simulation Subsystem, in interface with the IOS.

Due to its direct impact on the Student Station, the IOS is a primary determiner of the effectiveness of the training device as a training medium. The uses made of the Instructional Subsystem and the Simulation Subsystem by the instructor determine the overall effectiveness of the device in terms of transfer of training to the operational system, training costs, training times, and other important criteria.

Extensive investigations using expert instructional technologists and human factors specialists have been used to define the required parameters for use in designing the IOS. They defined six training system parameters used in determining what type of instructor support should be provided with a training device. These parameters include:

a. **Training phases**: the stage or level of training being conducted which must be supported by the trainer (i.e., the IOS).

b. **Training functions**: the instructional functions which must be implemented to support training.

c. **Trainer Manning**: the characteristics of the personnel who will man the instructor/operator station.

d. **Training strategy**: the type of training method which will be used.

e. **Instructor location**: the position of the instructor in the training complex when performing the training.

f. **Trainer Features**: the trainer features available to the operator/instructor to enhance or implement instruction.

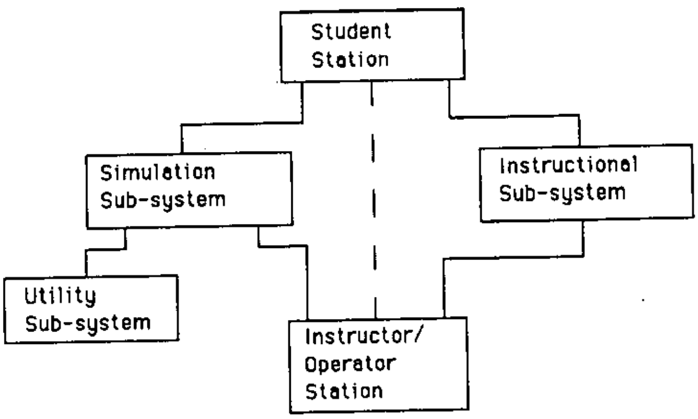

Figure 2: Block Diagram of the Simulation Training Subsystem

These six training system parameters have been arranged as a taxonomy of the components which describe the conceptual framework for the instructor considerations model routines. Table 1 outlines these parameters and the related terms describing the conceptual framework. Twenty-one matrices representing the governing variables or parameter and their interactions have been developed from expert judgments. Each of these matrices is two-dimensional and represents the interactions between two parameters. Each matrix is identified by a column parameter and a row parameter, the rows are identified by the levels of the row parameter, and the columns are identified by the levels of the column parameter. Each cell in the

Table 1: Instructional System Taxonomy

TRAINING PHASES

- Familiarization Training
- Part Task Training
  - Rules Training
  - Decision-Making Training
  - Detection Training
  - Classification Training
  - Voice Procedures Training
  - Procedures Training
  - Steering/Guiding Training
- Position Training
- Crew Training
- Mission Training
- Proficiency Training
- Advancement Training
- Special Training

TRAINING FUNCTIONS

- Preparing Function
- Briefing Function
- Initializing Function
- Training Function
- Evaluating Function
- Debriefing Function
- Documenting Function
- Developing Training Events Function
- Training Instructors Function

TRAINER MANNING

- Operational Instructor
- Simulator Instructor
- Operator/Instructor
- Technician/Operator
- Student Instructor
- Automated Instructor
- Peer Instructor

TRAINING STRATEGY

- Tutor Level
- Interactive Level
- Monitor Level

INSTRUCTOR LOCATION

- Remote IOS
- On-Board Remote IOS
- Over-the-Shoulder IOS

TRAINER FEATURES

- Operating Features
- Instructing Features
- Management Features

matrix is a 0-1 entry, where a 0 represents incompatibility between the levels of the row parameter and the column parameter the cell represents and a 1 represents compatibility. Table 2 summarizes the various sets of matrices developed, taking into account the primary compatibility between the parameters. For example, the Strategy parameter is insensitive to the Function parameter and, therefore, a matrix involving the compatibility between the levels of the Strategy and the Function parameters is not necessary. On the other hand, the Location parameter is sensitive to the Phase parameter (this is indicated by an X in Table 2) and, accordingly, a compatibility matrix between the levels of these two parameters is necessary. The compatibility between the levels of two parameters represents their interaction or appropriateness for each other. In any given design situation, the set of feasible designs is identified as follows.

4. CRITERIA FOR IOS DESIGN

In general, one or more IOS may be designed for each phase of the simulator training phase. It is also possible (and often required) to design an IOS which can be used for multiple training phases. For a given simulator training phase, the triad of parameters (MANNING, STRATEGY, LOCATION) is called an IOS CONFIGURATION. Given a configuration, the set of operational, instructional and management features is called the FEATURES SET. The set of functions performed in training for a phase is called the FUNCTIONS SET. The following algorithm identifies the set of feasible configurations for a given phase and their corresponding sets of feasible features.

Table 2: Parameter Matrices Developed

	Phase	Function	Manning	Strategy	Location
Phase	-	-	-	-	-
Function	x	-	-	-	-
Manning	x	x	-	-	-
Strategy	x	NA	x	-	-
Location	x	x	x	x	-
Operate	x	x	x	x	x
Instruct	x	x	x	x	x
Manage	NA	x	x	NA	NA

NA - not applicable - matrix insensitive
x - matrix developed

Algorithm: CONFIG

Input: A given PHASE and the set of twenty-one compatibility matrices.

Output: Set of feasible CONFIGURATIONS and their SETS OF FEATURES.

Step 1 Identify the sets of Mannings, Strategies and Locations appropriate for the Phase from the matrices (PHASE,MANNING), (PHASE, STRATEGY), (PHASE,LOCATION), respectively.

Step 2 Identify from the above sets of Mannings, Strategies and Locations the triads (MANNING, STRATEGY, LOCATION) such that the triad parameters are compatible with each other using the matrices mentioned in Step 1.

Step 3 Consider the FUNCTIONS SET for the phase. Test each triad whether the manning, strategy and location can support each function in the FUNCTION SET using the (FUNCTION, MANNING), (FUNCTION, STRATEGY) and (FUNCTION,LOCATION) matrices respectively. The set of triads that can support all these functions is the set of feasible IOS CONFIGURATIONS.

Step 4 Identify the FEATURES SET for each configuration as follows: Identify the features that are compatible with the configuration using the (FEATURES, MANNING), (FEATURES, STRATEGY) and (FEATURES, LOCATION) matrices. For each feature identified thus, test if it supports any of the functions in the FEATURES SET. Include a feature in the FEATURES SET only if it supports at least one function in the FUNCTIONS SET.

The above algorithm shows that a phase could have several feasible configurations with their associated sets of features. Therefore, the optimization problem in the design of the IOS can be stated as:

> Given a set of phases for which a training device is needed (i.e., a multiple phase IOS), we must determine and select the optimal set of configuration(s) and associated set of features, from among the feasible sets of configurations and associated features, to be included in the IOS such that:
>
> i. The overall cost of the IOS is minimized, *and*
> ii. The overall training effectiveness is maximized.

The above design problem is a multi-criteria optimization problem. Furthermore, several criteria of effectiveness such as transfer of training, safety, utilization, and job readiness can be simultaneously used. This problem could have several efficient solutions. We model this problem as a mixed-integer programming problem and develop interactive solution strategies in the following discussion.

In the process of developing a training program, an obvious question relates to which tasks (behaviors) should be trained and which should not. In the IOS design context, these tasks are being supported by the various IOS configurations and the related operating, instructing and management features based on the phase of training. Therefore, the above question is translated into the optimal selection of the IOS configuration and design features which provide the best training effectiveness. To date, there has been no algorithm established to assist in this decision.

A second decision, that has been particularly prominent in military training and is gaining visibility in industry, is related to the complexity (cost) of the training devices (or programs). As with the relation between learning rate and time, there are diminishing returns in learning rate as the complexity of the training device surpasses a particular level (Roscoe 1971). Therefore the overall decision problem, consists of arriving at the most cost-effective IOS design.

In general, the effectiveness of a simulator training device, such as the IOS, is defined to include the following five main factors according to criticality and importance:

1. **Transfer of training** (percentage of the time saved on the real equipment learning time).

2. **Safety** (accident prevention and/or reduction).

3. **Utilization** (the frequency of use of the device).

4. **Job readiness** (how close to the real system).

The IOS device effectiveness is a function of the above five attributes. In any particular design situation, the above five attributes might be in conflict among the decision alternatives that we have. Therefore the decision-maker's preferences for tradeoffs among the attributes should be evaluated in order to arrive at a workable solution. This reduces the problem to either evaluating a multi-attribute utility function for the decision maker or using an interactive approach where the decision-maker's preference structure is progressively assessed. This is a complex task, requiring extensive data collection from the experts and considerable cognitive effort on the part of the decision maker in using the system. Therefore, we simplify the design criteria by using only the Transfer of Training attribute (also known as the transfer of training ratio, TER) in our analysis.

The concept of TER is based on the work by Roscoe (1971). It is a simplifying concept based on the concept of the learning curves. The TER concept was used in earlier studies on flight simulator design investigations (see Bickley 1980; Bateman and Hottman 1983).

The measurement of the TER for any task of a given phase of training using any given IOS configuration involves the estimation of three learning curve parameters (say a, b, and c). These parameters are:

> **a** represents the total amount of training time on the actual equipment that can be substituted (partially or completely) using the training device.
>
> **b** represents a measure of the rate of substitution of the actual equipment training time with the training device training time.
>
> **c** represents the minimum time required for training on the actual equipment to achieve the desired level of proficiency regardless of the amount of time spent on the training device.

Related to TER, Bickley (1980) used experimental data on simulator training transfer effectiveness studies and found an experimental function. For modeling purposes it is expressed as

$$Y = a e^{-bx} + c$$

where x represents the normal time of the training on the real equipment and Y represents the corresponding time if the training is done on the simulator. The application of this equation is generally accepted in the simulator training literature. In general, the equation depicts a training system with high initial learning rates which are reduced as the system approaches its limit of usefulness. This equation shows that, once a simulator has been used to provide essential training, further increases in the simulator use will not provide additional benefits. The exact points of optimum benefit (diminishing returns) and negative utility (costs exceed benefits) depend upon the relative costs of using the simulator and operational system.

The cost of a IOS is known to include the following generally acceptable factors (based on life cycle cost considerations): acquisition, salvage, operating and maintenance costs. The first two costs are fixed, while the last two are variable. The fixed costs elements can be estimated from expert judgments for all of the features. The estimation of the variable costs for each feature may be quite difficult for the experts. We present a data collection methodology for these cost elements in Section 7.

5. THE TRADEOFF DECISION PROBLEM

Given a Training Plan comprising of a set of phases, the objective is to determine the best IOS design configuration(s) such that:

(1) total cost of the IOS is minimized,

(2) total time required for training on the actual system is minimized,

(3) safety of the IOS is maximized,

(4) utilization of the IOS is maximized, and

(5) job readiness is maximized.

Except for the first term, which deals with the cost, all of the remaining terms deal with the effectiveness of the IOS. This is a multiple objective problem. There may *not* be a single design that would achieve all of these objectives. Therefore, we must evaluate different tradeoffs among these objectives that are acceptable to a decision maker and arrive at the *best* compromise design. In addition, the design model is very large.

Given the above five criteria, the determination of efficient solution may be a difficult task. Therefore we restrict our consideration to only the following two important criteria:

i. Minimization of Cost
ii. Maximization of TER

Furthermore, we transform the above bicriteria problem into a single objective problem by treating the TER maximization objective as a problem constraint. In this way, the problem is reduced to a fairly manageable size. Using sensitivity analysis on the TER constraint, different solutions can be generated for consideration by a decision maker interactively. We develop two models of the design problem using the above strategy in the following discussion. It is assumed that the resulting IOS design configuration can be used to teach several phases of any training program. To begin with, we present the notation and terminology used in the model below.

NOTATION:

i	$= 1, \ldots, I$	denotes phases ($I = 14$)
j	$= 1, \ldots, J_i$	denotes configurations in phase i, $i = 1, \ldots, J_i$
k	$= 1, \ldots, K$	denotes operating features ($K = 19$)
l	$= 1, \ldots, L$	denotes instructional features ($L = 16$)
m	$= 1, \ldots, M$	denotes management features ($M = 8$)
t	$= 1, \ldots, T$	denotes Functions ($T=28$)

U_{ijk} = 1 if operating feature k is used in configuration j of phase i and 0 otherwise.

V_{ij} = 1 if instructing feature l is used in configuration j of phase i and 0 otherwise.

W_{ijm} = 1 if management feature m is used in configuration j of phase i and 0 otherwise.

F_i = denotes the set of functions on which training is conducted in phase i

G_i = denotes the set of configurations possible for phase i

a_{ijt}, b_{ijt}, c_{ijt}: three dimensional matrices of learning curve parameters (see discussion on transfer of training functions in the previous section) for training on function t of phase i using IOS configuration j.

P_k: fixed costs of operating features k
Q_l: fixed costs of instructional features l
R_m: fixed costs of management features m
P_k: variable costs of operating features k
q_l: variable costs of instructional features l
r_m: variable costs of management features m
α_{ij}: fixed manpower cost of configuration j in phase i
β_{ij}: variable manpower cost of configuration j in phase i
S: set of phases to be taught in the training plan.

We describe this optimization model (called MODEL 1) starting with its decision variables.

MODEL 1

DECISION VARIABLES:

Y_{ij} = 1 if configuration j is used for phase i
0 otherwise.

Z_{1k} = 1 if operating feature k is used
0 otherwise.

Z_{2l} = 1 if instructional feature l is used
0 otherwise.

Z_{3m} = 1 if management feature m is used
0 otherwise.

X_{ijt}: denotes the time spent in training function t of phase i using configuration j.

The formulation of the decision problem is:

Minimize

$$[\sum_{k=1}^{K} P_k Z_{1k} + \sum_{l=1}^{L} Q_l Z_{2l} + \sum_{m=1}^{M} R_m Z_{3m}] +$$

$$[\{\sum_i \sum_{j=1}^{J_i} \sum_t X_{ijt}\} \{\sum_{k=1}^{K} p_k Z_{1k} + \sum_{l=1}^{L} p_l Z_{2l} + \sum_{m=1}^{M} p_m Z_{3m}\}] +$$

$$[\sum_i \sum_{j=1}^{J_i} \mathring{a}_{ij} Y_{ij}] + [\{\sum_i \sum_{j=1}^{J_i} \sum X_{ijt}\} \{\sum_i \sum_{j=1}^{J_i} \beta_{ij} Y_{ij}\}]$$

Subject to:

$$\sum_i \sum_{j=1}^{J_i} u_{ijk} Z_{1k} \geq \sum_i \sum_{j=1}^{J_i} u_{ijk} Y_{ij}, \quad k = 1,\ldots,K \qquad (1)$$

$$\sum_i \sum_{j=1}^{J_i} v_{ijl} Z_{2l} \geq \sum_i \sum_{j=1}^{J_i} v_{ijl} Y_{ij}, \quad l = 1,\ldots,L \qquad (2)$$

$$\sum_i \sum_{j=1}^{J_i} w_{ijm} Z_{3m} \geq \sum_i \sum_{j=1}^{J_i} w_{ijm} Y_{ij}, \quad m = 1,\ldots,M \qquad (3)$$

$$\sum_{j=1}^{J_i} Y_{ij} = 1, \qquad (4)$$

$$\sum_{j=1}^{J_i} \{\sum_t \{a_{ijt} e^{-b_{ijt} X_{ijt}} + c_{ijt}\} Y_{ij}\} \leq B_i, \qquad (5)$$

$X_{ijt} \geq 0$

$Y_{ij}, Z_{1k}, Z_{2l}, Z_{3m} = 0$ or 1

where i belongs to S and t belongs to F_i.

ANALYSIS:

In the above model, the four terms in the objective function represent the total fixed costs of features, variable costs of features, fixed manpower costs and the variable manpower costs, respectively. Constraint set (1) enforces that an operating feature should be included in the overall IOS if it is needed for any of the configurations chosen. Constraint sets (2) and (3) enforce similar restrictions with respect to the instructional and management features, respectively. Constraint set (4) enforces that only one of the configurations for each phase of training to conduct be included in the IOS. Constraint set (5) stipulates the minimum Transfer Effectiveness Ratio acceptable or the maximum permissible training time for each phase of training conducted in the simulator.

The constraint sets (1), (2), and (3) together force the selection of all the features that are the union of all the chosen configurations for the phases. For example, if Y_{ij} = 1, then the Z-variables corresponding to all the features in configuration j must be equal to 1. Similarly, if Y_{ij} = 0, then the corresponding Z-variables must all be zero, unless any of these features are required for another chosen configuration of a different phase. This will be automatically enforced by the objective function, since it is to be minimized. Constraint set (5) restricts the total actual system training time for each phase. For example, if Y_{ij} = 1, then the X_{ijt}s can be nonzero, and this value is restricted because of the objective function. This can be seen from the fact that constraint (5) will cause the X_{ijt}s to be as high as possible, while the objective function will force them to values as low as possible, leading to intermediate values in the final solution. On the other hand, if Y_{ij} = 0, then automatically all the X_{ijt}s corresponding to this will be forced to zero because of the objective function.

Model 1 is a nonlinear integer programming problem of sufficiently large size. It might be very difficult to solve (especially on a microcomputer). Therefore, we simplify this model by using certain restrictions and linearization of the nonlinear constraints. This is shown in the following model.

MODEL 2

BASIC ASSUMPTIONS:

(1) The variable costs are directly assessed in terms of configurations (unlike previous model).

(2) The designer is asked to specify, at the design stage,

how much time he intends to see being spent in teaching each function of each phase. These intended times are indicated as œ_{it}. These values are treated as fixed parameters. Therefore, this model is also deterministic.

ADDITIONAL NOTATION:

∂_{ij} denotes the variable equipment cost of configuration j in phase i.

œ_{it} denotes the intended training time in function t in phase i.

The Model 2 formulation is:

Minimize

$$\text{(a)} \quad \sum_{j=1}^{J_i} \{ \sum_{t} \{a_{ijt}\, e^{-b_{ijt}\text{œ}_{ijt}} + c_{ijt} \}\, Y_{ij} \}$$

$$\text{(b)} \quad [\sum_{k=1}^{K} P_k Z_{1k} + \sum_{l=1}^{L} Q_l Z_{2l} + \sum_{m=1}^{M} R_m Z_{3m}] +$$

$$[\sum_{i} \sum_{j=1}^{J_i} \mathring{a}_{ij} Y_{ij}] + [\{ \sum_{i} \sum \text{œ}_{it} \} \{ \sum_{i} \sum_{j=1}^{J_j} (\beta_{ij} + \text{œ}_{ij})\, Y_{ij} \}]$$

Subject to:

Constraints (1), (2), (3) and (4) in Model 1, and

$Y_{ij}, Z_{1k}, Z_{2l}, Z_{3m} = 0$ or 1.

Model 2 is a simple (0-1) integer linear programming problem. In the current application of the smallest size, we have 43 Z variables, 121 Y variables, and, hence, 164 total (0-1) variables and 44 total constraints.

Therefore, it is much easier to solve a general 0-1 problem with 164 variables and 44 constraints under *reasonable conditions* than it is for the much larger nonlinear integer problem proposed in Model 1. We propose a solution methodology for this model in the following discussion.

6. SOLUTION METHODOLOGY

A number of methods have been suggested for the solution of multi-objective discrete alternative problems. Many of these methods construct a composite function to approximate an underlying utility function. Some of these methods can be classified as methods of conjoint analysis (Green and Srinivasan 1978), utility assessment procedures (Keeney and Raiffa 1976), and other interactive procedures (Korhonen, Wallenius and Zionts 1984; Koksalen, Karwan and Zionts 1984). There are also other methods that do not employ composite functions. Rivett (1977) uses a multi-dimensional scaling technique to rank order the solutions. The outranking relation methods suggested by Roy (1971) and Siskos (1982) also do not require the use of any composite functions.

In this research, we develop an interactive solution methodology that considers an implicit utility function of the decision maker and assesses it through pair-wise judgments. The preference structure is assessed and determined iteratively, and the algorithm converges to a locally optimal most preferred solution. In this approach, the preference structure assessment and the search for the most preferred solution are carried out side by side. To begin with, we present the problem structure with reference to Model 2 and then propose a solution procedure in the following discussion.

If a training plan consists of p phases with $n_1, n_2, ..., n_p$ configurations for each phase, then we have $[n_1 \times n_2 \times ... x n_p]$ configurations for the entire training plan. This constitutes a feasible solution space to the optimization problem and its cardinal number is denoted as

$$\emptyset = \prod_{i=1}^{p} n_i.$$

If $\emptyset$ is reasonable (which it will be in some cases), we can evaluate all of the combinations and rank them according to cost. Then, the œ_{it}s can be obtained from the decision maker and compute the training times for each overall configuration. As a simplification, we consolidate the training times by adding them up. This provides an evaluation of each overall IOS configuration in terms of the two objectives: cost and training time. This problem can then be solved interactively by either utility assessment methods (Farquhar 1984) or any discrete alternative programming method (Korhonen, Wallenius and Zionts 1984; Koksalen, Karwan and Zionts 1984, 1986). On the other hand if $\emptyset$ is very large, then the above approach is infeasible, given the combinatorial nature of the problem. To avoid this difficulty, we propose the following alternative interactive algorithm which proceeds through a systematic assessment of the decision maker's preference structure using pair-wise assessments of the decision alternatives by the decision maker. We hope the algorithm will generate a good local optimal solution since the problem is a nonlinear programming problem. The steps of the algorithm are as follow:

STEP 0 Ask decision maker to specify œ_{ij} for all functions of all the phases in the training plan.

STEP 1 Solve Model 2. Let $T_1^1, ..., T_p^1$ denote the configurations for the phases $i = 1, ..., p$ in its optimal solution. Therefore, $\{T_1^1, ..., T_p^1\}$ is the cheapest way we can build the IOS. Let μ^1 be its total actual training time and Ω^1 its total cost.

STEP 2 Identify the configuration that has the lowest total training time in each phase. Let $T_1^2, ..., T_p^2$ denote these times. Therefore, $\{T_1^2, ..., T_p^2\}$ minimizes the total training time. But this may be quite costly. Let μ^2 and Ω^2 be its objective values.

STEP 3 If $|\mu^1 - \mu^2| < \Delta$, where Δ is a pre-specified small quantity, stop. Otherwise, identify the Y_{ij}s that violate any of the following constraints:

$$\sum_{j=1}^{J_i} \{ \sum_t \{a_{ijt} e^{-b_{ijt}\alpha_{ijt}} + c_{ijt} \} Y_{ij} \} \leq \Omega_i^1 / 2$$

Eliminate the violating configurations and solve Model 2. Let this solution be $\{T_1^3, ..., T_p^3\}$ and its objectives μ^3 and Ω^3.

STEP 4 Ask the decision-maker to choose between (μ^1, Ω^1) and (μ^3, Ω^3). If he prefers (μ^3, Ω^3), go to Step 5. Otherwise, denote $(T_1^3, ..., T_p^3)$ as $(T_1^1, ..., T_p^1)$ and (μ^3, Ω^3) as (μ^1, Ω^1) and go to repeat from Step 3.

STEP 5 Denote $(T_1^3, ..., T_p^3)$ as $(T_1^2, ..., T_p^2)$ and $\mu^3, \Omega^3)$ as (μ^2, Ω^2) and go to Step 3.

The above algorithm uses an interactive solution procedure. This, however, can be a time consuming process, since we may have to solve several integer programming problems. We propose the following two heuristics to help minimize the amount of time the decision maker may have to spend on a computer.

Heuristic 1

Divide the interval between μ^1 and μ^2 into classes of equal intervals. Perform Step 3 for each interval and save each solution. Then, using these solutions (there will be a finite number of solutions), the decision maker's preference structure can be assessed interactively as suggested in Korhonen, Wallenius and Zionts (1984), and the most preferred solution can be found.

Heuristic 2

This heuristic follows a similar strategy as Heuristic 1. The interval between μ^1 and μ^2 is divided as before and the solutions obtained in Step 3 in each interval are saved. Then a multi-attribute utility function for the decision maker is determined interactively and the solution maximizing the utility function is also determined. Currently, we are implementing the two heuristics and extending them to treat the training times for each phase separately. However, this causes the number of objectives in the problem to increase and, accordingly, could result in an increase in the cognitive load on the decision maker. We are currently investigating the cognitive load related issues, the algorithm efficiency, and the quality of the solution obtained from the heuristics. Our preliminary investigation reveals that the proposed methodology is viable for solving practical decision problems of moderate size.

7. DATA COLLECTION

In this section, we describe the data collection instruments currently being used to obtain the input data necessary for the proposed model. Given the complexity of the design process, the difficulties in obtaining sensitive data on the design parameters, and the highly subjective nature of some of the critical data elements, the input data is pooled through a series of applications of the data collection instruments with a group of experts in the field of flight simulation. The data collection systems employ an anchored relative scaling procedure to obtain estimates on all of the data elements based on conjoint measurement techniques introduced by Luce and Tukey (1964). Procedures similar to these have been used to determine the utilities and weighs on criteria in decision-analytic models. Our procedures are similar to the Simple Multi-Attribute Rating Technique (SMART) developed by Edwards (1977) and extended by Adelman, Sticha and Donnell (1984). The framework of the data collection systems follows.

The data elements needed for the model can be broadly classified into cost data and training time data, respectively. The data were obtained from the experts and were corroborated with available documentation wherever possible. Initially, a morphological analysis of the documented data forms the basis of the data collection from field experts.

The Delphi Technique is used in the data collection (Turoff 1972). The Delphi technique was originally developed for the United States Air Force by the Rand Corporation to collect expert opinion in formulating nuclear missile strategies. The procedure followed is described below.

Cost Estimation

Initially, each expert assesses the relative costs of each type of operating, instructional and management feature. The fixed and variable components of cost are assessed separately. These costs, owing to their order of magnitude (which are sometimes of the order of hundreds of thousands of dollars) and their extent of variability, are measured on a scale in which the feature with the highest cost received a cost score of 100 and the feature with the lowest cost received a score of 1. The expert assesses the costs of each feature on this reduced relative scale of 1 to 100. This system of measurement is quite appropriate for military applications, since the experts can assess relative costs rather easily. The estimates are then verified by comparing the ranking of costs of a single feature and features in combination. For example, the expert also compares the cost of an instructional feature that receives a rating of 80 with the total cost of two instructional features that may receive ratings of 30 and 45. If the ratings are correct, then the single instructional feature should be slightly more expensive than the combination. This verification procedure continues until the analyst and the expert are satisfied with the ratings.

The expert then proceeds to compare the cost of four instructional features that received the score of 100. These costs have been estimated as discussed above. Each instructional feature is selected from a different functional area of the training process. In this assessment, the four features will be evaluated on a scale of 1 to 100, thus providing a comparison of costs between the functional areas. That is, the overall relative cost for an instructional feature will be just the product of the rating obtained in the first step and the rating of functional area assessed in the second step, which is used as the appropriate standard. The results are again verified by comparing the resulting cost estimates for instructional features both singly and in combination, in different functional areas.

Finally, the expert is asked to choose a set of features for which he could estimate the actual cost in dollars and provide dollar estimates for these features. For example, if the expert estimates that the cost of two instructional features, say remote graphics replay and hardcopy, would be \$170,000, this estimate will be used to set the scale for all other features. The total relative value of each feature is assessed using this estimate and the rating obtained earlier. Finally, the expert is asked to examine the calculated actual costs of the features, both singly and in combination, to determine if they are reasonable and appropriate.

The data collected from each expert is then pooled and summarized. This summary is then given to the experts to review their assessments in the light of the opinions of other experts involved. Several feedback information and reassessments of the data are carried out before the experts as a group converge on a final set of cost data. This is a time consuming data collection, verification and validation process requiring commitment of considerable resources. However, given the magnitude of the design decisions in terms of their dollar values, this investment in data collection is considered necessary.

Time Estimation

The learning curve parameters for each function of each phase for each IOS configuration constitute the input time data for the model. These parameters are indicated as a_{ijt}, b_{ijt} and c_{ijt} in the model. These parameters are estimated as follows.

Initially, the expert is provided with the set of functions for each phase and the set of feasible IOS configurations for each phase. The parameters a_{ijt}, b_{ijt} and c_{ijt} for a function t of a phase i using configuration j are estimated as follows.

The learning curve is given by

$$Y_{ijt} = a_{ijt}\, e^{-b_{ijt} X_{ijt}} + c_{ijt}.$$

Then, the expert is asked to assess the maximum proportion of the training task that can possibly be learned using the simulator training device assuming that a student can be provided infinite hours of training on the simulator equipped with the given IOS configuration. Assume that the expert feels that at most a fraction p of the total task can be learned with a simulator training device. Let T denote the total number of hours of training necessary on the real system if no simulator is available. Then, clearly, it can be seen from the experts' judgment that $Y_{ijt} = pT$ if X_{ijt} is infinitely large. Fitting these values in the learning curve model, it follows that $c_{ijt} = pT$. Furthermore, fitting these values in the learning curve model, it follows that $T = a_{ijt} + pT$ and hence, $a_{ijt} = T(1\text{-}p)$.

Next, the expert is asked the following question: If the training time on the real system were to be reduced from the maximum T hours by 10 percent, how many simulator hours would it take to replace it? If his answer is q hours, then for $X_{ijt} = q$, we have $Y_{ijt} = 0.9T$. Fitting these values, and the already computed values of a_{ijt} and b_{ijt} in the learning curve model, it follows that

$$0.9T = T(1\text{-}p)\, e^{-q\, b_{ijt}} + pT.$$

Solving for b_{ijt}, we get

$$b_{ijt} = -\ 1/q \log(\ 0.9\text{-}p/1\text{-}p).$$

Thus, the learning curve parameters are estimated. The estimates from different experts are then pooled, consolidated and summarized. The summary is then provided to the experts as feedback information for reevaluation. The Delphi procedure is then continued iteratively until the experts converge at a set of acceptable and validated estimates.

8. CONCLUSION

In this research paper, we present a 0-1 integer programming model for the optimal design of Instructor/

Operator Stations for the military flight simulator training device. We also present an interactive solution methodology and appropriate instruments for collecting the input data required for the model.

The proposed model is part of a comprehensive decision support system currently being developed at the Army Research Institute to facilitate the development of cost-effective training systems and plans. The Instructor/Operator Station is a critical component of a fight simulator and the proposed model will enable a designer to choose appropriate configurations and sets of instructional support features for IOS designs that are cost-effective. The model primarily attempts to minimize the cost of an IOS while maintaining acceptable levels of training efficiency. Although the model is tailored to the design of an IOS, it can be appropriately modified to aid design decisions for other training devices as well. In particular, the model addresses a wide range of flight simulators, ranging from small helicopters to sophisticated aircraft simulators. The model can easily be adopted to other training devices whose designs involve the selection of a set of features from a larger set, while achieving a tangible balance between cost and effectiveness. In this respect, the model has a wider scope of application in the design optimization.

Several avenues of future research have been identified from this study. Some of the principal avenues we are currently exploring include the cross-validation analyses on the data collection instruments, and development of a comprehensive expert system for data acquisition, as well as the coverage of a wider range of design decisions. The current model is the core of an expert system for design optimization. These design decisions have been traditionally made by design experts with little documentation on the optimization strategies used by different experts. The proposed system presented in this paper is the first of its kind. It integrates diverse knowledge sources and provides a DSS tool within the framework of mathematical programming for the design of training devices and systems.

9. REFERENCES

Adelman, L.; Sticha, P. J.; and Donnell, M. L. "The Role of Task Properties in Determining the Relative Effectiveness of Multi-Attribute Weighting Techniques." *Organizational Behavior and Human Performance*, Vol. 33, 1984, pp. 243-262.

Andrews, D.; Singer, M. J.; Ozkaptan, H.; and Hofer, R. *Optimization of Training Systems and Subsystems: Models, Methods and Data Bases*, Technical Report. Orlando, FL: Army Research Institute, Naval Training Systems Center, 1987.

Bateman, R. P., and Hottman, S. B. "Methodology for Estimating Transfer of Training Functions." *Proceedings of the Human Factors Society Twenty Seventh Annual Meeting*, 1983, 916-1918

Bickley, W. P. "Optimizing Simulation-Aircraft Mixes," *Proceedings of the Interservice/Industry Training Equipment Conference*, November 19-20, 1980, pp. 210-219.

Edwards, W. "How to Use Multi-Attribute Utility Measurement for Social Decision Making." *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-7, 1977, pp. 326-340.

Farquhar, P. H. "Utility Assessment Methods: State of the Art." *Management Science*, Vol. 30, No. 11, 1984.

Green, P. E., and Srinivasan, V. "Conjoint Analysis in Consumer Research: Issues and Outlook." *Journal of Consumer Research*, Vol. 5, 1978.

Keeney, R. L. and Raiffa, H. *Decisions with Multiple Objectives: Preferences and Values Tradeoffs*. New York: Wiley, 1976.

Koksalen, M.; Karwan, M. H.; and Zionts, S. "An Improved Method for Solving Multiple Criteria Problems involving Discrete Alternatives." *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-14, No. 1, 1984.

Koksalen, M.; Karwan, M. H.; and Zionts, S. "Approaches for Discrete Alternative Multi-Criteria Problems for Different Types of Criteria." *IIE Transactions*, Vol. 18, No. 3, 1986.

Korhonen, P.; Wallenius, J.; and Zionts, S. "Solving the Discrete Multiple Criteria Problem Using Convex Cones." *Management Science*, Vol. 30, No. 11, 1984.

Luce, R. D., and Tukey, J. W. "Simultaneous Conjoint Measurement: A New Type of Fundamental Measurement." *Journal of Mathematical Psychology*, Vol. 1, 1964, pp. 1-27.

Rivett, P. "Multi-Dimension Scaling for Multi-Objective Policies." *Omega*, Vol. 5, 1977.

Roscoe, S. N. "Incremental Transfer Effectiveness." *Human Factors*, Vol. 13, 1971, pp. 561-568.

Roy, B. "Problems and Methods with Multi-Objective Functions." *Mathematical Programming*, Vol. 1, 1971.

Siskos, J. "A Way to Deal with Fuzzy Preferences in Multi-Criteria Decision Problems." *European Journal of Operations Research*, Vol. 10, 1982.

Turoff, M. "The Design of a Policy Delphi." *Technological Forecasting and Social Change*, Vol. 2, 1972, pp. 149-171.