AN ON-LINE MANAGEMENT SYSTEM USING ENGLISH LANGUAGE

Andrew Vazsonyi
Ramo-Wooldridge
Canoga Park, Calif.

Summary

The demonstration model of an on-line management system presented in this paper aims to provide increased management capability to executives charged with planning and control of large scale research development and production programs. The technique is formulated as an exercise in Decision Gaming and special emphasis is laid to the problem of providing capability for quick and optimum reprogramming of dollars, manpower, facilities and other resources. The task of planning and control is structured into two components, the more routine tasks are assigned to the equipment, whereas problems requiring executive judgment are delegated to players of the Decision Game. Through the use of mathematical models and computer routines the consequences of proposed reprogramming actions are presented to the players in terms of financial and manpower requirements, facilities loading, etc. Through a step by step man-machine process, optimum programs and the best utilization of resources is reached. Management data is retrieved and manipulated on an on-line basis and all operations of the equipment are executed through every day English commands. All data is displayed on cathode ray tubes and projection screens, including instructions to the players on how to operate the equipment and how to play the Decision Game. Input to the equipment is provided through (1) a permanently labeled keyboard, (2) a blind keyboard that can be provided with appropriate labels through a set of plastic overlays. The computer action resulting from depressing of keys must be programmed and is not wired permanently. By providing a set of independently operated input-output consoles, connected on-line to the same computer system, a significant advance in the art of the design of management systems is provided.

Introduction

The management planning and control technique described in this presentation has been developed for certain military and civilian activities with the purpose of assisting executives in evaluating and re-programming complex activities. However, it is believed that the technique is equally applicable to the planning and control of other large scale research, development, production and construction programs.

In order to apply the planning and control technique to an activity it is necessary to divide the activity into "elementary" programming blocks. The technique requires that first alternate scheduling and financial data on each of these planning blocks be developed. In addition, it is necessary to formulate explicitly the inter-relationships between the planning blocks. These relationships specify the permissible time phasing of the elementary programming blocks, the associated dollar and manpower requirements, facilities requirements and other financial requirements. The planning and control technique employs a network analysis of the various activities involved and permits the exploration of a large number of planning combinations.

The primary purpose of the management planning and control system is to assist executives in re-programming. As an illustration, suppose that plans are compared with progress, a deviation is observed and re-programming of the different activities is required. For instance, it might be necessary (1) to cancel a program, (2) to stretch another one out, (3) to accelerate one, or (4) to decrease or increase production quantities. Another situation when the need for re-programming arises, when there is a budgetary change and financial trade-offs between various programs must be considered. For instance, executives may want to know that if a particular deadline is postponed by six months, how many dollars and what manpower can be made available to another program, and by how much can this other program be accelerated.

The actual program analysis and re-programming activity is carried out through the medium of a Management Decision Game.

Brief Description of Decision Gaming Technique

The Decision Game is to be played in three steps. As a first step the players of the Decision Game gather in the Control Room where the Game is to be played and where the various information displays can be retrieved with the aid of the computer system. The displays present all the important planning factors relating to the activities to be re-programmed. The time phasing of various missions, deadlines and goals, and the associated loading of various facilities can all be displayed. The associated financial information can also be shown with sufficient detail so that financial consequences of re-programming decisions can be made. Provisions are made to retrieve further back-up information when requested, from a file of status of progress and alternatives.

As a first step of re-programming a comprehensive analysis of the status of the programs is carried out. The computer system is provided with the capability of furnishing status information on a real-time basis and in everyday English. Information related to all matters pertaining to the progress of various programs is displayed in cathode-ray tubes or in projection screen. After this status analysis is completed, the players have adequate information to perform the second step of the Game.

This second step of the Game consists of making a "move". Such a "move" may involve a time shift of some of the deadlines, milestones or goals and/or a change in the delivery quantities involved in the program. As suggestions for reprogramming "moves" are made, the proposed changes are put into the computer system through the use of an appropriate keyboard and Communication Display Tube. At the direction of the operator the computer and associated equipment takes over and the third step of the Game, that is the re-programming computations, are carried out.

This third step of the Decision Game is executed by the computer in accordance with mathematical models and associated computer routines stored in its memory. Within a time span of seconds the computer prepares a new program, including all the new deadlines the new phasing of sub-programs, facilities loading, manpower and financial implications. When the computer finishes the computations, the data is presented to the players through cathode-ray tubes and/or slide projections. By examining the various displays and by retrieving more detailed information, the players can evaluate whether the suggested solution to the re-programming problem is satisfactory.

In most situations the first suggested program will result in conditions that are not acceptable to the players of the Game. Therefore, after considering the results of the "move" and discussing further implications of the data, a new proposal for re-programming will emerge and a new cycle of the Decision Game will be entered upon. By a series of steps it is possible to develop a final program that is acceptable to the players.

At the termination of the gaming exercise all the implications of the final program are recomputed with greater accuracy. It is not expected that this re-computation will result in major changes, but only that the re-computation will provide an accurate, acceptable and detailed plan.

All communications between man and machine are performed in a real-time manner and in everyday English.

The File of Status and Alternatives

The Gaming Technique described here allows the examination of a panorama of alternatives. The analysis can be performed only if in the memory of the computer, techniques for examining many alternate possibilities are stored and programmed. It is recognized that it is impossible to store all the possible alternatives and therefore, a method to study alternatives must be provided. The analysis is made possible by the application of mathematical modeling techniques and by the storage of certain basic system parameters. The mathematical model uses the elementary programming activities as basic building blocks and relates these activities to each other through mathematical relationships. For instance, alternate ways to accomplish a basic programming block can be associated with various estimates of completion dates and costs. The mathematical model summarizes the relationships, and also relates the different activities to each other through equations and inequalities. Manpower and financial requirements appear as dependent variables, whereas the time phasing of the various activities as independent variables.

In order to avoid the necessity of manipulating a large number of parameters, sub-optimizing techniques are introduced. For instance, it might be required that certain types of sub-programs be accomplished at a minimum cost and this policy can be embodied in a system of equations through mathematical programming techniques. By such relationships, the majority of the variables of the system can be made to depend on a few control variables. With the aid of mathematical models and sub-optimization techniques it becomes possible for the players to manipulate only a few of the major variables and still examine a large number of alternate plans.

Equipment Requirements

There is no equipment on the shelf today that can carry out in all its details the management planning and control technique described here. However, there is equipment available, which with minor modifications would possess the capability required. A detailed study of the Ramo-Wooldridge Polymorphic Computer System and Display Analysis Console, for instance, shows that essentially all the required features could be made available in a short time. This computer system has been described elsewhere,and in this discussion equipment details will not be included.

Decision Gaming

Detailed Description of Decision Gaming System

The fundamental concepts underlying the Decision Game are shown pictorially in a simplified form in Figure 1. Three displays enable the players to communicate with the computer. The first of these is a visual representation of the time phasing of all the important missions and goals. The information on this display is schematically represented in the upper part of Figure 1 and is to be displayed in the "Program Network Tube" of Figure 2 (projection capability can be provided if desired so that a group of participants can analyze the data). Sufficient details will be shown so that all milestones of importance are displayed, but the data will not be so detailed as to confuse the players. As the Game starts, various questions will arise which will not be immediately answerable by the displayed material. To meet this condition, back-up displays will be stored which can be retrieved by the players as requested. By this technique, it will be possible for the players to go into any degree of detail in the time phasing of the missions and goals without making the presentation too cumbersome or confusing.

A part of the display on the "Program Network Tube" is the visual representation of the utilization and loading of the different facilities associated with the programs considered. This display is shown by the third item from the top in Figure 1. All the previous comments made in connection with the visual representation of Programs A and B apply for the Facilities Loading displays, too. Sufficient detail will be given so that the player can appraise the state and progress of various programs, and again sufficient back-up information will be available through retrieval.

The second display refers to dollars, costs, manpower, and other resources. These are represented graphically in the lower part of Figure 1 and are to be displayed on the "Resources Requirements Tube" of Figure 2. The dollar and manpower profiles as they unfold in time will be represented in sufficient detail so that all the important information for the players will be furnished. In addition, when it is required, the players will be furnished with hard copies of printed financial information.

The display capability so far described furnishes the players of the Game with such pertinent information as past history, status, and future projections of programs. Particular emphasis is placed on the preparation of this information in such a form that organizational structure and responsibilities are directly tied in to the information displayed.

The lower right corner of Figure 2 shows the "Man Machine Communication Display". This is the tube that offers choices of instructions to the player in plain English. This tube is used mostly for non-standard type of instruction to the player, as ordinary instructions (say: "Machine Is Busy") are provided through the illumination of status lights.

So far we have described the display systems and the type of information stored. We are now ready to proceed to the description of how the Decision Game is to be played. In order to be able to speak in more specific terms, we take the hypothetical problem of a new requirement, that a particular mission is to be accomplished one year ahead of schedule. This new requirement requires the acceleration of a major program and a reorientation of the resources available.

When such a problem arises, various discussions take place at different managerial levels. We do not propose that the Decision Game is to replace these conferences. However, after a preliminary consideration of the problem the appropriate management group gathers in the control room to play the Decision Game. By a step by step procedure, they evaluate, modify and sharpen the preliminary ideas that have risen in connection with this problem of advancing the completion date of a major mission.

When the group meets the first time in the control room, the players begin by retrieving a number of different displays to update and verify their knowledge of the status programs. Such a review consists of inspecting the principal displays associated with the problem and of retrieving various back-up information. After such a preliminary discussion, a proposed first solution to the reprogramming problem is suggested and information defining the proposed change is key-punched into the computer.

At the instruction of the players the computer begins to carry out the routine associated with the particular reprogramming problem introduced. The computer consults the Data and Program Reservoir containing the file of status and alternatives shown on the lower left-hand side in Figure 2, and on the basis of stored information and routines, computes dollar and manpower requirements. In addition, facilities requirements and loading are checked and computations are made to determine whether the desired acceleration is feasible at all.

As the computer proceeds through its routine, it might find that the proposed acceleration is impossible or impractical. It is possible that even if all projects are put on a crash basis the mission could not be accomplished within an acceptable date. It may be that for instance manpower is not available, even if more shifts are employed. Under such conditions, the computer will indicate that the plan is not feasible and it will display on the "Communication Display Tube" a warning signal, which shows in detail why the proposed solution to the reprogramming problem is not feasible.

At this point, a group discussion follows to determine whether by a higher order of decision a solution could be found. For instance, it might be decided that another facility can be built or made available, or that another contractor can be called in. Information available to the decision maker will not always be programmed into the computer and, consequently, feasibility indicated by the computer will occasionally be considered as tentative.

If, indeed, a need for such a new alternative way of proceeding with the problem exists, this information must be put into quantitative form and fed into the machine. On the other hand, if the computer indicates general feasibility, then the players can immediately proceed to further evaluation of the proposed program.

When the program modification is feasible, the players are primarily concerned with resource requirements and with dollar and manpower profiles associated with the program. It is very likely that the first solution proposed will not be acceptable from the point of view of budgetary considerations. It is likely that the costs at certain phases of the program will be beyond possible funding, and perhaps at some other times there will be an indication of surplus funds. This, then, is the point where the players reconsider the time phasing of the mission and goals and propose an alternative. When the players agree on the next trial of the program phasing, information is fed into the computer and the computer proceeds with computations to prepare a new program. Again, the computer first explores feasibility and then proceeds to the detailed generation of the resource requirements.

It is seen that through a step by step process of deliberation, discussion and computer routines, the players will reach better and better solutions to the reprogramming problem. It is envisioned that programming computations will be carried out first by a "quick and dirty" method and then by a more accurate routine. This will allow the players to explore tentative alternatives rapidly and there will be no unnecessary delay in waiting for accurate computations which would not be utilized in actual program plans. The computer will carry out accurate computations either automatically (when computing time is available) or at the special direction of the player. This approach allows the decision makers to make rapid changes and explore and evaluate dozens of different program proposals. As the Decision Game progresses, more and more satisfactory solutions to the reprogramming problem will be found. Towards the terminal phase of the gaming exercise, the players may desire highly accurate estimates of the various program details. If this is so, it may be necessary to direct the computer to carry out more accurate special program computations, and it may then be necessary for the players to wait for a longer period of time to get the phasing of programs and the resource requirements. Finally, the computer is directed to develop and print a definitive program which will be used as a planning document. Computation of such a program may require hours, and consultation with other agencies and contractors.

So far, we have given only an outline of how the Decision Game is to be played and described only those phenomena that will be observed by the players. Now we proceed to take a look inside the equipment and see how the various logical steps, routines and computations are carried out.

Illustration of Reprogramming Computations

The basic principle in carrying out reprogramming operations is to provide the computer with data on possible alternatives and also with the myriads of details on how these alternatives can be combined into programs. The computer can be programmed to go through a large number of calculations in an efficient fashion, and therefore alternate programs can be generated by the computer in a matter of seconds. In order to illustrate the techniques, we will describe an extremely simple but still significant reprogramming problem.

Figure 3 is a chart showing six different jobs and the time phasing of the start and completion dates of each of these jobs. In this simplified programming Game, we are concerned only with the monthly dollar expenditures which are shown in the bottom of Figure 3. Suppose the player desires (1) to accelerate by two months the accomplishment of Goal B (that is the terminal dates of Job No. 3 and 5); (2) to accelerate by three months the final completion of the mission, that is of Goal A; (3) leave all other goals unchanged. The computer is to determine whether such an acceleration in the program is feasible, and what kind of dollar expenditures would be associated with this accelerated program.

As this reprogramming information is keyed into the machine, the machine examines all jobs to see which is immediately affected by the acceleration of Goals A and B. The computer selects Jobs 3, 5 and 6 and evaluates the possibility of accelerating those three jobs. It finds that the time span of Jobs 3 and 5 are to be compressed by two months and of Job 6 by one month.

At this point, the computer seeks information on alternative ways of accomplishing Jobs 3, 5, and 6. As the computer consults the file of alternatives, it finds for each job the time-cost relationship shown in Figure 4. The horizontal axis shows alternative time spans allowed for the job, the vertical axis shows the total dollars that must be expended, if the job is to be accomplished in the time specified. It is seen, for instance, that a crash program--doing the job in the shortest possible time--requires more total funds than a more orderly and efficient execution of the task. In the case of a stretch-out, due to overhead and some other supporting activities, the total cost of the job would also increase. The computer also finds how these dollars would be expended in time. (Dotted lines in Figure 4.) The file of alternatives has curves of this type for each of the jobs and therefore the computer can establish that the jobs can indeed be accelerated to the desired time span, but that a higher expenditure of funds is required. Using this information, the computer can replace the previous budgets for Jobs 3, 5, and 6 with the new budgets and determine a new dollar profile associated with the accelerated program. We see that when the computer reprograms, it first proceeds through these computational steps and then transmits the information to the display devices. The player can visually observe the required funding associated with the accelerated program.

We recognize that in a real problem we would deal with a much more complicated set of routines. Manpower profiles would have to be computed, facilities loadings would have to be checked, many other items of information on compatibility would have to be considered. In the case of prototype production, or in other tasks where quantities are involved, relationships dealing with "quantity made" would have to be included in the analysis. However, basically, these considerations would only complicate (admittedly by a great extent) the routines that the computer would have to go through, but, conceptually, reality would not add significant new difficulties to the method of solution.

The time cost relationships as shown in Figure 4 form the basis of the file of alternatives that a computer has to consult. As we already mentioned, there are types of problems where more complex mathematical models form the building blocks for the file of alternatives. However, for purposes of our discussion, we will concentrate on the concept of time-cost relationships and we will show how such relationships can be generated. We will show how the basic input data is to be obtained and how these data can be built into the appropriate files for representing various alternatives that the programming task may require.

Concept of Alternatives

Let us reiterate the type of information we seek. The player moves some of the gaming goals in time and certain jobs must be performed within the time limits indicated by the player. We need to find a way to determine the dollar requirements associated with the various alternatives.

Let us begin by considering a relatively simple job or task. Suppose that there is a single manager in charge, and let us assume that this manager has a good grasp of all the details involved of this particular task. The manager does his own planning with paper and pencil and by discussions with his associates. We ask him to determine how much would it cost to perform this job in an "orderly" fashion. After studying the problem, he estimates manpower, material, overhead and dollar requirements. In Figure 5, the financial information is shown in a graphical form. In the horizontal axis we show the time allowed to complete the task; on the vertical axis, we show the associated effort (say dollars per week) required. "Orderly" performance of the task is represented by the "most efficient" point in the chart. We also ask the manager to determine what it would take to complete the job on a crash basis. He would need more men, more resources, he would require a larger effort, but he could complete the job in a shorter time. This crash program is shown in our chart in Figure 5 by the "minimum time" point. We can also ask him to determine the minimum level of effort required to do the job at all. He needs two mechanical engineers, an electronic expert, a technician, a secretary. This establishes his minimum effort level and gives the "minimum effort" point in Figure 5. We connect the three points by a curve and obtain a time-effort relationship and we assume that we could also operate at intermediate points on this curve. With the aid of the curve shown in Figure 5, we can determine the time-cost relationship shown in Figure 4. All we have to do is to multiply the rate of effort by the time required for the job, to get total costs.

We see, then, that we have a technique to get time-cost relationships, at least for relatively simple jobs. However, if we want to extend this technique to more complex tasks, we run into problems. It is difficult or impossible to find managers who have all the details of a complex job. Consequently, in order to make cost estimates, the manager must work with his subordinates and must combine in a complex fashion many items of information. This combination of data is a tedious and difficult job but is precisely the kind of task that computers can execute with great efficiency. Therefore, we propose to prepare time-cost curves for complex jobs with the aid of computers. We will show how, with the aid of mathematical models and sub-optimization technique, one can construct time-cost relationships.

Sub-Optimization Considerations

Let us take a simple combination of two jobs which have to be performed in sequence. Various alternate time spans are allowed either for Job No. 1 or No. 2. This implies a number of combinations of ways that the two jobs can be performed. In Figure 6 we show the problem in a graphic way. Suppose tentatively we select a certain duration for Job No. 1, and we determine the associated dollars required with the aid of the time-cost relationship. In Figure 6 this time-cost relationship is represented by point A. Now by starting with this time span, we can assign different time spans to Job No. 2. A possible representation for Job No. 2 is point B. It is seen that we can combine the two time-cost curves in many different ways. In Figure 7, the various possible time-cost curves for Job No. 2 are shown by dotted lines. Now we need a policy to select, out of these many possibilities, the desirable ones.

Suppose we agree that we want to complete the two jobs within a given time span, but with the least amount of money. Let us recognize that when the combined time-span for the two jobs is specified, still there are many ways to do the two jobs; out of these many possibilities there is one that yields the lowest cost. In Figure 7, these low cost combinations are represented by the envelope of the dotted curves. We say then that this envelope, corresponds to our policy of minimum cost, and this envelope is the combined time-cost relationship for the two jobs to be performed. For instance, if we wish to complete the two jobs at point P in Figure 7, we draw the vertical line from point P until we reach the envelope at point Q. This gives the combined cost of the two jobs. Working backwards from point Q, we can get point R which represents the time and cost requirements of Job No. 1.

The policy we used here is to perform the two jobs with the lowest possible cost. If there is another policy such as say a constant manpower requirement or the utilization of a facility, etc., each of these policies would have to be programmed into the computer. The important point, however, is that even if complex policies are formulated, due to the high-speed capability of the computers, consequences of these policies can be deduced efficiently.

Actually, the computer would not construct the envelope of the curves, but would solve the appropriate mathematical problem. It is easy to show that the two jobs are to be combined in such a fashion that the following equation holds:

$$\frac{dC_1}{dT_1} = \frac{dC_2}{dT_2} \qquad (1)$$

Here on the left-hand side we have the derivatives of the time-cost relationship for the first task and on the right-hand side, the derivative relationship for the second task.

The computer would compute these derivatives, select the appropriate combinations of the tasks and generate the new time-cost relationships.

In Figure 8, we show a somewhat more complicated problem when a sequence of jobs is to be performed. Here it can be shown that the following equation must hold:

$$\frac{dC_i}{dT_i} = \lambda \qquad (2)$$

The meaning of these equations is that the derivatives of (that is the slopes to) the time-cost curve must be equated. This procedure can be observed in Figure 8 by considering the three upper curves and realizing that the three tangents shown are all parallel. Another representation of the same set of equations is shown by the lower set of curves. These are the derivatives (or slopes) of the time-cost curves. The corresponding points on the time-cost curves are selected by taking points on the same vertical level. Again, this is the type of computation that a computer can carry out very efficiently.

Another way to describe the technique used here is to realize that whereas many goals are to be manipulated during the course of a Decision Game, some of these goals are not sufficiently important to be manipulated directly by the players. Therefore, some "slave" goals are automatically manipulated by the computer. We can say that placing of the slave goals is accomplished by an appropriate sub-optimization technique. For instance, in the discussion so far, we sub-optimized by using least-cost job combinations. As stated before, some other principle might be involved in positioning of the slave goals and then other corresponding sub-optimization principles must be developed. It is also possible that in some complex situations, one would have to be satisfied by accepting a relatively "good" solution instead of trying to find a sub-optimum.

In Figure 9, we show a somewhat more complicated problem. Those goals marked with crosses can be made slave goals by the technique so far described. However, goals A and B are interconnected as they have to be completed at the same time, and therefore this interconnection must appear somehow in the computational procedure. What we have to do is to take the time-cost relationship for the first and second jobs up to A and B, add the cost of these two jobs together and construct a single time-cost relationship. We have to go through the same procedure for the jobs to be performed after B, and form a single time-cost relationship. When we have these two time-cost relationships, we have reduced our programming problem to the problem of having two jobs to be performed in sequence. Now we can use the technique already developed.

In a way we could say that first, we turn into slave goals those goals which are in series, and then those which are in parallel. By use of this principle step by step, we can construct the necessary time-cost relationships for complex programs.

In summary, we can say that we get basic data on relatively simple jobs from managers of simple projects. Then we formulate the rules of combining these simple jobs into complex jobs, and through some method of selecting the most appropriate combination, we construct combined time-cost relationships.

Let us, however, recognize that when we deal with really complex structures, it might not be possible to put into logical or mathematical form the policies that yield the most desirable combination. If this be the case, it is necessary to resort to auxiliary gaming technique to establish the file of alternatives.

The problem shown in Figure 9 would be solved now by a group of executives moving goals A and B and by examining the consequences of these moves. Here we have to perform the same type of gaming as we have previously described. In Figure 10 we show in a schematic form, the multi-state man-machine gaming system that we envision here. On the top we show the game that we have already described and which is to be played by top executives. On the lower level we show subsidiary games which would be performed by middle management personnel. The purpose of the lower level of gaming is to provide a planning and control system for middle level management and to provide the files of alternatives to top management.

Perhaps the most significant aspect of this multi-stage gaming technique is that various levels of management could participate in a most effective fashion in reprogramming efforts. As problems develop at lower levels of management, these are reviewed by middle management and the implications of changes in program phasing are incorporated into plans. Even more significant is that not only single plans are developed, but alternative possibilities of tackling jobs are considered. When middle management agrees on various alternatives, these are placed in the file of alternatives and thereby these alternatives are made available to top management. This way top management is apprised of the most recent and significant changes in the time phasing of programs and is provided with a capability of using the best updated information.

Confidence Factors in Programming and Scheduling

We have so far attempted to divide the planning task between equipment and man in a systematic way. We recognize that a great many logical and mathematical tasks must be performed in order to generate program plans, and that many of these tasks can be performed better by computers than men. We believe that the capability of computers surpasses human judgment in one more specific area, namely in connection with the problem of estimating the degree of uncertainty associated with estimates of dates of computations of various tasks.

It has been found that human judgment is fairly good in estimating upper and lower limits of when a job will be completed, provided the task to be performed is relatively simple, and provided the man who makes this judgment is completely familiar with the job to be performed. However, when people combine the various component estimates of complex jobs, we find that it is difficult to get reliable answers.

We show in Figure 11 the problem in a highly simplified form. Suppose there are three tasks to be performed in sequence and for each, there is an uncertainty of the completion date. These uncertainties are shown in the diagram by the shaded areas, lower estimates being the optimistic ones while the higher ones are the more pessimistic estimates. In the lower part of the diagram we add the times required for the three jobs together and also add the uncertainties (three shaded areas) into a single one. How to measure now the uncertainty in completing the three jobs?

The total variability is of course shown by the sum of the shaded areas. However, it is unlikely that all three jobs will be completed at the earliest possible completion date, or conversely, that all three jobs will take the longest time estimates. Therefore, we can say that whereas the total variability is shown by the combined shaded area, the area does contain some unrealistic completion dates.

If we think in terms of a more complex program, our problem becomes more acute. When there are hundreds of jobs to be performed, it is impossible for the unaided human brain to form a composite picture of the probabilities involved.

However, this is a sort of problem that statisticians have already studied. In Figure 12, we show a simple example when three different jobs are to be cascaded. If we estimate probability distributions of completion dates for each task and associated standard deviations, then at least under certain simplified conditions, we can use the following equation for determining the standard deviation of the composite probability distribution:

$$\sigma^2 = \sigma_1^2 + \sigma_2^2 + \sigma_3^3 \qquad (3)$$

In this equation on the right-hand side are the squares of the individual standard deviations, while on the left-hand side is the square of the composite standard deviation.

So far, we have talked only about a very simple situation. However, recently some mathematical studies have been made, on how to combine probability distributions for complex programs. It is believed that when these techniques are combined with machine computations, one can obtain more reliable estimates of completion dates than is possible today by unaided human judgment.

Weapons Systems Programming and Control System

(An Illustration)

The United States Air Force controls one of the largest and most complex research, development, production and operational programs in the world. Management of these programs represents a formidable task and considerable effort is devoted to develop new and better management techniques. One of these management planning efforts goes under the name of WSPACS, or Weapons Systems Programming and Control System.[1] The objective of this system is to provide the Air Force with a broad planning device and also to provide techniques of use to both Air Force and Industry in maintaining control and surveillance over the expenditures of development and production contracts. In order to demonstrate that a man-machine management system will furnish the required capability, a demonstration model has been recently constructed and tested. So far this WSPACS Mod 0 model has been programmed on a conventional computer, but here in this discussion we want to explore the possibility of how such a weapons system programming and control system could be carried out within the framework of our on-line management system. As a method of presentation we will use the description of a hypothetical exercise in reprogramming.

Let us say that at a certain day the planning staff of the appropriate Air Force organization is called together and is advised that a change in the planning of programs is required. Specifically, the problem arises from two new requirements. (1) There is a reduction in next year's fiscal expenditures from $4.6 billions to $4.4 billions; (2) it becomes extremely desirable to accelerate the Air Force's missile programs.

When this problem of reprogramming is presented to the staff they gather around the display console of the computer system and begin an analysis of the various Air Force programs. The man-machine system employed is shown in general, in Figure 2, the special keyboard overlay to be used in the reprogramming exercise is shown in Figure 13. The key labeled "Start Routine" is lit (from under) indicating that this is the key the operator must depress to start the analysis. As soon as this key is depressed the following instruction appears on the Man-Machine Communication tube:

```
THIS IS A WSPACS ANALYSIS. CONSULT
YOUR MANUAL BEFORE PROCEEDING.  IN
ORDER TO CARRY OUT ANALYSIS DEPRESS:
"PROCEED WITH ROUTINE" KEY.
```

The operator inspects the keyboard and realizes that in fact only the "Proceed with Routine" key is lit and that therefore this is the only key he is permitted to depress. He proceeds to depress this key.

On the "Resources Requirements Tube" of Figure 2, (on the right hand tube) the information shown in Figure 14 is displayed. The staff observes the various Air Force programs and associated financial information. For each program D and P, that is design and production and Sys-conn, or systems connected expenditures are shown. On the bottom Non-System costs, total Expenditures and expenditure Limitations are shown. Let us realize that in this display not all weapon systems are shown and that financial information is shown only up to 1965.

However, by inspecting the lower left-hand portion of the keyboard, we notice that provision is made to scan tables of information. For instance, if the key "Right Tube" is depressed and simultaneously the key labeled by the arrow pointing to the left is depressed, then the numbers shown in the columns in the right-hand tube will shift to the left and financial information for 1966, 67, etc., appears. By this means, any limitation on the horizontal and vertical capacities of the cathode-ray tubes is overcome.

The staff now analyzes the financial data shown on the right-hand tube and decides to proceed with the analysis. They note that the Man-Machine Communication tube is displaying the following statement:

```
YOU MAY SELECT PROGRAM FOR ANALYSIS
ON ALPHA-NUMERIC KEYBOARD.
```

The operator also notices on the keyboard that the key "Operator: Select on Alpha-Numeric Keyboard" is lit. (This is further verification to the operator that he is to use the Alpha-Numeric keyboard.) The staff decides to proceed with an analysis of the Atlas Program, therefore the number 1 is key punched on the Alpha-Numeric keyboard. At this instant on the Man-Machine Communication tube the following statement appears:

```
YOU SELECTED ATLAS FOR ANALYSIS.
YOU MAY INTRODUCE THE FIRST
ALTERNATE IN ATLAS PROGRAM BY
DEPRESSING "PROCEED WITH
ROUTINE".
```

Simultaneously on the Program Network Tube (to which we refer to as the left tube), details of the Atlas program appear. It is noted that there are in total 276 units in the program, that there are 12 units per squadron, that so far 8 units have been delivered and that there are no active squadrons as of today. The authorization of the Atlas (go-ahead date) was May 1958, and the last delivery date is June 1964. The left-hand tube also shows schedules and expenditures for Atlas. For instance, in 1961 there are 49 units to be delivered and 4 squadrons projected. Atlas expenditures are $65 million for design and production and $316 for systems connected costs. Total non-systems cost for all programs are $1,200,000,000, leading to a total Air Force expenditure of $4,591,000,000. (The new expenditure limitation is $4,400,000,000.)

Similar data is shown for each fiscal year up to 1970. We recognize that not all these data can be put on the tube simultaneously. However, with the aid of display control keys we have the capability of scanning these tables up, down, right, and left.

At this instant the planning staff is studying on the right-hand tube the financial aspects of all the weapons systems programs, and on the left-hand tube details of the Atlas program. With the aid of a retrieval system not described here, further information relating to Atlas and other Air Force programs is displayed and analyzed by the staff. After considerable exploration and discussion it is proposed that a trial be made to modify the Atlas program. The "Man-Machine Communication" tube indicates that such a change can be carried out by depressing the "Proceed with Routine" key.

When this key is depressed the following instruction appears:

```
YOU MAY AS FIRST ALTERNATE FOR ATLAS
CHANGE

  A. LAST DELIVERY DATE
  B. TOTAL NUMBER OF UNITS IN PROGRAM

OPERATOR: USE ALPHA-NUMERIC KEYBOARD.
```

The staff decides not to change the number of units but to require that the last unit be delivered by January 1963 instead of the original June 1964. On the Alpha-Numeric Keyboard the letter "A" is punched, then the date January 1963.

On the Man-Machine Communication tube a statement appears to verify that this is indeed the change desired. In addition, on the left-hand tube under the heading of "First Alternate" the proposed last delivery date of January 1963 appears. The new instruction to the operator indicates that he can have the Atlas Program recomputed on the basis of this new delivery by depressing the "Proceed with Routine" key. However, if he made a mistake, he can "Cancel Keyboard Input" or for that matter he can "Cancel Last Instruction".

When the "Proceed with Routine" key is depressed the computer goes into a complex routine, based on the mathematical model developed for WSPACS.[1] Units to be delivered, squadrons projected and all expenditures for the Atlas program are re-computed on the basis of the proposed last delivery date of January 1963. In addition new totals for all Air Force programs are computed. This new information appears on the left-hand tube, tabulated under the old rows of information.

Now the staff has the choice of introducing this proposed change on the right-hand tube into the complete Air Force program, or make further Atlas trials. Results of various trials will appear simultaneously with the original plan on the left tube. After the staff has experimented with sufficient number of alternatives, they agree on a single proposed change for the Atlas program. This change is introduced on the right-hand tube into the Weapons Systems Program.

Now the staff is ready to proceed to another weapons system. Without going into the details of the actual exercise, we state that the go-ahead date and the last delivery date of certain programs can be changed. Some programs can be cancelled, or in others the numbers of units per squadron can be changed. In certain instances the phasing-out of weapon systems can be modified. In addition, it should be pointed out that certain subsidiary weapons systems programs are automatically changed as the major programs are changed. For instance, as the primary weapons systems programs are changed, the requirements for KC-135 changes, and these changes are introduced automatically into the system. In addition, capability is given to change the "Bomber to Tanker" ratio and "Bomber to GAM" ratio. It is seen then that as the analysis progresses the various weapons systems listed on the right tube are scanned and proposed changes are introduced. Through a step-by-step process a new weapons systems program is developed that is within fiscal limitations and meets the requirements imposed by the accelerated need for missiles.

In an actual demonstration on November 29, 1960, the following changes were made:

A. Accelerated Atlas Program by advancing the last delivery date from January 1964 to January 1963.

B. Moved the go-ahead date of the Minuteman from July 1961 to December 1960.

C. Cancelled the B-58.

D. Speeded up the phase out of the B-47 by reducing to 70 squadrons in fiscal year 1961, instead of 79 squadrons as originally planned.

As these changes were introduced, computations were carried out to show increases in expenditures for the Atlas and Minuteman Programs. Savings due to the cancellation of the B-58 were also computed. In addition, due to the cancellation of the B-58, reduction occurred in the quantities of B-58's and GAM's required. This resulted in savings in the KC-135 and GAM areas. Finally, the accelerated phase-out of the B-47 resulted in additional savings.

As a result of these actions, fiscal 1961 expenditures were brought within the revised expenditure limitations and the missile programs were accelerated.

It is to be emphasized that the exercise described here was carried out on the basis of a highly simplified mathematical model. Currently an effort is underway to improve the mathematical model by making it more realistic and flexible. However, it is expected that through the man-machine management system described here, the Air Force will be provided by a new increased capability in solving the difficult task of re-programming complex weapons systems programs.

Implementation Considerations

As of today there is no management system in operation patterned along the lines discussed in this paper. However, all elements required for the establishment of such a system are in existence and we believe that within a few years we will indeed see systems of this type operational. We wish to finish our paper by a brief discussion of problems of implementation and the general outlook for on-line management type systems.

Let us focus our attention to three fields of effort required for implementing such systems: (1) design of management systems in general,

(2) development of mathematical models, and (3) equipment considerations:

As far as the design of quantitative management control systems is concerned, during the last few years significant advances have been made.[2,3] In particular, we refer to efforts like the Navy's "Program Evaluation Review Technique" (PERT) and the Air Force's "Program Evaluation Procedure" (PEP). Such advanced management techniques have shown significant success and there is today a substantial effort applied to extend these techniques. We recognize as one of the most significant weaknesses of current systems, that resource allocations and in particular financial considerations are not adequately treated yet. However, the critical need for such management systems exists and it is certain that significant further progress will be made within the next few years. Consequently, we believe that system design requirements for on-line type management systems could be met within a time span of about one to two years.

The second field of endeavor we want to talk about is the development of mathematical models. The system design work cannot be carried out without the appropriate mathematics. In the field of mathematical models significant progress is being made today[4,5,6] and it can be predicted with reasonable certainty that further progress will be made within the next few years. The type of mathematical model required for on-line management systems, has been only outlined in this paper and a great many of the details have not been worked out yet. In particular, the suboptimization techniques required for the gaming exercise need further development. However, we believe that with a relatively small effort and short time, these mathematical models could be developed.

As far as equipment is concerned, we already stated that there is no system operating that could carry out all the required routines and input-output procedures. However, all the components are available and we see no significant difficulty in integrating existing components into a workable hardware system. More serious problems with respect to equipment are cost considerations. The financial benefits that can be obtained from on-line management systems is difficult to estimate and as a consequence it is difficult to determine how much money could be spent on equipment to create such management systems. However, aside from financial considerations, we believe that the equipment required could be manufactured within a one to two year time period.

It seems then that from the scientific and technological point of view, a management system of the type described in this paper could be created within a time period of one to two years. However, there is one further element to be considered. Traditional techniques of management control do not involve such sophisticated quantitative techniques as described in this paper. As a consequence of this, the design and implementation of advanced management systems must be accompanied by a parallel development in management philosophy. During the last few years there has been a significant shift in managerial concepts towards more sophisticated quantitative outlook. It is difficult to make a prognosis as far as management philosophy is concerned but it is difficult to believe that it will take more than two to three years to reach the appropriate management environment.

In summary, then, we estimate that it will be between two to five years before on-line management systems of the type described in this paper will become operational.

References

1. Saul Hoch, "Weapon System Programming and Control System, Mod 0 - A Demonstration Model," Operations Analysis Office, Directorate of Plans and Programs, Headquarters Air Material Command. Dec. 1960.

2. D. G. Malcom, J. H. Roseboom, C. E. Clark, W. Fazar. "Application of a Technique for Research and Development Program Evaluation," Operations Research, vol. 7 (1959), pp. 646-669.

3. "PERT/PEP Management Tool Use Grows," Aviation Week, (Nov. 28, 1960), pp. 85-91.

4. D. R. Fulkerson. "Increasing the Capacity of a Network: The Parametric Budget Problem," Management Science, vol. 5 (1959), pp. 472-483.

5. --- "A Network Flow Computation for Project Cost Curves," Management Science, vol. 7 (1961), pp. 167-178.

6. J. E. Kelley, Jr. and M. R. Walker. "Critical Path Planning and Scheduling," Proc. of the Eastern Joint Computer Conference, (1959), pp.160-173.

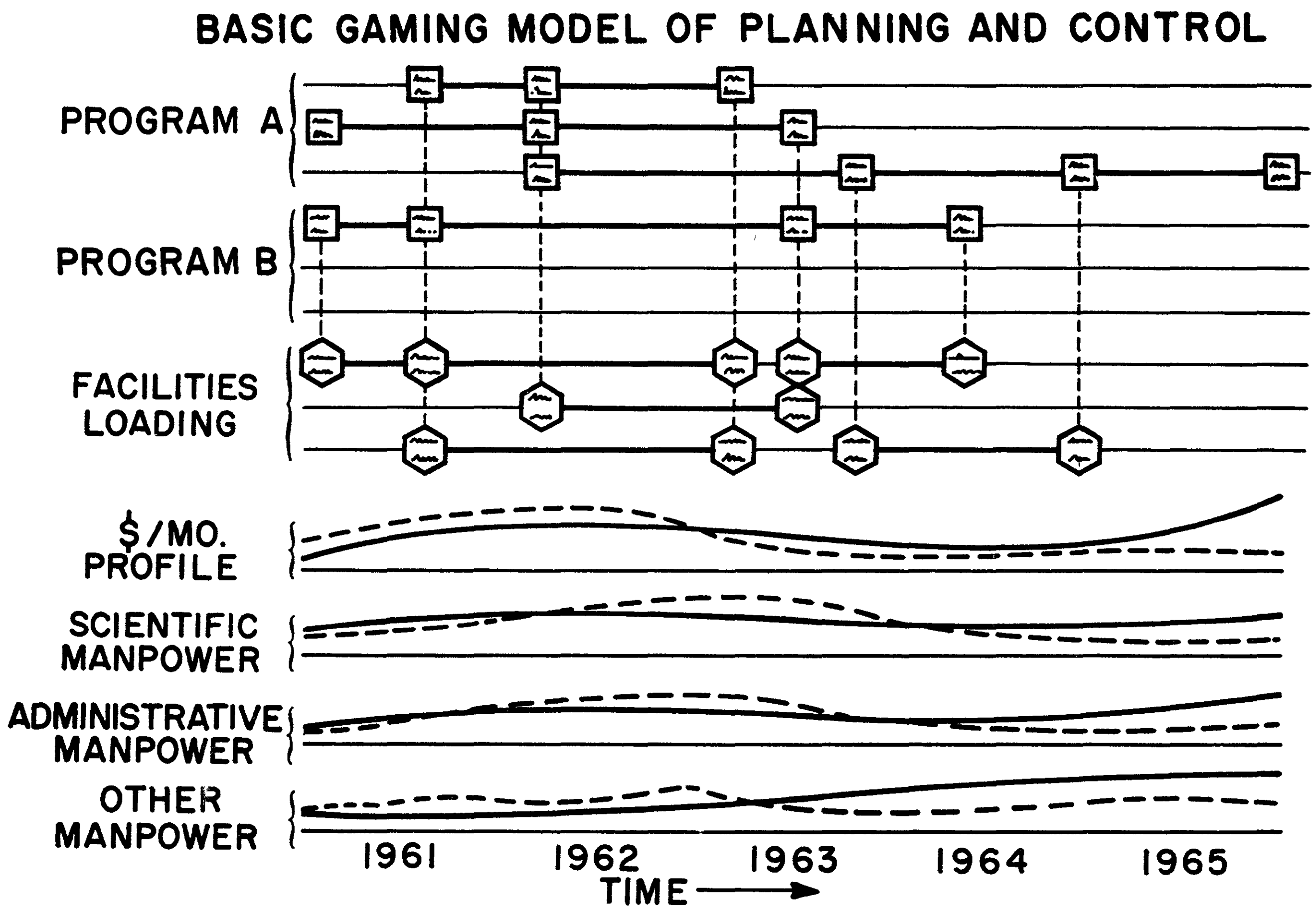


Figure 1. Basic Gaming Model

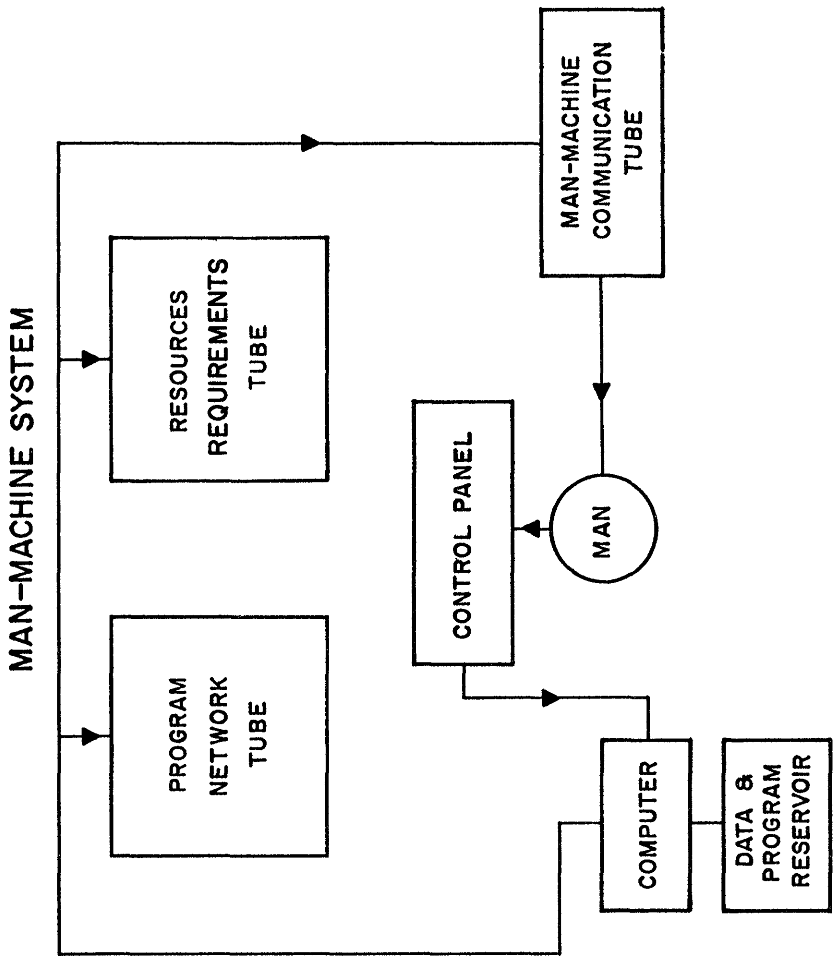


Figure 2. Man-Machine System

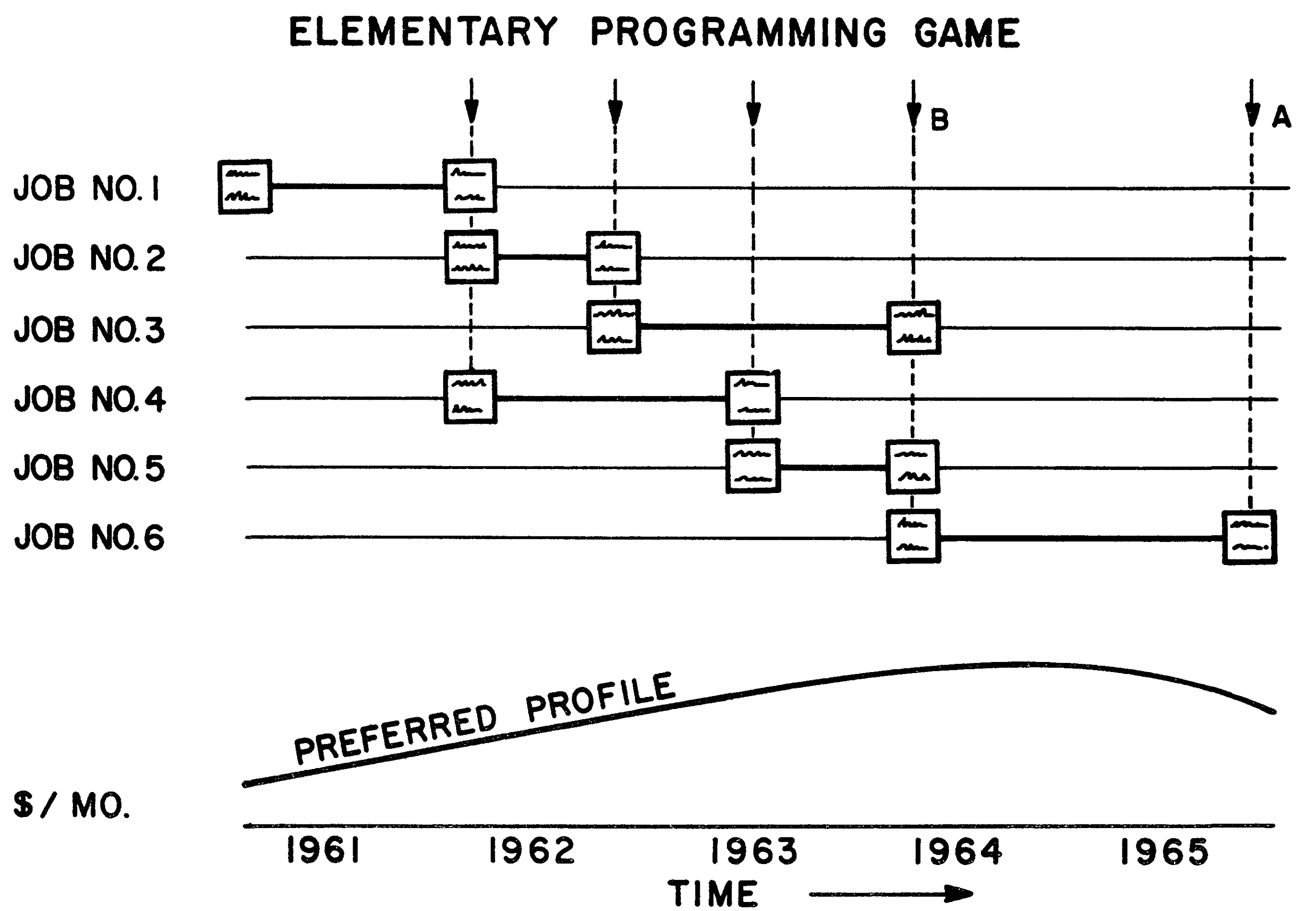


Figure 3. Elementary Programming Game

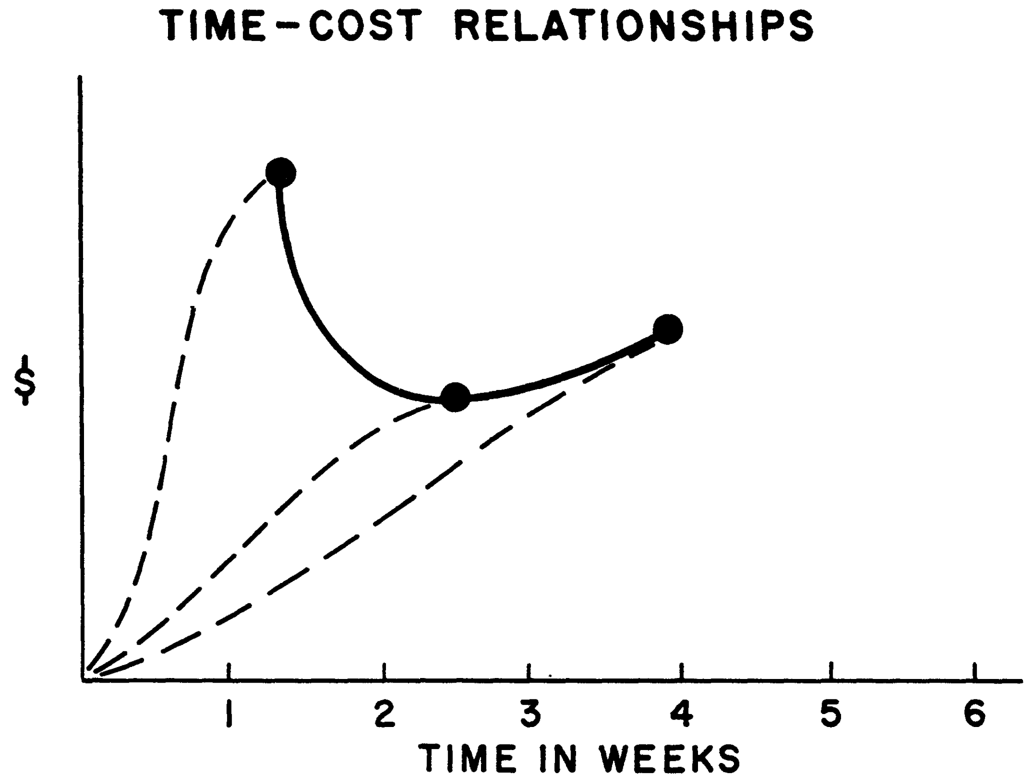


Figure 4. Time-Cost Relationships

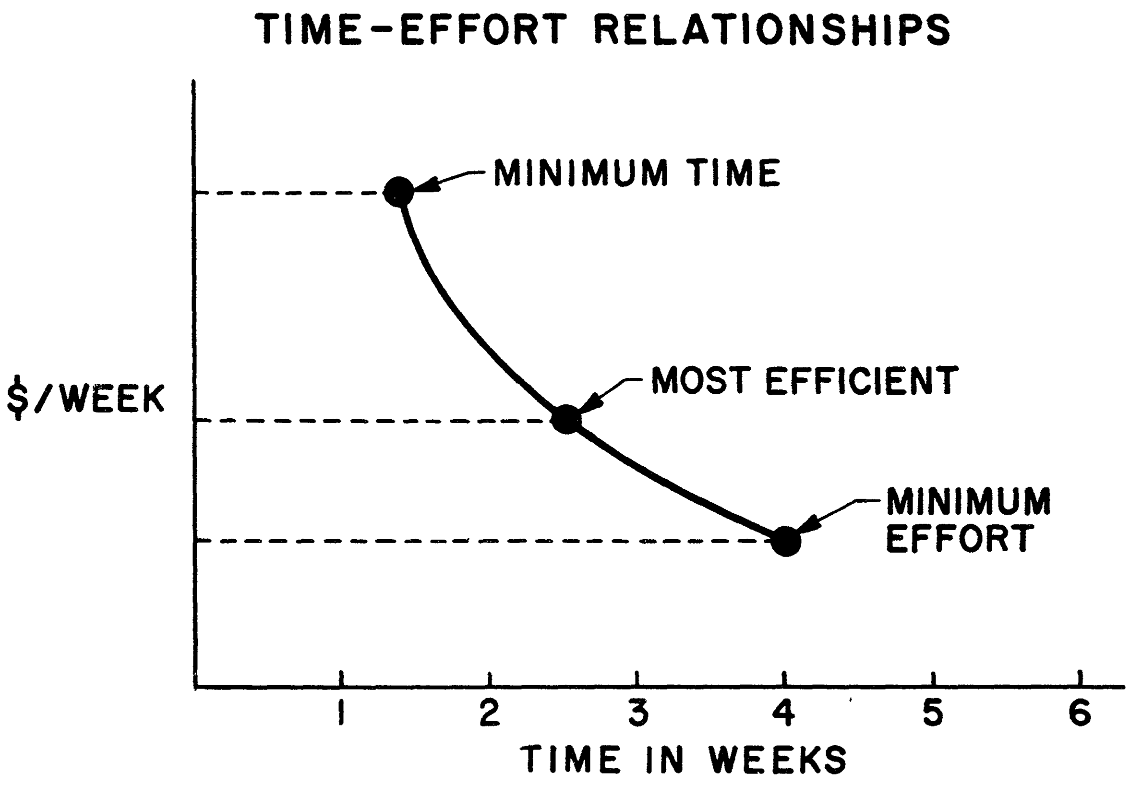


Figure 5. Time-Effort Relationships

CONCEPT OF GAMING GOALS

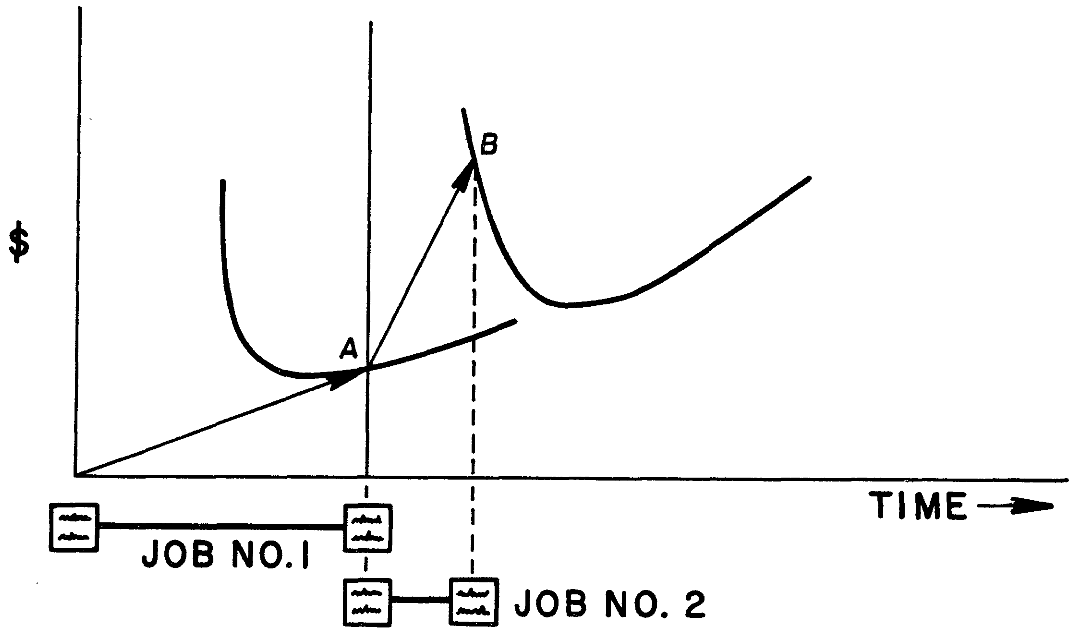


Figure 6. Concept of Gaming Goals

POSITIONING OF SLAVE GOALS

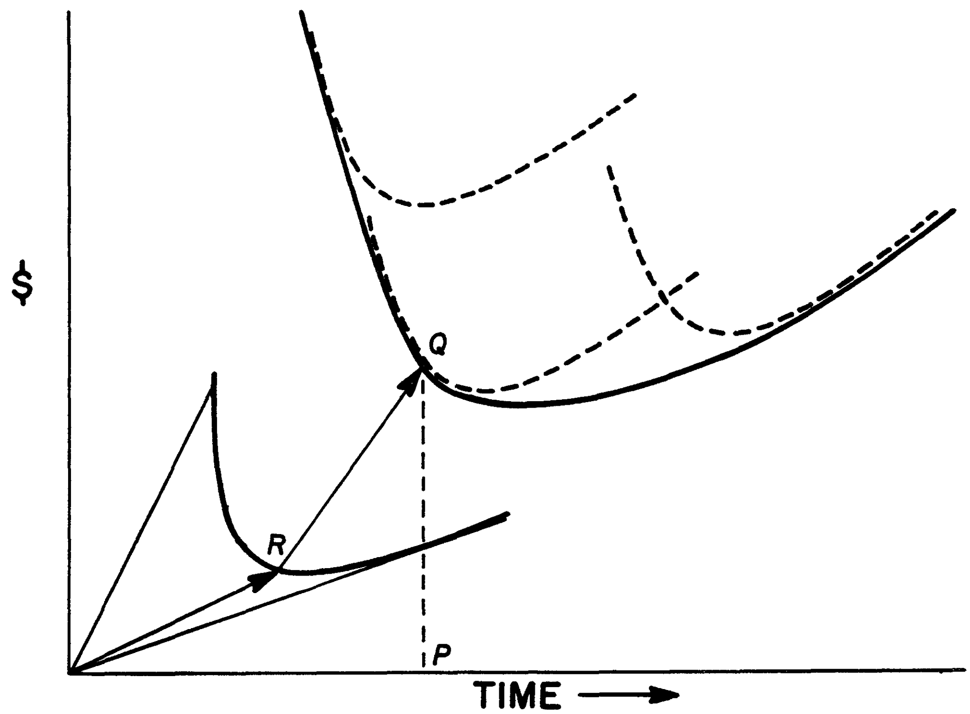


Figure 7. Positioning of Slave Goals

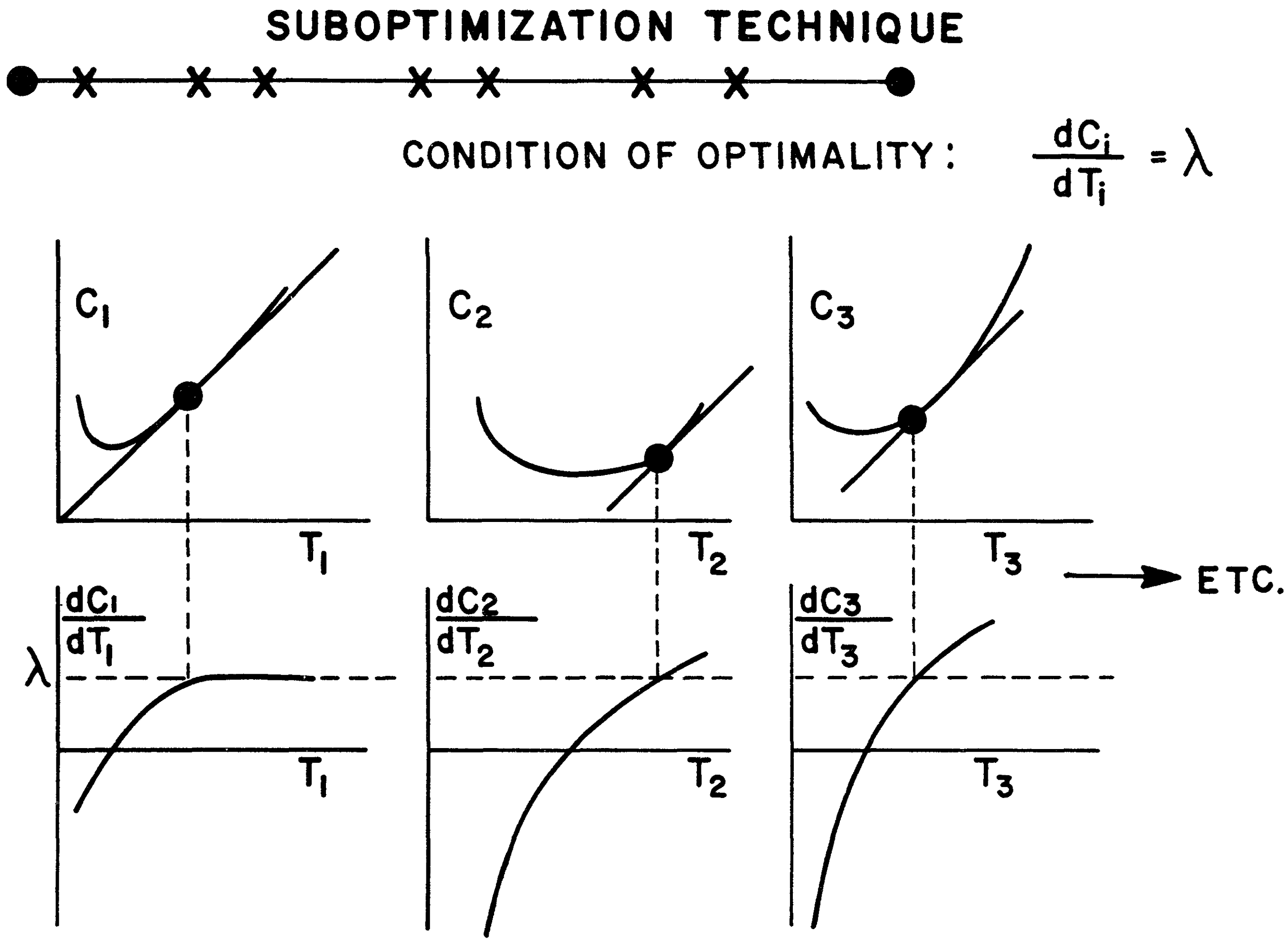


Figure 8. Suboptimization Technique

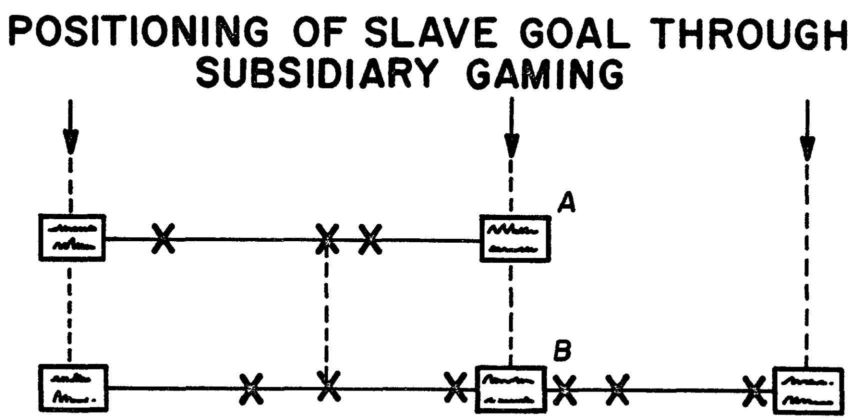


Figure 9. Subsidiary Gaming

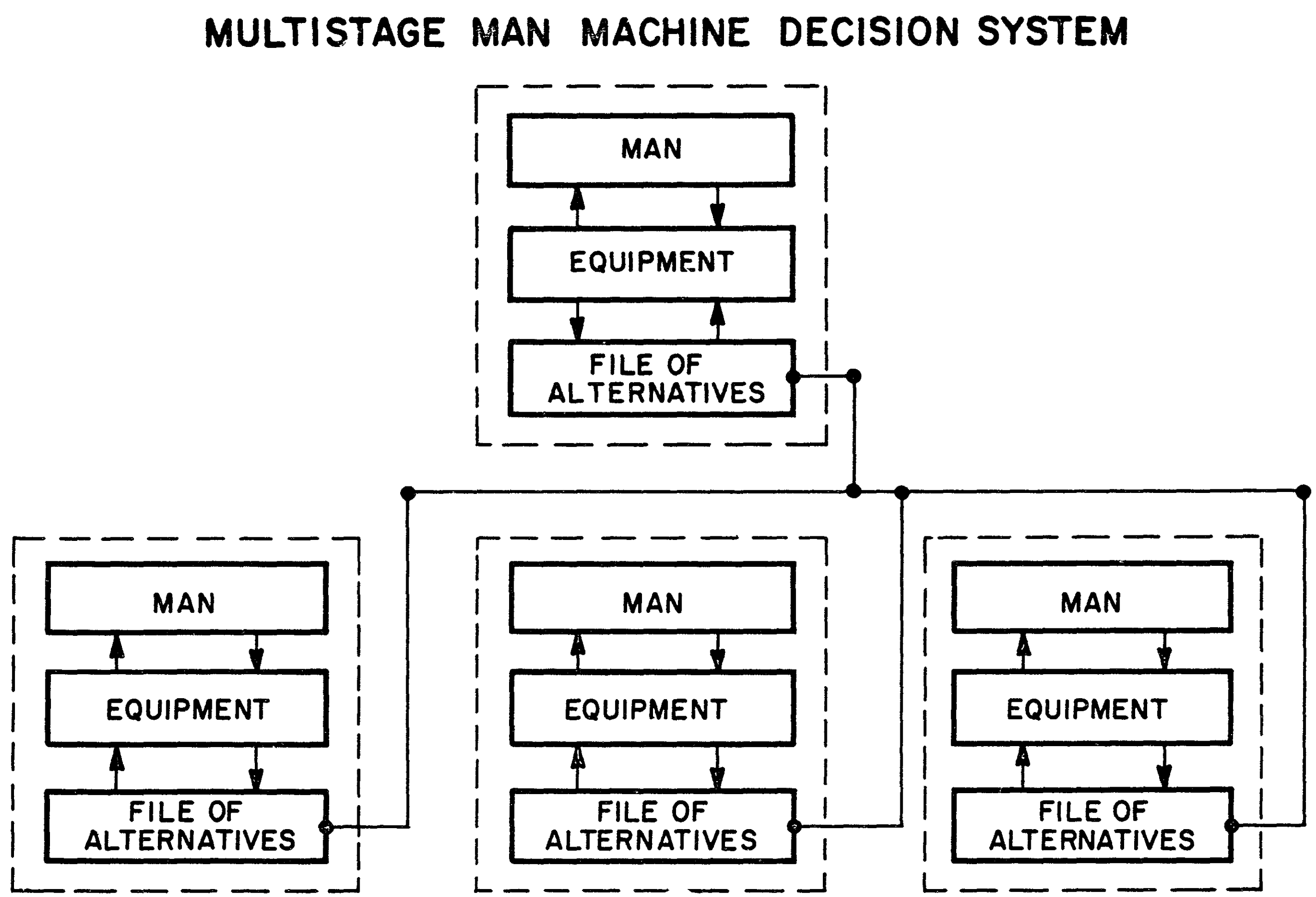


Figure 10. Multistage Gaming System

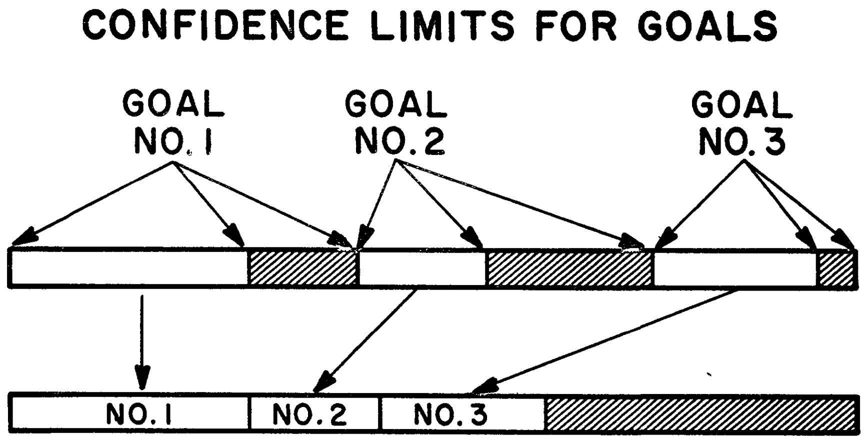
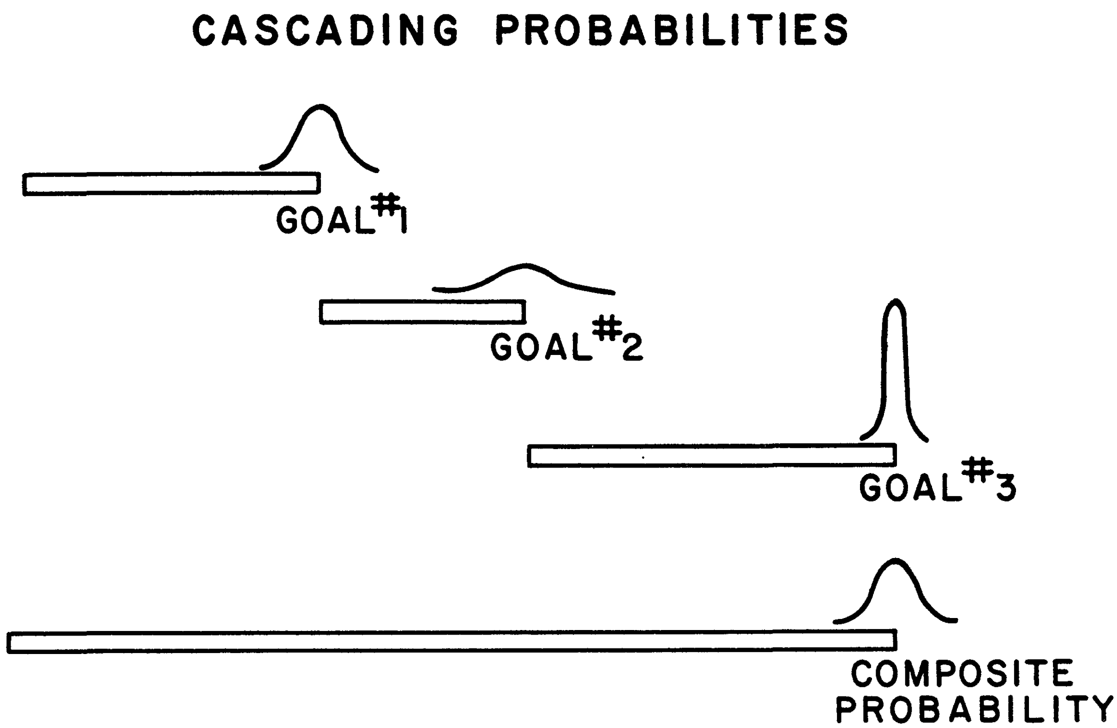


Figure 11. Confidence Limits



$$\sigma^2 = \sigma_1^2 + \sigma_2^2 + \sigma_3^2$$

Figure 12. Cascading Probabilities

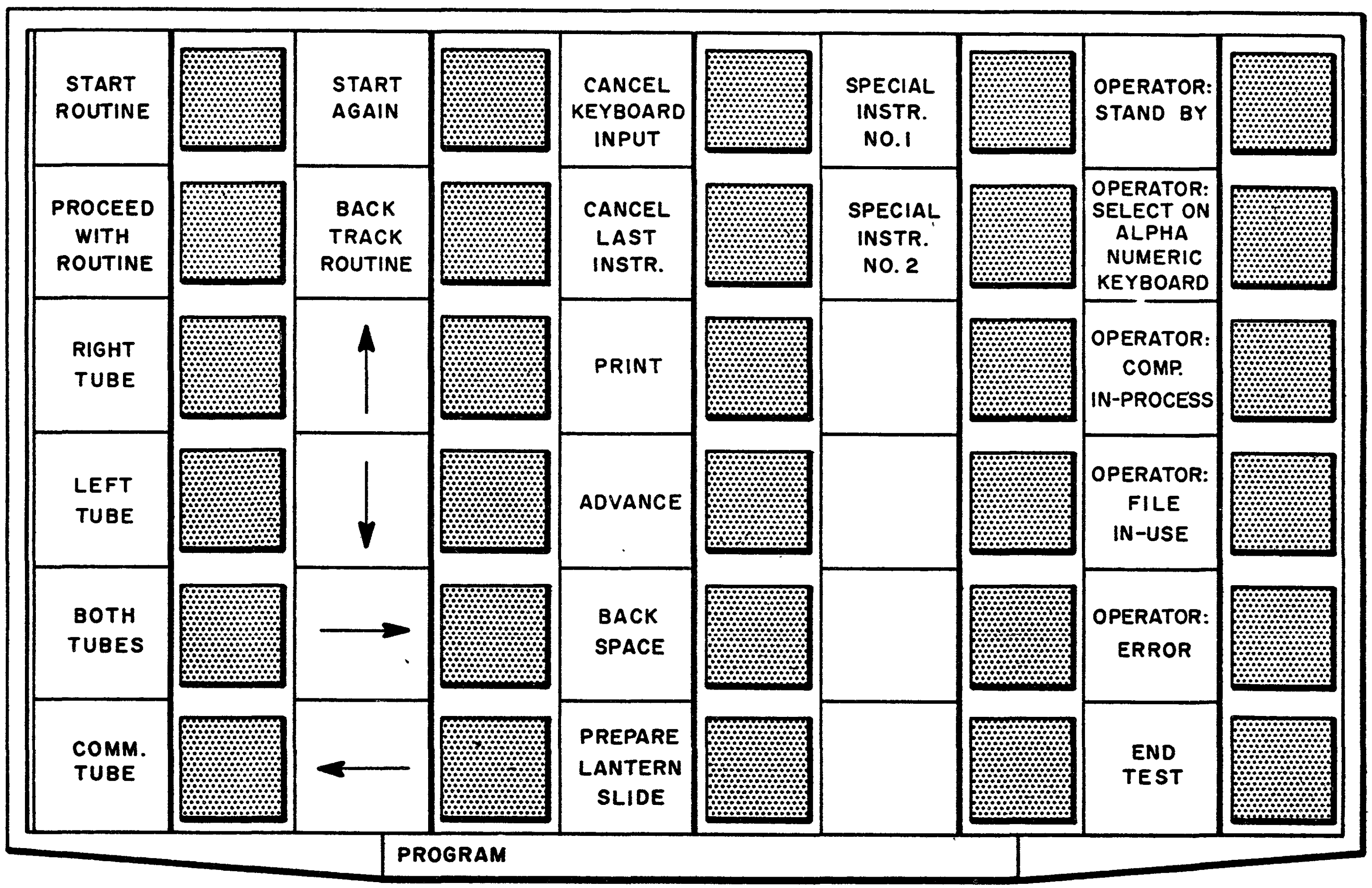


Figure 13. Control Panel

EXPENDITURES (Million Dollars)

		1961	1962	1963	1964	1965	Total 61-70
1. ATLAS	D and P	65	75	58	16		214
	Sys-conn	316	398	367	258	183	2437
2. MINUTEMAN	D and P		81	46	74	223	1245
	Sys-conn			10	39	102	1335
3. B-70	D and P			828	620	453	3115
	Sys-Conn					4	4013
4. B-52	D and P	95	25				120
	Sys-conn	1060	1149	1134	1134	1134	9931
5. B-58	D and P	58	25				83
	Sys-conn	159	324	309	309	309	2748
6. B-47	D and P						
	Sys-conn	915	741	510	197		2363
7. KC-135	D and P	163	87				250
	Sys-conn	174	230	230	230	230	2084
8. GAM-87	D and P			64	26	33	505
	Sys-conn					3	446
NON-SYSTEM		1200	1200	1200	1100	1100	
TOTAL FOR ALL WEAPONS SYSTEMS		4591	4795	5173	4367	4002	
LIMITATIONS		4600	4500				

Figure 14. Weapon Systems Expenditures

Total in Program	276	Total Delivery to Date	8		
Units per Squadron	12	Active Squadrons	0		
Go Ahead Date	May 1958	1st Alternate		2nd Alternate	3rd Alternate
Last Delivery	June 1964				

			EXPENDITURES				
			ATLAS		All Systems		
Year	To Be Delivered	Squadrons Projected	D&P	Sy. Con.	Non-Sys.	Total	Limitation
1961	49	4	65	316	1200	4591	4400
1st Alternate							
2nd							
3rd							
1962	138	11	75	398	1200	4795	
1st Alternate							
2nd							
3rd							
1963	230	19	58	367	1200	5173	
1st Alternate							
2nd							
3rd							

Figure 15. Atlas Program