ORGANIZATIONAL BEHAVIOR AND HUMAN PERFORMANCE 2, 329–374 (1967)

JUDGE: A Value-Judgment-Based Tactical Command System[1]

L. W. MILLER, R. J. KAPLAN
The RAND Corporation

AND

W. EDWARDS
University of Michigan

This paper reviews work completed on a value-judgment-based tactical air command system intended to dispatch missions from a limited supply in response to requests for immediate close air support.

A Judged Utility Decision Generator (JUDGE) assumes that value judgments can be made explicitly and in real time by appropriately trained personnel, and that decision systems should maximize expected utility. Its inputs include demand forecasts, numbers of aircraft available, and turnaround time distributions. As each request is received, JUDGE makes a dispatching decision based on the judged utility of destroying the target named in the request, kill probability data, number of sorties remaining before resupply of aircraft, and the time. In a war-game situation using experienced military subjects, JUDGE was considerably superior to a simulated current system in amount of expected utility gained. Concluding sections of the paper identify characteristics of judgment-based command systems, and discuss logical implications of such characteristics for system evaluation.

Algorithms for making decisions in real situations are usually extremely complicated. The search for such algorithms defines much of the work of operations researchers, systems analysts, and similar experts. But that search is often unsuccessful; many real decision tasks are intractable to full formal analysis. The designers of decision-making systems intended to cope with such tasks usually allocate the functions of information processing and decision making in such systems to men, presumably hoping that human intuition can deal effectively with problems too com-

[1] This research is sponsored by the United States Air Force under Project RAND —Contract No. F44620-67-C-0045—monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this paper should not be interpreted as representing the official opinion or policy of the United States Air Force.

© 1967 by Academic Press Inc.

plex for formal analysis. But in experiments that compare human performance with such very simple optimal rules as Bayes's theorem and the maximization of expected value, men are clearly second best (for reviews see Becker and McClintock, 1967; Edwards, 1954, 1961; and Peterson and Beach, 1967). There is no good reason to expect that men will perform better in real decision-making system contexts than in these simple experiments.

We believe that full mechanization of a decision task or completely intuitive performance of that task are the two extremes of decision-making system design, that intermediate possibilities exist, and that in some situations these intermediate possibilities are clearly preferable to either extreme. The essence of the intermediate possibilities we have in mind is that men, observing the real world, estimate numbers that are appropriate inputs to appropriate formal algorithms. The algorithms then generate the outputs of the systems. Such systems reverse the logic of present-day command systems. The mechanical parts of present-day command systems serve to bring the information to the decision maker, who generates the system output. We are proposing that machines should generate the system output, while men provide the inputs to those machines. This approach to decision-making system design provides the algorithm designer with kinds of inputs that he could not otherwise obtain, and thus permits him to use otherwise unavailable algorithms. It provides the human factors specialist with the possibility of dividing up the intellectual tasks that the system must perform into subtasks, allocating those that are best performed by machines to machines, and letting men perform the rest.

Two classes of systems such as we advocate can be distinguished. In one, the main purpose of the system is diagnosis: that is, the system must figure out what is happening in the world around it on the basis of the inadequate and inconclusive information available to it. Edwards (1965, 1966), has proposed a class of Probabilistic Information Processing (PIP) systems for diagnostic information processing. In a PIP, men make numerical estimates of the likelihood of each datum available to the system given each hypothesis in which the system is interested; a computer uses Bayes's theorem to synthesize these numbers into a picture of how all the hypotheses currently stand in the light of all available data. Research by Edwards (1966) and others (Schum, Goldstein, and Southard, 1966) indicates that PIPs are much more efficient information processors than men alone.

This paper is concerned with an example of the other class of systems. In these systems, the crucial problems have to do less with probabilities than with values. The system designer may have some global goal in

mind (e.g., winning the war, making a profit, doing important research). But he needs to know how well each possible outcome of a decision will serve that goal. If he can answer this question by means of a formal calculation, he should. Otherwise, it is appropriate to ask experts to estimate values or utilities. The computer combines these with probabilities into expected values, and chooses the course of action with the highest expected value.

The particular representative of this class of systems that we have chosen to study and simulate is concerned with aircraft-dispatching decisions in a tactical context. Most of the details of the system respond to particular characteristics of that context. But the essence of our idea is that men make value judgments and a computer uses them and an algorithm that maximizes expected utility to make decisions. That strategy for system design is extremely general.

A tactical commander must decide whether to grant a particular request for immediate close air support, or to deny it on the grounds that he must conserve resources to fulfill a possible later request of greater importance. We have assumed that resources (missions available for close air support) are seriously limited relative to demands and that there is a wide range of importance in the requests. Whether or not these assumptions characterize current Air Force operations, they do describe circumstances in which a tactical command system should be prepared to function. Since our proposed system is based on judged values or utilities, we have called it a Judged Utility Decision GEnerator (JUDGE).

This paper begins by summarizing the aspects of the TAC control problem that are relevant to our proposed system, and the present solution to that problem. Next it presents a set of formal rules for making mission-dispatching decisions. These rules assume that numerical measures are available for evaluating possible mission outcomes. They first translate values of mission outcomes into values of the missions themselves; a solution to the problem of predicting the effectiveness of missions is crucial to that translation. Next, the rules specify a procedure for making dispatching decisions based on mission values; the procedure responds to fluctuations in arrival rates of requests having various values.

The manner in which the numbers representing outcome values are obtained makes no difference to the formal rules for using them, so long as they have the usual numerical properties of measurements. But we assume that expert judges should ordinarily supply such numbers. We present a small experiment comparing numbers obtained by several judgmental procedures; the data indicate that one procedure is clearly preferable to the others, and that the numbers so obtained compare well with "correct" numbers (available in this experiment but not in TAC control

system settings). Thereafter we discuss a field study which incorporated this procedure and was the first test of a JUDGE system. Experienced Air Force officer subjects made value judgments about mission outcomes in a verbally-described, realistic, limited-war setting.

The designs both of JUDGE and of procedures for evaluating and refining it illustrate some general principles concerning the design and evaluation of judgment-based command and design systems. Since our views on this topic seem radical to us, and since we know of no published presentation of similar views, we conclude this paper with a rather extensive abstract discussion of the problem and our opinions about its solutions.

THE TACTICAL CONTROL PROBLEM

The tactical commander has numerous operational missions to perform including air defense, counter air, interdiction, assault airlift, reconnaissance, and close air support. Close air support (CAS) missions use tactical aircraft in direct and immediate response to requests for support from ground forces. A special agency, the Direct Air Support Center (DASC), controls strike aircraft on CAS missions. The DASC controls aircraft assigned to it by the Fragmentary Operations Order (frag order) written each day by the command organization of the Tactical Air Commander. The DASC is collocated with the Tactical Operations Center (TOC) of the Army, to facilitate Air Force-Army coordination.

Requests are made from the field by the Forward Air Controller (FAC) over the Tactical Air Request Net. The FAC is an Air Force fighter pilot stationed with the ground troops. He knows the capability of the aircraft and of the weapons they carry, and can advise the ground commander about feasibilities of missions or can suggest the use of air in cases where the ground commander might not otherwise consider it. The FAC transmits Army requests for CAS to the DASC; these requests are approved or disapproved by each echelon of Army command between the requester and the TOC. Each of these intermediate Army commands has an Air Liaison Officer to help with the approval decisions.

The DASC has the authority to communicate with air bases and send strike aircraft on missions specified by an Army-approved request. Whether the final decision to fulfill a request is made primarily by the Air Force or by the Army depends mostly on the personalities of the individuals involved. Doctrine says that when the request gets to the DASC, it has become a requirement and must be satisfied if it is within that organization's capability. If all requests for CAS could be satisfied immediately, the DASC's main function would be status recording and communications; it would be necessary only to send the aircraft off and

follow them to make sure that they go to the target and do what they are supposed to do. Forces must be prepared to function in situations where many more demands are placed on its resources than can be satisfied. In that case, a control system for CAS must consider every request as it arrives and decide whether it will be fulfilled or whether planes will be withheld for use in response to a possibly more important future request.

This sketch of the problem highlights two difficult tasks that a tactical control system should perform. The first is evaluating the relative importance of each request; the second is predicting the arrival of future, more important requests. If all requests are equal or nearly equal in importance, then a first-come-first-served policy will be as good as any other, and no allocation problem exists. If requests vary in importance, but nothing whatever can be said about the likelihood of future, more important requests, then although first-come-first-served is not a good policy, no better one seems feasible. Note, however, that the value problem has logical priority over the prediction problem. What must be predicted is not merely the arrival of future requests, but the arrival of future requests more important than the one now being considered; such a prediction is impossible unless the importance of each request can be measured at least roughly.

In current doctrine, the importance of each request is, in principle, evaluated by means of a priority number assigned to the request by the requester. Requests transmitted to the DASC are seldom assigned less than priority "1" by their originators. The informal rules specify that you should not ask for air support unless you really need it; if you do need it, you want to be sure that your request is attended to. This procedure, of course, makes the priority assignment system meaningless.

A closely related problem is the self-adaptation of demand for CAS to the supply of airplanes. Requesters all listen to the network on which requests are made; if no planes are available at a given time, no requests are made. Apparently, informal social pressures discourage unfulfillable requests. This phenomenon would not occur, we believe, in any environment in which resources were less overwhelmingly abundant relative to the need for them than is the case in typical exercises.

Even in the present abundant environment, this self-adaptive mechanism has three regrettable features. A larger supply of requests would provide more intelligence information to higher headquarters. It would permit higher level Air Force and Army commanders, rather than requesters in the field, to evaluate relative importance of potential requests. And it would reduce the chances that the man whose urgent need develops relatively late in the day, after almost all of the day's missions have

been flown or assigned, will have to do without needed CAS. So the essence of our proposal is that requests should be encouraged and that procedures based on value judgments should be used to determine which requests should be fulfilled and which should not.

A plane and pilot may fly several CAS missions in a day. How many such missions they can fly depends on the nature of each mission and on the turnaround time for the plane, which may vary from 30 minutes to many days. At present, the frag order specifies that the DASC will have a certain number of sorties available at the beginning of the day and additional sorties available at specified times thereafter. Air bases are responsible for turning their planes and pilots around sufficiently fast to provide the missions specified in the frag order. This rigid procedure does not allow unplanned variations in the number and nature of missions assigned early in the day to influence the number of missions available later—though in practice the system operates more flexibly than the formal description implies. A tactical control system should be designed to consider explicitly how the number and nature of early missions, whether anticipated or not, influence later availability of aircraft when DASC is deciding on early mission assignments.

THE LOGICAL DESIGN OF JUDGE

This section presents a set of formal rules that accept as inputs statements about the values of various possible mission outcomes, past experience with incidences of requests having various values, and plane availability. Outputs are dispatching decisions and predictions about future plane availability. Empirical questions about the availability of inputs are not a concern here. We assume that values will in general be made available by exploiting expert human judgment, that past experience with requests will be available, and that information of a specified nature about plane availability will be available. The following section examines procedures for obtaining value judgments.

In the following discussion, we assume that value should be maximized. For us, this assumption has somewhat the status of an axiom; the philosophical discussion at the end of this paper examines it and its implications to some extent. We also assume that under conditions of uncertainty, expected value should be maximized. The notion of expected value maximization as a criterion of optimal risky decision-making has far too long a history to summarize here. For an elementary discussion, see Edwards (1954). For situations in which the so-called gambler's ruin problem does not arise, we know of no alternative rule for risky decision-making that deserves consideration.

JUDGE's inputs include values of targets that might be attacked,

probabilities that attacks will destroy their targets, and number of planes available. To make dispatching decisions, it considers the value of sending one or more planes and thus of perhaps destroying the target, and the cost resulting from the fact that if the planes are sent against this target, they will not be available to send against other targets later. In order to evaluate this cost of later unavailability, JUDGE must consider future requests and future plane availability. JUDGE's outputs are forecasts concerning future plane availability and dispatching decisions. The computations that transform its inputs into its outputs are complicated. A technical but nonsymbolic summary of their nature follows.

To make computation possible, JUDGE divides time into a sequence of discrete periods called horizons. (It makes no difference to the formalities how long a horizon is; we tend to think of it as 1 or 2 hours.) The crucial point about horizons is that new planes, or planes turned around after earlier missions, become available for use only at the beginning of each horizon; the supply of planes within a horizon is taken as fixed. This oversimplification permits a computationally and intellectually convenient division of the problem into two parts: the dispatching problem and the planning problem. The dispatching problem is exclusively concerned with the decisions made within one horizon regarding aircraft assignment; the planning problem is concerned with both the process of planning a day's activities (important, for example, for maintenance planning purposes) and the task of supplying to the dispatching algorithm some numbers that tie together the several horizons that make up a day.

Within one horizon, the only decision to be made is how many planes, if any, to send in response to each request. Each request comes with two kinds of information attached: a value for destroying the target, and a number, or perhaps a function, that represents the effectiveness of planes against the target. The dispatching rule knows how many planes it has available for use in the remainder of the horizon, and it has a basis for estimating the arrival rates of requests with various values.

To use this information, the dispatching rule calculates an expected gain or loss for each possible number of planes that might be sent. The expected gain of sending no planes is zero. The expected gain of sending one or more depends on the target's value, the probability that the planes being sent will kill the target, and the cost of the lost opportunity to use those planes against some later target. Calculating this opportunity loss is complicated, and depends on the forecasted arrival rates for requests of various values. Among the various available dispatching decisions, the one with the highest expected value is selected. Increases in mission value, in supply of planes, and approach of the end of the horizon all increase the willingness of the system to dispatch a plane.

The dispatching rule operates within a horizon; the planning problem is concerned with the interrelations among horizons. The output of the planning algorithm is a forecast of the number of sorties to be available in each horizon. This forecast permits the dispatching rule to find out the value of planes left over at the end of a horizon. They are obviously valuable since they can be used during the next horizon. But they obviously lose value at the horizon boundary, since immediately before the boundary they are all the resources the system has, while after it the system has been supplied with a specified additional number of planes. Their value in the new horizon is simply the expected value of the additional missions they make possible in the next horizon, taking into account the fact that the end of the next horizon might conceivably also find planes sitting on the ground unused. The expected number of planes to be resupplied at the beginning of each horizon must be calculated, and this depends on the dispatching decisions made in previous horizons. The computations begin with a list of numbers, which in essence predicts the number of sorties to be flown during each horizon. This prediction reflects actual past dispatching decisions and forecasts future ones. This prediction is updated periodically by a linear programming procedure to reflect changes based on actual experience with requests and actual previous dispatching decisions. The updating will probably be done once per horizon, depending on how much conditions turn out to differ from those planned for, how much computer time is available for a rather demanding calculation, and so on.

The procedure taken as a whole is suboptimal in a number of ways, and depends on a number of special assumptions. However, it should be a good first approximation. As it becomes desirable to improve on it, it will be possible to change parts of the procedure piecemeal, or to add other features not now included.

The Dispatching Rule

The dispatching portion of JUDGE is developed under a set of assumptions concerning the resources, admissible decisions, request process, target values, and mission effectiveness measures. These assumptions are described in the following paragraphs.

Resources. The aircraft to be used over a horizon are all of the same type and loading, and are completely interchangeable. Situations involving collections of nonhomogeneous aircraft, in which the substitutability of one type for another depends on the target, would require the state variable representing inventory of remaining sorties to be multidimensional.

Admissible decisions. The requests JUDGE handles are of the "as soon

as possible" variety, and it is assumed that a dispatching decision is made immediately upon receipt of each request. The only admissible decisions consist of assigning to the mission a number of sorties, between zero and the number of remaining sorties. Collecting requests or delaying action on marginal requests is not permitted.

The request process. The arrival of requests over time is assumed governed by a Poisson process with a known rate of λ requests per hour. The substance of this assumption is that over a very small time interval of length h, the probability is: $(1 - \lambda h) + o(h)$ that there will be no requests; $\lambda h + o(h)$ that there will be one request; and negligible that there will be two or more requests. (The symbol $o(h)$ means a quantity very small relative to h.) These probabilities are independent of any events that may occur outside the h-interval under consideration. The request rate will be taken as a constant over any particular horizon, although allowing λ to vary over the horizon could easily be incorporated into the calculation if dependence of λ on time could be estimated.

In the absence of evidence, this Poisson hypothesis provides a plausible assumption regarding the arrival of requests. Certain mathematical results indicate that the Poisson process is an appropriate model if there are many potential requesters acting independently. (See, for example, Cox and Smith, 1954.)

Mission value functions. Associated with each request is a mission value function that specifies an immediate reward for each act that could be taken in response to the request. Let v be the judged value of destroying the target, and let $\eta(x)$ be the extent to which a mission of x sorties can be expected to achieve the desired effect. The mission value function, $u(x)$, is taken to be the product:

$$u(x) = v\eta(x) \qquad \text{for} \qquad (x = 0, 1, 2, \ldots).$$

For a target occupying a small area that could be destroyed by a single attacking aircraft, it is reasonable to propose that

$$\eta(x) = 1 - (1 - p)^x,$$

where p is the probability that one aircraft will be successful. Then the formula above represents the probability that not all x airplanes miss the target. The parameter, p, depends on many factors relating to the target, aircraft, weapons, and tactics, but current technology is sufficiently well developed to provide reasonable values. In an implementation of JUDGE, it is possible that p might also be obtained through the judgment of qualified personnel.

As a result of this form for $\eta(x)$, the mission value functions have the desirable properties that the marginal utility of sending an additional

plane decreases for increasing x, and that the reward for not responding to a request is zero.

With targets of significant area, say troops dispersed over a region the size of a football field, the effectiveness function should reflect the degree to which the target is covered by lethal ordnance. By making some coarse assumptions, the same simple form for $\eta(x)$ can be rationalized. Let T denote the target area, let L represent the lethal area of the weapons a single aircraft deposits on the target, and let p_1 be the probability that a plane successfully hits the target. If we subdivide the target into areas of size L and assume that a plane hitting the target hits any particular subdivision with probability L/T, then the probability that any subdivision is killed is $p_1(\mathrm{L}/\mathrm{T})$. The probability that any particular subdivision is killed by an attack of x aircraft is then given by

$$\eta(x) = 1 - (1 - p)^x,$$

where

$$p = p_1 \frac{\mathrm{L}}{\mathrm{T}},$$

and this is also the total expected portion of the target destroyed.

Sensitivity to the number of sorties can be varied by introducing a second parameter and writing the mission effectiveness formulas as

$$\eta(x) = 1 - (1 - p)^{x^\alpha}.$$

We have explored hypothetical effectiveness functions generated this way with α in the range of 0.8–1.4. Although we offer no interpretation for the extra parameter, a function of this form should be capable of fitting a wide variety of effectiveness functions that might be derived from very detailed analyses. For simplicity we shall assume that the mission effectiveness functions are specified by the choice of a single parameter, p.

In order to keep in mind that the mission value function depends on both the estimated target value and the parameter of the effectiveness function, a mission value function will be indicated by $u(x\colon v, p)$.

In developing the dispatching rule, the target values and effectiveness parameters will be treated as random variables. Since these variables may very well be correlated, we shall indicate a marginal distribution for target values and a conditional distribution for effectiveness parameters. Denoting the two random variables by V and P, respectively, define:

$$F(v) = \mathrm{Prob}(V \leq v),$$

and

$$G(p|v) = \mathrm{Prob}(P \leq p|V = v).$$

These distributions are assumed known; the forecasting problem will be discussed in the following subsection.

Derivation of the Dispatching Rule

Within a horizon, the state of the system may be described by specifying the number, n, of remaining sorties available for dispatching and the amount of time, t, left until the horizon is over. The cost associated with dispatching sorties is achieved by attributing a value, $W_n(t)$, to being in the state (n, t). This value is a measure of the total expected utility associated with the dispatching decisions to be made over the remaining t hours of the horizon when there are n sorties that can be used. Then the price of x sorties in the state (n, t) is $W_n(t) - W_{n-x}(t)$.

If a request having attributes v and p is received when the state is (n, t), the best decision is to choose the value of x, $(x = 0, 1, \ldots, n)$, that yields the maximum difference between reward and cost, the value of x that maximizes

$$u(x: v,p) + W_{n-x}(t) - W_n(t).$$

Since the decision does not affect the last term in the above expression, it may be dropped. The quantity

$$\max_{0 \leq x \leq n} [u(x: v,p) + W_{n-x}(t)]$$

represents the sum of the immediate reward and the value of the state resulting from the best decision in the event that a request with characteristics v and p is received when the system is in state (n, t).

Under the notion that the target value and effectiveness functions associated with a request are selected randomly, the expected value of the reward, plus the value of the state resulting from the action taken when a request arrives at state (n, t), is

$$\int_v \int_p \max_{0 \leq x \leq n} [u(x: v,p) + W_{n-x}(t)]\, dF(v)\, dG(p|v).$$

Taking h to be a very small number, suppose that there are n sorties available in a horizon that ends in $t + h$ hours. Given that a request arrives in the next h hours, the expression above represents the value of state $(n, t + h)$. On the other hand, if there are no requests in the small time interval, the sum of the expected rewards to be earned over the $t + h$ hours is the same as that for the slightly shorter horizon of length t hours, and then $W_n(t + h) = W_n(t)$. By assumption, the probabilities of a request and no request in the small interval are, respectively, λh and $(1 - \lambda h)$. Weighting the two expressions for $W_n(t + h)$ by the corresponding probabilities gives

$$W_n(t+h) = (1-\lambda h)W_n(t) + \lambda h \int_v \int_p \max_{0 \le x \le n} [u(x: v,p) + W_{n-x}(t)]\, dF(v)\, dG(p|v).$$

Following the standard procedure of subtracting $W_n(t)$ from both sides of the equation, dividing by h, and then taking limits as h becomes small, we obtain the differential equation:

$$\frac{dW_n(t)}{dt} + \lambda W_n(t) = \lambda \int \int_p \max_{0 \le x \le n} [u(x: v,p) + W_{n-x}(t)]\, dF(v)\, dG(p|v),$$

which will be referred to as the "value equation." For any horizon, a series of these equations are solved recursively starting with $n = 1$ and going up to the number of sorties allocated for use in the horizon. The recursive solution is necessary because for any value of n, the solution depends on the $W_m(t)$ functions for $m < n$.

The particular solutions depend on a set of boundary conditions as follows: the notion that the utility of no sorties available at any time is zero is embodied in the boundary condition

$$W_0(t) = 0 \qquad \text{for} \qquad (t \ge 0).$$

In addition, a boundary condition is needed for each value of n considered to specify the value of having n sorties left over when the horizon ends. For the last horizon, leftover sorties are useless so that in this special case it would be reasonable to impose the conditions:

$$W_n(0) = 0 \qquad \text{for} \qquad (n = 1, 2, \ldots).$$

For other horizons, boundary values should be related to the marginal value of having additional sorties available in the following horizon. The use of these boundary values to connect one horizon with the following one will be discussed under planning.

The dispatching rule itself is a byproduct obtained in solving the value equations, and consists of a table of the optimizing x's as functions of the parameters v, p, n, and t. However, in an implementation of JUDGE, it would be more practical to retain just the solutions, $W_n(t)$, as a two-dimensional table with some suitable grid for the time parameter. Then the optimizing decisions for any v and p combination could be readily computed as required.

A fruitful way of characterizing the dispatching rule is to define $R_x(n, t)$ as the set of points (v, p) such that the optimal choice is to dispatch x sorties for a request with attributes v and p received when the system state is (n, t).

A boundary point between $R_{x-1}(n, t)$ and $R_x(n, t)$ may be determined by fixing p and solving for the value of v that yields the same total of immediate expected reward plus expected value of the resulting state when either $x - 1$ or x sorties are dispatched. That is, for a particular mission effectiveness function $\eta(x)$ (which is determined by the choice of p) solve for the value of v that satisfies

$$v\eta(x - 1) + W_{n-x+1}(t) = v\eta(x) + W_{n-x}(t).$$

The solution, denoted by v_x is

$$v_x = \frac{W_{n-x+1}(t) - W_{n-x}(t)}{\eta(x) - \eta(x - 1)}.$$

For $v < v_x$, it is better to send $x - 1$ sorties; for $v > v_x$, sending x sorties is the preferred action. That is, v_x is the minimum value of v for which x sorties would be dispatched for a request whose mission success parameter is p and the state is (n, t).

Some care must be taken because the sequence of v_x's obtained in this way may not be increasing. It could happen that $v_{x+1} < v_x$. Such a reversal would indicate that there is no value of v such that dispatching x sorties is the optimal action. Then the choice is between $x - 1$ and $x + 1$, and a new value of v_{x+1} must be determined as

$$v_{x+1} = \frac{W_{n-x+1}(t) - W_{n-x-1}(t)}{\eta(x + 1) - \eta(x - 1)}.$$

Now it must be checked that $v_{x+1} > v_{x-1}$. If not, $x - 1$ sorties would never be dispatched, and v_{x+1} must be recomputed by comparing the results of dispatching $x - 2$ and $x + 1$ sorties, provided that $x \geqslant 2$. This checking and recomputing process continues until either there are no more reversals or we have found the point on the boundary of $R_0(n, t)$ and $R_{x+1}(n, t)$.

To illustrate how the sets $R_x(n, t)$ might look on the (v, p) plane, Fig. 1 was constructed for the mission value function $u(x: v, p) = v(1 - (1 - p)^x)$. The values of $W_n(t)$ were taken from a numerical example whose solutions to the value equations are shown in Fig. 2. To generate the curves of Fig. 1, n was chosen to be 20, while t was set at 2.0 at the end of horizon 1. It was assumed that only even numbers of sorties would be sent on any mission, so that for any p, v_x was calculated from

$$v_x = \frac{W_{22-x}(t) - W_{20-x}(t)}{(1 - p)^{x-2} - (1 - p)^x} \qquad \text{for } x = 2, 4, \ldots, 10.$$

These curves reflect the notion that the higher the value of the request, for a fixed parameter, p, the more sorties we are willing to expend in an effort to achieve success. The general U-shape of the regions in Fig. 1

are attributable to the behavior of $\eta(x)$ as a function of p. For missions of constant v, the additional expected reward for dispatching an additional aircraft is proportional to $\eta(x+1) - \eta(x)$. As a function of p, this difference increases for small values of p, reaches a maximum, depending on x, and then decreases. Suppose that the optimum action were to send x sorties on a mission with value v and some small success parameter, p. If p were a bit larger, the additional reward for sending another

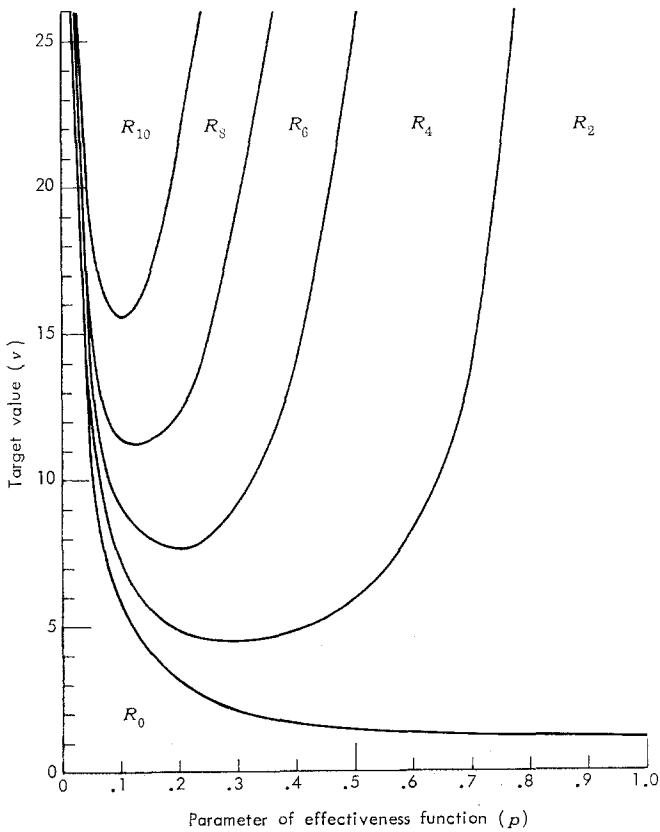


FIG. 1. Typical dispatching decisions for fixed state (n, t). For a combination of (v, p) ϵR_x, dispatch x aircraft.

aircraft might be great enough to justify another, according to our expected value criterion. But if p is large, the improvement in reward to be gained by another aircraft is insensitive to changes in p. In the limiting case where $p = 1$, there would never be any reason for dispatching more than one aircraft.

If n were smaller or t were larger, the region $R_x(n, t)$ would be moved upward, because under these the price of sorties would be higher.

The value equation may be written in terms of the regions, $R_x(n, t)$, as

$$\frac{dW_n(t)}{dt} + \lambda \left[1 - \iint\limits_{\substack{(v,p)\epsilon \\ R_0(n,j)}} dF(v)\, dG(p|v) \right] W_n(t)$$

$$= \lambda \sum_{x=1}^{n} \iint\limits_{\substack{(v,p)\epsilon \\ R_x(n,t)}} [u(x: v,p) + W_{n-x}(t)]\, dF(v)\, dG(p|v).$$

Approximate Solutions to the Value Equation

Even with the simplest assumptions about the forms of u, F, and G, it would be difficult to obtain exact solutions to the value equation because the coefficient of $W_n(t)$ and the right-hand side are complicated functions of t. A simple and satisfactory solution to the problem is available by approximating the true solutions with step functions.

To compute approximate solutions, the time horizon is divided into increments of length Δ, and approximate values of $W_n(t)$ are determined at the points Δ, 2Δ, 3Δ, . . . , under the assumption that $W_n(t)$ is constant over the half-open interval $[j\Delta, (j+1)\Delta]$. Let the approximating value for $W_n(t)$ for $t \epsilon [j\Delta, (j+1)\Delta]$ be denoted by $\bar{W}_n(j\Delta)$.

In carrying out the computations, one would start with $j = 1$ and calculate $\bar{W}_m(\Delta)$ for each m from 1 up to the maximum value desired. Then j would be stepped and the process repeated, calculating the $\bar{W}_m(2\Delta)$ values, etc.

Consider the computation of $\bar{W}_n[(j+1)\Delta]$. The quantities, $\bar{W}_m(j\Delta)$, $(m = 0, 1, \ldots, n)$ have already been obtained and are stored. Since the $\bar{W}_m(t)$'s are taken to be constant over the Δ interval, the decision rule is constant over the interval also. The sets, $\bar{R}_x(n, j)$, of points, (v, p), such that x sorties would be dispatched may be determined. For notational convenience define:

$$\alpha = \lambda \left[1 - \iint\limits_{\substack{(v,p)\epsilon \\ \bar{R}_0(n,j)}} dF(v)\, dG(p|v) \right],$$

$$\beta = \lambda \sum_{x=1}^{n} \iint\limits_{\substack{(v,p)\epsilon \\ \bar{R}_x(n,j)}} [u(x: v,p) + \bar{W}_{n-x}(j\Delta)]\, dF(v)\, dG(p|v).$$

It is understood that α and β are specific for particular n and j. They may be evaluated numerically by placing grids over the domains of v and p.

According to the approximation scheme, they too are constant over the Δ interval, and the approximate form for the value equation is

$$\frac{d\bar{W}_n(t)}{dt} + \alpha\bar{W}_n(t) = \beta.$$

The solution to this equation is

$$\bar{W}_n(t) = ce^{-\alpha t} + \frac{\beta}{\alpha},$$

where c is an arbitrary constant. The constant is evaluated from the boundary condition supplied by the known value of $\bar{W}_n(j\Delta)$. Thus,

$$c = \left[\bar{W}_n(j\Delta) - \frac{\beta}{\alpha}\right] e^{\alpha t\Delta},$$

and $\bar{W}_n[(j+1)\Delta]$ is given by the recursive relationship:

$$\bar{W}_n[(j+1)\Delta] = \left[\bar{W}_n(j\Delta) - \frac{\beta}{\alpha}\right] e^{-\alpha\Delta} + \frac{\beta}{\alpha}.$$

The estimates of $W_n(t)$ obtained by this method will be smaller than the true values. Upper bounds can also be obtained by using $\bar{W}_m$'s at $(j+1)\Delta$ rather than $j\Delta$ in the computation of α and β. In developing the $\bar{R}_x(n, j)$ sets, a preliminary estimate of $\bar{W}_n[(j+1)\Delta]$ is required (this is what we are trying to compute), but this can be done by extrapolating from $\bar{W}$ values associated with surrounding combinations of j and m. It is not difficult to develop compromise schemes that give estimates between the upper and lower bounds. All such variations, including the upper and lower bounds, are asymptotically correct (for large t) so that there is good control over error. Accuracy is improved by making Δ small, but the amount of computation varies directly with the number of points on the time grid. Computational experience has shown that some refinement in the method of choosing replacements for the $\bar{W}_m(j\Delta)$'s for use in computing α and β can make up for a rather coarse grid.

Figure 2 displays approximate solutions to the value equation for two horizons covering the last four hours of a day. The time scale is in hours remaining, and the boundary between the two horizons is at 2.0 hours. In both horizons, the number of available aircraft was set at 20 (curves for odd numbers of remaining sorties have been omitted). Since time 0 represents the end of the flying day, the boundary conditions were set to reflect zero value for any unused sorties at time 0. The boundary values for the next earlier horizon (horizon 2) were set in accordance with the conclusions reached in the next subsection. Additional details about the specific example used to generate the data in Fig. 2 are given at the end of this section.

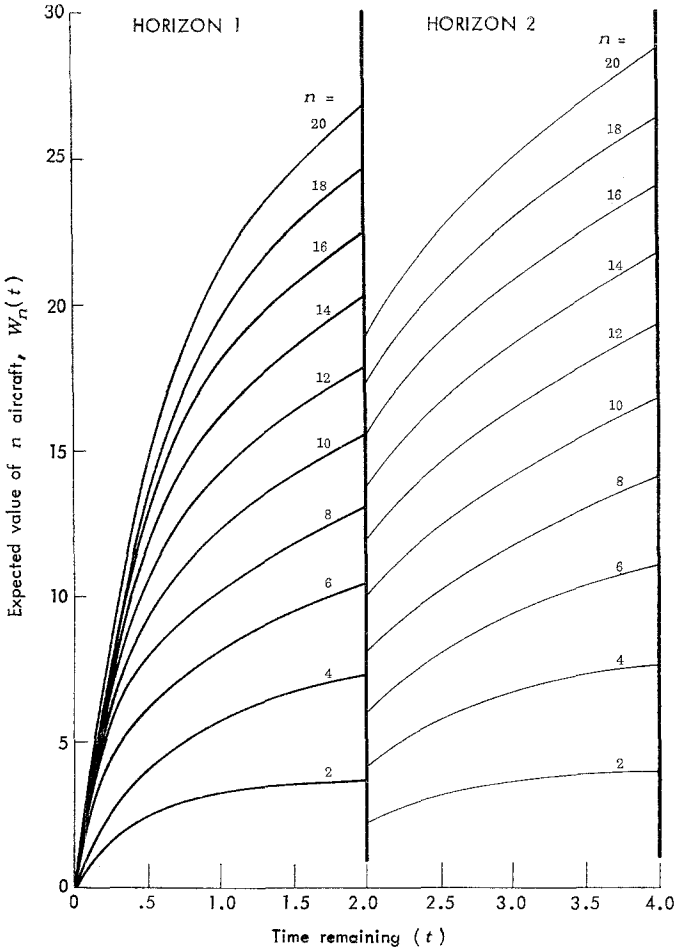


FIG. 2. Example solutions to the Value Equation for two horizons at end of day.

Models similar to the one presented in this section have been published by Kincaid and Darling (1963) and Kaufman (1963).

THE PLANNING TECHNIQUE

The purpose of the planning portion of JUDGE is to set values for the number of sorties available at the beginning of each horizon. The planning computation will probably be done once per horizon. For specificity, the following paragraphs describe procedures appropriate for the first such calculation of the day; minor modifications are required for subsequent updatings.

In our notational scheme, the horizons will be numbered from the last

occurring (horizon number 1) to the earliest. Such "backward numbering" is slightly more convenient than numbering horizons in their natural sequence and is analogous to our use of "t" in the previous discussion to represent the time remaining in a horizon. If there are r horizons in the flying day, the result of the planning stage is an r-dimensional vector $y = (y_1, y_2, \ldots, y_r)$, where the ith component represents the number of sorties planned to be available when the ith horizon begins. These numbers will not be exactly realized because the results of the dispatching procedure and the aircraft recovery are subject to uncertainty, and because the planning technique is based on certain approximations.

Since we must now consider a number of horizons, the $W_n(t)$ functions will be superscripted with the appropriate horizon indices. Thus $W_n{}^i(t)$ will be the expected value of the state (n, t) in the ith horizon. Also let the length of the ith horizon be denoted by t_i.

In developing the plan, we assume that there is a fixed number, s, of aircraft that are to be used for close air support during the day. The object is to determine y, subject to restrictions about the number of sorties that can be flown, that maximizes the quantity:

$$\sum_{i=1}^{r} W_{y_i}{}^i\ (t_i).$$

This objective function is appropriate since one term in the sum represents the total expected value of the sorties available within the associated horizon. Then the sum over horizons is a measure of the expected value to be gained from the allocation y.

It is clear that the planning technique should depend on the dispatching procedure, because it is impossible to plan without taking into account the effects of the dispatching method. This dependence is evident in our choice of objective function. On the other hand, the dispatching rule depends on the planning technique in the computation of the $W_n{}^i(t)$ functions. As discussed previously, the dispatching rule for the ith horizon depends on the boundary conditions $W_n{}^i(0)$, for $(n = 1, 2, \ldots, y_i)$.

The issue is: what should these boundary conditions be? Taking them to be zero is unsatisfactory because this would place an unrealistically low price on missions at the end of the horizon. Another alternative is to make the value of n sorties at the end of a horizon the same as the value of n sorties at the beginning of the next horizon. That is, to set $W_n{}^{i+1}(0) = W_n{}^i(t_i)$. But this would place too high a price on missions at the end of the $(i + 1)$st horizon in view of the imminent resupply.

Since unused sorties can be carried over to the next horizon, the bound-

ary values should reflect their expected utility in the next horizon. Suppose that the planned number of available sorties for the ith horizon is y_i. Then the boundary conditions for the next earlier horizon should be

$$W_n^{i+1}(0) = W_{y_i+n}^{i}(t_i) - W_{y_i}^{i}(t_i).$$

The right-hand side of this equation is a measure of the additional utility of n sorties brought from the $(i + 1)$st horizon to the next.

Suppose that there are n sorties still left h hours before the end of the $(i + 1)_{st}$ horizon, where h is a very small number. The cost of dispatching x sorties under these conditions would be

$$\begin{aligned} W_n^{i+1}(h) - W_{n-x}^{i+1}(h) &\doteq [W_{y_i+n}^{i}(t_i) - W_{y_i}^{i}(t_i)] - [W_{y_i+n-x}^{i}(t_i) - W_{y_i}^{i}(t_i)] \\ &= W_{y_i+n}^{i}(t_i) - W_{y_i+n-x}^{i}(t_i). \end{aligned}$$

Thus, this way of setting the boundary conditions satisfies our intuitive feeling that the cost of x sorties should be continuous across horizon boundaries.

The interdependence between planning and dispatching suggests that the plan be arrived at by an iterative procedure such as the following. The computation is begun with an initial guess at the allocation vector y. Based on this starting point, the values $W_m^i(t_i)$, associated with various numbers (say from 1 to s) of available sorties at the start of each horizon are calculated by the dispatching rule computation for use in the objective function of the planning algorithm (still to be described). The planning algorithm gives a new y vector that satisfies the sortie availability constraints and maximizes the objective function.

However, the boundary conditions used to calculate the $W_m^i(t_i)$ values depend upon the original y vector. If the new allocation is very different from the original, the $W_m^i(t_i)$ values corresponding to the new allocation might differ considerably from the original set of values, so that the new allocation would not really be optimal. The solution is to calculate the new set of end-of-horizon values based on the new allocation and use them in the planning algorithm to obtain a third allocation. This iterative procedure could be carried out until two successive allocations are fairly close.

Although there has not yet been any computational experience, the convergence should be rapid and it is doubtful that the process of recalculation would be (or should be) carried out more than once. This conjecture is supported by numerical results indicating that end-of-horizon values are not very sensitive to the boundary conditions from which they were obtained. (Compare the sets of end-of-horizon values for the two horizons in Fig. 2. The only difference between the horizons is in the boundary conditions.)

One difficulty arises in deriving the boundary conditions for a horizon from the end-of-horizon values of the next later horizon. To allow a full range of possibilities for the y vector, the planning algorithm should be supplied with values $W_m{}^i(t_i)$ running from 1 to s for each of the r horizons. It is natural to take all s boundary values for the last horizon to be zero. The boundary values for the next to last horizon (horizon 2) depend on the current y_1 since we take $W_m{}^2(0) = W_{m+y_1}{}^1(t_1) - W_{y_1}{}^1(t)$. This provides only $s - y_1$ boundary values for horizon 2. Continuing in this way, we would be able to calculate only $s - y_1 - y_2$ boundary values for horizon 3, and so on. A satisfactory way to avoid running out of boundary conditions is to set the missing boundary values for a horizon equal to the highest legitimate value available from the previous horizon. For example, in horizon 2 take

$$W_m{}^2(0) = W_{s-y_1}{}^2(0) = W_s{}^1(t_1) - W_{y_1}{}^1(t_1) \quad \text{for} \quad (m = s - y_1 + 1, \ldots, s).$$

A Planning Algorithm

The vector, y, which is chosen to maximize

$$\sum_{i=1}^{r} W_{y_i}{}^i(t_i),$$

is subject to constraints that reflect the limited number, s, of aircraft and the distribution of takeoff-to-ready times. The planning algorithm could be formulated as a stochastic programming problem, but such a formulation would be hopeless from the computational standpoint. Therefore, we shall retreat by writing the constraints in terms of expected values.

Since we are operating as though resupply occurs only at discrete points in time, a discrete representation of the takeoff-to-ready time distribution will be used. The most useful form in which to express the distribution is achieved by defining:

q_{ij} = Prob (an a/c launched in horizon j will not become ready by horizon i) for $(i < j, j = 2, \ldots, r)$,
$q_{jj} = 1$ for $(j = 1, \ldots, r)$.

The double subscript notation allows us to specify a different distribution for each launching horizon (j) if necessary. This will be necessary if the horizons are not all of the same length. For each j, there is a distribution of takeoff-to-ready times. Then q_{ij} is the sum of the tail of the distribution over horizons occurring later than i (horizons with indices less than i). Setting $q_{jj} = 1$ is equivalent to assuming that an aircraft cannot be used twice withon one horizon.

For planning purposes, assume that all sorties made available within a horizon will actually be used during that horizon. Since JUDGE is

intended for use under conditions of resources severely limited relative to expected demands, this assumption should not be far from the truth.

The total expected number of sorties that are unavailable (in the air or being recovered) by the end of horizon i is the sum over each horizon from r down to i of the product of sorties launched during the horizon times the proportion of those sorties that have not yet returned to the ready state. This number of sorties must be less than the total number of aircraft, s. Then the set of r constraints (one for each horizon) have the form:

$$\sum_{j=i}^{r} q_{ij} y_j \leq s \qquad \text{for } (i = 1, \ldots, r).$$

Although only integer solutions to this programming problem are desired, in view of the other approximations being made, one should not object to rounding fractions if s is not a small number.

An inconvenience arises because the objective function is nonlinear in the components of y, but this can be remedied by reformulating the problem in terms of another set of variables. Suppose that y is an allocation vector. For each j, $(j = 1, \ldots, r)$, define

$$z_{jk} = \begin{cases} 1 & \text{if} \quad k \leq y_j \\ y_j - k - 1 & \text{if } y_j < k < y_j + 1 \\ 0 & \text{if } y_j + 1 \leq k \end{cases} .$$

Going the other way, given z_{jk} $(j = 1, \ldots, r;\ k = 1, \ldots, s)$, then

$$y_j = \sum_{k=1}^{s} z_{jk} \qquad \text{for } (j = 1, \ldots, r).$$

Also define

$$c_{jk} = W_k{}^j(t_j) - W_{k-1}{}^j(t_j) \qquad \text{for } (j = 1, \ldots, r;\ k = 1, \ldots, s).$$

The variable z_{jk} has a value of 1 if the k_{th} sortie should be assigned to the j_{th} horizon and 0 if not. The coefficient c_{jk} is the incremental value of assigning the k_{th} sortie in the j_{th} horizon.

Consider the linear programming problem:

$$\text{maximize } Z = \sum_{j=1}^{r} \sum_{k=1}^{s} c_{jk} z_{jk},$$

$$\text{subject to } \sum_{j=i}^{r} q_{ij} \sum_{k=1}^{s} z_{jk} \leq s \qquad (i = 1, \ldots, r)$$

$$0 \leq z_{jk} \leq 1 \qquad (j = 1, \ldots, r;\ k = 1, \ldots, s).$$

In any reasonable situation, one would expect that $c_{jk} \leq c_{j(k-1)}$ (decreasing marginal utility of additional sorties), which would imply that $z_{j(k-1)} = 1$ if $z_{jk} > 0$. This assures us that a solution to the above problem would not, for example, refuse to assign the twelfth sortie and simultaneously indicate that there should be a thirteenth one, and that there would be at most one fractional z_{jk} for any j. Then the two programming problems are equivalent. Very efficient computer codes for solving quite large linear programming problems exist and good methods for solving the "zero-one" problem are becoming available.

Forecasts of future mission requests are incorporated into JUDGE through the specification of request rates and the joint distributions of the mission characteristics v and p for each horizon. These are used directly in computing the dispatching rule and indirectly in planning, since the planning technique uses the dispatching calculation.

In an implementation of JUDGE, it is likely that the planning computation would be carried out several times in the course of a single day, perhaps at the beginning of each horizon. This would make it possible to incorporate forecast modifications if they were indicated after the beginning of a day. It would also permit the system to adapt to the recovery of aircraft that have been dispatched in earlier horizons. When redoing the planning calculation, the current fleet status would be incorporated into the constraint set of the linear program, and the new forecasts would be used in the dispatching computations that provide the objective function.

Although a formal forecasting system for JUDGE has not yet been developed, experience in previous days, modified by knowledge of daily plans, should provide reasonable estimates of future demands.

A Numerical Example of the Dispatching Computation

A computer program that can calculate the dispatching rule for simple examples has been written in order to examine certain aspects of the dispatching rule. This program generated the data for Fig. 2. In the interest of simplicity, the program was designed to operate with a simpler mission value and probability structure than that described previously. Instead of working with an analytical formulation of $u(x: v, p)$ and a joint distribution of v and p, the program can accommodate up to eight arbitrary mission value functions in the form of a table. These functions are indexed by a single parameter. The distribution functions F and G are replaced by a set of up to eight probabilities representing a distribution over the indices of the mission value functions.

For the numerical results to be discussed, eight mission value functions were used. None gave an improvement in immediate reward for more than

eight sorties or offered a reward greater than 8.0 when the maximum of eight sorties was dispatched. All results are based on boundary conditions of $W_n(0) = 0$; n was carried up to 48 aircraft, and t was taken to 5.0 hours. The data used to plot the left side of Fig. 2 represent a portion of the results for the particular case of $\lambda = 6.0$ requests per hour. (The time units are rather arbitrary; by relabeling the time scale this would be equivalent to a two-hour horizon with a request rate of 15 per hour, or any other combination for which the expected number of requests in the horizon is 30.)

The primary outputs of the program are the solutions to the value equations and the dispatching rule in the form of a three-dimensional table. This table specifies the number of aircraft to be sent as a function of: the number of remaining sorties, the time left in the horizon, and the index number of the mission value function appropriate to the request. In addition, the program has the ability to test the dispatching rule with sequences of requests generated by the Monte Carlo method.

Table 1 illustrates how the dispatching rule behaves with respect to a particular type of mission request (i.e., a fixed mission value function) as the request rate, number of sorties available, and time remaining are varied. The sequence of appropriate decisions as a function of time is given for three values of λ and six values of n. As one would expect, the generosity with aircraft increases with the number of sorties left and as the end of the horizon approaches, but decreases with higher request rates.

The data of Table 2 are the result of a Monte Carlo experiment designed to examine the sensitivity of the dispatching rule to errors in forecasting the request rate. Five values of λ were used. The column labeled "Rule Value" contains the values of $W_{48}(5.0)$ obtained from the value equation in the process of calculating the rules. When the forecast is accurate, these numbers represent the expected total value of all requests satisfied. The remaining column headings indicate the actual rates that governed the arrival of mission requests in the Monte Carlo tests. The entries in these columns are the averages of values actually attained for missions dispatched for the indicated combinations of forecasted and actual rates. Within each column, the highest score was obtained when the forecast was correct, but it also appears that the dispatching rule can tolerate fairly high forecast errors without a great deal of degradation in the performance measure.

An interesting feature of the dispatching rule is suggested by Fig. 3, obtained from the Monte Carlo experiments. On this diagram are plotted the inventories of remaining sorties as functions of time for various actual request rates when the rule used was calculated from a forecasted request rate of 6.0. The rule seems to have a built-in ability to correct

TABLE 1
NUMBER OF A/C TO DISPATCH FOR A TYPE-1 REQUEST

Time Left in Horizon	$n = 7$			$n = 15$			$n = 23$			$n = 31$			$n = 39$			$n = 47$		
λ	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0
0.2	7	7	7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
0.4	7	5	5	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
0.6	5	5	0	8	7	7	8	8	8	8	8	8	8	8	8	8	8	8
0.8	5	0		8	7	5	8	8	8	8	8	8	8	8	8	8	8	8
1.0	4			7	5	1	8	8	7	8	8	8	8	8	8	8	8	8
1.2	0			7	3	0	8	8	5	8	8	8	8	8	8	8	8	8
1.4				5	0		8	7	1	8	8	8	8	8	8	8	8	8
1.6				5			8	4	0	8	8	4	8	8	8	8	8	8
1.8				3			8	1		8	8	1	8	8	8	8	8	8
2.0				0			7	0		8	6	0	8	8	4	8	8	8
2.2							6			8	3		8	8	0	8	8	8
2.4							4			8	0		8	8		8	8	3
2.6							2			8			8	5		8	8	0
2.8							0			8			8	1		8	8	
3.0										6			8	0		8	6	
3.2										4			8			8	3	
3.4										1			8			8	0	
3.6										0			7			8		
3.8													5			8		
4.0													3			8		
4.2													1			7		
4.4													0			5		
4.6																2		
4.8																0		

TABLE 2
VALUES ATTAINED FOR VARIOUS FORECASTED AND ACTUAL REQUEST RATES

λ	Rule Value	Actual Request Rate 2.0	4.0	6.0	8.0	12.0
2.0	46.6	45.7	60.4	62.4	64.2	66.0
4.0	61.7	42.9	61.2	66.9	71.1	73.9
6.0	68.5	39.0	59.2	68.7	73.0	78.9
8.0	73.7	36.3	56.6	68.0	73.3	81.9
12.0	83.1	30.2	51.1	62.5	70.6	83.4

itself. When demands are very high, the number of sorties available dropped rapidly, causing the criteria for dispatching a given number of sorties to become more stringent. Thus, even when the request rate is badly underestimated, some resources are still conserved throughout most of the horizon.

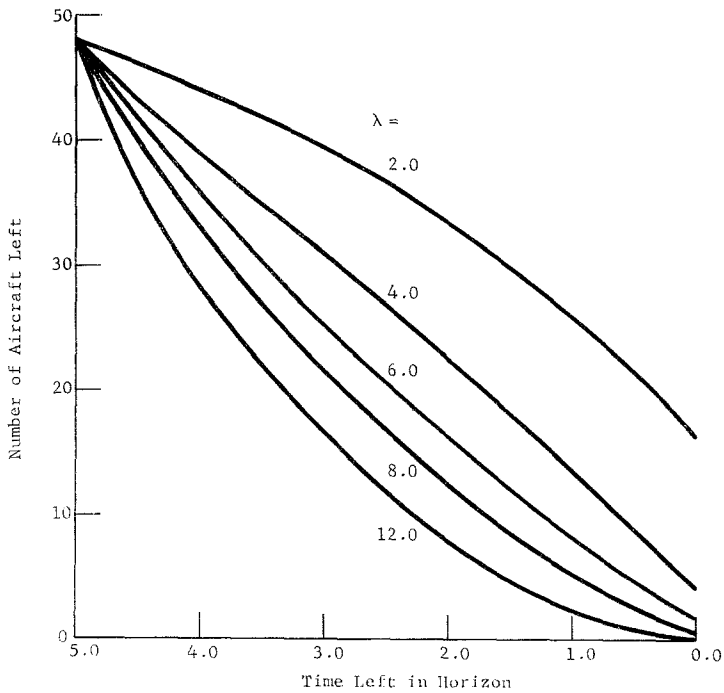


FIG. 3. Depletion of aircraft for various request rates under rule based on forecasted rate of 6.0.

EMPIRICAL STUDIES

The preceding discussion has shown how explicit values of possible mission outcomes can be used to make mission-dispatching decisions. It has assumed that such values can be obtained. In a sense, the truth of the assumption is self-evident. A man, presented with a table in which a set of such values is to be entered, can easily be persuaded to enter numbers. The question is not whether such numbers can be obtained, but whether they are appropriate bases for decision. A general and abstract discussion of this and some related questions is reserved for the conclusion of this paper.

Situations exist in which an appropriate external standard of value can be defined. One necessary, though far from sufficient, property that judged values should have if they are to be appropriate bases for decision is that they should in such situations agree reasonably well with the external standard. A study exhibiting that property is described below.

Another possible criterion for appropriateness of judged values is that judges should agree. But this criterion is tricky. Taken literally, it denies the obvious truth that men may disagree about values even when they know the same facts. Still, a system whose performance depends entirely on the identity of its value judge is uncomfortably subjective—though perhaps realistically so.

One way out of the dilemma is to exclude the man from the definition of the system by saying that the system exists for the purpose of implementing its value judge's values as effectively as possible. If human values in fact differ in irreconcilable ways, no other position is possible, since all decision systems must be based on human values in one way or another. But the facts may permit a less subjective resolution of the problem. Presumably training should serve to create a communality of values. If so, men should agree to some extent on their value judgments if they have been trained at all, and the extent of that agreement should be an increasing function of the amount and communality of their training and experience. If this kind of agreement is desired, then an appropriate performance criterion for JUDGE (or perhaps for the larger system, including selection and training procedures for its value judges, in which JUDGE is embedded) would be that it enhances this kind of agreement, as compared with its competitors. This is, of course, a property that can be studied in experiments and simulations.

This section presents the results of two studies: an experiment on methods for collecting value judgments, and a field study permitting some comparison of JUDGE with current procedures not based on explicit value judgments.

The Methodology Experiment

An experiment was run to determine the best procedure for asking human subjects for value judgments. Three procedures were used to estimate the prices of used automobiles.

(1) *Ratio.* The subject said how many times more or less valuable a car was then a carefully defined standard car, by placing a mark on a scale having the digits 1–5 logarithmically spaced on it, and checking beside the word MORE or LESS to indicate the direction of the judgment.

(2) *Difference.* The subject said how much more or less valuable a car was then the standard car, by placing a mark on a scale the same size as the one for ratio judgments, but with figures of $500–$2000 linearly spaced on it, and marking MORE or LESS to indicate the direction of the judgment. Both this scale and the ratio one were open ended at the top. The standard car was the same one used in the ratio procedure.

(3) *Direct.* The subject simply estimated the value of the car he was judging.

The stimulus material consisted of four equivalent sets of 10 automobiles, ranging in retail price from about $500 to about $4500 in the Kelly *Blue Book,* and described as they would be in an advertisement for used cars. One car, priced at $1800, was included on all four lists, and was used as the standard in the two procedures that required one. A fourth task, that of paired comparisons, was administered to the subjects, but the results from this task are not reported here. (The laboriousness of paired comparisons precludes their use in systems requiring numerous human judgments.)

The subjects, 40 volunteer college students who were paid for their two hours of participation in the experiment, were randomly divided into four groups. Each group received a different pairing of the four procedures with the four groups of cars. Thus each subject used all procedures and judged all cars, making a total of 40 judgments. The combination of groups of subjects, procedures, list of cars, and order of presentation was specified by a Greco-Latin square design so that all order effects and effects of combinations of procedures and lists of cars would be completely counterbalanced.

The subjects' responses were transformed into dollar values. For the ratio procedure, this was done by multiplying the *Blue Book* value of the standard by the subject's judgment for the other car. For the difference procedure, the subject's judgment for the other car was added to or subtracted from the *Blue Book* value of the standard. For the direct estimation procedure, the ratio of the subject's estimates for the other car and

for the standard car was calculated, and then processed by the same procedure used for the ratio scale estimates. (This served to adjust the direct estimates for a form of individual bias.) The performance measure was the mean square error for each subject on the nine cars he judged by each procedure, the error being the difference between the estimated dollar value and the "true" *Blue Book* value.

An analysis of variance was performed on these error scores. The Greco-Latin square design of course precludes calculation of variance estimates for interactions. None of the methodological variables (groups of subjects, lists of cars, and order of presentation) produced any systematic difference. The effect of scaling procedure was highly significant; inspection of means confirms this conclusion. The mean squared error ($\times 10^{-5}$) for the ratio procedure was 31.86; for the difference procedure, 6.62; and for the direct estimation procedure, 8.37.

On the basis of these results, the difference procedure was chosen for use in the field study described below. Although it and the direct procedure are equally attractive on the basis of these data, the availability of a natural scale of value (that is, dollars) for used cars makes the direct procedure more appropriate for them than it would be in general.

TABLE 3

MEAN PRODUCT-MOMENT CORRELATION COEFFICIENTS BETWEEN ESTIMATED AND ACTUAL VALUES OVER SUBJECTS WITHIN PROCEDURES

Ratio		Direct		Difference	
M	σ	M	σ	M	σ
.894	.066	.810	.074	.819	.084
.787	.116	.900	.046	.729	.150
.822	.105	.886	.132	.919	.051
.887	.078	.927	.135	.926	.051

To estimate the overall reliability of the groups of subjects, we calculated the product-moment correlation coefficients between the values obtained by each procedure and the *Blue Book* values. Table 3 shows the means over subjects within procedures and the standard deviations for the three procedures. Note that the correlations are gratifyingly high. This means not only that the subjects agreed well with the *Blue Book,* but also that they agreed well with one another.

As a further check on intersubject agreement, we calculated product-moment correlations between all possible pairs of subjects within each group of ten subjects. The means and standard deviations of these correlations are shown in Table 4. The size of these numbers is evidence of the high reliability of the estimation process.

TABLE 4
MEANS AND STANDARD DEVIATIONS
FOR PRODUCT MOMENT CORRELATION COEFFICIENTS
BETWEEN ALL SUBJECT PAIRS WITHIN EACH GROUP OF TEN SUBJECTS

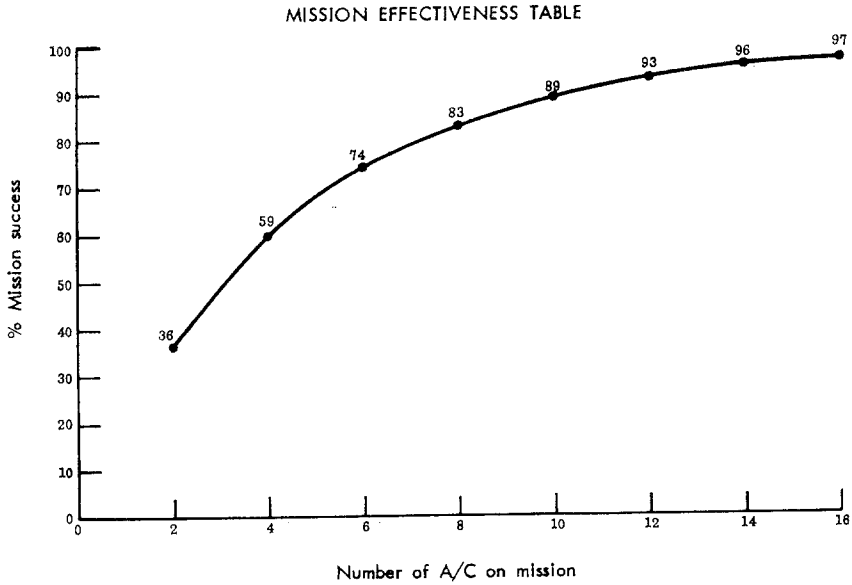
Ratio		Direct		Difference	
M	σ	M	σ	M	σ
.779	.114	.800	.113	.789	.115
.709	.157	.849	.086	.712	.130
.665	.192	.769	.166	.887	.063
.824	.090	.925	.043	.876	.062

THE FIELD STUDY

The field study was performed to explore techniques for obtaining value judgments in relatively realistic situations, and to examine JUDGE's performance based on those value judgments. The exercise, carried out in a classroom-like situation, used scenario materials based on an unclassified lesson plan developed at the U.S. Army Command and General Staff College at Fort Leavenworth, Kansas. Seventeen Air Force officers familiar with tactical air warfare served as subjects. These were a group of Forward Air Controllers and Air Liaison Officers stationed at Cannon Air Force Base in Clovis, New Mexico.

The subjects were given a brief description of the political situation leading to the hypothetical conflict, and a detailed description of the battle situation was given using a large-scale map. The battle was set in Southeast Asia, and involved three divisions of allied forces operating against a roughly comparable enemy force. The subjects completed the entire experiment described in the remainder of this paragraph in a single three-hour session. The experiment was concerned with two successive days of a battle, and two hours of each of these days were simulated. Each two-hour period is referred to as a "situation" in the experimental design. Additional narrative material provided the transition from the first to the second day.

During each simulated two-hour period, a sequence of targets was presented, one at a time. A target report was a printed form giving a description of the target, its location, the source of the report; for the DASC system, a mission effectiveness function and the time of the report were also included. (The system here called DASC is our abstraction of the currently employed TAC system.) A sample target report for the DASC system appears as Fig. 4. For the DASC portions of the experiment, subjects were told that the mission effectiveness function associated



TAC TARGET
CLOSE AIR SUPPORT

Target No.	Coordinates:	Location:
A–0	T 1019	Sadao
Reported by: 2nd and 3rd Bde. advanced elements		Time: 0900

Description:

Advanced elements under attack by enemy tank platoon. Attack consists of five tanks, type unknown, with 105 mm guns and heavy m.g.'s mounted.

FIG. 4. Target report for DASC.

with a particular target embodied all relevant information concerning aircraft and ordnance performance. The graph of a mission effectiveness function indicates the probability that the target will be destroyed or, in the case of a distributed target, the portion of the target expected to be destroyed as a function of the number of aircraft dispatched.

To compare JUDGE with sortie allocations made in the usual way (the DASC system), the subjects were exposed to each battle situation twice. Acting as decision makers in the DASC system, the subjects

allocated specific numbers of sorties against the targets as they appeared; acting as value estimators in the JUDGE system, they assigned values to the targets as they appeared. The subjects were instructed to try not to let their responses under one operating mode affect their decisions under the other. Two groups of subjects were formed and the ordering of systems was counterbalanced as shown below.

	Situation 1		Situation 2	
Order	1	2	3	4
Group 1	DASC	JUDGE	JUDGE	DASC
Group 2	JUDGE	DASC	DASC	JUDGE

Before operating in the DASC mode, the subjects were told that they had 40 sorties to dispatch in the two-hour situation and that they should expect to receive requests at the rate of about ten per simulated hour. (The actual times on the target reports were selected by a random process which yielded 18 targets for the first situation and 22 targets for the second.) Six different mission effectiveness functions of the form $\eta(x) = 1 - (1 - p)^{x^\alpha}$ were used and were displayed to the subjects during the instructional period. The subjects were told that the distribution of effectiveness functions over targets would be uniform.

For the JUDGE task, the target described in Fig. 4 was used as a standard. The subjects were to associate a value of 100 with this target and to consider an utterly worthless target as having a value of 0. All other targets were to be valued relative to these fixed numbers. The value to be judged was the importance of destroying the target.

To convert the subject's; value responses into dispatching decisions, a computer program was written to calculate the JUDGE dispatching rule. This program treats the prior distributions of value judgments and mission effectiveness functions as independent. In testing the JUDGE system, two different value distributions have been employed. Distribution U is a uniform (rectangular) distribution with a range of values between 0 and 150; distribution T is triangular with a mode of 0 and gives nonzero probabilities for values up to 225. Both of these one-parameter distributions have means of 75, their major characteristics are shown in Table 5. Distribution U was chosen because of its simplicity, while distribution T was suggested by observing a histogram of all value responses obtained in the field study. This histogram was constructed with very broad intervals to smooth over subject preferences for certain round numbers.

As a prior estimate for the distribution of mission effectiveness func-

tions over targets in each situation, we used the actual distribution taken over both situations. This distribution assigned nearly equal probabilities to the six functions and was not exactly correct for either situation 1 or 2. In addition, the program was supplied with the forecasted rate of 10 requests per hour, the horizon length of two hours, and the initial availability of 40 sorties.

TABLE 5
PRIOR DISTRIBUTIONS OF VALUES USED FOR JUDGE

Distribution	U (Uniform)	T (Triangular with Mode 0)
Density Function	$\frac{1}{c}$ for $(0 \leq v \leq c)$	$\frac{2}{c}\left(1-\frac{v}{c}\right)$ for $(0 \leq v \leq c)$
Mean	$\frac{c}{2} = 75$	$\frac{c}{3} = 75$
Standard Deviation	$\frac{c}{2\sqrt{3}} = 43.3$	$\frac{c}{3\sqrt{2}} = 53.0$

For each simulated time point, t, at which a target was presented during the DASC phases of the experiment, a vector of values, $\bar{W}_m(t)$, $(m = 0, 2, \ldots, 40)$ was stored. Given a subject's sequence of value judgments, the corresponding JUDGE dispatches can then be easily calculated. In both systems, missions were required to consist of multiples of two aircraft.

The means and standard deviations of the subjects' value responses are tabulated in Table 6. Comparing these data with the information in Table 5 indicates that both of our prior value distributions were in considerable error for many of the subjects. To be useful, however, JUDGE must be robust against errors of forecasting.

COMPARISON OF JUDGE AND DASC

To provide a basis for a meaningful comparison of JUDGE and DASC, we examined two other dispatching rules. One was a theoretical optimum, the best that could have been done with perfect foreknowledge of the incidence and value of all requests during the two-hour period. In the absence of such perfect foreknowledge, this optimum is of course unattainable. To calculate the perfect sequence of dispatching decisions, pairs of planes were assigned to targets in order of decreasing marginal utility, ignoring the sequence in which the targets arrived, until all planes were used up. This optimum differs from one subject to another, since it is based on value judgments.

The other dispatching rule was first-come, first-served (FCFS). We recognized that any dispatching rule, no matter how absurd, would be

certain to obtain some expected utility if it dispatched aircraft at all. Only improvements over some minimal performance level should be given credit. So to specify such a minimal system, we chose to dispatch planes four at a time to all targets until planes were exhausted.

Table 7 shows the results for situation 1, and Table 8 for situation 2. The first five columns contain the expected utilities earned by FCFS, DASC, JUDGE(U), JUDGE(T), and the optimum system for each subject, based on each individual's value judgments. The column headings JUDGE(U) and JUDGE(T) are, respectively, JUDGE computed with

TABLE 6
MEANS AND STANDARD DEVIATIONS OF VALUE JUDGMENTS

		Situation 1		Situation 2	
Group	Subject	Mean	*SD*	Mean	*SD*
1	1	55.8	47.7	58.3	45.5
	2	120.0	51.6	122.9	58.4
	3	68.1	58.8	92.5	68.8
	4	67.8	63.1	106.9	82.2
	5	85.0	41.8	84.6	59.4
	13	45.6	45.0	53.8	59.9
	14	47.5	49.2	58.1	48.8
	15	82.8	52.6	96.0	66.1
	16	75.2	36.0	90.4	23.5
	17	71.7	58.5	72.5	55.7
	Mean	72.0	50.4	83.6	56.8
	SD	20.5	7.9	21.6	14.8
2	6	124.2	54.1	120.2	40.6
	7	106.9	44.7	95.0	30.0
	8	65.6	27.7	69.8	36.3
	9	96.7	54.4	60.6	29.4
	10	68.6	23.3	80.6	35.1
	11	91.1	32.3	76.2	41.9
	12	57.5	51.9	53.1	39.6
	Mean	87.2	41.2	79.3	36.1
	SD	22.5	12.2	20.9	4.6
	Grand Mean	78.2	46.6	81.1	48.3
	SD	22.6	10.9	21.4	15.5

TABLE 7
SYSTEM PERFORMANCE: SITUATION 1

	Expected Utility					Effectiveness		
Subj	FCFS	DASC	JUDGE (*U*)	JUDGE (*T*)	Optimum	DASC	JUDGE (*U*)	JUDGE (*T*)
1	238	395	450	447	480	0.65	0.88	0.86
2	351	419	532	548	619	0.26	0.68	0.74
3	251	333	449	445	459	0.40	0.95	0.94
4	193	297	398	394	417	0.46	0.92	0.90
5	341	435	531	530	530	0.50	1.00	1.00
6	329	730	772	779	850	0.77	0.85	0.86
7	470	651	754	774	777	0.59	0.92	0.99
8	246	352	439	439	441	0.54	0.99	0.99
9	374	545	662	652	674	0.57	0.96	0.93
10	224	325	404	395	433	0.48	0.86	0.82
11	342	471	578	579	594	0.52	0.94	0.94
12	177	249	404	408	420	0.30	0.94	0.95
13	171	159	346	342	375	0.06	0.86	0.97
14	310	472	465	462	497	0.86	0.83	0.81
15	249	329	466	451	496	0.33	0.88	0.82
16	290	377	467	464	499	0.42	0.85	0.83
17	368	376	509	512	543	0.05	0.81	0.82
Mean						0.448	0.888	0.893

the U and T value distributions described in Table 5. The last three columns contain relative effectiveness numbers for DASC and the two versions of JUDGE. For each subject, they are calculated by treating the expected utility obtained by FCFS as 0 and the expected utility obtained by the perfect system as 1. With this definition of the origin and a unit of measurement of the utility function, DASC performs about 50% as well as the unattainable optimum, while JUDGE performs about 90% as well as the perfect system.

The large discrepancy between JUDGE and DASC indicates, as expected, that JUDGE is much more efficient in implementing a subject's values than the subject is himself. JUDGE separates the evaluation portion of the dispatching task (the portion that depends on human expertness) from the decision making portion, which, given the value judgments, is a difficult computational task that a computer can perform more effectively than a man.

It is important that DASC, while much inferior to JUDGE, is superior to FCFS. Since the scores justifying this assertion are based on the value judgments, this finding means that the DASC decisions are by no means unrelated to the JUDGE value judgments. It is reasonable to believe

TABLE 8
SYSTEM PERFORMANCE: SITUATION 2

	Expected Utility					Effectiveness		
Subj	FCFS	DASC	JUDGE (*U*)	JUDGE (*T*)	Optimum	DASC	JUDGE (*U*)	JUDGE (*T*)
1	328	410	436	461	467	0.59	0.78	0.96
2	514	629	666	708	769	0.45	0.60	0.76
3	362	493	539	550	563	0.65	0.88	0.94
4	410	536	592	605	629	0.58	0.83	0.89
5	374	442	563	562	603	0.30	0.82	0.82
6	492	658	769	783	841	0.48	0.80	0.84
7	389	581	683	694	695	0.63	0.96	1.00
8	310	456	510	524	536	0.64	0.86	0.95
9	275	331	411	408	424	0.38	0.91	0.89
10	313	510	552	575	575	0.75	0.91	1.00
11	282	323	488	479	488	0.20	1.00	0.96
12	176	250	296	282	296	0.62	1.00	0.88
13	292	325	427	461	461	0.20	0.80	1.00
14	241	496	499	489	536	0.86	0.88	0.84
15	383	454	551	580	626	0.29	0.69	0.81
16	432	488	603	599	616	0.30	0.93	0.90
17	247	494	555	559	567	0.77	0.96	1.00
Mean						0.511	0.861	0.908

that each subject's JUDGE dispatches and his DASC dispatches are attempts to implement the same set of values; the JUDGE dispatches are simply more effective at doing so.

An incidental observation makes the same point. One of the authors, familiar with the stimuli ahead of time, and with complete knowledge of the scoring rules, performed the role of a subject twice. Though he attempted to make DASC dispatches that would produce high scores, his data closely resembled those reported in Tables 7 and 8. It is simply difficult to translate a value system into dispatching decisions, and JUDGE does it much better than men can.

That JUDGE(*U*) and JUDGE(*T*) are so nearly alike indicates that the dispatching rule is quite robust against errors in forecasting the distribution of value judgments.

We attempted to scale the subjects' situation-2 responses based on their situation-1 value judgments to see if the effectiveness of JUDGE could be improved by making a subject's distribution of value judgments correspond more closely to the forecasted distribution. Since there was a considerable degree of correlation between the subject's means and standard deviations between the two situations, it is reasonable to sup-

pose that it would be profitable to modify a subject's responses using statistics obtained from observing him at an earlier time. The mean and standard deviation of distribution U are 75 and 43.3, respectively. Suppose that a particular subject's mean and standard deviation in situation 1 were m and s. Letting x represent a response in situation 2, an adjusted response for that target was computed by the formula

$$x_{adj} = \frac{43.3}{s}(x - m) + 75.$$

This simple adjustment is just a stretching (or compressing) and a movement of the origin of the subject's response scale based on his previous responses.

The application of this adjustment technique resulted in very modest effectiveness increases for JUDGE(U). That the improvement was only slight was to be expected from the already high effectiveness of JUDGE and from the large amount of robustness already evident.

Intercorrelations Among Subjects' Scores

So far, evidence indicates that JUDGE implements each subject's values better than his own decisions can. Ideally, we would like to show the relation between an individual subject's value judgments and some ultimate criterion of value, such as "winning the war." Unfortunately, we do not know how to derive target values from any such ultimate criterion. Indeed, if we could do so, such calculated values rather than human judgments should be the values that JUDGE translates into decisions.

Thus, no ultimate validation of JUDGE, or of any command system, is possible. The philosophical basis for this conclusion is the subject of the last section of this article. However, relevant questions can be examined. For example, intra-subject and inter-subject reliability are both relevant. If a subject's value judgments collected at one time systematically differ from his value judgments for the same targets in the same situation collected at a different time, there would be some doubt about the appropriateness of implementing either set of values. Unfortunately, the field study provides no information about intra-subject reliability since there was no replication. However, the use of many subjects permits us to examine the extent to which one subject agrees with another.

Table 9 presents the mean intercorrelations among subjects within groups and situations for three sets of numbers: DASC dispatching decisions, value judgments, and JUDGE dispatching decisions based on those judgments. Three points can be noted about these numbers. First, they are all low. Clearly these subjects disagreed with one another both about how valuable the targets were and about how many planes to send

against each. Second, in every case, the mean intercorrelation between JUDGE dispatching decisions is higher than that for the DASC decisions and higher than that for the value judgments on which they are based. These decisions reflect not only those values, which differ from subject to subject, but also mission effectiveness functions and times at which the requests arrived, both constant across subjects. It is gratifying that the JUDGE intercorrelations are higher than those for DASC. Perhaps the most suggestive feature of Table 9, however, is that the mean intercorrelations for situation 2 are higher in all cases than those for situation 1, and JUDGE gains more than DASC. Clearly learning is going on, and it seems unlikely that it has reached its limits.

TABLE 9
AVERAGE CORRELATIONS OF SUBJECTS WITH ALL OTHER SUBJECTS

	Situation 1			Situation 2		
Group	DASC Dispatches	JUDGE Values	JUDGE Dispatches	DASC Dispatches	JUDGE Values	JUDGE Dispatches
1	0.346	0.413	0.461	0.396	0.539	0.550
2	0.252	0.238	0.358	0.396	0.530	0.566

Further experimentation in a laboratory where greater demands on a subject's time can be made is necessary to establish the upper limits of learning and both kinds of reliability. The field study has established that JUDGE works; it is much superior to DASC, and it is robust against various kinds of deviations from prior expectation.

GENERAL COMMENTS ON VALUE-JUDGMENT-BASED COMMAND SYSTEMS

As we worked on JUDGE, we found ourselves strongly influenced by a set of philosophical ideas about judgment-based systems in general. These ideas have to do with the purpose, design, and evaluation of such systems. Most of our ideas about the TAC problem follow from these more general considerations—though many of them could be justified from other, less radical, points of view than the one we present here. Since we have seen no other presentation of the position we have come to, and indeed very little discussion of the issues underlying validation of command systems exists on paper, we have chosen to end this article with a fairly extended discussion of these philosophical questions.

We believe that all command systems are and must always be judgment-based. A command system exists to make decisions. We find it useful to distinguish between the decision, or selection of an action, and

the decision process. A decision process describes a complex sequence of events beginning with recognition that an action will have to be selected, and ending with implementation of the selected action. We believe that human judgments play an absolutely necessary role in all decision processes, at least for decisions important and interesting enough to concern command systems. Our goal is to analyze the decision process into functions, to ascertain how and by what means each function should be performed, and to allocate those functions best performed by men to men, and those best performed by machines to machines. The assertion that all command systems are and must always be judgment-based means, then, that the set of functions best performed by men will never turn out to be empty. One reason for that assertion is that command systems exist to serve human purposes, and require men to specify what those purposes are. We believe also that, at least for a long time to come, men will be indispensable for a number of other functions in the decision process, as the following discussion exhibits.

The fact that all command systems are judgment-based is often not explicitly recognized because of a faulty definition of system boundaries. The statement is sometimes made, for instance, that "the function of a command system is to help the commander do his job." This is too narrow a definition of the system. In the definition that we consider appropriate, the commander is a part of the system, and the function of the entire system, commander and all, is to make "the right decisions."

When is a decision right? Everyday evaluation of decisions is usually done by comparing the choice actually made with the one that would have been made either by the man doing the comparing or by some appropriate authority. If the person making the comparison is the one who made the decision, then most decisions are right by definition, and this definition of rightness is uninteresting. If someone else judges whether a decision is right or wrong, then when he says it is wrong we have two experts who disagree, and usually no satisfactory means of resolving the disagreement.

The search for a way out of this impasse has characterized almost all research on command systems. We feel that a departure is possible only after radical redefinition of what is meant by "right" in this context. The remainder of this discussion presents our redefinition and shows how to use it.

Validity and Reliability

The notion that there is a right action, answer, diagnosis, or other system output, and that the problem of system design is to devise a system that produces this right output reminds us of the problem of

validating the design of intelligence tests or other personnel selection instruments. The test designer may start out with the rather simple idea that some abstract quantity like, say, intelligence exists and that his task is to measure that quantity in the individuals he tests. He develops a possible method for measurement, and then must consider whether or not it really measures what he hopes it measures. How can he find out?

Two main approaches have been taken to the problem of validating tests. One, the criterion-oriented approach, depends on comparing the test under study with some other, already available measure of the quantity to be measured. If test A is "known" to measure intelligence, then test B can be considered valid if it correlates highly with test A. In the case of intelligence tests, Binet's original idea was to use success in school as his validating criterion.

Criterion-oriented validity has two major difficulties. One is that most criteria are themselves suspect, either because they are of doubtful relevance to the abstract entity to be measured (is success in school really primarily a function of intelligence?) or because as measures the criteria themselves have unattractive properties, such as unreliability, or both. The other is that for most abstract entities we might want to test (e.g., propensity to take risks) no appropriate criterion is available.

Because of these difficulties, the testers have developed a second and quite different approach to validity, which they call construct validity. (There is yet another concept called content validity, having to do with validating tests of mastery of a subject matter, but it can be neglected here.) The basic idea of construct validity is that a test should make sense and data obtained by means of it should make sense. One form of making sense is that different procedures that purport to measure the same abstract quantity should co-vary. If one procedure is taken as a criterion for the other, this is simply criterion-oriented validity. But even if neither procedure is taken as valid, a priori, the fact that they correlate highly indicates that to some extent they measure the same thing or closely related things. If, in addition, the kind of underlying quantity that each might tap seems on some a *priori* basis to be the same, then observation of covariation is an instance of what has been called converging operations, and contributes to validity. (The concept of construct validity is broader than this description indicates; this discussion serves only to give its flavor.)

A procedure (test, system, etc.), whether or not it is valid, should certainly be reliable. This word, taken from test theory, is a bit too specific for our purpose. We will say that a procedure should be intellectually coherent. One requirement of intellectual coherence is that repeated attempts to measure the same thing, or equivalent things, by

means of the procedure should produce the same measurements. Another requirement of coherence is that variables that seem irrelevant to the procedure should not affect it.

Other requirements of coherence exist. For decision-making procedures, logical consistency is one of them. Thus if a decision-making system prefers act A to act B and act B to act C, it should not prefer act C to act A; it should be transitive. Similarly, it should exhibit the properties known in decision theory as avoidance of dominated strategies, independence from irrelevant alternatives, and a few others.

A final class of coherence requirements is harder to describe. There are obvious expectations about the behaviors of certain systems. An information processing system, for example, should not act as though its odds for hypothesis A have been increased as a result of evidence that clearly makes A less likely than it was before. Information that an act has become more valuable than it was before should not cause a decision-making system to become less likely that it was before to choose that act.

Now that the concept of coherence has been introduced, it becomes easier to talk about validity. Validation is simply establishing the coherence of a procedure, or several procedures. Thus no sharp line separates the concept of reliability from that of validity; both concepts refer to agreements among measures, and a continuum exists from cases in which the measures essentially repeat the same procedure (reliability) to cases in which rather different procedures seem to measure the same thing (validity).

The Validation-Type Reliability of Judgmental Systems

We assert that no external measure of the performance of a judgment-based decision-making system is possible. Any such measure would have to compare the decisions the system made with decisions made some other way, and there would have to be some good reason to suppose that the decisions made the other way were the right ones. But if we reject the idea that the business of a decision-making system is to imitate some individual's decisions (in which case the only point of building the system would be to save the individual the trouble of making those decisions himself), then no basis remains for asserting that the decisions made by one procedure (e.g., by the commander) are inherently appropriate simply because they were made by that procedure, regardless of their content. An examination of the merit of decisions in terms of their content is a matter of intellectual coherence or reliability, not validity.

We assert also that intellectual coherence or reliability is very measurable and is in fact what we want the output of a decision-making system to have. The rest of this philosophical section concerns some thoughts

about how to obtain intellectual coherence in judgment-based command systems, and how to establish that it has (or has not) been obtained.

Before going on, however, we pause to answer a possible objection. It is possible to think of command situations and systems within which the quality of system performance is easily defined and easily measured. From a too-superficial viewpoint one could argue, for example, that a business management should maximize dollar return and that dollar return is easily measured. Actually, both of these statements are incorrect, since most businesses have many goals other than maximum dollar return (e.g., market position, stability of employment, maintenance of stock value, image, etc.) and the accounting fictions underlying the definition of profit are the output of an elaborate and basically subjective judgmental process. But lower-level command systems may have rather acceptable external standards of quality of performance. The goal of a cab dispatching system, for example, might be to minimize total customer waiting time—an easily measured quantity.

But why is that goal appropriate, rather than some other, possibly inconsistent goal, such as minimizing number of miles traveled by empty cabs? We can think of no example in which the choice of goal, or of weighting function by means of which to combine goals, is not a judgmental matter. So even command systems in which performance measures are easily identified are inherently and essentially based on value judgments—the value judgments that specify the system's goals.

A Task Analysis of Command Systems

As we envision it, the basic function of any command system is to make decisions. The formal analysis of decision-making is extensive and the basic principles are well understood. We will not review that analysis here; such references as Luce and Raiffa (1957) present it. Implementing that analysis in the system design is, we believe, necessary to attain intellectual coherence, but far from easy. Table 10 presents a detailed breakdown of functional steps in implementing the analysis, steps that must be performed in one way or another by any command system. Table 10 also contains our opinions (very much subject to modification) about whether each function should be performed by men, machines, or both, and our opinions (firmer) about whether each function should be performed at the moment of decision or in advance.

Functions 2–6 should be performed ahead of time if possible, but it will not always be possible. We are particularly interested in functions 5 and 10, which along with function 2 (about which we know nothing abstract) constitute the basic functions that men should perform in command systems. Functions 5 and 10 correspond to the two basic vari-

TABLE 10
FUNCTIONS OF COMMAND SYSTEMS

Function	Performed By	When Performed
1. Recognize that a decision problem exists	men	ahead of time
2. Identify available acts	men	ahead of time if possible
3. Identify relevant states that determine payoff for acts	men	ahead of time if possible
4. Identify the value dimensions to be aggregated into the payoff matrix	men	ahead of time if possible
5. Judge the value of each outcome on each dimension	men	ahead of time if possible
6. Aggregate value judgments into a composite payoff	machines	
7. Identify information sources relevant to discrimination among states	men	ahead of time
8. Collect data from information sources	both	at moment of decision
9. Filter data, put into standard format, and display to likelihood estimator	both	at moment of decision
10. Estimate likelihood ratios (or some other quantity indicating the impact of the datum on the hypotheses)	men	at moment of decision
11. Aggregate impact estimates into posterior distributions	machines	at moment of decision
12. Decide among acts by using principle of maximizing expected value	machines	at moment of decision
13. Implement the decision	both	at moment of decision

ables of decision theory: utility and probability. (Functions 10 and 11, as stated, imply a point of view about how relevant probabilities should be obtained in decision-making systems; systems that work this way are called Probabilistic Information Processing or PIP systems, and are discussed by Edwards and others. See Edwards, 1965; 1966; Edwards *et al.*, 1964; Edwards and Phillips, 1964; Schum *et al.*, 1966; Kaplan and Newman, 1966). Both of these quantities, we assert, are inherently judgmental; the basic roles of men in command systems are to make these judgments.

TWO PRINCIPLES FOR JUDGMENT-BASED DECISION SYSTEM DESIGN

Table 10 implies two principles for the design of judgment-based decision systems. These principles have somewhat the status of axioms; they are fundamental to our argument, but not directly demonstrable.

We will state and argue for them, but firm establishment of their appropriateness for guidance of system design must come from success of the resulting systems.

Principle 1. The judgments that men must make in decision systems should be fragmented into relatively small, elementary parts when possible. We see five advantages to using this principle:

(1) It permits the automation of significant elements of the decision-making task.

(2) It greatly simplifies the task of the human beings working in the system, by reducing the difficulty of each judgment.

(3) Because of 2, it reduces the difficulty of training the system operators.

(4) It permits allocation of the judgment task to several men rather than one; there is no requirement that all information must ultimately be evaluated by one man. In systems with a high information load, this consideration alone would be enough to justify the principle.

(5) Because of 1, it permits machine portions of the system to monitor and insure intellectual coherences of various kinds, either by checking judgments to insure coherence or by performing operations according to rules that guarantee coherence.

Since the fifth point is crucial, it is appropriate to add that examination of actual human decisions, in laboratory (see Edwards, 1954) and other contexts, indicates that incoherences frequently occur in them. In fact, the known rules for intellectual coherence of decisions are sufficiently demanding so that it is extremely difficult by unaided intuition to make a reasonably large set of decisions without violating some of them. So principle 1, properly applied, can be expected to result in major gains in coherence.

Clearly the functional analysis of Table 10 and the use of principle 1 are appropriate only if men can in fact perform fragmented judgments effectively. In an important sense, it is self-evident that they can, since men can and do generally make reasonably appropriate decisions, and some version of each of the functions listed in Table 10 must be performed, implicitly or explicitly, before any decision can be made. But it is not self-evident that men can perform such functions explicitly, as is necessary to implement the philosophy of system design implied by Table 10. Principle 2 asserts that they can.

Principle 2. Men can make explicit probability and value judgments, using appropriate response mechanisms and after appropriate training. Where external standards of correctness of such judgments are available, human judgments will usually be incorrect, but not severely so, and appropriate system design can minimize such errors. Whether or not

external standards of correctness are available, appropriately obtained judgments will be relatively coherent.

Principle 2 is an empirically testable assertion, and a variety of experimental evidence bears on it. Psychophysics is the branch of psychology devoted to the extraction of human judgments about reasonably simple sensory events; a basic conclusion is that men make such judgments very well indeed. (See Stevens and Galanter, 1957.) Recent research on probability estimation indicates that men can judge relative frequencies with great accuracy (Robinson, 1964; Shuford, 1961). More complex probability judgments suffer from an inherent deficiency that has been named conservatism; men are unable to extract from data as much certainty as the data justify. (See Peterson and Miller, 1965; Phillips, Hays, and Edwards, 1966.) But human conservatism in probability estimation can be overcome by appropriate system design; that is the point of the PIP system mentioned above. (See Edwards *et al.*, 1964.) Evidence on human value judgments is sketchy and fragmentary; such as it is, it indicates that men do rather well at translating even rather complicated value systems into numbers. (See, for example, Yntema and Torgerson, 1961.)

But final justification of the use of Principles 1 and 2 for system design can come only from success of the resulting systems.

Judgmental Systems are Self-Validating

The arguments presented above have led us to a set of ideas about how to design judgmental systems. One crucial feature of these ideas is that explicit value or utility judgments lie at the core of such systems. This fact has implications for the validation (in the sense already defined) of such systems.

As we have pointed out, validation in a classical sense consists of demonstrating coherence between the output of a system, in this case decisions, and the comparable output of some other system considered to be effective or valid. With decision systems, however, such acceptable external criteria are nonexistent. The usual expedient is to use some wise and experienced decision-maker's judgments as the criterion. However, unless the goal of the system is merely to reproduce that man's decisions, this procedure is unsatisfactory—especially since any man's decisions are likely to be incoherent to some extent.

As we have already argued, intellectual coherence, taken over as large a domain of thought as possible, is an alternative approach to validation, and the only approach available for the command systems of interest here. And a variety of kinds of coherence are built into the sort of judgmental system implied by Table 10.

Still other kinds of coherence can and should be examined by means of research on any proposed command system; such research can range from highly informal studies of system elements to major formal simulations of the system as a whole.

In the particular case of decision systems based on value judgments, a natural requirement of the system is that the decisions should cause as much value as possible to accrue to the entity in whose service the decisions are being made. Procedures internal to the system will guarantee this, given that the value judgments made by system operators are taken as the "true" values. The question of whether this criterion is met therefore reduces to two other questions. Are value judgments reliable, from time to time within one judge or from one judge to another? If not, can the unreliability be accounted for as true differences in values, from time to time or from one judge to another? This second question is bound to be a matter of opinion, since "true" values are inaccessible and perhaps undefined. Still, the opinion need not be entirely unguided by data.

Obviously such questions of reliability will be extensively studied in the course of system design. So will other forms of intellectual coherence. By the time any command system of the kind implied by Table 10 has been fully designed, formal and empirical information about the various relevant kinds of intellectual coherence will have been built into the design details at almost every point. Thus the design of such a system is self-validating, in a very important sense.

Of course no validation is ultimate; each conclusion that a system is valid for its purpose means no more than that a decision has been reached to proceed to the next step in its design and use. Conclusions reached during system design concerning the level of quality to be expected in system performance will be modified or replaced by conclusions based on the result of simulations; conclusions based on simulations will be modified or replaced by conclusions based on actual system use. And even actual use is not an ultimate criterion; each new use under new conditions may require a new judgment of system validity. Because, in the last analysis, all we can ever mean by stating that a procedure is valid is that, on the basis of what we know, it makes sense.

REFERENCES

Becker, G. M., and McClintock, C. G., Value: behavioral decision theory. *Annual Review of Psychology,* 1967, **18**, 239–286.

Cox, D. R., and Smith, W. L. On the superposition of renewal processes. *Biometrika,* 1954, **41**, 91–99.

Edwards, W. The theory of decision. *Psychological Bulletin,* 1954, **51**, 380–417.

Edwards, W. Probabilistic information processing systems for diagnosis and action

selection. *Information system sciences: proceedings of the second congress.* Spartan Books, Washington, 1965.

EDWARDS, W., LINDMAN, H., AND PHILLIPS, L. D. Emerging technologies for making decisions, *In* T. M. Newcomb (Ed.), *New directions in psycholology II,* New York: Holt, Rinehart, and Winston, 1964.

EDWARDS, W., AND PHILLIPS, L. D. Man as a transducer for probabilities in bayesian command and control systems. *In* M. W. Shelly and G. L. Bryan (Eds.), *Human judgments and optimality.* New York: John Wiley and Sons, 1964.

KAPLAN, R. J., AND NEWMAN, J. R. Studies in probabilistics information processing. *IEEE Transactions on Human Factors in Electronics,* HFE-7, 1966, 49–63.

KAUFMAN, G. M. *Statistical decision and related techniques in oil and gas exploration.* Prentice-Hall, Englewood Cliffs, New Jersey, 1963.

KINCAID, W. M., AND DARLING, D. A. An inventory pricing problem. *Journal of Mathematical Analysis and Applications,* 1963, **7**, 183–208.

LUCE, R. D., AND RAIFFA, H. *Games and decisions,* New York: John Wiley and Sons, 1957.

PETERSON, C. R., AND BEACH, L. R. *Man as an intuitive statistician,* Human Performance Center, Technical Report No. 4, University of Michigan, Ann Arbor, Michigan, November 1966.

PETERSON, C. R., AND MILLER, A. J. Sample size and the revision of subjective probabilities. *Journal of Experimental Psychology,* **70**, 1965, 117–121.

PHILLIPS, L. D., HAYS, W. L., AND EDWARDS, W. Conservatism in complex probabilistic inference. *IEEE Transactions on Human Factors in Electronics,* HFE-7, 1966, 7–18.

ROBINSON, G. H. Continuous estimation of a time-varying probability. *Ergonomics,* 1964, **7**, 7–21.

SCHUM, D. A., GOLDSTEIN, I. L., AND SOUTHARD, J. F. Research on a simulated Bayesian information-processing system. *IEEE Transactions on Human Factors in Electronics,* HFE-7, 1966, 37–48.

SHUFORD, E. H. Percentage estimation of proportion as a function of element type, exposure time, and task. *Journal of Experimental Psychology,* 1961, **61**, 430–436.

STEVENS, S. S., AND GALANTER, E. H. Ratio scales and category scales for a dozen perceptual continua. *Journal of Experimental Psychology,* 1957, **54**, 377–409.

YNTEMA, D. B., AND TORGERSON, W. S. Man-computer cooperation in decisions requiring common sense. *IRE Transactions on Human Factors in Electronics,* HFE-2, 1961, 20–26.

RECEIVED: MAY 17, 1967