

This excerpt from

Adaptation in Natural and Artificial Systems.

John H. Holland.

© 1992 The MIT Press.

is provided in screen-viewable form for personal use only by members  
of MIT CogNet.

Unauthorized use or dissemination of this information is expressly  
forbidden.

If you have any questions about this material, please contact  
[cognetadmin@cognet.mit.edu](mailto:cognetadmin@cognet.mit.edu).

### 3. Illustrations

The formal framework set out in chapter 2 is intended, first of all, as an instrument for uniform treatment of adaptation. If it is to be useful a wide variety of adaptive processes must fit comfortably within its confines. To give us a better idea of how the framework serves this end, the present chapter applies the framework in several different fields. It will repay the reader to skim through all of the illustrations on first reading, but he should skip without hesitation over difficult points on unfamiliar ground, reserving concentration for illustrations from familiar fields. Although each of the illustrations adds something to the substantiation of the framework no one of them is essential in itself to later developments. The interpretations, limited usually to one commonly used model per field, are of necessity largely informal, but two points can be checked in each case: (1) the facility of the framework in picking out and organizing the facts relevant to adaptation, and (2) the fit of established mathematical models within the framework.

#### 1. GENETICS

. . . genes act in many ways, affecting many physiological and morphological characteristics which are relevant to survival. All of these come together into the sufficient parameter “fitness” or selective value. . . . Similarly environmental fluctuation, patchiness, and productivity can be combined . . . in . . . [a] measure of environmental uncertainty. . . .

Levins in *Changing Environments* (pp. 6-7)

The phenotype is the product of the harmonious interaction of all genes. The genotype is a “physiological team” in which a gene can make a maximum contribution to fitness by elaborating its chemical “gene product” in the needed quantity and at the time when it is needed in development. There is extensive interaction not only among the alleles of a locus, but also between loci. The main locale of

these epistatic interactions is the developmental pathway. Natural selection will tend to bring together those genes that constitute a balanced system. The process by which genes are accumulated in the gene pool that collaborate harmoniously is called "integration" or "coadaptation." The result of this selection has been referred to as "internal balance." Each gene will favor the selection of that genetic background on which it can make its maximum contribution to fitness. The fitness of a gene thus depends on and is controlled by the totality of its genetic background.

Mayr in *Animal Species and Evolution* (p. 295)

We have already looked at genetic processes at some length in the preliminary survey, so this illustration will be brief, mostly recapitulating the main points of the earlier discussion, but within the formal framework. Typically, only a certain range of basic structures, i.e., chromosomes, is admitted to studies in genetics, so that only a species, family, or other taxonomic grouping is involved. Still, in principle, one can study all possible variations, including variations in chromosome number and type. The range of the study will be primarily determined by the set  $\Omega$  of genetic operators admitted, since the possible variants (genotypic and phenotypic) will be those produced by sequences of genetic operators from  $\Omega$ . Familiar examples of genetic operators are mutation, crossover, inversion, dominance modification, translocation, and deletion (see the formal definitions given in chapter 6).

The genetic adaptive plan develops in terms of an everchanging population of chromosomes which, interacting with the environment, provides a concurrent sequence of phenotype populations. For many purposes, it is convenient to represent a population as a probability distribution over the set of genotypes  $\Omega_1$ , where the probability assigned to genotype  $A \in \Omega_1$  is the fraction of the total population consisting of that genotype (cf. Crow and Kimura 1970). Thus the population at time  $t$  can be specified by  $\Omega(t) \in \Omega$ , where  $\Omega$  is the set of distributions over  $\Omega_1$ . In very general terms, each element of the population is tested against the environment and is ranked according to its fitness—its ability to survive and reproduce. It's often useful to think of the environment  $E$  in terms of environmental niches, each of which can be exploited by an appropriate set of phenotypic characteristics. Then fitness  $\mu_E$  becomes a function of the coadapted sets of alleles which produce these characteristics (see chapter 4). From this point of view the population  $\Omega(t)$  can be looked upon as a reservoir of coadapted sets, preserving the history of past advances, particularly the environmental niches encountered.

Most mathematical models of genetic adaptation are based on very simple reproductive plans, where each individual allele  $\sigma_i$  is assigned a fitness  $\mu_E(\sigma_i)$  and

the fitness of any set of alleles  $\{\sigma_1, \dots, \sigma_m\}$  is taken to be the sum of the fitnesses of the alleles in the set,

$$\sum_{i=1}^m \mu_E(\sigma_i).$$

However, in general, the fitness of an allele depends critically upon the influence of other alleles (epistasis). The replacement of any single allele in a coadapted set may completely destroy the complex of phenotypic characteristics necessary for adaptation to a particular environmental niche. The genetic operators provide for the preservation of coadapted sets by inducing a “linkage” between adjacent alleles—the closer together a set of alleles is on a chromosome, the more immune it is to separation by the genetic operators. Thus a more realistic set of adaptive plans provides for emphasis of coadapted sets through reproduction, combined with application of the genetic operators to provide new candidates and test established coadapted sets in new combinations and contexts.

More formally, an interesting set of plans can be defined in terms of a two-phase procedure: First the number of offspring of each individual  $A$  in a finite population  $\alpha(t)$  is determined probabilistically, so that the expected number of offspring of  $A$  is proportional to  $A$ 's observed fitness  $\mu_E(A)$ . The result is a population  $\alpha'(t)$  with certain chromosomes emphasized, along with the coadapted sets they contain. Then, in the second phase, the genetic operators from  $\Omega$  are applied (in some predetermined order) to yield the new population  $\alpha(t+1)$ . One class of plans of considerable practical relevance can be defined by assuming that operator  $\omega_i$  from  $\Omega$  is applied to an individual  $A \in \alpha'(t)$  with probability  $p_i$  (constant over time). It is easy to see that the efficiency of such a plan will depend upon the values of the  $p_i$ ; it is perhaps less clear that once each of the  $p_i$  has a value within a certain critical range, the plan remains efficient, relative to other possible plans, over a very broad range of fitness functions  $\{\mu_E, E \in \mathcal{E}\}$ . In particular, if chromosomes containing a given linked set of alleles repeatedly exhibit above-average fitness, the set will spread throughout the population. On the other hand, if a linked set occurs by happenstance in a chromosome of above-average fitness, later tests will eliminate it (see chapters 6 and 7). It is this mode of operation (and others similar) which gives such plans robustness—the ability to discover complex combinations of coadapted sets appropriate to a wide variety of environmental niches.

Because of the central role of fitness, it is natural to discuss the efficiency and robustness of a plan  $\tau$  in terms of the average fitnesses of the populations it produces. Formally, the average fitness in  $E$  of a finite population of genotypes  $\alpha_\tau(t)$  produced by  $\tau$  at time  $t$  is given by

$$\mu_E(\tau, t) = \frac{\sum_{A \in \alpha_t(\tau)} \mu_E(A)}{M(\tau, t)},$$

where  $M(\tau, t)$  is the number of individuals in  $\alpha_t(\tau)$ . If we take the ratio  $\mu_E(\tau, t)/\mu_E(\tau', t)$ , we have an indication of how close  $\tau$  comes to "extinction" relative to  $\tau'$  (in the sense that extinction occurs when the population produced by  $\tau$  becomes negligible relative to the population produced by  $\tau'$ ). If we take the greatest lower bound of this ratio relative to some set of possible plans  $\mathcal{J}$ , we have an indication of the worst that can happen to  $\tau$  in  $E$ , relative to  $\mathcal{J}$  at time  $t$ . Continuing in this vein we get the following criterion for ranking plans as to robustness over  $\mathcal{E}$ :

$$\operatorname{glb}_{E \in \mathcal{E}} \operatorname{glb}_t \operatorname{glb}_{\tau' \in \mathcal{J}} \mu_E(\tau, t) / \mu_E(\tau', t).$$

In effect this criterion ranks plans according to how close they come to extinction under the most unfavorable conditions.

The fantastic variety of possible genotypes, the effects of epistasis, changing environments, and the difficulty of retaining adaptations while maintaining variability (genetic variance), all constitute difficulties which genetic processes must surmount. In terms of the  $(\mathcal{J}, \mathcal{E}, \chi)$  framework these are, respectively, problems of the large size of  $\mathcal{G}$ , the nonlinearity and high dimensionality of  $\mu_E$ , the non-stationarity of  $\mu_E$ , and the mutual interference of search and exploitation. The  $(\mathcal{J}, \mathcal{E}, \chi)$  framework enables the definition of concepts (chapters 4 and 5) which in turn (chapters 6 through 9) help explain how genetic processes meet these difficulties in times consistent with paleological and current biological observations.

#### Summarizing:

- $\mathcal{G}$ , populations of chromosomes represented, for example, by the set of distributions over the set of genotypes  $\mathcal{G}_1$ .
- $\Omega$ , genetic operators such as mutation, crossover, inversion, dominance modification, translocation, deletion, etc.
- $\mathcal{J}$ , reproductive plans combining duplication according to fitness with the application of genetic operators; for example, if each operator  $\omega_i \in \Omega$  is applied to individuals with a fixed probability  $p_i$ , then the set of possible plans can be represented by the set

$$\{(p_1, \dots, p_i, \dots, p_b) \text{ where } 0 \leq p_i \leq 1\}.$$

- $\mathcal{E}$ , the set of possible fitness functions  $\{\mu_E : \mathcal{G} \rightarrow \mathcal{G}\}$ , each perhaps stated as a function of combinations of coadapted sets.

x, comparison of plans according to average fitnesses of the populations produced; for example,

$$\operatorname{glb}_{E \in \mathcal{S}} \operatorname{glb}_t \operatorname{glb}_{\tau' \in \mathcal{I}} \mu_E(\tau, t) / \mu_E(\tau', t).$$

## 2. ECONOMICS

The specification of how goods can be transformed into each other is called the *technology* of the model and the specification of how goods are transformed to satisfaction is called the *utility function*. Given this structure and some initial bundle of goods, the problem of optimal development is to decide at each point of time how much to invest and how much to consume in order to maximize utility summed over time in some suitable way.

Gale in "A Mathematical Theory of Optimal Economic Development" *Bull. AMS* 74, 2 (p. 207)

One of the most important formulations of mathematical economics is the von Neumann technology. This technology can be presented (following David Gale 1968) in terms of a finite set of *goods* and a finite set of *activities*, where each activity transforms some goods into others. If the goods are indexed, then the goods available to the economy at any given time can be presented as a vector where the  $i$ th component gives the amount of the  $i$ th good. In the same way, the input to the  $j$ th activity and the resultant output can be given by a pair of vectors  $W_j$  and  $W'_j$ , where the  $i$ th component of  $W_j$  specifies the amount of the  $i$ th good required by the activity, while the  $i$ th component of  $W'_j$  specifies the amount produced. An activity can be operated at various levels of effort so that, for instance, if the amount of input of each required good is doubled then the amount of output will be doubled. More generally, if the level of effort for activity  $j$  is  $c_j$  then the pair  $(W_j c_j, W'_j c_j)$  specifies the input and output of the activity. If a mixture of activities is allowed, the overall technology can be specified as the set of pairs

$$\{(Wc, W'c) \ni c \in Q\}$$

where  $W$  and  $W'$  are matrices having the vectors  $W_j$  and  $W'_j$  as their respective  $j$ th columns, each  $c$  is a vector having the level of the  $j$ th activity as its  $j$ th component, and  $Q$  designates the set of admissible activity mixes (corresponding to the real constraints limiting the total activity at any time). A *program* for utilizing the technology is given by a sequence of activities  $\langle c_i \rangle$  satisfying the intuitive "local" requirement that the total amount of each good required as input for the activities at time

## TYPICAL ACTIVITIES:

	"Coal Storage"		"Coal Mining"		"Tool Fabrication"	
	Input	Output	Input	Output	Input	Output
Wood	0	0	0	0	1	0
Coal	1	1	0	4	0	0
Iron Ore	0	0	0	0	0	0
Steel	0	0	0	0	1	0
Tools	0	0	1	0	0	1

$$\begin{array}{l}
 \text{Matrix } W \\
 (\text{Combining Activity} \\
 \text{Input Requirements})
 \end{array}
 \quad
 \begin{array}{l}
 \text{Matrix } W' \\
 (\text{Combining Activity} \\
 \text{Outputs})
 \end{array}$$

Wood	$\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$
Coal	$\begin{bmatrix} 0 & 1 & 0 & 1 & 2 & 0 & 0 & 0 \end{bmatrix}$
Iron Ore	$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$
Steel	$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$
Tools	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$

Wood	$\begin{bmatrix} 3/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$
Coal	$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 4 & 0 \end{bmatrix}$
Iron Ore	$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 4 \end{bmatrix}$
Steel	$\begin{bmatrix} 0 & 0 & 0 & 1 & 3 & 0 & 0 & 0 \end{bmatrix}$
Tools	$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$

## TYPICAL PRODUCTION SEQUENCE:

$\tau:$

$$\begin{array}{ll}
 Q(t): & Q(1) \xrightarrow{c_1} Q(2) \xrightarrow{c_2} \dots \\
 \text{Wood} & \begin{bmatrix} 10 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = W \cdot \begin{bmatrix} 5 \\ 5 \\ 5 \\ 5 \\ 0 \end{bmatrix} \quad W' \cdot \begin{bmatrix} 5 \\ 5 \\ 5 \\ 5 \\ 0 \end{bmatrix} = \begin{bmatrix} 15/2 \\ 5 \\ 5 \\ 5 \\ 0 \end{bmatrix} = W \cdot \begin{bmatrix} 5 \\ 0 \\ 5/2 \\ 5/2 \\ 5/2 \\ 0 \\ 0 \end{bmatrix}
 \end{array}$$

Fig. 3. Example of von Neumann technology

$t$  cannot exceed the total amount of that good produced as output in the preceding period:

$$W'c_{t-1} \geq Wc_t$$

(using matrix multiplication and the obvious extension of inequality to vectors). Activities which dispose of or store goods can be introduced so that the given inequalities can be changed to equalities without loss of generality. Thus, given an initial supply of goods  $V(0)$ , the set of admissible programs becomes

$\mathcal{C} = \{\text{sequences } C_\beta = \langle c_{\beta,t} \rangle, \text{ where } \beta \in \mathfrak{G}, \text{ an indexing set,}$   
 and  $t = 0, 1, 2, \dots \ni$  (i)  $c_{\beta,t} \in Q$ , (ii)  $Wc_{\beta,0} = V(0)$ ,  
 (iii)  $W'c_{\beta,t} = Wc_{\beta,t+1}\}$ .

It is assumed that each activity vector  $c$  can be assigned a unique *utility*  $\mu(c)$  designating the satisfaction to society of engaging in the mix of activities specified by the vector. (This way of assigning utility has the nice feature that satisfying activities which do not directly consume goods, such as viewing pictures in a museum or conserving goods for future use, can be included in the model.) The object of the study is to compare various programs in terms of the utility sequences they produce. Typically, programs are compared over some interval of time  $(0, T)$  by taking the difference of their accumulated utilities

$$U_\beta(T) - U_{\beta'}(T) \quad \text{where} \quad U_\beta(T) = \sum_{t=0}^T \mu(c_{\beta,t}).$$

A program  $C_{\beta'}$  is considered optimal if

$$\operatorname{glb}_{C_\beta \in \mathcal{C}} \liminf_{T \rightarrow \infty} [U_{\beta'}(T) - U_\beta(T)] \geq 0.$$

Because  $Q$  sets an upper limit to levels of effort, an optimal program always exists. A program  $C_\beta$  will often be satisfactory if its rate of accrual of utility  $U_\beta(T)/T$  is comparable to that of  $C_{\beta'}$ .

Generally interest centers on “noncontracting” economies where, once an activity is possible, it continues to be possible at any subsequent time. This can be guaranteed, for example, if there is a set of initial goods which are “regenerated” by all activities (cf. sunlight, water, and air) and from which all other goods can be produced by appropriate sequences of activities. In such economies a mix of activities can be tried and, if found to be of above-average utility, can be employed again in the future.

In the  $(\mathfrak{J}, \mathcal{E}, \chi)$  framework, the set of admissible activity mixes  $Q$  becomes the set of structures  $\mathfrak{G}$ . An adaptive plan  $\tau$  generates a program  $C$  by selecting a sequence of activity vectors  $\langle c_{\tau,t} \rangle$  on the basis of information received from the environment (economy). The environment  $E$  in this case makes itself felt only through the observed utility sequence  $\langle \mu_E(c_{\tau,t}) \rangle$ ; thus different utility functions correspond to different environments. Within this framework, the basic concern is discovery of an adaptive plan which, over a broad variety of environments, generates programs which work “near-optimally.” A typical criterion of “near-

optimality" would be that for all utility functions of interest the ratio of the rate of accrual of the adaptive plan  $\tau$ ,  $U_\tau(T)/T$ , to that of  $C_{\beta^*}(E)$ ,  $U_{\beta^*(E)}(T)/T$ , approaches 1 for each  $E \in \mathcal{E}$ . That is,

$$\lim_{T \rightarrow \infty} [U_\tau(T)/U_{\beta^*(E)}(T)] = 1, \quad \text{for all } E \in \mathcal{E}.$$

Generally there will be some additional requirement that the rates be comparable for all times  $T$ .

Adaptation becomes important when there is uncertainty about just what utility should be assigned to given activity mixes, or when it is difficult to project  $\mu_E$  into the future, or when  $Q$  is a function of time (reflecting technological innovations). The key to formulating an adaptive plan here, paralleling the procedure in other contexts, is continual use of incoming information (about satisfactions and dissatisfactions, changing technology, etc.) to modify activity levels. A well-formulated plan should respond automatically, specifying adjustments needed, as information accumulates. Since, in von Neumann's formulation, the environment is characterized by the utility assigned to different activity vectors, we can limit consideration to payoff-only plans. The fact that reproductive plans are payoff-only plans which can be proved near-optimal (in the sense defined above) for *any* set of utilities, makes it likely that such plans can supply the responsiveness required here. In  $(\mathfrak{I}, \mathcal{E}, \chi)$  terms the basic problems here, as in the genetics illustration, are the large size of  $\mathfrak{C}$  coupled with nonlinearity and high-dimensionality of  $\mu_E$ . Because the concepts of chapters 4 and 5 are formulated in terms of the general framework, they apply here as readily as to genetics. The resulting techniques are specifically interpreted as optimization procedures throughout chapter 6, at the end of section 7.1, and throughout section 7.2.

#### Summarizing:

- $\mathfrak{C}$ , the set of admissible activity vectors  $Q$ .
- $\Omega$ , transformations of  $Q$  into itself.
- $\mathfrak{I}$ , plans for selecting a program  $\langle c_i \rangle \in \mathfrak{C}$ , where  $c_i$  is an activity vector in  $Q$ , on the basis of observed utilities  $\{\mu_E(c_{t'})\}_{t' < t}$ , i.e., payoff-only plans.
- $\mathcal{E}$ , an indexing set of possible utility functions  $\{\mu_E : Q \rightarrow \mathbb{R}, E \in \mathcal{E}\}$ .
- $\chi$ , typically a requirement that, for all utility functions  $\mu_E$ ,  $E \in \mathcal{E}$ , the limiting rate of accrual of a plan,  $\lim_{T \rightarrow \infty} (U_\tau(T)/T)$ , equal that of the best possible program  $C_{\beta^*}(E)$  in each  $E \in \mathcal{E}$ .

### 3. GAME-PLAYING

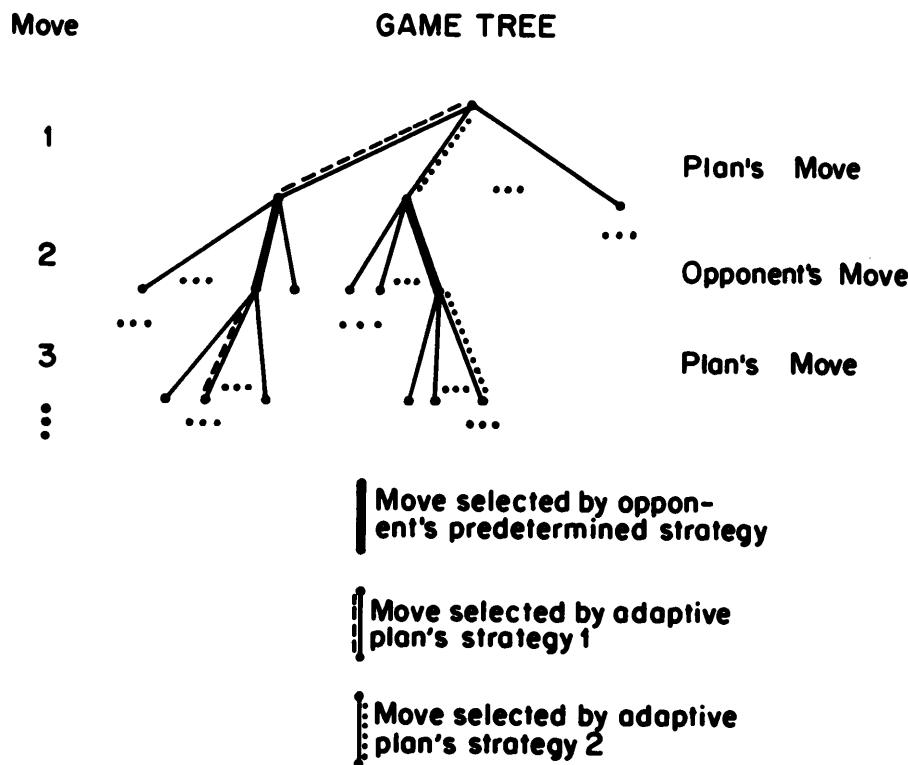
Lacking such knowledge [of machine-learning techniques], it is necessary to specify methods of problem solution in minute and exact detail, a time-consuming and costly procedure. Programming computers to learn from experience should eventually eliminate the need for much of this detailed programming effort.

Samuel in "Some Studies in Machine Learning Using the Game of Checkers" *IBM J. Res. Dev.* 3 (p. 211)

Most competitive games played by man (board games, card games, etc.) can be presented in terms of a *tree* of moves where each *vertex* (point, node) of the tree corresponds to a possible game configuration and each directed *edge* (arrow) leading from a given vertex represents a legal move of the game. The edge points to a new vertex corresponding to a configuration which can be attained from the given one in one move (turn, action); the options open to a player from a given configuration are thus indicated by the edges leading from the corresponding vertex. The tree has a single distinguished vertex with no edges leading into it, the *initial* vertex, and there are *terminal* vertices, having no edges leading from them, which designate outcomes of the game. In a typical two-person game which does not involve chance, the first player selects one of the options leading from the initial configuration; then the second player selects one of the options leading from the resulting configuration; the *play* of the game proceeds with the two players alternately selecting options. The result is a *path* from the initial vertex to some terminal vertex. The outcomes are ranked, usually by a *payoff* function which assigns a value to each terminal vertex.

In these terms, a *pure strategy* for a given player is an algorithm (program, procedure) which, for each nonterminal configuration, selects a particular option leading therefrom. Once each player chooses a pure strategy, the outcome of the game is completely determined, although in practice it is usually possible to determine this outcome only by actually playing the game. Thus, in a strictly determined (non-chance) two-person game, each pair of pure strategies (one for each player) can be assigned a unique payoff. The object of either player, then, is to find a strategy which does as well as possible against the opponent as measured by the expected payoff. This informal object ramifies into a whole series of cases, depending upon the initial information about the opponent and the form of the game.

One of the simplest cases occurs when it is known that the opponent, say the second player, has selected a single pure strategy for all future plays of the game.



*Fig. 4. Example of a game tree*

The object of the first player, then, is to learn enough of the strategy chosen by the second player to find an opposing strategy which maximizes payoff. When the game tree involves only a finite number of vertices, as is often the case, it is at least theoretically possible to locate the maximizing strategy by enumerating and testing all strategies against the opponent. However, if there is an average of  $k$  options proceeding from each configuration, and if the average play involves  $m$  moves, there will be in excess of  $k^m$  pure strategies. The situation is quite comparable to the examples of enumeration given earlier. Even for a quite modest game with  $k = 10$  and  $m = 20$ , and a machine which tests strategies at the exceptional rate of one every  $10^{-9}$  second, it would require in excess of  $10^{11}$  seconds, or about 30 centuries, to test all possibilities. Efficiency thus becomes the critical issue, and

interest centers on the discovery of plans which enable a player to do well while learning to do better. If plans are compared in terms of accumulated payoff, a criterion emerges analogous to the classical "gambler's ruin" of elementary probability. Let  $U_E(\tau, t)$  be the payoff accumulated to time  $t$  by plan  $\tau \in \mathcal{Z}$  confronting the (unknown) pure strategy  $E \in \mathcal{E}$ , and require that

$$\operatorname{glb}_{E \in \mathcal{E}} \operatorname{glb}_{\tau \in \mathcal{Z}} [U_{n,E}(t)/U_{\tau,E}(t)] \geq c.$$

That is, the payoff accumulated by  $\tau_1$  never falls to less than  $c$  of that accumulated by any other admissible plan  $\tau \in \mathcal{Z}$ , no matter what strategy the opponent chooses (even if that other plan by happenstance hits upon a good opposing strategy in its first trial). The smaller  $c$  is, the less stringent the criterion and, in general, the larger the number of plans satisfying the criterion. The usefulness of this criterion and the kinds of plans satisfying it will be discussed at length later (see especially the discussion of Samuel's algorithm in section 7.3); for now it is sufficient to notice that: (i) the criterion depends upon the accumulation function  $U_{\tau,E}(t)$ , (ii) for a given opposing strategy  $E$ , the lower the efficiency of a plan in accumulating payoff in relation to other plans  $\tau \in \mathcal{Z}$ , the smaller  $c$  becomes, and (iii) the rating of a plan will be determined by its performance against the opposing strategy which gives it the most difficulty.

Even when it is known that the opponent has selected a single pure strategy, there is a wide range of sophistication of adaptive plans. One class of simpler plans, the payoff-only plans, proves to be quite instructive because it sets a nontrivial lower bound on the performance of more sophisticated plans and it can be analyzed in some detail. In this context, a payoff-only plan ranks strategies it has tried according to the payoff obtained, and it generates new trial strategies on the basis of (selected parts of) this information alone (see section 7.3). More sophisticated plans use the large amounts of information generated during plays of the game, information concerning configurations encountered and the sequence in which they occur (see chapter 8). Obviously a plan which makes proper use of this additional information should do no worse than a payoff-only plan (since the sophisticated plan can reduce its operation to that of a payoff-only plan by ignoring the additional information), and there are certainly situations in which the information will enable the plan to accumulate payoff at a greater rate than a payoff-only plan.

The other extreme from a fixed opposing pure strategy occurs when any sequential mix of strategies is presumed possible on the part of the opponent. The object then (following von Neumann 1947) is usually to minimize the maximum

loss (negative of the payoff) the opponent can impose. It is interesting that often (checkers, chess, go) this minimax strategy is a pure strategy. Thus, although the payoff may vary on successive trials of the same strategy, the plan can still restrict its search to pure strategies in such cases. In more general situations, however, the plan will have to employ stochastic mixtures of pure strategies and, if it is to exploit its opponents maximally, it will even associate particular mixtures with particular kinds of opponents (assuming it is supplied with enough information to enable it to identify individual opponents).

Considered in the  $(\mathcal{S}, \mathcal{E}, \chi)$  framework, the strategies become the elements of the domain of action  $\mathcal{Q}$  and the plans for employing these strategies become elements of  $\mathcal{J}$ . The set of admissible environments  $\mathcal{E}$  depends upon the particular case considered. If it is known that the opponent has chosen a single pure strategy, then the set of admissible environments  $\mathcal{E}$  is given by the set of pure strategies. The criterion  $\chi$  for ranking the plans is then built up from the unique payoff determined by each pair of opposing pure strategies, the example given being the "gambler's ruin" criterion

$$\min_{\mathcal{E} \in \mathcal{E}} \max_{\tau \in \mathcal{J}} [U_{\tau, \mathcal{E}}(t)/U_{\tau^*, \mathcal{E}}(t)].$$

In the more complicated cases, the set of environments is enlarged, ultimately including plans over  $\mathcal{Q}$ ; however, the accumulation functions  $U_{\tau, \mathcal{E}}(t)$  are still defined and criteria such as the "gambler's ruin" criterion can still be used to rank the plans in  $\mathcal{J}$ .

Once again, as in the previous two illustrations, the large size of  $\mathcal{Q}$  and the complex relation of its elements to performance constitute a major barrier to improvement. Section 7.3 specifically discusses the role of adaptive algorithms in game strategy spaces defined in the manner of Samuel. In addition, the necessity of using non-payoff information generated *during* the play of more complex games presents special difficulties. This latter problem is addressed in section 8.4 as an elaboration of the concepts and techniques developed in the earlier chapters.

#### Summarizing:

$\mathcal{Q}$ , strategies for the game.

$\Omega$ , dependent upon the way strategies are represented; genetic operators will function if descriptors are used so that each strategy is designated by a string of descriptor values (see the predictive modeling technique of the next section for suggestions concerning operations on the strategy *during* the play; section 8.4 extends these ideas).

$\mathcal{J}$ , plans for testing strategies.

$\mathcal{E}$ , the strategic options open to the opponent; in simple cases, the set of pure strategies.

$x$ , a ranking of plans using the cumulative payoff functions, the “gambler’s ruin” criterion being an example.

#### 4. SEARCHES, PATTERN RECOGNITION, AND STATISTICAL INFERENCE

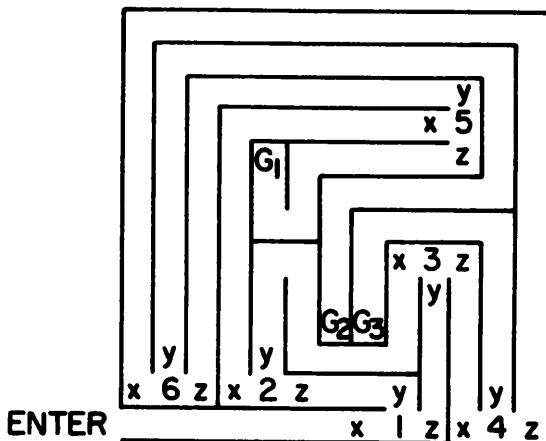
Searches occur as the principal element in most problem-solving and goal-attainment attempts, from maze-running through resource allocation to very complicated planning situations in business, government, and research. Games and searches have much in common and, from one viewpoint, a game is just a search (perturbed by opponents) in which the object is to find a winning position. The complementary viewpoint is that a search is just a game in which the moves are the transformations (choices, inferences) permissible in carrying out the search. Thus, this discussion of searches complements the previous discussion of games.

In complicated searches the attainable situations  $S$  are not given explicitly; instead some initial situation  $S_0 \in S$  (position in a maze, collection of facts, etc.) is specified and the searcher is given a repertory of transformations  $\{\eta_i\}$  which can be applied (repeatedly) in carrying out the search. As in the case of games, a tree is a convenient abstract representation of the search. For searches, each edge corresponds to a possible transformation  $\eta_i$  and the traverse of any path in the tree corresponds to the application of the associated sequence of transformations. The vertex at the end of a path extending from the initial vertex corresponds to the situation produced from the initial situation by the transformations associated with the path. The difficulty of solving a problem or attaining a goal is primarily a function of the size of the search tree and the cost of applying the transformations. In most cases of interest the trees are so vast that hope of tracing out all alternative paths must be abandoned. Somehow one must formulate a search plan which, over a wide range of searches, will act with sufficient efficiency to attain the goal or solve the problem.

A typical search plan (see Newell, Shaw, and Simon’s [1959] GPS or Samuel’s [1959] procedure) involves the following elements:

- (i) An (ordered) set of feature detectors  $\{\delta_i : S \rightarrow V_i\}$ , where  $V_i$  is the range of readings or outputs of the  $i$ th detector}. Typically, each detector is an algorithm which, when presented with a “scene” or situation, calculates a number; if the number is restricted to 0 or 1, it is convenient to think of the algorithm as detecting the presence or absence of a

property (cf. the simple artificial adaptive system of section 1.3). The need for detectors arises from the overwhelming flow of information in most realistic situations; the intent is to filter out as much "irrelevant" information as possible.



At each choice point, 1 through 6, there is to be a sign associated with each of the 3 possible directions  $x$ ,  $y$ ,  $z$ . If the symbol " $\wedge$ " occurs at the top of a sign the associated corridor belongs to the shortest path from the entrance to the goal; on the other hand, if the symbol " $\vee$ " occurs at the bottom of a sign the associated corridor is to be avoided. Either symbol may be dark on a light background or vice versa. Thus, reduced to a 4-by-4 array of sensors (see section 1.3), either of the configurations



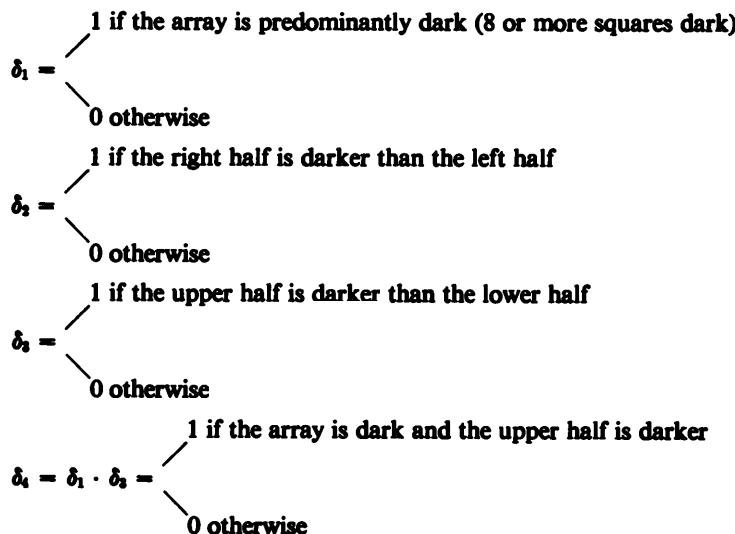
indicates the direction of the goal. Each experiment involves a set of signs indicating uniquely the shortest path to one of the three possible goals  $G_1$ ,  $G_2$ ,  $G_3$ . In the terminology of section 3.4, the state at each choice point is given by the triple of signs there. That is,

$$S = \{\text{triples of } 4\text{-by-}4 \text{ arrays}\}$$

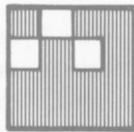
$$\{\eta_1, \eta_2, \eta_3\} = \{"\text{follow direction } x", "\text{follow direction } y", "\text{follow direction } z"\}$$

*Fig. 5. A simple search setting: a maze with six choice points*

The device is supplied with four detectors (see section 1.3)



Thus the array



would be assigned the quadruple of detector values (1,1,0,0). (Note the large reduction in the number of situations to be evaluated—there are  $2^8 \cong 64,000$  different arrays but only  $2^4 = 16$  detector value quadruples.)

The threshold devices of interest are specified by

$$f(S) = \sum_{i=1}^4 w_i \delta_i(S)$$

where each  $w_i$  can be chosen from the set  $\{-2, -1, 0, 1, 2\}$ . A 4-by-4 array  $S$  is assigned to  $C^+$  if  $f(S) \geq \frac{1}{2}$ , otherwise it is assigned to  $C^-$ .

To determine what transformation from  $\{\eta_1, \eta_2, \eta_3\}$  is to be invoked at choice point  $j$  each of the three arrays  $S_{jz}, S_{jy}, S_{jx}$  is submitted to  $f$ . If exactly one of  $S_{jz}, S_{jy}, S_{jx}$  is assigned to  $C^+$  the corresponding corridor is followed, otherwise a corridor is chosen at random (and, presumably, the adaptive plan is invoked to modify the weights because of the lack of a unique prediction).

*Fig. 6. A threshold device for the setting of figure 5*

Setting I: The goal is at  $G_1$ , the signs at choice point 1 are



at choice point 2



and so on (i.e., the shortest path is indicated by dark symbols on a light background).

Setting II: The goal is at  $G_3$ , the signs at choice point 1 are



and at choice point 2 they are the same as in Setting I except for



If the shortest path to the goal were always indicated as in Setting I, i.e., with dark symbols on a light background, then the function  $f'(S) = \delta_3(S)$  (i.e.,  $w_3 = 1, w_1 = w_2 = w_4 = 0$ ) would always suffice for following the path. Notice, however, that in Setting II  $f'$  assigns exactly the same set of values (0,1,0) at point 1, indicating that  $f'$  does not distinguish the two settings. But, in Setting I  $f'$  assigns (1,0,0) at point 2, while in Setting II  $f'$  assigns (0,0,0) at point 2. Thus, starting from the same initial state (0,1,0) and invoking the same response  $\eta_v$ ,  $f'$  arrives at two *different* states. Changing the weight assigned to  $\delta_4$  cannot correct the difficulty. This is a clear indication that the set of detectors ( $\delta_3$  in this case) is inadequate.

A quick check of the possibilities shows that consistently correct choices in the two settings can be achieved only by assigning a nonzero weight to  $\delta_4$ , which is a nonlinear combination of  $\delta_1$  and  $\delta_2$ . The function  $f''(S) = \delta_1 + \delta_2 - 2\delta_4$  then performs correctly in both settings and, in fact, performs consistently with any proper sequence of signs.

*Fig. 7. Some searches using the devices of figure 6 in the settings of figure 5*

- 
- (ii) An evaluator. The evaluator calculates an estimate of the “distance” of any given situation from the goal, using the detector outputs (an ordered set of real numbers) produced by that situation. The estimates are supposed to take the costs of the transformations, etc., into account; that is, the “distances” are usually weighted path lengths, where the paths involved are (conjectured) sequences of transformations leading from given situations to the goal. The intent is to use these estimates to determine which transformations should be carried out next. An evaluation is made of each of the situations which could be produced from the current one by the application of allowed (simple sequences of) transformations, and then that (sequence of) transformation(s) is executed which leads to the new situation estimated to be “nearest” the goal.
  - (iii) Error correction procedures. Before the search plan has been tried, the detectors and evaluator must be set up in more or less arbitrary fashion, using whatever information is at hand. The purpose of the error correction procedures is to improve the detectors and evaluators as the plan accumulates data. The shorter term problem is that of evaluator improvement. A typical procedure is to explore the search tree to some distance ahead of the current situation, either actually or by simulation, evaluating the situations encountered for their estimated distances from the goal. The evaluation of the situation estimated to be “nearest” the goal is then compared with the evaluation of the current situation and the evaluator is modified to make the estimates consistent. This “lookahead” procedure decreases the likelihood of contradictory distance estimates at different points on the same path. (A similar procedure can be carried out without lookahead using predictors to make predictions about future situations, subsequently modifying the predictors to bring predictions more in line with observed outcomes.) As a result, the consistency of the evaluator is improved with each successive evaluation. At the same time, in most searches, the difficulty of estimating the distance to the goal decreases as the goal is approached, becoming perfect when the lookahead actually encounters the goal. Thus increasing the consistency ultimately increases the relevance of the evaluator.

There is, however, a caveat. If the set of detectors is inadequate, for whatever reason, the improvement of the evaluator will be blocked. This raises the broad issue of pattern recognition, for the set of detectors is, of course, meant to enable

the plan to recognize critical features for goal-attainment. The plan must be able to classify each situation encountered according to the goal-directed transformation which should be applied to it. The long-term problem is that of determining whether the set of detectors is adequate to this task. Important shortcomings are indicated when, from application of identical transformations to situations classed as equivalent by the detectors, situations with critically different evaluations result. When this happens, the detectors have clearly failed to distinguish some feature which makes a critical difference as far as the transformations are concerned. The object, then, is to generate a detector which gives different readings for the previously indistinguishable situations. Among the obvious candidates are modifications of the detectors which made the distinctions *after* the transformations were applied. Usually simple modifications will enable such detectors to make the distinction *before* the transformation as well as *after*.

We can look at this whole problem in another way, a way which makes contact with standard definitions in the theory of probability. Assume that the search plan assigns to each transformation  $\eta$  a probability dependent upon the observed situation. That is, if  $S_a$  is the current situation, then each situation  $S_b \in S$  can be assigned a conditional probability of occurrence  $p_{ab}$ , where  $p_{ab}$  is simply the sum of the probabilities of all transformations leading from  $S_a$  to  $S_b$ . (It may, of course, be that there are no transformations of  $S_a$  to  $S_b$ , in which case  $p_{ab} = 0$ .) A sequence of trials performed according to the probabilities  $p_{ab}$  is a *Markov chain*, the outcome of each trial being a *random variable* (dependent upon the outcome of prior trials). The *sample space* underlying this random variable is the set of situations  $S$ . Let us assign a measure of utility or relevance to each of these situations. (For example, goals could be assigned utility 1 and all other situations utility 0, or some more complicated assignment ranking goals and intermediate situations could be used.) Then, formally, the function  $W$  making this assignment is also a random variable. Accordingly, we can assign an expected utility to the random variable representing the outcome of each trial in the Markov chain. In these terms, the plan continually *redefines* the Markov chain (by changing the transformation probabilities). It attempts in this way to increase the average (over time) of the expected values of the sequence of random variables corresponding to its trials.

The role of detectors here is, as already suggested, reduction of the size of the sample space and simplification of the search. More formally, consider a set of  $n$  detectors (not necessarily all those available),  $H = \{\delta_1, \dots, \delta_n\}$ , where  $H$  is arbitrarily ordered. The detectors in  $H$  assign to each  $S \in S$  an  $n$ -tuple of readings  $(v_1, \dots, v_n)$  belonging to the direct product

$$\prod_{i=1}^n V_i.$$

In general there will be many situations producing a given set of detector readings; let  $\mathcal{S}(v_1, \dots, v_n)$  be the set of situations in  $\mathcal{S}$  producing the particular  $n$ -tuple of readings  $(v_1, \dots, v_n)$ . In probabilistic terms,  $\mathcal{S}(v_1, \dots, v_n)$  is an *event* defined on the sample space  $\mathcal{S}$ . Events themselves can be treated as random variables. (In fact, an occurrence of the situation  $S$  can be construed as the occurrence of all the events of which it is an instance.) Moreover, the function  $W$  assigning values to elements of  $\mathcal{S}$  can be restricted to the event  $\mathcal{S}(v_1, \dots, v_n)$  so that it becomes a random variable  $W(v_1, \dots, v_n)$  over  $\mathcal{S}(v_1, \dots, v_n)$ . As such  $W(v_1, \dots, v_n)$  has a well-defined expected value  $\bar{W}(v_1, \dots, v_n)$  over  $\mathcal{S}(v_1, \dots, v_n)$ .

This probabilistic view of search plans is closely related to statistical inference based on sampling plans. The estimation of  $\bar{W}(v_1, \dots, v_n)$  from observation of a few samples drawn from  $\mathcal{S}(v_1, \dots, v_n)$  is a standard problem of statistical inference. We can think of a subset of detectors  $H$  as detecting one kind of critical feature when the corresponding  $\bar{W}(v_1, \dots, v_n)$  is greater than  $\bar{W}$ , where  $\bar{W}$  is the average value of the random variable  $W$  over the sample space  $\mathcal{S}$ . Search plans go further in attempting to infer something of the value of  $\bar{W}(v_1, \dots, v_n)$  for  $\mathcal{S}(v_1, \dots, v_n)$  which have not been sampled. For example,  $\mathcal{S}(v_1, v_2)$  is contained in both  $\mathcal{S}(v_1)$  and  $\mathcal{S}(v_2)$ ; often it is possible to infer something of  $\bar{W}(v_1, v_2)$  from knowledge of  $\bar{W}(v_1)$  and  $\bar{W}(v_2)$ , though not necessarily by standard statistical techniques.

The earlier concern with distinguishability is also directly stated in these terms: Let  $\delta(t)$  be the particular  $n$ -tuple of detector readings at time  $t$  ( $\delta(t) \in \Pi_i V_i$ ) and let  $f: \Pi_i V_i \rightarrow \{\eta\}$  be a search plan. That is,  $f$  is a prescription which specifies, for each set of detector readings, a transformation. The object of the search plan is to transform the current situation into one of high utility. But, for this to be possible, the effects of the transformations must be reliably indicated by the detectors. In particular, consider  $S_1$  and  $S_2 \in \mathcal{S}(v_1, \dots, v_n)$ , so that at  $t = 1$  either would show the same reading  $\delta(1) = (v_1, \dots, v_n) \in \Pi_i V_i$ . The plan  $f$  specifies the action  $\eta(1) = f(\delta(1))$ , and this in turn produces a new detector reading  $\delta(2)$ . The whole procedure is iterated to yield a sequence of pairs  $\langle [\delta(1), f(\delta(1))], [\delta(2), f(\delta(2))], \dots, [\delta(t), f(\delta(t))] \rangle$ . The requirement on distinguishability is simply that, using the information provided by the detectors,  $f$  *reliably* transforms  $S_1$  and  $S_2$  into situations  $S'_1$  and  $S'_2$ , respectively, for which  $W(S'_1) \cong W(S'_2)$ . (Notice that this is a much weaker requirement than would be necessary for a completely “autonomous” model wherein future situations would be wholly predictable on the basis of  $\delta(1)$  without any further information from the environment. That is, in an autonomous model, knowledge of  $\delta(1)$  and  $\eta(1), \dots, \eta(t)$  must suffice to determine  $\delta(t+1)$ . This requirement for “autonomy”—technically a requirement that the detectors induce a homomorphism—can be quite difficult to meet and, for intricate

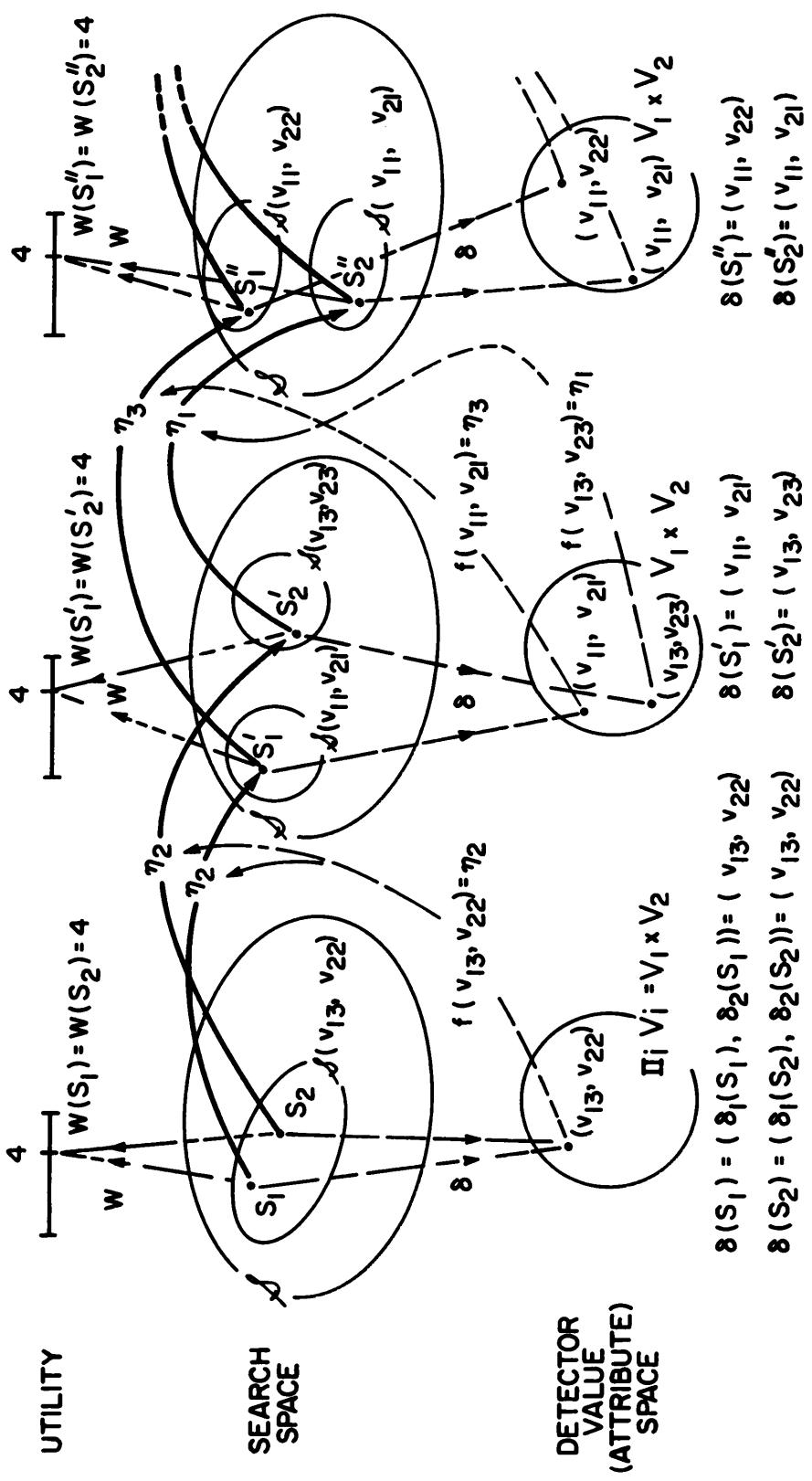


Fig. 8. A utility-consistent model using detector outputs to determine successive transformations

situations, there may be no nontrivial homomorphisms.) The requirement on distinguishability permits the detectors to lump together situations having approximately equal expected utilities. This permits us to construct new, much smaller Markov chains based on events of interest. With this interpretation, the object of the adaptive plan  $\tau$  is to test different sets of detectors,  $H$ , and search plans,  $f$ .

The error correction procedures for the detectors and the evaluator can be coordinated via a model of the search environment. In complex search environments, it is the detectors which make a model possible. From considerations of storage alone, a model would be impossible if, for each observed transformation, the initial situation and its successor had to be recorded in full. If the system records just the effect of transformations on the detector readings, it can be reduced to manageable proportions. Stated another way, the state space of the search environment is reduced to the manageable space of detector readings, and the effects of the transformations are observed on this reduced space. The construction of the model proceeds as data accumulate. When new detectors are required to increase distinguishability, an augmented model can be built around the old model as a nucleus. In particular, when a new detector is added to the set of detectors, this does not affect the data or the part of the model concerned with the old set of detectors. The task is to add information about the effect of the transformations upon the new detector, particularly in those situations that were causing difficulty. Once a model is available, it can be used with the evaluator to generate predictions and these can be checked against the outcome of that segment of the search. The resulting error indications, together with simulated lookahead, can then be used to improve the consistency of the evaluator.

The sophistication of a model-evaluator search plan can only be justified when repeated searches must be made in the same overall search environment. As a prototype, we can consider an environment which (i) is complicated enough to make the exact recurrence of any particular situation extremely unlikely, but (ii) is regular enough to exhibit critical features (patterns) "pointing the way" to goals. The object of a plan, then, is to search out a succession of goals, improving its performance by incorporating the critical features in its detector-evaluator scheme.

The overall objective of a formal study of this area is to find a plan which, when presented with any of a broad range of complex search environments, rapidly increases its search efficiency by extracting and exploiting critical features. Considered within the  $(\mathfrak{J}, \mathcal{E}, \chi)$  framework, the domain of action  $\mathfrak{G}$  of a search plan  $\tau \in \mathfrak{J}$  consists of the various combinations of detectors, models, and evaluators that the plan can generate. Usually these will all be specified as algorithms or

programs in some common formal language (see chapter 8). A search plan  $\tau \in \mathfrak{I}$  thus amounts to a data-dependent algorithm for modifying the combination of detectors, model, and evaluator along the lines indicated above. The outward effect, at each point in time, of the combination produced by the search plan is a transformation  $\eta$  in the search environment. The range of the plan's action at each moment is therefore circumscribed by the set of possible transformations  $\{\eta\}$ . The set of admissible environments  $E \subset \mathcal{E}$  consists of the set of search environments over which the search plan is expected to operate, each element  $E$  being presentable as a tree generated by the possible transformations. Let  $U_{\tau, E}(t)$  be the cost in  $E$  of the transformations applied by  $\tau$  through time  $t$ . If  $n_{\tau, E}(t)$  is the number of goals achieved to time  $t$ , then  $c_{\tau, E}(t) = U_{\tau, E}(t)/n_{\tau, E}(t)$  is the average cost to time  $t$  of each goal in  $E$ . A conservative measure of a plan's performance over all of  $\mathcal{E}$  would then be

$$\text{lub}_{E \in \mathcal{E}} \lim_{t \rightarrow \infty} c_{\tau, E}(t)$$

which yields the criterion  $\chi$  wherein plan  $\tau$  has a higher rank than plan  $\tau'$  if it is assigned a lower number by the above measure. Suggestions for a model-evaluator plan, based on the genetic algorithms of chapter 6 and capable of modifying its representations, are advanced in section 8.4.

#### Summarizing:

- $\mathfrak{Q}$ , [probabilistic] Markov chains induced by the sets of conditional probabilities  $\{p_i(S)$ , the probability of applying transformation  $\eta_i$  to situation  $S \in \mathcal{S}\}$ ; [general] admissible detector-evaluator-model combinations.
- $\Omega$ , [probabilistic] rules for modifying the conditional probabilities  $\{p_i(S)\}$ ; [general] "lookahead" error correction, detector generation, and model revision procedures.
- $\mathfrak{J}$ , algorithms for applying operators from  $\Omega$  to  $\mathfrak{Q}$  using information about the (sampled) average cost of goal attainment and (in the general case) errors in prediction ("lookahead") and observed inadequacies in detectors.
- $\mathfrak{E}$ , the set of search environments characterized by search trees along with a transformation cost function  $\mu_E(\eta_i, S)$  giving the cost of applying  $\eta_i$  in situation  $S \in \mathcal{S}$ .
- $\chi$ , the ranking of plans in  $\mathfrak{I}$  according to performance measures such as

$$\text{lub}_{E \in \mathcal{E}} \lim_{t \rightarrow \infty} c_{\tau, E}(t).$$

## 5. CONTROL AND FUNCTION OPTIMIZATION

The fact that we need time to determine the minimum of the [performance] functional or the optimal [control] vector  $c^*$  is sad, but unavoidable—it is a cost that we have to pay in order to solve a complex problem in the presence of uncertainty. . . . adaptation and learning are characterized by a sequential gathering of information and the usage of current information to eliminate the uncertainty created by insufficient *a priori* information.

Tsyplkin in *Adaptation and Learning in Automatic Systems* (p. 69)

In the usual version, a controlled process is defined in terms of a set of variables  $\{x_1, \dots, x_k\}$  which are to be controlled. (For example, a simple process of air conditioning may involve three critical variables, temperature, humidity, and air flow.) The set of states or the *phase space* for the process,  $X$ , is the set of all possible combinations of values for these variables. (Thus, for an air conditioning process the phase space would be a 3-dimensional space of all triples of real numbers  $(x_1, x_2, x_3)$  where the temperature  $x_1$  in degrees centigrade might have a range  $0 \leq x_1 \leq 50$ , etc.) Permissible changes or *transitions* in phase space are determined as a function of the state variable itself and a set of control parameters  $C$ . Typically  $X$  is a region in  $n$ -dimensional Euclidean space and the control parameters assume values in a region  $C$  of an  $m$ -dimensional space. Accordingly, the equation takes the form of a “law of motion” in the space  $X$ ,

$$dx/dt = f(X(t), C(t)), \quad \text{where } X(t) \in X, C(t) \in C.$$

Often  $X$  will have several components  $X_1, \dots, X_k$  following distinct laws  $f_1, \dots, f_k$  so that

$$f(X(t), C(t)) = (f_1(X_1(t), C(t)), \dots, f_k(X_k(t), C(t))).$$

For example, given a pursuit problem with a moving target having coordinates  $X_2(t)$  at time  $t$ ,  $f_2(X_2(t), C(t))$  would be the law of motion of the target while  $f_1(X_1(t), C(t))$  would determine the pursuit curve. If some component, say  $X_t$ , represents time, then  $f_t(X_t(t), C(t)) = t$  and the law of motion becomes an explicit function of time.

When a rule or *policy A* is given for selecting elements of  $C$  as a function of time, a unique trajectory  $((X(t), C(t)))$  through  $X \times C$  is determined by the law of motion  $f$ . The object is to select a policy  $A$  for minimizing a given function  $J$  which assigns a performance or cost to each possible trajectory  $((X(t), C(t)))$ . In practice, the function  $J$  is usually determined as the cumulation over time of some instanta-

neous cost rate  $Q(X(t), C(t))$ ; i.e.,  $J((X(t), U(t))) = \int Q(X(t), C(t)) dt$ . Typically, the cost function is derived from an explicit control objective such as attainment of a target state or a target region in minimal time or minimization of cumulative error. (Error is defined in terms of a measure of distance imposed on the phase space; the distance of the current state from the target region is the current error.) Control is thus a continuing search in phase space for the (usually moving) target or goal—as such the considerations of the preceding illustration are directly relevant. In the formulation of the pursuit problem stated above a natural measure of the cost of pursuit over some interval  $T$  would be the change in distance between target and pursuer divided by the fuel expenditure (with suitable conventions for trajectories where the distance does not decrease).

Although the controlled process is defined above in terms of continuous functions, discrete finite-state versions closely approximating the continuous version almost always exist. Indeed, if the problem is to be solved with the help of a digital computer, it must be put in finite-state form. Because the framework we are using is discrete, we will reformulate the problem in discrete form. The law of motion is given by

$$X(t + 1) = f(X(t), C(t)),$$

and the cumulative cost for a given trajectory over  $T$  units of time is given by

$$J((X(1), C(1)), \dots, (X(T), C(T))) = \sum_{t=1}^T Q(X(t), C(t)).$$

If we look at the controlled process in the  $(\mathcal{S}, \mathcal{E}, \chi)$  framework we see that the law of motion  $f$  determines the environment of the adaptive system. A problem in control becomes a problem of adaptation when there is significant uncertainty about the law of motion  $f$ ; that is, it is only known that  $f \in \{f_\mathcal{S}, E \in \mathcal{E}\}$ . Such problems are generally unsolvable by contemporary methods of optimal control theory (cf., for example, the comments of Tsypkin [1971, p. 178]). Clearly under such circumstances the adaptive plan will have to try out various policies in an attempt to determine a good one. To fix ideas, let us assume that each policy  ${}^1A \in \mathcal{G}_1$  can be assigned an average or expected performance  $\bar{Q}({}^1A, f)$  for each possible  $f$ . Moreover let us assume that this average can be estimated as closely as desired by simply trying  ${}^1A$  long enough from any arbitrary time  $t$  onward. The object then is to search for the policy in  $\mathcal{G}_1$  with the best average performance  $\bar{Q}$ , exploiting the best among *known* possibilities at each step along the way.

A control policy  ${}^1A \in \mathcal{G}_1$  generates a sequence of control parameters  $(C(t))$ . Different trials of the policy  ${}^1A$ , say at times  $t_1, t_2, \dots, t_k$ , will in general elicit different costs  $Q(t_1), Q(t_2), \dots, Q(t_k)$ . However, the  $(\mathcal{S}, \mathcal{E}, \chi)$  framework requires

that each  $A \in \mathcal{G}$  be assigned a unique cost  $\mu_E(A)$ . To satisfy this requirement we can let  $\mathcal{G} = \mathcal{G}_1 \times \mathfrak{N}$  where  $\mathfrak{N}$  is the set of natural numbers  $\{1, 2, 3, \dots\}$ . Then unique elements of  $\mathcal{G}$ , namely  $(^1A, t_1), (^1A, t_2), \dots, (^1A, t_k)$ , correspond to the successive trials of  $^1A$  and the cost  $Q(t_i)$  of trial  $t_i$  can be assigned as required,

$$\mu_E(A) = \mu_E((^1A, t_i)) = Q(t_i).$$

An adaptive plan  $\tau$  will modify the policy at intervals on the basis of observed costs. With the definition of  $\mathcal{G}$  just given this means that, if  $^1A$  is tried at time  $t$  and is to be retained for trial at time  $t + 1$ ,

$$\tau(J(t), \mathcal{G}(t)) = \tau(I(t), (^1A, t)) = (^1A, t + 1);$$

on the other hand, if a new policy  $^1A'$  is to be tried,

$$\tau(J(t), \mathcal{G}(t)) = \tau(I(t), (^1A', t)) = (^1A', t + 1).$$

A sophisticated adaptive plan will probably retain a measure of the average performance of various policies tried so that  $\mathcal{G}$  would be further extended by a component  $\mathfrak{M}$  (see section 2.2) to  $\mathcal{G} = \mathcal{G}_1 \times \mathfrak{N} \times \mathfrak{M}$ . A still more sophisticated plan will progressively reduce uncertainty about the environment by deliberately selecting elements of  $C$  to elicit critical information, perhaps constructing a model of  $f_E$ . Then by exploiting predictions of the model  $\tau$  can adjust the sequence  $\langle C(t) \rangle$  to better performance as measured by the function  $J$ . At this level the illustration concerning searches, pattern recognition, and statistical inference applies in toto. If the plan is to be a payoff-only plan, then

$$I(t) = \mu_E(\mathcal{G}(t)) = Q(t),$$

and  $\mathfrak{M}(t + 1)$  is updated by using  $Q(t)$  in a recalculation of the average performance of  $\mathcal{G}_1(t)$ .

Finally the function  $J$  determines a ranking for every control sequence  $\langle C(t) \rangle$ , whether or not it is generated by a single policy. That is, an adaptive plan  $\tau$  confronted with a law of motion  $f_E$  may try several policies, thereby generating a control sequence which no single  $^1A \in \mathcal{G}_1$  could generate. However every control action  $C(t)$  has a definite cost  $Q(t)$ . Thus the trajectory  $\langle C(t) \rangle$  through  $C$  generated by  $\tau$  can be ranked according to  $J$ . In this way  $J$  determines a criterion for ranking any  $\tau \in \mathfrak{J}$  in any  $E \in \mathcal{E}$ . As a specific example, consider the case where the object is minimization of cumulative error. By assigning maximum payoff to the target region and reducing the payoff of other states in proportion to the associated error, the performance of a plan  $\tau$  can be measured in terms of the cumulative payoff function  $U_E(\tau, t)$ . The greater  $U_E(\tau, t)$  the less the cumulative error to time  $t$ .

The foregoing discussion can be made applicable to function optimization by so arranging it that a single "trial" of a policy  ${}^1A$  produces a sufficient estimate of its average performance  $\bar{Q}({}^1A, f)$ . For instance the policy could be repeatedly tried over some extended interval of time which would then be taken to be a single time-step in the discrete formulation. In any case let us assume that each  ${}^1A \in \mathcal{G}_1$  has a unique value  $\mu_E({}^1A) = \bar{Q}({}^1A, f)$  which can be determined (with sufficient accuracy) in a single time-step. Moving the problem of estimating  $\bar{Q}$  into the background in this way, reduces the control objective to finding the optimum of the function  $\mu_E$ .

If the elements of  $\mathcal{G}_1$  are represented as points in an  $n$ -dimensional Euclidean space  $\mathbb{R}^n$  the problem becomes one of optimizing an  $n$ -dimensional (nonlinear) real function. For example the elements of  $\mathcal{G}_1$  can be represented as strings of length  $n$  over some basic alphabet  $\Sigma$ . Since  $\Sigma$  can be recoded as a subset of  $\{0, 1\}^m$ , where  $m$  is the first integer greater than  $\log_2(\text{card } \Sigma)$ , this can be looked upon as optimization of an  $n$ -dimensional function having  $m$ -place binary fractions as arguments. (Thus, if  $\Sigma = \{\sigma_0, \sigma_1, \sigma_2, \dots\}$ , the coding  $\sigma_0 \leftrightarrow .00\dots00$ ,  $\sigma_1 \leftrightarrow .00\dots01$ ,  $\sigma_2 \leftrightarrow .00\dots10$ , etc., can be used. Then with  ${}^1A \in \mathcal{G}_1$  represented by the string  $\sigma_2\sigma_2\sigma_1$ , say, the argument of  $\mu_E$  becomes  $(.00\dots010, .00\dots010, .00\dots001)$ .)

With this arrangement an adaptive plan  $\tau$  uses its operators to generate a sequence of points  $\mathcal{G}_1(1), \mathcal{G}_1(2), \mathcal{G}_1(3), \dots$  converging to an optimum, much in the manner of standard iterative procedures. The adaptive approach, however, suggests important differences in what information from prior calculations should be retained (in  $\mathfrak{M}(t)$ ) in preparation for generation of the next point  $\mathcal{G}_1(t+1)$ . In particular certain adaptive plans proceed simultaneously and efficiently with global and local optimization of  $\mu_E$ . (See chapters 4 and 5 for basic techniques.)

In the case of function optimization, high-dimensionality and nonlinearity of the function to be optimized ( $\mu_E$ ), in all but a few special cases, constitute insurmountable barriers to standard optimization algorithms. In the general control problem there is the added difficulty of nonstationarity. The schemata concept (first interpreted in function optimization terms in chapter 4, pp. 70-71) and the algorithms based upon it (chapter 6) provide specific remedies for the first two problems. The latter problem is substantial and difficult—it is discussed in section 9.3.

#### Summarizing:

$\mathcal{G}$ , [control] a set having as its basic component the set of admissible control policies  $\mathcal{G}_1$  augmented by a memory component  $\mathfrak{M}$  and a set of time subscripts  $\pi$  so that  $\mathcal{G}(t) = (\mathcal{G}_1(t), t, \mathfrak{M}(t))$ ; [function optimization] the domain of the function  $\bar{Q}$  to be optimized.

$\Omega$ , [control] procedures for generating a new control policy from some set of given policies; [function optimization] procedures for generating a new point in the domain of  $\bar{Q}$  from some set of given points. includes sections 3 to 5, plans for applying procedures from  $\Omega$  to generate new policies [control] or points [function optimization] on the basis of observations.  
 $\mathcal{E}$ , an indexing set corresponding to the initial uncertainty about the “law of motion” or the function to be optimized.  
 $\chi$ , the extension of the ranking  $J$  on control sequences to the plans inducing the sequences, assigning  $\tau \in \mathcal{J}$  the average (or minimum, etc.) ranking over the uncertainty indexed by  $\mathcal{E}$ .

## 6. CENTRAL NERVOUS SYSTEMS

Behavior is primarily adaptation to the environment under sensory guidance. It takes the organism away from harmful events and toward favorable ones, or introduces changes in the immediate environment that make survival more likely.

Hebb in *A Textbook of Psychology* (pp. 44–45)

I introduce this last example of an adaptive system with some hesitation. Not because the central nervous system (CNS hereafter) lacks qualifications as an adaptive system—on the contrary, this complex system exhibits a combination of breadth, flexibility, and rapidity of response unmatched by any other system known to man—but because there is so little prior mathematical theory aimed at explaining adaptive aspects of the CNS. Even an intuitive understanding of the relation between physiological micro-data and behavioral macro-data is only sporadically available. Perforce, mathematical theories enabling us to see some overall action of the CNS as a consequence of the actions and interactions of its parts are, when available at all, in their earliest formative stages.

Here, more than with the other examples, the initial advantage of the formal framework will be restatement of the familiar in a broader context. The best that can be hoped for at this stage is an occasional suggestion of new consequences of familiar facts: Without the advantages of a deductive theory, statements made within the framework can do little more than provide an experimenter with guideposts and cautions, suggesting possibilities and impossibilities, phenomena to anticipate, and conclusions to be accepted warily. This is a preliminary, heuristic stage marking the transition from unmathematical plausibility to the formal deductions of mathematical theory. In common with most heuristic and loose-textured

arguments, it is difficult to eliminate ambiguities and contradictions—as with proverbs, proper application depends upon the intuition of the user. Specific applications of the formalism may arise, but these will probably be in areas of little uncertainty, where theory was not actually required; the formal procedures will be primarily corroborative. Only after considerable effort at this level can we hope for a theory mathematically rigorous and conceptually general—a genuinely predictive theory organizing large masses of data at many levels.

One of the earliest suggestions (or corroborations) from the formal underpinnings of the  $(J, \mathcal{E}, \chi)$  framework is quite fundamental. It can be established that major aspects of the behavior of any very complex system fall outside the explanatory power of simple input-output (*S-R* or switching) theory. This result is a rigorous version of the observation that ongoing activity in a complex system usually depends upon the past history of that system. This dependence, which both psychologists and computer theorists call “memory,” finds its formal counterpart in the notion of state: distinct stimulus-state pairs generally giving rise to different responses. If there are many states (and, by any reasonable definition of state, the CNS has an astronomical number) the same stimulus may give rise to a great many different responses. Thus, observation of stimulus-response pairs will *not* enable us to discover the mode of operation of *any* system with a substantial number of states. For a system as complex as the CNS, such a result can be ignored only to the great detriment of the ensuing theory. It is a corollary of this result that complex systems can act in autonomous fashion, producing continuing response sequences in the absence of new stimulus. Thus, a stimulus may serve only to modify ongoing activity rather than to initiate it. In short, the responses of the CNS cannot be explained wholly in terms of concurrent stimuli.

The  $(J, \mathcal{E}, \chi)$  framework also emphasizes a second important point. An adequate theory must include more than a formal counterpart of the internal processes of the system being studied. The environment (or range of possible environments), the information received therefrom, and the ways the system can affect the environment, must also be represented. Moreover, the criterion  $\chi$  emphasizes the importance of performance “along the way.” The CNS cannot wait indefinitely for “useful” outcomes; some minimal level of ongoing performance is required. (E.g., if food is not obtained with sufficient frequency, death ensues, totally removing the possibility of further goal-oriented behavior.) Such observations are not new, but the  $(J, \mathcal{E}, \chi)$  framework does provide a form for fitting and arranging them, and it lends them emphasis. This at least gives us a fresh look at familiar facts, occasionally suggesting new consequences which might otherwise be overwhelmed in the plethora of macro- and micro-data (behavioral and physiological).

The discussion which follows will be based upon the informal theory of CNS action introduced by Hebb in his signal 1949 work and subsequently importantly enlarged by P. M. Milner (1957) and I. J. Good (1965). I will attempt a brief recapitulation of some of the main assumptions here, with the intention of orienting the reader having some knowledge of the area. This is only one view of CNS processes and the presentation has been kept deliberately simplistic. (E.g., a more sophisticated theory would take account of substantial evidence for distinct physiological mechanisms underlying short-term, medium-term, and long-term memory.) The object is to indicate, in as simple a context as possible, the relevance of the  $(\beta, \delta, \chi)$  framework to understanding the CNS as a means of "adaptation to the environment under sensory guidance." The reader without a relevant background can gain a significant understanding by reading the papers of Milner and Good; a reading of Hebb's excellent textbook (1958) will give a much more comprehensive view.

The basic element of Hebb's theory is the *cell assembly*. It is assumed to exhibit the following essential characteristics. (Comments in parentheses in the presentation of characteristics refer to possible neurophysiological mechanisms):

1. A cell assembly is formed in response to repetitions of some relatively simple critical feature of the sensory input, such as pressure on a particular skin area, a simple odor, a vowel sound, an increase of brightness, a line of particular slope in the visual field, and so on. (From a more physiological point of view, a cell assembly is taken to be a collection of hundreds of neurons interconnected via synapses of high conductivity.) After some assemblies have been formed others may be formed in response to the repetitive actions of already extant (precursor) assemblies.
2. Whenever the critical feature causing the cell assembly's formation subsequently appears in the sensory input, the cell assembly tends to become active. Cell assemblies formed in response to precursor assemblies tend to become active when their precursors are active. (Once a significant number of neurons in the cell assembly attain a high pulse rate, the remaining neurons quickly follow suit because of the highly conductive interconnections. Because the interconnections form many loops, a reverberation results and the neurons tend to remain active for a period of time long compared to the stimulus time.)
3. Cell assemblies exhibit changing positive and negative associations with each other. A cell assembly which is active increases the likelihood of activity in all assemblies with which it is positively associated and de-

creases the likelihood of activity in all assemblies with which it is negatively associated. Positive association between a pair of cell assemblies increases whenever they are active at the same time. Negative association is asymmetrical in that one cell assembly may be negatively associated with a second, while the second is not necessarily negatively associated with the first; this negative association increases each time the first assembly is active and the second is inactive. (The underlying neural assumption here is that, if neuron  $n_2$  produces a pulse immediately after it receives a pulse from neuron  $n_1$ , then  $n_1$  is better able to elicit a pulse from  $n_2$  in the future; contrariwise, if  $n_2$  produces no pulse upon receiving a pulse from  $n_1$ , then  $n_1$  is more likely to inhibit  $n_2$  in the future. It is usually assumed that this process is the result of changing synapse levels. The same process can be invoked in explaining the origin of cell assemblies.) It should be noted that, under this assumption, there is a tendency for cell assemblies to become active in fixed combinations, at the same time actively suppressing alternative combinations. (Because a cell assembly involves only a minute fraction of the neurons in a CNS, a great many can be excited at any instant, different configurations corresponding to different perceived objects, etc.) Temporal association (i.e., probable action sequences) can occur via appropriate asymmetries; e.g., assembly  $\alpha$  can arouse  $\beta$  via positive association while  $\beta$  inhibits  $\alpha$  through negative association. Thus the action sequence is always  $\alpha\beta$ , never the reverse.

4. At any instant the response of the CNS to sensory input is determined by the configuration of active cell assemblies. (Overt behavior such as eye movement, activation of reflexes, release of voluntary muscle sequences, and so on will accompany most sensory events. Via the mechanisms of (3), neurons involved in this behavior will tend to become components of cell assemblies active at the same time. Since pulse trains from the active cell assemblies dominate overall CNS activity, overt behavior will thus be determined by the active configurations. In effect, the sensory input modulates the ongoing activity in the CNS to produce overt behavior.)
5. Cell assemblies involved in temporal sequences yielding “need satisfaction” (satisfaction of hunger, thirst, etc.) have their associations enhanced; the greater the “need,” the greater the enhancement. (“Needs” are internal conditions in the CNS-controlled organism, conditions primarily concerned with survival, which set basic restrictions on CNS

action. In typical environmental situations, overt behavior is required for "need satisfaction," and then the satisfaction is only temporary—the organism consumes the resources involved in order to maintain itself. Innate internal mechanisms in the organism automatically "reward" satisfaction of hunger, thirst, etc., and perhaps some more generalized needs such as curiosity. These "rewards" may be mediated by innately organized neural networks which exhibit increasing activity as a corresponding need increases. Such internally generated stimuli would progressively disturb established configurations and sequences, unless they resulted in reduction of the corresponding need. Ultimately, in the absence of satisfaction, this disruption would cause an increasingly broad search through the organism's behavioral repertory—a kind of hunt through increasingly unusual cell assembly configurations in an attempt to produce an appropriate overt response. Temporal associations of cell assemblies, active when such a disturbance is reduced, would retain their incremented synapse levels. Those active during a period of increasing disturbance would encounter subsequent interference, causing synapse level increments to be transitory. Assemblies having precursors occurring early in "need satisfaction" sequences acquire a particular role. They serve as "leading indicators," becoming active in advance of actual primitive needs; they may serve as "learned needs" [goals]. A hierarchy of precursors of precursors, etc., can provide the system with a hierarchy of "learned needs," some of them quite remote from the primitive needs. That is, assemblies containing substantial segments of the innately organized networks as components, or assemblies closely associated therewith, could give rise to secondary and higher-order "learned needs." The effects of these new assemblies will be much like those generated by the innately organized networks.)

6. An active cell assembly primes cell assemblies associated with it as successors in temporal sequences, making them more likely to be active subsequently. (A neuron producing pulses at a high rate tends to become fatigued, with a consequent drop in pulse rate. A neuron that is being inhibited tends to exhibit less fatigue than normal because of its very low pulse rate. A kind of inhibitory priming results, because the neuron is hyperresponsive once the inhibition ceases. It is also likely that priming occurs by transmission of "priming molecules" through the synapses of active neurons. Priming provides the CNS with expectations and predictions. In effect the system expects and is ready to respond to selected sets

of features from the myriads of possibilities. When primed cell assemblies subsequently become active—i.e., when the corresponding predictions are verified—the associations involved are strengthened by the mechanisms of (3). The resulting network of associations constitutes a model of the environment within the CNS. The model is dynamic in the sense that it takes sensory data as input and primes different temporal sequences on the basis of the model's predictions. Introspection confirms that, for the human CNS, models of the environment are indeed used to compare alternative courses of action. This model is ultimately dedicated to keeping primitive needs fulfilled, but it incorporates “leading indicators,” etc., so that needs rarely become acute enough to determine action directly.)

How can the  $(\mathcal{J}, \mathcal{E}, \chi)$  framework help in analyzing this model? Certain analogies with other processes are suggestive. Individual cell assemblies act, in part, like the detectors in pattern recognizers: they are activated by particular features of the environment, features presumably relevant to the organism's needs. At the same time, the configuration of cell assemblies active at any given time defines the organism's response to the environment. In the terms used earlier, such a configuration is an element of the system's repertory. Assuming that the set of all cell assemblies is fixed (as it might be, to a first approximation, in a mature organism) or, at least, that the potentially available cell assemblies can be enumerated, the set of all possible assembly configurations constitutes the system's repertory of techniques for confronting the environment.

When cell assemblies are in mutual negative association (cross-inhibition), they act much as the alleles of a chromosomal locus; any active configuration can contain at most one of the assemblies, because it will actively suppress the others in the set. Positive associations between cell assemblies which favor particular configurations are analogous to the linkage of coadapted alleles in a chromosome. Indeed there are many potentially fruitful “genetic” analogies. As the CNS gains experience, some assemblies in a cross-inhibited set are likely to be expressed in a broadened range of environmental conditions, at the expense of others in the set—a process suggestive of the evolution of (partial) dominance. Various genetic operators such as crossover and inversion find their counterparts in the ways in which cell assembly associations are modified. Temporal associations correspond to feedback among gene-products and sequential expression of genes. The list can be extended easily.

The needs of the organism define its goals, and ultimately set a criterion on

performance. Just as in games and searches, there is the problem that individual responses do not often directly yield need satisfaction. However, the internal model discussed in connection with property (6) enables the CNS to constantly improve performance in the absence of current need satisfaction. Two kinds of improvement are possible. First of all, cell assemblies typically respond to too broad a range of situations when first formed, yielding inconsistencies in the model. That is, situations activating the same combinations of cell assemblies, and hence the same responses, are followed by radically different outcomes. The remedy here is much like that for inadequacy of detectors discussed in the illustration on searches. Because of the inconsistencies new associations are formed between the cell assemblies involved, causing them to split and recombine so that their responses are more discriminative. (Hebb calls the related procedures fractionation and recruitment.) The second kind of improvement consists in "filling in" the model—generally there will be many situations where no expectations or predictions have been developed. This clearly provides an important role for curiosity. The CNS must experience a wide enough range of situations to provide an adequate repertory of *relevant* temporal sequences. Just as with the coadapted sets of genetics, the basic laws of cell assemblies permit flexible recombination (association) under environmental (sensory) guidance, the actual combinations being influenced by the parts of the model (associations) already extant. In this way a tremendous range of useful procedures can be formed from relatively few elements. More importantly, a single experience then constitutes a trial of a great many relevant associations, just as in genetics a single organism tests a great many coadapted sets. Property (3) assures that many associations will be tested and modified. The ultimate "survival" of various combinations of assemblies is determined by their consistency within the model and their success in contributing to learned or unlearned need satisfaction.

While the foregoing analogies are ready offshoots of the formal framework, the basic task of theory in this area is quite difficult. It must enable one to judge whether proposed mechanisms for CNS operation permit the learning rates, utilization of cues, transfer of learning, etc., that one actually observes. How does the CNS maintain its rapidity and appropriateness of response, while extending its breadth and filling in its model of the environment? Section 8.4 indicates one way in which concepts from the  $(\mathcal{J}, \varepsilon, \chi)$  framework can be brought to bear. In particular, the robustness of reproductive plans, when interpreted in this area, indicates some promising directions, but we even lack good general measures of performance here. A kind of error function based upon average need levels might be interesting for organisms not quite so efficient as man at keeping their primitive needs satisfied. A criterion  $\chi$  could then be formulated, much as it was in the optimal control

illustration, and it might be possible to use the framework more precisely in this context (especially for animals in the wild state). Some suggestions for bringing cell assembly theory within the range of the  $(\mathfrak{I}, \mathfrak{E}, \chi)$  framework are made in section 8.4. There is much to be done before we can hope for definite, general results from theory.

**Summarizing:**

- $\mathfrak{Q}$ , repertory of possible cell assemblies.
- $\Omega$ , possible association rules (Hebb's rule for synapse change, short-term memory rules, etc.).
- $\mathfrak{I}$ , possible (or hypothetical) organizations of the CNS in terms of conditions under which the rules of  $\Omega$  are to operate.
- $\mathfrak{E}$ , the range of environments in which the CNS being studied is expected to operate (relevant features, cues, etc.).
- $\chi$ , the ranking of organizations in  $\mathfrak{I}$  according to performance over  $\mathfrak{E}$ , for example, according to ability to keep average needs low under any situation in  $\mathfrak{E}$  (cf. optimal control illustration).

These illustrations are intended to demonstrate the broad applicability of the  $(\mathfrak{I}, \mathfrak{E}, \chi)$  framework. They can also serve to demonstrate something else. The obstacles described informally at the end of section 1.2 do indeed appear as central problems in each of the fields examined. This is an additional augury for making a unified approach to adaptation—common problems should have common solutions. Much of the work that follows is directed to the resolution of these general problems. In section 9.1 the problems are listed again, more formally, and the relevance of this work to their resolution is recapitulated.

This excerpt from

Adaptation in Natural and Artificial Systems.

John H. Holland.

© 1992 The MIT Press.

is provided in screen-viewable form for personal use only by members  
of MIT CogNet.

Unauthorized use or dissemination of this information is expressly  
forbidden.

If you have any questions about this material, please contact  
[cognetadmin@cognet.mit.edu](mailto:cognetadmin@cognet.mit.edu).