

On the Mechanical Simulation of Habit-Forming and Learning*

SAUL GORN

*Moore School of Electrical Engineering and University of Pennsylvania
Computer Center, Philadelphia, Pennsylvania*

This paper discusses digital techniques by which habit-forming and learning may be simulated. After classifying the types of simulation mechanisms it discusses types of habit-forming and learning to be simulated, focusing attention upon reinforcement. It uses the language of computer programming to describe the flow of control, and the language of mathematical probability to analyze the effect of various reinforcement functions on the asymptotic behavior of simulating programs. It shows further, again in programming terms, how the "delayed random selector" part of the simulating process may be "factored out" as a separate unit applicable either to habit-forming or learning, which latter are distinguished by whether the reinforcements are applied immediately or upon "comparison with a goal."

Several reinforcement models are considered, including the "linear asymptotic" model used extensively by Bush and Mosteller, two simple "absorbing boundary" models, and a "nonlinear asymptotic" model currently being investigated by Bush, Galanter, and Luce. A sketch is given of the Harris-Bellman-Shapiro analysis of the linear asymptotic model. Contrasted with this, a complete analysis is given of the simpler absorbing boundary model, with explicit proof of eventual absorption, and formulae for probability of absorption in n trials, and the expected number of steps to absorption. Finally, a special example is given of the second absorbing boundary model to show how its structure differs from the others.

* The research in this paper was, in part, made possible by a grant from the National Science Foundation to the University of Pennsylvania Computer Center. Portions of its material have been presented at several University of Michigan summer courses ("The Applications of Logic to Digital Computer Programming" in August, 1957, and "The Applications of Digital Computers to Artificial Intelligence" in June, 1958) as well as to the seminar on "Automatic Computers and Their Capabilities" jointly sponsored in 1957-58 by the Moore School of Electrical Engineering and the Society for Industrial and Applied Mathematics.

INTRODUCTION

Why should "life" scientists be interested in mechanical models for experimentation?

It is because the living subject, being "irritable" and intricate, often yields more information about the disturbance caused by the experiment than it does about the factor being investigated.

A machine, on the other hand, can be constructed to abstract only what is under investigation.

This paper will concern itself with a class of mechanical models of the type obtained by programming a general purpose digital computer to simulate many special mechanical models. Each such program is a linguistic description in "command" form of the special mechanical model desired. (In just such a way a complete abstract theory is a "linguistic mechanism.") These programs are relatively easy to form, and one could try hundreds of such mechanisms and experiments on the same general purpose machine. In fact, one could have a "library" of such programs and "compile" intricate larger mechanisms by suitably programming the order in which the sub-mechanisms are to appear.

This paper, because it discusses programmed models of the "reinforcement" type, will, perforce, use two types of language in their description. The first is a language suitable for presenting the flow of controls in the models, i.e. the "command language" of programming as presented in flow charts. It will provide the simplest means to show how control selections are made, delayed, and randomized; for, whatever else they may be, habit-forming and learning are delayed random selection processes. The second is a language suitable for presenting the asymptotic analysis of the feedback of communications in these models, that is, the "descriptive language" of mathematics as it is used to analyze the random "memory" processes being mechanized.

We can expect such mixed command and descriptive languages to appear whenever mechanisms are specified and analyzed.

Such emphasis on language, is, however, forced upon us by the fact that our models are programmed. We should here pause to remark on other mechanical approaches.

The mechanical simulation of habit-forming and learning may be analog or digital. Examples of analog simulation mechanisms appear in the path-breaking work by W. Ross Ashby (1952) and W. Grey Walter (1953). In these studies, selection by the mechanism is from a continu-

ous data range rather than from discrete information domains; the emphasis is on the study of behavior stability rather than on the variety of selection procedures.

If the mechanical simulation is digital, it may be achieved either by a special purpose machine or by programming for a general purpose machine. The special purpose machines appear in digital hardware, rather than programming. Such machines have been designed by Minsky (1956) and by Solomonoff (work as yet unpublished).

However, these two methods, special purpose digital and general purpose digital, hardware and programming, are completely equivalent, for all programs for general purpose machines are descriptions of special purpose machines in a serial command language. Conversely, any special purpose digital machine may be simulated on a general purpose machine by a program designed to give the same results. However, because the present general purpose machines are essentially serial, such programs do violence to the internal timing being simulated, since it could have a degree of concurrent operation.

The lines in the flow chart of a program indicate the flow of control, which will branch at certain critical nodes. These nodes might indicate simultaneous action or selective action; they might be "and-nodes" or "or-nodes." Since present machines are essentially serial, and-nodes do not appear in digital flow charts as they do in analog flow charts. When concurrently operating general purpose digital computers appear, it will no longer be necessary to simulate simultaneous actions by stringing them out serially. The equivalence of programming and hardware will become immediately apparent; flow charts for routines and logical designs of corresponding machines will match box for box and arrow for arrow if the degree of detail of the language describing each is the same.¹

We see, then, that no loss in generality occurs if we describe the mechanical simulation of habit-forming and learning in terms of programs for general purpose machines. The distinction is in manner of simulation rather than in what is simulated. The same is essentially true in the analog-digital dichotomy.

A more essential dichotomy in "learning" is concerned with whether the storage of relevant experience during the process is explicit or implicit. The explicit method stores suitably coded and classified lists of

¹ Feedback in the machine corresponds to looping back in the program. All the general purpose machines for which we program such simulation possess a command language permitting loop control (Gorn, 1957).

relevant experiences to be used in the delayed selection of the proper response to a stimulus. The implicit method affects the memory merely by reinforcing the more appropriate responses whenever an experience so warrants, without any more detailed storage of information. Control at the moment of selection is automatic and immediate, without elaborate digestion of past information.

Evidently the explicit method is more likely to simulate the intelligent development of *gestalts* in habit-forming and learning. But it must be used with discretion, for, with no other intermediary, the piling up of experience rapidly develops an information retrieval problem. The more experience piles up, the longer is the selection of information therefrom. The action of the control choosing the proper response to the stimulus becomes more delayed. It is as though advice were asked of an old, old man—the type who responds, in ancient mariner fashion, with a lengthy biographical sketch of largely irrelevant experiences before arriving, if ever, at the required suggestion for action.

The implicit method, on the other hand, though it yields faster responses to stimuli, simply simulates reinforcement or avoidance reactions; it is thereby less “intelligent” and more like brute habit-forming or like learning of the conditioning or reflex type.

There is no doubt, then, that when experimental models are constructed, they should include a number of judicious combinations of the implicit and explicit simulations. This paper will be restricted to a discussion of the implicit type. The reader will find interesting information on the explicit type in the studies by Newell and Simon (1957).

In this paper, the question of asymptotic behavior will emerge as the result of studying stochastic difference equations rather than of examining singular points of differential equations, as would be done with analog models.

We study, then, functional equations connected with decision-making processes, such as the Bales-Householder model which is analyzed by Harris, Bellman, and Shapiro (1953) and used extensively by Bush and Mosteller (1955).

Although, in the digital decision (that is, selection) processes discussed, the question of whether a habit will be established is an important one, the main issue will nevertheless be the selection procedures themselves. We therefore begin by discussing selection methods, the or-nodes referred to above.

SELECTION METHODS

Three ways to classify selection methods are relevant to our subject. First, selection methods can vary according to the categorization structure of the list from which the selection is made. Second, selection methods can vary by the amount of delay between the moment of choice and the moment at which such choice affects further action. Third, selection methods can vary from the completely deterministic to the rectangularly distributed random.

Consider first the classification by the structure of the domain from which the selection occurs. There are two extremes. If we have n objects or courses of action from which the choice is to be made, the most direct—but least economical—is to examine the list in any order, asking whether the one at hand is elected. At the other extreme, if we have a system of classification levels defined for the n objects, each of which approximately bisects a class at the previous level, we can make our selection in the manner of the game “twenty questions”.

The table-look-up method requires n questions but is extremely flexible; the binary tree method needs approximately $\log_2 n$ questions but is rigid. [A theorem in set theory shows that any polyadic system can be reorganized into a dyadic one; see Hansdorff (1957).] In hardware, these extremes are exemplified by contrasting the “order” type selector in the control of a binary machine to the “matrix” type selector for a machine like the Univac (the latter regains much of the time economy by concurrent action below the instruction level). In our introductory remarks, we noted that the explicit method of storing relevant information could present an excessive delay at the time of selection. The same delay could occur in the selection of responses to a stimulus. The models we will consider hereafter will look like binary-tree selections, but they will be essentially of the table-look-up type. Flexibility is needed to achieve generality.

This is all we need say about the first method of classification. Henceforth in this paper we will focus our attention on the method of delaying the effect of selection and more or less making it random.

Distinguishing “routines” from “live” or “intelligent” behavior depends not so much on their simulation of decision making or thinking but on their completely deterministic pattern.

Most routines do simulate “routine” thinking, as opposed to “executive” thinking in which creative or random choices of alternatives are involved. They are most sharply and economically represented by flow

charts, and these flow charts branch off at or-nodes either to loop back with or without modification (feedback) or to indicate selection alternatives at certain critical points. At these critical branching points, the choices are either made on the spot by "discriminations" or achieved by delayed decisions made previously. In the latter case, they are called "variable exits" or "variable remote connections." Thus, the branching in Fig. 1a could be achieved in a routine either by the method in Fig. 1b or that in Fig. 1c (note: the symbol $x \rightarrow y$ means that the contents of storage x are put into storage y , erasing what was previously in y but not affecting the contents of x).

The method indicated in 1c will be our standard method of delaying selection. In programming it is, as the name of α indicates, a standard way to exit from a portion of a program which we would like to use in many sections of the main program. Our models will, therefore, be "sub-routines" in the usual sense of the word.

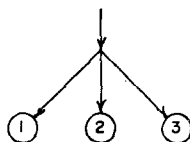


FIG. 1a

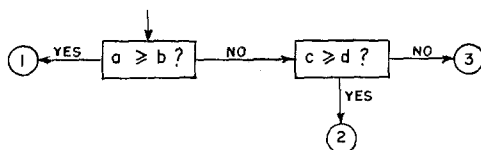


FIG. 1b. Immediate selection

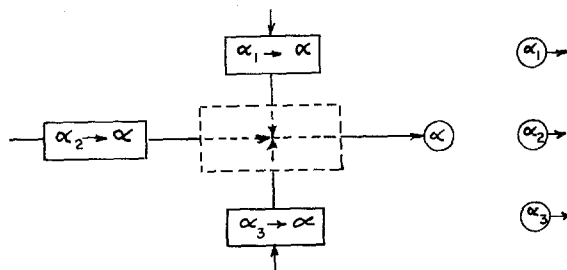


FIG. 1c. Delayed selection

Turning now to the possibility of randomization, we are interested in simulating systems in which the choice at or-nodes is made at random. A flow chart (such as that in Fig. 1a) in which or-nodes have selection probabilities attached—such as 50 % to exit 1, 20 % to exit 2, and 30 % to exit 3—would simulate not a procedure but a process in the statistical sense. This is especially true if these probabilities may vary at each feed-back to the same selection point.

We may wonder how a purely syntactical and rigidly routine instrument such as a general purpose computer may be made to simulate random phenomena. We could, of course, attach some random process to the input of the computer. Such a method is not entirely satisfactory, since we lose sight of or control over the process. We could of course record these inputs and run the problem again with the copied inputs. In that case we could have used a table of random numbers, preassigned, in the first place. There is, however, a more simple and direct way, using a simple routine, to simulate the production of a random table. The paradox that randomness may be simulated on a deterministic machine is explained by a theorem of de Leeuw, Moore, Shannon, and Shapiro (1956). The theorem states, roughly, that if a probabilistic machine works on random information which has a computable distribution, it is possible to simulate the probabilistic machine by an appropriate deterministic device.

A number of simple routines produce such “pseudo-random” numbers [see Taussky and Todd (1955); Juncosa (1953)]. Perhaps one of the simplest is Lehmer’s method, in which, beginning with an arbitrary number r_0 which fills a storage position and an appropriate number r , we find, recursively,

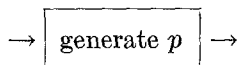
$$r_{n+1} \equiv r_n r \pmod{R^s}$$

by multiplying each r_n (in double precision) by r and retaining as r_{n+1} the least significant half (assuming an s -place machine whose arithmetic operates with radix R). A computation by elementary number theory, for example, shows that for Univac, which is an 11-place decimal machine, using

$$r = 54, 638, 671, 877$$

we get the maximum cycle of numbers r_n possible, namely 5×10^9 . Samples of r_n which are small compared to 5×10^9 pass most of the statistical tests for randomness.

We will represent any such subroutine which generates "pseudo-random" numbers by the symbol



in a flow chart.

The stage is now set to program random selection routines. As in Fig. 1, we can achieve the effect of the random or-node 2a immediately, as in Fig. 2b. We are now ready to achieve the effect of the delayed random selection 2c by combining the method of 2b with that of 1c. We will then be able to look upon α and what precedes it as a "stimulus" with the responses α_1 , α_2 , and α_3 .

THE GENERAL REINFORCEMENT SUBROUTINE

As the introduction states, we are restricting our discussion to the programmed simulation of habit forming and learning as delayed selec-

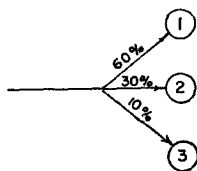


FIG. 2a

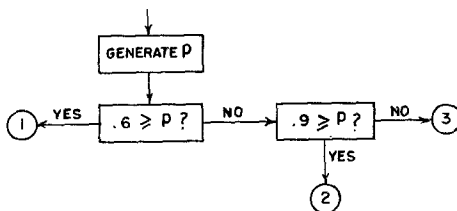


FIG. 2b



FIG. 2c

tion processes of the implicit type. If the stimulus simulation α has n possible responses $\alpha_1, \alpha_2, \dots, \alpha_n$, there will be a probability vector $(p_1, p_2, \dots, p_n) = p$ for which $p_1 + p_2 + \dots + p_n = 1$; at each selection, p_i will be the probability that α_i be chosen. The implicit method can be achieved by making this vector a variable which is different at each feedback to the selecting subroutine. We can now distinguish the learning type from the habit-forming type because the variation of p occurs in learning when there is a comparison with some goal. Thus, in learning, the transformation of the vector p must occur outside the random selection subroutine. In habit forming, such transformation does not have to occur outside the subroutine. For a general habit-forming subroutine, then, for each response, α_i , there is a set of transformations T_{ijs} of the vector p , the distribution of responses; here j is an index describing the past history of responses, and s an index—whether random, deterministic, or a combination of both—which describes the “strength of the stimulus.” The vector p is transformed at each stimulus with response i (occurring with probability p_i), past history of responses j , and strength s into the vector $T_{ijs}p$, ready for the next application of the stimulus. Presumably, the resulting component p_i will be greater than the previous p_i if there is a habit-forming tendency to response α_i ; we might call this a “positive” or “reinforced” response. However, the whole habit-forming pattern might tend to avoid response α_i because of its consequences; i.e., the i th component of $T_{ijs}p$ might be smaller than the original p_i . In such a case, we might call α_i a “negative” response (or an avoidance response).

Further classifications of habit-forming and learning processes might be made according to the nature of s or j . Apart from these, an important classification concerns whether a component p_i can ever become 1 or 0 and whether, if such is indeed possible, it could ever change from such a value, once attained. If no p_i can ever become 1 or 0, but can approach them as close as we please, we say that the model is asymptotic. If the p_i can attain these values and, having done so, cannot change, we say that the model is of the absorbing boundary type. Psychologists call this last the “perfect learning” type.

Other classifications depend upon the nature of the transforming functions T (linear, projective, etc.), the resulting process (Markovian or not), and similar factors.

In this paper we will restrict our discussion to a number of models most of which are special cases of the general reinforcement type habit-

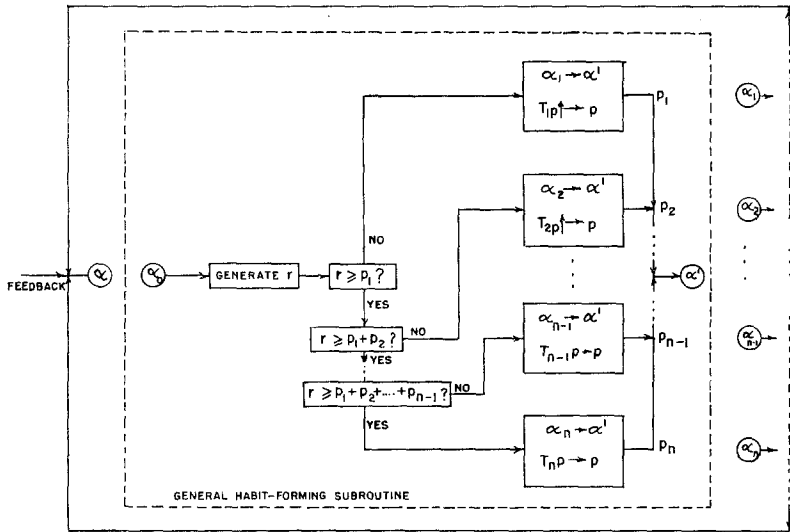


FIG. 3. T_i : reinforcement functions; $p = (p_1, p_2, \dots, p_n)$; $p_i \geq 0$, $i = 1, \dots, n$; $\sum_{i=1}^n p_i = 1$.

forming subroutines indicated by Fig. 3. In this figure T_i are the various reinforcement functions, $p = (p_1, p_2, \dots, p_n)$, $p_i \geq 0$ for

$$i = 1, 2, \dots, n,$$

and

$$\sum_{i=1}^n p_i = 1$$

In such a subroutine we could defer the reinforcement, $T_i p \rightarrow p$, until after the response α_i has been taken. By doing so we can use the resulting subroutine either for habit forming or for learning. It has, in fact, become a pure random selector. On any machine with loop control (Gorn, 1957) this random selector may be achieved more compactly as shown in the "schematic" flow chart of Fig. 4, in which instructions such as $i + 1 \rightarrow i$ must modify a number of instructions. By such a method we have "factored" out the pure random selector and separated it from the reinforcement functions.

Because of the tremendous variety of choices for the operators T_i ,

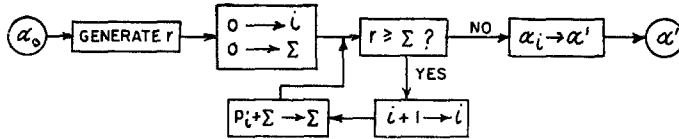


FIG. 4. General random selector for a vector for responses

not only need the model not be reinforcing, it need not even be stochastic. For example, suppose

$$T_1 p = T_2 p = \cdots = T_n p = (p_2, p_3, \cdots, p_n, p_1)$$

that is, all the operators T_i merely permute the components cyclically, and suppose further that p begins initially as $(1, 0, \cdots, 0)$. In that case, the model will merely cycle the responses $\alpha_1, \alpha_2, \cdots, \alpha_n$ in that order, leaving no randomness at all. Suppose, again, that $n = 2$, and that $(T_i p)_j$ represents the j th component of $T_i p$; we might have $(T_1 p)_1 < p_1$ and $(T_2 p)_2 < p_2$; i.e., each response weakens the probability of its own occurrence at the next stimulus.

All that can be said *a priori* about the operators T_i is that they are functions mapping the simplex

$$p_1 + p_2 + \cdots + p_n = 1, \quad p_i \geq 0, \quad i = 1, \cdots, n$$

into itself in n -space. They can be graphed by presenting $n(n-1)^2$ two-dimensional graphs of $(T_i p)_j$ versus p_k . For $n = 2$ and $T_1 = T_2 = T$, determined as in Fig. 5, if p_1 begins at the first stimulus at p_{10} , where $(T p)_1 = p_{20} = 1 - p_{10}$ and $(T p)_2 = p_{10}$, each response causes the probabilities of the two responses to be interchanged. On the other hand, if $p_1 = p_{11}$ initially, the model will develop the asymptotic behavior whereby α_1 will tend to appear with probability $p_{1\infty}$.

Evidently we can expect a variety of models, each of which must be judged by its usefulness in approximating observed behaviors.

One attempt to determine uniquely properties of the T_i on *a priori* grounds is interesting enough to receive special mention. This is the so-called "combining of classes" condition (see Bush and Mosteller, 1955; Bush, Mosteller, and Thompson, 1954).

Suppose the responses α_i represent, really, classes of responses. Suppose, further, that we have decided that the mutually exclusive and exhaustive response possibilities $\alpha_1, \alpha_2, \cdots, \alpha_n$ included some irrelevant subdivisions. We would, therefore, like to combine certain subsets of

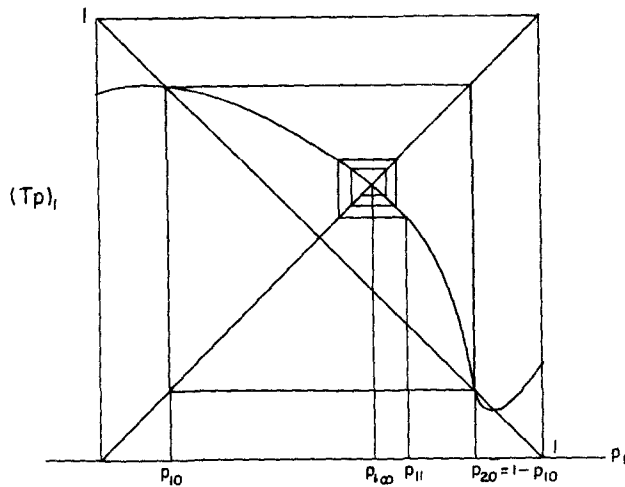


FIG. 5

these response classes into single response classes. Consider whether there are any operators T which, when used for any T_i , are unaffected by such combining of classes (unaffected in the sense that those left ungrouped will not have their T -transformed p components changed, independent of the choice of subsets).

If $n = 2$, the answer is trivial, because the only combination of classes leaves only a single response class, which must be assumed with probability one. Any functions T_i will satisfy this condition because

$$(Tp)_1 + (Tp)_2 = 1$$

as well as $p_1 + p_2 = 1$.

If, however, $n > 2$, this question changes from a very trivial one to a very powerful one, so powerful that it is surprising that there is any answer at all. The condition has been formulated in this way in the cited reference:

Let σ be a subset of the indices $1, 2, \dots, n$ of the components of p ; further, let n_σ be any index of the set σ . We define the projection operator C_{σ, n_σ} in the (p_1, p_2, \dots, p_n) simplex as the transformation yielding the sum of the σ components for the new n_σ component, giving zero for all the other new σ components, and leaving all the non- σ components unchanged. If p is represented as a column matrix, $C_{\sigma, n_\sigma}p$ may be obtained by multiplying p on the left by the matrix obtained from the

identity matrix upon (1) replacing every diagonal element in a σ -column unequal to n_σ by zero and (2) replacing every zero in the n_σ row of a σ -column by one. The combining-of-classes (*CC*) condition, then, has the formal statement: The operator T fulfills the *CC* condition if

$$C_{\sigma, n_\sigma} T p = C_{\sigma, n_\sigma} T C_{\sigma, n_\sigma} p$$

for every σ, n_σ , and p , where $n_\sigma \in \sigma$ and $\sigma \subset \{1, 2, \dots, n\}$.

A generalization by Bush, Mosteller, and Thompson (1954) of a theorem of Savage states that any operator mapping the p -simplex into itself and satisfying the *CC* condition must be linear and of the form $Tp = \alpha p + (1 - \alpha)\lambda$, where λ is a fixed vector and α is a constant.

Note that λ is an eigenvector of T with eigenvalue one. If

$$T_1 = T_2 = \dots = T_n = T, \quad \lambda = (p_{1\infty}, p_{2\infty}, \dots, p_{n\infty})$$

in the notation of Fig. 5. Repeated T -reinforcement produces an asymptotic distribution of responses.

The combining-of-classes argument can hardly be considered a cogent *a priori* reason for restriction of the models to linear asymptotic reinforcement functions. It demands first that the model be impervious to an error in judgment in the choice of possibly irrelevant response classes. It also demands a like relationship between the scientist's method of analysis (possibly mistaken) and the nature of the system analyzed for $2^n - n - 2$ possible basic reorientations (or $n(n - 1)/2$ primitive reorientations).

On the other hand the linear model it yields should be one of the prime reinforcements of the asymptotic type to be studied.

For most of the remainder of this paper, however, the previous argument is of academic interest only, since the models we propose to discuss in greater detail all have $n = 2$.

ASYMPTOTIC AND ABSORBING BOUNDARY MODELS

When $n = 2$, the general reinforcement model is completely determined by two functions, namely, the two functions of p_1 yielding the first components of $T_1 p$ and $T_2 p$.

If, for example, in the linear asymptotic model resulting from the *CC* condition, we want T_1 to reinforce asymptotically to α_1 with probability 1 and T_2 to reinforce α_2 similarly, we must have $\lambda = (1, 0)$ in

the first case and $\lambda = (0, 1)$ in the second. This means that

$$\text{and} \quad \begin{cases} T_1 p = \begin{pmatrix} 1 & \alpha \\ 0 & 1 - \alpha \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \\ T_2 p = \begin{pmatrix} \sigma & 0 \\ 1 - \sigma & 1 \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \end{cases} \quad (1)$$

(Here we have $1 - \alpha$ instead of α in T_1 and σ instead of α in T_2 .)

Thus $(T_1 p)_1 = p_1 + \alpha p_2 = \alpha + (1 - \alpha)p_1$, and $(T_2 p)_1 = \sigma p_1$. The model is therefore determined by the graphs in Fig. 6.

The asymptotic nature of the model is indicated by the crossing of the 45° line within or on the boundary of the square. If the model is to be symmetric in the reinforcement for the responses α_1 and α_2 , the parameters must reduce to one; i.e., $\sigma = 1 - \alpha$. This symmetric model has the following simple interpretation: at each response α_i , the reinforcement consists of adding to p_i an additional α times $(1 - p_i)$, that is, a fixed percentage of the probability of not getting response α_i .

This symmetric linear asymptotic model, proposed first by W. K. Estes (1950), immediately suggests the construction of several simple symmetric models of the absorbing boundary type, models H_{1t} and $H_{2\sigma}$. Note that H_{1t} is less likely than $H_{2\sigma}$ to have physical significance. Our purpose, however, will be to show that, on the one hand, the asymptotic and the absorbing boundary types call for completely different methods of analysis and that, on the other hand, such simple models as H_{1t} and $H_{2\sigma}$ can present completely different structures.

At each response α_i , model H_{1t} will have a fixed addition t to the probability p_i of achieving that response at the next stimulus. As soon as

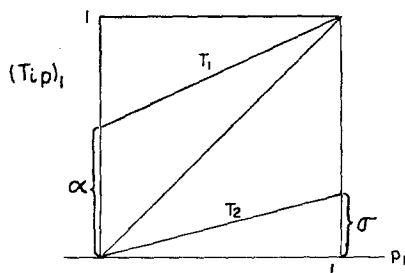
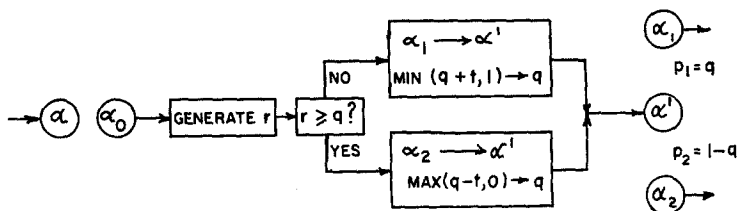
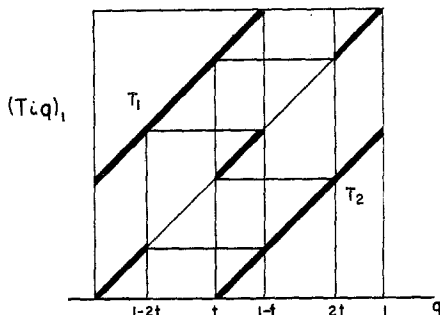


FIG. 6. Linear asymptotic reinforcement, $n = 2$

FIG. 7. Subroutine for model H_{1t} FIG. 8. Symmetric, additive, absorbing boundary reinforcement (model H_{1t}), $n = 2$.

$p_i + t$ becomes greater than or equal to one, we will take $p_i = 1$ thenceforth. In other words, if we let q_n be the value of p_1 after the n th stimulus,

$$q_n = \begin{cases} \text{Min } (q_{n-1} + t, 1) & \text{if } r_n < q_{n-1} \\ \text{Max } (q_{n-1} - t, 0) & \text{if } r_n \geq q_{n-1} \end{cases} \quad (2)$$

The flow chart for the subroutine corresponding to model H_{1t} is shown in Fig. 7, and its $T_1 - T_2$ diagram is shown in Fig. 8.

Any routine using model H_{1t} as a subroutine would set its free variables, t and the initial value of $q(q_0)$. Thus, somewhere in the main routine we will find the instructions setting t , and substituting $q_0 \rightarrow q$ and $\alpha_0 \rightarrow \alpha$. That the model is symmetric follows immediately from the equations:

$$\begin{aligned} 1 - \min [(1 - q) + t, 1] \\ &= 1 + \max [-1 + q - t, -1] = \max [q - t, 0] \\ 1 - \max [(1 - q) - t, 0] \\ &= 1 + \min [-1 + q + t, 0] = \min [q + t, 1] \end{aligned}$$

Furthermore, since the subroutine is of the "perfect learning" type, it may be set to seal itself off by placing just before the exit α^1 the flow chart in Fig. 9.

An example of how model H_{1t} could be extended to allow for more than two responses is given in Fig. 10.

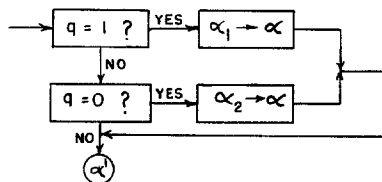


Fig. 9. Self-sealing exit for perfect learning models

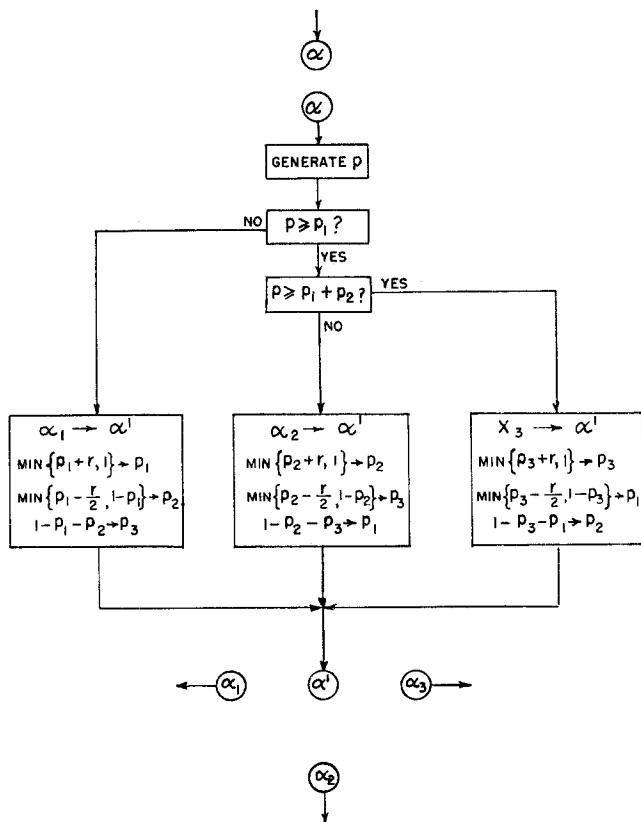
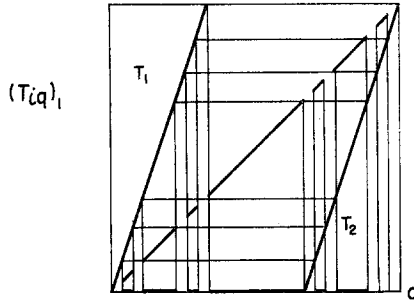


Fig. 10. Extension of model H_{1t} to more than two responses

FIG. 12. Reinforcement (model $H_{2\sigma}$), $n = 2$, $\sigma = 200$

(Fig. 9), and it is possible to generalize the model to more than two responses, as in Fig. 10.

For each of these models, the following fundamental questions arise:

(a) Is there an asymptotic response pattern established, or is there a probability greater than zero that a floundering between responses will continue indefinitely?

(b) In the case where it can be shown that an asymptotic response pattern is established (with probability 1), can we define a measure of the "degree" with which it is established, and, for any such degree, can we find the number of stimuli expected to produce such a degree of establishment?

For the most trivial of the models here shown, model H_{1t} , these questions will be answered completely.

ANALYSIS OF ASYMPTOTIC MODELS

For asymptotic models we expect that $(T_1 p)_1 = 1$ for $p = (1, 0)$, and that $(T_2 p)_1 = 0$ for $p = (0, 1)$ as in Fig. 6. If we let $x = p_1$, then let us define the functions as in Fig. 6, expressed as functions of x alone, as follows:

$$\begin{aligned} t_1(x) &= (T_1 p)_1, & t_1(1) &= 1 \\ t_2(x) &= (T_2 p)_1, & t_2(0) &= 0 \end{aligned} \quad (4)$$

Thus the linear asymptotic model of Eqs. (1) has:

$$\begin{aligned} t_1(x) &= \alpha + (1 - \alpha)x \\ t_2(x) &= \sigma x \end{aligned} \quad (5)$$

and we note that, for $0 \leq x \leq 1$, we have $t_1(x) \geq t_2(x)$.

In this section we will sketch the analysis given by Harris, Bellman, and Shapiro (1953) in order to do the following:

1. Define the expressions: a sequence of trials "concludes α_1 " or "concludes α_2 ", and define the probabilities $\pi_1(x)$ and $\pi_2(x)$ of doing so.

2. Prove the following:

THEOREM 1. Both $\pi_1(x)$ and $\pi_2(x)$ satisfy the functional equation:

$$f(x) = xf(t_1(x)) + (1 - x)f(t_2(x)) \quad (6)$$

with boundary conditions:

$$\begin{aligned} \pi_1(0) &= 0, & \pi_1(1) &= 1 \\ \pi_2(0) &= 1, & \pi_2(1) &= 0 \end{aligned} \quad (7)$$

THEOREM 2. If $f(x)$ satisfied the functional equation (6), and $g(x) = f(1 - x)$, then $g(x)$ satisfies

$$g(x) = xg(1 - t_2(1 - x)) + (1 - x)g(1 - t_1(1 - x)) \quad (8)$$

THEOREM 3. If, for the linear asymptotic model of (5) we define:

$$f_{\alpha, \sigma}(x) = \pi_1(x), \quad (9)$$

then

$$\pi_2(x) = f_{1-\sigma, 1-\alpha}(1 - x) \quad (10)$$

THEOREM 4. For each set of boundary conditions:

$$f(0) = f_0 \quad \text{and} \quad f(1) = f_1,$$

where

$$0 \leq f_i \leq 1, \quad (i = 1, 2)$$

there is a unique solution of the functional equation (6), where $t_1(x)$ and $t_2(x)$ are defined by Eqs. (5), and this solution is absolutely monotonic [that is, $f^{(k)}(x) \geq 0$ for all integers k] and analytic.

THEOREM 5. $\pi_1(x) + \pi_2(x) \equiv 1$

3. It therefore follows that the analysis of asymptotic models can be expected to have the following features:

a. It will yield an asymptotic probability distribution which satisfies a functional equation.

b. The uniqueness of the solution shows directly that the probability of "floundering" is zero.

c. The distribution is analytic.

Thus question *a* at the end of the last section is completely settled, and it is fairly obvious that question *b* can be handled, though a constructive formulation of it will be left an open question. The next section will show how the analysis of absorbing boundary models has a completely different flavor.

Definition. Let B be the sequence of events

$$B = (B_1, B_2, \dots, B_n, \dots)$$

where each event B_n is the appearance of either the response α_1 , or the response α_2 . A sequence B is said to "conclude α_1 " if there is a number n such that $B_m = \alpha_1$ if $m \geq n$; a similar definition applies to the expression " B concludes α_2 ".

Definition. Let $\pi_i(x)$ be the probability of concluding α_i when, at the initial trial B_1 , we have $p_1 = x$.

Proof of Theorem 1. B concludes α_1 if: either $B_1 = \alpha_1$ and (B_2, \dots, B_n, \dots) concludes α_1 with $p_1 = t_1(x)$, or $B_1 = \alpha_2$ and (B_2, \dots, B_n, \dots) concludes α_1 with $p_1 = t_2(x)$. Thus Eq. (6) holds for $\pi_1(x)$. If we replace the expression "concludes α_1 " by "concludes α_2 " in the first sentence of this proof, the resulting sentence is again true, and Eq. (6) holds for $\pi_2(x)$.

Proof of Theorem 2. This is obvious when one first substitutes $1 - x$ for x in (6) and then substitutes $g(x)$ for $f(1 - x)$ and $g(1 - y)$ for $f(y)$ for the apparent expressions y . *Note:* Since $n = 2$, we have

$$\begin{aligned} p_1 + p_2 &= 1 & \text{or} & & p_2 &= 1 - x \\ (T_1p)_1 + (T_1p)_2 &= 1 & \text{or} & & (T_1p)_2 &= 1 - t_1(x) \\ (T_2p)_1 + (T_2p)_2 &= 1 & \text{or} & & (T_2p)_2 &= 1 - t_2(x) \end{aligned}$$

Thus an interchange of α_1 and α_2 , with the corresponding redefinition of x , would have led to the deduction of (8) and (7) for the newly defined $\pi_1(x)$ and $\pi_2(x)$; they would be $\pi_2(1 - x)$ and $\pi_1(1 - x)$ respectively.

Proof of Theorem 3. For $1 - t_2(1 - x) = 1 - \sigma(1 - x) = (1 - \sigma) + \sigma x$, and $1 - t_1(1 - x) = 1 - \alpha + (1 - \alpha)(1 - x) = (1 - \alpha)x$. Thus Eq. (8) is obtained from (6) by replacing α by $1 - \sigma$ and σ by

$1 - \alpha$. Equation (10) is thus an immediate corollary of theorem 2. *q.e.d.*

Proof of Theorem 4 (sketch). Let us define the functional operator:

$$\Lambda h = xh(\alpha + (1 - \alpha)x) + (1 - x)h(\sigma x), \quad (11)$$

applicable to any function $h(x)$ for which $0 \leq h(x) \leq 1$ when $0 \leq x \leq 1$. Λ is, then, a functional operator with the following properties:

a. It preserves the boundary values; that is, if $h_1 = \Lambda h$, then $h_1(0) = h(0)$ and $h_1(1) = h(1)$.

b. It is linear; $\Lambda(h_1 + h_2) = \Lambda h_1 + \Lambda h_2$ and $\Lambda(ch) = c\Lambda h$ for any constant c .

c. If $h_1 = \Lambda h$, then we also have $0 \leq h_1(x) \leq 1$ when $0 \leq x \leq 1$; that is, Λ maps its domain into itself. Similarly for $\Lambda^n h$ for all natural numbers n .

d. If h is continuous in the closed interval $[0, 1]$, and $0 \leq \alpha \leq 1$, $0 \leq \sigma \leq 1$, then Λh is also continuous in $[0, 1]$.

e. If h is constant, then $\Lambda h = h$.

f. $\Lambda x = x[\alpha + (1 - \alpha)x] + (1 - x)\sigma x = x + (\alpha + \sigma - 1)x(1 - x)$.

g. If $h(0) = h(1) = 0$, h is continuous on the closed interval $[0, 1]$, $h = \Lambda h$, and neither α nor σ is 0 or 1, then $h(x) \equiv 0$. For if x_0 is a point of $(0, 1)$ at which $|h(x_0)|$ is maximum, then $h(x_0) = h(\sigma x_0)$ because $h = \Lambda h$ requires that the maximum or minimum value, $h(x_0)$, lie between $h(\sigma x_0)$ and $h(\alpha + (1 - \alpha)x_0)$ and must, therefore, be equal to both. A repetition of this argument shows that $|h|$ must assume its maximum at $\sigma x_0, \sigma^2 x_0, \dots, \sigma^n x_0, \dots \rightarrow 0$. By continuity this maximum must be $h(0) = 0$.

h. If $h_1(0) = h_2(0)$, $h_1(1) = h_2(1)$, h_1 and h_2 are continuous on the closed interval $[0, 1]$, $h_1 = \Lambda h_1$, $h_2 = \Lambda h_2$ and neither α nor σ is 0 or 1, then $h_1 \equiv h_2$. For $h = h_1 - h_2$ fulfills all the conditions of g. Thus, under the conditions on α and σ , there can be no more than one continuous solution of $h = \Lambda h$ for any pair of boundary conditions.

i. $\pi_1(x) = \lim \Lambda^n x$. For if $h_0 \equiv x$, then by f,

$$h_1(x) = \Lambda h_0 = x + (\alpha + \sigma - 1)x(1 - x)$$

$$h_2(x) = \Lambda h_1 = h_1 + (\alpha + \sigma - 1)\Lambda[x(1 - x)]$$

$$h_{n+1}(x) = \Lambda h_n = h_n + (\alpha + \sigma - 1)\Lambda^n[x(1 - x)]$$

But, by c, $\Lambda^n[x(1 - x)]$ is positive. Hence the $h_n(x)$ form a monotonic

sequence bounded by 0 and 1, nondecreasing if $\alpha + \sigma \geq 1$, and non-increasing if $\alpha + \sigma \leq 1$. In either case, convergence follows, and, if $\alpha + \sigma - 1 \neq 0$, $\lim \Lambda^n[x(1-x)] \equiv 0$. Taking the limit of $h_{n+1} = \Lambda h_n$ shows that the limit satisfies (5). From the expression for h'_{n+1} obtained from (5) one shows that h'_n is uniformly bounded and positive, whence h , the limit, is continuous and monotonic; it must therefore be $\pi_1(x)$ by step h . A similar treatment of h''_{n+1} shows that $\pi_1(x)$ is convex if $\alpha + \sigma \geq 1$ and concave if $\alpha + \sigma \leq 1$. An induction on the successive derivatives of h_{n+1} then yields analyticity and absolute monotonicity of π_1 . Theorem 4 now follows by noting that $f_0 + (f_1 - f_0)\pi_1(x)$ fulfills all the conditions, and must be the unique solution by h . We note that the cited reference, by summing $h_{n+1} = h_n + (\alpha + \sigma - 1)\Lambda^n[x(1-x)]$, obtains such identities as:

$$\sum_{n=0}^{\infty} \Lambda^n[x(1-x)] = \frac{f(x) - x}{\alpha + \sigma - 1}$$

if $\alpha + \sigma \neq 1$, $\alpha \neq 0$, $\sigma \neq 1$; and if $\alpha + \sigma = 1$, then $\Lambda^n[x(1-x)] = (1 - \alpha^2)^n x(1-x)$ and

$$\sum_{n=0}^{\infty} \Lambda^n[x(1-x)] = \frac{1}{\alpha^2} x(1-x) \text{ if } \alpha \neq 0.$$

Theorem 4 is then generalized to cover the equation

$$f(x) = p(x)f(G(x)) + (1 - p(x))f(H(x))$$

where

$$G(x) = \alpha(x) + (1 - \alpha(x))x, H(x) = x\sigma(x)$$

and $\alpha(x)$, $\sigma(x)$, $p(x)$ are continuous and fulfill:

$$0 < k \leq \alpha(x) \leq 1, \quad 0 \leq \sigma(x) \leq 1 - k, \quad 0 \leq p(x) \leq 1$$

$$p(0) = 0, \quad p(1) = 1$$

Proof of Theorem 5. This is an immediate corollary of Theorem 4; since $\pi_2(0) = 1$, $\pi_2(1) = 0$, hence $\pi_2(x) = 1 - \pi_1(x)$.

ANALYSIS OF MODEL H_{1t}

Model H_{1t} does not possess an analytic probability distribution such as $\pi_1(x)$ in the asymptotic case. Instead it has discrete probabilities of

absorption at the n th trials by α_1 or α_2 . Even their generating functions are nonanalytic. On the other hand we will obtain explicit formulae answering question b of the section before last.

Definition. Let $N = N(t)$ be the integer determined by $Nt \leq 1 < (N + 1)t$. Then

$$\begin{aligned} 0 \leq 1 - Nt < t \leq 1 - (N - 1)t < 2t \leq \dots \\ < (N - 1)t \leq 1 - t < Nt \leq 1. \end{aligned} \quad (12)$$

For example, in Fig. 8, $N = 2$.

Definition. Let $P_1(x, n)$ be the probability of absorption by α_1 , in exactly n trials when the probability of α_1 at the first trial is x ; that is, $P_1(x, n) = \text{Pr}\{p_n = 1/p_{n-1} < 1, p_0 = x\}$. Similarly, for $P_2(x, n)$ and α_2 :

$$P_2(x, n) = \text{Pr}\{p_n = 0/p_{n-1} > 0, p_0 = x\}$$

Definition. Let $G_1(x, u)$ and $G_2(x, u)$ be the generating functions for absorption at α_1 and α_2 when $p_0 = x$; that is,

$$G_1(x, u) = \sum_{n=0}^{\infty} P_1(x, n)u^n, \quad G_2(x, u) = \sum_{n=0}^{\infty} P_2(x, n)u^n$$

Definition. Let $E(x)$ be the expected number of steps to absorption (either at α_1 , or at α_2) when $p_0 = x$; that is,

$$E(x) = \sum_{n=0}^{\infty} n\{P_1(x, n) + P_2(x, n)\}$$

Now, to obtain explicit formulae for the G_i and E , let us define the following determinants:

$$D_0(x, t, u) \equiv 1,$$

$$D_1(x, t, u) \equiv 1,$$

and, for $k = 2, 3, \dots, N + 1$, the determinant D_k given in Eq. (13). And let the determinants $F_{k,j}(x, t, u)$ be obtained from D_k by replacing the j th column by a column of ones. It is easy to see that the determinants D_k satisfy the recursion formula:

$$\begin{aligned} D_{j+1}(x, t, u) &= D_j(x, t, u) \\ &\quad - (x + (j - 1)t)(1 - x - jt)u^2 D_{j-1}(x, t, u) \end{aligned} \quad (14)$$

$$\begin{array}{c}
D_k(x, t, u_i) \\
\hline
\begin{array}{cccccccc}
1 & & & & & & & \\
-(1-x-t)u & & -xu & 0 & \cdots & 0 & & 0 \\
0 & & 1 & -(x+t)u & \cdots & 0 & & 0 \\
-(1-x-2t)u & & 1 & & \cdots & 0 & & 0 \\
\vdots & & \vdots & \vdots & & \vdots & & \vdots \\
0 & & 0 & 0 & \cdots & 1 & & -(x+(k-3)t)u \\
0 & & 0 & & \cdots & -(1-x-(k-2)t)u & & 1 \\
0 & & 0 & 0 & \cdots & 0 & & -(x+(k-2)t)u \\
0 & & 0 & 0 & \cdots & 0 & & -(1-x-(k-1)t)u \\
& & & & & & & 1
\end{array} \\
\hline
\end{array}
\tag{13}$$

By a slight modification of the analysis of the problem of "Gambler's Ruin" [see Feller (1950)] we will prove the following theorems.

THEOREM 6. $G_1(x, u)$ and $G_2(x, u)$ are the following piece-wise rational functions in $0 \leq x \leq 1$:

$$G_1(x, u) = \begin{cases} \frac{(x + jt) \cdots xu^{j+1} D_{N-j}(x - (N - j)t, t, u)}{D_{N+1}(x - (N - j)t, t, u)} \\ \text{for } (N - j)t \leq x \leq 1 - jt; \quad j = 0, 1, \dots, N \\ \frac{(x + jt) \cdots xu^{j+1} D_{N-j-1}(x - (N - j - 1)t, t, u)}{D_N(x - (N - j - 1)t, t, u)} \\ \text{for } 1 - (1 + j)t \leq x \leq (N - j)t; \quad j = 0, 1, \dots, N - 1 \end{cases} \quad (15)$$

$$G_2(x, u) = \begin{cases} \frac{[1 - x + (N - j)t] \cdots (1 - x)u^{N-j+1} D_j(x + t, t, u)}{D_{N+1}[x - (N - j)t, t, u]} \\ \text{for } (N - j)t \leq x \leq 1 - jt; \quad j = 0, 1, \dots, N \\ \frac{[1 - x + (N - j - 1)t] \cdots (1 - x)u^{N-j} D_j(x + t, t, u)}{D_N(x - (N - j - 1)t, t, u)} \\ \text{for } 1 - (1 + j)t \leq x \leq (N - j)t; \quad j = 0, 1, \dots, N - 1 \end{cases} \quad (16)$$

THEOREM 7. $G_1(x, 1) + G_2(x, 1) \equiv 1$, so that the probability of absorption is one (there is no indefinite floundering).

THEOREM 8. The expected number of steps to absorption is a piece-wise rational function of x and t :

$$E(x) = \begin{cases} \frac{F_{N+1,j}[x - (j - 1)t, t, 1]}{D_{N+1}[x - (j - 1)t, t, 1]} \\ \text{for } (j - 1)t \leq x \leq 1 - (N - j + 1)t; \quad j = 1, \dots, N + 1 \\ \frac{F_{N,j}(x - (j - 1)t, t, 1)}{D_N[x - (j - 1)t, t, 1]} \\ \text{for } 1 - (N - j + 1)t \leq x \leq jt; \quad j = 1, \dots, N \end{cases} \quad (17)$$

If, for example, $N = 2$, as in Fig. 8, then from (14)

$$D_0 \equiv 1$$

$$D_1 \equiv 1$$

$$D_2(x, t, u) = 1 - x(1 - x - t)u^2$$

$$D_3(x, t, u) = 1 - [x(1 - x - t) + (x + t)(1 - x - 2t)]u^2$$

Therefore, for the successive intervals, we get from (15), (16), and (17):

for $0 \leq x \leq 1 - 2t$,

$$\begin{aligned} G_1(x, u) &= \frac{(x + 2t)(x + t)xu^3}{D_3(x, t, u)} \\ G_2(x, u) &= \frac{(1 - x)u[1 - (x + t)(1 - x - 2t)u^2]}{D_3(x, t, u)} \\ E(x) &= \frac{F_{3,1}(x, t, 1)}{D_3(x, t, 1)} = \frac{2(x + t)^2 + (1 - t)}{D_3(x, t, u)} \end{aligned}$$

for $1 - 2t \leq x \leq t$,

$$\begin{aligned} G_1(x, u) &= \frac{(x + t)xu^2}{1 - x(1 - x - t)u^2} \\ G_2(x, u) &= \frac{(1 - x)u}{1 - x(1 - x - t)u^2} \\ E(x) &= \frac{F_{2,1}(x, t, 1)}{D_2(x, t, 1)} = \frac{1 + x}{1 - x(1 - x - t)}, \end{aligned}$$

for $t \leq x \leq 1 - t$,

$$\begin{aligned} G_1(x, u) &= \frac{(x + t)xu^2}{1 - [(x - t)(1 - x) + x(1 - x - t)]u^2} \\ G_2(x, u) &= \frac{(1 - x + t)(1 - x)u^2}{1 - [(x - t)(1 - x) + x(1 - x - t)]u^2} \\ E(x) &= \frac{F_{3,2}(x - t, t, 1)}{D_3(x - t, t, 1)} = \frac{2}{1 - [(x - t)(1 - x) + x(1 - x - t)]} \end{aligned}$$

and similarly for the last two subintervals. Thus, for $x = \frac{1}{2}$ we have

$$E(\tfrac{1}{2}) = \frac{4}{2t + 1}$$

For example, for $t = \frac{5}{12}$, $E(\frac{1}{2}) = \frac{24}{11}$.

Proof of Theorem 6. If $t < x < 1 - t$,

$$P_1(x, n + 1) = xP_1(x + t, n) + (1 - x)P_1(x - t, n)$$

and

$$P_2(x, n + 1) = xP_2(x + t, n) + (1 - x)P_2(x - t, n)$$

The boundary conditions are

$$P_1(x, 0) = \begin{cases} 0 & \text{for } x < 1 \\ 1 & \text{for } x \geq 1 \end{cases} \quad \text{and } P_2(x, 0) = \begin{cases} 1 & \text{for } x \leq 0 \\ 0 & \text{for } x > 0 \end{cases}$$

$$P_1(x, n) = 0 \text{ for } x \leq 0, \quad \text{or for } x \geq 1 \text{ and } n > 0$$

$$P_2(x, n) = 0 \text{ for } x \geq 1, \quad \text{or for } x \leq 0 \text{ and } n > 0$$

Thus, for $1 - t \leq x \leq 1$ we have:

$$P_1(x, n+1) = (1-x)P_1(x-t, n) \quad \text{if } n > 0, P_1(x, 1) = x$$

and

$$P_2(x, n+1) = (1-x)P_2(x-t, n)$$

while, for $0 \leq x \leq t$ we have:

$$P_2(x, n+1) = xP_2(x+t, n) \quad \text{if } n > 0, P_2(x, 1) = 1-x$$

and

$$P_1(x, n+1) = xP_1(x+t, n)$$

The boundary conditions on the generating function are, therefore:

$$G_1(x, u) \equiv \begin{cases} 1 & \text{if } x \geq 1 \\ 0 & \text{if } x \leq 0 \end{cases} \quad G_2(x, u) \equiv \begin{cases} 0 & \text{if } x \geq 1 \\ 1 & \text{if } x \leq 0 \end{cases}$$

while these generating functions fulfill the following conditions:

$$\begin{aligned} G_1(x, u) &= \begin{cases} xuG_1(x+t, u) & \text{for } 0 \leq x \leq t \\ xuG_1(x+t, u) + (1-x)uG_1(x-t, u) & \text{for } t < x < 1-t \\ xu & + (1-x)uG_1(x-t, u) \\ & \text{for } 1-t \leq x \leq 1 \end{cases} \\ G_2(x, u) &= \begin{cases} xuG_2(x+t, u) + (1-x)u & \text{for } 0 \leq x \leq t \\ xuG_2(x+t, u) + (1-x)uG_2(x-t, u) & \text{for } t < x < 1-t \\ (1-x)uG_2(x-t, u) & \text{for } 1-t \leq x \leq 1 \end{cases} \end{aligned} \quad (18)$$

Now if

$$0 \leq x \leq 1 - Nt$$

then

$$\begin{aligned} t &\leq x + t \leq 1 - (N - 1)t \\ &\vdots \\ (N - 1)t &\leq x + (N - 1)t \leq 1 - t \end{aligned}$$

and

$$Nt \leq x + Nt \leq 1$$

If, therefore, in the functional equations (18), we successively substitute $x + t, \dots, x + (N - 1)t$ for x in the middle equations, and $x + Nt$ for x in the third of each set, we will obtain two systems of $N + 1$ equations in the unknowns $G_i(x, u), G_i(x + t, u), \dots, G_i(x + (N - 1)t, u)$, and $G_i(x + Nt, u)$; in both cases the matrix of coefficients has the determinant $D_{N+1}(x, t, u)$, and the system is valid whenever $0 \leq x \leq 1 - Nt$. The constant terms will all be 0 except for the last in the G_1 system, namely $(x + Nt)u$, and the first in the G_2 system, namely $(1 - x)u$. Similarly, whenever $1 - Nt \leq x \leq t$, corresponding substitutions yield two systems of N equations each in the quantities $G_i(x, u), \dots, G_i(x + (N - 1)t, u)$. Applying Cramér's rule to the two systems for which

$$0 \leq x \leq 1 - Nt \text{ yields}$$

$$\begin{aligned} &G_1(x + (N - j)t, u) \\ &= \frac{(x + Nt) \cdots (x + (N - j)t) u^{j+1} D_{N-j}(x, t, u)}{D_{N+1}(x, t, u)} \\ &G_2(x + (N - j)t, u) \\ &= \frac{(1 - x)(1 - x - t) \cdots (1 - x - (N - j)t) u^{N-j+1} D_j(x + (N - j + 1)t, t, u)}{D_{N+1}(x, t, u)} \end{aligned}$$

while the two systems for which

$$1 - Nt \leq x \leq t$$

yield the same formulae in which $N - 1$ is substituted throughout for N . We have now only to substitute $x - (N - 1 - j)t$ for x for the

first set of intervals and $x - (N - 1 - j)t$ for x for the second set to obtain Theorem 6.

Proof of Theorem 7. Let

$$G(x, u) = G_1(x, u) + G_2(x, u). \quad (19)$$

Now add the two systems (18) equation by equation to get:

$$G(x, u) = \begin{cases} xuG(x + t, u) + (1 - x)u & \text{for } 0 \leq x \leq t \\ xuG(x + t, u) + (1 - x)uG(x - t, u) & \text{for } t < x < 1 - t \\ xu & + (1 - x)uG(x - t, u) \\ & \text{for } 1 - t \leq x \leq 1 \end{cases} \quad (20)$$

with boundary conditions

$$G(x, u) = \begin{cases} 1 & \text{if } x \geq 1 \\ 1 & \text{if } x \leq 0 \end{cases} \quad (21)$$

The same method as that used on Eqs. (18) therefore yields a unique solution for $G(x, u)$ in each subinterval via Cramér's rule. This unicity, as in the asymptotic case, quickly yields what we are after, for it is obvious when we substitute $u = 1$ in (20) and (21) that $G(x, 1) \equiv 1$ is the solution. Since

$$G(x, 1) = \sum_{n=0}^{\infty} \{P_1(x, n) + P_2(x, n)\}$$

this means that the probability of absorption, whether by α_1 or by α_2 , is one.

Proof of Theorem 8. Apply to (20) the process described in the proof of Theorem 6 to (18). This will yield for $0 \leq x \leq 1 - Nt$ a system of $N + 1$ equations in $G(x + (j - 1)t, u)$, $j = 1, \dots, N + 1$, whose determinant is again $D_{N+1}(x, t, u)$ and whose column of right hand sides is the $N + 1$ dimensional vector $[(1 - x)u, 0, \dots, 0, (x + Nt)u]$. Now the sum of all the columns of $D_{N+1}(x, t, u)$ is the vector

$$(1, \dots, 1)(1 - u) + [(1 - x)u, 0, \dots, 0, (x + Nt)u];$$

hence the numerator for $G(x + (j - 1)t, u)$ in Cramér's rule is

$$D_{N+1}(x, t, u) + (u - 1)F_{N+1,j}(x, t, u)$$

by the rule for adding determinants with all but one column in common. In other words:

$$\begin{aligned} D_{N+1}(x, t, u)G(x + (j - 1)t, u) \\ = (u - 1)F_{N+1,j}(x, t, u) + D_{n+1}(x, t, u) \end{aligned} \quad (22)$$

for

$$0 \leq x \leq 1 - Nt$$

Similarly, for $1 - Nt \leq x \leq t$, substitute $N - 1$ for N in (22). (Note that substituting $u = 1$ yields $G \equiv 1$, as in Theorem 7.) Differentiating (22) with respect to u , substituting $u = 1$, and hence $G \equiv 1$, yields

$$D_{N+1}(x, t, 1) \frac{\partial}{\partial u} G[x + (j - 1)t, u] \big|_{u=1} = F_{N+1,j}(x, t, 1)$$

But

$$\frac{\partial}{\partial u} G(x, u) \big|_{u=1} = \sum_{n=0}^{\infty} n \{P_1(x, n) + P_2(x, n)\} = E(x)$$

and (17) follows by the usual substitutions.

REMARKS ON MODEL $H_{2\sigma}$ AND OTHER MODELS

Naturally, the analysis of $H_{2\sigma}$ (see Fig. 12) would more closely resemble that of H_{1t} than it would resemble the asymptotic case. However, its structure is completely different. Rather than embark on such an analysis, we will examine a special case which, though extreme, makes the difference in structure evident. Let us take $\sigma = 200$; that is, at each response α_i , the probability of that response at the next stimulus is tripled.

Here, if $\frac{1}{3} \leq p_1 \leq \frac{2}{3}$, absorption, whether by α_1 or α_2 , *must* occur at the first stimulus. Similarly, for each of the middle thirds of the remaining two intervals (that is, for $\frac{1}{3} \leq p_1 \leq \frac{2}{9}$ and for $\frac{7}{9} \leq p_1 \leq \frac{8}{9}$), absorption *may* occur at the first stimulus, but *must* occur by the second. Continuing this process, we see that 2^{n-1} intervals of length 3^{-n} may last through $n - 1$ stimuli but must be absorbed at the n th. If p_1 is expanded in radix 3, it can have, at most, two representations (occurring in the

case of a finite expansion). If either representation of p_1 uses a digit one, let the first occur at digit n beyond the ternary point. Then absorption must occur on or before the n th stimulus.

The residual set when all intervals of necessary absorption are removed is the well-known "Cantor Ternary Set". This nowhere dense set consists of the only points at which a "floundering" is at all possible. It is, therefore, intuitively evident that the probability of absorption is one.

If such a model is programmed for a digital computer and, as is most usual, the probability transformation is computed with finite precision, the preceding analysis requires the further complication of taking round-off into consideration. Thus, unlike the infinite precision analysis,² the description of the model does not require an infinite number of intervals. Nevertheless, we have sufficient indication that the structure is completely different from the finitely intervalled model H_{1t} . Indeed, it is evident that the same can be said for the general model $H_{2\sigma}$, at least when $\sigma > 100$.

We have seen that the absorbing boundary models, even when they looked linear, lost all semblance of linearity. Model H_{1t} involved non-analytic functions. It is an interesting conjecture that no continuous asymptotic models have such nonanalytic features.

An interesting nonlinear model is being studied by Bush, Galanter, and Luce. This model derives from an axiom system developed by Luce (1958).

In this model there are $n + 1$ "response alternatives", E^0, E^1, \dots, E^n , with the usual variable probability vector (p_0, p_1, \dots, p_n) . Since this model is closer to learning than the habit forming models we have been discussing, to each response alternative there will be $m + 1$ possible outcomes $E_{i0}, E_{i1}, \dots, E_{im}$ with a *fixed* probability vector $(\pi_{i0}, \dots, \pi_{ij}, \dots, \pi_{im})$, for which $\sum_{j=0}^m \pi_{ij} = 1$ for every i between 0 and n . The $(m + 1)(n + 1)$ reinforcement operators for all outcomes of all response alternatives are provided by the $(m + 1)(n + 1)^2$ functions

$$(T_{ij}p)_k = \frac{\beta_{kj}^i p_k}{\beta_{kj}^i p_k + (1 - p_k)}$$

where $\beta_{kj}^i = 1$ if $i \neq k$.

$$(\text{Thus } \beta_{kj}^i p_k + (1 - p_k) = \sum_{k=0}^n \beta_{kj}^i p_k)$$

² We note that most classical numerical analysis assumes infinite precision at some point.

"schematic" flow chart, in which such instructions as " $1 \rightarrow i$ " must modify a number of instructions.

As before, this random selector may be achieved more compactly with judicious use of storage and with one multipurpose loop. For example, the Bush-Galanter-Luce reinforcement model has been programmed for the University of Pennsylvania's Univac by P. Z. Ingerman in such a way that the random selector subroutine has essentially the schematic of Fig. 14.

CONCLUSION

A feature of general habit forming and learning simulation which has become evident is the possibility of separating the random selector portion of the simulation from the response type portion. The latter may be simulated by "experience retrieval" methods, by "reinforcement function" methods, or by a combination of both. From a machine point of view, then, we visualize separate sections, either in hardware or in programming, dedicated to these three functions. For example, simulator programs will be formed by assembling suitable subroutines of these three types, for each of which there may be a sizeable "library".

We might, therefore, have a master program—called a "compiler"—which would automatically assemble an appropriate selection from such a library at the call of a psychologist. Such a procedure would free the psychologist from one of the main problems natural scientists beyond the physical sciences have always had to face, namely the fact that experiments usually disturb unduly the system being controlled during the experiment. The main use of machine models is to run the experiment without fear of such disturbance. Psychologists can then choose the appropriate model from among many by comparing their outcomes with direct observations of the systems being simulated.

RECEIVED: August 26, 1958; revised March 16, 1959.

REFERENCES

- ASHBY, W. R. (1952). "Design for a Brain," Wiley, New York.
BUSH, R. R., AND FREDERICK MOSTELLER (1955). "Stochastic Models for Learning." Wiley, New York.
BUSH, R. R., FREDERICK MOSTELLER, AND G. L. THOMPSON (1954). A formal structure for multiple-choice situations. In "Decision Processes." Wiley, New York.
DELEEUW, K., MOORE, E. F., SHANNON, C. E., AND SHAPIRO, N. (1956). Computability by probabilistic machines. In "Automata Studies" (C. E. Shannon and J. McCarthy, eds.). Princeton Univ. Press, Princeton, New Jersey.

- ESTES, W. K. (1950). Toward a statistical theory of learning. *Psychol. Rev.* **57**.
- FELLER, W. (1957). "An Introduction to Probability Theory and Its Applications," Vol. 1. Wiley, New York.
- GORN, S. (1957). Standardized programming methods and universal coding. *J. Assoc. Computing Machinery* **4**, July issue.
- HARRIS, BELLMAN, AND SHAPIRO, (1953). Studies in functional equations occurring in decision processes. Rand Corp. Rept. P-382.
- HAUSDORFF, F. (1957). "Set Theory." Chelsea, New York.
- JUNCOSA, M. L. (1953). Random number generation on the BRL high-speed computing machines. Ballistic Research Lab. Rept. #855.
- NEWELL, A., AND H. A. SIMON (1957). Programming the logic theory machine; Proc. Western Joint Computer Conf.
- MINSKY, M. L. (1956). Heuristic aspects of the artificial intelligence problem. *Mass. Inst. Technol. Lincoln Lab. Group Rept.* **34-55**, December, 1956.
- TAUSSKY-TODD, O. AND TODD, J. (1955). Generation and testing of pseudo-random numbers. *Natl. Bur. Standards (U.S.) Rept.* **3370**.
- WALTER, W. G. (1953). "The Living Brain." Norton, New York