THE UNIVERSITY OF MICHIGAN

COLLEGE OF LITERATURE, SCIENCE, AND THE ARTS
Communication Sciences Program

Final Report

MACHINE ADAPTIVE SYSTEMS

John H. Holland
Arthur W. Burks
J. Willison Crichton
Marion R. Finley, Jr.

ORA Project 05089

under contract with:

DEPARTMENT OF THE ARMY
U.S. ARMY SIGNAL SUPPLY AGENCY
CONTRACT NO. DA-36-039-SC-89085
(Continuation of Contract No. DA-36-039-SC-87174)
FORT MONMOUTH, NEW JERSEY

administered through:

OFFICE OF RESEARCH ADMINISTRATION ANN ARBOR

May 1963

CONTENTS

INTRODUCTION: GROWING AUTOMATA THEORY

Growing automata theory can reasonably be said to begin with the work of A. M. Turing, "On Computable Numbers with an Application to the Entscheidungsproblem." Turing's theory originated in his investigation of expressions calculable by finite means. In order to give a rigorous formulation to this kind of calculation, he conceived of it as being carried out by a machine. A tape, divided into squares and indefinitely extendible, supplies instructions to the machine and records the results of the computation. The machine itself is considered to have only a finite number of distinguishable internal conditions, or states. At any moment of time it scans one square of the tape and under certain conditions it may cause a symbol to be printed on the tape. For each particular machine these conditions, and the actions of the machine as a whole, are determined from a table called the "machine description." The table can be thought of as having four columns:

(i) the first column lists the possible states of the machine;

(ii) the second column lists (for each state) the symbols which can occur on the tape square being scanned;

(iii) the third column specifies what action the machine is to take for each possible configuration of the first two columns -- the action consists of printing (or not printing) a symbol, and moving the tape one square to the right or left (or not moving);

(iv) the fourth column gives the new state of the machine (the internal state of the machine after it completes the action specified by the third column).

Turing identified the set of expressions calculable by finite means with the set of computations which his machines can perform; for every such expression there is a Turing machine capable of carrying out the corresponding calculation.

Thus every behavior or process that can be specified in mechanistic terms can be imitated by some Turing machine. Turing was able to go further. He was able to show that there is a Turing machine -- called a universal machine -- which when supplied with appropriate instructions on its tape can imitate any other Turing machine. Hence for every behavior specifiable in mechanistic terms there must be a set of instructions, a program, which will employ this universal Turing machine to imitate the behavior. Furthermore, there is a routine way of writing down these programs one by one in an ordered list. The position of the program in the list can then be used to index the corresponding behavior. (These behaviors can then be rightfully referred to as "effectively described" behaviors.) Thus the logician using Turing's approach is assured not only of being able to define and investigate rigorously all effectively described behaviors; he also has a convenient way to designate the behaviors via the index.

Though Turing machines provide strong foundations for a theory of automata they have several severe limitations as tools for the investigation of systems which are intended to mirror processes such as "learning" and "adapting." In such an investigation we aim to understand behavior as a product of the action and interaction of structural components. Thus the investigation depends directly upon the manner in which the organism's structure is represented in the theory. There may be little correspondence between the structure of a natural system which exhibits adaptation and the structure of the Turing machine which could imitate it behaviorally. The Turing machine is very much a single sequence device, generating complex behavior by executing simple operations, one after another. On the other hand, natural systems operate in a very parallel fashion, with a multitude of inter-dependent operations, going on simultaneously. In applying automata theory to the study of the characteristics of natural systems, or of systems which are to mirror some of the behavioral properties of natural systems, there is an additional criterion to meet: while preserving the breadth and precise definitions of Turing's approach, provision must be made for a reasonably straightforward representation of important spatially complex structural properties of adapting systems.

In order to provide for structural as well as behavioral representation, two further steps must be taken:

(i) The theory must be enlarged to allow description of many processes acting simultaneously and interdependently; in terms of the earlier correspondence, this means many Turing machines or programs interacting at one time.

(ii) In addition, the interrelations between the machines must be represented in the same structural sense as the machines themselves. This latter condition insures a decomposition of the formal representations into parts corresponding to the components of the natural system.

Such a broadening of Turing's approach can be accomplished by providing a "space" within which we can describe various structures and their interconnections. One approach is via a cellular space with a given set of "state-transitions" -- rules holding, without change, for each cell in the space. Such formulations of growing automata in a cellular space have been discussed by von Neumann (The Theory of Automata: Construction, Reproduction, Homogeneity), Church ("Application of Recursive Arithmetic to the Problem of Circuit Synthesis"), Burks ("Cellular Automata" and "The Logic of Fixed and Growing Automata"), and Holland ("Iterative Circuit Computers"). Such a cellular space can be used as an imbedding space in which to study the information processing properties of natural systems. It can be shown that the cellular spaces, with appropriate choices of the local "state transition" rules, can be treated as parallel computers. That is, just as the universal Turing machine is programmed by placing instructions on the squares of its tape, so can the cells of the cellular space be programmed. Different groups of cells can be allocated to different programs and all programs can operate simultaneously. From the formal point of view, then, there is little distinction in this space between a set of interacting programs and a connected set of components. One can write a program, in the cells of the space which not only simulates the behavior of a given system but does so by means of interacting subprograms which simulate in detail the local operations of the system's components. Any mechanism can be represented in detail, and to as great a precision as desired, by this means. It can be shown that several other formulations of growing automata occur as special cases of this format; thus the ideas embodied in these formulations are retained in the cellular spaces. Some of the cellular spaces fitting the foregoing description have been mathematically defined and so can serve as a basis for theoretical investigation.

How can a mechanism organize itself to meet conditions set by an environment largely unknown to it initially? How can a mechanism "adapt"? "Learn"? These questions can be recast in a form accessible to automata theory if we use the correspondence between general mechanism and Turing machines just discussed. Whenever a procedure for solving the problems set by the environment can be given a mechanistic explanation, i.e., whenever a system's behavior can be effectively described, there must exist a Turing machine which can simulate that procedure. But then recalling that the Turing machines can be listed or enumerated, there would seem to be a rather simple method of adapting to the environment in such cases. One need simply try the possible Turing machines, or the corresponding programs, one by one until a machine is encountered capable of meeting the conditions set by the environment. Such an "enumerative" approach will eventually satisfy any conditions set by the environment if the conditions can be satisfied at all by a mechanism. Still, as an answer to the question posed, the enumerative approach is certainly a false lead.

The flaw in the enumerative approach, and it is a fatal one, reveals itself in the extreme inefficiency of the method. Consider the relatively simple problem of learning to play a good game of checkers. The environment in this case is quite restricted and well defined. However, as Samuel has pointed out (in "Some Studies in Machine-Learning, Using the Game of Checkers") an attempt to enumerate all the lines of play in checkers would take approximately 10^{21} centuries even if several choices could be made every millimicrosecond. (Current estimates put the age of the Universe at around 10^{8} centuries.) Yet Samuel has written a program which learned to play tournament calibre checkers, and humans do manage to adapt to very complex environments in times considerably less than one century. Even though the enumerative approach always produces a solution if one exists, it quite obviously is not the method employed by any system of interest. In effect enumeration fails because the order in which various behaviors or programs are attempted is set *ab initio*. This order is not modified by any of the information the system gains as a result of its trials. In other words, the system is denied any opportunity to modify its strategy on the basis of information received.

The central role of information-processing in self-organization is thus affirmed at the very beginning of a theoretical investigation. This outcome emphasizes the inadequacy of just representing the system's internal processes in the theory. The environment (or range of possible environments), the information received therefrom, and the ways the system can affect the environment must also be represented.

The environment of a natural system, even when it is homogeneous in the large, involves important local variations. If the natural system is to survive it must at least partially control its interactions with the environment. Because the environment does vary, this control can only be *a propos* if the system receives and employs information about its environment. On the other hand, if the system collects information about the environment and *if the environment has some regularities* then the system has the possibility of departing from chance behavior. However, the rate of improvement of its control possibilities must at least be limited by the rate at which it processes relevant information about regularities. Let us call any such improvement in control, "adaptation" -- using the term in a very broad sense. Then the rate at which a system processes relevant information should set a limit on its rate of adaptation; a system can alter its organization no more rapidly than it receives the relevant information. Or so it would seem intuitively.

This suggests both a "criterion" for adaptation (i) and a "conjecture" (ii) to be proved:

(i) An adaptive system should seek out and exploit environmental regularities -- opportunities to depart from enumerative behavior -- and should proceed at random only where nothing better is possible. The process of adaptation is essentially that of locating and utilizing such regularities.

(ii) The rate at which a system obtains and uses information about regularities sets a limit to its rate of adaptation.

While this thesis may be suggestive, it is another matter to give it precision. In common with most heuristic arguments, the argument given has a very loose texture. As presented, it can hardly be proved or disproved and it is almost impossible to discern what elements of truth lie within it. In particular, although a relation between regularities, information processing rates, and adaptive efficiency is suggested, the precise nature of the relation and its usefulness can only be evaluated in a more rigorous context.

The Logic of Computers Group at the University of Michigan has been investigating adaptive systems via three interdependent research efforts: (1) an investigation of adaptation of mechanisms expressed in a structure of iterative circuit computers, (2) a program of research in the fundamental properties of various defined systems of growing automata, (3) simulation of a class of nerve nets.

In the first of these areas, a class of growing automata -- iterative circuit computers -- which includes a wide range of parallel computers, was examined in detail with special attention being given to the general principles of adaptation, realizable adaptive mechanisms, and lastly to problems of adaptive efficiency.

In the second of these areas fundamental questions of mathematics and growing automata theory were considered, viz. what are appropriate sets of primitives for systems of growing automata, what mathematical or logical languages are best suited for the description of the structure and behavior of such systems, what are the ultimate capabilities and limitations of variously defined systems, what properties of systems of growing automata can be established by mechanistic means?

In the third area of research, a study of idealized nerve nets by means of computer simulation was conducted. This research supplied relevant data for the more theoretical portions of the research program, tested the usefulness of concepts arising in the theoretical research, and in general served as an adaptive mechanism which could be examined in detail and altered at will according to dictates of the theoretical part of the research program.

I. ADAPTIVE SYSTEMS THEORY

* * *

Abstract: "A Logical Theory of Adaptive Systems Informally Described"

J. H. Holland

A description of a theory of machine adaptive systems embedded in a "space" of iterative circuit computers is presented. The structure and arrangement of the modules of a particular iterative circuit computer can be completely described by selection of five properties which determine a particular element of the mathematically characterized class of all such machines; sub-programs can be stored at different locations throughout the computers; any growing automaton can be simulated, in structure and behavior, by an appropriately chosen set of interacting sub-programs; these sub-programs can shift and combine as well as copy themselves. Generation procedures depend upon the modulation of parallel random processes whereby many programs are being generated and tested simultaneously. Connections of the generators from which sub-programs are formed is a function of "valence" and "activation". An indirect method of control of connection and disconnection is proposed. The process of random movement and connections producing trial programs can be described in a generation tree. The deliberate introduction of rate-controlling programs into the computer can be employed to alter the distribution of programs.

Adaptation takes place relative to an automaton and its environment, both described in terms of an embedding in an iterative circuit computer. The environment should consist of well-defined problems. If a trial program presents a solution or partial solution to a problem posed by its environment a corresponding pay-off of activation is released. Rate of accrual of activation can be made the basis for inducing generalizations of problem-solving procedures. The generalization process is accelerated when the environment of the adaptive mechanism is composed of graded sequences of problems. Sub-goal generation, where the mechanism divides the problem into sequences of sub-problems can be made a characteristic of the adapting system. The deterministic nature of the system allows the ready stating and proving of theorems. Probabilistic and random processes are not absolutely necessary but they often simplify the formulation and proof of theorems.

[This report appeared in full in "A Logical Theory of Adaptive Systems Informally Described," Technical Report, The University of Michigan Office of Research Administration, July 1961, 51 pages.]

Abstract: "Outline for a Logical Theory of Adaptive Systems"

J. H. Holland

The study of adaptation involves the study of both the adaptive system and its environment and should attempt to explain how systems can generate procedures enabling them to adjust efficiently to their environments. The theory outlined here is based upon formal definitions and investigations of (parallel) procedures for generating programs. Without any essential loss of generality the study can be restricted to systems wherein generation of programs is accomplished by the biasing of (parallel) random generation procedures. A generated population of programs will act upon a population of problems (the environment) in an attempt to produce solutions. For adaptation to take place the adaptive system must at least be able to compare generation procedures as to their efficiency in producing solutions. This can be accomplished by assigning to problems a "reward" which is released to the adaptive mechanism whenever the associated generation procedure solves the problem by means of one of its programs. To implement this central control, a supervisory program, acting over associated generation procedures can be provided; this program serves as an implicit definition of the distribution of problem-solving programs to be expected at any time. Two provisions are necessary in order to implement differential selection by this means; (1) supervisory programs must be capable of self-duplication in order to provide successive generations as grist for the selection principle, (2) the rate of duplication of a supervisory program must be determined by the rate at which it accrues rewards from the problem-solving programs it controls. With each supervisory program is associated a "substitution probability matrix" which gives the probability of substitution of one instruction for another, at each position,

during duplication. This, in effect defines neighborhoods of similarity for supervisory programs, and hence for the generation procedures they implicitly define. Under differential selection, the sequence of supervisory programs will tend toward the best supervisory program in the neighborhood of similar programs while an unsuccessful supervisory program will disappear from the population and also the chance of similar programs arising from it by modification will be eliminated. Realizations of these systems are attained by indicating precisely the way in which connection and disconnection probabilities, densities, generation rates, etc., are specified in systems of iterative circuit computers. These same systems of computers are also employed to realize the environment of problems of the adapting devices. The rate of adaptation of the device to the environment is limited by the rate of information accrual about the environment.

[This paper appeared in full, under the same title, in the Quarterly Progress Report for 1 August - 31 October 1961. Some of the contents of this paper were discussed at the Seminar on the Relationship Between Non-numerical Programming and the Theory of Formal Systems, held at Blaricum, The Netherlands, October 1961. A slightly revised version of this paper was published in the *Journal of the Association for Computing Machinery*, *9*, 3, July 1962, pp. 297-314].

Abstract: "Concerning Efficient Adaptive Systems"

J. H. Holland

The study of efficiency has lead to important results in related areas such as information theory; the study of efficiency also opens the way to the application of statistics and statistical mechanics to the generation procedures and generated systems of automata theory. The concept of efficiency is discussed here as it applies to a restricted subclass of adaptive systems in a framework of iterative circuit computers. It is shown that acquisition of new information generally forces a concurrent less efficient use of information already accumulated. The main result gives the rate of acquisition of information which (as a function of the concurrent loss of efficiency) produces optimal adaptation for the systems considered. Loosely, the result shows both that no system of the type considered should consign more than half of its effort to acquisition of new information, and that there are reasonable conditions under which this limit should be closely approached. It is believed that the methods employed in obtaining this result can be extended to some larger classes of adaptive systems.

[This paper appeared in full in the Quarterly Report for 15 April - 15 July 1962. Some of the results were presented at the second meeting on Self-Organizing Systems, in Chicago in May 1962. The paper was later published in Self-Organizing Systems - 1962, Spartan Books, pp. 215-230.]

* * *

Adaptive Efficiency

What kinds of supervisory programs correspond to strategies which seek out and use environmental regularities, and how can the adaptive system select a supervisory program appropriate to the environment confronting it? The following example indicates that selection based upon environmental regularities is possible:

Adaptation, in this example, is based upon differential selection of supervisory programs. In order to implement differential selection two requirements must be met:

(i) each supervisory program must be able to produce duplicates of itself;

(ii) the rate of duplication must be determined by how well the strategy defined by the supervisory program meets the conditions set by the environment.

There are no essential difficulties in defining iterative circuit computers and generators which satisfy these requirements. Under these conditions each supervisory program will produce several duplications of itself in the iterative circuit computers over an extended interval of time. If the generators composing the supervisory program are made subject to substitution upon duplication, then, during duplication variants can also emerge. If the strategy of a supervisory program is relatively successful it will duplicate, producing a population which contains not only copies of the original program but also modifications of it. A careful analysis will show that (for a time) the offspring of two programs with substitution rules which differ slightly, will also differ only slightly in average effectiveness in handling the environment. That is, supervisory programs similar according to this criterion will have offspring which have about the same average effect upon the environment. As a result, a sample drawn from some set of similar supervisory programs can be expected to give a good estimate of their adaptation. Now, if one of the modified offspring of a supervisory program develops a more successful repertory of problem-solving techniques, it will begin to predominate in the population of offspring because of its higher rate of duplication. The result will be that _new_ modifications will come increasingly from the modified program. Thus, under differential selection, the sequence of offspring can "tune in" on good supervisory programs while only "trying out" a relatively small number of programs in any neighborhood of similar supervisory programs. On the other hand, not only will an unsuccessful supervisory program disappear from the population (because it does not duplicate or does so inefficiently) but, at the same time, the possibility of similar programs arising from it by substitution is eliminated. In effect the entire neighborhood of the program is selected against. The combination of the "tuning in" effect with this "avoidance" effect enables the adaptive system to seek out and use regularities in its environment.

We saw earlier that an enumerative strategy could eventually adapt to any environment or set of conditions which can be satisfied by an "effectively-described" behavior. The present system is demonstrably faster than any given enumerative process over all but a vanishingly small proportion of such environments which are homogeneous in the large. But is it still "too slow"? This question is still largely unsolved. The problem of determining whether the system just examined is "too slow" is part of the large problem of determining what strategies are "efficient" for various kinds of environments. It is relatively easy to be convinced of the inefficiency of the enumerative approach since it is inefficient in almost every environment. However, for a more complex model, it is generally quite difficult to decide whether it will be too inefficient (too slow) to serve as an explanation of some observed behavior. At present, efficiency criteria are not available for any very considerable class of adaptive systems. For example, we do not have even approximate results for evolutionary systems based upon natural selection. Yet, discussions and criticism of theories of adaptation in natural systems as currently conceived often turn on questions of efficiency. Similar statements apply to the rate of information and selection of cell assemblies in lebb's theory and to related computer simulations of the central nervous system.

At the base of any discussion of adaptation is the assumption of an environment unknown to the adaptive system in one or more of its characteristics. If a part of the environment is to be designated as unknown, there must be several possible alternatives for that part. Each distinct possibility in effect designates a distinct environment that the system may possibly face; the set of all possibilities constitutes the range of environments the system must be prepared to face. In order to establish an efficiency criterion one must be able to assign an "optimal rate of adaptation" to this range of environments. For some environmental ranges there may be a consistent way of defining an overall optimal rate, while for others it may be necessary to resort to average or minimax rates. In any case it is the behavior of the system over the whole range that is of interest, not its behavior in one or another of the particular environments in the range. One way of approaching this problem is to attempt to discover adaptive systems which are the best possible ("universally efficient") over (almost all of) some wide range of environments. Such machines, if they exist, could play much the same role here that the "efficient codes" play in information theory.

Recent Research

Preliminary results indicate that a process of differential selection on rating polynomials provides an adaptive process which behaves in "near-optimal" fashion over the entire range of tree-search environments:

Let S_{ij} be the state assigned to vertex j at level i of the tree.

Let U_k be the utility or pay-off assigned to the kth termination of the tree

$\Theta_h : \{S_{ij}\} \rightarrow$ rationals, ith parameter of the rating polynomial

$B = \{b_1, \ldots, b_n\}$ = df. a (finite) set of weights which can be applied to parameters

$V(S_{ij})$ = df. $\sum_{h=1}^{m} a_h\Theta_h$ where $a_h \varepsilon B$,

the rating polynomial based upon the parameter set $\{\Theta_h\}$ and the weighting set B

[cf. Samuel, A. L. "Some Studies in Machine-Learning, Using the Game of Checkers," IBM J. Res.Dev., 3, 3, 210-229 (1959), for a good discussion of rating polynomials as a strategy for playing checkers.]

Assume each individual polynomial of type γ achieves an expected pay-off u_γ after each traverse of the tree.

Employ a parallel-processing procedure (realizable along lines suggested in "Outline for a Logical Theory of Adaptive Systems") wherein a set of polynomials is played against the tree at each time t.

Let $N_\gamma(t)$ = df. the number of polynomials of type γ at time t. (For most γ at any given time t, $N_\gamma(t) = 0$).

New polynomials will be formed for the play at time t+1 by allowing each individual polynomial to duplicate in proportion, k, to the pay-off it achieves at time t; thus a polynomial of type γ will produce ku_γ duplicates. These polynomials are to undergo a process of random recombination to yield the set of polynomials for time t+1 (cf. Fisher, R.A., The Genetical Theory of Natural Selection, Ch. II; Dover).

$$N_\gamma(t+1) = N_\gamma(t)u_\gamma k$$

or in the limit $\dfrac{dN_\gamma}{dt} = N_\gamma(u_\gamma k-1)$

or $$\frac{dN_\gamma}{N_\gamma} = u_\gamma k-1$$

but, using Fisher's methods

$$\frac{dN_\gamma}{N_\gamma} = m_\gamma$$ where m_γ is Fisher's "fitness" parameter.

$$\bar{u}(t) = \frac{\Sigma_\gamma N_\gamma(t)\ u_\gamma}{\Sigma_\gamma N_\gamma(t)} = \frac{\Sigma_\gamma N_\gamma(t)\ [m_\gamma+1]\frac{1}{k}}{\Sigma_\gamma N_\gamma(t)}$$

$$= \frac{1}{k}\left[\frac{\Sigma_\gamma N_\gamma(t)m_\gamma}{\Sigma_\gamma N_\gamma(t)} + 1\right] = \frac{1}{k}(\bar{m}(t) + 1)$$

But then $$\frac{d\bar{u}(t)}{dt} = \frac{1}{k}\frac{d\bar{m}(t)}{dt}$$

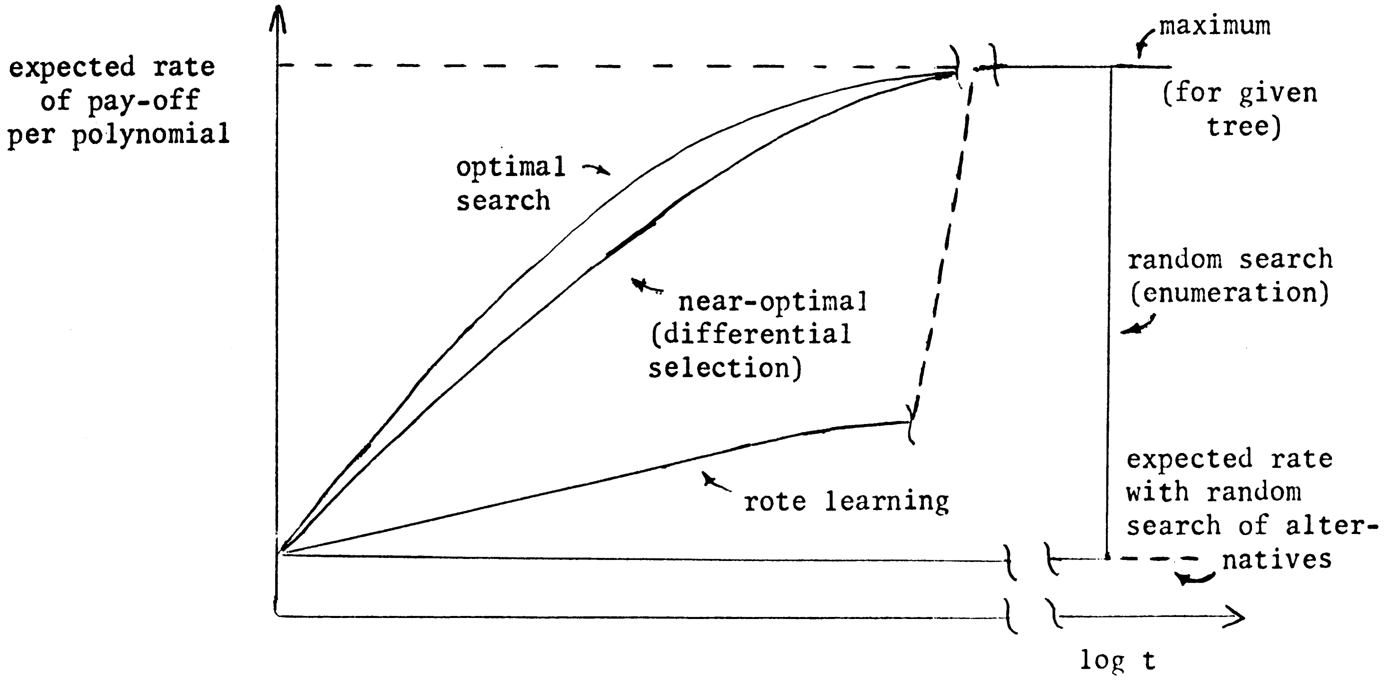
Fisher's fundamental theorem shows

$$\frac{d\bar{m}(t)}{dt} \simeq W(t)$$

where W is the "genetic variance" defined by Fisher (which can be applied here to the weights assigned to each parameter, i.e. weight b_u applied to parameter Θ_v corresponds to allele (u,v) in Fisher's treatment)

Thus $$\frac{d\bar{u}(t)}{dt} \simeq \frac{W(t)}{k}$$

Following this line one can apparently prove the following relationships:



THEORY OF GROWING AUTOMATA

* * *

Abstract: "John von Neumann's Theory of Automata"

A. W. Burks

During the last years of his life John von Neumann made many contributions to the theory of automata. He was led into the automata and computing field by his studies in fluid dynamics; finding that existing analytic methods of solution of the pertinent non-linear partial differential equations were inadequate he began employing computers to solve crucial cases numerically; by this means he hoped to obtain insight into the nature of the underlying mathematical difficulties. He believed that this "heuristic" use of computers had broad implications for mathematics in general, and, once interested in computing, he made contributions to all aspects of the subject and its technology: computational algorithms and programs, matrix inversion, Monte Carlo methods, solving linear systems of equations by random sampling techniques, computer design, automata theory.

In the theory of automata he worked in particular on the problem of securing reliability from unreliable components and on the problem of automaton self-reproduction. Von Neumann thought that the mathematics of automata theory should start with mathematical logic and move toward analysis, probability theory, and thermodynamics. When developed, automata theory would enable us to understand automata of great complexity, in particular the human nervous system. [This paper appeared in full (57 pages) in the Quarterly Progress Report for 1 August-31 October, 1961.]

Abstract: "Programming and the Theory of Automata"

A. W. Burks

A. M. Turing showed that a theoretical machine, consisting of an indefinitely long *tape* and a *finite control automaton*, could, by means of a *universal control automaton*, behave as a *universal simulator*. It is not reasonable always to identify Turing machines and general-purpose computers. Turing machines can compute infinite sequences, actual computers cannot. The "special-purpose computer" versus, "general-purpose computer" distinction is not the same as the distinction between a Turing machine and a universal Turing machine. The distinction between the two kinds of computers is a bifurcation of the class of universal control automata based on a practical criterion. The "program" versus "data" distinction in computing is quite arbitrary from a formal point of view. Automatic programming is an instance of the Turing universal simulation result with the additional requirement that the language of the automatic program be more natural for a human than the machine programming language. There is as yet no equivalent machine construction language which would allow the precise description of machines through a language convenient to humans. Such a language should be created. The Turing universal simulation result can be shown to hold for von Neumann's cellular automata. Von Neumann showed how to design a *universal constructing automaton*. Such results for growing automata systems can be used in obtaining a *machine design language* where the structure of the designed machine would parallel the structure of the problem (perhaps directly incorporating spatial concepts of the problem and of features of the problem such as memory organization). This machine description could then be supplied to a general-purpose computer which would translate it into its own machine language and solve the problem.

[This paper appeared in full in the Quarterly Progress Report for 15 April-15 July, 1962. Some of the ideas appearing in this report were discussed in the Seminar on the Relationship Between Non-numerical Programming and the Theory of Formal Systems, at Blaricum, The Netherlands, October 1961. A slightly rewritten version of the report has been published in *Computer Programming and Formal Systems*, (Braffort and Hirschberg, eds.), Studies in Logic Series, North-Holland Publishing Company, Amsterdam, 1963, pp. 100-117.]

Abstract: "Toward a Theory of Automata Based on More Realistic Primitive Elements"

A. W. Burks

Idealized primitive elements and systems of these elements are first considered, after which sets of primitive elements which simulate more closely both artificial and natural systems and processes are considered. The computational powers of realistic systems as opposed to idealized systems are compared. All the systems considered are synchronous, with time taking on non-negative integral values.

An idealized automaton is constructed from instantaneous "nan" switches, unit delays, and unlimited branching. Both finite and infinite automata are considered.

The behavior of an automaton is the mapping it produces from infinite input histories to infinite output histories. The computation of an automaton is defined in terms of an input tape reader and an output tape writer. Both tape units are controlled from the automaton, and both can move their tapes in only one direction. Computation is an important concept for Turing machines as well as for finite automata.

A slow automaton is constructed from the following primitive elements; a "nan" switch followed by n unites of delay, a simple unit delay, and a branching element having two outputs and b units of delay. First algorithm: when given an idealized finite automaton A it produces a computationally equivalent slow automaton A' . The rate of computation of A' is a fraction of that of A. Second algorithm: given an idealized finite automaton A whose

read control is periodic, this algorithm produces a computationally equivalent automaton A" which computes at the same rate as A. The second algorithm is due to R. McNaughton. The preceding algorithm illustrates the possibility of interchanging speed and equipment.

A probabilistic automaton has probabilistic elements rather than deterministic elements. Each element produces the wrong answer at each moment of time with probability ε. A theorem about regular Markov chains is applied to probabilistic automata to show that the memory span of a finite probabilistic automaton is finite.

The reliability problem is that of synthesizing a probabilistic automaton which is more reliable than a given automaton, i.e., one that will forget more slowly and compute more accurately. The two preceeding algorithms are combined with J. von Neumann's multiplexing procedure to get two new algorithms. First algorithm: given a finite idealized probabilistic automaton A this algorithm produces a probabilistic automaton A* which is more reliable than A; if A and A* were deterministic they would be computationally equal. A* computes at a fraction of the speed of A. Second algorithm: given a finite idealized probabilistic automaton A whose read control is periodic this algorithm produces a probabilistic automaton A** which is more reliable than A; if A and A** were deterministic they would be computationally equal. A** computes as fast as A. Both algorithms illustrate a gain of reliability at the cost of equipment.

Attention is given to automata which sense and act, and to cellular automata, following ideas of J. von Neumann.

[This paper appeared in full in the Quarterly Report for 15 July - 15 October, 1962. Some of the material was presented at the University of Michigan Summer

Conference Course in "Parallel Computers, Automata, and Adaptive Systems," in June 1962, and also at the 1962 Congress of the International Federation for Information Processing, at Munich, Germany, August 1962. The complete version of the paper will appear in the proceedings of the Munich conference.]

III. NERVE-NET SIMULATION

*   *   *   *   *

Abstract: "Cell Assembly Studies Using Several Thousand Simulated Neurons"

J. W. Crichton

This study has its origin in a program of research conducted at IBM (Rochester, et al., IRE Trans. I.T., 2, 3, September 1956) where a system of up to 500 neurons, each with 10-15 connections was simulated on a large digital computer. The present study employs a revised program so that a system of 10,000 neurons, each with approximately 150 synaptic connections can be simulated. With this larger program cell assembly and phase sequence structure hypotheses will be studied. The individual neurons of the model have the following characteristics: all or none activity; absolute and relative refractory periods; fatigue factor; spatial summation of exitatory and inhibitory stimuli; behavior in discrete time; alteration of synaptic value by law of effect (a fiber which participates in the firing of a neuron is made more likely to aid in future firing of the neuron). In the large, the neurons are distributed evenly over a surface or set of surfaces, and the distribution of fibers over neurons is locally random with a density which is a function of distance and direction. The program allows layered models of separated systems. Any set of neurons can be selected as input or output region.

Such a large simulation has required a great deal of programming ingenuity; certain neuron "grouping" assumptions have had to be made. This simulation approach, however, once the program has been completed, allows for great flexibility in experiments. Experiments planned include differentiating input, processing, and "muscular" parts of the system and attempting to obtain "fixation" of a single stationary object. Other, far more difficult experiments may be attempted: selection of alternative objects; net training by reward and punishment.

[The full version of this paper appeared in the Quarterly Progress Report for 1 August-31 October 1961. A slightly modified version of this paper was presented at a symposium on computer simulation of biological processes at the meeting of the American Psychological Association, in New York, September 1961.]

Cell Assemblies

The following might be given as the fundamental hypothesis underlying the present nerve-net project: The critical properties of a neuron system capable of versatile adaptive behavior are not the individual properties and relations of single neurons, but the tendency to develop semi-autonomous subsystems, and the relations among such subsystems. The term "cell-assembly" has been used to refer to a minimal semi-autonomous subsystem. Informally characterized, this is a system of cortical (association layer) neurons which are capable of acting as a closed autonomous functional unit for a brief period of time. These neurons are anatomically diffuse, but functionally connected. The functional unity of the cell-assembly results from the initial existence of the proper inter-connections among the neurons of the system together with a particular (i.e. selective) sequence of cortical events that forces these neurons to act briefly as a unit via a growth of synaptic strength at the connections such that after a period of time the assembly may be activated by appropriate excitatory stimuli.

Some necessary properties and relations of cell-assemblies appear to be the following: within a cell-assembly synaptic connections are predominantly excitatory and relatively strong; between cell-assemblies which have interconnections the connections are predominantly inhibitory, so that neighboring cell-assemblies tend to be competitive; the fatigue of individual neurons is essential in providing for the successive activation of several cell-assemblies over a period of time. This last is a critical aspect of the behavior of cell-assemblies in larger subsystems which correspond more closely to conceptual units (cf. Hebb's discussion of phase-sequences).

The formation of such subsystems requires that the individual neurons, through the mechanism of synapse value change, should tend to become more sensitive to patterned stimuli and less sensitive to random stimuli. This property is the basis for the experiments described below.

Experimental Results

In the late winter and spring of 1963 work continued on the three-neuron experiment, in the course of which a substantial modification of the experimental configuration was found to be necessary; this modification led immediately to successful results. The basic idea underlying the three-neuron experiment is that due to the recovery characteristics of neurons, a stimulus S which fires less frequently to a given neuron, C, than do other stimuli to the same neuron carries more information in the sense that C tends to become responsive to S only. This means that the synapse levels relating S to C rise to high values, while those relating the other stimuli to C fall to very low or zero values. This is an extremely important property of the model that is essential in the behavior of cell assemblies. The initial conceptualization of this experiment took the following form:

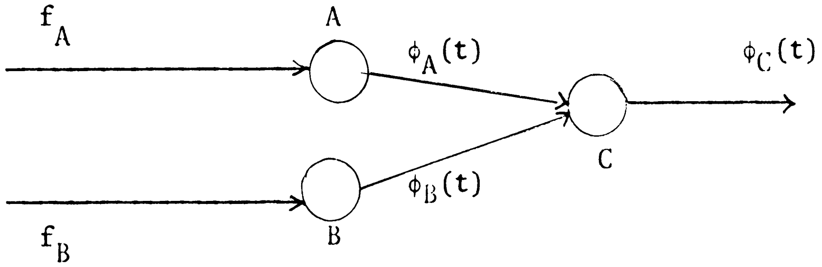


Figure 1.

Neurons A, B, and C are connected such that A and B each send one line (synapse) to C; $\phi_A(t)$, $\phi_B(t)$, and $\phi_C(t)$ are the outputs of neurons A, B, and C respectively at time t. f_A and f_B are the probabilities of stimulating A and B respectively; let $SL_{AC}(t)$ and $SL_{BC}(t)$ be the synapse levels from AC and BC respectively. The basic experimental hypothesis is that if f_A and f_B are chosen such that $f_B < f_A$ (and $1/f_B$ is less than the total recovery period) and if the SL's are choosen initially at moderate values such that a joint firing of A and B is required to fire C at the next time step, then eventually $SL_{BC}(t)$ will rise to a large value, $SL_{AC}(t)$ to a small value with the effect that the output of C correlates to that of B, i.e.,

$$\phi_B(t) = 1 \Longrightarrow \phi_C(t+1) = 1$$

$$\phi_B(t) = 0 \Longrightarrow \phi_C(t+1) = 0$$

Two experimental tests of this hypothesis were devised. In the first, f_A and f_B are simply constant over time, f_B in the interval $(\frac{f_A}{2}, f_A)$. In the second a moderate value of f_A is chosen and f_A and f_B were related as follows: $f_A = f_B$ over the time intervals [xm, xm+k] where x = 0, 1, 2, ...; $0 < k < m$ (k,m integers); $f_B = 0$ otherwise.

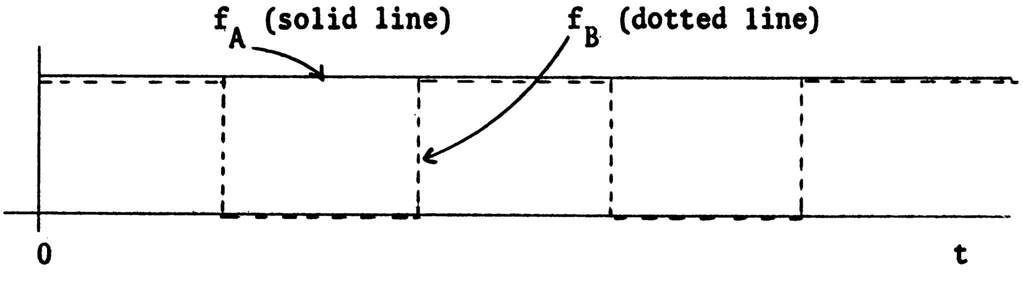


Figure 2.

Results: The first experiment did not produce consistent results, therefore we went over to the second. The early results for the second again were not consistent, but were suggestive: we learned that the probabilities of synapse change up should be much greater than those for down. However, even with this change, consistent results were not forthcoming and we finally realized that with two neurons alone driving C the hypothesis would not in general hold for the reason that the probability of joint firing of A and B is so small, i.e., $f_A \cdot f_B$, that the terminal values of SL_{AC} and SL_{BC} could never be reached, and in fact the contrary of the hypothesis would prevail in some cases -- i.e., C would correlate to A.

Modification of the experimental configuration: The original choice of a three-neuron net was made just for simplicity; however, the results above forced us to replace the single neurons A and B by groups of neurons: The configuration remains the same, except that A and B are now blocks of neurons A_i and B_i (i = 1, ..., n; n = maximum of 12), each sending out one synapse to C with synapse level SL_{A_iC} or SL_{B_iC}, with output $\phi_{A_i}(t)$ or $\phi_{B_i}(t)$, etc. The frequencies f_A and f_B apply uniformly to each neuron of group A or B, however, i.e., $f_{A_i} = f_A$ and $f_{B_i} = f_B$ for all i.

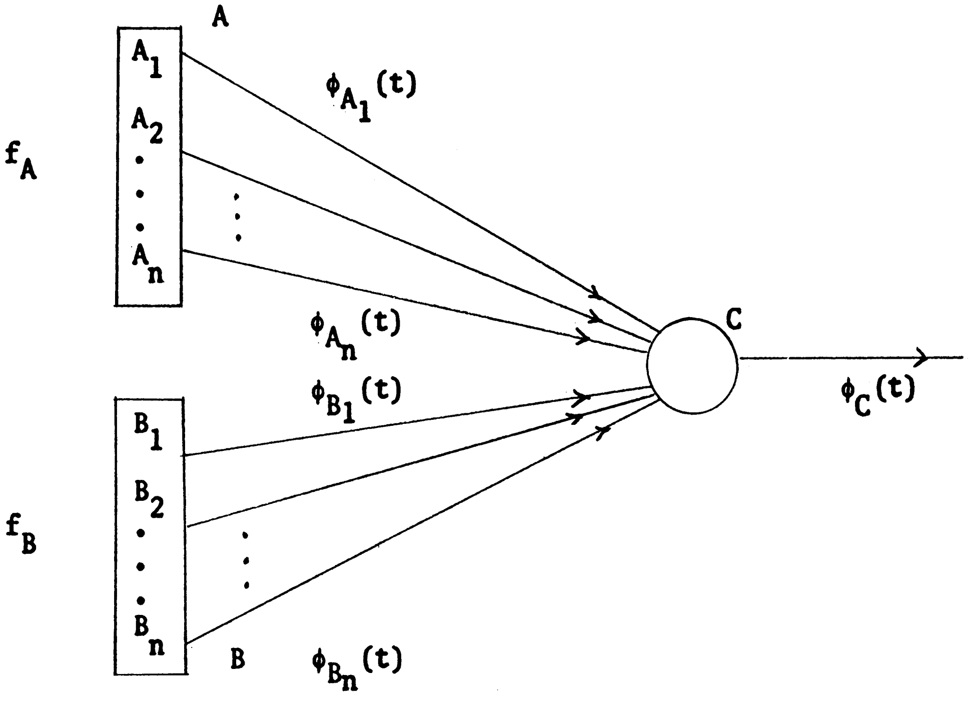


Figure 3.

The experimental hypothesis is just as before, only it applies to the two groups A and B, instead of to two single neurons. The difficulty mentioned above arising from the low probability of joint firing is avoided since the firing of a neuron of group B has a fairly good chance of coinciding with that of a neuron of group A, hence, the synapse levels from group A will rise, etc.

Results: The second experiment described above is currently in progress. It has been broken down into two cases: (a) that in which the neurons in group B fire in synchrony, and (b) that in which the curve in Figure 2 above represents a true envelope -- i.e., the neurons of B fire individually with probability f_A in the intervals [xm, xm+k] and with probability 0 otherwise. Case (a) so far has checked out correctly. Case (b) will be tested in the immediate future.

Analytical Investigations

Recently we have begun some analytical investigations which promise to provide a valuable supplement to the experimental program.

The first of these concerns the application of finite Markov chain theory to the relatively simple nets of the first experiments. The results of this theory that are particularly *a propos* are those that give the expected waiting times for absorption into an ergodic sub-chain or absorbing state for stationary processes: this is just the situation of the first experiment where the hypothesis states that the net will achieve certain final states -- basically a certain range of synapse levels -- within a finite period of time. The main problem here reduces to the writing down of the transition matrix of the process or making a reasonable approximation to it via possible state reductions, etc., since the total number of states is large even for the simple nets of the first experiment. These results should provide some deeper insight into the relationship between the frequencies f_A and f_B, since the waiting time above will be in part a function of these. This work is in progress now and we hope to finish it shortly.

Another analytical approach is aimed at determining conditions on the parameter of a net related to the existence of semi-autonomous subsystems such as cell-assemblies. The main device on which this approach is based is that of considering the distribution of neurons belonging to the subsystem over equivalence classes with respect to recovery . That is, we consider the number of neurons which fire at a given time t, t-1, t-2, etc. By assuming a steady-state pattern of activity, one can determine some properties of this distribution. These can then be related to the threshold function to derive conditions on the density of interconnections within the subsystem. The more difficult question whether small deviations from the steady-state pattern lead to still further deviations (instability) or back to the steady-state pattern (stability) is also being investigated. It is hoped that, at the least, these investigations will provide estimates of desirable net parameters and thus obviate a certain amount of exploratory experimentation.

* * * *

DISTRIBUTION LIST

(One copy unless otherwise noted)

OASD (R&E), Room 3E1065
The Pentagon
Washington 25, D.C.
Attn: Technical Library

Chief, Research and Development
OCS, Department of the Army
Washington 25, D.C.

Commanding General
U. S. Army Materiel Command
Washington 25, D.C.
Attn: R&D Directorate

Commanding General 3
U. S. Army Electronics Command
Fort Monmouth, N.J.
Attn: AMSEL-AD

Commander 10
ASTIA
Arlington Hall Station
Arlington 12, Virginia
Attn: TIPCR

Commanding General
USA Combat Developments Command
Fort Belvoir, Virginia
Attn: CDCMR-E

Commanding Officer
USA Communication and Electronics
Combat Development Agency
Fort Huachuca, Arizona

Chief, U.S. Army Security Agency 2
Arlington Hall Station
Arlington 12, Virginia

Deputy President
U.S. Army Security Agency Board
Arlington Hall Station
Arlington 12, Virginia

Director
U.S. Naval Research Laboratory
Washington 25, D.C.
Attn: Code 2027

Commanding Officer and Director
U.S. Navy Electronics Laboratory
San Diego 52, California

Aeronautical Systems Division
Wright-Patterson Air Force Base, Ohio
Attn: ASAPRL

Air Force Cambridge Research
Laboratories
L. G. Hanscom Field
Bedford, Massachusetts
Attn: CRZC
Attn: CRXL-R

Hq., Electronic Systems Division
L. G. Hanscom Field
Bedford, Massachusetts
Attn: ESAT

APSC Scientific/Technical Liaison
Office
U.S. Naval Air Development Center
Johnsville, Pennsylvania

Commanding Officer
U.S. Army Electronics Materiel
Support Agency
Fort Monmouth, New Jersey
Attn: SELMS-ADJ

Director, Fort Monmouth Office
USA Communication and Electronics
Combat Development Agency
Fort Monmouth, New Jersey

DISTRIBUTION LIST (Concluded)

Corps of Engineers Liaison Office
U.S. Army Electronics Research
and Development Laboratory
Fort Monmouth, New Jersey

Marine Corps Liaison Office
U.S. Army Electronics Research
and Development Laboratory
Fort Monmouth, New Jersey

Commanding Officer
U.S. Army Electronics Research
and Development Laboratory
Fort Monmouth, New Jersey
Attn: Logistics Laboratory
Attn: Director, Research/Engineering
Attn: Technical Documents Center
Attn: SELRA/SL-NPE 4
Attn: Technical Information Div. 3
Attn: Mr. Anthony V. Campi 14

Commanding General
U.S. Army Materiel Command
Washington 25, D.C.
Attn: R&D Directorate
Research Division
Electronics Branch

Rome Air Development Center
Griffiss Air Force Base, New York
Attn: RAALD, Documents Library

Professor Heinz Von Foerster
215 Electrical Engineering Research
Laboratory
University of Illinois
Urbana, Illinois

Dr. Saul Amarel
RCA Laboratories
Princeton, New Jersey

IBM Research Laboratory
Bethesda, Maryland
Attn: Dr. A. Babbitt

Sterling Perkes
Department of Electrical Engineering
University of Utah
Salt Lake City, Utah