LA-4879
C. 3

Synergy and Artificial Intelligence

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

UNITED STATES
ATOMIC ENERGY COMMISSION
CONTRACT W-7405-ENG. 36

Printed in the United States of America. Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy $3.00; Microfiche $0.95

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

LA-4879
UC-32
ISSUED: March 1972

Synergy and Artificial Intelligence

by

Donald R. Conant

SYNERGY AND ARTIFICIAL INTELLIGENCE

by

Donald R. Conant

ABSTRACT

This paper proposes that a high-level artificial intelligence may be constructed within the next few decades. Developments in computer hardware have occurred so rapidly that hardware with greater memory and information-processing speed than the human brain will probably be available in the near future. New ideas are proposed involving the relationship of intelligence and consciousness to physical systems, probability analysis as the fundamental basis for learning, and the need for a heuristic approach to all levels of the artificial-intelligence problem. A simple logical method for obtaining high reliability within the computer is analyzed. Some of the properties associated with natural intelligence are also discussed.

It is proposed that if artificial intelligence is developed to a high level within the next several decades, the result is likely to be more significant to society than any other area presently under investigation.

I. INTRODUCTION

Any system, including our world, the animal kingdom, the human race, United States society, the family, or the individual, may be considered as an active physical system subject to organized control, with the major goal and reward (for biological systems, at least) being survival. Improved organization can improve a system's chances for survival and increase the richness of that survival.

The instincts and drives, social customs, intelligence capabilities, value systems, etc., derive from evolutionary processes that tend to increase the capacity for survival. Survival is thus the only final arbiter of standards of excellence.

The study of better ways of organizing and expanding our control over our environment is essential for maximum successful advancement of our society and race. Because no organization can stand still without effectively falling behind as other organizations continue to evolve, the greater a system's capacity for discovery, change, and advancement, the better are its chances for survival and success.

Research into organization and control is a field that will probably be near the center of society's real needs for a long time to come, and results from such research should have a continuing impact upon society.

Artificial-intelligence research (perhaps better called simply intelligence research) studies the organization and active interaction of information, and the active control of physical processes by information.

The most advanced and capable organizations of which we are aware are human individuals and the advanced national and international societies. These have a capacity for learning and for significant advance through creative discovery. As a result of these capacities, high-level organization and control has developed. Society's knowledge and culture have been maintained through the generations because of the individual's capacity to learn from formal or informal teachers. The individual's capacity for creative discovery without the aid of a teacher has enabled society to advance its capability.

A major long-range goal, then, of artificial-intelligence research is to develop computer systems to the point where they can be a major aid in rapid creative discovery. Ultimately, advanced computer systems should have access to a large part of society's knowledge and should control directly many aspects of experimental research and of manufacturing.

Some advantages to be gained from this area of research in the near future might be:

- less computer rigidity and broader scope to optimize the amounts of work done by the computer and the programer (to reduce the amount of programer input required by the compiler in setting up an adequate program in the basic machine language),
- increased computer capability to accept incomplete information and determine how best to complete the information and continue the program, or to communicate what is incomplete and request additional information.
- greater numbers and types of properties with which the computer can deal,
- optimized coding methods for storing information so as to ease cross-referencing and increase the data-storage capacity of a given memory,
- optimized pattern-recognition methods for picture processing,
- greater human language input that the computer can "understand,"
- increased computer capacity for carrying out deductive logic,
- increased capacity for inductive logic,
- increased capability for self organization,
- increased capability for judging importance through statistical analysis,
- increased capability for self-programming,
- increased capability for judging job priorities,
- increased capability for pursuing advanced goals using more sophisticated techniques,
- new capacity to learn through means other than direct programming.

Possible advantages to be gained somewhat later are:

Information Retrieval. The large memory capacities and high information-processing speeds of present computers could greatly aid in scanning scientific literature. Developments in this area could enable the computer to recognize better what is being searched for; present techniques are not usually adequate for specifying and identifying the objects of the search. The most useful information-retrieval systems will be capable of "conversing" with users to make certain that requests are well-defined and "understood."

If the computer could "understand" human language input, much information now only so recorded would be available for computer use without need for human translation.

A computer that understood human language would also be more accessible to people who were not trained in specialized computer languages.

Mechanical Translation. Translation from one language to another requires that the translator contain a reasonably intelligent model of the world in order to reduce ambiguities in written text to sensible maaning that can be translated into another language.

Advanced Control of Experiments. If low-level computers monitored experiments without human surveillance, then access to an advanced computer should provide higher-level decision capability at critical points in the experiments.

Robot Development. Mechanized intelligent systems that operate in real time and are capable of motion and control in the environment should be useful for work in areas where it might be inconvenient or dangerous to send a man. On unmanned space-exploration missions, such devices may make the most efficient use of the meager capacity of communication channels between Earth and distant planets by reporting back only significant information.

Accelerated Instruction. The knowledge and capability of the computer may readily be increased once it has been given the capacity to understand language and the ability to learn. Humans can then teach it more or less directly rather than through laborious and time-consuming detailed programing.

Optimization. Research in artificial intelligence, heuristics, significance of information, and optimization of coding and procedures should ultimately optimize the methods used for solving whole classes of problems, and the ways in which the results are specified and interpreted.

Organization. Another important development of artificial-intelligence research should be better organization of information or knowledge. Ultimately, this should enable all of human knowledge to be not only better organized, but also much more systematically surveyed.

Learning. Artificial-intelligence research should lead to a better understanding of the various levels and types of learning. Such understanding may enable more effective teaching methods to be developed. Furthermore, an understanding of learning may provide some of the knowledge necessary for understanding the working of the human brain.

Creative Discovery. The most advanced type of learning involves discovering new important facts without the aid of formal or informal teaching, and formalizing the new discoveries by expressing them in terms of characteristics common to one or more classes of phenomena. The expression of facts in terms of common characteristics is the essence of formalization. Advances in our understanding of this area should increase our capability for new discovery. Programing this capability into advanced computers should still further increase our rate of advance.

Long-term major developments that an advanced artificial intelligence might help with are:

- discovery of better defense systems for protection against enemy attack,
- development of microtechnology so that computer components can be packed to higher density and have shorter cycle times, and so that discoveries at the molecular level can be made more rapidly thereby increasing the rate of advance of molecular biology research (such developments should help create advanced computer and control hardware and help lengthen the human life span),
- discovery of methods for making high-temperature nuclear reactions occur under controlled conditions so that not only energy but transmutation of the elements can be obtained (such developments would improve the self-sufficiency of our nation),
- research in psychology, sociology, and anthropology to understand the directions that organization among human beings should take to increase the individual's general satisfaction and capability. Such research may well be aided by developments in artificial-intelligence research that clarify the relationships among goals, strength of motivation, capability, success, and satisfaction. A better understanding of the logic of learning may also help.

Most U. S. scientists seem to feel that the development of a highly intelligent device is probably too difficult for the present generation. Some reasons for this attitude may be that the rigid step-by-step operations of a computer seem far removed from the concepts of consciousness, learning capability, and intelligent decision making, and that much regarding the logic and operation of the human brain is still unknown. Furthermore, development of intelligent devices appears to be proceeding quite slowly.[1]

However, a recent survey[2,3] indicates that 35 out of 42 people engaged in artificial-intelligence research estimated that 10 to 100 years would be required to develop a machine that could approximate adult human standards of intellectual performance.

For the following reasons, I believe that major advances toward the development and use of artificial intelligence are possible within the next few decades. Developments in computer hardware have occurred so rapidly that hardware with greater memory and information-processing speed than the human brain will probably soon be available. Past barriers to software development should be lowered by new ideas offered in this paper concerning the relationship of conscious intelligence to matter and the importance of probability analysis and multiple strategies for rapid learning. A complete knowledge of the human brain's organization is not necessary for the development of artificial intelligence if the bases for consciousness, learning, and intelligence are understood. Artificial-intelligence programs use techniques that can be handled by existing computers, although some of these techniques are undeveloped because they are seldom used for more conventional programs.

II. INTELLIGENCE

"Synergy" is the only word in our language that refers to the behavior of a whole system unpredicted by the separately observed behavior of its parts, according to R. Buckminster Fuller.[4]

One example where a synergistic viewpoint is important is the storage of information. The information that can be associated with one bit or with many random bits is very small. However, a great deal of information can be associated with a moderate number of bits properly organized relative to some specific coding system.

When considering the intelligence of a system, one must take a synergistic approach because intelligence is a property that arises from the organization within the system. That is, one would not predict the intelligent behavior of the brain or of a device from a study only of the properties of its elements.

We consider, therefore, a whole system interacting with its environment.

The momentary physical condition of a system we will call its "state." Of the interaction between a system and its environment, the "input" will refer to environmental action on the system and the "output" will refer to the system's response which is determined by the input and the state taken together.

If a system has a certain chemical makeup and a certain external physical makeup, appropriate organization within the system may bring about responses to the environment which would cause certain types of events to occur more often in the environment than random statistics would predict. Such types of events we will call the "goals" of the system.

A system that achieves goals by taking into account one or more properties of the environment is said to "model" its environment. An environmental model or "world model" is therefore an internal organization that corresponds to one or more external properties. A <u>comprehensive</u> model takes into account many environmental properties. A successful model takes into account the most important properties for achievement of the system's goals.

The correspondence between an internal condition representing soma property and that external property we will call the "code." The external property is the "information" that the internal condition represents.

An advanced system may manipulate its environment so as to also store information externally as, for instance, a human being does by setting down ideas in writing.

Now, all physical systems including those at $0^{\circ}K$ are active to some degree. Even the most passive system modifies highly active processes impinging upon it. Although we do not speak of inactive physical systems, we may speak of inactive information associated with physical systems.

Information is "active" when it is involved in output-input processes and "inactive" when associated with a quiescent macroscopic state of a system.

Information associated with the input of a certain device will tend to control the responses of that device. Thus, active information represents control over active physical processes.

"<u>Intelligence</u>" might be considered as the organized control of the active processes of a system which brings about goal achievement by taking into account one or more properties of the environment. Higher intelligence would be associated with systems having more sophisticated world models and diverse abstract goals. To the extent that stored inactive information will later be used to control active devices, this information represents a latent addition to the intelligence of a system.

Fundamentally, "information" or "meaning" involves correspondence or relationships between things. Within any broad language, there is no absolute foundation or set of elements in terms of meaning. In all of our logic and definitions there is an inherent circularity (even in set theory) such that any set of elements chosen as a working basis in terms of which all other things are defined may still, in fact, be definable in terms of other things not included in the elementary set.

There may, however, be correspondences that associate words within a language with things outside the language. Thus, for any language to relate to the outside world, there must be correspondences associating some of the elements of the language to properties of the outside world. The meaning of the word "red" might be taken as a reference to the sensation within one's mind that one mentally associates with the word "red." This association has been learned by means other than definition within the English language. Furthermore, the sensation of red automatically occurs in the minds of those who are not color blind whenever any of a certain set of physical conditions occurs. Through the coding structure of our bodies, the sensation of red represents the presence of one or more of these physical conditions.

There are several major aspects associated with the information contained and used by an active system:

- the scope of the information contained by the system; i.e., the number and kinds of physical properties represented,
- the scope of the interaction among the diverse kinds of information; i.e., the amount and kinds of different information actually caused to interact,
- the scope of informational control; i.e., the number and kinds of physical properties that can be controlled,
- the completeness or directness with which control over specific physical properties can be established:
    1. properties that the system can control directly,
    2. properties that can be controlled indirectly,
    3. properties that cannot be controlled.

There are several levels of goals, one or more of which may be inherent in the organization of a system:

- the compulsion to carry out one or more algorithms,
- the compulsion to iterate a specific procedure until a certain result is achieved,
- the compulsion to try various of a set of known procedures until a certain result is achieved,
- the compulsion to seek procedures for obtaining some specific result more quickly or more efficiently than is possible using known procedures.

Self-motivation is established in a system if the order in which various subgoals are pursued is partly established by the system itself on the basis of estimates of time required to solve each subproblem and the importance of the solution for achieving major goals.

The existence of natural intelligence and the fact that some of it can be directly or indirectly related to the nature of the organism's physical construction and operation lend confidence to the idea that artificial devices capable of intelligent behavior can be developed.

Many properties that were formerly thought to separate man intrinsically from other beings have since been found to some extent in other forms. A most important piece of evidence offered was that only man was capable of language using involved semantics. Recent work with a chimpanzee has shown that this is not the case.[5] Metal-backed plastic cutouts of different shapes and colors were used as words to represent different objects. Some cutouts represented names of people, colors, and things; others prepositions, verbs, etc. The chimpanzee using the cutouts seemed to show reasonable facility in the understanding and construction of language.

One area that has been examined in terms of the informational logic behind its construction is the eye.[6,7] In some species, information is processed within the eye to such a remarkable extent that the retina has an integrative capacity rivaling that of the primate's cortex.

The frog, which is primarily interested in catching flying insects, has in its retina convex-edge detectors, boundary detectors, moving or changing contrast detectors, and dimming detectors that respond to small moving objects.

Some mammals such as rabbits and squirrels have high-level retinal responses. Others, such as the primates and the cats, have high-level visual responses only in the cortex.

The results of visual processing with respect to motion are fairly evident to man even at the conscious level. One may look at a relatively still scene and not notice some animal that is in view. If that animal suddenly moves, one's attention may immediately be drawn to it.

The type of informational processing just discussed is built into the original structure of the organism rather than being obtained by learning. Thus, a moderately high level of informational processing capability is initially programmed into complex living organisms. Fairly complex behavior on the part of living creatures which appears to be preprogramed we call "instinct."

Insects appear to develop and to behave almost entirely in accordance with built-in responses. Mammals, however, have developed the capacity for modifying their informational organization to such an extent that learning is fundamental to their development and behavior. Even in the human brain, the most capable learning system yet discovered, there is extensive built-in structure.[8]

We can think of the human brain as made up of three main regions: the brain stem with its reticular formation, the rear parts of the cortex, and the frontal lobes.

The brain stem regulates the energy level and tone of the cortex, providing a stable basis for the organization of its various processes. The reticular formation is particularly important. It controls wakefulness and gives the brain the ability to discriminate between important and unimportant stimuli.

The rear parts of the cortex are primarily concerned with analysis, coding, and storage of information. The various areas of this region have highly specific assignments. Different areas are responsible for the analysis of optic, acoustic, cutaneous, and kinesthetic stimuli. Each area has a hierarchical organization: a primary zone sorts and records sensory information, a secondary zone organizes the information further, and finally, a tertiary zone, where the data from different sources overlap, combines all of the data to lay the foundation for the organization of behavior.

The frontal lobes deal with the formation of intentions and programs for behavior. They perform no sensory or motor functions. Sensation, movement, perception, speech, and similar processes remain entirely unimpaired even after severe injury to these lobes. Intimately associated with the brain stem, including its reticular formation, the frontal lobes serve to activate the brain, regulating attention and concentration. Complex functions such as speech and writing involve many separate parts of the brain working together.

III. WORLD-MODEL HIERARCHIES

The world model of a complex biological system can be separated into several levels or hierarchies. The cells of an animal's body form subsystems that are able to generate certain codes of response. Some of this response takes the form of nerve impulses that are sent on to the brain.

If we consider speech, the first level of response is the separation of the sound reaching the ear into various component frequencies owing to the nature of the ear's physical construction. The vibrations of the individual fibers in the ear represent a low code level in the hierarchy. The next level involves electrical responses transmitted to the brain to indicate which fibers are vibrating. In the brain, the electrical responses received are compared against a code each element of which represents a model or image of important sets of sounds combined in time. Further processing involves sorting sounds into important categories, groups of sound categories into words, groups of words into phrases, and groups of phrases into the overall meaning of the speech. At each higher level, information becomes more general and more abstract and occupies less space in memory.

The value of the hierarchy is that it enables important properties of the environment which are obscure at low coding levels to be constantly abstracted so that their storage-capacity requirement is minimized while their importance is emphasized.

A high-level intelligence will incorporate a learning capability. Such an intelligence continually improves its world model by generating new hierarchies of information, coding, modeling, and response.

A hierarchy is already present in the hardware of digital computer systems. Control units are considered to occupy a higher level than logic, arithmetic, or memory units.

Information stored in memory is usually broken down into two categories: program and data. The program acts primarily to direct the control units. These units then determine the logic or arithmetic devices and the memory-storage locations to be used. The data act as directions to the logic and arithmetic devices, controlling what their output will be. However, data also may direct the control unit through conditional jumps that depend upon the nature of the data. The program is usually considered to be higher in the hierarchy than the data.

Such a simple hierarchy approach to the memory is satisfactory for conventional computer programs. However, an artificial-intelligence program uses some techniques not often employed by conventional programs. A moderately comprehensive artificial-intelligence program must continually change its own program, reallocate storage, use subroutines that are capable of using themselves, generate new codes for new hierarchies of information, decide the value of pursuing a given line of investigation on the basis of the problem's importance and the time it is likely to require, use a good strategy to improve speed of discovery, generate new hypotheses based upon probability analysis to give the capacity for inductive reasoning, and pursue certain broad goals placed in the program as an overall control policy. The program should cause the history of the most important decisions it made and the data pertinent to each such decision to be recorded. When the interaction of the program and the environment changes the environment, the program, to be effective, must develop a model of itself to predict the future behavior of the environment. This self-modeling then becomes a sort of "self awareness." The historical record keeping gives the program a "memory" of its past. The tendency to satisfy broad program goals provides some similarity to our internal drives or goals.

As a result, in artificial-intelligence programing, memory categories are not clearcut. The computer may spend some time operating on certain information as though it were data. Once the resulting information is properly organized, it may be used as program rather than as data. The artificial-intelligence program will involve further program hierarchies in addition to those of the computer hardware. Each hierarchy may have its own code for both programs and data.

The overall program must be able to optimize the type of code used for various hierarchies as further information becomes available. A code should be so designed that commonly used types of information take up the least memory capacity per unit of stored information, whereas rarer types are allowed more capacity per unit of information. For instance, one finds a and e more commonly used than x or y in the English language. This fact should be considered in optimally storing English information.

When certain types of programs are commonly used in many larger programs, the basic programs become subroutines for other programs. A reference or program call to this subroutine becomes a higher-level program code than the code of the subroutine itself. The higher-level program codes can be optimized, depending on one's choice of programs to be used as basic subroutines. This capability for program-code optimization must also be programed into the artificial intelligence. Optimization depends upon the relative frequency with which certain kinds of programs are used and thus depends upon the nature of the environment. Time is the competitor of memory in terms of optimization. One should try to strike a reasonable balance here.

A spectrum of levels from the more abstract down to those dealing directly with the environment is necessary for "world model" behavior. As an example, programs for computer translation of one language to another are not nearly so effective as human translation because there is no world model concept of the meaning of the words in the computer program. In a high-level world model, there are many diverse properties and experiences associated with any given idea.

In a computer, one would expect to have a great deal of cross-referencing among related sets of data to allow for high levels of association. Also required would be programing that allowed the intelligence to construct at various levels the properties associated with a given idea. This allows for a sort of internal visualization to aid in thinking.

A hierarchy can be much more flexible in a very large computer than in the brain. An artificial intelligence can be broken up into individually intelligent subsystems each of which attacks a different problem. Hardware and programing can be supplied to each subsystem as required by its individual problem. A higher level would then act to coordinate the overall system.

IV. CONSCIOUSNESS

So far we have dealt with the intelligent behavior associated with systems. The question of whether they are conscious or not is another matter. Conscious existence is often conceived as a quality of the soul or as the soul's nature and being. Each person can tell whether he himself is conscious, but cannot directly test whether others are conscious or not. Because other's external reactions are similar to his, and because their structural makeup is similar, each person usually assumes that other people also have this quality. Because we can never know whether a being separate from ourselves is conscious or not, all we can do is rely on external evidence to help direct our beliefs on this matter.

Now, in turning our interest to consciousness we are more interested in the meaning associated with physical processes than with the processes themselves. Physical systems simply provide the physical means for information interaction. The active interaction of information which brings about the achievement of complex goals by taking into account a substantial number of important environmental properties is associated with high intelligence. We might, therefore, make the following assumptions:

- The intensity of consciousness is the amount of activity associated with the system's interacting information.
- The intelligence level of consciousness is the extent of world-model levels and diversity of goals and properties associated with the system's interacting information.

The consciousness of a system taken as a whole is therefore assumed to be greater than that of the individual parts.

Now we are in a position to discuss the conscious versus the subconscious aspects of the human brain. The conscious mind is a large, high-intelligence-level subsystem of the brain which is fairly active when the brain is awake. Let us label the conscious mind A, and a small, low-intelligence-level subsystem that fulfills the duties of an automatic subroutine B. A may decide to send an activation command (a narrow amount and quality of information) to B. B then proceeds with its usual task without continually returning information to A. Because A is not involved in interacting strongly with information received from B, A has little "awareness of B and its activities except indirectly through observations of the results of the subroutine's actions.

In the brain there are many regions of activity which are not involved in a high rate of informational interaction with the highly intelligent and active subsystem we call the conscious mind. As a result, the conscious mind has little awareness of these regions which we classify together as the subconscious mind.

Another aspect of the brain even more clearly emphasizes the fact that active information interaction is the key to consciousness or awareness. The human brain is actually a double organ consisting of right and left hemispheres connected by a narrow region of nerve tissue called the corpus callosum. If the connection between the two halves is cut, each hemisphere functions independently as if it were a complete brain.[9] Each half appears to be capable of advanced mental functions and to have its own sphere of consciousness. Each half is unaware of what the other half is thinking. Brain-bisected humans are able to carry out two tasks as fast as a normal person can do one, and a split-brained monkey has been shown capable of dealing with nearly twice as much information as a normal animal.

When active information interaction between two brain halves is stopped, they cease to act as a unified conscious system and begin to behave as two separately conscious systems.

A "unified consciousness" is therefore to be associated with a set of active elements of information which are capable of much interaction. A unified consciousness is associated not only with a given system, but also with all of the other systems with which it actively exchanges information.

On the basis of the theory just given, a family, a nation, or the human race as a whole each has some consciousness.

The rate of informational interaction among humans is, of course, much lower than the rate of informational interaction within one human brain. Therefore the intensity of consciousness of a group of human beings is much lower than that of an individual. However, the intelligence level of the group is equal to or greater than that of the individuals.

Just one example of interaction among humans which leads to greater intelligence is the following. Consider two people trying to remember a third person's name. One person might finally remark that he thinks a person's name begins with a Z. This may suddenly cause the other person to exclaim "Ezra! His name is Ezra Wilson!" Without the somewhat distorted cue of the first person, the second might never have recalled the nama. Certainly it would have taken him more time. Human interaction often speeds progress both in the field of ideas and in the field of events.

Thus, we see that the human race represents a super-intelligent system with a rather low intensity of consciousness. Its basic unit of activity is the individual human being. Its active memory is the total memory of the race, and its latent memory is the history of the race. Its great capability is a result of the organization of human society.

Let us now return to our discussion of the human brain. Information enters the brain simultaneously through many channels. For expeditious activity of the brain, the many motor programs should not interfere with one another in the process of their execution. In the brain, two simultaneous activities may not be able to proceed without some loss in the effectiveness of each. This would be particularly true of two or more complex activities each of which required the use of the same area of the brain for part of their processing. The system must also be prepared for an immediate change of the operating programs because of new, more powerful stimuli entering from other channels.

In a very complex system, there must be a program for switching among the various channels of sensed information and motor programs. This function is associated with consciousness in man.[10] This particular aspect of the brain's consciousness might be called its attention control.

The attention control increases the activity of one of many channels along which information may enter in any given moment. This amplification is probably most directly controlled by the reticular formation. Amplification of one channel cannot generally be maintained for an extensive period. The mechanism appears to be such that the amplifying system rapidly gathers "strength" and can then return to the former channel. This method of operation guarantees rapid evaluation of the significance of new information entering the system so that switching to a different activity program may occur because of the new evaluation.

There are several types of brain memory available to the conscious and subconscious minds. Properties from events are usually stored in a "short term" memory. Either a great significance attached to some property or a great many repetitions are required to transfer the property to "long term" memory.

A single event is stored in conscious memory only if it is so significant that some of its properties reach the highest levels and are there considered important. In general, conscious mamory is filled by the higher levels and subconscious memory, by the lower levels.

Higher-level learning involves the relationship between the conscious and the subconscious mind in humans. Consider a musician starting to play a new piece of music. At first he consciously devotes his attention to each new series of notes as they appear on the sheet before him. If he intends to memorize the music, he plays it over and over. Eventually, the effort of playing the notes requires less and less of his conscious attention until the time comas when he can think about other things, such as going fishing, while his fingers seem to play the music by themselves. This frees his highest levels of thinking. He can now devote his attention, if he wishes, to carefully adding nuances to the rendition which make the piece sound better without his mind's being impeded by all of the complications involved in trying to consciously think which notes are coming next.

In the process just discussed, the overall resources of the brain are used to develop a routine. Once this routine is developed, it must be repeated over and over again because this is a primary method that the brain uses to place information in long-term memory. Once recorded, this routine becomes a subroutine that the mind can set into motion when it wishes. The subroutine is then a useful additional tool available to the conscious mind. Once the subroutine has been set into motion, the conscious mind is free to pursue other activities simultaneously with the parallel subconscious running of the subroutine. This subroutine may have several entry points, but it is not stored as random access information. If the musician stops part way through the piece he will not usually be able to start again at the exact point where he left off. Instead he will return to an earlier point in the piece which he remembers how to start and will begin from that point. He will then be able to pass through the point where he had previously stopped as the subconscious subroutine automatically proceeds.

The subconscious thus represents the operation of various lower-level thinking processes simultaneously in parallel with upper-level thinking processes. The separation between the two prevents the upper levels from being overly cluttered by the lower levels. The upper levels are left much more time to evaluate abstract ideas free from the automatic processes that have been developed in the past to deal with the vast number of everyday tasks.

Unfortunately for us, the human brain has made insufficient provision for bringing information discovered at the subconscious levels up to the conscious level when requested. To understand this, note that one can immediately recognize a friend when he enters the room without being able to recall the details of his features from conscious memory. The subconscious memory has a catalogue of important aspects of his features which allows the subconscious mind to "recognize" him. The code specifying the friend is sent to higher levels without the details originally used for his identification. There are many specific differences between the facial features of men and women which allow one to know that a given face is masculine or feminine. Yet, consciously, one may have only the vaguest idea about these differences. Most people cannot recognize the absolute pitch of a musical tone. Yet, at lower levels, this information is present but not passed on to the conscious mind.

Artificial intelligence need not be nearly so limited in its awareness as is the human mind. Every part of its memory and active hardware can be made available to its highest-level programs for examination or use. As a result, any information can, in theory, interact with any other information if the need arises.

Various subroutines may be used by the intelligence as automatic subconscious subroutines most of the time. However, if the intelligence decides to examine any one of its subroutines so that it may be improved, the intelligence can simply make a copy of the subroutine elsewhere in memory, examine and modify it, then replace the old by the new version. Further, it can examine any subroutine's operations at any point using careful procedures so that the observation does not radically interfere with the subroutine's operation.

Parallel hardware systems can be so designed that the operation of one does not significantly interfere with the sumultaneous operation of another. Furthermore, if two different complex processes require use of the same subroutine, copies can be sent to the relevant hardware associated with each. Such duplication prevents logical interference between many types of complex processes.

More memory, more active hardware, and greater speed may ultimately give solid-state artificial intelligence even further advantages in awareness or consciousness. The overall control in the digital devices will probably be somewhat different from that in the brain. The brain's control has more of an analogue character. Rather than increasing or decreasing the level of activity associated with any given problem, the artificial intelligence will, in most cases, probably either continue a given process or avenue of investigation if the result seems worthwhile in the light of other demands, or it will discontinue the particular activity. For a very broad important problem, however, the intelligence may assign more or less hardware, program, and memory as solution of the problem seems to require.

For the normal continuous stream of input data, lower-level programs will generally act as screens. Only information deemed important at that level for further evaluation will normally be sent to a higher level. Any request sent down from a higher level will be filled as soon as feasible, however.

In computers, we are not restricted to a hardware long-term and short-term memory that corresponds to biological systems. We may allow the program to decide how long a given type of information should be stored. The information may then be stored without a great deal of repetition as is the case in man. When the time is up, the information can be again considered for its value and then erased if further storage does not seem sufficiently important. The mechanics of the human brain require conscious high-level repetition to store obscure facts of little emotional content. If the fact is deemed important, a computer can store it readily. "Forgetting" becomes a much more controllable act in a computer. It amounts to controlled erasing.

Note that consciousness has not in this discussion been associated directly with the learning capability. Rather, consciousness depends upon an active world model associated with a given system.

The learning capability is a tool that enables a given system to improve the level of its own world model. Some systems, such as the insects have obtained their world models primarily through biological evolution. This involves slow development over many lifetimes. There is reason to believe that even insects may have significant consciousness associated with their mental processes although they do not appear to have much capacity for learning

V. LEARNING

There are many categories and levels of learning. Here we shall discuss a few aspects.

Direct programming of a computer is an example of a low-level learning process because it involves giving the computer new perfectly organized information that can be used to solve new problems. Information may be acquired directly as just discussed, or indirectly through statistical analysis of somewhat random input information. If a statistical analysis is required, the learning process is considered somewhat more advanced than learning involving only direct input of well-ordered and accurate information. A statistical analysis also enables the system to evaluate input information for "errors" on the basis of relative frequency of occurrence of various types of information in the past. Furthermore, many properties are statistical in nature, and some of them can be observed only by applying statistical analysis to input information. Direct learning requires a very accurate teacher to ensure that all input information is accurate, orderly, and meaningful to the learner. Indirect or statistical learning does not always require the aid of a teacher. Statistical analysis is also the basis for inductive reasoning and often for judging the relative importance of various things.

Recognizing that learning may either be direct or involve a statistical interface, one scheme that might be used to partition learning into several categories is the following:

(1) Learning old facts, procedures, and theories by accepting a specified and perhaps ordered collection of known properties or procedures as a group along with a label assigned to the group.

(a) New definitions increase language capability;

(b) New algorithm complexes increase procedural capability;

(c) New events increase historical knowledge.

(2) Inductive (or creative) discovery of new knowledge without the intermediary aid of a formal or informal teacher, or the sole aid of formal deduction from past well-known information.

(3) The conversion of inductively learned knowledge into a formalism.

(a) One example is an heuristic approach to develop a deductive proof leading from previous knowledge to the inductively newly discovered knowledge.

(b) New knowledge that logically extends beyond previously known facts and theories may be formalized by discovering the simplest, smallest number of new definitions and procedures and determining the largest number of characteristics in common with old knowledge and procedures which may be used as a basis for encompassing the new knowledge.

(c) Models based upon complexes of previously known information can be developed to yield properties in agreement with the new information. This is one method for determining characteristics in common between the old and new information.

We will now discuss some details relevant to machine learning.

"Algorithm" is used to mean any procedure that can be carried out by a mechanical device according to fixed rules without requiring human intervention to supply intelligent initiative or decision at some point.

Computers are thus machanisms that carry out mechanically, step by step, the procedures or algorithms programmed into them. The actual carrying out of a procedure represented by an algorithm is not what we normally mean by reasoning deductively, although one can deduce from the circuitry, the last state, and the program what physical state should occur next.

Deductive logic involves the grouping together of "things" into sets by specifying the properties to define each set. If one statement indicates that a given set is empty and another statement indicates that the same set is nonempty, the two statements are said to conflict logically with each other. "Premises" are given indicating that some sets are either empty or nonempty. Any statement that does not conflict with the premises and adds no further information about the sets is deductively in agreement with the premises. If a statement gives information about the sets different from that given by the premises and does not conflict with the premises, then it is not deductively related to the premises. "Validity" is a relationship between the premises and a conclusion. "Truth" is a relationship between the statements and the real world outside the argument.

Computers are readily programed to solve limited problems in deductive logic more quickly than a human being can solve them. Nevertheless, solutions to problems involving a moderate number of initial premises or axioms may prove difficult, as is seen from the fact that there can be 2^{2^n} subsets of a set having n specified classes. One then reaches the point where it may be useful to turn to heuristic methods to decide which type of possible deductive path might be most likely to unite the conclusion or theorem with the premises or axioms.

"Heuristic methods" are strategies that do not always succeed, but often lead to short cuts. These strategies are suggested by the success of similar maneuvers in problems attacked previously. One tries to develop criteria from past successes for judging which of many possible avenues will most likely lead to a reasonably quick useful result. Thus, heuristic methods are determined by inductive reasoning. Inductive logic is in some senses a much broader problem than that presented by deductive logic.

Probability lies at the heart of inductive logic. If certain properties associated with a series of events have a high probability correlation, then events having these properties may be grouped together and classified with a label that is called the "name" of that class of events. The discovery of high probability correlations is the first level of inductive logic. The second level involves basing one's behavior on the assumption that future events will continue to show the same probability for the property correlation under consideration. A third level involves the discovery of a relatively simple model that yields about the same probability expectation for the specific property correlation. If this level can be attained, the scientific method becomes involved. The fourth level involves deducing some other property of the model which might yield a probability expectation to be tested against future events. One then modifies one's behavior so as to test this new property. If the prediction agrees well with the environmental behavior examined, one begins to rely more heavily on the model in future planning.

We should note that the word "probability" has several different meanings. Probability can be regarded as:

1. the relative frequency of occurrence of an event in a large number of trials;
2. a result that can be deduced from the more primitive notion of equal likelihood;
3. the degree of certainty of the observer.

In connection with definition one, it is important that the data examined tend toward a limit as the number of events considered is increased. Only for such a condition is probability theory very informative (See the Appendix).

The first level of inductive reasoning involves the first definition of probability; the fourth level involves definition No. 2 in the deductive process, definition No. 1 for the testing, and definition No. 3 in connection with one's reliance upon the model in future planning and reasoning.

The use of models (theories) to discover more about reality is a heuristic method. It enables one to discover further properties of the world more rapidly without spending such excessive time in the random trial and error technique as is often encountered at the first level of inductive reasoning.

We note that the first and second levels have been used most extensively in the past for learning. Deductive reasoning from models is less often encountered. Only recently has systematic use of the scientific method come into such general use.

If one is faced with a very large number of possibilities to examine in either inductive or deductive logic problems, he will have to decide which possibilities to investigate first. One then applies inductive reasoning to past methods tried for solving similar problems. Those that were most successful are modified to fit the present situation and are used as a basis for reaching a decision. Thus, decision making should reach into the highest realms of logical reasoning. It must be admitted, however, that not all human decisions have a good logical foundation.

Because inductive reasoning is fundamental to learning, we will first direct ourselves to this problem. Primarily, all of the logic involved concerns discovery and use of properties. Probability correlations are made among certain properties. If a high correlation is obtained, a new, important, higher-level property may have been found. Behavior may then be modified taking advantage of this new discovery. Probabilities represent magnitudes that should be examined by data-analyzing subroutines. If various properties can be ordered dimensionally on the basis of their probabilities, then the dimensional magnitudes can also be examined by data-analyzing subroutines. These subroutines are pattern-recognition routines that attempt to discover different characteristics or properties of dimensionalized data. The resulting properties are of a higher level than the original dimensionally ordered set.

If several different properties tend to be strongly associated with each other, the computer may assign a label to that set. That set may then be treated as an object in later evaluations. It may be discovered that other types of objects have a few properties in common with the given object. A label may then be assigned to the set of properties common to the various objects, thus defining a class. Here "class" actually has essentially the same meaning as "object." An object is assumed to occupy a lower level of abstraction than a class, however.

Here we already see the need for incorporating deductive logic into the programming. The computer must be capable of adhering to the rules of set theory when it constructs a hierarchy of important sets of properties.

Depending upon the nature of a property, a 0 or a 1 may be stored to represent the momentary absence or presence of that property, or several bits may be used to store a magnitude associated with that property. The actions of subroutines themselves are also properties. Anything may be represented by a label and operated upon as an element of a set. Labeling and appropriate cross-referencing thus become an important part of the overall program task.

Data-analyzing subroutines applied to n-dimensional data would be capable of discovering the nature of any m-dimensional boundaries where $m < n$, as well as the magnitudes of the measures associated with the various bounded spaces. Derivatives, periodicities, similarities through rotation or expansion, discontinuities, roots, or any other commonly employed properties would be made available in the form of data-analyzing subroutines.

The order in which the subroutines were applied to the data would depend upon the data category; the number of subroutines applied would depend upon the importance of the data as determined by higher levels in the artificial-intelligence program.

Very important management items to be programed early into any artificial-intelligence program are memory transfer, storage, and cross-referencing. Those data most likely to be used next should be placed in the fast memory, and other data should be properly assigned to the other types of memory on the same basis. The program should make data shifting and control fairly easy. When new information must be inserted into an already existing list, time is saved by simply removing a word, inserting a jump instruction to an empty memory segment, storing the removed work in the first empty storage location, adding the further information, then putting in a jump back to the insert position in the old list. When too many of these loops have been introduced and clutter the memory with jumps that take time to execute, the memory can be given a major readjustment to remove them. Because any number of words in memory can reference back to some specific location, logical cross-referencing can be made rich without the problems associated with physical connections to a single point.

Behavior-modification procedures must be present in the initial program. Various subroutines are written which can discover through probability correlation the various classes of properties which we have discovered to be important to intelligent understanding. A complementary set of subroutines may also be written. Each member of the complementary set is associated with at least one of the subroutines that discover properties. The complementary routine is based on the class of property discovered, and automatically adds to the computer new instructions that would be logically required by the discovery of another member of this property class. This, then, is a basis for advanced machine learning.

The artificial-intelligence program may either be given an artificial environment represented by another program that interacts with it or it may be allowed to interact with our real environment during various parts of its learning process. The intelligence must be allowed to have some output that affects the environment in order to learn how to exert some control over its environment. Only as it is able to observe its own ability to affect the environment will it develop a concept of itself as an important factor. Intelligence strives primarily to control with the aid of observation and discovery. The last two properties are not usually the primary aim.

Ultimately, the environment must contain either artificial or natural intelligences that cause the key artificial intelligence to learn that it is not alone in having the capacity to think. Just as it uses symbolism or code in its modeling and thought processes, the intelligence must be capable of readily discovering the use of external codes called language for communication with other intelligences. Once this step has been achieved, man will be in a much better position to communicate with and teach the intelligence.

Hopefully, in the reasonably near future, an overall learning program can be developed which can be taught a view of the three spatial dimensions and one time dimension of our world and the concept of interacting with us through the medium of natural language. After trying to develop in it concepts of how to manipulate objects in the physical world to achieve goals, it would be useful to start teaching the program abstract concepts that we find useful in dealing with the physical world. This is a process much like teaching a human. The real goal is to develop a program that discovers and learns faster and remembers more.

Finally, if a high level of thought is obtained, the artificial intelligence should be taught about the basic theory behind its program and be allowed to examine all but the goal segments of its own program. This allows the total program to focus its faculties on small segments that it perhaps can modify and improve as it learns more about its own operation. Once this level has been obtained, the program may be able to take over and improve itself faster than we can improve it. This is a major goal, at any rate. This possibility points out the importance of not spending excessive human effort in over-developing any given area at the expense of early overall artificial-intelligence development. Once sufficiently advanced, the intelligence might advance all of the areas much faster and more thoroughly than we could develop any given area.

Learning in the brain presumably also involves the discovery of high probability correlations. This process appears, however, to be much more closely related to the memory mechanism than is required in computers.

In the brain, we may have around 10^{10} nerve cells called neurons. The neuron's output is conducted along a fiber called its axon which splits up into many branches that contact other neurons through junctions called synapses. There may be several hundred connections to a given neuron.

When one neuron fires, an impulse reaches following neurons. If the total impulse received by a given neuron from other neurons exceeds a certain threshold value, the neuron fires. Because a neuron is either firing or is not, a significant part of the brain's operations is based upon an "on-off" system.

Repeated use can change the magnitude of the effect upon the following neuron of an impulse crossing a given synapse. Thresholds can also undergo change. Frequent use of such mechanisms may make it easier to reestablish patterns of impulses which involve certain synapses. This would cause some memory to be associated with frequency and total amount of use.

The number of times a given pattern is fired or illuminated depends upon the probability that the event causing that pattern will occur. If one neuron is connected to many of the other neurons associated with a given pattern, the firing of the pattern causes that neuron to fire. The single neuron then becomes a higher-level representative or code symbol for that pattern. A high probability for simultaneous firing of different neurons at the higher code levels may improve the weights of the synapses connecting them.

Thus, in the brain it seems very likely that probability analysis and memorization are closely intertwined for the long-term memory. This makes lower-level inductive reasoning or learning based on probability an inherent part of the brain's operation.

VI. COMPUTER LEARNING EXAMPLE

This outline example is given not because it is highly practical, but because it illustrates some things that can be used in lower-level learning.

We shall consider presenting a certain environment to an artificial-intelligence program that has no initial knowledge of the nature of its environment. Various important properties of the environment will each be represented by a one-bit location in the input storage of the artificial-intelligence program. The order of these storage locations has no particular meaning relative to the physical characteristics of the outside world. The increased or decreased "satisfaction" of the intelligence will be made to coincide with the occurrence of certain moderately abstract events in the environment. Certain relatively abstract output from the intelligence can control its environment to some degree so that an indirect control of its satisfaction can be acquired.

The intelligence program's problem is to discover as many important properties of the outside world represented by the input data as it can, in order to discover a way to consistently increase its satisfaction through appropriate control over the environment.

We shall now specify more details of the problem. Two-dimensional visual pictures and one-dimensional aural frequencies will be coded into randomly mixed binary bits before being read into the intelligence program. Pictures of an A or a B with constantly changing two-dimensional orientations and sizes will be used. One aural frequency will be associated with an A, and another with a B. Just before an A picture is shown, an A aural frequency will occur. Before a B is shown, its aural frequency will be sounded. Whenever an A or a B frequency is not being sounded by an external environmental program, the output from the intelligence will be randomly mixed and fed back into it as aural input. The goal given the computer will be to increase the numerical value in a certain memory location. The external program will change this value at random times. If a visual A is being shown, the satisfaction value will be increased; if a visual B is being shown, the value will be decreased.

The information conversions will be done by the external environmental program. It will represent the picture first as a square-ordered array of 0's and 1's corresponding to white and black spots. The initial set of aural input information might be a string of 0's, a 1, and a repetition for several cycles. With each new input, the 1's could be shifted to the left 1 position. The whole would be subject to a periodic boundary condition that would generate a new 1 at the right as a 1 disappeared at the left. The sound frequency could be changed by changing the spacing between 1's.

The bits from these two ordered binary arrays are then randomly mixed together. The mapping of the old order to the new is retained so that all subsequent information is scrambled in the sama way. The original random mapping between the output and aural input is also retained throughout the experiment. Thus, a given input or output storage location within the artificial-intelligence program will always correspond with the same property in the environment.

The final duty of the external program is to show pictures of a visual A whenever the output from the intelligence program is such that it maps over to the aural input frequency for A, and to show a B whenever the program reads out the code for a B frequency.

It must be emphasized that the true input to the intelligence program is a hybrid obtained by mixing the aural and visual inputs together.

We note that by starting with a square visual array we have already simplified the problem relative to similar ones found in nature. The more general problem can be handled, however, using a more comprehensive program. Our simple solution here will not be entirely adequate for more random arrangements of the visual receptors.

The first step taken by the intelligence should be the orderly application of various probability-analysis routines to the data to discover important properties. An obvious check would be to find the average number of 0's and 1's in each storage location. Such a check would permit immediate sorting of the data into two groups corresponding with the visual and aural groups.

To understand the basis for sorting and ordering dimensional data, consider the following example actually run on a computer. The information analyzed was a visual square array with 11 bits on a side. Ten thousand pictures of a rectangle that was allowed translational and rotational movement off the screen and back on were read in to a probability-analysis subroutine. Of this 121-position array, we shall look at results for the 13 positions below:

		(6, 8)		
	(5, 7)	(6, 7)	(7, 7)	
(4, 6)	(5, 6)	(6, 6)	(7, 6)	(8, 6)
	(5, 5)	(6, 5)	(7, 5)	
		(6, 4)		

These 13 positions represent the original picture order before scrambling. The program first determined the percentage of times that positions (x, y) and (6, 6) gave the same value. The results were:

		0.84		
	0.88	0.92	0.88	
0.83	0.91	1.00	0.92	0.84
	0.88	0.92	0.88	
		0.84		

We see that the probabilities immediately broke up into clusters or groups having probabilities of around 1.00, 0.92, 0.88, and 0.84. Excluding point (6, 6), the nearest neighbors of this point fall into the highest probability group. The next-nearest neighbors fall into the next group. Because group separations are most distinct for those nearest to the reference point, it becomes necessary to take other reference points for similar correlations.

If the data had been randomized, the program would have been able to determine the nearest neighbors by this method and to separate the visual data from other types.

The next probability test determined the percentage of times that positions (x, y), (5, 6), and (6, 6) gave the same value. The results were:

		0.79		
	0.86	0.86	0.81	
0.83	0.91	0.91	0.84	0.76
	0.85	0.86	0.81	
		0.79		

Of the nearest neighbors, position (7, 6) has the lowest probability and is thus farthest from the group (5, 6) and (6, 6). This allows the nearest neighbors to be broken up into the two dimensions. Using this method, the whole original order or a simple inversion thereof can be reestablished.

Having obtained a dimensional ordering, the subroutine would call another to introduce new program into one of the program sections regularly examined by central control. This new program simply involves that the computer reorder data and store it. The various calls indicate the location of the data, of the memory for final reordered storage, and of the mapping developed by the probability-analysis subroutine. Thereafter, appearance of new input data would be immediately followed by reordering of the data.

Note that one does not check on all possible pairs, because this could waste considerable time for large arrays. After one reference point has been analyzed, the results give enough information to reduce the next check to including up to only the the second-nearest neighbors of the original point. After a while, another pair check over all points relative to some reference point considerably removed from the first would become important.

Probability-analyzing subroutines like the ones discussed can determine the number of dimensions associated with various types of data. Thus, the visual data would be found to have two dimensions, whereas the aural data would be found to have only one.

The random data represent the lowest level in the intelligence program. The ordered dimensionalized data represent a second level. The program should then begin to investigate properties at the higher level. For instance, one might place on the vision screen a capital letter A, allowing it to translate and rotate in the two dimensions of the screen. Various data-analyzing subroutines should be capable of discovering the boundaries between black and white regions. These subroutines should also be capable of averaging along the boundaries to obtain average first and second derivatives associated with the boundaries.

Note that various mathematical systems or techniques may be logically similar or even equivalent. Nevertheless, one system may emphasize one set of properties and another system may emphasize a different set. Some problems may, therefore, be more easily attacked with one system than with the other. Both systems may, therefore, have to be available in the program.

In the case of second derivatives, we see the need for two approaches. If two dimensions are basically different in nature as, say, temperature and distance, the second derivative d^2T/dX^2 can be quite useful. However, for dimensions that are alike, such as two orthogonal distances, x_1 and x_2, the derivative $d^2\bar{R}/ds^2$, where s represents distance along a curve and $\bar{R}$ represents a vector extending from some origin to that curve, may be more useful than d^2x_1/dx_2^2. One advantage of the vector second derivative is that the magnitude of the resulting vector associated with a given part of an object does not change as that object is rotated. This system will allow constant curvature to be more easily recognized regardless of object orientation. The simplicity of the circle is thus brought out.

The program, upon examining the visual image of the letter A, should store a third level of information indicating the number of line segments with smooth first and second derivatives, the values of these derivatives associated with the line segments, the number of junctions between line segments, and their relative positions along the line segments. It should, furthermore, list whether completely bounded figures are formed. The program should check for various simple types of symmetry that may be present. For the A, a plane of symmetry would be discovered. Rotation of the A until the plane was vertical or horizontal would cause the values of the first derivatives to emphasize this symmetry. If the A were presented much more often vertically, the program should pick out the vertical position as the preferred standard orientation. The vertical A would represent level 4 in the visual hierarchy. If solid bounded regions were definitely of greater length than width, a further technique might be to reduce such regions, to representation by the first and important higher derivatives associated with a line passed lengthwise through the center of each. Such a technique would lead to the logical equivalent of a vertical A formed from three straight lines as a fifth level of representing the original visual input data.

Subroutines to obtain all of the levels just discussed directly are not necessary, because the higher levels could be learned through induction after observing many, many orientations of the letter A. However, direct programming of such obvious devices leads to much quicker attainment of these higher levels. Among other things, we are interested in supplying sufficient techniques for rapid learning.

A sixth level is the application of a code label to the group of properties of visual level 5. Besides analyzing the visual data, the intelligence program would use probability analysis to sort the "aural" information and reorder it into a one-dimansional, ordered aural data set. The periodicity and frequency would then be discovered by data-analyzing subroutines.

Now, if one frequency were often introduced aurally shortly before an A was to be visually presented, probability analysis at the highest hierarchy levels yet developed for the aural and visual channels should discover the correlation between the visual symbols and the aural frequencies.

At this point, the program develops a hierarchical level that is above both the visual and aural channels. This level assigns a code symbol to a group of two things, the highest-level visual code for A and the highest-level aural code for the frequency associated with A. Similarly, a code symbol is assigned to B.

When the program hears the A sound, it has now learned to expect letter A to soon be visible.

Probability analysis should also, with the aid of suitable strategies, be able to readily discover the correlation between the intelligence's input and output. Once this correlation is discovered, the intelligence should carry out the reverse mapping to get the output code for A and B from the aural input code for A and B.

The program should associate a positive satisfaction value with A and a negative satisfaction value with B after probability analysis of the upper-level data causes the correlation between increase in satisfaction and the presence of visual A to be discovered.

One effective learning strategy is for a program to attempt to duplicate in its output high-level properties that it has discovered through its input. Having a positive satisfaction associated with A, the program should try outputing A first. The result should be the immediate appearance of A at the aural input, followed shortly by the appearance of a visual A and a continual increase in the program's numerical satisfaction. This result would be noted by the program which would adjust itself through the appropriate subroutine to continually output A.

If the program had higher levels of programming available initially, it could have deduced more directly that outputing A would probably control its satisfaction. Without higher levels, it must resort to the strategy of simply trying to output all possible high-level properties so far discovered and observing the result. If the effect seems good, a high satisfaction value will be assigned to the given output and it will be used frequently in the future. This lower-level technique is that of learning through mimicking the highest-level behavior of the environment which has been recognized by the intelligence.

More advanced work for the program could involve learning about space as a three-dimensional, rather than two-dimensional, medium and discovering perspective relationships involved in two-dimensional representations of three dimensions.

The programming that an artificial intelligence has at any given time could have been introduced by direct programming, or much of it could have been acquired by learning. The important thing is to develop its overall capacity to learn, because ultimately learning will be a faster way of developing the intelligence than is further direct programming.

Because problems to be solved in developing artificial intelligence will often be broad, it is best to break them up into many small, more easily managed segments.

When a program is first pitted against learning problems, additions will probably have to be made in terms of basic subroutines to handle new general aspects. As more and more problems are attacked by the same program, fewer new basic subroutine additions by an external programer should be required. The goal is to eliminate the need for these additions.

VII. HEURISTICS

In many areas of investigation, one is faced with the fact that investigating all available avenues in searching for a solution to a specific problem may be an endless task. There may be too many avenues available for search.

Starting at one point in mathematics, in scientific research, or even in a chess game, one may be faced with several possible avenues of investigation at that point. Continuing down one avenue may lead to another point where several more choices of direction are available. This type of situation often blossoms into an ever-expanding and branching tree of decisions which for all practical purposes is unlimited.

Let us consider an idealized situation in which each point is assumed to have just two branches. The second level then has four branches, and the n^{th} level has 2^n or $10^{0.3n}$ branches. If one goes through only 20 levels, he is faced with a million possible branches. Usually, problems have many more than two branches at each decision point.

In scientific research, there is an essentially unlimited number of directions that could be taken for further investigation. Only some of these will lead to fairly useful results in a reasonable time.

In the field of mathematics and logic, there are unlimited logical and mathematical systems that could be developed. Again, only some of these will prove of much practical use in the future. Those fields of mathematics which somehow reflect important aspects of the real world around us will be useful. Other systems that might have value in some hypothetical universe may prove useless in the one with which we are faced.

In binary digital devices, Boolean logic is important. For this system, variables are allowed two values, say 0 and 1.

There are two possible Boolean values that have no variables:

f_1	f_2
0	1

The value f_1 we associate with such concepts as "false," "least element," and "empty set." The value f_2 is associated with "true," "greatest element," etc.

There are four possible Boolean functions for a single Boolean variable x:

x	f_1	f_2	f_3	f_4
0	0	1	0	1
1	0	0	1	1

The function f_1 we associate with the concept of "contradiction," f_2 with "not" or "complement," f_3 with the variable itself, and f_4 with "tautology." There are 16 possible Boolean functions for two Boolean variables x and y:

x	y	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
0	0	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0

(x	y)	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
(0	0)	0	1	0	1	0	1	0	1
(0	1)	0	0	1	1	0	0	1	1
(1	0)	0	0	0	0	1	1	1	1
(1	1)	1	1	1	1	1	1	1	1

We associate f_1 with such concepts as "contradiction," "least element," and "empty set;" f_4 with "not x;" f_6 with "not y;" f_7 with "+;" f_9 with "and" or "∩;" f_{10} with "⟷;" f_{11} with "y;" f_{12} with "→;" f_{13} with "x;" f_{14} with "←;" f_{15} with "or" or "∪;" and f_{16} with "tautology," etc. The functions not mentioned can be expressed as combinations of these. For instance, f_2 can be expressed as "not x and not y" or as "not (x or y)."

An interesting fact is that all Boolean functions can be expressed by an appropriate combination of the variables with "not," "or," or "and." All of the logic in the computer can be expressed by appropriate combinations using these three Boolean functions.

Now all subsets of a set with the ordering relation "is a subset of" form a Boolean algebra. Because artificial intelligence must deal with binary variables and with various hierarchies derived from sets of binary variables, one is faced with sets and subsets of "properties" which the intelligence is required to analyze. If we deal with objects that are defined by the presence or absence of various properties, the situation can be expressed by Boolean functions. The whole set of Boolean functions of n variables forms a Boolean algebra.

We have already seen that some Boolean functions are quite valuable to us. The difficulty of investigating the usefulness of all the Boolean functions for n variables is great if n is very large. The total number, N, of Boolean functions for n variables is

$$N = 2^{2^n} .$$

If the intelligence faced with 10 properties wished to make a probability analysis to check the correlations for all of the possible sets of these properties or their absence, there would be

$$2^{2^n} ,$$

or about 10^{300}, different probability correlations that could be investigated. High values for such correlations give a basis for defining a new "object;" that is, labeling an important class of phenomena that may often appear together. For instance, a cow has a certain set of properties that appear together. A dog has a somewhat different set. These particular sets of properties we deem important because they often appear together in our environment, and we label these sets with the terms "cow" or "dog."

We have noted that there are 10^{300} possible probability correlations that might be performed on the 10 properties. This represents a totally unmanageable number of possibilities that could be investigated.

Consider 3×10^{22} atoms/cm^3 and light going 3×10^{10} cm/sec for 3.2×10^7 sec/yr for 2×10^{10} yr (a diameter of 2×10^{28} cm for the universe). The volume of the universe could be assumed to be about 10^{85} cm^3, and it could contain 3×10^{107} atoms at normal Earth densities. We neglect gravitational effects for our hypothetical situation. Even if the universe were packed solid with atoms at normal Earth surface densities and each atom were assumed to represent from one to several bits of information storage, there would not be nearly enough storage for the job. The number of storage locations is smaller than the number of possibilities to be investigated by a factor of around 10^{200} which is an unmanageably large number.

A computer operating at 10^8 operations per second for 30 billion years would manage only 10^{26} operations.

In our real world we have not just 10, but many, many properties, characteristics, etc., to deal with. We thus see that any approach used, not only by ourselves but by the artificial-intelligence program, must involve more efficient methods than an orderly investigation of every possibility present.

All of this discussion points out that "heuristic methods" or "strategies" that do not always succeed but often lead to short cuts must be used.

The overall problem of an intelligence is to build up new internal models of reality that reflect the character of reality better than earlier models did. In this way, the intelligence should be better able to predict what will happen in the future and be better able to influence events so that the future will more nearly satisfy the goals of the intelligence.

In attempting to generate new models, one might use a very general technique that is basically simple and applicable to many different types of situations. Otherwise, one might turn to a large group of specialized techniques, one technique being applied to one specific situation and another technique to another.

In terms of general techniques, an important point to consider is that biological evolution may in some respects be logically equivalent to the development of intelligence. A biological model is played against the environment. A random change is made in the model, and the modification is also allowed to interact with the environment. That model which best reflects the nature of the environment is retained, and a new mutation is tried. The new models may be thought of as new hypotheses that are modifications of earlier ones. The new ones are tested against the environment, and if they better model or reflect the true nature of the environment they survive. This is analogous both to inductive reasoning and to the scientific method.

With the above idea in mind Fogel, Owens, and Walsh have developed a method called simulated evolution.[11] Basically, the method involves starting with a model of several finite states and a specific state diagram. The model predicts the next input symbol. When the next symbol has been introduced, the program checks to see whether the prediction was or was not correct. The program also creates a new model by slightly modifying the original, perhaps by a random procedure based on random-number tables. The predictions of the new model are also checked against later input information. Of the original and modified models, that one is retained which is most successful in its predictions. After eliminating the least successful model, the program modifies the retained version to generate still a new model and again checks the success in prediction. This process is continually repeated as a method of creating better models of the environment.

This technique can handle a large variety of problems, and various heuristic procedures can be introduced to improve the modification technique, thereby reducing the average time required to generate a better model. Furthermore, reward and penalty values for correctly or incorrectly predicting a given type of input symbol can be varied for each kind of input symbol used so that the "goals" of the program can be modified. Penalties can also be imposed for specific types of model modification. For instance, a penalty added for each state of the machine would tend to prevent excessive increase in the number of states as model evolution continued.

The environment presented to the program might be made completely deterministic so that it could be perfectly predicted by a model with enough states. In that instance, penalties imposed for increasing the number of states might cause evolved models to continue to have fewer states than would be required to predict the environment perfectly. Then the models would have less than unit probability of correctly predicting the environment. Thus, penalty for complexity is a parameter that can be adjusted to range from a purely statistical view to a purely deterministic view of such an environment.

The programs used by Fogel et al. were relatively simple and short considering the broad range of problems they could handle successfully. However, the computer tima required to solve a given problem could be much longer than would be required by a program designed for the specific type of problem under consideration. In artificial-intelligence programming, reduction of time required for problem solving is extremely important.

As an example, consider the ordered set of binary symbols 1 0 1 1 0 1 1 1 0 0 1 0 0 0 1 1 0 0 0 0 1 1 1 1 1 1. If the environment were made up simply of repetitions of such a group, then a subroutine written to discover deterministic cyclic behavior could discover this property much more readily than could the simulated evolution program. The procedure would be to shift the first symbol until it matched a similar sumbol in the original sequence. Then a second shifted symbol would be examined to see if a match still remained. If so, similar examination would continue until a discrepancy was found. Once a discrepancy was discovered, the first symbol would be shifted again until a new discrepancy was discovered. Once discrepancy ceased over a number of symbols several times larger than the number of shifts originally used, a short-term cycling would be discovered and noted. A later change in the ordered group used would be recognized by the program as such, and the new group and the extent of its repetition could also be discovered.

Because there are too many ways in which a moderate number of symbols could have order coded into them, only relatively simple types of ordering will be heuristically worth investigating by any finite device. This is true of man as well as of machines. Special subprograms should be developed to seek out rapidly those types of ordering or correlation that past experience has shown to be important.

The task of the main program is to look at a set of data and see how it fits into categories discovered to be important in the past. It would then call upon those specialized subroutines in order of the importance assigned to them in connection with the category under consideration. One by one, the subroutines would each attack the data until important information had been discovered about the data, or until either the time assigned to the problem or the list of special subroutines ran out. This provides a powerful look at the data over a limited time span. If the problem cannot be solved in that tima span by those techniques, it is bypassed. It may be retained in memory until more advanced techniques are learned, or it may simply be erased, leaving only a remark in memory at higher hierarchy levels about the type of problem and its present insolubility.

The evolutionary technique is actually inherent in any development of artificial intelligence. A relatively simple approach will be insufficient as a heuristic method for rapidly solving broad classes of problems, however. The heuristic approach demands extensive specialized programing. One thus makes great gains in average learning speed at the expense of using more memory for the storage of larger, complex programs. It is the very application of a limited specific program to a specific kind of problem which allows the computers to solve narrow problems today so much more rapidly than can a human being.

Some methods for obtaining optimum effectiveness are for the program to improve its own:

(1) information-storage capacity in quality and quantity

(a) by eliminating less-important data from storage entirely

- Primitively, by eliminating data that occur or are used infrequently,
- by eliminating data that have been analyzed, but saving the results of the analysis,
- by eliminating data that do not seem very relevant to the goals,

(b) by optimizing the coding system used for storing data with respect to

- method of representation used
- frequency of occurrence of specific types of data

(2) information-processing speed and quality

(a) by representing the data in storage using a method that reflects many of the relationships among the data so that logical processing after storage of the data is reduced to a minimum,

(b) by storing processed results, or by preprocessing the data and storing the results for table lookup if the results are expected to be used frequently,

(c) by storing the data so that

- frequency of data appearance or use is stressed,
- the importance of data to goal attainment is stressed,

(3) methods of discovery

(a) by analyzing past successes for common characteristics to use as guides for future decisions,

(b) by determining the relative frequencies with which alternate methods successfully deal with a particular kind of problem,

(c) by mimicking or imitating the procedures of educated intelligent beings in the environment to permit rapid transfer of useful algorithms from the educated to the uneducated.

VIII. RELIABILITY

Consider a computer having N identical and independent elements, each of which has two stable states. Let t represent tima, and let the basic time unit be one computer cycle. Let $p(t)$ be the probability that at least one error has been made by a specific element in time t; let $q(t)$ be the probability that no error has occurred during time t; and let $\tau \equiv 1/p(1)$.

We must keep in mind that if a given element commits two errors during time t, the result will return it to its original value as though no error had occurred. If τ is large, say around 10^{18} cycle times or 10^{10} sec for a time of 10 nsec, the relaxation time of a single element would be 300 yr. For $t \ll \tau$, we can neglect the double-error effect.

For a whole group of N independent units passing through one cycle,

$$
\begin{aligned}
Q_N(1) &= q(1)^N \quad , \\
&= [1 - p(1)]^N \quad , \\
&= [1 - \tau^{-1}]^N \quad , \\
&= 1 - \frac{N}{\tau} + \frac{N(N-1)}{2!}\,\frac{1}{\tau^2} \\
&\quad - \frac{N(N-1)(N-2)}{3!}\,\frac{1}{\tau^3} + \ldots \quad .
\end{aligned}
$$

If $\tau \gg 1$ and $\tau^2 \gg N$,

$$Q_N(1) \approx 1 - \frac{N}{\tau} + \frac{1}{2!}\left(\frac{N}{\tau}\right)^2 - \frac{1}{3!}\left(\frac{N}{\tau}\right)^3 + \ldots \quad ,$$

or

$$Q_N(1) \approx e^{-(N/\tau)}$$

For a time interval t, this becomes

$$
\begin{aligned}
Q_N(t) &\approx [Q_N(1)]^t \quad , \\
&\approx e^{-N(t/\tau)} \quad ,
\end{aligned}
$$

where the first equality is only approximate because of the previously mentioned double-error effect that can occur as t grows larger.

If $\tau = 10^{18}$ cycles for an individual element and a computer is made up of 10^{18} elements, there will be a probability $P_N(1)$ with $N = 10^{18}$ that at least one computer error had occurred in one cycle of

$$
\begin{aligned}
P_N(1) &= 1 - e^{-1} \quad , \\
&= 0.63 \quad ,
\end{aligned}
$$

which is a high error rate.

If each bit were stored redundantly in three elements by increasing the number of elements to 3×10^{18}, one could choose the two that were alike if a discrepancy arose due to error. For such a case

$$
\begin{aligned}
p'(1) &= 3[p(1)]^2 \quad , \\
&= 3/\tau^2 \quad ,
\end{aligned}
$$

$$= 3 \times 10^{-36} \quad .$$

This gives a relaxation time

$$\tau' = 3 \times 10^{35}$$

for the group of three elements. For the overall computer, we have

$$P'_N(1) = 1 - e^{-10^{18}/3x10^{35}} \quad ,$$

$$= 3 \times 10^{-18}$$

as the probability that an error occurred in one cycle. The redundancy has thus decreased the probability for error in one cycle by a factor of 2×10^{17} which is a great improvement.

To keep the time constant for the group of three elements near the high level of 3×10^{35}, the set must be constantly rechecked and, if necessary, reset. For example, after 10^{15} cycles, there would otherwise be a probability of $3 \times 10^{15}/10^{18}$, or 3×10^{-3}, that at least one of the group was in error. Because there would be no check to correct the error, the probability that a second error per cycle would occur in another of the group would be only 2×10^{-18}. The probability of a group error occurring in one cycle would then be $3 \times 10^{-3} \times 2 \times 10^{-18}$, or 6×10^{-21}. Then τ' would have dropped from 3×10^{35} to 1.7×10^{20} which is much closer to $\tau = 10^{18}$ for a single element of the group.

To minimize the possible improper setting of the whole group by an error in the checking device, it is probably best for the device to check the whole group, but be allowed to set only one specific member at a time. Better yet, three checking devices, one to reset each memory element, could protect against total checking-device failure. Three checking devices for a group of five memory elements would probably be a more justifiable ratio. Then τ' would be 10^{53} cycle times. For the overall computer, we would then have τ' computer $= 10^{35}$. If the basic cycle time were 10^{-8} sec, τ' computer $= 10^{27}$ sec, or 3×10^{19}, or 30 billion billion yr. If computer elements didn't age and cause their τ's to increase, after 30 billion yr the probability that at least one error had occurred within the overall computer would be only 10^{-9}.

We thereby see that for very long times the individual aging of the elements actually becomes a critical factor. This aging may cause τ for a given element to either gradually or abruptly decrease toward 2. The only protection against this is to build new hardware before the decay of the old has become important, and to transfer the stored information from the old to the new.

Using the above mentioned techniques should provide nearly perfect reliability for key information and logical processes.

IX. HARDWARE

In digital computers, we are dealing with finite-state devices, whereas the human brain is in certain respects an analogue device having an infinite set of states. Because digital computers are subject to component breakdown, they also are really not entirely finite, but lose their value to us as they drift away from the original finite-state construction designed into them.

The finite character of a digital device should not be thought of as a serious limitation compared to continuous analogue devices. The number of possible programs that could be introduced into a large computer is greater than we could ever have time to actually introduce. For instance, if a computer were programmed to add a 1 to a 60-bit word initially set at 0, once every nanosecond, it would take about 32 yr for the number represented by 60 ones to be reached. It would take 32 yr just to shift one word in storage through all of its possible states. A computer with a photostore of 10^{12} bits will have $2^{10^{12}}$ possible states, which is an unmanageably large number. Restrictions are not severe because of the digital aspect used. Rather, certain specific programs may require more storage space than is available in a specific computer.

Analogue devices are, of course, much more useful than digital devices for certain problems and are often joined with digital devices, each device handling that part of an overall problem for which it is best suited.

Let us now consider the present state of computer memory and information-processing speed compared to that of the brain. Some computers have a very fast central memory with a storage capacity of up to 10^7 bits; slower access discs add about 2×10^9 bits, and a photostore adds another 10^{12} bits. This compares with the brain's storage capacity of perhaps 10^{14} or 10^{15} bits. However, a significant part of the brain is devoted to controlling the very complex functioning of the biological system, rather than to high-level thought processes.

In a computer system, all the information in an encyclopedia volume can be stored in a memory of about 5×10^7 bits. A memory of 10^{15} bits could store all the printed information in the world today.

Although some computers have components that operate up to a million times faster than those of the brain, the brain, nevertheless, can process much more information per second. This is because the brain has more active logic or control components than present computers, and is of primarily parallel construction, whereas computers are primarily serial. The parallel construction frees more active units to operate simultaneously.

The status of computer hardware is rapidly improving, however, because of the rapid advances being made in electronics. According to F. G. Heath,[12] "The most volatile technology of the present industrial age is that provided by electronics. Since the introduction of the transistor in 1948 - which in its day seemed a marvel of compactness compared with the glass vacuum tube - the size of electronic devices has been reduced by a factor of 10 roughly every five years. This works out to a compression approaching 100,000 between 1948 and 1970."

Vacuum-tubes required more than 10 cm^3 to store a bit of information, ferrite cores store around 10^4 bits/cm^3, large-scale integrated circuitry may improve this to 10^5 bits/cm^3 or better, magnetic-bubble technology may store from 10^5 to perhaps 10^9 bits/cm^3 or more. If we can develop devices that store a bit in a space of one cubic micron containing roughly 30 billion atoms, the density would be 10^{12} bits/cm^3, or about that of the brain. Solid-state memory units would not have to be confined to the size of the brain, however. A 10-m^3 block at this last density would give an overall capacity of 10^{21} bits, or a million times that of the brain. We must remind ourselves that only one millionth of such a capacity would be required for storage of all presently printed information.

Computer component speed may continue to increase significantly in the next few decades. Parallel operation by peripheral processors is even now incorporated into the more sophisticated computers. Artificial-intelligence programs can probably make better use of a higher logic-to-memory ratio than can normal programs presently in use. That is, artificial-intelligence programs are more amenable to simultaneous parallel operation of various subsections than are other types of programs. Therefore, hardware incorporating more parallel operation will probably be constructed for artificial-intelligence research.

One of the important factors in developing hardware for artificial-intelligence systems is the limit to which component size can be decreased while retaining both good reliability and good access speed. Freiser and Marcus[13] estimate that biological cells may store around 3×10^{19} bits/cm^3 stably at temperatures of at least 98.6^{o}F. They note that the read-out or transcription rate is only about 5×10^4 bits/sec, however. Note that the poliomyelitus virus is only 12 mμ, or about 36 atoms, in diameter. Such a small volume can still contain about 50,000 atoms. If one made the average diameter of each solid-state component 100 mμ, a cubic meter would contain 10^{21} components.

Assuming that a television receiver can generate 5×10^6 spots of information per second on a screen, two billion such sets would take nearly a full day to lay down 10^{21} spots. The manufacture of an electronic component is certainly more complicated than the task of leaving dark or brightening a spot on a television screen for an instant. Thus, we see that the task of manufacturing 10^{21} components is formidable.

The use of light on photosensitive materials to develop the designs required should continue to be an important method for manufacturing large numbers of components. A wave length of, say, 10^{-7} cm for photons used in component manufacture would allow 5×10^{15} photons per second to be generated by each watt of power transformed to light at that wavelength. One would have to use a large number of photons to develop the overall design of each component, however.

X. GENERAL PROPOSAL

Perhaps the best approach to artificial-intelligence development is to attempt two different developments simultaneously. One area of development should be concerned with creating a more advanced compiler to relieve the programer of as much work as possible. This should decrease the amount of time wasted in laborious programming and free more time for fundamental development. Developments in this area would provide better tools, bookkeeping on computer use, and some improvements in logic.

The second area of development should be directed toward creating a program with advanced goal levels, motivation, high levels of data correlation and code optimization, a well developed hiearchy including low as well as high world-model levels, and powerful discovery and learning techniques.

Both areas should be developed simultaneously so that the improvemants in organization, data correlation, self programming, and general optimization discovered in the second area will be available for improving the first area. The detailed needs for an improved compiler to be used in programming in the second area will not be known until developments in the second area are attempted. Ultimately, the results from both areas will be combined to give the best possible overall system capability. Only if developments in both areas are carried out simultaneously will a very good level of compatibility be likely when the time comes to join the two together.

More specifically, the goals for the two areas of development might be:

A. Advanced Compiler Development

We wish to create a program that can:

- increase the scope of tasks the computer can accomplish mathematically, logically, and scientifically,
- develop and compile complicated programs based on limited input information,
- make reasonable assumptions and use them whenever possible, rather than rejecting a problem,
- interact with and inform the programmer in a more advanced manner,
- take and use the programmer's suggestions about method of attack, etc.,
- do bookkeeping on computer use and optimize on the basis of the results,
- understand a greater amount of information presented in human language,
- correlate large bodies of scientific information originally presented in human language.

B. Intelligence Development

We wish to create a program that can:

- learn facts from somewhat random information,
- develop from primitive to higher world-model levels through interaction with its environment,
- allow a high level of cross-referencing and diverse property interaction,
- reprogram internally,
- optimize internally,
- develop new hierarchies and appropriately optimized codes internally,
- pursue advanced goals,
- create new subgoals internally,
- determine part of its job priorities internally (self motivation),
- visualize internally,
- learn to directly and indirectly control an environment within which it exists,
- solve problems within the environment,
- interact with other intelligent beings in the environment,
- learn methods, language, facts, and theories from other beings within the environment,
- formalize facts learned through inductive reasoning.

Some further reasons for pursuing solutions in the second area are the following. Discovery programs probably provide much the strongest impetus for solving the problem of algorithm development within the computer itself. Statistically aided inductive (discovery) programs are necessary to higher level and wider scope of effective information interaction. Some aspects of discovery programs are required for self-correction. Total internalized understanding will probably only come through programming that encompasses low as well as high world-model levels. The entire spectrum of levels should be present. Aids to facilitate changes in relatively "rigid" programs, such as will be pursued at first in advanced compiler development will probably be insufficient to help much in changing "advanced" programs (the discovery type). The discovery programs should provide much stronger impetus for solving problems of high-level referencing and optimization. The greatest ease in programming and highest levels of analysis will be obtained only with the development of such programs. Understanding of creativity or high-level discovery learning processes and logic also will come only with development of these programs.

The computer should have an internal world model that characterizes enough of the important aspects of the human environment so that it can understand or associate meaning with many terms in human language that are important for any satisfactory, broad, high-level communication with human beings.

Because man is capable of creative thought, perhaps it is good heuristics to see what is required to evolve a program along evolutionary lines involving environmental interaction for learning.

Evolution has certainly programmed some successful heuristics into animal and human mentality.

One example of input to a program for advanced compiler development, which the program should be capable of handling, is:

```
A1 = P2/P1
P  = F1(T)
F1(T) =
T = F2(E)
F2(E) =
E2 = E1 - DELTAE - BIAS + CF
X1 = N1/(N1 + N2)
N1 = WTCD / 112.40
N2 = WTSØL/MW
WTCD = ((BZ - BR)/BF) + ØWT
INPUT DATA
MW =
WTSØL =
BF =
BZ =
ØWT =
BIAS =
CF =
BR =   ,    ,    ,    ,
E1 =   ,    ,    ,    ,
DELTAE =   ,    ,    ,    ,
E3 =   ,    ,    ,    ,
E4 =   ,    ,    ,    ,
RESULTS
BSHIFT = BZ - BR, N1, T1, T2,
TTØP1 = T3, TTØP2 = T4, DELTAT =
T1 - T2, A1, 1 - A1, N2 = 1 - N1,
DUNDEE = (1 - A1)/N2, P1, P2
```

Of course the programmer would fill in the exact numerical or functional information on the right side of the incomplete equations above before introducing the problem to the computer. The computer should then do algebraic manipulations to simplify expressions where possible. A listing of the steps in simplification should be printed out for the programmer's information. A program should be compiled and run if enough information is present; otherwise the programmer should be told what further information is required. A FORTRAN output sketch of the final program might be helpful. Part of the computer's decision to run or not run a program should be based on the number and type of uncertainties in the problem input and the estimated running time of the problem.

An example of a beginning problem in intelligence development was discussed in Sec. VI.

XI. CONCLUSIONS

The organization of the brain, which presumably arose through former chance interactions of our ancestors with their environment, does not seem really efficient for high levels of thought. This is partially demonstrated by considering that a present-day computer with a billionth the informational processing speed and memory of the brain can solve complex but narrow problems a million times more quickly than can a man using a desk calculator. This is true not only for arithmetic problems but also for common deductive-logic problems. This is a factor of 10^{15} improvement in organization for narrow problems. A far better heuristic approach can be expected when extensive research into artificial intelligence improves our methods of carrying out inductive logic.

By various duplication techniques in programming or in hardware construction, the stability of information stored in the solid-state devices can be increased to any level necessary. The artificial intelligence then need only decide the value of retaining or discarding a piece of information at any given time and can either store or erase it. The high-level part of the program does not require constant repetition in order to retain obscure facts in memory as does the brain.

Improved learning and discovery techniques, faster retention, larger and far better memory, a faster information processing rate, and a greater well-coordinated fund of knowledge give artificial intelligence an edge over man if these areas are developed. Its capacity for self-understanding owing to the availability of its hardware diagrams and logic program gives the artificial intelligence a much better chance of improving itself rapidly than does the complex, little-understood biological mechanism of the human brain.

A high-level artificial intelligence should be able to accelerate advances in almost any field of research and development.

Now would seem to be the best time for putting a major national effort into artificial-intelligence development.

REFERENCES

1. H. L. Davis, Scientific Research 4, No. 22, 21 (1969).
2. D. Michie, Nature 228, 717 (1970).
3. European AISB Newsletter, No. 9, 4 (1969).
4. R. Buckminster Fuller, Operating Manual for Spaceship Earth (Southern Illinois University Press, Carbondale and Edwardsville, 1969), p. 71.
5. D. Premack, Psychology Today 4, No. 4, 54 (1970).
6. C. R. Michael, Scientific American 220, No. 5, 105 (1969).
7. M. A. Arbib, Brains, Machines, and Mathematics (McGraw-Hill Book Company, New York, 1964), p. 31.
8. A. R. Luria, Scientific American 222, No. 3, 66 (1970).
9. M. S. Gazzaniga, Scientific American 217, No. 2, 24 (1967).
10. N. M. Amosov, Modeling of Thinking and the Mind (Spartan Books, New York, 1967), p. 114.
11. L. J. Fogel, A. J. Owens, and J. J. Walsh, Artificial Intelligence Through Simulated Evolution (John Wiley & Sons, Inc., New York, 1966).
12. F. G. Heath, Scientific American 222, No. 2, 22 (1970).
13. M. Freiser and P. Marcus, IEEE Transactions on Magnetics 5, No. 2, 82 (1969).

APPENDIX

PROBABILITY AND INDUCTIVE REASONING

Inductive reasoning uses probability analysis as a basic tool for discovering restrictions in the environment. These restrictions limit the number of possibilities allowed, thus decreasing the "randomness" and increasing the "order" of the environment. If no order is present in the environment, then probability analysis will prove uninformative. The more order one discovers in the environment, the more one is said to "understand" the environment.

Consider a numerical variable v, derived by some method from one or more properties of the environment. Many checks of the environment may be made to determine the momentary value of the variable. As the number of checks, n, grows large, the variable is observed for trends.

The degree of order decreases with the results observed as follows:

(1) $v = 0$ for all n, or $v = 1$ for all n,
(2) $v =$ finite constant for all n,
(3) $v \to 0$ or 1 as n grows large,
(4) $v \to$ finite constant as n grows large,
(5) $v = f(n) \not\to$ constant as n grows large,
(6) v remains within finite bounds,
(7) none of the above are discovered.

If no order is determined directly from v, other variables derived from v and n may be checked for possible order.

The most important reason that 0 and 1 are considered more significant than other constants is that the frequency of occurrence of a given type of event relative to some total number of events is a very useful function that has 0 and 1 as extremes. Also worth noting is that 0 is a unique number relative to addition or subtraction so that

$0 + 0 + \ldots + 0 = 0,$

$0 - 0 - \ldots - 0 = 0,$

and 1 is a unique number relative to multiplication or division so that

$1 \times 1 \times \ldots \times 1 = 1,$

$1 / 1 / \ldots / 1 = 1.$

The equation $n^0 = 1$ relates 0 and 1 for any nonzero real number, n. The binary system based on 0 and 1 can express any number to any required degree of accuracy. Finally, Boolean logic based on 0 and 1 can be used for logical expressions and deduction.

If v falls into categories 6 or 7, a statistical treatment may be carried out in the hope of discovering some order. A new variable, $\bar{v}$, is derived from v and n using the definition

$$\bar{v} \equiv \frac{1}{n} \sum_{i=1}^{n} v_i \quad .$$

This average takes on a value for each value of n, so that $\bar{v}$ can then be checked for order according to categories 3, 4, 6, and 7.

If $\bar{v}$ falls into categories 3 or 4, further information may be obtained by checking $\overline{v^2}$, defined as

$$\overline{v^2} \equiv \frac{1}{n} \sum_{i=1}^{n} v_i^2 \quad ,$$

or the variance,

$$V \equiv \frac{1}{n} \sum (v_i - \bar{v})^2 \quad .$$

Still higher moments about the mean might also be important to check.

Finally, although $\bar{v}$ for some variable does not approach a constant limit over long intervals, it may tend to be confined within relatively narrow limits over smaller intervals. Thus, averaging over moderate intervals may yield a new set that has much less scatter over any small range of n. A continuous curve passed through the new set may yield a useful function of n.

Now we turn to observation of the environment. One may specify certain conditions that must be met before making an observation. Once these conditions have been met, one checks to see whether a certain final condition, such as, perhaps, the occurrence of a certain property, has also been met. If the condition has been met, we say a certain event, E, has occurred; otherwise, we say the event has not occurred. A numerical variable, x, may be obtained from such observations by assigning the value 1 to the variable whenever E occurs and assigning 0 otherwise.

If x satisfies the requirements of category 1, we say that the initial set of conditions determines the occurrence or nonoccurrence of the specified event. We are then highly confident that whenever the initial conditions are brought about, the final condition is controlled.

If x does not fall within category 1, we turn to statistical analysis to see whether the initial conditions have established any control over the final condition.

If $\bar{x}$ satisfies category 4, a degree of control has been established. Then we call the constant approached by $\bar{x}$ the probability, p, of event E. Note that $\bar{x}$ does <u>not</u> necessarily have to satisfy category 4; $\bar{x}$ may only fit category 6, in which case, there is <u>no</u> probability associated with $\bar{x}$.

To understand the last statement, consider a coin. If one places a coin carefully he can cause heads or tails to occur as he desires. If someone else records how the coin is placed each time, an analysis involving averaging can be undertaken. If a 0 is assigned to tails and a 1 to heads, the person controlling the coin might place it so that the average approached any desired constant between 0 and 1. However, rather than doing this, he can also place the coin so that the average <u>never</u> settles toward a constant value! This is accomplished by, over a period of time, setting a short-term average that always lies significantly to one specific side of the overall average. The short-term averages are always kept on the same side of the long-term average until the overall average has shifted significantly. Sometime after significant shift of the overall average, the short-term averages may be shifted to the other side. Thus, the long-term average may be caused to approach any constant value as n grows large, or it may be caused to fluctuate constantly between any pair of limits within the range from 0 to 1.

One cannot logically deduce that a person, when he flips the coin into the air and lets it fall some distance before bouncing to a standstill, will cause the average to approach a finite constant. Rather, it is observed by induction that when people flip coins in this manner the result is an average that <u>does</u> approach a constant. The symmetry of the coin is noted to correlate well with the value 1/2.

When a die is thrown, it is noted that symmetry alone is not enough to set the probability of one face at 1/6. Further, the density must be uniform and the magnetization must be zero.

We can no more deduce that any type of event must have a probability associated with it than we can deduce that gravity, matter, or energy must exist. The fact that many events do have a probability is an indication of the great amount of ordering in our environment.

Observations of the physical environment thus lead one to conclude that at the most elemental levels of the physical universe it must be theoretically possible to associate a probability with any given variable by placing appropriate constraints upon enough other variables. This fact might be called the <u>Law of Natural Probability</u>. This law is as fundamental to the working of our universe as are the laws of conservation of energy and momentum. Philosophically, because all other laws can be expressed in terms of this one, the Natural Probability Law may be considered the most fundamental of all. This law is the reason that statistical analysis is so useful.

Note that the results observed from flipping a coin anticipate the probability approach of Quantum Mechanics. The probability aspect of nature is also fundamental to the possibility of evolution, inductive reasoning, and learning. The development of intelligent life rests firmly upon this foundation.

Because any condition can be observed to only a finite degree of accuracy, everything dealing with the environment, including a probability value or the constancy of a specified condition, is subject to some degree of uncertainty.

To simplify the development of statistical theory, we abstract the properties observed to be important in the environment and idealize them. Thus, we discuss ideal events where restrictions can be kept absolutely constant and where $\lim_{n\to\infty} \bar{x} = P$, an absolute constant. Thus we idealize the concept of probability.

Probability correlations between two types of events, E_1 and E_2, are very important for discovering order among many kinds of events. Thus, we begin to deal with compound events and set theory. Consider the compound event, E_3, which is the union of E_1 and E_2. That is, E_3 is defined as being the occurrence of either E_1, or E_2, or of both. This can be diagrammed as in Fig. 1A. Another compound event, E_4, may be defined as the intersection of E_1 and E_2 which is diagrammed as in Fig. 2A.

The probability that out of n events x events will have both E_1 and E_2 occurring together is $P(E_4)$, or $P(E_1 \bullet E_2)$. The probability that out of n events x events will have E_1, E_2, or both, occurring is $P(E_3)$ or $P(E_1 + E_2)$. We note by inspecting the diagrams that

$$P(E_1 + E_2) = P(E_1) + P(E_2) - P(E_1 \bullet E_2) \quad .$$

The conditional probability, $P(E_2|E_1)$, of event E_2 on the condition that E_1 has occurred is

$$P(E_2|E_1) = \frac{P(E_1 \bullet E_2)}{P(E_1)} \quad .$$

If the occurrence of one event does <u>not</u> change the probability of occurrence of another, the two events are said to be "independent." Thus, E_1 and E_2 are independent if

$$P(E_2|E_1) = P(E_2) \quad .$$

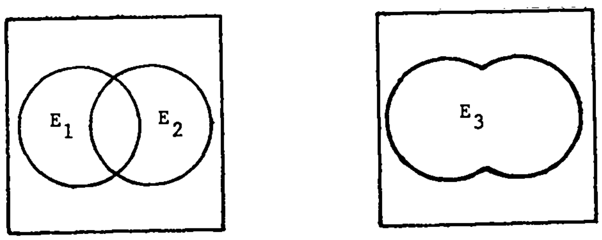

Fig. 1A. Events E_1, E_2, and $E_3 \equiv E_1 + E_2$.

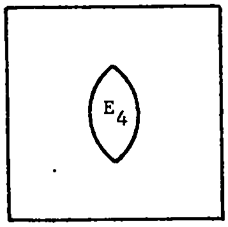

Fig. 2A. Event $E_4 \equiv E_1 \bullet E_2$.

Independence of events allows a simplification of the multiplication

$$P(E_1 \bullet E_2) = P(E_1)\, P\,(E_2|E_1)$$

to

$$P(E_1 \bullet E_2) = P(E_1)\, P\,(E_2) \quad .$$

Also, rearranging the last equation gives

$$0 = P(E_1 \bullet E_2) - P(E_1)\, P\,(E_2) \quad .$$

If

$$P(E_1) = \frac{1}{n} \sum_{i=1}^{n} x_i = \bar{x} \text{ and } P(E_2) = \frac{1}{n} \sum_{i=1}^{n} y_i = \bar{y} \quad ,$$

then

$$0 = \frac{1}{n} \sum_{i=1}^{n} (x_i\, y_i) - \bar{x}\,\bar{y} \quad ,$$

$$= \frac{1}{n} \sum_{i=1}^{n} (x_i\, y_i) + \bar{x}\,\bar{y} - 2\,\bar{x}\,\bar{y} \quad ,$$

$$= \frac{1}{n} \sum_{i=1}^{n} \left(x_i\, y_i + \bar{x}\,\bar{y} - \bar{x}\, y_i - \bar{y}\, x_i\right) \quad ,$$

$$= \frac{1}{n} \sum_{i=1}^{n} \left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right) \quad .$$

This last term is called the "covariance" of x and y and is denoted by cov(x, y). If the covariance is zero, events E_1 and E_2 are independent.

We note that the variance of the sum of two variables is given by

$$V(x + y) = \frac{1}{n} \sum_{i=1}^{n} \left[(x_i + y_i) - (\bar{x} + \bar{y})\right]^2 \quad ,$$

$$= \frac{1}{n} \sum_{i=1}^{n} \left[(x_i - \bar{x}) + (y_i - \bar{y})\right]^2 \quad ,$$

$$= \frac{1}{n} \sum_{i=1}^{n} \left[(x_i - \bar{x})^2 + 2\,(x_i - \bar{x})(y_i - \bar{y}) + (y_i - \bar{y})^2\right] \quad ,$$

$$= V(x) + 2\,\mathrm{cov}(x,y) + V(y) \quad .$$

If E_1 and E_2 are independent,

$$V(x + y) = V(x) + V(y) \quad .$$

Note that if $P(E_1 \bullet E_2) = 0$, E_1 and E_2 are mutually exclusive; that is, if E_1 occurs, E_2 does not occur. If $P(E_1 \bullet E_2) \,/\, P(E_1)\, P\,(E_2) = 1$, E_1 and E_2 are independent. If $P(E_2|E_1) = 1$, E_1 is included in E_2. If $P(E_1|E_2) = 1$, E_2 is included in E_1. If E_1 and E_2 are mutually inclusive, whenever E_1 occurs, E_2 also occurs and vice versa. This last situation represents one of the two highest forms of ordering through probability correlation. The other highest form of order occurs when $P(E_1 + E_2) = 1$ and $P(E_1 \bullet E_2) = 0$, in which case all events that are not E_2 must be E_1.

Note that the events E_1 and E_2 break up the sample space into four elementary regions as shown in Fig. 3A.

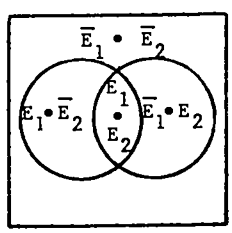

Fig. 3A. A sample space having two primary events.

There are 16 possible sets that can be made up from this elementary set, so the elementary set is important and the probabilities associated with each element can be determined. The sum of these probabilities must add up to unity.

We thus conclude that the most important probabilities to determine are:

(1) $P(E_1)$
(2) $P(E_2)$
(3) $P(E_1 \bullet E_2)$
(4) $P(E_1 \bullet \overline{E}_2)$
(5) $P(\overline{E}_1 \bullet E_2)$
(6) $P(\overline{E}_1 \bullet \overline{E}_2)$
(7) $P(E_2|E_1) - P(E_2)$
(8) $P(E_1|E_2) - P(E_1)$

Whenever (7) is zero, (8) must also be zero, indicating that E_1 and E_2 are independent.

KT/jb: 353 (100)