THE MYSTERY OF

PROBLEM REPRESENTATIONS

A Tutorial Introduction To A Young

Area of Research

by

Judea Pearl
School of Engineering and Applied Science
University of California, Los Angeles

March 1972

<u>Table of Contents</u>

		Page
1.	Introduction.	1
2.	Representations in Puzzle Solving.	5
3.	Isomorphic vs. Homomorphic Transformations.	9
4.	Spectral Representations.	12
5.	Steps Toward Formal Theories	20
6.	A Practical Application of Isomorphic Transformations	26
	References.	28

1. INTRODUCTION

Natural languages, mathematical equations, computer programs, visual diagrams, are all various forms devised by man to represent his environment and the problems it poses. Man's attitude toward his environment is so intimately connected with the form invented for its representation that these two, the modelled and the model, the content and its form, have been the subjects of endless misuse and confusion. Thousands of interpretations and inferences were constructed on the basis of the numerical value of letters (Gimatria) in the Bible. The axioms of Euclid were for years regarded as a gift of the gods, just about the only thing we "know" for sure. In our times stock-market chartists manipulating the geometry of "resistance levels" and "consolidation lines" still command high respect in financial circles.

Modern science has brought about a clear distinction between the empirical world and its modal representation, a distinction which tends to underscore the role played by the latter. An attitude frequently encountered in scientific circles is that since model-language is only an artificial device invented to describe the "real-thing", variations among different representations are secondary in importance, pertaining merely to such trivial conveniences as compactness ~~conciseness~~, simplicity, mnemonic economy, ease of communication, etc.

That such attitude ignores one of the most essential factors in man's ability to understand, predict and control his environment is clearly demonstrated by the historical pattern that both conceptual and technological advances always follow progress (change) in notational schemes. Modern science can hardly be envisioned without Galileo's discovery that empirical observations can be represented in algebraic form, and this recognition took place only after the advent of positional number system. Einstein's theory of relativity came into being only after the development of tensorial vocabulary, while quantum mechanics owes its impetus to notational developments in the calculus of linear operators. This pattern suggests that the choice of working representations is not a matter of sheer luxury but is a prime determinant in configuring our cognitive structure.

Facing todays environmental, sociological, and economical problems the question of representation assumes double importance. The complexity of these problems renders them utterly untractable without substantial abstraction which in turn is a sensitive function of the cognitive structure of the problem solver. Any working solution of these problems must employ interdisciplinary activities which involve individuals with diverse layouts of cognitive structures. In our rapidly changing world we witness the phenomenon that classification schemes of yesterday quickly lose their effectiveness when new problems are created. Thus, the introduction of novelties can no longer be accomplished by superposition upon established representations.

The advent of fast digital computers brings about an added dimension. These new creatures are not born with an innate structure similar to ours, it stands to reason therefore that representations suitable for us will not find natural matchings to machine structures. It is also a fact that programmable machines are potentially more flexible in changing their internal processing structure than human beings. This suggests that if new representations of problem situations are warranted these would be more likely to be accomplished in machine environments rather than in our brain structure.

The aim of a Theory of Representations is to establish generic connections between the formal structure of representations and the functional attributes of their usage, and thus help guide the process of selection or generation of representation to match a particular class of problems.

Several factors may account for the absence of Theory or Representations. Cognitive psychologists have long been experimenting with the effect of representation on human problem-solving performance. However, the absence of adequate means for measuring and controlling the cognitive structure of the problem-solver, prevented the development of quantitative psychological theories of representations. Computer Science - the science of symbols - could have mobilized its resources toward the development of representation theory. Unfortunately, many computer scientists are preoccupied with speed-cost improvements of their symbol-manipulating machines with very little attention to the environment which generates the input symbols and the external implications of the output symbols.

The present report aims at calling attention to the fascinating problems of representation, with the hope that common efforts of workers from various fields would provide the necessary impetus for the development of the theory. A logical first step in the development of any theory is the compilation of case studies demonstrating the relationships which the theory attempts to capture. Unfortunately, in the case at hand, we are in possession of only very few examples where the role of representation and its relation to the structure of the problem environment and the problem-solver is well understood. Understanding this relation implies that we are able to fully predict the effect of any change in representation on the performance of the problem-solving procedure. That such understanding is lacking in human problem solving situations is not surprising, as we usually lack adequate descriptions of many human activities. But even in machine environments we only have very crude means for describing efficiency of problem-solving procedures, as the concept "complexity of computation" itself is not yet well developed [see Ref. 1].

The state of affairs of representation theory is best summarized in H. Simon's [Ref. 2] words:

> "An early step toward understanding any set of phenomena is to learn what kinds of things there are in the set--to develop a taxonomy. This step has not yet been taken with respect to representations. We have only a sketchy and incomplete knowledge of the different ways in which problems can be represented and much less knowledge of the significance of the differences."

Given this state of affairs, the addition of any case where the effect of representation is well understood (not merely recognized) should be welcome.

The program followed in this report is first to present simple examples where the role of representation is clearly recognized as the dominant factor leading to a solution. In Section 3 we will discuss some generic properties of representations as manifested by the examples above. In Section 4 we will present an example where the role of representation is not only recognized, but can be described quantitatively. The example, employing spectral representations in the field of signal processing, belongs to the rare class of well understood cases, so badly needed for the development of a general theory of representation. In Section 5 a brief description will be given of possible directions to formalize a theory of representation. Section 6 describes a medical application that could benefit directly from the study of representations.

2. REPRESENTATIONS IN PUZZLE SOLVING

Let us make it clear what we mean by "representation" by citing two classical puzzles which contain most of the essential ingredients and at the same time are free from unnecessary details.

Example 1: EXCHANGE THE KNIGHTS.

Figure 1a contains a 3x3 section of a checker board with two black and two white knights. It is required to make the black knights trade positions with the white ones using minimum number of legal chess moves. In its present representation it takes the average individual a good fraction of an hour to figure out the answer. The solution becomes much more difficult if no pencil and paper are allowed.

Figure 1b describes another representation of the same problem. Recognizing that a knight can reach any square from exactly two other positions, the squares can be arranged along a closed loop with neighbors corresponding to positions separated by a single knight move. In this representation the task of exchanging the positions of the knights is trivial; one immediately realizes that all figures must move four steps clockwise (or counterclockwise).

Example 2: DOMINO COVERING PUZZLE.

Figure 2a shows an 8x8 checkerboard with two corners removed. It is required to completely cover the board with domino tiles each the size of two adjacent squares. It is obviously an enormous task to investigate all possible tile patterns, instead, Figure 2b contains a simplified representation of the problem. Recognizing that each tile must cover one black and one white square, one realizes that the number of uncovered black squares as well as that of the uncovered white squares decreases by one for each tile added. Starting with 32 white and 30 black squares (32-30) it is obvious

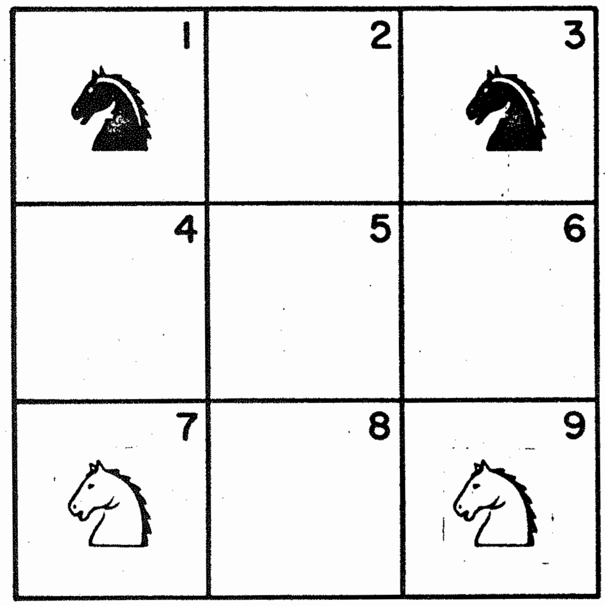

FIGURE 1A

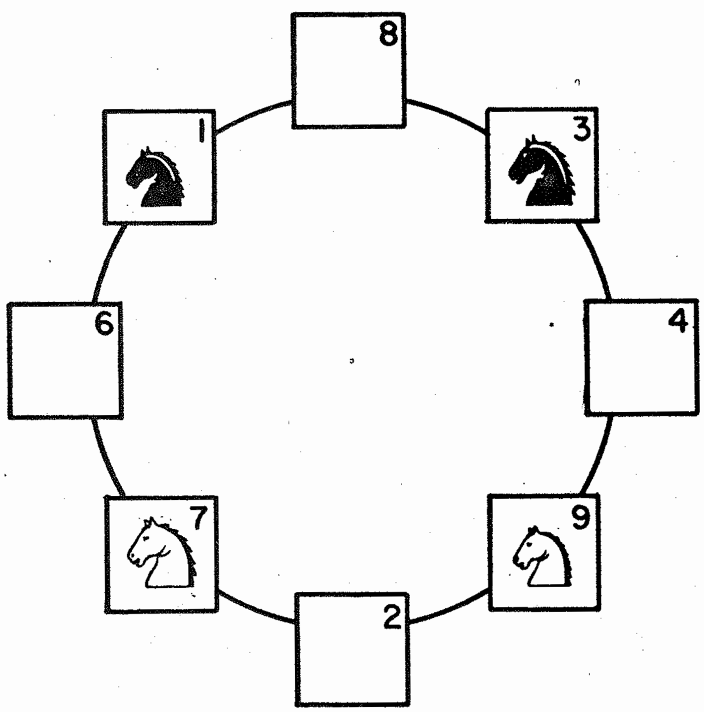

FIGURE 1B

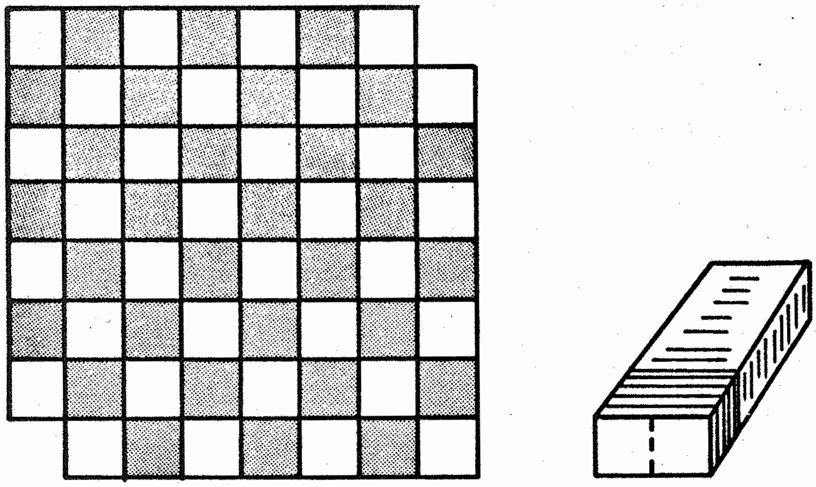

FIGURE 2A

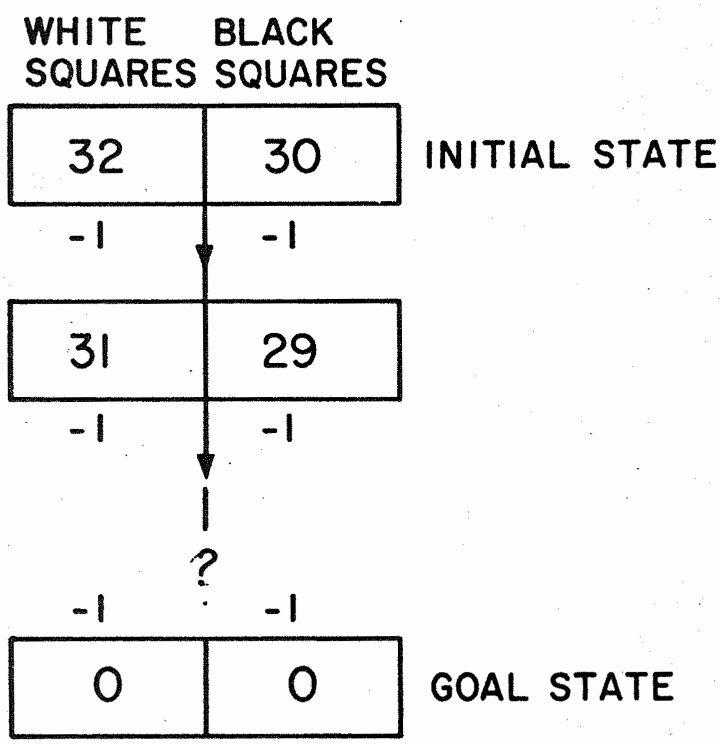

FIGURE 2B

that complete coverage (O-O) is impossible.

While these examples demonstrate how changes in representations can be instrumental in attaining a desired solution, they immediately bring to mind several questions:

1. Can most problems (especially real-life problems) benefit from a clever change of representation, or is it merely a curious trick among few artificially constructed puzzles?
2. Is the reduction in complexity seen in these two examples a universal phenomenon, or does it vary from individual to individual and from one machine to another?
3. Is there a systematic way of searching for a "good" representation? Can it be mechanized? Will it lead to a better overall economy than solution procedures processed in the original formulation?

Unfortunately, although it is generally recognized that the choice of representation is a critical step in every problem-solving situation, only very few steps were taken toward systematization of the choice procedure. Amarel [Ref. 3] demonstrated how the classical puzzle of Missionary and Cannibals can be solved by successive changes in representation, each change makes the problem simpler until at last the solution becomes apparent. Similar effects were considered in Syntactic Analysis, Theorem Proving and Question-answering Systems [Ref. 4]. The question of the feasibility and economy of mechanized procedures, however, remains open. To the questions stated above, only the second can be given a fairly complete answer, as is attempted in the next section.

3. ISOMORPHIC VS. HOMOMORPHIC TRANSFORMATIONS

Without going too deep into the details of the examples cited in Section 2, a very clear distinction can be made between the two. In the Knights Puzzle every configuration in representation 1a has corresponding to it one and only one configuration in representation 1b. This is not the case in the Domino Puzzle; knowing the number of covered black and white squares does not permit us to go back and identify the exact configuration on the board. It is sufficient, though, to prove the impossibility of the covering task.

This observation leads to a distinction between Isomorphic and Homomorphic transformations:

Isomorphic Transformations preserves a one-to-one correspondence between configurations (states and operators) as exemplified in the Knights Puzzle. Other names used for this class are "Resolution Preserving Transformations" and "Epistemologically Equivalent Representations" indicating ~~the fact~~ that both representations contain the same amount of knowledge or information.

An interesting phenomena of the Knights Puzzle is that if programmed for solution on an electronic computer (say through a flying spot scanner input) the two representations would assume almost identical program forms, and therefore would be regarded equally difficult. The simplification affected by isomorphic transformations is obviously in the eyes of the beholder--the problem solver, depending on the particular internal structure of his processor. The peculiar structure of our visual system combined with our common experience of pushing shopping carts in supermarkets or perhaps driving around single-lane tracks are instrumental in rendering the loop structure "simpler" than its original square-board representation.

Homomorphic Transformations as exemplified by the Domino Puzzle involve reduction in detail (also called "loss of resolution" or "abstraction") whereby groups of configurations (states and operators) sharing common properties are lumped together to form "super-states" or categories, thus losing their individual identity. The categories chosen for such lumping must be those "relevant" for the solution process in the sense that a solution found for the super-states representation can be referred back and applied to the original representation. Problem reduction of this type is usually considered universally beneficial (to both man and machine) as it reduces the dimensionality of the search space and thus simplifies the search task seemingly regardless of the structure of the processor doing the search. For this, and similar reasons, the significance of Homomorphic Transformations seems easier to analyze than that of Isomorphic Transformations.

The independence on processor structure, however, is only apparent, in fact the merit achieved by any detail-reducing transformation can only be evaluated in the context of the language available to the problem-solver. Imagine for instance, that representations 2a and 2b are offered to an individual who hasn't yet grasped the concept of "numbers" and therefore is unfamiliar with arithmetic operations. To such an individual representation 2a would seem infinitely simpler than 2b. To conclude that the uncovered board corresponds to a state: (32, 30) presupposes that one knows how to count. Without the rules of counting the correspondence between the two representations may become as difficult (perhaps more difficult) than the solution itself. It is similarly obvious that the concept of "parity" (black or white squares) must be mastered by the problem-solver before he can utilize representation 2b.

To further stress this point let us cite an extreme example of "useless abstraction". Consider the problem of covering an arbitrary collection of squares with domino tiles. To reduce unnecessary detail let us focus on a single configurational property, a property we call "coverable". In the abstracted representation each configuration will be represented by a single number; "1" if it is coverable and "0" if it is not coverable. Clearly, to determine whether a certain collection of squares can be covered by domino tiles it suffices to glance at the number appearing in its abstracted representation, no additional search is required. Needless to say, in spite of its substantial reduction in search space such abstraction would be utterly useless for discovering whether or not the collection can be covered. A sheer reduction into operationally useful categories is not by itself sufficient to guarantee a simple solution. It must be coupled with a processor containing a classification mechanism to decide whether or not a new configuration possesses the categories considered, and a procedural language for manipulating these categories.

Homomorphic transformations can be thought of as Isomorphic transformations followed by omission of "irrelevant" categories. While the omission itself is a trivial part of the solution process, it is the <u>separation</u> between relevant and irrelevant categories which should be credited to isomorphic changes in representation. It is for that reason that this report focuses on isomorphic transformations to reveal the true significance of shifts in representation.

4. SPECTRAL REPRESENTATIONS

In 1807 a jury consisting of Legendre, Lagrange, and Laplace rejected Fourier's memoir on heat conduction because it lacked proof. Today much work in Mathematics, Physics, and Systems Science stems from Fourier's unproved conviction that his trigonometric series can represent arbitrary functions. It is a tribute to science that it compromised its proclaimed policy of truth and vigor and allowed its hunger for convenient representations to take its course.

On their first exposure to spectral analysis, engineering students are usually convinced that the advantage of using Fourier transforms (and similarly its associated Laplace and z-transforms) is that it reduces linear-time-invariant differential equations to algebraic equations which can be readily solved. This motivation quickly loses its main punch as students begin to solve differential equations on digital computers which renders savings in hand calculations - a meager prize indeed for their agonizing struggle with Fourier Integral Tables. The practicing communication engineer, on the other hand, (with his first lesson in Fourier series long forgotten) would become utterly incapacitated if forbidden the use of the frequency decomposition concept.

Clearly, the question of representing signals in the time domain or in the frequency domain constitutes a pure example of choice among isomorphic representations. Its extensive use in the mature field of communication could have served as a test vehicle to reveal the generic principles underlying tradeoffs among arbitrary isomorphic representations. This, however, could not be accomplished so long as the assessments were made by subjective judgments of human beings (engineers) whose measure of computational convenience depends on such variables as natural ability, training, motivation, aesthetics, etc. Fortunately, the introduction of digital computers into real-time communication systems has stripped communication engineers from their position of sole judges of convenience and has provided us with a more objective slide-rule for measuring 'goodness' of representations and its relation to the structure of the problem solver. Moreover, new spectral representations (such as Walsh, Haar, etc.) have become popular contending the supremacy of the Fourier representation on the basis of better fitness in digital environment. The question of choosing a proper representation has finally assumed practical technological importance.

What do we mean when we say that we process information in the frequency domain? We ~~It~~ mean~~s~~ that after a time signal is sampled and quantized the vector so obtained is multiplied by a unitary matrix $F\left[F_{kl} = \frac{1}{\sqrt{N}} \exp(ikl\ 2\pi/N)\right]$ and additional processing is then performed on the resultant vector called the frequency spectrum or the Fourier transform. The question immediately arises: if a time-signal and its Fourier transform differ only by a rotation of the coordinates system but otherwise contain the same information (epistemological equivalence), what difference does it make if the additional processing takes place in the time or frequency domain? Indeed, looking at a communication system from the outside there is no way of telling whether Fourier transform was ever employed.

It turns out that when a communication engineer admits to have been using Fourier processing he invariably implies some additional restrictions on the processing. What he actually implies is that after the frequency spectrum vector is obtained, its components are to be handled <u>independently</u> of each other in the immediate subsequent processing stage (see Figure 3). For instance, if he is doing filtering he would multiply each spectral component

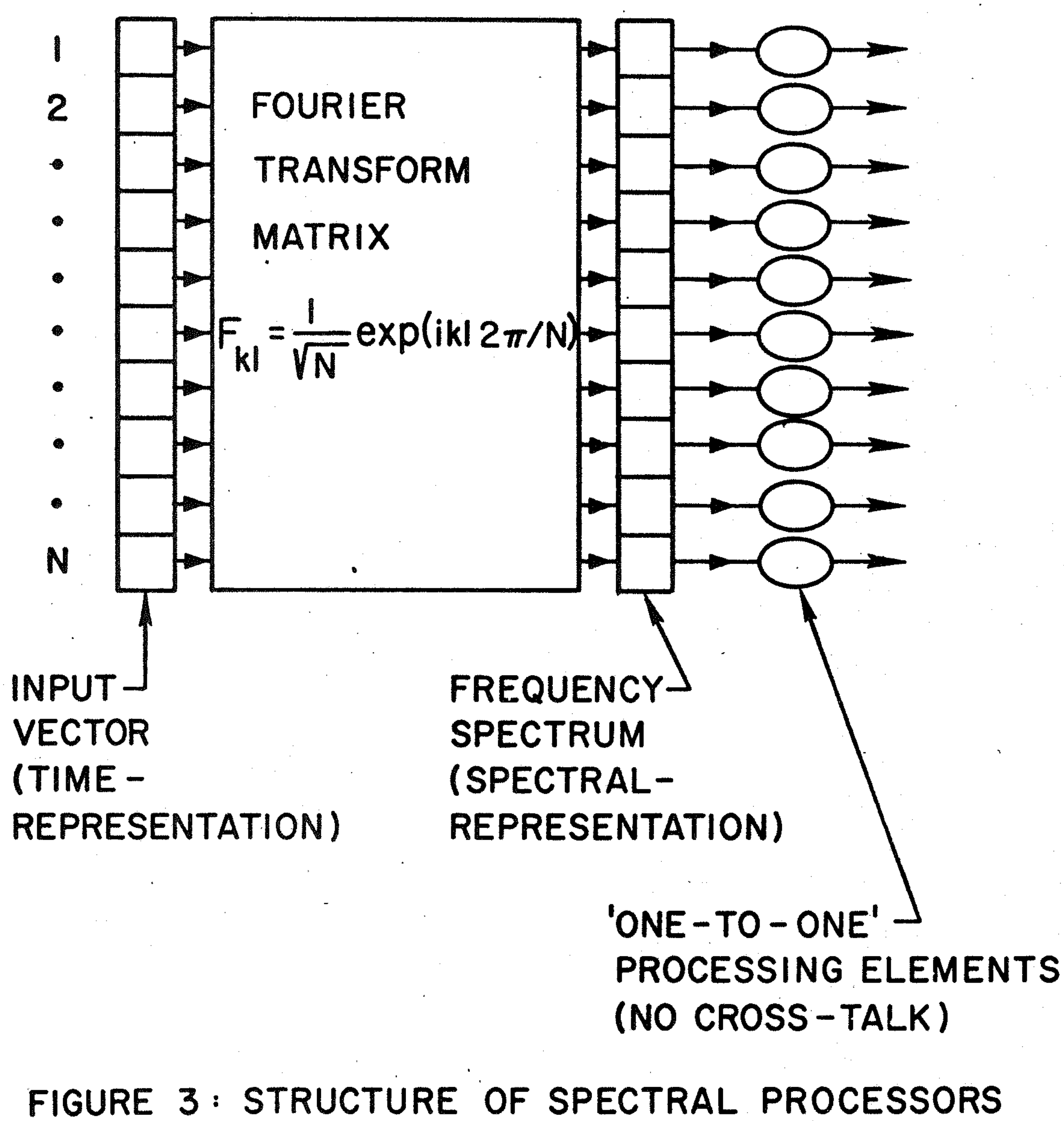

FIGURE 3: STRUCTURE OF SPECTRAL PROCESSORS

by a pre-determined scalar in order to amplify the desired frequencies and attenuate the undesired ones. If he is doing coding then he would assign a binary code to each spectral component regardless of the magnitude of its neighbors. If he is attempting to classify patterns he may wish to truncate his spectral vector, throwing away some irrelevant information, abstracting only few components (also referred to as features, categories, predicates, attributes, etc.) which he believes would suffice for subsequent classifying decisions. The common feature among these operations is that each spectral component is recalled from memory only once. Thus, if the length of the spectral vector is N, we may say that these operations require N primitive computations or a computational complexity of degree N.

The identification of computational complexity with the number of memory lookups is somewhat coarse in that some primitive operations (e.g., coding) can be more complex than others (e.g., multiplication). However, so long as we stay within the framework of the same type of tasks, the number of memory lookups is a realistic measure of complexity. It also has a simple invariant property, if instead of a serial computer one attempts to use a parallel processor (for which the most celebrated ideal is the visual section of the brain) then the requirement of N memory cycles is converted into a requirement for N single-input processing elements, one for every spectral component.

What is the alternative to such restricted, one-to-one, structures*? In general, the operation we may wish to apply to each spectral component can depend on the magnitude of all other components, yielding complexity of degree N^2. In the case of linear filters, for example, the unrestricted filtering operation can be performed by multiplying the input vector by an NxN

*This structure was originally given the name "Basis Restricted", see Ref. 5.

matrix. Indeed, such operation requires on the order of N^2 multiplications and additions. Thus, the motivation behind the restricted computational structure is reducing its computational complexity from degree N^2 to N. This reduction would seem even more impressive if we consider adaptive systems in which the N processing operations are not fixed but must be learned from past experience. It is certainly much easier to control, train, and adjust a system with N undetermined elements than it is with N^2 undetermined elements.

Granted that complexity-reducing structures are desired and sometimes even necessary, what is the role of representations in all this? The reduction in computational complexity is certainly not being accomplished without a price. The performance (e.g., signal to noise ratio, classification errors, etc.) achievable by optimizing a restricted structure with N elements is naturally lower than that achievable with N^2 elements. The difference between the two, however, varies widely with the choice of representations, and in many cases it can be shown that a representation (unitary transformation) exists which makes this difference vanish.

The connection between the representation used and the performance degradation of "basis restricted" processors was treated in Ref. 5-6 for the tasks of linear filtering and source coding. In these two cases (and many others of the same generic type) it can be shown that the optimal representation is achieved by that unitary transformation which renders the spectral components statistically uncorrelated (also called the Karhunen-Loeve expansion) and that for the whole wide class of stationary random-inputs, the Fourier transform approaches the optimal performance like $\sqrt{1/N}$. The inquisitive engineering student is usually forced to wait until his third college year before learning the second desirable property of the Fourier representation: The Fourier coefficients of stationary random-signals are "almost" uncorrelated.*

There still remains the question of whether the computational effort connected with the transformation itself is not larger than the effort saved in subsequent computations. The answer is twofold. First, the Fourier transform matrix obeys some very useful group properties which permits its execution with N log N computations [Ref. 8]. Second, even in the absence of such luxury, the fact that the Fourier representation is close to optimal for such a large and common class of statistical environments makes it very attractive for use in adaptive systems. Here, if only the stationarity of the environment is known but not its ~~exact~~ detailed characteristics, it would be advisable to make a one time investment and implement the Fourier-transform in hardware form, thus freeing the software system to handle the search for the set of N variable elements. The nice thing about such a system is that the fixed and variable properties of the environment are handled by two corresponding but separate sections of the processor. The former by a fast, fixed, and expensive hardware section, the latter by a slow, variable but inexpensive software.

At this point it is very hard to resist temptations to draw analogies with the structure of the brain. Similar hardware-software tradeoff considerations can be argued from an evolutionary viewpoint. The tasks of handling recurrent environmental tasks such as edge enhancement, corner detection, etc. are said to be given to innate sections such as the retinal network and Hubel-Wiesel type cells [Ref. 9]. The handling of the variable parts of our

*A similar theorem was proved by Root [Ref. 7] for the case of continuous-time signals.

environment is attributed to higher and trainable mental processes which are operationally slower but offer better genetic economy.

This brings us back to our original objective of guessing some general principles governing representations from particular examples we thoroughly understand. Spectral representations in signal processing applications offer such an example. It exhibits several characteristics:

1. The influence of isomorphic representations on the input-output performance of a system is significant only when the internal structure of the system is restricted, (e.g., by considerations of computational simplicity) and thus prohibited from taking full advantage of the input information.

2. Equivalently, for a given input-output performance the choice of representation may alter the computational complexity of the processor.

3. An important factor influencing computational simplicity is the possibility of processing each data item (symbol, predicate, etc.) independently of all the others.

4. To what degree can such independent processing be permitted is determined by the statistical nature of the input environment.

5. The purpose of a good representation is to transform the raw input into a data structure with the least statistical dependence among its various components.

6. In learning situations a good representation is one that minimizes the number of connections and components that need to be readjusted during the learning period.

To what degree can any of these characteristics be generalized, depends on the outcome of additional case studies where shifts in representation produce quantifiable differences. Among the most promising directions we should mention J. Savage's work on Complexity of Coding [Ref. 10] and S. Winograd's work on Complexity of Group Multiplication [Ref. 11]. The latter leads to conclusions similar to those in point 3 above. For instance, the most efficient representation for multiplication is the prime-decomposition number system, as it permits independent digit-by-digit operations.

It is my personal feeling that the most significant characteristics revealed in spectral analysis is number 6 above pertaining to behavior under learning.

The task of finding a good representation for a single isolated problem would probably always demand at least as great an effort as the solution task itself. The reason being that the space of possible representations is much larger than the search space. It is only when one considers a class of problems with some common features (e.g., stationarity) that the search for adequate representations may bear economical fruits. In such cases the available knowledge can be incorporated into an effective form of description language whereby elementary operations on individual symbols do not disturb the invariant properties of the class. The search in such representations can be easily confined to the variety of the class of the situation, avoiding its commonalities.

5. STEPS TOWARD A FORMAL THEORY

Having seen that the question of representation is essentially that of matching between a class of problems and a processor with restricted structure, the next logical step in our program should be to thoroughly investigate the behavior of some well defined processors models in the solution of some well defined problem situations. The narrower the model the more likely we are to find a description of its behavior, but at the same time, the less likely it is for such description to become generic. Pursuing our conviction that general theories are developed on the basis of sets of particular cases, we will outline in this section some possible (and by no means exhaustive) approaches.

We begin by a short description of Minsky and Papert's work on Perceptrons [Ref. 12], and then proceed to outline possible extensions of this work relevant to the problem of representation.

A perceptron (see Figure 4) is a model of a pattern recognition system consisting of three parts: A retina R, a set of predicates Φ, and a linear threshold function ψ. The retina R consists of a spatially arrayed collection of cells whose outputs are "1" or "0" depending on whether the intensity of the incoming light exceeds a certain threshold. Each predicate in the set $\Phi = \{P_1, P_2, \cdots\}$ computes a specified Boolean function of the retinal image (denoted by X). The output $\psi(x)$ of the perceptron is either 1 or 0 depending on whether the inequality

$$\sum_{P \in \Phi} \alpha(P) \cdot P(X) > \theta$$

is satisfied. In the context of human problem-solving, the predicates may correspond to previously established concepts, and the α's represent the

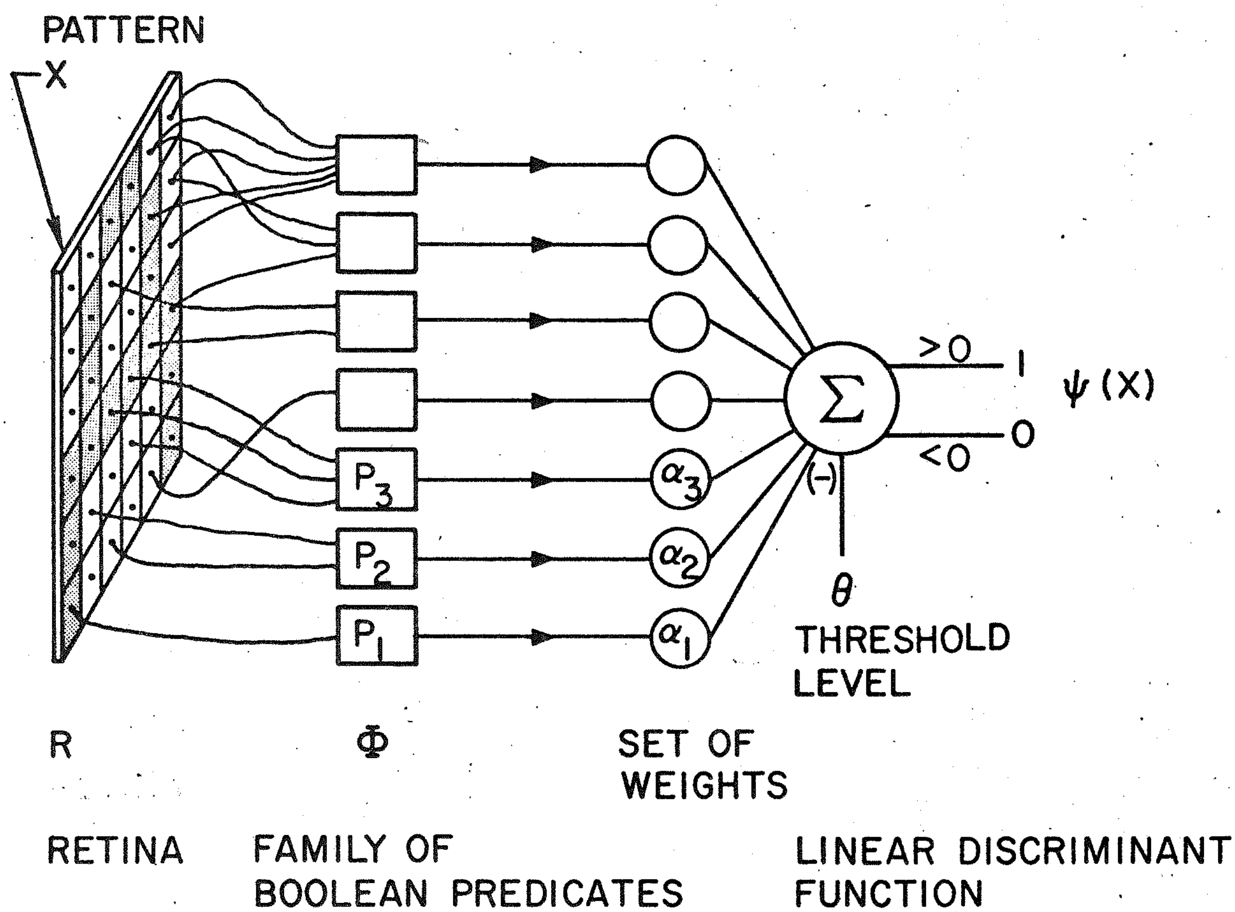

FIGURE 4 : STRUCTURE OF A PERCEPTRON

weights of these concepts in the formation of the new concept $\psi(x)$.

The question originally asked about perceptrons [Ref. 13] was whether such a given network can be 'trained' to recognize a given set of patterns by adjusting the weights $\alpha(P)$ in accordance with past performance. The answer is that if a set of weights exists which properly classifies the given patterns then it can be reached from any starting weights by a training procedure which is guaranteed to converge.

Minsky and Papert attack a different kind of question. They ask: "For a given pattern classification problem, what kind of predicates must be included in the set Φ in order for a working weighting scheme to exist?" Obviously, if one does not limit the content of the predicate set in some way, all patterns are classifiable by a perceptron-like network. It turns out that a certain classification problem requires more 'complex' predicates than others. The measure of predicate complexity used by Minsky and Papert is the number of retinal cells upon which the function P(X) depends. The "order" of a given pattern recognition problem is the minimum number of retinal cells that at least one predicate must 'see' if the network is to do its job.

Minsky and Papert derive many interesting relations between common pattern classification problems and the 'order' they necessitate. For instance, the parity question of whether the number of illuminated cells in a pattern is odd or even cannot be answered unless at least one of the predicates is connected to all the retinal cells. On the other hand, questions like whether a pattern X 'is a solid rectangle', 'is convex', or 'is a hollow square' can be determined with predicates of order no larger than three. The method laid down by Minsky and Papert to derive such relations may serve as a cornerstone for the theory of representation.

Before applying the perceptron model to the problems of representation few extensions are required. The 'order' of a classification problem may be a useful tool for determining whether it is separable by a given family of predicates. It is not however a good measure of classification complexity as it does not account for the number of predicates that must be considered for a solution. The complexity of training procedures is a direct function of the number of weights that need to be adjusted. Therefore, we could benefit a lot from investigating the following kind of problems:

1. Given a pattern classification problem which is solvable by a given family of predicates Φ. What is the minimum number of predicates that must be linearly weighted to make the solution feasible? What properties of Φ are affecting that number?
2. Given a pattern classification problem which is unsolvable by a given family Φ of predicates. How many new logical connections must be made among the members of Φ, if the new set Φ' of predicates is to be sufficient for solution? What properties of Φ affect that number?

Equipped with answers to such questions concerning complexity we can attack the problem of representation ~~using~~ in the following way: Given a pattern classification problem and two isomorphic pattern representations, $\{X\}$ and $\{X'\}$. Which of these representations would result in easier computation given a perceptron-like processor with a set Φ of 'built-in' predicates.

The perceptron is a restricted model. Patterns which are fairly easy to describe in Boolean notation may sometimes require complex perceptron-like networks for realization. The practicality of the perceptron as a pattern classifier is also limited. The weights necessary for classification often require magnitudes and precisions which are practically unrealizable. Why then concentrate on the perceptron as a test vehicle for shifts in representations?

The answer is twofold. First, only for perceptron-like models we now have a calculus connecting complexity of computation to useful geometrical properties, as developed by Minsky and Papert. Second, the perceptron is about the only computational model for which we have economical ~~simple~~ and tractable learning procedures. Less restrictive models of pattern description are discussed by Banerji [Ref. 14]. Banerji models contain the retina and the predicate set but remove the restriction that classification is to be made by linear threshold functions. Instead, the formation of new concepts is accomplished by disjunctive and conjunctive (Boolean 'OR' and 'AND') operations on the available predicates. Pennypacker [Ref. 15] developed algorithms for finding (and learning) the shortest description of a concept using the union of conjunctive predicates. The advantage of this model is that the classification task and the predicate set are expressed in the same language (Boolean expressions). The drawbacks are the lack of recursive learning algorithms like those available for the single-layer perceptron.

The studies of Banerji and Minsky and Papert focus on models with static languages (fixed set of predicates and combination rules thereof) quite a different type of problem arises in studying the dynamics of languages as they adapt to the pattern environment. The problem of representation is then turned inward: Given an environment with fixed representation (corresponding to fixed symbol coding at the input of the processor) what is the proper selection of a language to describe this environment and handle the problems it poses. Here we must first find a meta-language for characterizing the environment independently of the set of languages we investigate.

This might not perhaps be a problem in artificial situations where artificial environments can be constructed having specific set-theoretical and probabilistic properties. Finding a reference description for real environments, however, involves inductive inferences which by their own nature are language dependent.

6. A PRACTICAL APPLICATION OF ISOMORPHIC TRANSFORMATIONS

Several laboratories across the country are engaged in research toward aiding sensory impairments. Various schemes have been proposed for providing visual aid to the deaf, auditory aid to the blind and tactual aid to both, apparently without considerable success.

Let us consider these ideas from the viewpoint of problem representation, and focus for the moment on schemes for enabling the blind to read printed material using tactual stimuli. Such schemes usually employ a transducer which convert the optical character patterns to patterns of vibrations on the reader's finger tips. The reader is to move the transducer along the printed lines and interpret the vibrational pattern he receives on his finger tips.

The question immediately arises: what should the vibrational pattern look like to facilitate fast reading? Now, the structure of the retinal networks is substantially different from that of the tactual nerve system. Without insisting on a detailed knowledge of the two structures one can readily agree that the set of innate predicates used by the retina is different from that of the tactual sense. Our fingers are very good in confirming predicates like 'sharp', 'flat', 'smooth', 'rough' while the eye exhibits its outmost skill in detecting 'lines', 'angles', 'loops', etc. (Much wider differences exist between the visual and auditory systems.) It stands to reason, therefore, that the English alphabet which has evolved to facilitate visual recognition may not be the most suitable representation for tactual recognition. It is not by chance that the Braille alphabet is not a replica of the English alphabet.

What then is a suitable tactual code? and why not use the Braille code which has already been proven useful? The difficulty is transforming English characters to Braille code is that characters must first be recognized then coded, and optical character recognition systems are still complex, expensive, and unreliable. The complexity of these systems is mainly due to the enormous abstraction they comprise; a reduction from the space of all black and white patterns to the space of only twenty six classes. Now, abstraction happened to be the specialty of our brain for millions of years, it can probably continue to outperform machines in this domain. The kind of help we should provide the reading finger is a representation that makes the abstraction simple, taking full advantage of its hardware structure (the innate tactual predicate set which evolved to handle tasks other than character reading). The English letter representation may require many levels of additional interconnections upon this predicate set while with a more suitable isomorphic representation recognition may be achieved mostly by modifying the weights of existing predicates.

The advantage of using isomorphic transformation is that we do not need to guard against the risk of throwing away useful information, which is the main motive behind the complexity of recognition machines. In case of ambiguity, it would always be possible for the reader to consider finer and finer details, until a correct classification is accomplished.

The question still remains, how to go about finding a more effective representation. The only open avenue at present is to try out several transformations and then select the one proven most efficient. With the help of a theory of representation though, it is hoped that we would be able to conduct systematic experiments to reveal some essential characteristics of the sensory hardware, and translate these findings to a set of specifications on the desired transformations.

REFERENCES

1. Minsky, M., "Form and Content in Computer Science," Journal of the ACM, vol. 17, no. 2, April 1970, pp. 197-215.

2. Simon, H. A., "The Sciences of the Artificial," (The M.I.T. Press, Cambridge, Mass. 1969), p. 78.

3. Amarel, S., "On Representations of Problems of Reasoning About Actions," in Machine Intelligence 3, D. Michie, ed., (Edinburgh University Press, 1968).

4. Amarel, S., "On the Representation of Problems and Goal-Directed Procedures for Computers," in Theoretical Approaches to Non-Numerical Problem Solving, R. Banerji and M. D. Mesarovic, ed. (Springer-Verlag, New York, 1970).

5. Pearl, J., "Basis Restricted Transformations and Performance Measures for Spectral Representations," IEEE Trans. on Information Theory, vol. IT-17, no. 6, November 1971, pp. 751-752.

6. Pearl, J., "On the Finite Fourier Transform of Stationary Time-Series," to be published.

7. Root, W. L. and Pitcher, T. S., "On the Fourier-Series Expansion of Random Functions," Annals of Math. Statistics, vol. 26(2), June 1955, pp. 313-318.

8. Cooley, J. W. and Tuckey, J. W., "An Algorithm for the Machine Calculation of Complex Fourier Series," Math. Computation, vol. 19, April 1965, pp. 297-301.

9. Hubel, D. H. and Wiesel, T. N., "Receptive Fields, Binocular Interaction, and Functional Architecture in the Cat's Visual Cortex," Journal of Physiology, vol. 160, 1962, pp. 106-154.

10. Savage, J. E., "Complexity of Decoders: I - Classes of Decoding Rules," IEEE Trans. on Information Theory, IT-15, no. 6, November 1969.

11. Winograd, S., "On the Time required to Perform Multiplication," Journal of ACM, 14 no. 4, October 1969, pp. 793-802.

12. Minsky, M. and Papert, S., "Perceptrons: An Introduction to Computational Geometry," (M.I.T. Press, Cambridge, Mass., 1969).

13. Nillson, N. J., "Learning Machines," (McGraw-Hill, New York, 1965).

14. Banerji, R., "Theory of Problem Solving - An Approach to Artificial Intelligence," (Elsevier, New York, 1969).

15. Pennypacker, J. L., "An Elementary Information Processor for Object Recognition," Rep. No. SRC 30-I-63-1, Case Inst. Technol, Cleveland, Ohio, 1963.