THE UNIVERSITY OF MICHIGAN
COLLEGE OF LITERATURE, SCIENCE, AND THE ARTS
Computer and Communication Sciences Department
Computer Information and Control Engineering
Department of Industrial and Operations Engineering

A NEURAL NETWORK MODEL OF HUMAN PATTERN RECOGNITION

by

Bruce Alan Whitehead

JANUARY 1978

Logic of Computers Group
Computer and Communication Sciences Department
Technical Report No. 209

with assistance from:

National Science Foundation Grant No.
MCS76-04297

THE UNIVERSITY OF MICHIGAN
COLLEGE OF LITERATURE, SCIENCE, AND THE ARTS
Computer and Communication Sciences Department
Computer Information and Control Engineering
Department of Industrial and Operations Engineering

A NEURAL NETWORK MODEL OF HUMAN PATTERN RECOGNITION

by

Bruce Alan Whitehead

JANUARY 1978

Logic of Computers Group
Computer and Communication Sciences Department
Technical Report No. 209

with assistance from:

National Science Foundation Grant No.
MCS76-04297

A NEURAL NETWORK MODEL OF HUMAN PATTERN RECOGNITION

by
Bruce Alan Whitehead

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer & Communication Sciences)
in The University of Michigan
1977

Doctoral Committee:

Professor Stephen Kaplan, Chairman
Professor John H. Holland
Associate Professor Robert G. Pachella
Assistant Professor Arnon S. Rosenthal
Associate Professor Bernard P. Zeigler, The Weizmann Institute

ABSTRACT

A NEURAL NETWORK MODEL OF HUMAN PATTERN RECOGNITION

by
Bruce Alan Whitehead

Chairman: Stephen Kaplan

A mechanism for human pattern recognition must not only detect, it must infer. The pattern recognition system must sample an unreliable, noisy environment in such a way that the information in each sample focuses further sampling upon those features most likely to contribute further significant information. When this cumulated partial information becomes significant, the system must then infer the remaining features to complete the pattern.

Formal modeling theory allows such a mechanism to be expressed in terms of rigorously connected neural and psychological constructs. Briefly, (i) consistency between neural and psychological formulations of the model is shown as a mathematical homomorphism; (ii) simulation of the model yields behavioral predictions; and finally (iii) these predictions are tested (and confirmed) in a human pattern recognition experiment.

The model is based on Hebb's [1949] theory of the neural reverberatory circuit (termed a *cell assembly* or *netlet*) as the fundamental neural unit of cognition. A quantitative transition function for such a circuit is derived from the transition function of individual neurons. The components of the resultant lumped model are thus cell assemblies rather than individual neurons. With this change in scale, neural networks large enough to produce psychological behavior may feasibly be modeled and simulated.

The central hypothesis identifies this neural network mechanism with the psychological construct of *schema* [Attmeave, 1957; Posner and Keele, 1968], the internal representation of an object or concept. Simulation of the neural network structure then predicts a property of the schema not previously investigated. This property concerns the information stored in the schema about its constituent psychological features. The prediction is that, in addition to a central tendency and range for each feature, the schema contains information about the environmental correlations *between* features.

This prediction is tested with human participants in a pattern recognition experiment. By isolating the effect of feature-correlation information, the design differentiates the predictions of the neural network model from those of other currently accepted models. The effect predicted by the neural simulation--in which feature-correlation information influences the pattern recognition decision--is confirmed in the experimental results.

References

Attneave, F. (1957) Transfer of experience with a class-schema to identification-learning of patterns and shapes. Journal of Experimental Psychology 54 81-88.

Hebb, D. O. (1949) The Organization of Behavior: A Neuropsychological Theory, Wiley, New York.

Posner, M.I. and Keele, S.W. (1968) On the genesis of abstract ideas. Journal of Experimental Psychology 77 353-363.

ACKNOWLEDGEMENTS

Thesis-writing is an opportunity to become so engrossed in a network of ideas that the ideas seem to be completely one's own. Acknowledgement-writing, at least for me, has been a way to look outward again, to appreciate how much I have drawn from others. Thoughtful people have supplied not only ideas, but also criticism and suggestions, in each part of my research. All they are receiving in return is a brief mention here, and the promise that I will try to be equally generous and helpful when I have the opportunity.

Nearly five years ago, Stephen Kaplan opened up a new world for me. Through his insight, the findings of neurophysiology, psychology, and human evolution were drawn together rigorously into a theory that actually explains how thought works. I am grateful for his teaching, for his guidance and help, and for his example of what it means to be a responsible scientist in a troubled world.

The inspiration and ideas necessary for deriving the formal model are almost entirely due to Bernard Zeigler. Without his constant interest and encouragement, the mathematical derivation would have been given up long ago.

Arthur Burks and John Holland, directors of the Logic of Computers Group at the University of Michigan, have provided a working environment that a graduate student almost anywhere would envy. Their financial support was provided with the freedom to pursue my own research, and with criticism and guidance that respected my ideas. John Holland also served on my committee, and I have valued his insights into the formalization of biological principles. I also thank Richard Laing for listening to my ideas, providing good advice, and helping me to feel at home in the academic world.

Arnon Rosenthal's contribution went far beyond that usually expected of a committee member. He read the thesis--especially the mathematical derivation--quite thoughtfully, as attested by the quality of the suggestions he provided. Robert Pachella also served conscientiously as a committee member, and I have learned a great deal from his criticism and suggestions.

The experimental design benefitted substantially from Rachel Kaplan's expertise. In carrying out the experiment, Joe Verla's enthusiastic, competent help was greatly appreciated. I am also grateful to Morton Hoffman, who has maintained the Logic computer system superbly. Whenever I have called him (usually nights and weekends) with problems, he has always helped cheerfully.

Any section of the thesis that seems clear and well-stated is likely to have been influenced by Blanche Grosswald. I also owe a debt to the other fellow students who have discussed, argued, and clarified the ideas presented here. In particular, Gary Strong and Mark Weiser have taught me a good share of the valuable things I have learned.

Judith Bonwit and Alice Gantt typed the manuscript, but I would prefer to call them artists than typists. Any reader who examines the mathematical derivation will appreciate the clarity added by Alice Gantt's arrangement of the equations.

I thank my family for many years of support and encouragement. I'm grateful to my mother for the hours she took away from her poetry journal to do some very prosaic proofreading.

With the quality of help I have received, almost any error or shortcoming in the dissertation could have been avoided. If it wasn't, the responsibility is mine.

In the acknowledgements above, I have left out one ingredient that is essential, not only to a thesis, but to any human enterprise. This is the personal warmth that has come from each person listed above and from the many other generous people I have known in graduate school.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS ii

LIST OF TABLES vi

LIST OF FIGURES vii

LIST OF APPENDICES viii

CHAPTER

1. INTRODUCTION 1

2. INTUITIVE DESCRIPTION OF THE MODELING TECHNIQUE 7

3. FORMAL SPECIFICATION OF THE BASE MODEL 18

   Assumptions defining the structure of the base model
   Stochastic properties of the base model

4. DERIVATION OF THE LUMPED MODEL 32

   Derivation of the transition function
   Formal specification of the lumped model

5. FUNCTIONAL INTERPRETATION OF THE LUMPED MODEL 47

   Psychological constraints
   Pilot simulations

6. HUMAN PATTERN RECOGNITION: AN EMPIRICAL DEMONSTRATION. . 55

   Stimuli and procedures
   Predictions obtained from the netlet simulation
   Results and discussion

7. CONCLUSIONS 96

APPENDIX 100

REFERENCES 121

LIST OF TABLES

1. The nine feature dimensions of the Hatfield-McCoy recognition task. 63

2. Connections in the lumped neural model of the pattern recognition task. 77

3. Analysis of variance for the experimental results 91

LIST OF FIGURES

1. Neural stimulus equivalencing. 10

2. How a neural hypothesis can be tested in a psychological experiment 15

3. Using formal modeling techniques to constrain the theory . . . 17

4. Expressing the distribution of input over a netlet as a random variable. 30

5. Variables used to calculate the new activity in a netlet . . . 38

6. Input-output functions for an individual neuron and for a netlet 40

7. The structure of a sample network in the lumped model. 44

8. The feature-independence hypothesis. 54

9. Training stimuli for participant group III 62

10. Definition of schemata for participant group III 65

11. The Hatfield-A, McCoy-B correlation pattern. 68

12. The Hatfield-B, McCoy-A correlation pattern. 70

13. Factorial design of the experiment 71

14. Neural network model used in the simulation for the Hatfield-A, McCoy-B condition. 75

15. Experimental results 90

LIST OF APPENDICES

Derivation of Lemmas 2 and 3. 100

Derivation of Lemmas 4, 6, and 7. 102

Derivation of the COVARIANCE component of the transition function. . . . . . , ... 109

Generation of Correlated Random Variables 119

CHAPTER 1

INTRODUCTION

How can a system of simple, rule-governed components exhibit intelligent behavior? To answer this question for electronic components, one must design such a system -- the goal of artificial intelligence research. To answer the question for neural components, one must understand a design that evolution has produced. Our central concern here will be to understand the existing neural design, but such a design is bound to appear quite arbitrary unless we know the problems that the design is solving.

Of course, when a problem is solved, it's not a problem any more. In that sense, a successful design covers its own tracks, and it is difficult to see the problems that the design is solving. Designs produced by evolution are often prime examples of this. After millions of years of fine tuning, such designs may operate so gracefully and effortlessly that one is deceived into thinking that easy problems are being solved. For example, perceiving the objects right in front of you does not seem to be a problem at all -- there they are. Only the experience of surgical patients who have been blind since birth and see for the first time as adults [reviewed by Hebb, 1949] convinces people that ordinary perception is a sophisticated process that takes months to learn.

Observing a successful design at work may obscure the problem being solved, but by the same token trying to create such a design yourself may make the problems painfully obvious. "Anything looks easy until you try it", and artificial intelligence researchers have tried. Their experience, gleaned from long hard effort, has given artificial intelligence (AI) research two qualities:

(i) Skepticism for any claims not accompanied by working models; and

(ii) Appreciation of the fundamental problems that must be solved by any design for intelligence.

The first gives AI a type of "show me" rigor that is sometimes difficult to attain in cognitive psychology -- though we will try to attain it here. The usefulness of the second quality to psychologists depends upon a conviction that certain inescapable problems must be solved in order for any system, neural or electronic, to be intelligent. Although the solutions to the problems may be very different in the two cases, the problems themselves should be the same.

This conviction rests upon the observation that the two fields, cognitive psychology and AI -- employing different assumptions, different methods, and different sources of data -- are nevertheless converging to the same list of fundamental problems. First let us hear from Patrick Winston of MIT, a leading spokesman for the AI community. Winston's recent [1977] book identifies the crucial problems as knowledge representation, search, constraint exploitation, and control mechanisms. *Knolwedge representation* is the question of how to code knowledge into the system in such a way that any piece of this knowledge can be efficiently accessed when it is needed. The organization or structure of knowledge as coded in the system is termed a knowledge base. For a human or computer attempting to make a living in the world, the knowledge to be coded concerns the physical and cultural environment in which it operates.

The other fundamental problems listed by Winston all arise because, out of the total knowledge represented in an intelligent system, only a small part is appropriate for guiding the system's behavior at any time. Thus, in any particular situation, cues about that situation must somehow guide a search through the knowledge base to find knowledge appropriate to the situation. Winston divides this task into three problem areas: design of the *search process* in general, *exploitation of constraints* from the knowledge base to narrow down the search, and *control mechanisms* which evaluate the type of search needed and adjust the search process accordingly.

Cognitive psychologists, relying on completely different sources of data, have come to quite similar conclusions about the basic problems that must be solved in the design of an intelligent system. The idea that the operation of the brain might be explicable in terms of knowledge representation and search appears as early as the work of William James [1892]. James conceives of the knowledge base as an associative network:

> In mental terms, the more other facts a fact is associated with in the mind, the better possession of it our memory retains. Each of its associates becomes a hook to which it hangs, a means to fish it up by when sunk beneath the surface. Together, they form a network of attachments by which it is woven into the entire tissue of our thought. [p. 161]

He also considers the problem of search through this network:

> In short, we make search in our memory for a forgotten idea, just as we rummage our house for a lost object. In both cases we visit what seems to us the probable neighborhood of that which

> we miss. We turn over the things under which, or within which, or alongside of which, it may possibly be; and if it lies near them, it soon comes to view. But these matters, in the case of a mental object sought, are nothing more than its associates. The machinery of recall is thus the same as the machinery of association, and the machinery of association, as we know, is nothing but the elementary law of habit in the nerve-centers. [p. 157]

Even the particular questions of how the search is constrained by the content of the knowledge base, and how it is guided by control mechanisms, are considered in James' chapters "Association" and "Memory" respectively.

The attempt to understand the human information processing system in terms of the fundamental problems it is solving is even more explicit in present-day cognitive psychology. Michael Posner, for example, has written a textbook [1973] divided into two sections, statics and dynamics of cognition. The statics section is concerned with the representation of knowledge, with a chapter "Representation in Memory" and others investigating mental structures to represent particular kinds of information. The dynamics section, on the other hand, addresses the use of this knowledge to guide behavior. One chapter is devoted to search strategies, with a subsection showing how the representational structure constrains the search. Another chapter investigates control mechanisms (such as conscious attention) which adjust processing to fit particular situations and focus it on particular parts of the knowledge base.

We have seen that artificial intelligence and cognitive psychology are close to agreement on the fundamental problems that must be solved in the design of an intelligent system: Knowledge representation, search, constraint exploitation, and control mechanisms. These two fields also share a common conceptual framework in which these problems can be carefully stated. This framework seeks to describe the system in terms of the knowledge being represented, independently of the physical structures in which the knowledge is coded. The relations among different items of knowledge defines the *structure* of the knowledge base, while the operations performed on items of knowledge define how the knowledge base is *processed.* This conceptual framework provides a language for explaining how the fundamental problems described above may be solved in the physical design of an intelligent system. Thus, the knowledge representation problem should be answered in terms of the knowledge base structure, which is in turn reducible to the physical structure of the system. The other three problems -- search, constraint exploitation, and control -- should be answered in terms of knowledge base processing, which is in turn reducible to the physical transition function of the system.

The central concern of this dissertation will be the knowledge representation problem. In attempting to show how the design of the brain solves this problem, however, we will consider not only the structure of the knowledge base, but also the information processing functions that such a structure must be able to perform. Since the design of the brain is ultimately neural, our approach will be to make a rigorous connection between two different descriptions of the human information processing system: a description in terms of knowledge base structure and processing, and a description in terms of neural structure and processing.

Our proposed mechanism will be based upon a theory put forth by Hebb [1949] and further developed by Milner [1957] and Kaplan [1973], in which each object or concept in a person's knowledge base is represented by a neural reverberatory circuit. Associations between various objects or concepts in the knowledge base are then coded as excitatory connections between the corresponding neural circuits. While Hebb, Milner, and Kaplan have each argued that human intelligence can be explained in these terms, their arguments have been mainly verbal. Our specific task here will be to develop a mathematically rigorous technique for connecting theories of what the brain is accomplishing (in terms of knowledge base structure and processing) with theories of how the brain works (in terms of neural structure and processing). Chapters 2, 3, and 4 will develop this technique in general, for any hypothesis in which the components of the knowledge base are neural reverberatory circuits. Chapters 5 and 6 will then apply this technique to a particular pattern recognition task, yielding model predictions which are empirically tested.

CHAPTER 2

INTUITIVE DESCRIPTION OF THE MODELING TECHNIQUE

> How do we recognize the identity of the features of a man, whether we see him in profile, in three-quarters face, or in full face? How do we recognize a circle as a circle, whether it is large or small, near or far; whether, in fact, it is in a plane perpendicular to a line from the eye meeting it in the middle, and is seen as a circle, or has some other orientation, and is seen as an ellipse? How do we see faces and animals and maps in the clouds, or in the blots of a Rorschach test? All these examples refer to the eye, but similar problems extend to the other senses, and some of them have to do with intersensory relations. How do we put into words the call of a bird or the stridulations of an insect? How do we identify the roundness of a coin by touch? [Wiener, 1948]

> A polyp would be a conceptual thinker if a feeling of 'Hollo! thingumbob again!' ever flitted through its mind. [James, 1892]

Wiener and James have described an essential characteristic of intelligence -- the ability to recognize the same object or concept even if its detectable features differ from one instance to the next. Psychologists often refer to this processing as stimulus equivalence, emphasizing the equvalence-classing operation which maps many particular stimuli into the same generic object or concept. Computer scientists usually refer to this process as pattern recognition.[1]

The reason that stimulus equivalence or pattern recognition is essential to intelligence is well explained by Anderson [1977]:

[1] Pattern classification is perhaps a better though less common term for the case considered here (and in most of the literature) in which the set of features and the set of categories or equivalence classes remain constant throughout the process. Feature integration is an equivalent psychological term for this case.

> Situations rarely repeat exactly. One of the most useful functions of a nervous system would be to be able to learn from varying and noisy experience. In particular, the system might seek to reduce the immense number of possible inputs into a relatively few equivalence classes, which can be manageably associated with learned responses: this is food, this is danger, this is a friend.

In other words, this equivalence classing is crucial if behavior in new situations is to be based on experience -- experience similar to, but not identical with the new situation. Additionally, equivalence classing vastly reduces the amount of information that must be stored in the knowledge base, since

(i) only generic objects and concepts need be represented in the knowledge base structure; provided that

(ii) the processing associated with the knowledge base performs the desired equivalence classing, i.e. each particular stimulus is mapped into the appropriate generic representation.

These suggestions about knowledge base structure and processing are so far based on functional arguments: how the knowledge base ought to be organized for intelligent behavior in an unreliable environment. There is also psychological evidence, to be reviewed in Chapter 5, that human knowledge is organized in this manner.

The goal of our modeling enterprise is to develop the suggestions above into a hypothesis that not only agrees with functional arguments and psychological data, but is also solidly based on the properties of individual neurons.

Hypothesized neural mechanisms which operate on the retinal image without analyzing its content (size normalization, for example) are not in themselves sufficient to account for human pattern recognition. Neither is the hypothesis that an object can be recognized on the basis of a set of essential or critical features which are always reliable indicators of the object's presence. As Kaplan [1976] points out, the environment is just not that cooperative. Both the environment in which human information processing capability evolved and the present human environment are characterized by uncertainty and by the need to make reasonable inferences on the basis of partial information. Not only must an object or concept be recognized from unreliable features, but the neural activity corresponding to a single feature is variable in itself, due to the stochastic nature of individual neurons.

Human pattern recognition therefore requires an equivalencing mechanism: a mechanism by which the different neural activities that signify a feature become equivalent at some level of processing, and in turn a mechanism by which different subsets of features that indicate the same object become equivalent at a deeper level of processing. This is precisely the function of Hebb's cell assembly mechanism (Fig. 1). In Hebb's mechanism, the different neural activity patterns that signify a given feature are all part of the same reverberatory circuit, and once the activation of this circuit is sufficient to achieve self-sustaining reverberation, it does not matter which particular activity pattern started the process. (The reverberatory circuits which detect sensory features are shown as solid circles in Figure 1.) Detectors for the different features of an object are in turn connected together into a network; and in different instances, different subsets of the features will activate the network. (Depending upon which feature detectors are active and

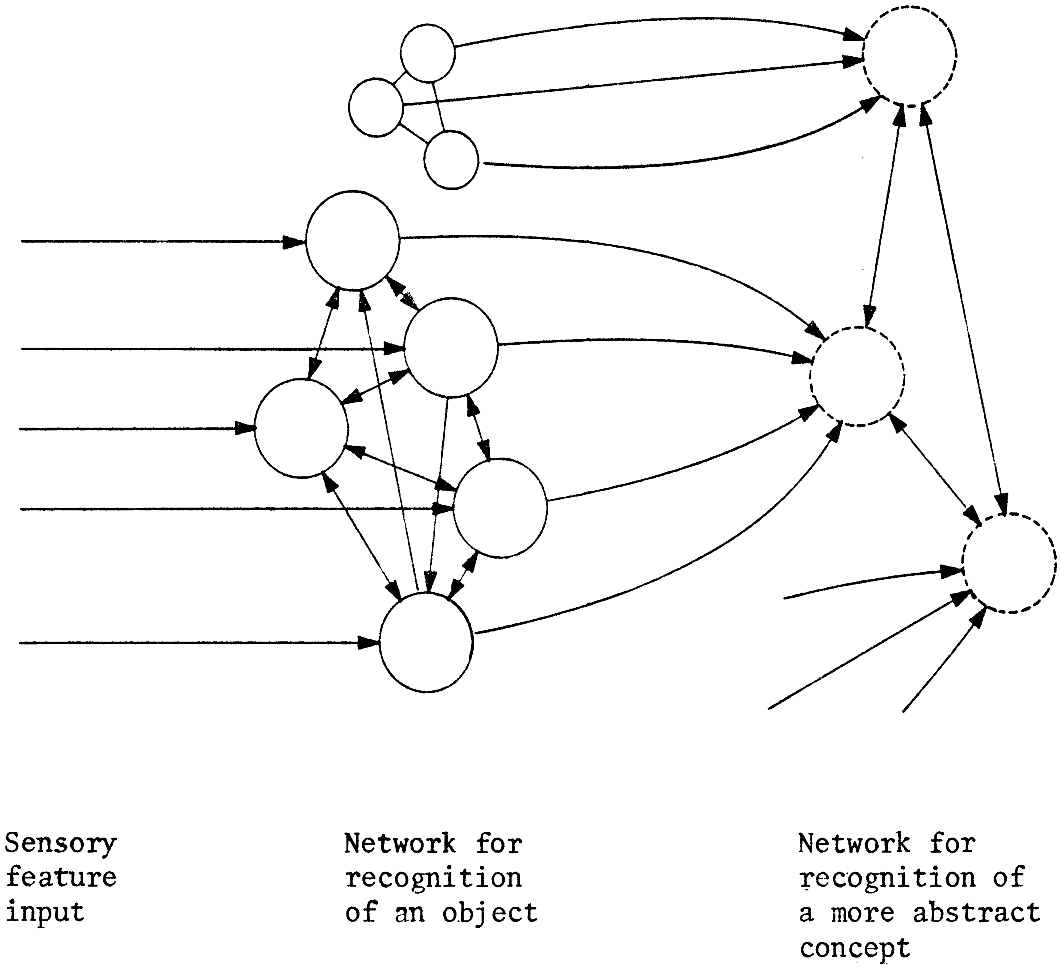


FIGURE 1. Neural stimulus equivalencing. Each circle represents a netlet (a strongly interconnected reverberatory circuit).

how strongly they are connected to the others, the activation of undetected features may range from lowering their threshold, in search of environmental support, to "filling in" such features even **without** environmental support.) If the interaction of feature elements for a given object is sufficient to produce sustained activity in the solid-circle network, then the object is recognized.

This network representation of an object may itself become a feature in the recognition of more abstract patterns, as shown in the dotted-circle network. (The dotted circles are reverberatory circuits just like the solid circles, except that the features they detect are not sensory features, but rather the more abstract features that arise from the first level of processing.) We will use the term *netlet* [Harth, Csermely, Beck, and Lindsay, 1970] to refer to such a reverberatory circuit, which detects a sensory or abstract feature. Our netlet is synonymous with Hebb's *cell assembly*.

This mechanism may be iterated to further levels of processing, where the activity in a pattern-recognition network at one level is detected as a single feature at some deeper level. Milner [1974] has shown how such a mechanism might begin with the elementary feature detectors (such as line and edge detectors) found by Hubel and Wiesel [1962].

Figure 1 should not be taken to mean that netlets are spatially localized in the brain. Anderson [in press, 1977] has shown mathematically that even if netlets overlap considerably, so that an individual neuron may belong to more than one netlet, the netlets can still be "tuned" so as to remain functionally separate units.

Anderson shows that this tuning will automatically result from a simple contiguity learning rule, similar to that proposed by Hebb.

Anderson also points out that any model which involves *distributed processing*, in which the functional units are not spatially distinct, would be hard to test by neurophysiological methods. A netlet spread out over a volume of cortex, tangled up with and even sharing neurons with other netlets, would be difficult to find, let alone do experiments on. Anderson therefore suggests that "the verification of some of these physiologically based distributed systems may be through the properties of the global system; restated, in terms of the 'psychology' of the whole system."

We are therefore faced with the problem of generating psychologically testable predictions from a neural model. Unfortunately, psychological behavior is, as Anderson points out, a global property of the system, depending upon the complex interaction of a huge number of neurons. So a neural model capable of generating psychological predictions would have to be a pretty big model. Now of course computers are big and powerful, and in principle capable of simulating an arbitrarily large neural model -- as long as we specify every detail precisely. But in practice, a theory of the type we are considering is a conceptual framework, focusing on underlying principles but leaving many details unspecified. Such a conceptual framework is really a whole class of models with the same structure but with different parameter settings. When we ask whether the conceptual framework is sufficient to account for the observable properties of human pattern recognition, we are really asking whether there exists some particular combination of parameter settings for which the model will generate the expected behavior.

In this light, the parameter-adjusting phase of a simulation is essentially a search process. The set to be searched is the set of all reasonable parameter settings, a set which grows exponentially with the number of parameters. Moreover, expectations for the model usually cannot be formalized rigorously enough to let the computer do the search alone. In such cases, human judgment is a necessary part of the search algorithm.

There is thus an upper bound on the number of parameters one can work with in a manageable simulation, and hard experience [notably that of Finley, 1967] has confirmed that this upper bound is not large. It is true that human judgment can substantially improve the efficiency of the search process, but only if one can understand what the model is doing and interpret the results. If the model isn't behaving as expected, one must be able to follow its inner workings closely enough to figure out what went wrong and what changes are likely to fix it. Thus we are faced with a definite limitation on the size and number of parameters we can work with in any single model. Yet we wish to model a neural system complex enough to generate psychologically observable behavior.

The solution is to realize that all our knowledge about a system does not have to be represented in one big model which tries to explain everything. Instead, we need a technique by which several models, each emphasizing some particular aspect of the system, can be fit together into a consistent structure. This is exactly the technique provided by Zeigler's [1976] theory of modeling. Using Zeigler's conceptual framework, we will specify a *base model*, whose components are neurons, and a *lumped model*, whose components are netlets. The base model will formalize our hypotheses about how the characteristics of the netlet structure can be derived from the characteristics of the individual neurons that comprise it. The lumped model will in turn formalize our hypotheses about how the characteristics of human behavior (in particular, human pattern recognition) result from this netlet structure.

But these two models are not independent; they are tightly connected by the relationship of a system homomorphism. This homomorphism means that (i) there is a correspondence between the state spaces and transition functions of the two models; and (ii) any simulation of the lumped model will follow the same trajectory as if the corresponding base model had been simulated, and the resultant base model states translated into their equivalent lumped model states. This homomorphism is the link which allows hypotheses about neural structure to be tested against psychological observations, as in Figure 2.

Figure 2 might also be considered as a diagram of the constraints we can bring to bear on a theory of human pattern recognition. If psychological data alone is taken as a constraint, there are many plausible theories of human pattern recognition. In the same vein, if neurophysiological data is taken as the sole constraint, there is another class of theories that all fit the data. But if we can bring both sets of constraints to bear at the same

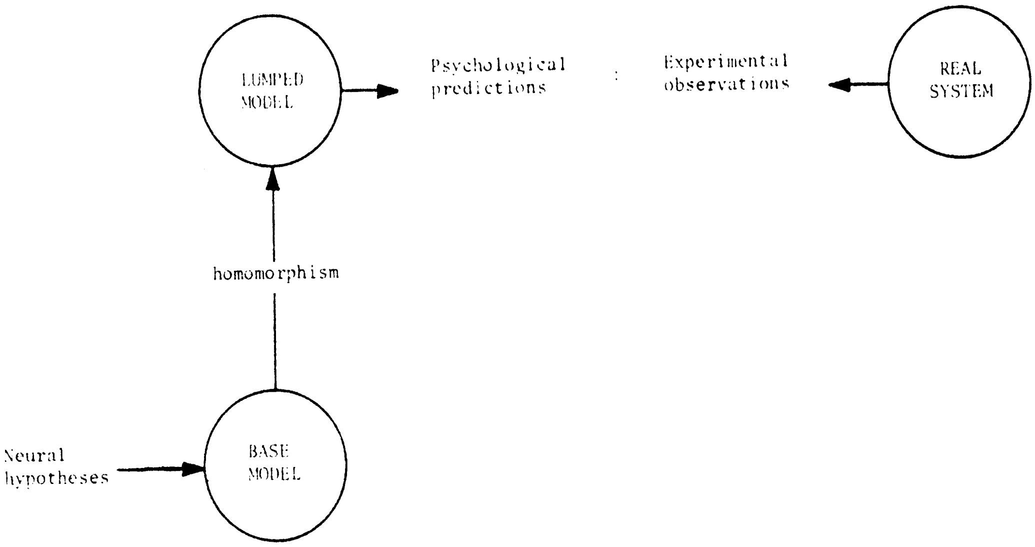


FIGURE 2. How a neural hypothesis can be tested in a psychological experiment.

time, the range of plausible theories should be drastically reduced. Thus, a more complete picture of our modeling enterprise should look like Figure 3. For our purposes, "properties of neurons" are the transition functions of individual neurons and our hypotheses about the randomness of neural interconnection, both formally incorporated in the base model. But the lumped model is also forced to be consistent with these hypotheses, via the homomorphism. "Properties of human pattern recognition", on the other hand, refer to the results of previous psychological experiments (reviewed in Chapter 5) and to evolutionary arguments [Kaplan, 1976] concerning the information processing demands which our perceptual system evolved to meet. Chapter 5 explains how these constraints are incorporated into the lumped model, as the hypothesized connection pattern of the netlet system. But of course netlets are connected only insofar as their constituent neurons are connected, so the base model is constrained as well. In short, we will define a formal correspondence from the parameters and state variables of the base model to the parameters and state variables of the lumped model, so that constraints on either model can be interpreted as constraints on the other.

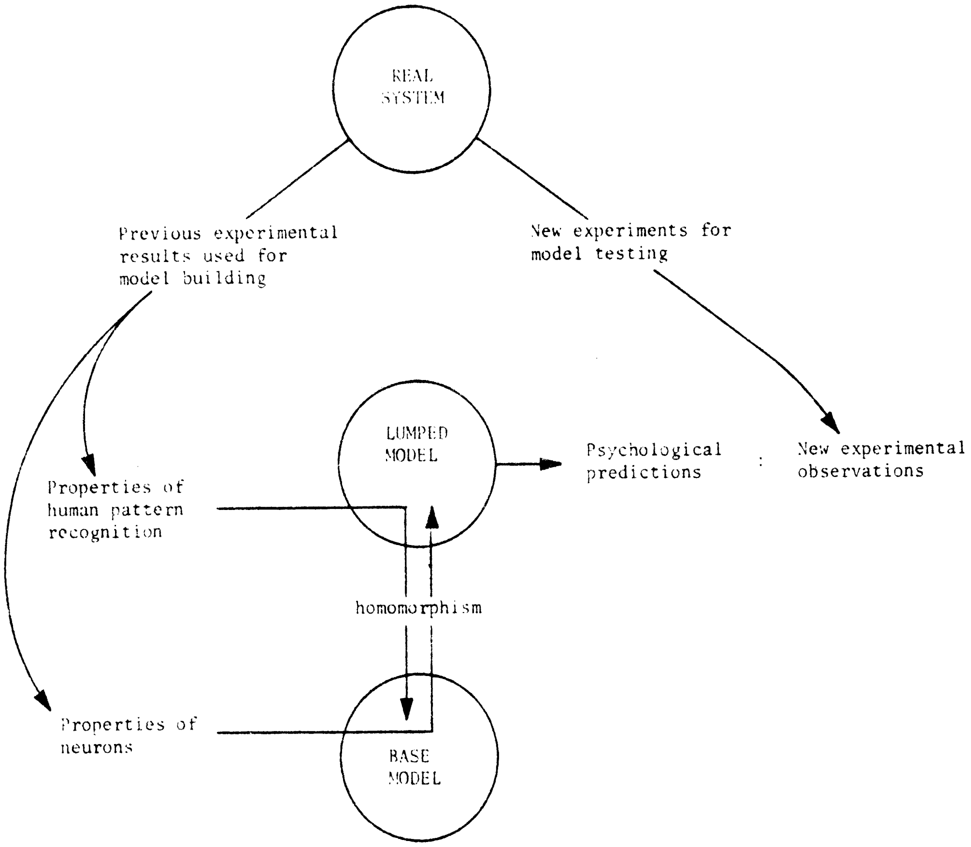

REAL SYSTEM
Previous experimental results used for model building
New experiments for model testing
LUMPED MODEL
Psychological predictions
:
New experimental observations
Properties of human pattern recognition
homomorphism
Properties of neurons
BASE MODEL

FIGURE 3. Using formal modeling techniques, our theory will be constrained to be consistent with both neural and psychological data.

CHAPTER 3

FORMAL SPECIFICATION OF THE BASE MODEL

Our lumped model will be derived from and shown equivalent to a base model whose components are individual neurons connected in a network. The base model can be formally described as a structured automaton [Zeigler, 1972] $M = \langle Q,\delta\rangle$ such that $Q = \times_\beta Q_\beta$, where each Q_β is a nonnegative integer representing the recovery state (time since last firing) of neuron β. The global transition function $\delta: Q \rightarrow Q$ is specified by a set of local transition functions δ_β, where each δ_β specifies the next state of neuron β according to whether a weighted sum of inputs from other neurons exceeds the threshold of neuron β. If we let I_β denote the set of all input neighbors to β, then the transition function $\delta_\beta: Q_\beta \times \{Q_\alpha | \alpha \in I_\beta\} \rightarrow Q_\beta$ is given by

$$\delta_\beta(q_\beta,\{q_\alpha|\alpha\in I_\beta\}) = \begin{cases} 0 \text{ if } \Sigma_\alpha s_{\alpha\beta}\cdot\lambda(q_\alpha) \geq T_\beta \\ q_\beta + 1 \text{ otherwise} \end{cases} \tag{1}$$

where the summation is over all $\alpha \in I_\beta$; $s_{\alpha\beta}$ is the weight (synapse strength) applied to input from neuron α, T_β is the threshold of neuron β; and λ is the output function for any neuron, defined as follows:

$\lambda(q_\beta) = 1$ if $q_\beta = 0$ (i.e. if β is firing); $\lambda(q_\beta) = 0$ otherwise.

Once we develop the defining assumptions of our netlet model, the neurons $\alpha \in I_\beta$ will be indexed according to netlet of origin. The definition of the transition function δ_β will be re-expressed in terms of this new indexing scheme following the statement of Conditions 1-6.

Assumptions defining the structure of the base model

Harth et al. [1970] review anatomical and physiological evidence for the existence of more or less discrete populations of interconnected neurons, termed netlets. A similar aggregated unit has been modeled by Zeigler [1975] and Anderson [1972; in press 1977]. All the assumptions about netlets upon which our analysis is based will be stated here, as Conditions 1-6.

Summary of Conditions. Consider all synaptic connections from a netlet B_j to a netlet B_k. Condition 1 requires the presynaptic neurons of these synapses to constitute a random sample of B_j. Condition 2, on the other hand, requires the distribution of the corresponding postsynaptic neurons in B_k to be independent of the distributions of postsynaptic neurons receiving input from other netlets. Condition 3 specifies the strengths of the synapses, and may be considered a refinement of the approximation made in either Anderson's [1972] or Zeigler's [1975] model, in the sense that both Anderson and Zeigler employ the following "equal influence" condition: If any neuron B_k receives input from B_j, then *each* neuron in B_k must receive input through one or more synapses of the same total strength from B_j. That is, either Anderson's or Zeigler's conditions would require the strength parameters $\mu_{j\beta}$ and $\nu_{j\beta}$ (as defined in Condition 3 below) to be constant over all neurons β in B_k, while we only require connections to a subset B_{jk} of B_k, holding $\mu_{j\beta}$ and $\nu_{j\beta}$ constant over this subset. Wong and Harth, while avoiding the "equal influence" assumption, accomplish the same effect (that of avoiding inhomogeneity in the input) by unrealistically requiring that at each time step there must be a different random set of connections from B_j to B_k. Part (ii) of Condition 3 may also be considered a refinement of Anderson's linearity assumption. Anderson's assumption 3 is equivalent to approximating the integral of f over its entire domain by a single linear function, while we use a linear approximation only over small intervals of f, resulting in a piecewise linear approximation.

Condition 4 defines the threshold of a neuron β as a stochastic variable T_β. This threshold undergoes random perturbations, but its expected value decreases with time, via exponential decay. Condition 5 assumes that we can ignore two types of second order differences. Condition 6 specifies independence requirements on the distributions of initial recovery states, incoming synapses, and outgoing synapses, and assumes that we can use the law of large numbers.

The intent of Conditions 1-3 and 6 may become clearer if we intuitively define the macrostate of the system to be the statistical information which specifies, for each netlet, the proportion of its neurons in each recovery state, without specifying exactly which neuron is in which state. Similarly, the macrostructure of the system specifies the total effective connection strength between any two netlets, without specifying which neurons are involved in the connections. The microstate and microstructure, on the other hand, specify the recovery state of each individual neuron, and the strength of each individual synapse. Thus, the same macrostate and macrostructure could be realized in many different microstates and microstructures. The intent of the conditions above is to allow the modeler to choose freely the macrostructure and initial macrostate, but that within these constraints the microstructure and initial microstate will be random, and independent of each other. With these initial conditions, it will be possible to calculate the effects of the microstate distribution at later time steps.

Even though the initial microstate is random and independent of the microstructure, it will not stay that way for long. After all, the signals coming through the input synapses to a netlet directly affect the states of the receiving neurons in the netlet. So it should not be surprising that the distribution of neural states over the netlet soon becomes correlated with the distribution of incoming synapses. The randomness of the initial state, however, together with the other assumptions and approximations of Conditions 1-6, will allow us to calculate the effect of such correlations on the subsequent macrostates of the network. The fact that we explicitly calculate these effects of microstructure, rather than assuming them to be negligible, is the major difference between our approach and those of Wong and Harth [1973] or Zeigler [1975].

We will now state Conditions 1-6 in detail. The neurons in the base model are divided into a set of netlets $\{B_k | k \in D'\}$. Throughout this and the next chapter, the indices j, k, and ℓ will range over D' (i.e. they will index netlets), while β will index the neurons in some specific netlet. Lower case n will index multiple input neighbors to a given neuron from a given netlet, as specified in Condition 1. Unless otherwise specified, it is understood that summations are over all values of indices for which the summated term is defined, and that all free indices are universally quantified in the same way.

Definitions. The following symbols, defined more fully within Conditions 1-6, are summarized here for quick reference.

$I_{nj\beta}$: The neurons in netlet B_j having input connections to a particular neuron β are denoted as the set $\{I_{nj\beta}, n = 1,\ldots,N_{j\beta}\}$. The counter n thus distinguishes different inputs to β from netlet j.

$s_{nj\beta}$: Denotes the strength of the synpase onto neuron β from its particular input neighbor $I_{nj\beta}$.

$\mu_{j\beta}$: The sum of synaptic strengths from neurons in B_j to neuron β. $\mu_{j\beta} = \Sigma s_{nj\beta}$ for $n = 1,\ldots,N_{j\beta}$.

$\nu_{j\beta}$: The sum of *squared* synaptic strengths from neurons in B_j to neuron β. $\nu_{j\beta} = \Sigma s_{nj\beta}^2$ for $n = 1,\ldots,N_{j\beta}$.

B_{jk}: Denotes the subset of neurons in B_k which receive input from B_j.

μ_{jk}: The constant value of $\mu_{j\beta}$ over all neurons β in B_{jk}.

ν_{jk}: The constant value of $\nu_{j\beta}$ over all neurons β in B_{jk}.

T_β: The threshold of neuron β, containing a noise component ξ_β that varies randomly at each time step.

$\Phi_{a,\nu}$: A cumulative distribution function, determined by two parameters (*recovery parameter* a and *variance parameter* ν), as defined in Condition 4.

Condition 1. Each neuron β in B_k may have an arbitrary number $N_{j\beta} \geq 0$ of input neighbors from each netlet B_j. These neighbors are chosen randomly (with replacement) from B_j, so that each neuron β in B_k receives an independent random sample of input neighbors from B_j. More precisely, the input neighbors to neuron β from netlet B_j are given by $I_{nj\beta}$, $n = 1,\ldots,N_{j\beta}$, where if we set $N = \max\{N_{j\beta} | j,k \in D', \beta \in B_k\}$, then $\{I_{nj\beta}, n = 1,\ldots,N\}$ is a set of independent random variables, each with range B_j and each uniformly distributed over this range. (Note that only if $1 \leq n \leq N_{j\beta}$ does $I_{nj\beta}$ index an actual input neighbor in B_j; for $N_{j\beta} < n \leq N$, $I_{nj\beta}$ is introduced only as a notational convenience and will be assigned a synapse value of 0.)

Condition 2. Let $B_{jk} \subseteq B_k$ denote the subset of neurons in netlet B_k which receive input from netlet B_j, i.e. $B_{jk} = \{\beta \in B_k | N_{j\beta} > 0\}$. Then we require all such subsets B_{jk} of B_k to be chosen independently. In other words, if B_k is considered as a discrete sample space (in which each neuron $\beta \in B_k$ is a sample point) then the subsets B_{jk}, $j \in D'$ are mutually independent. Each subset B_{jk} may have arbitrary measure b_{jk} in B_k, $0 \leq b_{jk} \leq 1$. Note that the case where $j = k$ (representing feedback of a netlet to itself) is treated no differently; we allow an arbitrary fraction b_{kk} of the neurons in B_k to receive input from within the netlet.

Condition 3. For each input neighbor $I_{nj\beta}$ to neuron β, there is a corresponding synapse strength $s_{nj\beta}$. (In neural terms, $s_{nj\beta}$ represents the excitatory post-synaptic membrane potential in neuron β caused by the firing of the presynaptic neuron $I_{nj\beta}$.) Each neuron $\beta \in B_{jk}$ which does receive input from netlet j may receive this input through one or more synapses of different strengths. Let $\mu_{j\beta}$ denote the sum of all synaptic connection strengths impinging upon β from B_j; and let $\nu_{j\beta}$ be the sum of their squares:

$$\mu_{j\beta} = \Sigma_n s_{nj\beta} \quad \text{and} \quad \nu_{j\beta} = \Sigma_n s^2_{nj\beta} \tag{2}$$

We then require that each of $\mu_{j\beta}$ and $\nu_{j\beta}$ be constant over all β in B_{jk}. The constant value of $\mu_{j\beta}$ over all $\beta \in B_{jk}$ will be denoted μ_{jk}; similarly, the constant value of $\nu_{j\beta}$ over the same subset will be denoted ν_{jk}. (The parameters μ_{jk} and ν_{jk} of the connection pattern will be used in calculating the mean and variance of the activity coming through the connections.) We further require that the influence μ_{jk} of any single netlet B_j on B_k be small relative to the total influence $\Sigma_j \mu_{jk}$ of all netlets having input to B_k; small enough to make these two approximations: (i) the probability density f of the sum $\xi_\beta - \Sigma_j X_{jk}$ of independent random variables can be approximated normally, where ξ_β is the normal variate defined in Condition 4, and each X_{jk} (defined in section 3 and shown independent in Lemma 6) ranges over an interval of length μ_{jk}. (ii) the integral of f over an interval of length L μ_{jk} can be approximated by the area $L \cdot f(x)$, where x is in the interval.

Condition 4. All neurons are identical except for their thresholds, which fluctuate due to random noise at each time step. If neuron β is in state q_β (where $q_\beta = 0$ when the neuron fires, and increases by 1 each time step the neuron does not fire), then its threshold will be denoted by a random variable T_β. Now let $N_\beta(x,\nu)$ denote the normal cumulative distribution function with mean 0, variance ν, and argument x. (The subscript β is used to emphasize that the thresholds for different neurons β will be drawn from independent distributions.) We now specify that $P[T_\beta < x] = \Phi_{a,\nu}(x)$--that is, each T_β has a cumulative distribution function $\Phi_{a,\nu}$: Reals → [0,1], where the 2-parameter distribution $\Phi_{a,\nu}$ is defined in general as

$$\Phi_{a,\nu}(x) = aN_\beta(x-\tau_0,\nu) + (1-a)N_\beta(x-\tau_\infty,\nu)$$

where $a = m^{q_\beta}$ depends on the recovery state q_β of the neuron, and m, τ_0, τ_∞, and ν are constants, $m < 1$, $\tau_0 > \tau_\infty$. We will refer to a as the *recovery parameter* of the distribution and ν as the *variance parameter*. In the expression above, $N_\beta(x,\nu) = P[\xi_\beta < x]$ where $\{\xi_\beta\}$

is a set of independent identically distributed random variables, each normally distributed with mean 0 and variance ν. Each ξ_β represents a random noise component of the threshold of neuron β, and is unaffected by recovery state or input. From the definition of $\Phi_{a,\nu}$ above it is clear that, given a value of the recovery parameter a, the expected threshold $E(T_\beta)$ is $a\tau_0 + (1-a)\tau_\infty$. Since the recovery parameter is an exponentially decaying function m^{q_β} of the time since last firing, we see that the expected threshold is τ_0 for a neuron which has just fired, and exponentially decays towards τ_∞ as the neuron recovers.

Condition 5. In the next section below, we will define a random variable X_{jk} to represent the input from netlet B_j to netlet B_k, or more precisely the distribution of this input over netlet B_k. We will also define a random variable T_k to represent the distribution of neural thresholds over netlet B_k. If we let X_{jk} and X^δ_{jk} represent the distributions of input at two successive time steps, then we will make the following two approximations: (i) $\mathrm{Var}(X^\delta_{jk}-X_{jk})$ will be neglected with respect to $\mathrm{Var}(X_{jk})$ and $\mathrm{Var}(X^\delta_{jk})$; and (ii) $\mathrm{Cov}(X^\delta_{jk}-X_{jk},T_k)$ will be neglected with respect to $\mathrm{Cov}(X_{jk},T_k)$ and $\mathrm{Cov}(X^\delta_{jk},T_k)$.

Condition 6. At the initial time step, we allow each netlet to have an arbitrary proportion of its neurons in each recovery state. But we require that the detailed assignment (according to these proportions) of particular neurons to each recovery state be done randomly, and independently for each netlet. We further require that over each netlet, the distribution of *initial recovery states* within the netlet, the distributions of *incoming synapses* (and their strengths) from various netlets, and the distributions of *outgoing* *synapses* to various netlets must all be mutually independent distributions. Finally, each netlet must contain enough neurons so that the law of large numbers can be applied to random variables defined on the neurons (considered as sample points) in the netlet.

Determination of model parameters. We have so far described a model containing two types of parameters: free macrostate parameters (to be set arbitrarily by the modeler), and randomly determined microstate parameters. Since some of these parameters are constrained by others, we present the following scheme to clarify the dependencies. If one were to actually simulate the base model, then the parameters should be determined in the following order:[1]

1. The model may contain an *arbitrary* number of netlets B_k indexed by $k \in D'$.
2. For each pair j,k of netlets such that B_j has input to B_k, the following macrostate parameters may be chosen *arbitrarily*:

   b_{jk}: the proportion of neurons in B_k receiving input from B_j;

   μ_{jk}: The total of synaptic strengths from B_j impinging upon any receiving neuron in B_k; and

   ν_{jk}: the total of squared synaptic strengths

[1] Of course the whole purpose of this chapter and the next is to make simulation of the base model unnecessary by deriving a lumped model which will yield the same essential information with far less computational effort. From this standpoint, the scheme for determining base model parameters has the following meaning: If one simulates the *lumped* model with a particular choice of parameter settings, then the results are the same as if one had simulated the *base* model with corresponding parameter settings and randomizations determined in the order shown here.

from B_j impinging upon any receiving neurons in B_k.

3. Within each netlet B_k, a collection of independent *random* subsets $\{B_{jk} \subseteq B_k | B_j \text{ has input to } B_k\}$ is chosen, such that the fraction of B_k's neurons in each subset B_{jk} is b_{jk}. (Condition 2)

4. For each such B_{jk}, *arbitrarily* create many sets of positive real numbers, such that the sum of the numbers in each set is μ_{jk}, and the sum of their squares is ν_{jk}. (Condition 3) Create as many such sets as there are neurons in B_{jk}.

5. For each B_{jk}, *randomly* assign a 1-1 correspondence between these sets and the neurons in B_{jk}. This 1-1 correspondence then gives each neuron β in B_{jk} a set of strengths for those of its synapses that come from B_j. Let $N_{j\beta}$ be the cardinality of the set of synapse strengths assigned to each neuron β. We then define $\{s_{nj\beta}, n=1,\ldots,N_{j\beta}\}$ to be this set of synapse strengths, taken in random order. (Condition 3)

6. For each $s_{nj\beta}$ defined in #5 above, *randomly* sample (with replacement) a presynaptic neuron from B_j. Let $I_{nj\beta}$ denote this presynaptic neuron. (Condition 1)

7. For each netlet B_k, *arbitrarily* assign the fraction of its neurons to be in each recovery state at the initial time step. (Condition 6)

8. Fill this quota for each recovery state by *randomly* picking (without replacement) neurons from B_k.

The random numbers used in different steps must be generated independently, to meet Condition 6.

Having defined $I_{nj\beta}$ in Condition 1 and $s_{nj\beta}$ in Condition 3, we can now rewrite equation (1) from the beginning of the chapter so that the input neighbors to neuron β and their corresponding synapse strengths are indexed according to the netlets these input neighbors belong to. This allows us to express the transition function δ_β of each neuron β as follows:

$$\delta_\beta(q_\beta, \{q_\alpha | \alpha \in I_\beta\}) = \begin{cases} 0 \text{ if } \Sigma_{nj} s_{nj\beta} \cdot \lambda(q_\alpha) \geq T_\beta \\ q_\beta + 1 \text{ otherwise} \end{cases} \qquad (3)$$

where in the summation above, $\alpha = I_{nj\beta}$, and the indices range according to the conventions given just before Condition 1.

Stochastic properties of the base model

Let us consider each netlet B_k as a discrete sample space, and each neuron β in B_k as a sample point in this space. Any property defined on each neuron in the sample space may then be considered to be a random variable. Now a random variable is simply a quantity that takes a value at each point in the sample space, so we are doing nothing new by redefining each neuronal variable in equation (3) as a random variable. The translation into random variables, however, allows us to use the notation and methods of probability theory, which will be convenient in our derivation.

First, the variable $I_{nj\beta}$ giving the input neighbors to a neuron β is translated into the random variable H_{njk}, giving the input neighbors for any neuron in netlet B_k. The recovery state q of a neuron in any netlet B_j having input to B_k is translated into the random variable R_j over netlet B_j. Thus the recovery state q_α of an input neighbor $\alpha = I_{nj\beta}$ becomes the random variable $R_j(H_{njk})$. Similarly, the synapse strengths $s_{nj\beta}$ to a neuron β in B_k becomes the random variable S_{njk} over netlet B_k, and the threshold T_β becomes the random variable T_k. Finally, the recovery state of neuron β after the application of the transition function δ will be translated into the random variable R_k^δ. With these translations, equation (3) can be restated

$$R_k^\delta = \begin{cases} 0 \Longleftrightarrow \Sigma_{nj} S_{njk} \cdot \lambda(R_j(H_{njk})) \geq T_k \\ R_k + 1 \text{ otherwise} \end{cases} \tag{4}$$

This equation gives us a way of dividing up the total input to netlet B_k according to netlet of origin (even though any neuron β in B_k may receive input from more than one netlet). The input received in B_k from any netlet j is given by the term $\Sigma_n S_{njk} \cdot \lambda(R_j(H_{njk}))$ for that particular j. Therefore let us represent this term (for any j) as a single random variable X_{jk} (see Fig. 4). We can then rewrite equation (4) as

Lemma 1.

$$R_k^\delta = \begin{cases} 0 \Longleftrightarrow \Sigma_j X_{jk} \geq T_k \\ R_k + 1 \text{ otherwise} \end{cases}$$

If we further define $X_k = \Sigma_j X_{jk}$, then Lemma 1 simply says that those neurons for which the total input X_k exceeds the threshold T_k will be the neurons that fire.

We can now derive a new distribution which combines the variances in input and threshold over any netlet B_k, taking account of the covariance. More formally, define a random variable $Z_k = X_k - T_k$. If we could find the distribution of Z_k, then we could calculate the new probability of firing

$$P[R_k^\delta=0] = P[X_k \geq T_k] = P[Z_k \geq 0] \tag{5}$$

by Lemma 1 and the definitions of X_k and Z_k. Therefore our approach will be to find the distribution of Z_k, and to use the parameters of

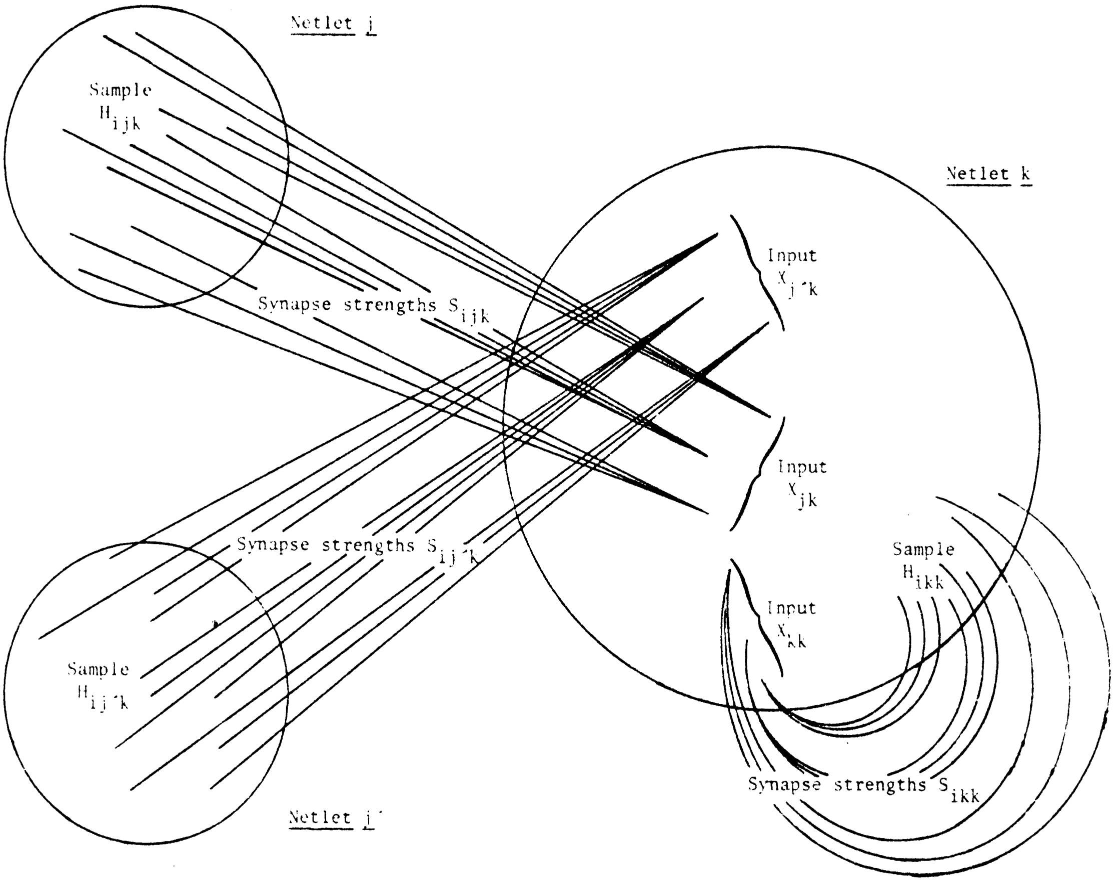


FIGURE 4. X_k is a random variable over netlet k whose value at any sample point (neuron) in the netlet is the total input arriving at that point. X_k can be expressed as the sum of component random variables X_{jk}, $X_{j'k}$, etc., each representing the contribution from a particular netlet j. Thus each X_{jk} represents the input from netlet j as this input is distributed over netlet k. (At any point in netlet k, the value of X_{jk} depends upon which neurons H_{njk} from netlet j are sampled, and upon the synapse values S_{njk} for each.)

this distribution as the state variables for the lumped model.

Finding the distribution of Z_k is essentially a matter of combining the between-neuron variance in the input X_k with the within- and between-neuron variances in the threshold T_k. This statistical calculation is carried out in the appendix. The results shown there are

Lemma 2. $P[T_k<x] = \Phi_{A,\nu}(x)$ where $A = E(m^{R_k})$

Lemma 3. $P[Z_k \geq z] = \Phi_{A,\nu'}(E(X_k)-z)$ where $A = E(m^{R_k})$ and $\nu' = \nu + \text{Var}(X_k) - 2\text{Cov}(X_k,T_k)$.

Lemmas 2 and 3 are similar in form to (and were derived from) the threshold distribution $P[T_\beta<x] = \Phi_{a,\nu}(x)$ of Condition 4. But while Condition 4 specified a probability distribution for a single neuron, Lemmas 2 and 3 specify distributions over a whole netlet B_k.[2] Correspondingly, while a and ν in Condition 4 are parameters of a single neuron, A and ν' in Lemma 3 are statistical parameters of the netlet. These statistical parameters will soon become the state variables of the lumped model.

[2] Using the law of large numbers (according to Condition 6) probabilities over the netlet are then interpreted as actual proportions of neurons in the netlet.

CHAPTER 4

DERIVATION OF THE LUMPED MODEL

Let us recapitulate the development thus far. We began the previous section by considering the neurons in a netlet as points in a sample space. Thus, the fact that different neurons have different recovery states and different input connections could be represented by defining the random variables R_k, H_{njk}, S_{njk}, T_k for recovery states, input connections, synapse strengths, and thresholds, respectively. Lemma 1 then gives the new recovery state as a random variable R_k^δ over the netlet. But the purpose of our lumped model will not be to keep track of the value of these random variables at every neuron in the netlet, but rather to save just enough information about each netlet to calculate the fraction of its neurons that will fire on the following time step. Equation (5) shows how this fraction (expressed in our notation as the probability of firing over the netlet) depends upon the distribution of Z_k, where $Z_k = X_k$ (representing total input) - T_k (representing threshold).

Expressed in our notation, the rerandomization assumption of Wong and Harth [1973, p. 80] and the equal influence assumption of Zeigler [1975, p. 391] are both essentially equivalent to assuming that, over each netlet B_k, the random variables X_k and T_k are independent (i.e. uncorrelated). More descriptively, this is assuming that the differences in threshold T_k (due to differing recovery states over the netlet) remain uncorrelated with the differences in input X_k received at different places in the netlet. Wong and Harth express this as a model in which the input connections to a netlet are rerandomized at each time step. Zeigler obtains a mathematically equivalent assumption by requiring that if netlet B_k receives any input from netlet B_j, then every neuron in B_k must have a similar input connection from B_j.

In contrast with these two approaches, we will not assume that X_k and T_k are independent. Instead, the covariance between X_k and T_k will be accounted for in our derivation of the lumped model. To see how this covariance will be used to represent the microstructure of each netlet B_k in our lumped model, let us look back at Lemma 3. As we have seen, this lemma expresses the stochastic threshold of the netlet in the same form as the original probability distribution for a single neuron's threshold. Thus, the only items of information about the microstate of the netlet needed to calculate its subsequent activity are

(i) $\text{RECOVERY}_k = E(m^{R_k})$, the mean of the recovery parameter over the netlet; (6)

(ii) $E(X_k)$, the mean (over different neurons) of the total input received by a neuron;

(iii) $\text{Var}(X_k)$, the variance in input over the netlet; and

(iv) $\text{COVARIANCE}_k = \text{Cov}(X_k, T_k)$, the covariance between the distributions of input and threshold over the netlet. (7)

Items (i) and (iv) have been given the names RECOVERY_k and COVARIANCE_k because they will be state variables in the lumped model.

Items (ii) and (iii), on the other hand, will be derived in terms of the ACTIVITY of all netlets having input to netlet B_k, where the state variable ACTIVITY will be defined in equation (13) below. Our lumped model will thus have three state variables, ACTIVITY_k, RECOVERY_k, and COVARIANCE_k, for each netlet B_k in the system. The interaction between netlets in the lumped model will be represented by fixed parameters CONNECT_{jk} and CONNECTSQ_{jk}, representing the total strength and the variability of the synaptic connections from any netlet B_j to any netlet B_k. (If there are no connections from B_j to B_k, then CONNECT_{jk} and CONNECTSQ_{jk} will both be 0.)

We will begin by deriving CONNECT_{jk} and CONNECTSQ_{jk} from the information about synaptic connections given in Condition 3 of the base model. (The role of these parameters in the transition function of the lumped model will be shown in Lemma 5 below and Lemma 7 in the appendix.) Let us consider the distribution of the synapse strengths from any netlet B_j to any netlet B_k. This distribution is represented formally as the distribution over B_k of the random variables S_{njk}, $1 \leq n \leq N$, defined in the previous section. (Recall that n indexes the multiple input neighbors to each neuron.) We need to know the sums of the first and second moments $\Sigma_n E(S_{njk})$ and $\Sigma_n E(S^2_{njk})$ of this distribution in order to calculate items (ii) and (iii) above, the mean and variance of the input X_k. Using the definition of S_{njk} and Condition 3 of the base model, we obtain

$$\begin{aligned}\Sigma_n E(S_{njk}) &= E(\Sigma_n S_{njk})\\ &= E(\Sigma_n S_{njk} | B_{jk}) P[B_{jk}] + E(\Sigma_n S_{njk} | B_k - B_{jk})(1 - P[B_{jk}])\\ &= E(\Sigma_n s_{nj\beta}) b_{jk} + 0 \cdot (1 - b_{jk})\end{aligned}$$

$$= \mu_{jk} b_{jk} \tag{8}$$

where each μ_{jk} and b_{jk} is a fixed parameter of the base model as specified in Conditions 2 and 3. The same reasoning applied to S^2_{njk} rather than S_{njk} yields

$$\Sigma_n E(S^2_{njk}) = E(\Sigma_n s^2_{nj\beta}) b_{jk} = \nu_{jk} b_{jk}. \tag{9}$$

The two summed moments just calculated will be the only information about the neuronal connections required by the lumped model. Accordingly, we define the following two fixed parameters of the lumped model in terms of the original base model parameters:

$$\text{CONNECT}_{jk} = \mu_{jk} b_{jk} \tag{10}$$

$$\text{CONNECTSQ}_{jk} = \nu_{jk} b_{jk} \tag{11}$$

Finally, we note that the proportion of neurons firing (expressed as a probability over the netlet B_k) is

$$\begin{aligned} P[R_k=0] &= P[\lambda(R_k)=1] \text{ by the definition of } \lambda \\ &= E[\lambda(R_k)] \end{aligned} \tag{12}$$

since $\lambda(R_k)$ is a discrete $\{0,1\}$ variate. So we define

$$\text{ACTIVITY}_k = P[R_k=0] = E(\lambda(R_k)) \tag{13}$$

to represent the mean firing activity over netlet B_k at any given time step.

Derivation of the transition function

In order to derive a transition function for the system in terms of netlet variables, we need a notation which distinguishes between a current state of the system and its successor state. We will distinguish the successor state by a superscript δ (as in the beginning of this section). In other words, given any state q of the base model and its successor state $\delta(q)$, the variables ACTIVITY_k, RECOVERY_k, and COVARIANCE_k refer to the system in state q, while ACTIVITY_k^δ, RECOVERY_k^δ, and $\text{COVARIANCE}_k^\delta$ refer to the system in state $\delta(q)$. Thus, if the variables without superscripts are defined as above, then

$$\text{ACTIVITY}_k^\delta = P[R_k^\delta=0] \tag{14}$$

$$\text{RECOVERY}_k^\delta = E(m^{R_k^\delta}),\ \text{and} \tag{15}$$

$$\text{COVARIANCE}_k^\delta = \text{Cov}(X_k^\delta,T_k^\delta) \tag{16}$$

where R_k^δ, X_k^δ, and T_k^δ are defined at the beginning of this section.

Using this notation, we see that the current recovery state and the mean, variance, and covariance of input to a netlet are sufficient to calculate its subsequent ACTIVITY:

$$\begin{aligned}\text{ACTIVITY}_k^\delta &= P[R_k^\delta=0]\\ &= P[Z_k\geq 0]\\ &= \Phi_{A\nu'}(E(X_k))\end{aligned} \tag{17}$$

where $A = \text{RECOVERY}_k$ and $\nu' = \nu + \text{Var}(X_k) - 2\cdot\text{COVARIANCE}_k$, obtained by substituting the definition of these variables into Lemma 3.

Now in order to write a transition function for ACTIVITY solely in terms of the three desired state variables (ACTIVITY, RECOVERY, and COVARIANCE), we must resolve the input term $E(X_k)$ into the state variables of netlets from which this input originates. Part of this derivation will depend upon an induction argument applied to the initial distribution of recovery states specified in Condition 6. Since this argument is peripheral to our derivation of the lumped model, it is given in the appendix. The result shown there is

Lemma 4. The recovery states of neurons in a netlet are independent of their output synapse weights. More precisely, for any n, j, and k, $R_j(H_{njk})$ is independent of S_{njk} (where by the definition of H_{njk}, $R_j(H_{njk})$ represents the states of the set of neurons whose output synapse weights are S_{njk}: see figure 4).

Using this lemma, we can calculate

$$
\begin{aligned}
E(X_k) &= \Sigma_j E(X_{jk}) \\
&= \Sigma_{nj} E(S_{njk} \cdot \lambda(R_j(H_{njk}))) \text{ by def. of } X_{jk} \\
&= \Sigma_{nj} E(S_{njk}) \cdot E(\lambda(R_j(H_{njk}))) \text{ by Lemma 4} \\
&= \Sigma_{nj} E(S_{njk}) \cdot \text{ACTIVITY}_j \text{ since } H_{njk} \text{ is uniform} \\
&= \Sigma_j \text{CONNECT}_{jk} \cdot \text{ACTIVITY}_j \qquad (18)
\end{aligned}
$$

using equation (8) and the definitions of CONNECT_{jk} and ACTIVITY_j. The formula for $E(X_k)$ can now be substituted into equation (17) to obtain the transition function for ACTIVITY in our lumped model. (See Figure 5.)

Lemma 5. $\text{ACTIVITY}_k^{\delta} = \Phi_{A,\nu'}(\Sigma_j \text{CONNECT}_{jk} \cdot \text{ACTIVITY}_j)$, where A = RECOVERY_k, $\nu' = \nu + \text{Var}(X_k) - 2 \cdot \text{COVARIANCE}_k$, and the definition of the 2-parameter distribution Φ is given in Condition 4.

Note that the input term (i.e. the summation) in Lemma 5 is a weighted sum of signals from other netlets, analogous to the weighted sum $\Sigma_{nj} s_{nj\beta} \lambda(q_\alpha)$ in the transition function for an individual neuron (equation 1). But now consider the function which translates this

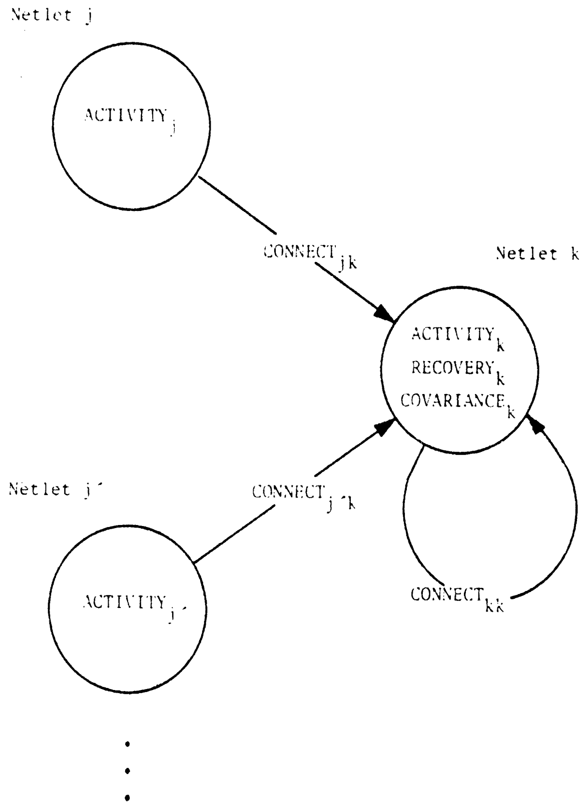


FIGURE 5. Each of the variables shown for netlets j and k contributes to the new activity (ACTIVITY$^{\delta}_{K}$) calculated for netlet k. The exact formula is shown in Lemma 5.

total input into output. In the case of an individual neuron (Fig. 6a), this function is a sharp threshold. If the input exceeds the threshold, the output is 1; otherwise the output is 0. But in the case of a netlet (Fig. 6b) the sharp threshold is replaced by a sigmoid function--the function $\Phi_{A,\nu}$, of Lemma 5. This sigmoid function represents the combined effect of the differing thresholds of neurons in the netlet.

While the ACTIVITY of a netlet will be the main variable of interest at the level of psychological constructs, we have seen that two other state variables, RECOVERY and COVARIANCE, must be included in our lumped model. The purpose of these two additional variables is to maintain sufficient information about the microstructure of a netlet to calculate subsequent ACTIVITY accurately. Of course, once we include these two variables in our lumped model, we must derive *their* transition functions, just as we derived a transition function for ACTIVITY. The crucial step here will be to express the transition functions for RECOVERY and COVARIANCE solely in terms of the three state variables we have defined. In other words, we must show that the three variables form a self-contained lumped model: that knowing the ACTIVITY, RECOVERY, and COVARIANCE for each netlet, we can calculate not only the subsequent ACTIVITY, but also the subsequent RECOVERY and COVARIANCE.

First we will derive the transition function for RECOVERY:

$$\begin{aligned} \text{RECOVERY}_k^\delta &= E(m^{R_k^\delta}) && \text{by eq. (15)} \\ &= P[R_k^\delta=0]\cdot E(m^0) + P[R_k^\delta\neq 0]\cdot E(m^{R_k^\delta}|R_k^\delta\neq 0) \\ &= P[R_k^\delta=0] + P[R_k^\delta\neq 0]\cdot E(m^{R_k+1}) && \text{by Lemma 1} \end{aligned}$$

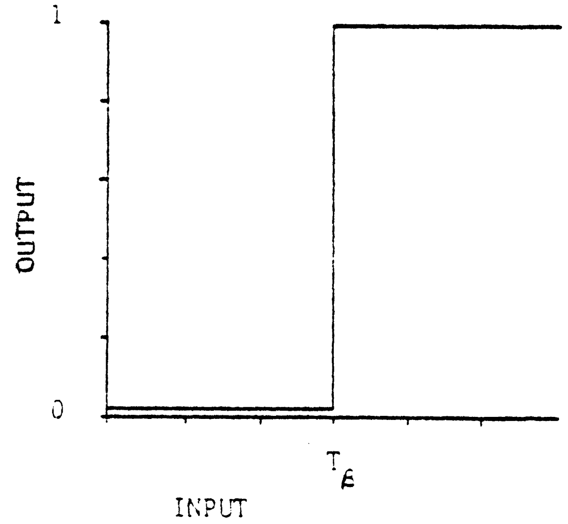


FIGURE 6a. Input-output function for an individual neuron β with threshold T_β.

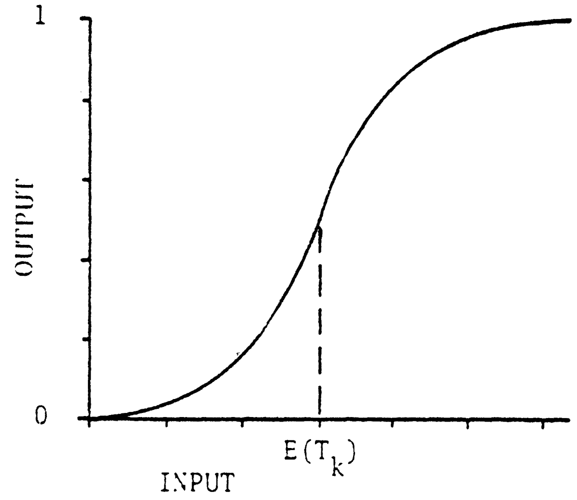


FIGURE 6b. Input-output function for a netlet with mean threshold $E(T_k)$.

$$= \mathrm{ACTIVITY}_k^{\delta} + (1-\mathrm{ACTIVITY}_k^{\delta})\cdot mE(m^{R_k}) \qquad \text{by eq. (14)}$$

$$= \mathrm{ACTIVITY}_k^{\delta} \quad m\cdot(1-\mathrm{ACTIVITY}_k^{\delta})\mathrm{RECOVERY}_k \qquad \text{by eq. (6)}$$

Thus, given the $\mathrm{ACTIVITY}_k$, $\mathrm{RECOVERY}_k$, and $\mathrm{COVARIANCE}_k$ of each netlet (at a given time step) one may calculate the $\mathrm{ACTIVITY}_k^{\delta}$ and $\mathrm{RECOVERY}_k^{\delta}$ for the next time step, where $\mathrm{ACTIVITY}_k^{\delta}$ is calculated according to Lemma 5 and then its value is used in the calculation of $\mathrm{RECOVERY}_k^{\delta}$ above. The constants m, ν, and $\mathrm{CONNECT}_{jk}$ appearing in Lemma 5 and equation (19) are fixed parameters obtained from the base model according to Condition 4 and equation (10).

It remains to derive the transition function for COVARIANCE. Since this derivation is tedious and does not introduce any essential concepts, it is given in the appendix. Rather than expressing $\mathrm{COVARIANCE}_k^{\delta}$ completely in terms of the state variables (which would make the formula below even longer) we make use of quantities already calculated in the expression for $\mathrm{ACTIVITY}_k^{\delta}$:

$$\mathrm{COVARIANCE}_k^{\delta} = \tfrac{1}{2}\Delta\tau(1+m\cdot\mathrm{RECOVERY}_k)\Phi'_{A,\nu'}(E(X_k))\cdot(\mathrm{Var}(X_k^{\delta}) + \mathrm{Var}(X_k)) + m(1-\mathrm{ACTIVITY}_k^{\delta})\cdot\mathrm{COVARIANCE}_k$$

where the parameters A and ν' still have the same values as in Lemma 3: $A = E(m^{R_k}) = \mathrm{RECOVERY}_k$, and $\nu' = \nu + \mathrm{Var}(X_k) - 2\cdot\mathrm{COVARIANCE}_k$. The probability density function Φ' is simply the derivative of the cumulative distribution function Φ defined in Condition 4.

The appearance of quantities such as $\mathrm{ACTIVITY}_k^{\delta}$ and $\mathrm{Var}(X_k^{\delta})$ in the formula above means that the formula is useful only if the transition function is calculated in a certain order. Such a computational sequence is spelled out in the next section.

Formal specification of the lumped model

The results of the previous two sections can now be brought together into a concise specification for the lumped model we have derived. The components of the lumped model are netlets. Briefly, each netlet is a group of neurons with the property that input connections to the netlet (both from itself and from other netlets) are distributed randomly over the neurons within the netlet. (This is spelled out more precisely in Conditions 1-6 of Chapter 2.) Beyond these conditions, the derivation is not committed to any particular functional interpretation of the netlet structure, and our netlets could be used to model functional units such as Mountcastle's (1957) columnar response units, Hubel and Wiesel's (1962) feature detectors, Konorski's (1967) gnostic units, John's (1967) reverberatory circuits, or the cell assemblies of Hebb (1949) and Milner (1957, 1974). These theories share a common hypothesis--that the functional units of perception, association, and action are not the stochastic responses of individual neurons, but rather the more reliable statistical responses of neural aggregates.

The state of each netlet k is specified by 3 real numbers: ACTIVITY_k, representing the proportion of neurons in the netlet that are firing; RECOVERY_k, representing the mean recovery state of neurons in the netlet; and COVARIANCE_k, representing the inhomogeneity in recovery state over the netlet due to the fact that inputs from different sources arrive at different places in the netlet. (More technically, COVARIANCE_k is the covariance between the distribution of inputs and the distribution of recovery-state thresholds over the netlet.)

The lumped model also has fixed parameters CONNECT_{jk} and CONNECTSQ_{jk} to represent the connection strength from each netlet j to each netlet k in the model (Fig. 7). Intuitively, CONNECT_{jk} gives the total strength of the synapses from netlet j to netlet k, while CONNECTSQ_{jk} is a measure of the variability in the strengths of the different synapses that contribute to CONNECT_{jk}. (The precise definitions of these parameters are given in Condition 3 and equations (10) and (11).)

The state-transition function of the lumped model can now be expressed in terms of these variables and parameters. Given the ACTIVITY_k, RECOVERY_k, and COVARIANCE_k of each netlet at any time step, the values of these variables on the following time step (indicated as $\text{ACTIVITY}_k^{\delta}$, $\text{RECOVERY}_k^{\delta}$, and $\text{COVARIANCE}_k^{\delta}$) can be calculated for each netlet k as follows:

Step 1. Calculate the mean and variance of the input to netlet k:

$$E(X_k) = \Sigma_j \text{CONNECT}_{jk} \cdot \text{ACTIVITY}_j$$

$$\text{Var}(X_k) = \Sigma_j (\text{CONNECTSQ}_{jk} \cdot \text{ACTIVITY}_j - \text{CONNECT}_{jk}^2 \cdot \text{ACTIVITY}_j^2)$$

(from equation (18) and Lemma 7).

Step 2. Calculate the new activity

$$\text{ACTIVITY}_k^{\delta} = \Phi_{A,\nu'}(E(X_k))$$

(from equation (17)), where $A = \text{RECOVERY}_k$; $\nu' = \nu + \text{Var}(X_k) - 2 \cdot \text{COVARIANCE}_k$. (The constant ν is the variance in the threshold of an individual neuron due to random noise, and the 2-parameter distribution Φ is defined in Condition 4. Briefly, $\Phi_{A,\nu'}$ is the weighted

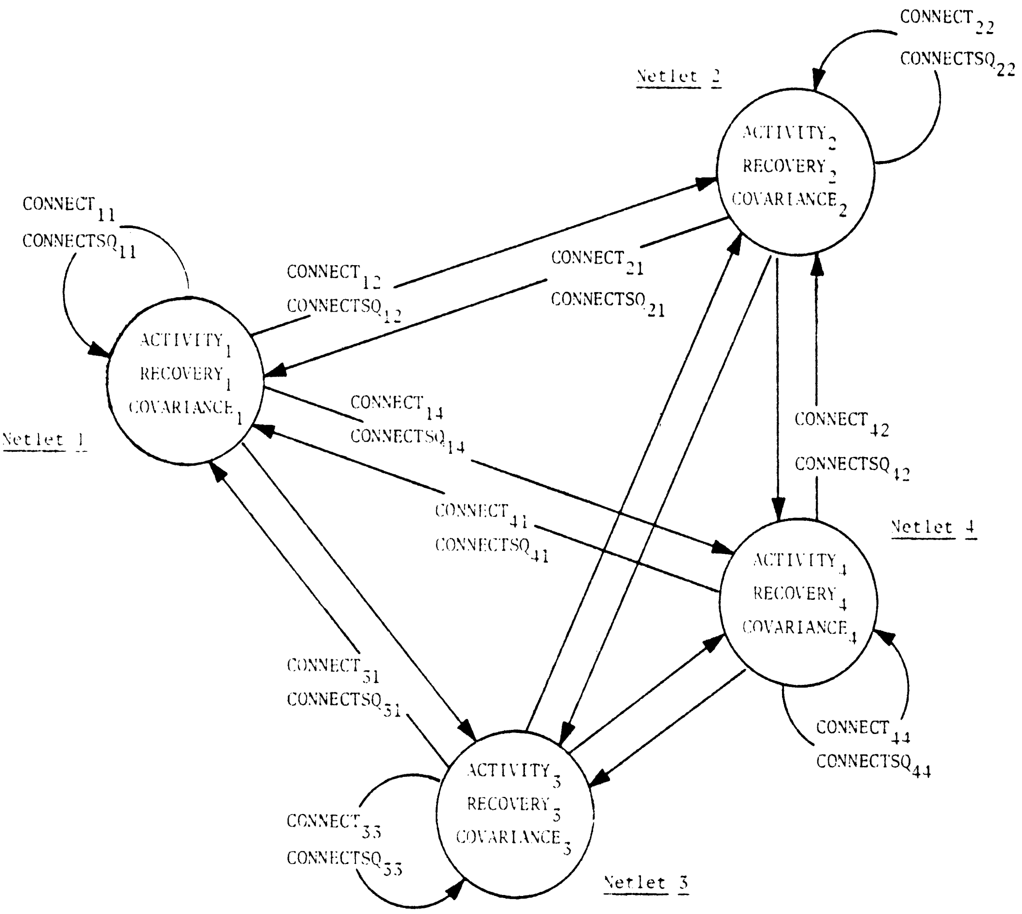


FIGURE 7. The structure of a sample network in the lumped model. ACTIVITY, RECOVERY, and COVARIANCE are state variables; CONNECT and CONNECTSQ are fixed parameters of the lumped model derived from synaptic strengths in the base model.

sum $A \cdot N_0 + (1-A) \cdot N_\infty$ where N_0 and N_∞ are cumulative normal distributions. N_0 has mean τ_0 and variance ν', where the constant τ_0 is the expected threshold of a neuron which has just fired. N_∞ has mean τ_∞ and variance ν', where τ_∞ is the expected threshold of a completely recovered neuron.)

Step 3. Calculate the new recovery state

$$\text{RECOVERY}_k^\delta = \text{ACTIVITY}_k^\delta + m(1-\text{ACTIVITY}_k^\delta) \cdot \text{RECOVERY}_k$$

(from equation (19)), where m is a constant (defined in Condition 4) giving the rate at which an individual neuron's threshold decays exponentially from τ_0 towards τ_∞. (That is, the expected threshold of a neuron is $m^q \tau_0 + (1-m^q)\tau_\infty$, where q is the number of time steps since the neuron has fired.)

Step 4. Calculate the new covariance

$$\text{COVARIANCE}_k^\delta = \frac{1}{2}\Delta\tau(1+m \cdot \text{RECOVERY}_k)\Phi'_{A,\nu'}(E(X_k)) \cdot (\text{Var}(X_k^\delta)+\text{Var}(X_k)) + m(1-\text{ACTIVITY}_k^\delta) \cdot \text{COVARIANCE}_k$$

(from equation (38) of the appendix), where the probability density $\Phi'_{A,\nu'}$ is the derivative of the cumulative distribution function $\Phi_{A,\nu'}$ explained under Step 2, and where $\text{Var}(X_k^\delta) =$ $\Sigma_j(\text{CONNECTSQ}_{jk} \cdot \text{ACTIVITY}_j^\delta - \text{CONNECT}_{jk}^2 \cdot (\text{ACTIVITY}_j^\delta)^2)$, from Lemma 7.

Steps 1 to 4 are to be carried out for each netlet separately, but with the provision that Step 2 must be completed for *all* netlets before Step 4 can be done on any netlet. (This is because Step 2 results for all netlets enter into the calculation of $\text{Var}(X_k^\delta)$ in Step 4.)

This lumped model, while somewhat involved, can be simulated with far less computational effort than the equivalent base model. The state of each netlet in the lumped model can be stored in 3 variables, where the base model would require a separate state variable for each neuron in the netlet. Steps 1-4 are calculated only for each netlet of the lumped model, while a separate calculation along the lines of equation (1) would have to be carried out for each neuron of the base model.

By carrying along not only the ACTIVITY, but also the RECOVERY and COVARIANCE of each netlet, we have been able to construct a lumped model which, under Conditions 1-6, is a homomorphic image of the base model. That is, given any initial state and parameter settings, the state trajectory of the lumped model will be the same as if we had simulated the corresponding base model and tallied up the states of the individual neurons to obtain the ACTIVITY, RECOVERY, and COVARIANCE of each netlet at each time step.

CHAPTER 5
FUNCTIONAL INTERPRETATION OF THE LUMPED MODEL

Psychological constraints

In Chapter 2, we characterized human pattern recognition ability in terms of stimulus equivalence. From this point of view, an object or concept is an *abstraction* that applies to a whole equivalence class of stimuli. Even though different subsets of features are present in different instances, each stimulus in the equivalence class is recognized as an example of the same abstraction.

This recognition ability might be accounted for by an economical storage hypothesis: that humans do not store each particular stimulus they see, but rather store abstract representations whose features characterize a whole equivalence class of stimuli. Attneave [1957] tested this idea in a study of paired-associate learning of random shapes. Learning was aided by prior familiarization with a prototype stimulus whose features showed the central tendency of the equivalence class. Posner and Keele [1968] presented degraded instances of an abstraction and showed that the prototype of the abstraction (not previously presented!) was recognized more reliably than any of the instances that were presented. Franks and Bransford [1971] obtained the same result using a different task. Rosch and Mervis [1975] emphasize the idea that the abstract representation might contain no features common to all stimuli in the class, suggesting that perceptual and cognitive abstraction are based upon the integration of a large number of unreliable features.

While these and other psychological studies have substantiated the idea that abstract representations are stored, they leave open the question of how such a representation might be stored in the structure of the brain. Ideally, the structure which stores the abstraction should also do the stimulus equivalencing required for retrieval (i.e. for recognizing instances of the abstraction). The netlet mechanism, as discussed in the introduction and as formalized in the base and lumped models, can meet this requirement. We suggested in the introduction how a network of feature detectors might serve as an equivalencing mechanism; it is now time to introduce this hypothesis into the formal model, as a constraint on the connection strengths between netlets (ii below). Our hypothesis may be summarized as follows:

(i) The unit of storage is the abstract internal representation of an object (an equivalence class of stimuli) in the environment. The features of this representation are the perceptual similarities among the stimuli in the equivalence class.

(ii) This unit of storage is implemented as a neural network.

a. A network of connected feature detectors represents an object in the environment.

b. Each feature element is a netlet, as formalized in the base and lumped models. Thus, for each feature of the object, there is a corresponding netlet in the model.

c. Each netlet represents and receives input from one sensory feature. The feedback connections from a netlet to itself are set so that the netlet will become reverberatory if its threshold is exceeded.

d. Modeling of associations in the environment: The connection strength from netlet j to netlet k is a monotonic function of the degree to which the feature that j represents predicts the feature that k represents. Here "degree of prediction" refers to the likelihood ratio of Bayes' theorem: the conditional probability of feature k given feature j, divided by the unconditional probability of feature k. This probability is taken over the equivalence class of stimuli in (i), weighted by frequency of occurrence in the environment.

e. If the network representations of two different objects are likely to be activated by the same features, then there are inhibitory connections between the netlets belonging to one representation and those belonging to the other.

Item d is the mechanism for stimulus equivalence in the model. When the network representation of an object receives input, some of its feature detectors will receive enough input to reverberate; others will receive partial input or none. Now each reverberatory feature detector will facilitate each inactive feature detector in proportion to its predictive validity (that is, according to the validity with which the detected feature predicts the presence of the undetected feature). Thus, probable but so far undetected features will be facilitated, lowering their threshold for responding to environmental input. And if the circumstantial evidence for a missing feature becomes overwhelming, then it can be facilitated so much that it becomes "filled in" -- active even in the absence of environmental support.

The associative connections within a network and the inhibitory connections between competing networks (items d and e above) will normally force activity in the system to converge rapidly to the best inference that can be made on the basis of the partial and unreliable information available. When information from the stimulus is scarce, one may consciously experience this competition between alternative perceptual hypotheses -- for example, approaching a fuzzy object in dim light that finally turns out to be a crumpled newspaper on the sidewalk.[1] When stimulus information is plentiful, on the other hand, convergence is so rapid that the losing alternatives are never consciously experienced. In this case, one must rely on more sensitive psychological measurements (such as Posner's reaction-time technique[2]) to show that these losing alternatives are indeed partially activated.

Constraint d above is plausible not only because it leads to an appropriate stimulus equivalencing mechanism, but also because such connections could be formed automatically as the result of a simple contiguity learning rule [Hebb, 1949; Anderson, 1972]. Milner [1957] originally showed the advantages of incorporating inhibitory connections (of the type suggested in constraint e above) into Hebb's basic hypothesis.

Although developed in the context of perception, the mechanism outlined in hypotheses (i) and (ii) above will also serve as a model of abstraction at the cognitive level. In this case, the network represents a concept rather than an object. The "feature detectors" constituting such a network do not detect features directly from the environment, but instead detect features that arise from perceptual equivalencing. That is, the result of a network equivalencing

[1] This example was suggested by Stephen Kaplan. James' [1892] chapter on perception provides many other examples, all suggesting that stimulus information does not in itself constitute a perception, but must be constructed into a perception.

[2] Posner, M.I. *Chronometric Explorations of Mind*, Fitts Memorial Lectures delivered at the University of Michigan, September 1976, to be published as a book. A specific example of the chronometric method is given by Conrad [1974].

mechanism at any level of processing may in turn be the input to a feature detector at the next level. Milner [1974] further discusses the possibility of such a hierarchical organization.

Pilot simulations

Model building is an effort to produce a structure that satisfies a series of constraints and expectations. The constraints are imposed by the necessity of consistency with models at other levels; expectations for the model are based upon previous observations of the system, together with an underlying idea about the function of the system. These expectations ultimately concern the behavior of the model over time. Since the behavior of a model is usually not obvious from its structure, model building is a process of successive approximation: trying a structure; seeing if the behavioral expectations are met; modifying the structure (within the given constraints); and repeating the cycle.

Model testing is an effort to see if the model meets new expectations which it was not explicitly designed to meet. Generally these arise in the context of new experiments. Except for this difference in the type of expectation, model testing proceeds by the same 4-step successive approximation cycle just given for model building.

Despite this conceptual similarity between model building and model testing, most disciplines have rigorous standards only for model testing. Thus, a model is judged solely on the basis of a few critical experimental tests designed specifically for that model.

Rigorous model building, on the other hand, would be an attempt to translate other sources of knowledge about the system (such as experimentally validated models at other levels) into constraints upon the model being built. Rigorous model building is difficult, and understandably rare, but the reward is an enormous increase in the size of the data base upon which one's model is based. (See Figure 3 in Chapter 2.)

Thus, the emphasis is on computer simulation, which allows the behavioral consequences of a model to be precisely determined -- not only for generating experimental predictions, but also for carefully evaluating the model against the constraints and expectations of rigorous model building. The simulations discussed below were based upon the lumped model of Chapter 4. The first step (as in most simulations) was the adjustment of parameters to obtain reasonable behavior, within the constraints of hypothesis (ii) of the previous section. The previous section also outlines, in a less formal way, the behavioral expectations that guided model building. Altogether, 3 versions of the model were developed, each simulating a different pattern-recognition task. The first two versions simulated experiments drawn from the literature [Franks and Bransford, 1971; Reed, 1972], while the third simulated a new task. The predictions of the third version were then tested in a psychological experiment.

The third version of the model was based upon the parameter settings obtained by fitting the first two versions to the data of experiments they simulated. In addition, the whole idea for the third version was based upon a specific finding from the second simulation. This finding concerned the discriminability of two hypotheses, the network hypothesis discussed here and an alternative viewpoint of feature independence [Reed, 1973] illustrated in Figure 8. Several types of pattern recognition models, including discriminant analysis, summation of cue validities, and distance models with various metrics, all assume feature independence, in which each feature makes an independent additive contribution to the global decision. This is in contrast to the network model (Fig. 1) discussed in the previous section, in which active feature detectors may facilitate partly active detectors, depending upon the connections between them.

These two alternative models seem quite different, yet the simulation showed them to yield almost identical predictions for Reed's [1972] experiment. It seemed that this lack of discriminability between the alternative models might be due to the fact that in Reed's experiment, only four features were used to distinguish the patterns. (This small number of features is typical for the studies on which the feature-independence hypothesis is based.) For this reason, a new pattern recognition task with a larger number of unreliable features was constructed. The purpose of this task was to differentiate the feature-network hypothesis from the feature-independence hypothesis. Both the simulation of this task and the empirical test of the simulation predictions are discussed in the next chapter.

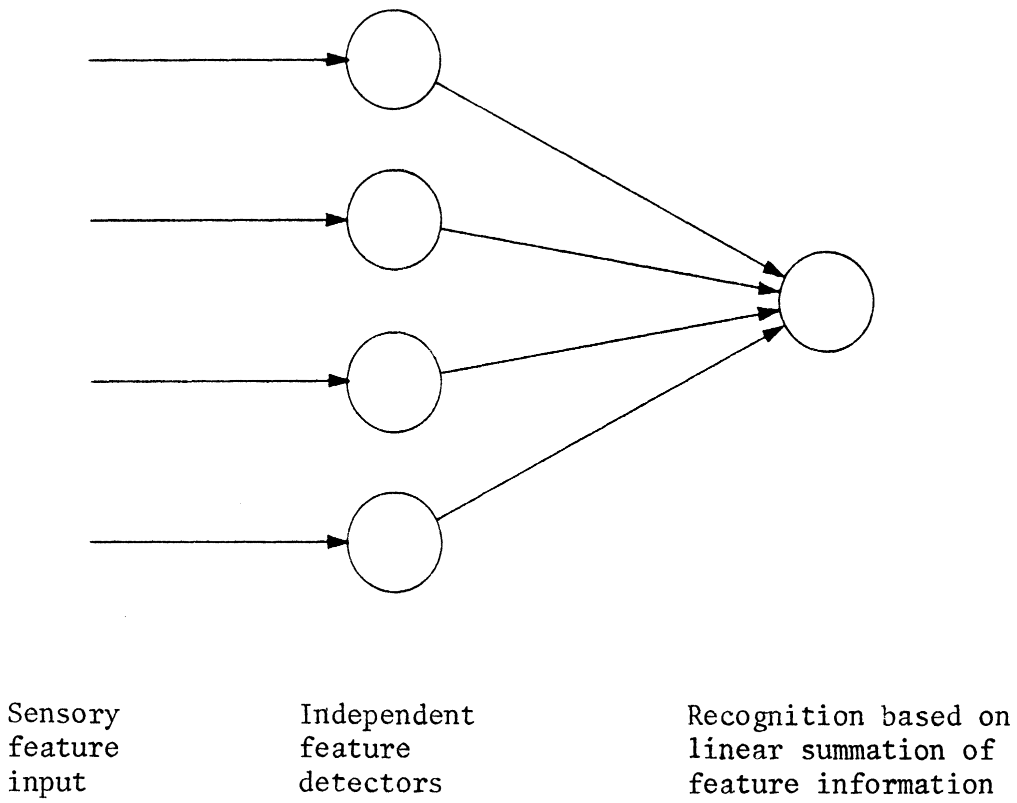

Sensory feature input
Independent feature detectors
Recognition based on linear summation of feature information

FIGURE 8. The feature-independence hypothesis (to be contrasted with the neurally based hypothesis that we introduced in Figure 1 and developed formally in Chapters 3 and 4). The feature independence hypothesis asserts that the information from each feature makes an independent additive contribution to the global recognition decision.

CHAPTER 6
HUMAN PATTERN RECOGNITION: AN EMPIRICAL DEMONSTRATION

A controlled experiment can be viewed as a restriction or simplification of the natural setting for the behavior studied. Stimuli, procedures, and other factors must be extracted from the natural setting and standardized so that the results of the experiment can be rigorously attributed to the factors being manipulated. The danger, of course, is that the restrictions and simplifications may result in behavior that is not representative of the natural setting.

One way of minimizing this danger is put forth by Brunswick [1956]. Brunswick maintains that the context of an experiment -- all the factors not manipulated in the design -- should be viewed as a sample from a population of possible contexts. Normally one studies a psychological process in the laboratory with the idea that the same process occurs in real-world behavior. More specifically, the process is presumed to occur in a variety of real-world[1] contexts, and these constitute the population over which we wish to generalize. The point is that any behavioral experiment must sample contexts just as it samples participants, and in each case the results are (at

[1] We will loosely use the term "real world" to refer to the natural setting for a behavior, acknowledging with Runkel and McGrath [1972] that both natural settings and experiments yield real behavior, so "natural setting" is more precise terminology.

least in the privacy of the investigator's own mind) generalized to a more inclusive population. The researcher should therefore give at least some thought to the representativeness of context as well as participant samples.

Of course the ideal way to get a representative sample is to draw it randomly from an explicitly specified population. For participant samples, this ideal is even approachable in high-budget survey research. But for context samples, an unbiased random selection is probably never realized. For one thing, a single experiment only samples one or a few contexts, so many experiments are needed to establish a context sample of any size. (This is why there are no critical experiments in psychology, but instead a concept of validity based on converging evidence from different experimental contexts.) Even so, the necessity for standardization, measurement, and control severely constrains the type of context that can be studied, and so precludes fair sampling of natural-setting contexts.

The impossibility of fair sampling of experimental contexts does not mean the issue should be ignored; it means that one should do the best one can. If a sample must be biased, it should at least be biased intelligently. In most experiments, the representativeness of the context depends upon implicit assumptions about the process being studied.[2] The implicit argument is that, for the process being studied, the experimental context captures the essential properties of the natural setting, in spite of the differences between the two.[3]

Of course if one suspects that some essential factor of the natural setting has *not* been adequately sampled in previous experimental contexts, then one should design contexts that sample new values of this factor. That is precisely the motivation for the present study. As described in the previous section, there is a specific empirical question which would differentiate our network model from a large class of alternative models of feature integration. The setting for the question is a task in which people learn

[2] Representativeness of the *participant* sample also depends upon such assumptions, if one follows the usual practice of sampling introductory psychology classes (as we do here). We also sample only one sex. Such restrictions increase the likelihood of significant results for the population sampled (internal validity), but the assumption that the process would not differ qualitatively in a larger population (external validity) is usually a bit shaky. Our study assumes that the basic neural mechanism for perception is the same over a quite large population, but certainly does not assume that parameters of the perceptual process (such as speed, accuracy, feature weightings) are so generalizable.

[3] Since there are potentially infinite number of factors to observe or control, the choices of which factors to worry about and which to ignore can never be given a complete explicit justification. Most of the choices are made intuitively, with an intuition trained by research experience and the examples of others. The need to develop such a trained intuition may be the reason empirical disciplines stress the need for research experience. The common implicit assumptions shared by a community of investigators because their intuitions are based on similar experience is perhaps the chief element of Kuhn's [1962] scientific paradigm. A researcher working within such a paradigm does not attempt to communicate the intuition and experience shared by the community. Instead, explanations concentrate on those particular features of an experimental context which differ from previous studies or which might be subject to disagreement among colleagues.

to classify a set of stimuli into categories or equivalence classes. The stimuli vary along a certain number of well-defined feature dimensions, and the participants make their classification decisions on the basis of the feature information. The question is whether the information from each feature makes a separate additive contribution to the classification decision (the feature-independence hypothesis), or whether there is interaction between the features, as in the feature-network hypothesis. The feature-independence hypothesis (alternatively termed the principle of interdimensional additivity) was explicitly put forth by Tversky and Krantz [1969]. Their study which tested and confirmed this hypothesis, however, utilized a task in which the stimuli varied along only three feature dimensions. The feature-independence hypothesis is also an assumption in discriminant analysis [Rodwan and Hake, 1964] and in cue validity, proximity, average distance, and prototype models [defined, reviewed, and tested by Reed, 1972], but in each case the supporting evidence is based on a task with four or fewer feature dimensions. The implicit assumption, of course, is that four features are enough to uncover the principles of feature integration -- that adding more features to the task would not make the results qualitatively different.

If one's approach to understanding feature integration is not constrained by possible neural mechanisms, then one would have no particular reason to suspect such a qualitative change. If one's approach *is* so constrained, on the other hand, then there is good reason to suspect feature interactions to be much more significant if there are a large number of unreliable features. I don't mean to suggest that the decision to push empirical inquiry in this particular direction is immediately obvious from neural constraints. Formal analysis and simulation are important tools for making the link from hypothesis to testable consequence -- provided the hypothesis is well-constrained in the first place. In the terminology of the previous chapter, model testing is more likely to proceed in fruitful directions if preceded by rigorous model building.

We have so far explained why it was desirable for the present experiment to study pattern recognition based upon a large number of features. This was one major difference from previously sampled experimental contexts for feature integration. Another property of the natural setting that was represented in the experimental context was feature correlation. It will be more convenient to discuss this property in the next section, after developing some necessary terminology.

Finally, representativeness of the experimental context dictated the particular choice of stimulus. It was desired to have a task as close as possible to pattern recognition in the natural setting without sacrificing the careful experimental control necessary for internal validity. One type of object that is encountered frequently in the natural setting and which people learn to recognize is the common residential house. Perhaps because houses are so frequent in pictures and sketches as well as in the world, they are readily recognized in quite simple line drawings. In an experimental design, such drawings can be easily standardized and varied in precise measurable ways. Houses also seem ideal because they can be varied along many feature dimensions and still look like houses. Once houses with many features were chosen to be the stimulus, the exact number of features was set at 9 to allow correlations to be established within three subsets of 3 features each, as discussed in detail later in the chapter.

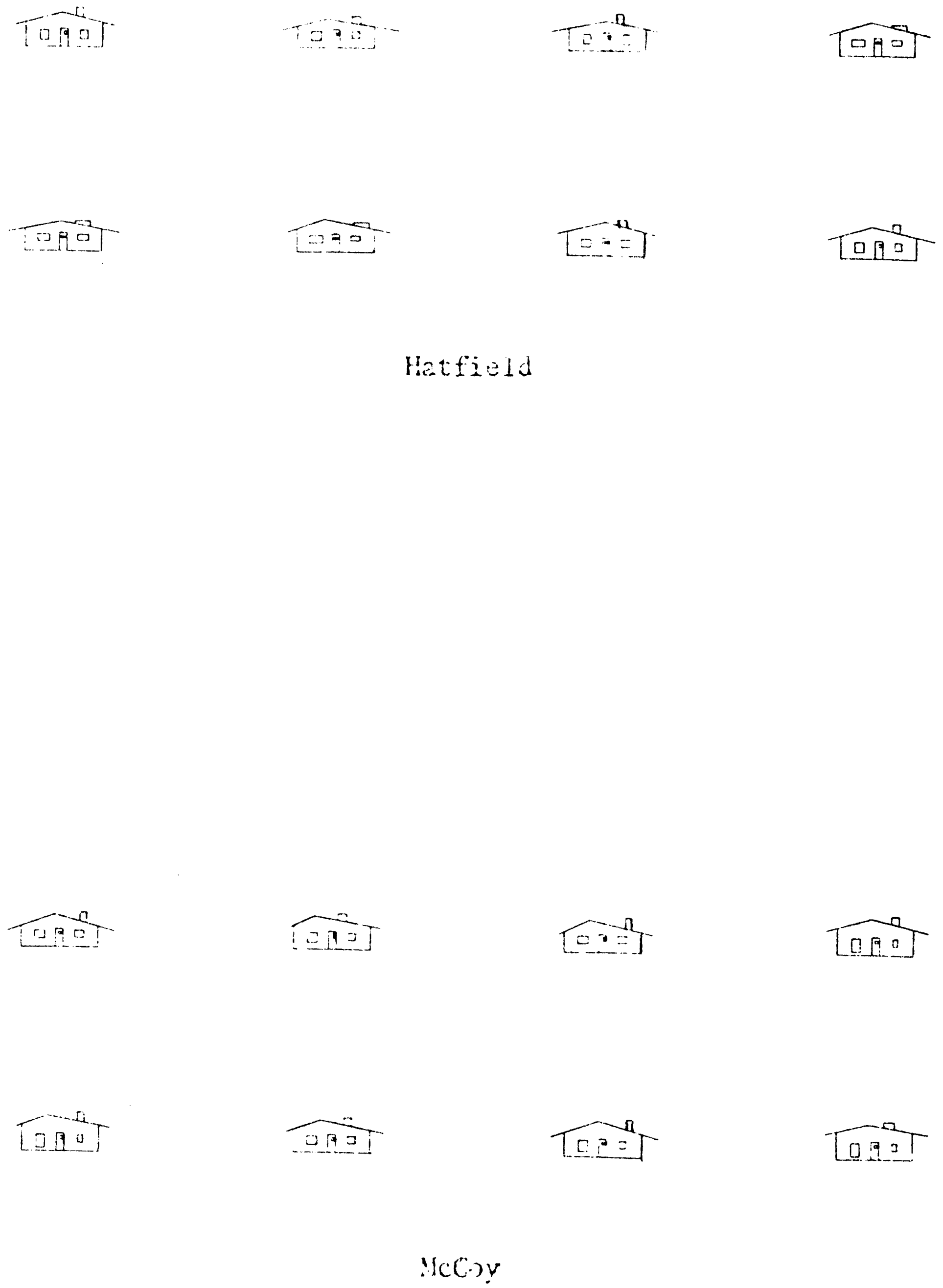
We have seen how simulation contributed to the experimental design by showing what sort of empirical test might produce significant results. The simulation also produced specific predictions to be tested in the experiment. The simulation and predictions are easier to understand, however, if one has a concrete picture of the task being simulated. For this reason, we will first describe the pattern recognition task, then explain the simulation with its predictions, and finally show the experimental results testing these predictions.

Stimuli and procedures

The stimuli were line figures of houses such as those shown in Figure 9. In order to design the houses to meet the needs of the experiment, a computer graphics program was developed which would accept a list of numbers giving dimensions of the house, and then display such a house on a cathode-ray tube. The first significant design problem was to establish 9 feature dimensions and a range of allowable values for each dimension such that any combination drawn from these ranges would look like a house. (For the mathematically minded, the cross-product of the 9 feature ranges can be thought of as a 9-dimensional hypercube, in which the coordinates of each point specify a complete set of feature values. The problem was then that each point in the hypercube had to specify a combination of features that looked like a house. The initial finding of the study was that there are a lot of points in a 9-dimensional hypercube.)

The next step was to reduce some of the ranges in order to give the different features roughly equal saliency, but significant results seemed most likely under such conditions. (This judgment was again based upon simulation experience.)

Once the design of the houses was fixed, the stimuli for the experiment were prepared by photographing the graphic display on 35mm high-contrast transparency film. The slides were developed and mounted with careful attention to avoiding extraneous cues such as intensity, orientation, and unintentional markings. When projected on a screen, the houses appeared as approximately 1 foot high by 2 feet wide black line figures on a white background.



Hatfield

McCoy

FIGURE 9. Training stimuli for participant group III

TABLE 1

The nine feature dimensions of the Hatfield-McCoy recognition task. The range and normalized scale are shown for each feature.

FEATURE NUMBER	FEATURE DESCRIPTION	RANGE OF FEATURE VALUES***	
		Zero value on normalized scale	100 value on normalized scale
1	Height (avg. of peak height and wall height, units*)	182	224
2	Steepness of roof (gable ht. / house width)	$\frac{3}{12}$	$\frac{5}{12}$
3	Horizontal extent of roof overhang / house width	$\frac{9}{24}$	$\frac{3}{24}$
4	Area of left window as % of total window area	49	75
5	Horizontal displacement of chimney from roof peak**	$\frac{4}{24}$	$\frac{12}{24}$
6	Displacement of roof peak to left of ctr. / house wid.	$\frac{-1}{24}$	$\frac{7}{24}$
7	Height of each window as % of (area of that window)$^{\frac{1}{2}}$	71	121
8	Chimney height, units*	31	81
9	Dist. from left edge of door to door pane, units*	14	34

*The cathode-ray-tube display is 1024x1024 units.

**as a fraction of peak-to-right-wall distance

***Allowable discrete values are spaced at intervals indicated by the denominator.

In the training phase of the study, the participants learn to classify a set of houses into two categories, labeled "Hatfield" and "McCoy" by the experimenter. Before presenting the details of the procedure, it is best to lay out the theoretical framework used to generate the examples of each category. The total range for each feature is normalized to a scale of 0-100, as labeled in the column headings of Table 1. Due to limitations of the graphics display, some of the features (such as slopes) could only assume a discrete set of values, which translates into a set of equally spaced points on the normalized 0-100 scale. We will speak of the feature ranges as continuous with the understanding that each continuous value is rounded to the nearest of the equally spaced discrete points.

We can now show how the examples given for each category were generated from a schema for that category. Each schema is operationally defined as a set of mean values and ranges for the set of 9 features. For clarity of exposition we will go through the schema definitions for participant group III before presenting the design in full generality.

Figure 10 shows the schema definitions for group III. The full range of each feature covers the 0-100 scale just defined. The figure is intended to show that Hatfield houses vary over the subrange 10-70 for each feature。 More precisely, the Hatfield schema is defined by giving a distribution for each of the 9 features. In this case, all features are uniformly distributed with mean 40 and range 60. In the McCoy schema, on the other hand, each feature is uniformly distributed with mean 60 and range 60. Note that for each

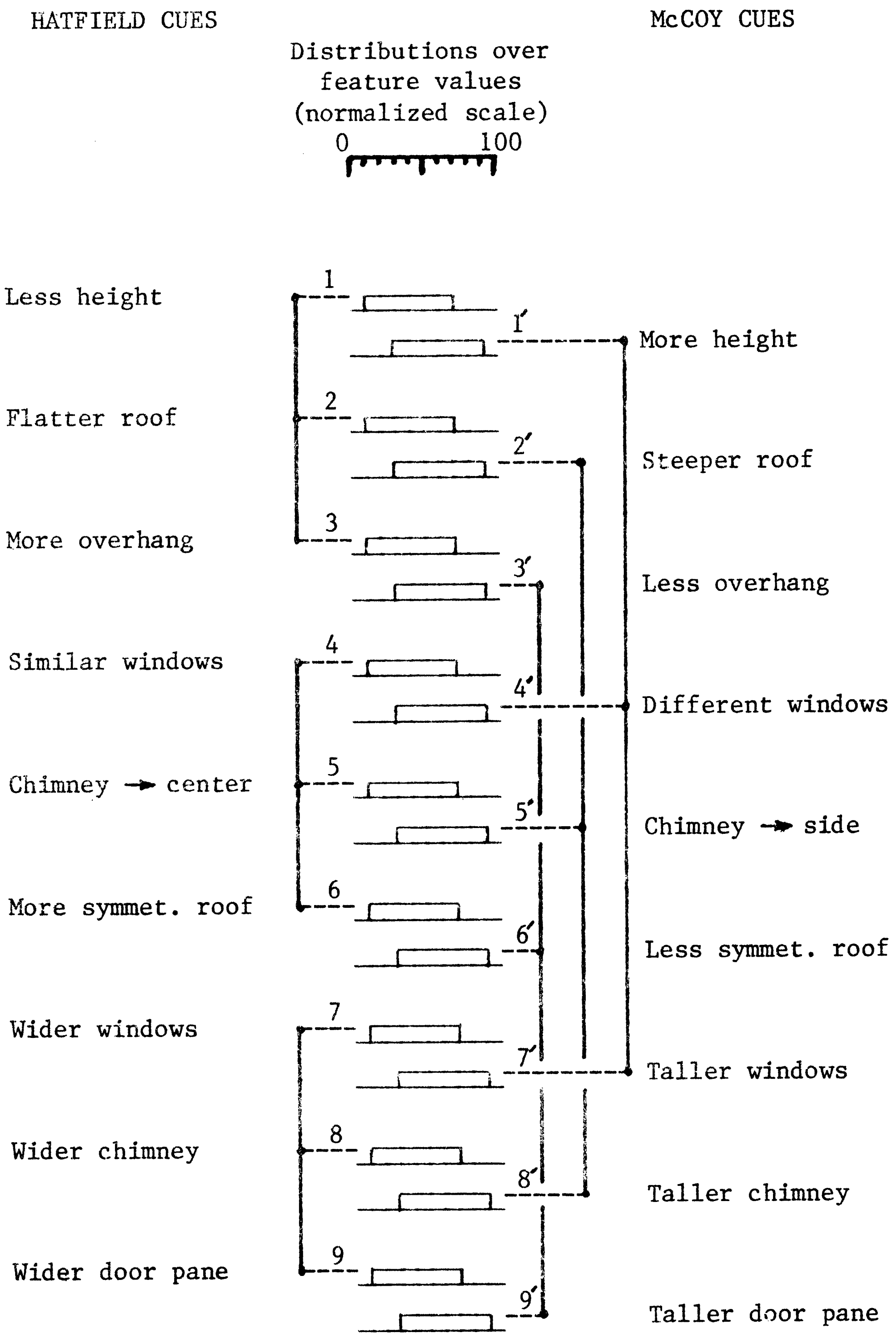


FIGURE 10. Definitions of schemata for participant group III.

feature, the distributions defined for Hatfield and McCoy overlap, so that no feature by itself is a reliable indicator for differentiating the two categories.

In formal terms, we are defining a schema as a probability distribution over the space of all possible houses. Each schema is used to generate a random sample of houses. A Hatfield house, for example, is generated by sampling a value for each feature using a random number generator with mean 40 and range 60. Our definition of such a schema is still not complete, however, because we have not specified the joint distributions of the features. In other words, we have not specified whether the distributions for different features are independent or correlated. Rosch *et al.* [1976] argue as follows:

> The world is structured because real-world attributes do not occur independently of each other. Creatures with feathers are more likely also to have wings than creatures with fur, and objects with the visual appearance of chairs are more likely to have functional sit-on-ableness than objects with the appearance of cats. That is, combinations of attributes of real objects do not occur uniformly. Some pairs, triples or ntuples are quite probable, appearing in combination sometimes with one, sometimes with another attribute; others are rare; others logically cannot or empirically do not occur.

The pattern recognition task here, and particularly the feature correlations in this task, were designed to maximally differentiate the predictions of the feature-network hypothesis from those of the feature-independence hypothesis. We will first describe the correlation patterns and then in the next section consider the resultant predictions.

To complete the definitions of the Hatfield and McCoy schema, we must specify correlations among the distributions in each. These are shown by dashed lines in Figure 10. Any two distributions connected by a dashed line are specified to have a correlation of .9; unconnected distributions are independent (have a correlation of 0).[4] The distribution and correlation information that defines each schema is also a specification for generating a random sample of houses for each category. A uniformly distributed random number generator is simply scaled to give the range specified for each feature. In order to introduce the correlations, a uniform random number is chosen to represent the common covariance in a correlated triplet. Each feature in the correlated triplet is then determined by the sum of this common variate and another random number generated for that feature alone. The precise procedure (which can be used to generate random variates with any desired correlation) is shown in the appendix.

Figure 10 specifies the correlations among the 9 Hatfield feature distributions, and among the 9 McCoy distributions. These correlation patterns may be easier to visualize if we arrange the distributions belonging to each schema in a geometrical array, as shown in Figure 11. (Positioning of features in the array is merely to show the correlations clearly, and has no other significance.) In general, the pattern shown in Figure 11a -- specifying a .9 correlation within but independence between the feature subsets {1,2,3},

[4] Mathematicians will note that pairwise correlations alone do not uniquely specify a joint probability distribution. We therefore specify that, within a correlated triplet, the three pairwise correlations and single triplet correlation are all accounted for by a single common source of variance (a single random variable).

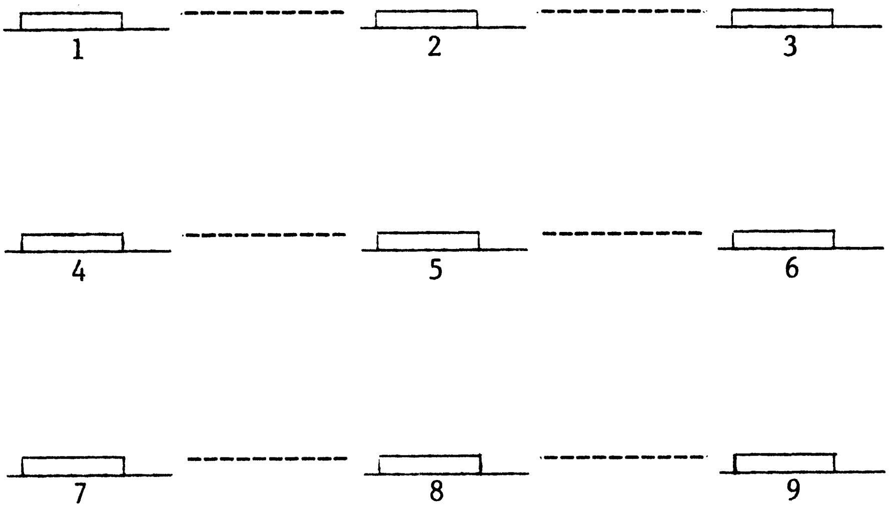


FIGURE 11a. The Hatfield schema with correlation pattern A. The feature distributions of Figure 10 are here arranged geometrically, in order to show the correlations more plainly.

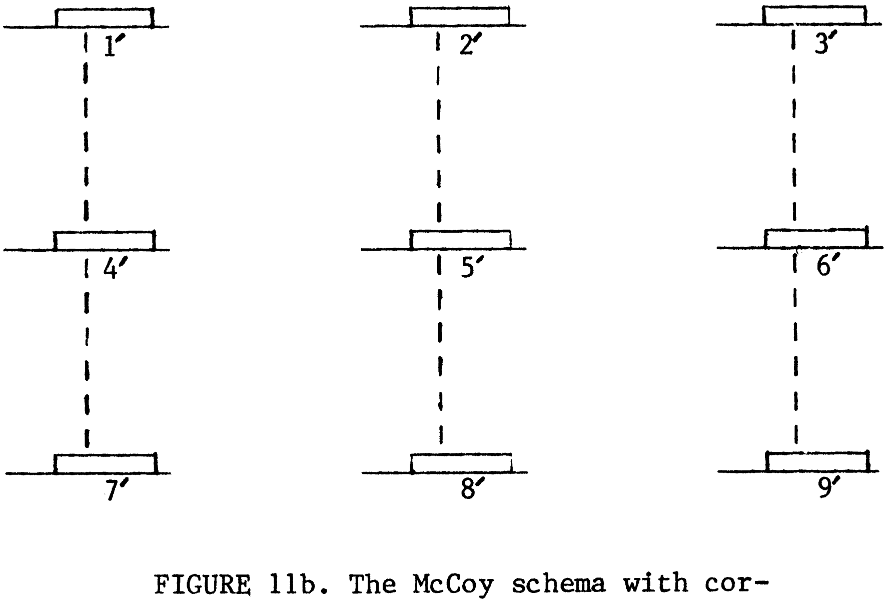


FIGURE 11b. The McCoy schema with correlation pattern B.

{4,5,6}, {7,8,9} -- will be termed correlation pattern A. Correlation pattern B, on the other hand, specifies a .9 correlation within but independence between the subsets {1,4,7}, {2,5,8}, {3,6,9}. When, as in Figure 11, we wish to emphasize not just the correlated features, but also the particular distributions of these features, then unprimed numbers will refer to distributions specified in the McCoy schema. Thus, the correlation pattern shown in Figure 11 can be identified simply as Hatfield-A, McCoy-B. The opposite correlation pattern shown in Figure 12 can be identified as Hatfield-B, McCoy-A.

In the context of the present experiment, we will always use uniform feature distributions with a range of 60 units somewhere on the 0-100 normalized scale. Within this context, therefore, the Hatfield and McCoy schemata are completely specified by giving

(1) the means of the feature distributions for each schema, and
(2) the correlation pattern (A or B) for each schema.

These are the two factors manipulated in the experimental design.

Participants. The participants in the study were 108 female undergraduates taking introductory psychology at the University of Michigan. Participation in the research was one (and probably the easiest) way of satisfying a course requirement of learning about empirical research.

Design. The two-factor 3 x 2 design is shown in Figure 13. Factor 1, with three levels, specifies the feature distribution means within each schema. The three levels were Hatfield-30, McCoy-70; Hatfield-40, McCoy-60; and Hatfield-50, McCoy-50. These means apply to each of the 9 features within the schema indicated.

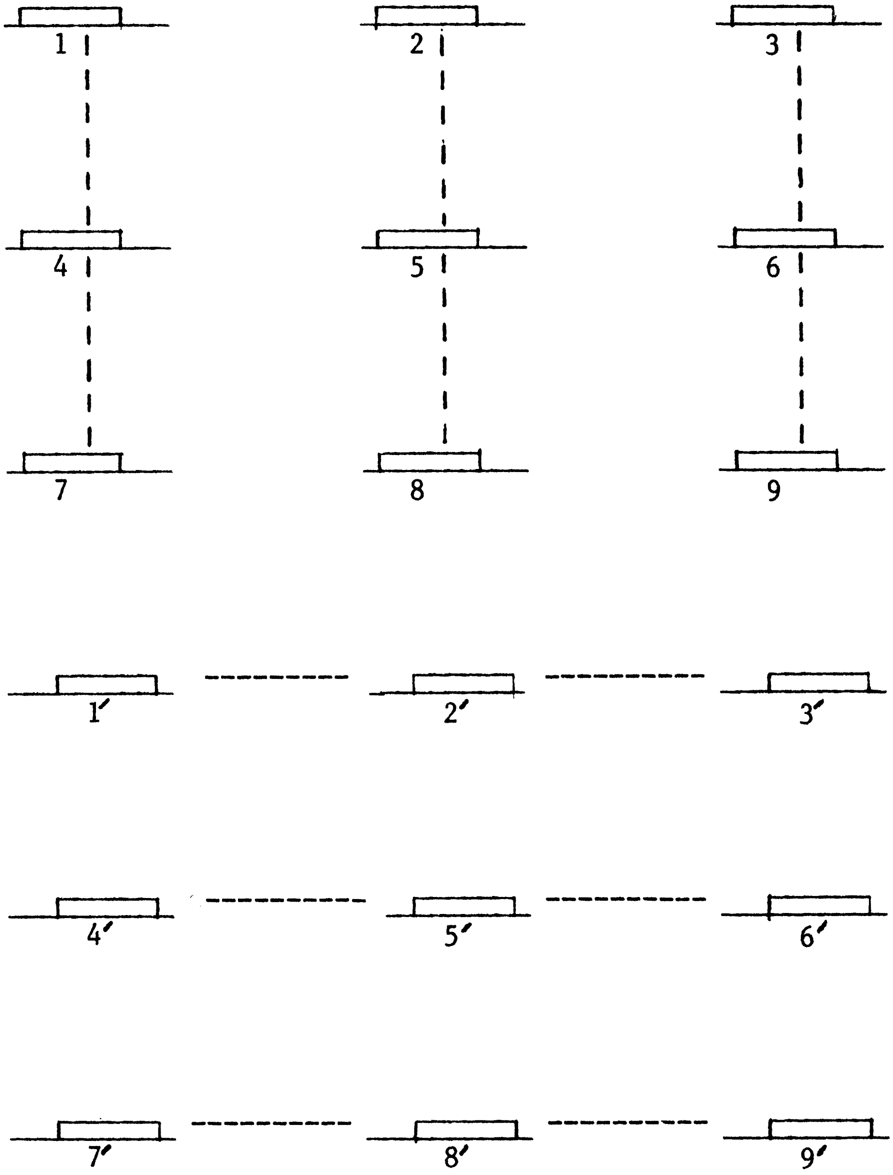


FIGURE 12. The opposite correlation pattern: Hatfield-B, McCoy-A. (Figure 11 gives the correlation pattern for participant groups I, III, and V; Figure 12 for groups II, IV, and VI.)

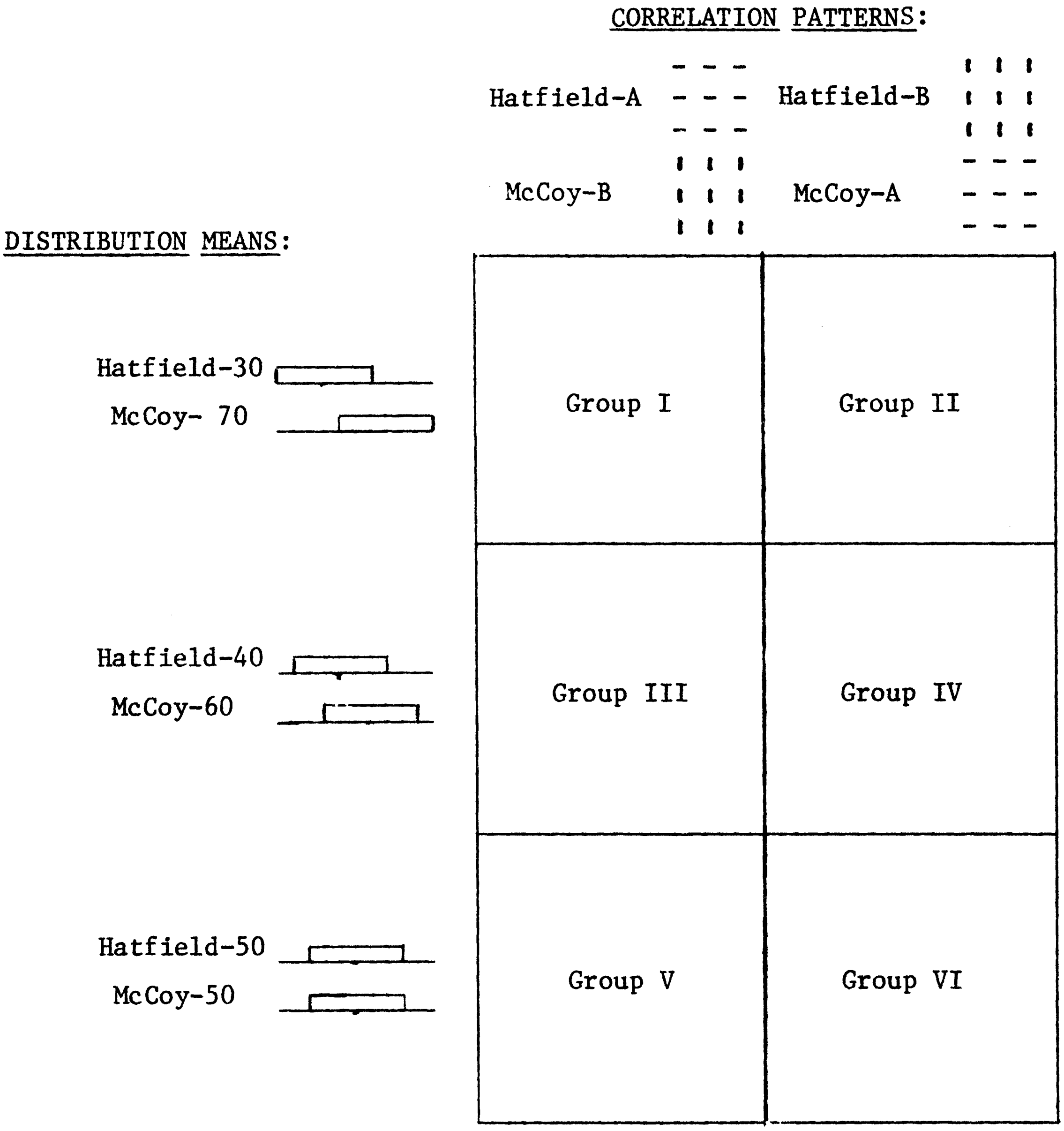


FIGURE 13. Factorial design of the experiment.

Factor 2, with 2 conditions, specifies the correlation pattern for each schema. The two conditions are Hatfield-A, McCoy-B (Figure 11) and Hatfield-B, McCoy-A (Figure 12). These two conditions are opposites in the sense that if two features of a schema are correlated under one condition, then they are independent under the other and vice-versa. The intuitive idea here (which will be made rigorous in the next section's discussion of the simulation) is that any effect due to the correlation patterns will be working oppositely under the two conditions. On the other hand, all other effects (due to the distributions of individual features, the comparative saliency of the features, the labels "Hatfield" and "McCoy", and the rest of the experimental context) will be the same across these two conditions.

The 3 x 2 design yields six participant groups, as shown in Figure 13. The factors determining each group define the Hatfield and McCoy schemata assigned to that group. Eight random samples were drawn from each schema, using a random number generator as explained previously. Thus, the training stimuli for each participant group consisted of 16 houses divided into two categories, Hatfield and McCoy. The whole set of 16 was repeated five times in different random orders for a total of 80 slides to be presented during the training phase. In order to minimize the effects of the particular ordering, the sequence of 80 slides was shown to half the participants in each group in forward order and to the other half in reverse order. There were at least 8 participants in each such half-group, and therefore at least 16 per group.

Training phase. Each session was conducted with 3 to 5 participants seated in a small room with a screen for viewing the slides. Participants were initially given the following written instructions:

> Let's say you have just accepted a job as a traveling salesman. Your territory is a remote mountainous region inhabited by two clans, the Hatfields and the McCoys. You'll undoubtedly get along much better in your new job if you can learn to recognize the type of houses that Hatfields live in, and the type McCoys live in. We're going to show you some examples of each and ask you to try to learn them.
>
> Altogether there will be 16 houses, 8 Hatfield and 8 McCoy. We'll show you the whole set twice to get you familiar with them. Then we'll show the whole set three more times (in different orders) and ask you to guess whether each house is Hatfield or McCoy.
>
> We'll give you an answer sheet when it's time to start guessing; until then, just watch. There are too many differences between the Hatfield and McCoy houses to analyze or keep track of, and we'd like you <u>not</u> to think about or analyze them. Don't think, just look!

The first 32 slides were each shown for a 6-second interval with the experimenter saying "Hatfield" or "McCoy" in the middle of the interval to identify the correct category. Participants were then given an answer sheet with the following instructions at the top:

> Now we'd like you to start guessing. For each house, please circle H if it looks like a Hatfield, M if it looks like a McCoy. Please try to guess quickly. After each house has been shown for about 5 seconds, the correct answer will be announced. If you haven't made your guess by this time, just cross out the question and go on.
>
> Note that your answers should go down each column. After the first two columns are finished, we'll stop announcing correct answers, and instead give you a lot of new houses to classify.
>
> Some of the houses may be difficult to recognize, so don't worry if you get them wrong. Just keep guessing as best you can.

The remaining 48 training slides were then each shown for a 13-second interval, with the experimenter identifying the correct category in the middle of the interval as before. All training slides were shown in contiguous intervals, except for a brief pause to pass out the answer sheets; and the test slides followed immediately, after changing slide trays on the projector.

Testing phase. Before showing the test slides, the experimenter said "Now you'll see some new houses, without feedback from me. You should be ready to start column 3." 75 test slides, not previously seen, were then shown for 7 seconds each, without comment by the experimenter. The 75 test slides were the same for all subject groups, and consisted of three repetitions of a set of 25 slides.

Each of these 25 houses had feature values meeting the following conditions: Arrange the nine numbered features in the usual geometric array

$$\begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{array}$$

and cross out any row and any column. Give the crossed-out features values of 50 (neutral) on the 0-100 scale. Of the four features not crossed out, take any two in the same row or same column and give them values of 17 (Hatfield cue). The remaining two features receive values of 83 (McCoy cue). It is possible to construct 36 distinct houses in this way, and 25 of these were selected in a pilot study as having indifferent enough responses on the basis of individual features alone to allow correlation information to affect the response.

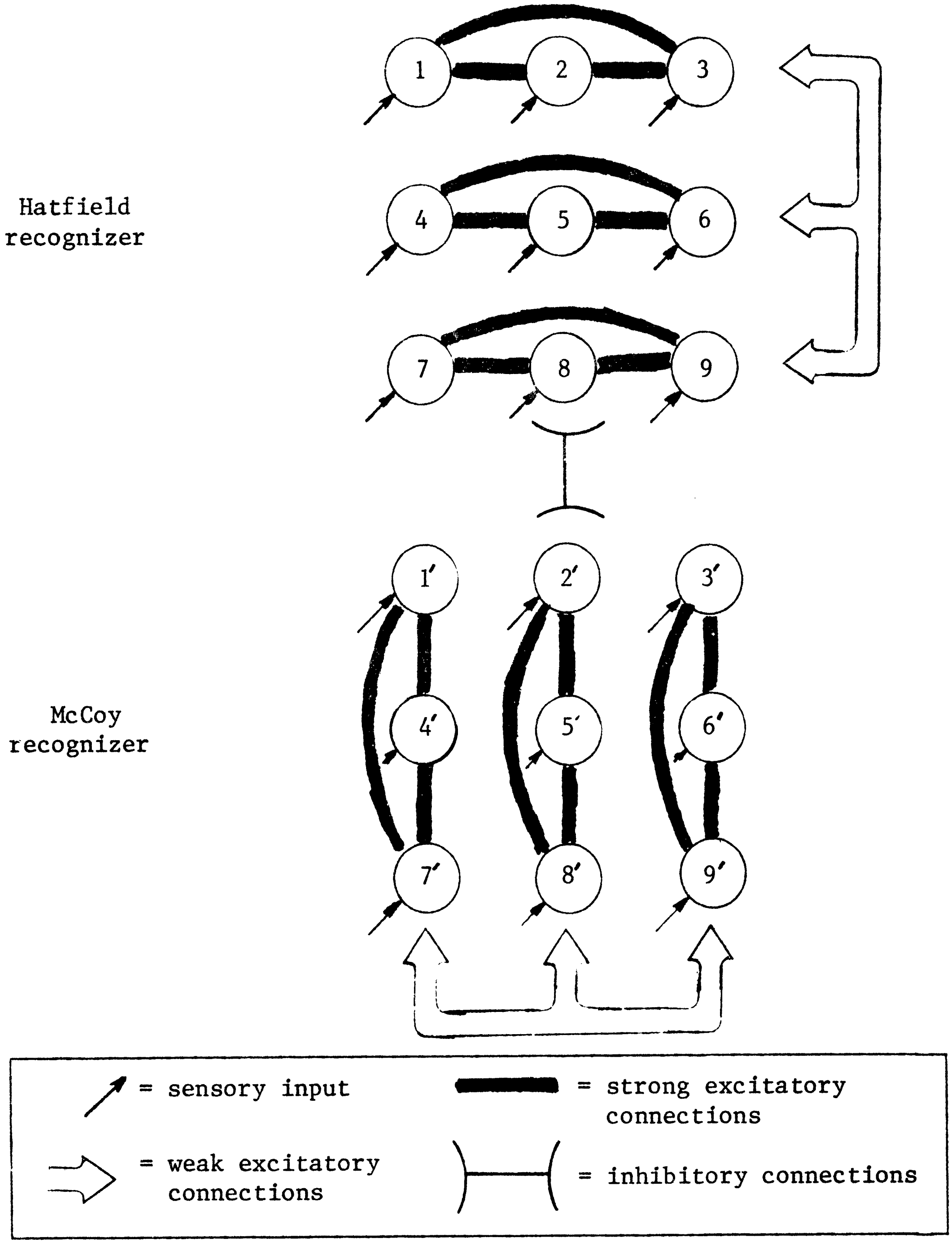


FIGURE 14. Neural network model used in the simulation for the Hatfield-A, McCoy-B condition. Note that the neural connections represent the environmental correlations of Figure 11. Exact connection strengths are shown in Table 2.

TABLE 2 (next page)

Connections in the lumped neural model of the Hatfield-McCoy pattern recognition task. Simulations of this model yielded predictions for the experiment. The neural structure consists of two associative networks, one for recognizing the "Hatfield" pattern and one for recognizing the "McCoy" pattern. There are excitatory connections within each associative network and inhibitory connections between the two networks, as shown in the table. (Because of their mutual inhibition, the two networks compete, the result after many time steps being that one is reverberatory and the other is nearly quiescent. Note that the purpose of this lumped model is to model the competition and predict the result, not to model subsequent fatigue and return to baseline activity.)

	Relation between features j and k in the environment	Connection between the corresponding netlets j and k in the neural structure	
		CONNECT_{jk}	CONNECTSQ_{jk}
Connection from a netlet to itself	j = k	.75	1.125
Connection between two netlets j and k belonging to the same recognition network	correlated	.06	.036
	independent	.02	.004
Connection between two netlets j and k which belong to competing recognition networks	mutually exclusive*	-.07	-.0245
	independent	-.2	-.04
Input connection to netlet k from an external sensory source j		.25	.3125

Predictions obtained from the netlet simulation

Having described the pattern recognition task in some detail, we can now look at the simulation of this task in a netlet model. Our model of the categorization process contains 18 netlets, 9 in the Hatfield network and 9 in the McCoy network. (See Figure 14 and Table 2.) Each netlet receives sensory input from one feature. Thus, for any feature in the environment, there are two netlets -- one Hatfield and one McCoy -- receiving input from that feature, but these two opposing netlets are maximally excited by opposite ends of the feature range. As one would expect from Figure 10, feature values in the middle of the range will mildly stimulate both netlets. On the other hand, feature values towards one side or the other will strongly stimulate the appropriate netlet and have no effect on the other. In short, each netlet labeled 1...9,1′...9′ in Figure 14 is sensitive to the correspondingly-numbered cue in Figure 10.

We will be modeling the postulated netlet mechanism as it exists just after the training trials. The *structure* of the netlet model will reflect information obtained from the training trials, according to hypothesis (ii) of the previous chapter. We will then simulate the *processing* done by this netlet structure upon receiving each of the test stimuli. Such processing will categorize each test stimulus as Hatfield or McCoy; therefore, our simulation will predict the categorizations made by humans in the experiment.

Knowledge base structure in the netlet model. The fundamental hypothesis (ii)d of the previous chapter postulates that whenever features are correlated in the environment, the neural representations of the features will become more strongly connected in the brain. This idea that a psychological contiguity learning rule can be derived from a neural contiguity learning rule goes back as far as James [1892], and was carefully worked out by Hebb [1949]. Hebb begins by postulating

> When an axon of cell A is near enough to excite a cell B and repeatedly or persistently take part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased. [p. 62]

He then presents a convincing argument as to the large-scale effect of such a mechanism on the ensemble of connections between two netlets. Stated in our terminology, the argument is that such a fine scale neural rule, applied over the ensemble of connections over a period of time, should produce a lumped connection strength (between the two netlets) that is indicative of the correlation between the environmental features represented by the netlets.

On the basis of this argument, we hypothesize that, at the conclusion of the training trials, the connection structure of the netlet model corresponds to the correlational structure of the task environment. In other words, if the environmental features are correlated as in Figure 10 or 11, then we postulate that the ultimate effect of Hebb's synaptic strengthening rule will be to produce the connection pattern shown in Figure 14.

In more theoretical terms, the structure of the netlet model is our postulated answer to the *knowledge representation* problem of Chapter 1. In our pattern-recognition task, the knowledge to be stored consists of the cues associated with each category, and the correlations among the cues. In the neural network model, each cue is represented by a netlet which receives input from the appropriate range of feature values. Cue correlations are represented by lumped connections between the netlets. In short, Figure 14 can be interpreted in two ways. On one hand, it shows the structure of a simple but realistic neural network with a transition function derived from the properties of its constituent neurons. On the other hand, Figure 14 shows the structure of the *knowledge base* that is coded into the neural network. From this point of view, each component of the neural structure stands for a cue or cue correlation in the world. Thus, the network structure of Figure 14 is a model of the environment defined by Figure 10. (We will return to this idea in the concluding chapter.) The precise correspondence between the neural interpretation and the knowledge base interpretation is, of course, made possible by the homomorphism derived in Chapters 3 and 4.

Knowledge base processing in the netlet model. A knowledge base will not be much good unless the information it contains can be accessed and applied to new situations. Since we do not allow a homunculus[5] to look over the new situation and decide what to pull out of the knowledge base, it is necessary to have the neural network itself do the processing. Let us state this requirement specifically for the present pattern recognition task: When a new stimulus is

[5] A homunculus in a mechanism is an intelligent little man inside who keeps things running smoothly, but in doing so begs the question of explaining intelligence. In today's theories, a homunculus sometimes takes the form of a black box labeled "central executive".

input to the network and the network simply follows its transition function (laid down in Chapter 4), the result should be an equivalence-classing operation based on information in the knowledge base.

When we look at the neural network as an information processor, the connections are not just sitting there representing knowledge; they are transmitting activity from one netlet to another. This is how the information represented in the network is put to use. More specifically, if netlets j and k represent cues j and k in a category, then the connection from netlet j to netlet k indicates the degree of prediction from cue j to cue k.[6] Thus, if cue j is present in the environment at a given time, it is reasonable to infer an increased likelihood for cue. The connection carries this partial inference by transmitting activity from netlet j to netlet k. The effect is that the detection of cue j lowers the threshold for detecting cue k. If there is an ambiguous or doubtful cue in the environment which *might* be cue k, then the presence of cue j

[6] This degree of prediction is P[k|j]/P[k], or equivalently P[j|k]/P[j], the likelihood ratio of Bayes' theorem. The lumped connection strength is assumed to be a monotonic function of the degree of prediction. This likelihood ratio definition, taken alone, suggests symmetric connections ($\text{CONNECT}_{jk} = \text{CONNECT}_{kj}$), which is the case in the Hatfield-McCoy model. If, however, one of the correlated features consistently preceded the other in time, then one would expect the connection representing the forward-time prediction to be stronger -- simply because predicting the future is of greater adaptive value than predicting the past.

helps[7] to infer that it *is* cue k.

In more functional terms, the network has learned that certain feature-combination-packages are common in the equivalent, and it attempts to resolve stimuli into whichever feature-combination-package is most easily inferred.[8] This happens because each feature-combination-package is represented in the neural network structure. This structure not only represents the distribution and correlation information about each package, but also performs the inference operation that transforms each stimulus into the most plausible package. This is, of course, the equivalence-classing operation we have been taking about since Chapter 2.

This equivalence-classing or categorizing operation is the testable consequence of our neural network hypothesis. In particular, we are testing whether feature-correlation information can change the categorization decision, even if feature means and distributions are held constant.

According to the feature-independence hypothesis, the information from each feature makes an independent additive contribution to the categorization decision. So if feature means and distributions are the same across Factor 2 (correlation pattern Hatfield-A, McCoy-B

[7] In most natural settings, it is assumed that converging facilitation from many cues would be necessary to activate (infer) a cue with weak environmental support. In the present experiment, there are only 9 cues altogether and convergence from only 2 is required to infer a strongly correlated third, but the principle is the same.

[8] As Bruner [1957] puts it, we tend to "go beyond the information given".

versus Hatfield-B, McCoy-A in the training examples), then the feature-independence hypothesis would not predict a significant variance in categorization decisions attributable to this factor.

In the feature-network model, individual feature distributions are still important, but they are not the whole story. Suppose, for example, that on the basis of feature distributions alone a stimulus has 2 Hatfield cues, 2 McCoy cues, and 5 neutral cues. If the stimulus should fit the Hatfield correlation pattern better than the McCoy pattern, it would be reasonable to categorize it as a Hatfield, and this is exactly what the netlet mechanism would do. This is not in itself a particularly strong prediction of the network model, since in general the feature-independence models could come up with a set of feature weights that would match the prediction. But if we reverse the correlation pattern used during training and look at the same test stimulus, any feature-independence model will still be stuck with the same prediction.[9] The feature-network hypothesis, on the other hand, would predict that any effect due to matching correlation patterns would be reversed by reversing the correlation patterns used during training.

The simulation confirms this line of reasoning, and also shows the mechanism which brings it about. Suppose, for example, that we are simulating the Hatfield-A, McCoy-B condition of the experiment,

[9] This is because, under the feature-independence assumption, the weight assigned to each feature is based only on its individual reliability as an indicator of the category. If the individual feature distributions each remain the same, then the individual feature reliabilities also remain the same.

i.e. the correlation patterns shown in Figure 11. Then our hypothesized neural network (as developed in the knowledge base structure subsection) is that shown in Figure 14. Let us investigate how this network processes a particular stimulus with 2 Hatfield cues and 2 McCoy cues. If features 1 and 2 are Hatfield cues, while 4 and 5 are McCoy cues, then netlets 1, 2, 4', and 5' will receive strong sensory input. (Netlets in both networks for features 3, 6, 7, 8, and 9 receive an equal amount of lesser stimulation, while the absent cues 1', 2', 4, and 5 receive none.) Now the fact that the Hatfield cues fit the Hatfield correlation pattern (Figure 11) means that these 2 cues both have strong excitatory connections to the third member of the triplet, netlet 3. In functional terms, cues 1 and 2 are both good predictors of cue 3. When netlets 1 and 2 become reverberatory, their outputs both converge upon netlet 3 and reduce its threshold so that the weak sensory input to netlet 3 is sufficient to fire it.

In other terms, we might say that the sensory information for feature 3 taken alone does not favor either cue 3 or cue 3', so netlets 3 and 3' receive equal small sensory inputs. On later time steps, however, netlets 1 and 2 become reverberatory, and the convergent facilitation from them gives netlet 3 an edge over netlet 3'.

Since the two McCoy cues do not fit the McCoy correlation pattern, their output does not converge upon a common third cue as in the Hatfield network. Instead, when netlets 4' and 5' become reverberatory, their output is dispersed among netlets 1', 2', 7', and 8', and none of these receiving netlets get enough of a boost to become reverberatory. The crux of the matter is the nonlinear threshold-like response curve for the netlet developed in Chapter 4 and shown in Figure 6b. This nonlinearity means that the network does not simply take a weighted sum of its inputs. If two cues fit a correlation pattern, then their combined predictive validity is greater than the sum of the individual predictive validities. The neural manifestation of this extra significance is that the convergent facilitation from the two cues is enough to boost the netlet detecting a third cue above its "threshold", into the range of self-sustaining reverberation.

The events just described are indeed what happens in the simulation. Once netlet 3 is boosted into reverberatory activity, the balance is tipped in favor of the Hatfield network. This small advantage is repeatedly magnified by the positive feedback of the excitatory connections within each network, and by the mutual inhibition between the two networks. After a few tens of time steps, an equilibrium state is reached in which all the Hatfield netlets are reverberatory, and all the McCoy netlets are at a low level of activity. The simulation thus predicts that the correlation effect would bias the participants' responses to this stimulus in the Hatfield direction.

Now if the same stimulus is presented to participants in the Hatfield-B, McCoy-A condition (Figure 12), then the two *McCoy* cues would fit the McCoy correlation pattern, and the responses would be biased in the McCoy direction. (This is confirmed by simulating a netlet structure with connections corresponding to Figure 12.)

Due to the symmetry of the neural connection patterns, all of the test stimuli described under *Testing phase* in the "Stimuli and Procedures" section yield simulation results just like one of the two cases above, except in cases where two Hatfield cues and two McCoy cues are in the same column rather than the same row. In these cases identical dynamics give the McCoy network an advantage under the Hatfield-A, McCoy-B condition, and the Hatfield network an advantage under the opposite condition.

Note that the simulation is predicting *only* the variance in categorization decision attributable to the different correlation patterns (Factor 2). In the actual experiment, there are many sources of variance, such as differential feature saliency and ease of learning the various correlations, but these factors are held constant over the two correlation conditions.

Summary of predictions. The purpose of the simulation is to predict the effect of training correlation patterns upon later responses to test stimuli. For each test stimulus, the simulation specifically predicts that one of the correlation patterns will bias the response in the Hatfield direction, while the other will bias the response in the McCoy direction. It is important that we are not predicting the actual categorization decision, which might be substantially affected by unequal feature saliency and correlation saliency. Rather, we are predicting the effect of the correlation patterns as measured by the *difference* in the categorization decisions across Factor 2. Finally, this effect is being tested at three levels of Factor 1, the separation between means for individual feature information alone, in which case we would expect the correlation patterns to be less noticeable, and the learning of these patterns to be less necessary to distinguish the two categories during training. As the separation between feature means decreases, the correlation information would be expected to play a larger role during training, and hence to have a larger effect on the response to the test stimuli.

Results and discussion

Each group originally included at least 16 participants, two half-groups (viewing the test stimuli in different orders) of at least 8 participants each. The predictions of the previous section are intended to apply not just to participants who learn the categories well, but to all participants who learn some information about them. For this reason, all participants who met a quite minimal training criterion were included in the analysis. This criterion included all participants whose final training score was chance or better (8 or more correct categorizations out of the final presentation of the 16 training stimuli). After imposing this criterion, at least 7 participants remained in each half-group. All half-groups were then brought to an equal size of 7 by randomly excluding participants from the larger half-groups. The resultant six groups of 14 participants each (Figure 14) were the basis of all further analyses.

Scoring. As described in the previous two sections, all test houses had 2 Hatfield cues, 2 McCoy cues, and 5 neutral cues. For the purpose of scoring, the test stimuli can be divided into two prediction classes, in which

(1) the two cues for each category are in the same row, or
(2) the two cues for each category are in the same column

where "row" and "column" refer to the usual convention for geometrically arranging the features:

1 2 3
4 5 6
7 8 9

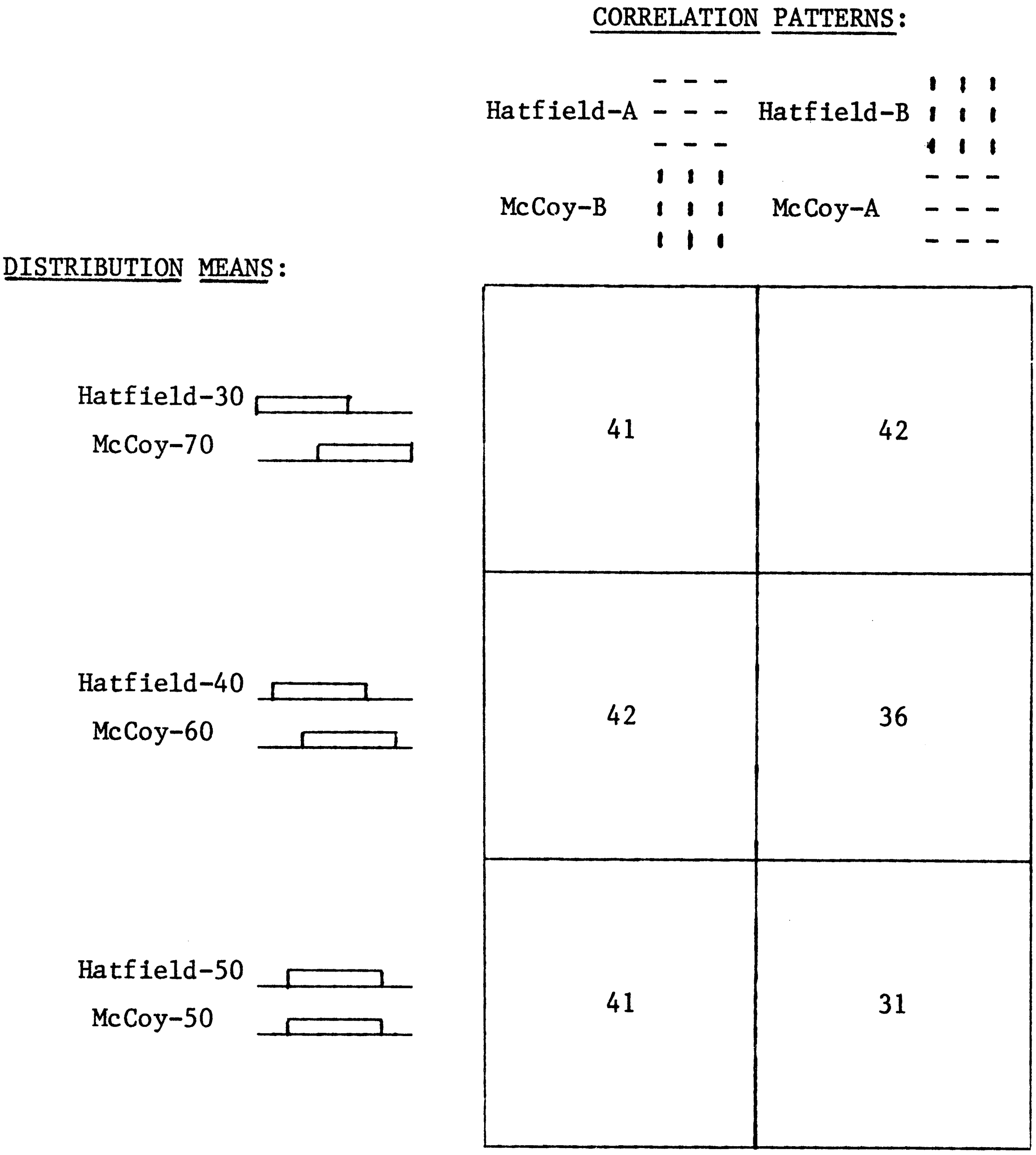
The participants' responses to the two classes of test stimuli were then scored as follows:

(1) Hatfield = 1, McCoy = 0.
(2) Hatfield = 0, McCoy = 1.

This scoring scheme allows the simulation predictions to be stated succinctly. The Hatfield-A, McCoy-B correlation pattern is predicted to *increase* the score for each test stimulus, while the Hatfield-B, McCoy-A pattern is predicted to *decrease* the score for each test stimulus. This uniform statement of the prediction allows us to add each participant's 75 scores and apply the prediction to the sum. Each participant's scores will then be in the range 0-75, and the prediction is that the scores will be significantly higher under the Hatfield-A, McCoy-B condition than under the opposite direction.

Statistical analysis. The mean participant's total score for each group is shown in Figure 15. The main prediction (of a directed[10] difference between mean scores of the two correlation conditions) was tested separately at each level of Factor 1. The two opposite-correlation groups at each level were compared with an ordinary t-test. At the highest level of Factor 1, corresponding to individual feature distributions with greatest separability (40 mean-mean distance) the slight difference between the two means was not significant ($p > .3$). At the medium and low levels of Factor 1, specifying separations of 20 and 0 between the feature distribution means, the differences were significant (at $p < .05$ and $p < .01$,

[10] The Hatfield-A, McCoy-B mean was predicted to be greater than the Hatfield-B, McCoy-A mean.



	Hatfield-A McCoy-B	Hatfield-B McCoy-A
Hatfield-30 McCoy-70	41	42
Hatfield-40 McCoy-60	42	36
Hatfield-50 McCoy-50	41	31

FIGURE 15. Experimental results.

TABLE 3

Analysis of variance for the experimental results shown in Figure 15.

SOURCE OF VARIANCE	df	MEAN SQUARE	F
MEAN	1	42380.25	-
BETWEEN PARTICIPANTS:			
(1) Separation of feature means	2	82.41	5.51**
(2) Diff. in correlation pattern	1	175.00	11.70***
(1) x (2) Interaction	2	70.51	4.72*
Between-participants error	78	14.95	-
WITHIN PARTICIPANTS:			
(3) Repeated measures	2	4.60	1.22
(1) x (3) Interaction	4	6.79	1.79
(2) x (3) Interaction	2	4.01	1.06
(1) x (2) x (3) Interaction	4	1.35	0.35
Within-participants error	156	3.78	-

*$p < .025$

**$p < .01$

***$p \approx .001$

respectively) in the predicted direction. The increasing significance of the difference with decreasing levels of Factor 1 suggests an interaction between Factors 1 and 2, which is further investigated in the following analysis of variance.

In addition to the between-participants Factors 1 and 2, a third within-participants factor was suggested because the 75 test slides consisted of 3 repetitions of a set of 25 houses. Factor 3 was thus created by summing a participant's scores separately for each block of 25 slides. The analysis of variance for Factors 1, 2, and 3 is shown in Table 3. The analysis included 84 participants (6 groups of 14), therefore allowing one degree of freedom for the grand mean and 83 remaining degrees of freedom between participants. The main effect predicted -- variance attributable to the difference in correlation patterns -- was highly significant(df = 1, F = 11.7, $p \approx .001$). Variance due to the different levels of separation between feature means was also significant (df = 2, F = 5.51, $p < .01$), and there was a significant (df = 2, F = 4.72, $p < .025$) interaction between the level of feature separation and the correlation factor. Neither the repeated measures factor nor its interactions with other factors were significant.

There was no significant correlation between the test scores and the earlier training performance (number correct) on the last set of training slides.

Discussion. The neural network model has apparently predicted a set of results that would not likely be predicted before the fact by other theories of feature integration. This does not of course imply that the neural network hypothesis is the only way to account for the results, but we shall present arguments that it is the most parsiminious and least *ad hoc* interpretation.

The interpretation closest to the data would consider the 9 individual features and their correlations as nothing more than operationally defined pieces of information presented to each participant during training. Under this interpreation, it seems clear that feature correlation information is picked up, held, and later used in the categorization decisions.

Disagreement over interpretation of the results is possible as soon as one begins to treat the features as psychological constructs -- as distinguishable units of information within the *participants*' heads, not just the investigator's. Since the hypothesis under consideration asserts that feature correlation information is learned and later used, it is important to ask whether an alternative dimensionalization of the feature space would eliminate the need to learn such correlations. In other words, we are asking whether the information previously termed "feature correlations" can be reasonably attributed to some set of individual feature distributions.

More specifically, we have seen that a correlated triplet of features in the experiment can be factored into a common source of variance (i.e. a common random variable) and three individual perturbations. Then why not consider each such common variate to be a psychological feature, avoiding the need to store correlation information? Such an explanation, taken by itself, depends upon an *ad hoc* specification of features. That is, a detector for any such common variate (unlike a detector for any of the 9 original features) would have to integrate information from non-contiguous loci of the sensory array. In view of the fact [Hubel and Wiesel, 1962 and a subsequent body of research reviewed by Thompson, 1967] that receptive fields for the first layers of visual processing are spatially contiguous, it would seem unreasonable to postulate detectors for noncontiguous features *a priori*.

On the other hand, such noncontiguous "common variate" features need not be postulated *a priori*, but could instead be postulated as being derived from some set of reasonable primitive features. With this modification, however, the feature-independence hypothesis is no longer distinguishable from a feature-correlation model. In fact, the neural network we have specified would be a quite adequate mechanism for such a two-stage primitive features→complex features→ categories recognition process. Primitive feature detectors would simply be interpreted as netlets, and complex feature detectors as strongly interconnected triplets of netlets.

To put the argument in more general terms, we are considering the possibility that the actual psychological features are different from the 9 original features we used to operationally characterize each house. It is quite plausible that the psychological features are different from, but linear combinations of, the operational features. Such a transformation would not affect our conclusions, however, since a linear function of a linear combination is still a linear function, and so under any such linear transformation the feature-independence hypothesis would still be unable to account for the experimental results. This means that some mechanism for storing and later using correlation information would still be necessary. If, on the other hand, we postulate psychological features complex enough to make the feature-independence hypothesis work, then we would be begging the question in the sense argued in detail in the three paragraphs preceding this one.

Let us therefore distinguish two degrees of conclusions that can be drawn from this study. On the basis of the data alone, taken in isolation, the conclusion must be limited to a statement about the information taken in, held, and later used by the participants. Such a statement appears at the beginning of this discussion subsection. But if our reasoning employs the additional constraint of neurophysiological plausibility, we can tentatively conclude that feature correlations *per se* are part of the knowledge base structure. In the absence of other proposals to account for such processing in neural terms, the netlet model is offered as a plausible explanation with predictive power.

CHAPTER 7
CONCLUSIONS

> My hypothesis then is that thought models, or parallels, reality...
>
> [Craik, 1943]

To understand any biological system, it is not enough to predict its behavior. Suppose, for example, that we had a complete and exact "wiring diagram" of the brain, showing how each neuron behaves and how it influences neighboring neurons. Would we then understand how that particular connection pattern gives rise to intelligent behavior? Understanding, in a biological context, means more than showing a mechanism -- it means seeing a relationship between mechanism and *function*. Within the constraints of physical and chemical laws, biological systems are designed by natural selection. Thus they are understood, not simply by giving the transition rule of the design, but also by asking what the design is accomplishing. The design could only have evolved if it makes some contribution to survival or reproduction. This contribution is what we mean by the function of the design [Williams, 1976].

What then is the function of the neural[1] design? It is to maintain an up-to-date model of the environment, and to insure that the appropriate parts of the model guide behavior in each new situation. That is the answer we have been driving at, though of course we have not proven it here. What we have done is to develop a technique by which neural structure and processing can be translated

[1] Cortical, to be more precise.

into knowledge base structure and processing -- by which design can be translated into function.

The cornerstone of this technique is the netlet. As the postulated unit of neural design, the netlet is a highly interconnected circuit capable of self-sustaining reverberation. The equivalence-classing performed by netlets themselves and by larger networks containing them is hypothesized to be the basic mechanism of pattern recognition. As suggested by Milner [1974] and as sketched in Figure 1, this mechanism may be iterated so that netlets deeper in the system (further from the sensory interface) are able to detect increasingly abstract concepts. Thus the netlet is advanced not only as a unit of neural design, but also as a functional unit: a component of the knowledge base that makes intelligent behavior possible.

Our modeling technique therefore allows Craik's internal model hypothesis to be expressed in mechanistic neural terms. Each netlet in the brain stands for the entity (anything from an elementary sensory feature to an abstract concept) that it detects; and the connection between two netlets stands for the environmental association between the entities they represent. Thus the specific synapse strengths, both within and between netlets, form a *structural code* that represents long-term information about the environment. Long-term memory, or knowledge base structure, is nothing more than this structure-code internal model.

Using the structural model in a particular situation, however, requires selecting those parts of the model relevant to the situation. This is accomplished by a second mechanism -- the *activity code*.

That is, netlets representing entities being perceived (or thought about) are distinguished simply be their activity levels[2] (i.e. neuronal firing rates). Thus the activity code, like the structure code, is a model of the environment -- a very short term model of the actual situation at hand.

The relation between these two codes (illustrated in the simulation of Chapter 6) is that activity flows through the structure, along the facilitatory connections. We have characterized this flow as *knowledge base processing* of the input. Activity coming from the sensory interface is channeled by the structure into various reverberatory circuits. In this way, the network analyzes a new situation in terms of those previously learned equivalence classes that are most similar to it. This allows the previously learned associates of these equivalence classes to come into play, since activity flows along associative connections. In particular, those associates receiving convergent excitation from several sources will become the most active on following time steps. These convergent associates represent the predictions or actions that best fit the current situation, on the basis of previous experience.

Thus, our general hypothesis is that thought (activity in reverberatory circuits) is a simplified, equivalence-classed model of the situation at hand, a model which can predict consequences of the current situation, and consequences of one's imagined actions. In short, organization of the knowledge base as an internal model allows

[2] Activity in a circuit representing an object may be experienced as either perceiving or imagining the object, depending upon the strength of activity. This explains Perky's [1910] findings.

one to form plans. This ability to run the model ahead in one's imagination -- to think about possible consequences before they occur -- is obviously adaptive. It was probably essential to early human survival in an environment where the good niches were already taken. It is undoubtedly essential to present-day human survival.

Summary. The basic neural mechanism for the internal model hypothesis was first proposed by Hebb. The purpose of the technique developed in Chapters 2 to 4 is to present this mechanism as a carefully constrained quantitative model, one which will yield precise predictions. Using the netlet (rather than the individual neuron) as the basic component allows simulation to be computationally tractable, and also allows the neural network to be understood in the functional terms of knowledge base structure and processing.

Chapters 5 and 6 show the application of the netlet model to a particular pattern recogntion task: simulating the hypothesized neural mechanism, and testing the resultant predictions in a psychological experient. The transition function shown in Chapter 4, however, is not limited to this application. Under the conditions for lumping (stated in Chapter 2), an arbitrary netlet structure can be modeled and simulated. Thus, as the internal model hypothesis is further developed into specific neural mechanisms, these mechanisms may be simulated using the same netlet transition function. The predictions obtained should allow further empirical tests of the netlet approach to the knowledge representation problem.

APPENDIX

Derivation of Lemmas 2 and 3

From Condition 4, we have the distribution of the threshold T_k within a single neuron, $P[T_k<x|\beta] = P[T_\beta<x] = \Phi_{a,\nu}(x)$. Using laws of probability, we can then combine the within-neuron threshold distributions for all neurons β in a netlet. (In the following summations, β ranges over B_k and $P[\beta]$ = 1/(number of neurons in B_k.)

$$\begin{aligned} P[T_k<x] &= \Sigma_\beta P[\beta]P[T_k<x|\beta] \\ &= \Sigma_\beta P[\beta]\Phi_{a,\nu}(x) \text{ where } a = m^{q_\beta} \\ &= \Sigma_\beta P[\beta]aN_\beta(x-\tau_0,\nu) + (1-a)N_\beta(x-\tau_\infty,\nu)] \\ &= E(a)N_\beta(x-\tau_0,\nu) + E(1-a)N_\beta(x-\tau_\infty,\nu) \text{ where } a = m^{R_k} \\ &= \Phi_{A,\nu}(x) \text{ where } A = E(m^{R_k}) \end{aligned}$$

This result is stated as Lemma 2 in the text. Next, the complement of the cumulative distribution of Z_k can be expressed

$$\begin{aligned} P[Z_k\geq z] &= P[X_k-T_k\geq z] \\ &= \int_x P[T_k<x-z]dP[X_k\leq x] \\ &= \int_x \Phi_{A,\nu}x-z)dP[X_k\leq x] \text{ where } A = E(m^{R_k}) \\ &= A\int_x N_\beta(x-z-\tau_0,\nu)dP[X_k\leq x] + (1-A)\int_x N_\beta(x-z-\tau_\infty,\nu)dP[X_k\leq x] \\ &= A\ P[\xi_\beta-X_k\leq -z-\tau_0] + (1-A)P[\xi_\beta-X_k\leq -z-\tau_\infty] \end{aligned}$$

by the definition of N_β in Condition 4. But by Condition 3, the distribution of $\xi_\beta - X_k = \xi_\beta - \Sigma_j X_{jk}$ can be approximated by a normal distribution. The mean of this distribution will be $E(\xi_\beta-X_k) = E(\xi_\beta) - E(X_k) = -E(X_k)$, and its variance ν' will be expressed in

Lemma 3. Thus we have

$$P[Z_k \geq z] = A \cdot N(E(X_k)-z-\tau_0,\nu') + (1-A)N(E(X_k)-z-\tau_\infty,\nu')$$
$$= \Phi_{A,\nu'}(E(X_k)-z)$$

In order to find ν' we need a general expression for the variance of $\Phi_{a,\nu}$:

Variance Lemma. For any a and ν $(0 \leq a \leq 1, \nu \geq 0)$ the variance of the distribution $\Phi_{a,\nu}$ is $\nu \div a(1-a)(\tau_0-\tau_\infty)^2$.

Proof. Let ξ be normally distributed with mean 0 and variance ν. Define a random variable τ independent of ξ such that $\tau=\tau_0$ with probability a and $\tau=\tau_\infty$ with probability (1-a). Then using the definition of $\Phi_{a,\nu}$

$$\Phi_{a,\nu}(x) = a\,P[\xi \leq x-\tau_0] + (1-a)P[\xi \leq x-\tau_\infty]$$
$$= P[\tau=\tau_0]P[\xi \leq x-\tau|\tau=\tau_0] + P[\tau=\tau_\infty]P[\xi \leq x-\tau|\tau=\tau_\infty]$$
$$= P[\xi \leq x-\tau]$$
$$= P[\xi+\tau \leq x] .$$

Therefore, since ξ and τ are independent, the variance of $\Phi_{a,\nu}$ is

$$\mathrm{Var}(\xi) + \mathrm{Var}(\tau) = \nu + E(\tau^2) - E^2(\tau) = \nu + a(1-a)(\tau_0-\tau_\infty)^2.$$

Lemma 3. $P[Z_k \geq z] = \Phi_{A,\nu'}(E(X_k)-z)$ where $A = E(m^{R_k})$ and $\nu' = \nu + \mathrm{Var}(X_k) - 2\mathrm{Cov}(X_k,T_k)$.

Proof. The Variance Lemma applied to the previously calculated distributions for T_k and Z_k yields $\mathrm{Var}(T_k) = \nu + A(1-A)(\tau_0-\tau_\infty)^2$ and $\mathrm{Var}(Z_k) = \nu' + A(1-A)(\tau_0-\tau_\infty)^2$ using the same value for A in each case. Therefore

$$\nu' - \nu = \mathrm{Var}(Z_k) - \mathrm{Var}(T_k)$$
$$= \mathrm{Var}(X_k-T_k) - \mathrm{Var}(T_k)$$
$$= \mathrm{Var}(X_k) + \mathrm{Var}(T_k) - 2\mathrm{Cov}(X_k,T_k) - \mathrm{Var}(T_k)$$

$$= \mathrm{Var}(X_k) - 2\mathrm{Cov}(X_k,T_k).$$

Derivation of Lemmas 4, 6, and 7

The microstructure of the input connections to a netlet clearly influences the microstate of that netlet. As a result, the microstate of the netlet will become correlated with the microstructure of the *input* connections, although not with the microstructure of the *output* connections (since the output connections are distributed independently of the input connections, according to Condition 6). The correlation between microstate and input microstructure is formally defined as the state variable COVARIANCE, and the effect of this COVARIANCE on later macrostates of the netlet is calculated in the next section.

We wish to show here that the effect of this correlation is confined to the netlet in which it occurs. In other words, we wish to show that the connections from one netlet to another carry only macrostate information, and not additional microstate information which might be correlated with the microstate of the receiving netlet. In brief, we will argue that the output connections, by virtue of their statistical independence from the input connections, carry a random sample of the netlet's activity which is uncorrelated with any inhomogeneities in activity resulting from the input distribution.

To make this argument more carefully, we will phrase it inductively:

Inductive Assertion. At any time step, the microstate of each netlet (i.e. the distribution of recovery states

> over the netlet) is independent of both (i) the distribution over the netlet of output synapse strengths to any other netlet, and (ii) the microstates of other netlets.

The argument (which includes two subsequent paragraphs) will be by induction over time steps. For the initial time step, the assertion is given by Condition 6. Next, assuming the whole assertion to be true at arbitrary time step t (our induction hypothesis), we must show that it continues to hold at time step t + 1. Now the state (i.e. complete neural state) of any netlet B at time t + 1 is determined by the states of itself and of its input neighbors at time t. Accordingly, for any two netlets B and B', let S be the set of netlets having input to B, and S' the set of netlets having input to B'. Thus the state of the netlets in S at time t will completely determine the state of B at time t + 1, and similarly for S' and B'. If S and S' were disjoint, then by the induction hypothesis the microstates of all netlets in S would be independent of all those in S'. In that case, the distributions of recovery states over B and B' at time t + 1 would arise from completely independent sources, and could not be correlated. The more difficult case is the case in which the sets S and S' overlap, meaning that one or more netlets B_i have input to both B and B'. It certainly seems conceivable that, in this case, the microstates of B and B' might be correlated due to their common source of input. But on more careful analysis, the input connections to B from B_i carry information from only a sample of B_i. The input connections to B' from B_i carry information from a different sample of B_i. Now these two samples arise from two different distributions of output synapses, and by Condition 6, these two distributions of synapses are independent in B_i. That is, the two sets of sample points (output synapses) are chosen independently. Furthermore, neither of these output synapse distributions is correlated with the distribution of recovery states over B_i at time t, by the induction hypothesis. Thus, the distribution of the output sent from B_i to B is independent of the distribution of output from B_i to B', and therefore the common input from B_i does not cause any correlation between the microstates of B and B' at time t + 1. This argument holds for any netlet B_i in $S \cap S'$. We conclude that even in the case where B and B' share one or more common input neighbors B_i, the independence of the microstates of B and B' at time t will be preserved at time t + 1.

Before proceeding to show that part (i) of the inductive assertion also continues to hold at time t + 1, it may be best to caution the reader that the assertion and its proof are concerned with microstates only, and not with macrostates. To clarify this distinction, consider again the situation of the previous paragraph in which a netlet B_i has one set of output synapses going to B, and another set going to B'. In effect we have two independent samples of the overall activity level (proportion of neurons firing) in B_i. We expect both of these samples to carry similar *macrostate* information; that is, the proportion of the neurons firing in each sample will closely approximate the proportion of neurons firing over B_i as a whole, by the law of large numbers. But our concern here is whether there is a correlation between the *microstates* of the two samples, and because of the statistical independence of the samples, the answer is no. In short, the firing neurons within the first sample are distributed independently of the firing neurons within the second sample.

So far, assuming that the inductive assertion is true at time t, we have shown that part (ii) continues to hold at time t + 1. It remains to show that part (i) also continues to hold at time t + 1. The distributions of output synapses to various netlets from any netlet B do not change over time, so the question is whether the microstate of netlet B will become correlated with any of its output synapse distributions. Since such a correlation does not exist at time t (by the induction hypothesis), it could only arise at time t + 1 if some trace of the distribution of the netlet's earlier *output* returns as *input* to the netlet at time t + 1. We claim, however, that such a trace could not be carried through any intervening netlets. More precisely, even if the microstate of some intervening netlet at time t were correlated with netlet B's output synapse distribution, this correlation could not be passed from the intervening netlet back into the microstate of netlet B. The reason is that if such a correlation were passed, then at time t + 1 the microstates of the intervening netlet and of netlet B would both contain traces of (i.e. be correlated with) the output synapse distribution of B, and therefore the microstates of both these netlets would be correlated with each other, contradicting the independence we have already shown (in the previous paragraph) between the microstates of any two netlets at time t + 1. (This argument depends upon the fact that the microstate of the intervening netlet at time t affects the threshold distribution and therefore the microstate of the same netlet at time t + 1; meaning that the microstate of the intervening netlet is correlated with the output synapse distribution of B at time t + 1.) We have thus demonstrated our claim that no information correlated with the distribution of B's output synapses could be passed back into netlet B via any intervening netlet. The only remaining possibility is that such correlation information might be passed via the synapses from B to itself. However, Condition 6 guarantees that the distribution of neurons in B which receive input from B is independent of the distribution of neurons in B which send output to B, and that this independence also applies to the distribution of strengths of the synapses involved. In other words, if we take the set of synapses from netlet B into itself, then the distribution of these synapses over neurons in the netlet when each synapse is associated with its presynaptic neuron is independent of the distribution of these synapses when each synapse is associated with its postsynaptic neuron. Therefore, since the input to B that samples the activity of B itself is distributed independently of the sample taken (i.e. independently of the output synapse distribution); since the microstate of B at time t is also independent of this output synapse distribution (by the induction hypothesis); and since the distribution of inputs from other sources maintains this same independence (as shown in the first part of this paragraph); we can therefore conclude that the resultant microstate of netlet B at time t + 1 continues to be independent of the distribution of output synapses over netlet B. In sum, we have shown that both parts (i) and (ii) of the inductive assertion continue to hold at time t + 1, and therefore by induction the assertion is true for all time steps t.

If we consider the distribution of synapses from any netlet B_j to any netlet B_k, then part (i) of the inductive assertion tells us that the postsynaptic neurons (in B_k) receive a fair sample of the activity in B_j, fair in the sense that the synapses and their strengths are distributed independently of the activity in R_j that they sample, and therefore do not distort the sample. To translate this assertion into terminology that will fit right into the derivation of Chapter 4, recall that the input X_{jk} (defined in Chapter 3) received by any neuron in B_k is determined by two factors: the recovery states $R_j(H_{njk})$, $1 \leq n \leq \dot{N}$, of the neurons in B_j that are sampled, and the corresponding strengths S_{njk} of the synapses that do the sampling. Our assertion simply implies that these two factors, $R_j(H_{njk})$ and S_{njk}, are independent, and this conclusion is formally stated in Lemma 4.

Having shown the independence of the two factors that determine a single input component X_{jk}, we can now proceed to show independence between the different components X_{jk} of the input coming from different netlets B_j. As we have just seen, for each netlet B_j (having input to B_k), X_{jk} is determined by two independent factors, the distribution of recovery states $R_j(H_{njk})$, and the distribution of the corresponding synapses S_{njk}. But for different netlets B_j, their distributions of recovery states $R_j(H_{njk})$ are independent by part (ii) of the inductive assertion; and their distributions of synapse strengths S_{njk} are independent by Condition 6. Thus the input components X_{jk} for all j are independent.

Moreover, if we look at any of the input netlets B_j one time step later, its microstate (i.e. distribution of recovery states) $R_j^{\delta}(H_{njk})$ could not be correlated with the earlier microstate $R_{j'}(H_{njk})$ of any of the other input netlets $B_{j'}$. The reason is that the later microstate $R_{j'}^{\delta}(H_{njk})$ of $B_{j'}$ is partly determined by its earlier microstate $R_{j'}(H_{njk})$, and therefore if the microstate of B_j were correlated with $R_{j'}(H_{njk})$, it would also be correlated with $R_{j'}^{\delta}(H_{njk})$. But this implied correlation between $R_j^{\delta}(H_{njk})$ and $R_{j'}^{\delta}(H_{njk})$ would be between two netlets at the same time step, and would therefore violate part (ii) of our inductive assertion. Therefore $R_j^{\delta}(H_{njk})$ is independent of $R_{j'}(H_{njk})$for $j'=j$. Now we showed in the previous paragraph that any input X_{jk} is independent of the other inputs $X_{j'k}$, $j' \neq j$. This independence can now be extended to the next time step input X_{jk}^{δ}, which depends upon the same fixed connections S_{njk} as X_{jk}, and upon the new microstate $R_j^{\delta}(H_{njk})$ which was just shown to satisfy the same independence conditions as the earlier microstate $R_j(H_{njk})$. The conclusions of this and the previous paragraph are stated as Lemma 6 in Chapter 4.

Finally, knowing that the different input components X_{jk} are independent allows us to compute the variance of their sum:

Lemma 7. $\mathrm{Var}(X_k) = \Sigma_j(\mathrm{CONNECTSQ}_{jk}\cdot\mathrm{ACTIVITY}_j - \mathrm{CONNECT}_{jk}^2\cdot\mathrm{ACTIVITY}_j^2)$

To verify this formula, we first note that $X_k = \Sigma_j X_{jk}$; and because the separate inputs X_{jk} are independent, their variances are additive:

$$
\begin{aligned}
\mathrm{Var}(X_k) &= \Sigma_j \mathrm{Var}(X_{jk}) \\
&= \Sigma_j [E(X^2_{jk}) - E^2(X_{jk})] \\
&= \Sigma_{ij} E(S^2_{njk} \cdot \lambda^2(R_j(H_{njk}))) - \Sigma_j E^2(X_{jk}) \\
&= \Sigma_{ij} E(S^2_{njk}) \cdot E(\lambda^2(R_j(H_{njk}))) - \Sigma_j E^2(X_{jk})
\end{aligned}
$$

because the two factors S_{njk} and $R_j(H_{njk})$ that determine each X_{jk} are independent by Lemma 4. Next we use the fact that the output function λ is restricted to the values 0 and 1, and therefore $\lambda^2 = \lambda$. This means that $E(\lambda^2(R_j(H_{njk}))) = E(\lambda(R_j(H_{njk}))) = \mathrm{ACTIVITY}_j$, so the formula for variance now reads

$$
\begin{aligned}
\mathrm{Var}(X_k) &= \Sigma_{ij} E(S^2_{njk}) \cdot \mathrm{ACTIVITY}_j - \Sigma_j E^2(X_{jk}) \\
&= \Sigma_j \mathrm{ACTIVITY}_j \Sigma_i E(S^2_{njk}) - \Sigma_j E^2(X_{jk}) \\
&= \Sigma_j \mathrm{ACTIVITY}_j \cdot \mathrm{CONNECTSQ}_{jk} - \mathrm{CONNECT}^2_{jk} \cdot \mathrm{ACTIVITY}^2_j)
\end{aligned}
$$

where the first term is obtained using equations (9) and (11), and the second term uses the expression for $E(X_{jk})$ obtained in the development of equation (18).

Derivation of the COVARIANCE component of the transition function

The state variable $\mathrm{COVARIANCE}_k$ represents the covariance between the input and threshold distributions--in symbols, $\mathrm{Cov}(X_k, T_k)$. The role of this quantity in the ACTIVITY transition function was developed in Lemma 3 and formally given in Lemma 5.

Thus we must express the next-time-step $\mathrm{COVARIANCE}^\delta_k = \mathrm{Cov}(X^\delta_k, T^\delta_k)$ solely in terms of the current-step ACTIVITY, RECOVERY, and COVARIANCE. Before plunging into the detailed formal derivation, it might be helpful to explain our approach in intuitive terms. The covariance between X^δ_k and T^δ_k is easiest to analyze if we divide up the input X_k according to netlet of origin, as shown in Figure 4 (near the end of Chapter 3). For example, consider how the input X_{jk} from netlet j will be distributed over netlet k. Presumably only a small fraction of the neurons in netlet k will receive any input from netlet j at all, and each neuron which does receive such input will receive it from its own sample $\{H_{njk}\}_{1\leq n\leq N}$ of neurons in netlet j. Furthermore, the strengths S_{njk} of these synaptic connections may also vary, as discussed just before equation (8). These considerations mean that the input X_{jk} will tend to be distributed inhomogeneously over netlet k; and since H_{njk} and S_{njk} are fixed structural attributes of the model, they will tend to produce the same inhomogeneities from one time step to the next (i.e. the same inhomogeneities in X_{jk} and X^{δ}_{jk}). This is precisely why we cannot assume that X^{δ}_{jk} and T^{δ}_{k} are independent: Each input component X_{jk} received by a neuron in the netlet influences the probability of firing, and hence the next-time-step recovery state, and hence the next-time-step threshold T^{δ}_{k}. This means that some trace of the inhomogeneity in the input X_{jk} will still be present in the next-time-step threshold T^{δ}_{k}. Thus the next-time-step input X^{δ}_{jk} over the netlet will not be acting upon an independent distribution of thresholds, but rather there will be some degree of correspondence between the distributions of X^{δ}_{jk} and T^{δ}_{k}, because both are influenced by the same structural attributes H_{njk} and S_{njk}.

In more formal terminology, we are saying that part of the variance in X^{δ}_{jk} and part of the variance in T^{δ}_{k} are accounted for by a common source: the distribution of the preceding input X_{jk}. X_{jk}

accounts for part of the variance in X^{δ}_{jk} because these successive inputs come in through the same connections H_{njk}. On the other hand, X^{δ}_{jk} is independent of the inputs $X_{j'k}$, $j' \neq j$, because these inputs from other netlets come through different random connections $H_{nj'k}$ which are distributed independently over netlet k. A rigorous version of this argument (shown in the previous section) demonstrated

<u>Lemma 6</u>. For any netlet k, the random variables X_{jk} (for all netlets j with some input neighbors to netlet k) are mutually independent. Furthermore, any X^{δ}_{jk} is independent of all inputs $X_{j'k}$ from different netlets $j' \neq j$.

To analyze the variance in T^{δ}_{k}, we note from Condition 4 that the threshold of a neuron depends upon its recovery state q_{β} (later translated into the random variable R_k) and upon a noise component ξ_{β} unaffected by recovery states or inputs. If we use the general notation = ρ( ) to mean "is a function of", then we can succinctly express this dependence as

$$\begin{aligned} T^{\delta}_{k} &= \rho(R^{\delta}_{k}, \xi) \\ &= \rho(\Sigma_{j'} X_{j'k}, T_k, \xi) \text{ by Lemma 1; def. of } X_{jk} \\ &= \rho(X_{jk}, T_k, \Sigma_{j' \neq j} X_{j'k}, \xi) \end{aligned}$$

Thus, if we factor out the dependence of T^{δ}_{k} upon X_{jk} and T_k (as shown in equation (2) below), the remaining variance will be completely determined by the variables $X_{j'k}$, $j' \neq j$ and ξ; and by Lemma 6, X^{δ}_{jk} will be independent of this remaining variance in T_k. We will use this conditional independence to obtain equation (21).

We now proceed to divide $\text{Cov}(X_k, T_k)$ into its components and to factor out the common dependence in each. For any random variable Y, define $\hat{Y} = Y - E(Y)$. Using laws of probability we obtain

$$\begin{aligned}\mathrm{Cov}(X_k^\delta,T_k^\delta) &= E(\hat{X}_k^\delta\cdot\hat{T}_k^\delta)\\ &= E(\Sigma_j\hat{X}_{jk}^\delta\cdot\hat{T}_k^\delta)\\ &= \Sigma_j E(\hat{X}_{jk}\cdot\hat{T}_k^\delta)\\ &= \Sigma_{j,x,t}P[X_{jk}=x,T_k=t]\cdot E(\hat{X}_{jk}^\delta\cdot\hat{T}_k^\delta|X_{jk}=x,T_k=t) \qquad (20)\end{aligned}$$

For compactness we will let a lower case x above (within square brackets or after a vertical bar) stand for the event $X_{jk} = x$ in the sample space B_k, and similarly a lower case t above will stand for $T_k = t$. Equation (20) then becomes

$$\begin{aligned}\mathrm{Cov}(X_k^\delta,T_k^\delta) &= \Sigma_{j,x,t}P[x,t]\cdot E(\hat{X}_{jk}^\delta\cdot\hat{T}_k^\delta|x,t)\\ &= \Sigma_{j,x,t}P[x,t]\cdot E(\hat{X}_{jk}^\delta|x,t)\cdot E(\hat{T}_k^\delta|x,t) \qquad (21)\end{aligned}$$

by the conditional independence shown in the previous paragraph. Next observe that the t in $E(\hat{X}_{jk}^\delta|x,t)$ is superfluous, since the only part of the variance in T_k that helps to predict $\hat{X}_{jk}^\delta$ is that part which is correlated with X_{jk}, and we already have the condition $X_{jk} = x$. We therefore eliminate the superfluous condition and simplify:

$$\begin{aligned}\mathrm{Cov}(X_k^\delta,T_k^\delta) &= \Sigma_{j,x,t}P[x]\cdot P[t|x]\cdot E(\hat{X}_{jk}^\delta|x)\cdot E(\hat{T}_k^\delta|x,t)\\ &= \Sigma_{j,x}P[x]\cdot E(\hat{X}_{jk}^\delta|x)\Sigma_t P[t|x]\cdot E(\hat{T}_k^\delta|x,t)\\ &= \Sigma_{j,x}P[x]\cdot E(\hat{X}_{jk}^\delta|x)\cdot E(\hat{T}_k^\delta|x) \qquad (22)\end{aligned}$$

Now from equation (4), $E(T_k) = E(m^{R_k})\tau_0 + [1-E(m^{R_k})]\tau_\infty = \tau_\infty + (\tau_0-\tau_\infty)E(m^{R_k})$ and therefore $E(T_k^\delta) = \tau_\infty + (\tau_0-\tau_\infty)E(m^{R_k^\delta})$. We can also derive $P[T_k^\delta<x|x]$ using the same steps as in the development of

Lemma 2, except that the summation is only over neurons β for which $X_{jk} = x$. This will yield $E(T_k^\delta|x) = \tau_\infty + (\tau_0-\tau_\infty)E(m^{R_k^\delta}|x)$ and therefore

$$E(\hat{T}_k^\delta|x) = E(T_k^\delta|x) - E(T_k^\delta)$$
$$= [\tau_0-\tau_\infty]\cdot[E(m^{R_k^\delta}|x) - E(m^{R_k^\delta})] \ . \tag{23}$$

We have already calculated $E(m^{R_k^\delta})$ in equation (19). Plugging the definitions of ACTIVITY_k^δ and RECOVERY_k^δ into that equation gives

$$E(m^{R_k^\delta}) = P[R_k^\delta=0] + m\cdot P[R_k^\delta\neq 0]\cdot E(m^{R_k}) \tag{24}$$

The steps from equation (19) just used to calculate $E(m^{R_k^\delta})$ can also be used to calculate $E(m^{R_k^\delta}|x)$, by simply carrying along the extra condition. Thus we have

$$E(m^{R_k^\delta}|x) = P[R_k^\delta=0|x] + m\cdot P[R_k^\delta\neq 0|x]\cdot E(m^{R_k}|x) \tag{25}$$

and the difference to be evaluated in equation (23) becomes

$$E(m^{R_k^\delta}|x) - E(m^{R_k^\delta}) = U + V \tag{26}$$

where

$$U = P[R_k^\delta=0|x] - P[R_k^\delta=0] \text{ and}$$

$$V = m\cdot P[R_k^\delta\neq 0|x]\cdot E(m^{R_k}|x) - m\cdot P[R_k^\delta\neq 0]\cdot E(m^{R_k}) \ .$$

Now following the analysis just above equation (23) for $E(T_k)$ rather than $E(T_k^\delta)$ gives

$$E(T_k) = \tau_\infty + \Delta\tau\cdot E(m^{R_k}) \quad \text{and} \tag{27}$$

$$E(T_k|x) = \tau_\infty + \Delta\tau\cdot E(m^{R_k}|x) \tag{28}$$

where for convenience we define $\Delta\tau = \tau_0 - \tau_\infty$ and $\Delta\tau^{-1} = 1/(\tau_0-\tau_\infty)$. Solving equations (27) and (28) for $E(m^{R_k})$ and $E(m^{R_k}|x)$, respectively, yields

$$E(m^{R_k}) = \Delta\tau^{-1}[E(T_k)-\tau_\infty] \text{ and}$$

$$E(m^{R_k}|x) = \Delta\tau^{-1}[E(T_k|x]-\tau_\infty] \ .$$

Substituting these into our expression for V above, we get

$$\begin{aligned} V = {} & m\Delta\tau^{-1}(P[R_k^\delta\neq 0|x]\cdot E(T_k|x) - P[R_k^\delta\neq 0]E(T_k)) - \\ & m\Delta\tau^{-1}\tau_\infty(P[R_k^\delta\neq 0|x]-P[R_k^\delta\neq 0]) \ . \end{aligned} \tag{29}$$

Now note that $P[R_k^\delta\neq 0|x] = 1 - P[R_k^\delta=0|x] = 1 - U - P[R_k^\delta=0] = P[R_k^\delta\neq 0] - U$. If we make this substitution in equation (29), we can then factor out the common term $P[R_k^\delta\neq 0]$ to obtain

$$\begin{aligned} V &= m\Delta\tau^{-1}(P[R_k^\delta\neq 0]\cdot(E(T_k|x)-E(T_k)) - E(T_k|x)\cdot U) + m\Delta\tau^{-1}\tau_\infty U \\ &= m\Delta\tau^{-1}(P[R_k^\delta\neq 0]\cdot(E(T_k|x)-E(T_k)) - (E(T_k|x)-\tau_\infty)\cdot U) \\ &= m\Delta\tau^{-1}((1-\text{ACTIVITY}_k^\delta)\cdot E(\hat{T}_k|x) - (E(T_k|x)-\tau_\infty)\cdot U) \ . \end{aligned} \tag{30}$$

Next we will evaluate

$$\begin{aligned} P[R_k^\delta=0|x] &= P[Z_k\geq 0|x] && \text{by eq. (3)} \\ &= \Phi_{A,\nu'}(E(X_k|x)) && \text{by Lemma 3} \\ &= \Phi_{A,\nu'}(\Sigma_{j'}E(X_{j'k}|x)) && \text{by def. of } X_k \\ &= \Phi_{A,\nu'}(\Sigma_{j'}E(X_{j'k}|X_{jk}=x)) \\ &= \Phi_{A,\nu'}(\Sigma_{j'\neq j}E(X_{j'k})+x) \\ &= \Phi_{A,\nu'}(\Sigma_{j'}E(X_{j'k})+x-E(X_{jk})) \\ &= \Phi_{A,\nu'}(E(X_k)+x-E(X_{jk})) \end{aligned} \tag{31}$$

where the parameters A and ν' have the same values as in Lemma 3: $A = E(m^{R_k})$, and $\nu' = \nu + \text{Var}(X_k) - 2\cdot\text{Cov}(X_k,T_k)$. Therefore

$$U = P[R_k^\delta=0|x] - P[R_k^\delta=0]$$

$$= \Phi_{A,\nu'}(E(X_k)+x-E(X_{jk})) - \Phi_{A,\nu'}(E(X_k)) . \qquad (32)$$

where the two terms were obtained from equations (31) and (17), respectively. If we recall that $\Phi_{A,\nu'}$ is a cumulative distribution, then the expression for U above is simply the integral of the density $\Phi'_{A,\nu'}$ over the interval from $E(X_k)$ to $E(X_k) + x - E(X_{jk})$. Now all the terms we are evaluating occur within the summation Σ_x of equation (22) where x ranges over the values that X_{jk} can take with nonzero probability. Therefore we can apply approximation (ii) of Condition 3 to obtain

$$U = (x-E(X_{jk}))\cdot\Phi'_{A,\nu'}(E(X_k)) .$$

With these expressions for U and V we can return to equation (23) and substitute

$$\begin{aligned} E(\hat{T}_k^\delta\ x) &= \Delta\tau\cdot[E(m^{R_k^\delta}|x)-E(m^{R_k^\delta})] \\ &= \Delta\tau\cdot(U+V) \\ &= \Delta\tau\cdot U + m(1-\text{ACTIVITY}_k^\delta)\cdot E(\hat{T}_k|x) - m(E(T_k|x)-\tau_\infty)U \qquad (33) \end{aligned}$$

which in turn can be substituted into equation (22). The result is

$$\begin{aligned} \text{Cov}(X_k^\delta,T_k^\delta) = {} & \Sigma_{j,x}P[x]\cdot E(\hat{X}_{jk}^\delta|x)\cdot\Delta\tau\cdot(x-E(X_{jk})\cdot\Phi'_{A,\nu'}(E(X_k)) + \\ & \Sigma_{j,k}P[x]\cdot E(\hat{X}_{jk}^\delta|x)\cdot m(1-\text{ACTIVITY}_k^\delta)\cdot E(\hat{T}_k|x) - \\ & \Sigma_{j,k}P[x]\cdot E(\hat{X}_{jk}^\delta|x)\cdot m(E(T_k|x)-\tau_\infty)(x-E(X_{jk})\cdot\Phi'_{A,\nu'}(E(X_k)). \end{aligned}$$

We will now recombine the probabilities conditional on $X_{jk} = x$ that are summed over the possible values of x. Simplifying each summation separately, we obtain

$$\begin{aligned}\mathrm{Cov}(X_k^\delta,T_k^\delta) = {} & \Delta\tau\cdot\Phi'_{A,\nu'}(E(X_k))\cdot\Sigma_j E(\hat{X}_{jk}^\delta\cdot\hat{X}_{jk}) + \\ & m(1-\mathrm{ACTIVITY}_k^\delta)\cdot\Sigma_j E(\hat{X}_{jk}^\delta\cdot\hat{T}_k) + \\ & m\Phi'_{A,\nu'}(E(X_k))\cdot\Sigma_j E(\hat{X}_{jk}^\delta\cdot\hat{X}_{jk}\cdot[T_k-\tau_\infty])\end{aligned} \qquad (34)$$

Using definitions from probability theory, $E(\hat{X}_{jk}^\delta\cdot X_{jk}^\delta) = \mathrm{Cov}(X_{jk}^\delta,X_{jk}) = \frac{1}{2}[\mathrm{Var}(X_{jk}^\delta) + \mathrm{Var}(X_{jk}) - \mathrm{Var}(X_{jk}^\delta-X_{jk})]$, where the last term may be neglected by Condition 5(i). Thus we can approximate $E(\hat{X}_{jk}^\delta\cdot X_{jk}^\delta)$ in the first line of equation (34) by $\frac{1}{2}[\mathrm{Var}(X_{jk}^\delta) + \mathrm{Var}(X_{jk})]$. We also note that inputs and thresholds are correlated only because the threshold distribution contains a trace of the previous input distribution, and therefore the covariance between either of the inputs $\hat{X}_{jk}^\delta$ and $\hat{X}_{jk}$ and the threshold T_k is completely accounted for within the covariance of the two inputs. Thus we can replace $E(\hat{X}_{jk}^\delta\cdot\hat{X}_{jk}\cdot[T_k-\tau_\infty])$ with $E(\hat{X}_{jk}^\delta\cdot\hat{X}_{jk})\cdot E(T_k-\tau_\infty)$ without losing any covariance. The approximation for $E(\hat{X}_{jk}^\delta\cdot\hat{X}_{jk})$ can thus be applied to the first and third terms of equation (34), and these two terms can then be combined to yield

$$\begin{aligned}\mathrm{Cov}(X_k^\delta,T_k^\delta) = {} & \tfrac{1}{2}(\Delta\tau+m(E(T_k)-\tau_\infty))\Phi'_{A,\nu'}(E(X_k))\cdot\Sigma_j[\mathrm{Var}(X_{jk}^\delta)+\mathrm{Var}(X_{jk})] + \\ & m(1-\mathrm{ACTIVITY}_k^\delta)\cdot\Sigma_j E(\hat{X}_{jk}^\delta\cdot\hat{T}_k)\ .\end{aligned} \qquad (35)$$

Next we note that $E(\hat{X}_{jk}^\delta\cdot\hat{T}_k) = E((\hat{X}_{jk}+\Delta\hat{X}_{jk})\cdot\hat{T}_k) = E(\hat{X}_{jk}\cdot\hat{T}_k) + E(\Delta\hat{X}_{jk}\cdot\hat{T}_k)$, where the second term $E(\Delta\hat{X}_{jk}\cdot\hat{T}_k) = \mathrm{Cov}(X_{jk}^\delta-X_{jk},T_k)$ can be ignored by Condition 5(ii). Hence the summation on the second line of equation (35) reduces to a sum of covariances

$$\begin{aligned}\Sigma_j E(\hat{X}_{jk}\cdot\hat{T}_k) &= \Sigma_j \mathrm{Cov}(X_{jk},T_k) \\ &= \mathrm{Cov}(X_k,T_k)\end{aligned} \qquad (36)$$

by the independence of the X_{jk}'s shown in Lemma 6. This same independence allows us to express the summation on the top line of equation (35) as

$$\begin{aligned}\Sigma_j[\mathrm{Var}(X^\delta_{jk}) + \mathrm{Var}(X_{jk})] &= \Sigma_j\mathrm{Var}(X^\delta_{jk}) + \Sigma_j\mathrm{Var}(X_{jk}) \\ &= \mathrm{Var}(X^\delta_k) + \mathrm{Var}(X_k) \ . \qquad (37)\end{aligned}$$

Substituting equations (36) and (37) into equation (35) yields

$$\mathrm{Cov}(X^\delta_k,T^\delta_k) = \tfrac{1}{2}(\Delta\tau+m(E(T_k)-\tau_\infty))\Phi'_{A,\nu'}(E(X_k))\cdot(\mathrm{Var}(X^\delta_k)+\mathrm{Var}(X_k)) + m(1-\mathrm{ACTIVITY}^\delta_k)\cdot\mathrm{Cov}(X_k,T_k) \ .$$

Finally, recall that $\mathrm{COVARIANCE}^\delta_k$ and $\mathrm{COVARIANCE}_k$ are defined as $\mathrm{Cov}(X^\delta_k,T^\delta_k)$ and $\mathrm{Cov}(X_k,T_k)$ respectively. Also, we showed just after equation (22) that $E(T_k) = \tau_\infty + (\tau_0-\tau_\infty)E(m^{R_k}) = \tau_\infty + \Delta\tau\cdot\mathrm{RECOVERY}_k$ from the definition of $\Delta\tau$ and $\mathrm{RECOVERY}_k$. Substituting this expression for $E(T_k)$, we obtain

$$\mathrm{COVARIANCE}^\delta_k = \tfrac{1}{2}\Delta\tau(1+m\cdot\mathrm{RECOVERY}_k)\Phi'_{A,\nu'}(E(X_k))\cdot(\mathrm{Var}(X^\delta_k)+\mathrm{Var}(X_k)) + m(1-\mathrm{ACTIVITY}^\delta_k)\cdot\mathrm{COVARIANCE}_k$$

where the parameters A and ν' still have the same values as in Lemma 3: $A = E(m^{R_k}) = \mathrm{RECOVERY}_k$, and $\nu' = \nu + \mathrm{Var}(X_k) - 2\cdot\mathrm{COVARIANCE}_k$.

At this point we could expand $E(X_k)$, $\mathrm{Var}(X_k)$, $\mathrm{Var}(X^\delta_k)$, and $\mathrm{ACTIVITY}^\delta_k$ in terms of the state variables form which they must be calculated, according to equation (17), Lemma 7, and Lemma 5. Such an expansion would express the next-time-step COVARIANCE directly in terms of the current-time-step ACTIVITY, RECOVERY, and COVARIANCE (and the fixed parameters of the model). But since these same expectation and variance calculations also occur as subexpressions in the calculation of $\mathrm{ACTIVITY}_k^\delta$ and $\mathrm{COVARIANCE}_k^\delta$, it makes sense to calculate these subexpressions as intermediate values which are then plugged into the expressions for $\mathrm{ACTIVITY}_k^\delta$, $\mathrm{RECOVERY}_k^\delta$, and $\mathrm{COVARIANCE}_k^\delta$. Such a computational sequence is specified in the concluding section of Chapter 4.

Generation of Correlated Random Variates

Begin with three independent random variables X_0, X_1, and X_2, where each X_i is distributed uniformly over the interval from $-L_i$ to L_i. In other words, each X_i has density f_i where $f_i(x) = \frac{1}{2L_i}$ if x is in the interval $(-L_i,L_i)$; 0 otherwise. Then the variance of each X_i is the integral of $X_i^2 f_i(x)dx$ over the interval, which turns out to be $L_i^2/3$.

Now define for i=1,2 $Y_i = X_0+X_i$. Then the variance of Y_i is $(L_0^2+L_i^2)/3$, and the covariance of Y_1 and Y_2 is

$$\begin{aligned} \mathrm{Cov}(Y_1,Y_2) &= E(X_0+X_1)(X_0+X_2)) - E(X_0+X_1)E(X_0+X_2) \\ &= E(X_0^2) \text{ since each } E(X_i) \text{ is } 0 \\ &= \mathrm{Var}(X_0) \end{aligned}$$

which makes sense since X_0 is the common source of variance in Y_1 and Y_2. The correlation between Y_1 and Y_2 is this covariance divided by the square root of the product of the individual variances of Y_1 and Y_2, i.e. $\mathrm{Cov}(Y_1,Y_2)\ /\ (\mathrm{Var}(Y_1)\mathrm{Var}(Y_2))^{\frac{1}{2}}$

$$\begin{aligned} &= (L_0^2/3)\ /\ ((L_0^2 + L_1^2)(L_0^2 + L_2^2)/9)^{\frac{1}{2}} \\ &= L_0^2 / ((L_0^2 + L_1^2)(L_0^2 + L_2^2))^{\frac{1}{2}}. \end{aligned}$$

Now suppose that $L_1 = L_2 = kL_0$. These values, when plugged into the formula just above, yield a correlation between Y_1 and Y_2 which is simply $1/(1+k)^2$. Thus k can be varied to yield any desired correlation between Y_1 and Y_2. (To be more specific, setting $k = (1/c -1)^{\frac{1}{2}}$ will yield a correlation of c.)

Thus, to obtain any desired correlation c between Y_1 and Y_2, begin by generating three independent random variables X_0, X_1, and X_2 where all three are picked from uniform distributions, but the ranges for the distributions of X_1 ahd X_2 are each k times the range for X_0, $k = ((1/c)-1)^{\frac{1}{2}}$. Then $Y_1=X_0+X_1$ and $Y_2=X_0+X_2$ will have the desired correlation c.

REFERENCES

Anderson, J.A. (1972) A simple neural network generating an interactive memory. Mathematical Biosciences 14 197-220.

Anderson, J.A. (in press, 1977) Neural models with cognitive implications. In LaBerge, D. and Samuels, S.J. (eds.) Basic Processes in Reading: Perception and Comprehension. Erlbaum Associates, Potomac, Md.

Attneave, F. (1957) Transfer of experience with a class-schema to identification-learning of patterns and shapes. Journal of Experimental Psychology 54 81-88.

Bruner, J.S. (1957) Going beyond the information given. In Bruner *et al.*, Contemporary Approaches to Cognition, Harvard University Press, Boston.

Brunswick, E. (1956) Perception and the Representative Design of Psychological Experiments (2nd ed.), University of California Press, Berkeley.

Conrad, C. (1974) Context effects in sentence comprehension: A study of the subjective lexicon. Memory and Cognition 2 130-138.

Cooper, L.N. (1974) A possible organization of animal memory and learning. In Lundquist, B. and Lundquist, S. (eds.) Proceedings of the Nobel Symposium on Collective Properties of Physical Systems, Academic Press, New York.

Craik, K.J.W. (1943) The Nature of Explanation, The University Press, Cambridge, England.

Finley, M.R. Jr. (1967) An experimental study of the formation and development of Hebbian cell-assemblies by means of a neural network simulation. Unpublished doctoral dissertation, Department of Computer and Communication Sciences, University of Michigan, Ann Arbor.

Franks, J.J. and Bransford, J.D. (1971) Abstraction of visual patterns. Journal of Experimental Psychology 90 65-74.

Harth, E.M., Csermely, T.J., Beek, B. and Lindsay, R.D. (1970) Brain functions and neural dynamics. Journal of Theoretical Biology 26 93-120.

Hebb, D.O. (1949) The Organization of Behavior: A Neuropsychological Theory, Wiley, New York.

Hubel, D.H. and Wiesel, T.N. (1962) Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex. Journal of Physiology (London) 160 106-154.

James, William (1892) Psychology: The Briefer Course, Harper Torchbook edition, New York, 1961.

Kaplan, S. (1973) Cognitive maps in perception and thought. In Downs, R.M. and Stea, D. (eds.) Image and Environment: Cognitive Mapping and Spatial Behavior, Aldine, Chicago.

Kaplan, S. (1976) Adaptation, structure, and knowledge. In Moore, G.T. and Golledge, R.G. (eds.) Environmental Knowing: Theories, Research, and Methods, Dowden, Hutchinson and Ross, Stroudsburg, PA.

Kuhn, T.S. (1962) The Structure of Scientific Revolutions, University of Chicago Press, Chicago.

Milner, P.M. (1957) The cell-assembly: Mark II. Psychological Review 64 242-252.

Milner, P.M. (1974) A model for visual shape recognition. Psychological Review 81 521-535.

Mountcastle, V.B. (1957) Modality and topographic properties of single neurons of cat's somatic sensory cortex. J. Neurophysiol. 20 408-434.

Perky, C.W. (1910) An experimental study of imagination. American Journal of Psychology 21 422-452.

Posner, M.I. (1973) Cognition: An Introduction, Scott, Foresman and Company, Glenview, Ill.

Posner, M.I. and Keele, S.W. (1968) On the genesis of abstract ideas. Journal of Experimental Psychology 77 353-363.

Reed, S.K. (1972) Pattern recognition and categorization. Cognitive Psychology 3 382-407.

Reed, S.K. (1973) Psychological Processes in Pattern Recognition, Academic Press, New York. Part 4: Perceptual classification (see page 186).

Rodwan, A.S. and Hake, H.W. (1964) The discriminant function as a model for perception. American Journal of Psychology 7 380-392.

Rosch, E. (1977) Human categorization. In Warren, N. (ed.) Studies in Cross-cultural Psychology (vol. 1), Academic Press, London.

Rosch, E. and Mervis, C.B. (1975) Family resemblances: Studies in the internal structure of categories. Cognitive Psychology 7 573-605.

Rosch, E., Mervis, C.B., Gray, W.D., Johnson, D., and Boyes-Braem, P. (1976) Basic objects in natural categories. Cognitive Psychology 8 382-439.

Runkel, P.J. and McGrath, J.E. (1972) Research on Human Behavior: A Systematic Guide to Method, Holt, Rinehart, and Winston, New York (Section 4-1: A framework for comparing some major research strategies).

Thompson, R.F. (1967) Foundations of Physiological Psychology, Harper and Row, New York.

Tversky, A. and Karntz, D.H. (1969) Similarity of schematic faces: A test of interdimensional additivity. Perception and Psychophysics 5 124-128.

Wiener, N. (1948) Cybernetics, or Control and Communication in the Animal and the Machine, Massachusetts Institute of Technology Press, Cambridge.

Williams, M.B. (1976) The logical structure of functional explanation in biology. Proceedings of the Philosophy of Science Association 1 37-46.

Winston, P.H. (1977) Artificial Intelligence, Addison-Wesley, Reading, Massachusetts.

Wong, R. and Harth, E. (1973) Stationary states and transients in neural populations. Journal of Theoretical Biology 40 77-106.

Zeigler, B.P. (1972) The base model concept. In Mohler, R.R. and Ruberti, A. (eds.) Theory and Applications of Variable Structure Systems, Academic Press, New York.

Zeigler, B.P. (1975) Statistical simplification of neural nets. International Journal of Man-Machine Studies 7 371-393.

Zeigler, B.P. (1976) Theory of Modelling and Simulation, Wiley, New York.