PART I

THE PDP PERSPECTIVE

CHAPTER 1

The Appeal of Parallel Distributed Processing

J. L. McCLELLAND, D. E. RUMELHART, and G. E. HINTON

What makes people smarter than machines? They certainly are not quicker or more precise. Yet people are far better at perceiving objects in natural scenes and noting their relations, at understanding language and retrieving contextually appropriate information from memory, at making plans and carrying out contextually appropriate actions, and at a wide range of other natural cognitive tasks. People are also far better at learning to do these things more accurately and fluently through processing experience.

What is the basis for these differences? One answer, perhaps the classic one we might expect from artificial intelligence, is "software." If we only had the right computer program, the argument goes, we might be able to capture the fluidity and adaptability of human information processing.

Certainly this answer is partially correct. There have been great breakthroughs in our understanding of cognition as a result of the development of expressive high-level computer languages and powerful algorithms. No doubt there will be more such breakthroughs in the future. However, we do not think that software is the whole story.

In our view, people are smarter than today's computers because the brain employs a basic computational architecture that is more suited to deal with a central aspect of the natural information processing tasks that people are so good at. In this chapter, we will show through examples that these tasks generally require the simultaneous consideration of many pieces of information or constraints. Each constraint may be imperfectly specified and ambiguous, yet each can play a potentially decisive role in determining the outcome of processing. After examining these points, we will introduce a computational framework for modeling cognitive processes that seems well suited to exploiting these constaints and that seems closer than other frameworks to the style of computation as it might be done by the brain. We will review several early examples of models developed in this framework, and we will show that the mechanisms these models employ can give rise to powerful emergent properties that begin to suggest attractive alternatives to traditional accounts of various aspects of cognition. We will also show that models of this class provide a basis for understanding how learning can occur spontaneously, as a by-product of processing activity.

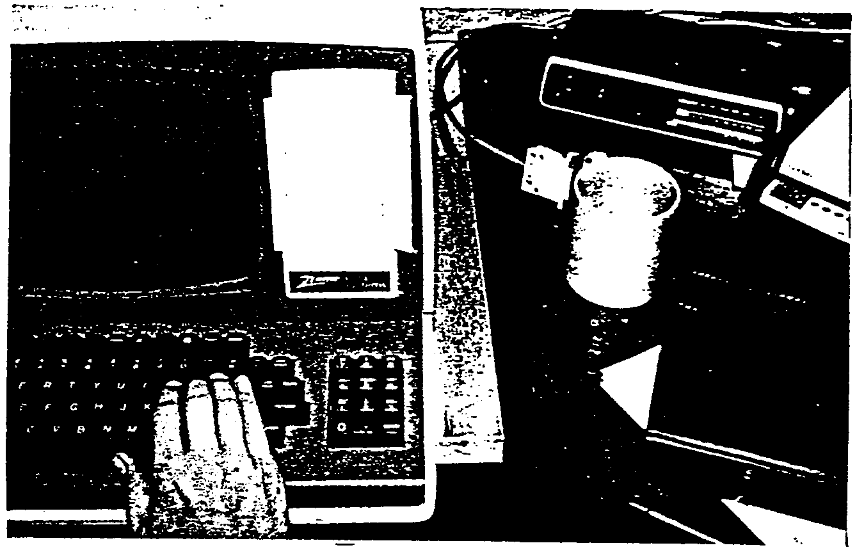
Multiple Simultaneous Constraints

Reaching and grasping. Hundreds of times each day we reach for things. We nearly never think about these acts of reaching. And yet, each time, a large number of different considerations appear to jointly determine exactly how we will reach for the object. The position of the object, our posture at the time, what else we may also be holding, the size, shape, and anticipated weight of the object, any obstacles that may be in the way—all of these factors jointly determine the exact method we will use for reaching and grasping.

Consider the situation shown in Figure 1. Figure 1A shows Jay McClelland's hand, in typing position at his terminal. Figure 1B indicates the position his hand assumed in reaching for a small knob on the desk beside the terminal. We will let him describe what happened in the first person:

> On the desk next to my terminal are several objects—a chipped coffee mug, the end of a computer cable, a knob from a clock radio. I decide to pick the knob up. At first I hesitate, because it doesn't seem possible. Then I just reach for it, and find myself grasping the knob in what would normally be considered a very awkward position—but it solves all of the constraints. I'm not sure what all the details of the movement were, so I let myself try it a few times more. I observe that my right hand is carried up off the keyboard, bent at the elbow, until my forearm is at about a 30° angle to the desk top and parallel to the side of the terminal. The palm is facing downward through most of this. Then, my arm extends and lowers down more or less parallel to the edge of the desk and parallel to the side of the terminal and, as it drops, it turns about 90° so that the

A



B

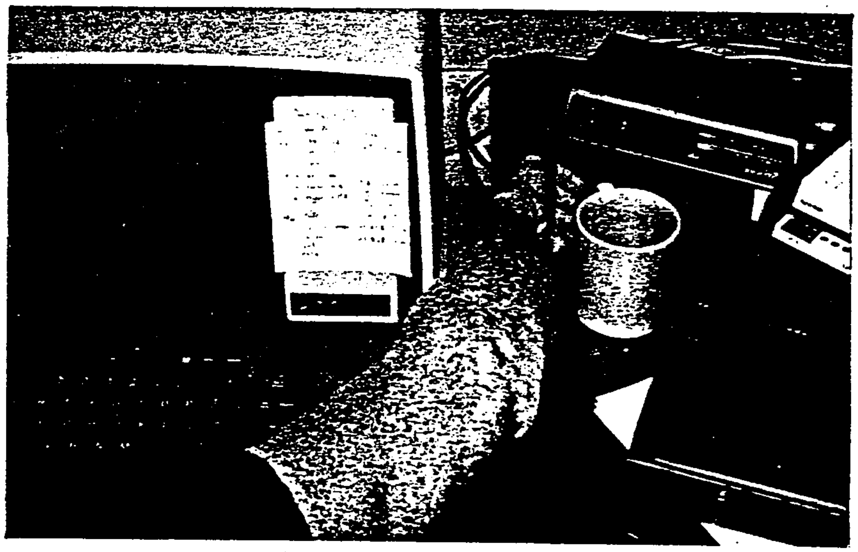


FIGURE 1. *A:* An everyday situation in which it is necessary to take into account a large number of constraints to grasp a desired object. In this case the target object is the small knob to the left of the cup. *B:* The posture the arm arrives at in meeting these constraints.

> palm is facing the cup and the thumb and index finger are below. The turning motion occurs just in time, as my hand drops, to avoid hitting the coffee cup. My index finger and thumb close in on the knob and grasp it, with my hand completely upside down.

Though the details of what happened here might be quibbled with, the broad outlines are apparent. The shape of the knob and its position on the table; the starting position of the hand on the keyboard; the positions of the terminal, the cup, and the knob; and the constraints imposed by the structure of the arm and the musculature used to control it—all these things conspired to lead to a solution which exactly suits the problem. If any of these constraints had not been included, the movement would have failed. The hand would have hit the cup or the terminal—or it would have missed the knob.

The mutual influence of syntax and semantics. Multiple constraints operate just as strongly in language processing as they do in reaching and grasping. Rumelhart (1977) has documented many of these multiple constraints. Rather than catalog them here, we will use a few examples from language to illustrate the fact that the constraints tend to be reciprocal: The example shows that they do not run only from syntax to semantics—they also run the other way.

It is clear, of course, that syntax constrains the assignment of meaning. Without the syntactic rules of English to guide us, we cannot correctly understand who has done what to whom in the following sentence:

> The boy the man chased kissed the girl.

But consider these examples (Rumelhart, 1977; Schank, 1973):

> I saw the grand canyon flying to New York.
> I saw the sheep grazing in the field.

Our knowledge of syntactic rules alone does not tell us what grammatical role is played by the prepositional phrases in these two cases. In the first, "flying to New York" is taken as describing the context in which the speaker saw the Grand Canyon—while he was flying to New York. In the second, "grazing in the field" could syntactically describe an analogous situation, in which the speaker is grazing in the field, but this possibility does not typically become available on first reading. Instead we assign "grazing in the field" as a modifier of the sheep (roughly, "who were grazing in the field"). The syntactic structure of each of these sentences, then, is determined in part by the semantic relations that the constituents of the sentence might plausibly bear to one another. Thus, the influences appear to run both ways, from the syntax to the semantics and from the semantics to the syntax.

In these examples, we see how syntactic considerations influence semantic ones and how semantic ones influence syntactic ones. We cannot say that one kind of constraint is primary.

Mutual constraints operate, not only between syntactic and semantic processing, but also within each of these domains as well. Here we consider an example from syntactic processing, namely, the assignment of words to syntactic categories. Consider the sentences:

I like the joke.
I like the drive.
I like to joke.
I like to drive.

In this case it looks as though the words *the* and *to* serve to determine whether the following word will be read as a noun or a verb. This, of course, is a very strong constraint in English and can serve to force a verb interpretation of a word that is not ordinarily used this way:

I like to mud.

On the other hand, if the information specifying whether the function word preceding the final word is *to* or *the* is ambiguous, then the typical reading of the word that follows it will determine which way the function word is heard. This was shown in an experiment by Isenberg, Walker, Ryder, and Schweikert (1980). They presented sounds halfway between *to* (actually /tˆ/) and *the* (actually /dˆ/) and found that words like *joke*, which we tend to think of first as nouns, made subjects hear the marginal stimuli as *the,* while words like *drive*, which we tend to think of first as verbs, made subjects hear the marginal stimuli as *to.* Generally, then, it would appear that each word can help constrain the syntactic role, and even the identity, of every other word.

Simultaneous mutual constraints in word recognition. Just as the syntactic role of one word can influence the role assigned to another in analyzing sentences, so the identity of one letter can influence the identity assigned to another in reading. A famous example of this, from Selfridge, is shown in Figure 2. Along with this is a second example in which none of the letters, considered separately, can be identified unambiguously, but in which the possibilities that the visual

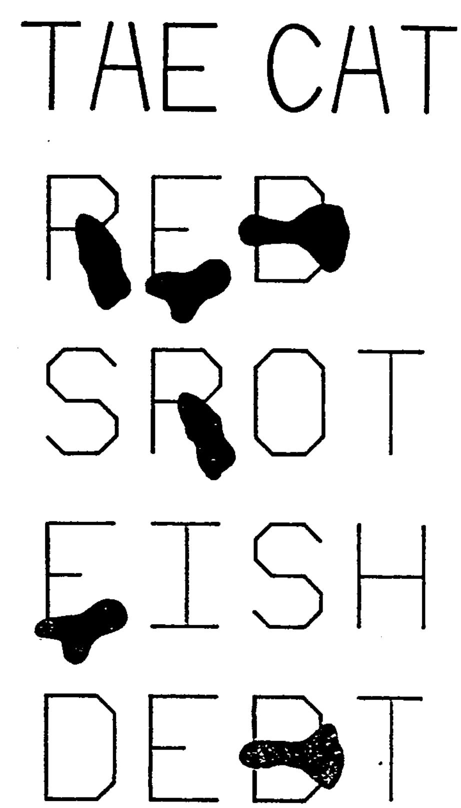


FIGURE 2. Some ambiguous displays. The first one is from Selfridge, 1955. The second line shows that three ambiguous characters can each constrain the identity of the others. The third, fourth, and fifth lines show that these characters are indeed ambiguous in that they assume other identities in other contexts. (The ink-blot technique of making letters ambiguous is due to Lindsay and Norman, 1972).

information leaves open for each so constrain the possible identities of the others that we are capable of identifying all of them.

At first glance, the situation here must seem paradoxical: The identity of each letter is constrained by the identities of each of the others. But since in general we cannot know the identities of any of the letters until we have established the identities of the others, how can we get the process started?

The resolution of the paradox, of course, is simple. One of the different possible letters in each position fits together with the others. It appears then that our perceptual system is capable of exploring all these possibilities without committing itself to one until all of the constraints are taken into account.

Understanding through the interplay of multiple sources of knowledge. It is clear that we know a good deal about a large number of different standard situations. Several theorists have suggested that we store this knowledge in terms of structures called variously: *scripts* (Schank, 1976), *frames* (Minsky, 1975), or *schemata* (Norman & Bobrow, 1976; Rumelhart, 1975). Such knowledge structures are assumed to be the basis of comprehension. A great deal of progress has been made within the context of this view.

However, it is important to bear in mind that most everyday situations cannot be rigidly assigned to just a single script. They generally involve an interplay between a number of different sources of information. Consider, for example, a child's birthday party at a restaurant. We know things about birthday parties, and we know things about restaurants, but we would not want to assume that we have explicit knowledge (at least, not in advance of our first restaurant birthday party) about the conjunction of the two. Yet we can imagine what such a party might be like. The fact that the party was being held in a restaurant would modify certain aspects of our expectations for birthday parties (we would not expect a game of Pin-the-Tail-on-the-Donkey, for example), while the fact that the event was a birthday party would inform our expectations for what would be ordered and who would pay the bill.

Representations like scripts, frames, and schemata are useful structures for encoding knowledge, although we believe they only approximate the underlying structure of knowledge representation that emerges from the class of models we consider in this book, as explained in Chapter 14. Our main point here is that any theory that tries to account for human knowledge using script-like knowledge structures will have to allow them to interact with each other to capture the generative capacity of human understanding in novel situations. Achieving such interactions has been one of the greatest difficulties associated with implementing models that really think generatively using script- or frame-like representations.

PARALLEL DISTRIBUTED PROCESSING

In the examples we have considered, a number of different pieces of information must be kept in mind at once. Each plays a part, constraining others and being constrained by them. What kinds of mechanisms seem well suited to these task demands? Intuitively, these tasks seem to require mechanisms in which each aspect of the information in the situation can act on other aspects, simultaneously influencing other aspects and being influenced by them. To articulate these intuitions, we and others have turned to a class of models we call *Parallel Distributed Processing* (PDP) models. These models assume that information processing takes place through the interactions of a large number of simple processing elements called units, each sending excitatory and inhibitory signals to other units. In some cases, the units stand for possible hypotheses about such things as the letters in a particular display or the syntactic roles of the words in a particular sentence. In these cases, the activations stand roughly for the strengths associated with the different possible hypotheses, and the interconnections among the units stand for the constraints the system knows to exist between the hypotheses. In other cases, the units stand for possible goals and actions, such as the goal of typing a particular letter, or the action of moving the left index finger, and the connections relate goals to subgoals, subgoals to actions, and actions to muscle movements. In still other cases, units stand not for particular hypotheses or goals, but for aspects of these things. Thus a hypothesis about the identity of a word, for example, is itself distributed in the activations of a large number of units.

PDP Models: Cognitive Science or Neuroscience?

One reason for the appeal of PDP models is their obvious "physiological" flavor: They seem so much more closely tied to the physiology of the brain than are other kinds of information-processing models. The brain consists of a large number of highly interconnected elements (Figure 3) which apparently send very simple excitatory and inhibitory messages to each other and update their excitations on the basis of these simple messages. The properties of the units in many of the PDP models we will be exploring were inspired by basic properties of the neural hardware. In a later section of this book, we will examine in some detail the relation between PDP models and the brain.

FIGURE 3. The arborizations of about 1 percent of the neurons near a vertical slice through the cerebral cortex. The full height of the figure corresponds to the thickness of the cortex, which is in this instance about 2 mm. (From *Mechanics of the Mind*, p. 84, by C. Blakemore, 1977, Cambridge, England: Cambridge University Press. Copyright 1977 by Cambridge University Press. Reprinted by permission.)

Though the appeal of PDP models is definitely enhanced by their physiological plausibility and neural inspiration, these are not the primary bases for their appeal to us. We are, after all, cognitive scientists, and PDP models appeal to us for psychological and computational reasons. They hold out the hope of offering computationally sufficient and psychologically accurate mechanistic accounts of the phenomena of human cognition which have eluded successful explication in conventional computational formalisms; and they have radically altered the way we think about the time-course of processing, the nature of representation, and the mechanisms of learning.

The Microstructure of Cognition

The process of human cognition, examined on a time scale of seconds and minutes, has a distinctly sequential character to it. Ideas come, seem promising, and then are rejected; leads in the solution to a problem are taken up, then abandoned and replaced with new ideas. Though the process may not be discrete, it has a decidedly sequential character, with transitions from state-to-state occurring, say, two or three times a second. Clearly, any useful description of the overall organization of this sequential flow of thought will necessarily describe a sequence of states.

But what is the internal structure of each of the states in the sequence, and how do they come about? Serious attempts to model even the simplest macrosteps of cognition—say, recognition of single words—require vast numbers of microsteps if they are implemented sequentially. As Feldman and Ballard (1982) have pointed out, the biological hardware is just too sluggish for sequential models of the microstructure to provide a plausible account, at least of the microstructure of *human* thought. And the time limitation only gets worse, not better, when sequential mechanisms try to take large numbers of constraints into account. Each additional constraint requires more time in a sequential machine, and, if the constraints are imprecise, the constraints can lead to a computational explosion. Yet people get faster, not slower, when they are able to exploit additional constraints.

Parallel distributed processing models offer alternatives to serial models of the microstructure of cognition. They do not deny that there is a macrostructure, just as the study of subatomic particles does not deny the existence of interactions between atoms. What PDP models do is describe the internal structure of the larger units, just as subatomic physics describes the internal structure of the atoms that form the constituents of larger units of chemical structure.

We shall show as we proceed through this book that the analysis of the microstructure of cognition has important implications for most of the central issues in cognitive science. In general, from the PDP point of view, the objects referred to in macrostructural models of cognitive processing are seen as approximate descriptions of emergent properties of the microstructure. Sometimes these approximate descriptions may be sufficiently accurate to capture a process or mechanism well enough; but many times, we will argue, they fail to provide sufficiently elegant or tractable accounts that capture the very flexibility and open-endedness of cognition that their inventors had originally intended to capture. We hope that our analysis of PDP models will show how an examination of the microstructure of cognition can lead us closer to an adequate description of the real extent of human processing and learning capacities.

The development of PDP models is still in its infancy. Thus far the models which have been proposed capture simplified versions of the kinds of phenomena we have been describing rather than the full elaboration that these phenomena display in real settings. But we think there have been enough steps forward in recent years to warrant a concerted effort at describing where the approach has gotten and where it is going now, and to point out some directions for the future.

The first section of the book represents an introductory course in parallel distributed processing. The rest of this chapter attempts to describe in informal terms a number of the models which have been proposed in previous work and to show that the approach is indeed a fruitful one. It also contains a brief description of the major sources of the inspiration we have obtained from the work of other researchers. This chapter is followed, in Chapter 2, by a description of the quantitative framework within which these models can be described and examined. Chapter 3 explicates one of the central concepts of the book: *distributed representation*. The final chapter in this section, Chapter 4, returns to the question of demonstrating the appeal of parallel distributed processing models and gives an overview of our explorations in the microstructure of cognition as they are laid out in the remainder of this book.

EXAMPLES OF PDP MODELS

In what follows, we review a number of recent applications of PDP models to problems in motor control, perception, memory, and language. In many cases, as we shall see, parallel distributed processing mechanisms are used to provide natural accounts of the exploitation of multiple, simultaneous, and often mutual constraints. We will also see that these same mechanisms exhibit emergent properties which lead to novel interpretations of phenomena which have traditionally been interpreted in other ways.

Motor Control

Having started with an example of how multiple constraints appear to operate in motor programming, it seems appropriate to mention two models in this domain. These models have not developed far enough to capture the full details of obstacle avoidance and multiple constraints on reaching and grasping, but there have been applications to two problems with some of these characteristics.

Finger movements in skilled typing. One might imagine, at first glance, that typists carry out keystrokes successively, first programming one stroke and then, when it is completed, programming the next. However, this is not the case. For skilled typists, the fingers are continually anticipating upcoming keystrokes. Consider the word *vacuum.* In this word, the *v, a,* and *c* are all typed with the left hand, leaving the right hand nothing to do until it is time to type the first *u*. However, a high speed film of a good typist shows that the right hand moves up to anticipate the typing of the *u*, even as the left hand is just beginning to type the *v*. By the time the *c* is typed the right index finger is in position over the *u* and ready to strike it.

When two successive key strokes are to be typed with the fingers of the same hand, concurrent preparation to type both can result in similar or conflicting instructions to the fingers and/or the hand. Consider, in this light, the difference between the sequence *ev* and the sequence *er.* The first sequence requires the typist to move up from home row to type the *e* and to move down from the home row to type the *v,* while in the second sequence, both the *e* and the *r* are above the home row.

The hands take very different positions in these two cases. In the first case, the hand as a whole stays fairly stationary over the home row. The middle finger moves up to type the *e*, and the index finger moves down to type the *v*. In the second case, the hand as a whole moves up, bringing the middle finger over the *e* and the index finger over the *r*. Thus, we can see that several letters can simultaneously influence the positioning of the fingers and the hands.

From the point of view of optimizing the efficiency of the typing motion, these different patterns seem very sensible. In the first case, the hand as a whole is maintained in a good compromise position to allow the typist to strike both letters reasonably efficiently by extending the fingers up or down. In the second case, the need to extend the fingers is reduced by moving the whole hand up, putting it in a near-optimal position to strike either key.

Rumelhart and Norman (1982) have simulated these effects using PDP mechanisms. Figure 4 illustrates aspects of the model as they are illustrated in typing the word *very*. In brief, Rumelhart and Norman assumed that the decision to type a word caused activation of a unit for that word. That unit, in turn, activated units corresponding to each of the letters in the word. The unit for the first letter to be typed was made to inhibit the units for the second and following letters, the unit for the second to inhibit the third and following letters, and so on. As a result of the interplay of activation and inhibition among these units, the unit for the first letter was at first the most strongly active, and the units for the other letters were partially activated.

Each letter unit exerts influences on the hand and finger involved in typing the letter. The *v* unit, for example, tends to cause the index finger to move down and to cause the whole hand to move down with it. The *e* unit, on the other hand, tends to cause the middle finger on the left hand to move up and to cause the whole hand to move up also. The *r* unit also causes the left index finger to move up and the left hand to move up with it.

The extent of the influences of each letter on the hand and finger it directs depends on the extent of the activation of the letter. Therefore, at first, in typing the word *very*, the *v* exerts the greatest control.

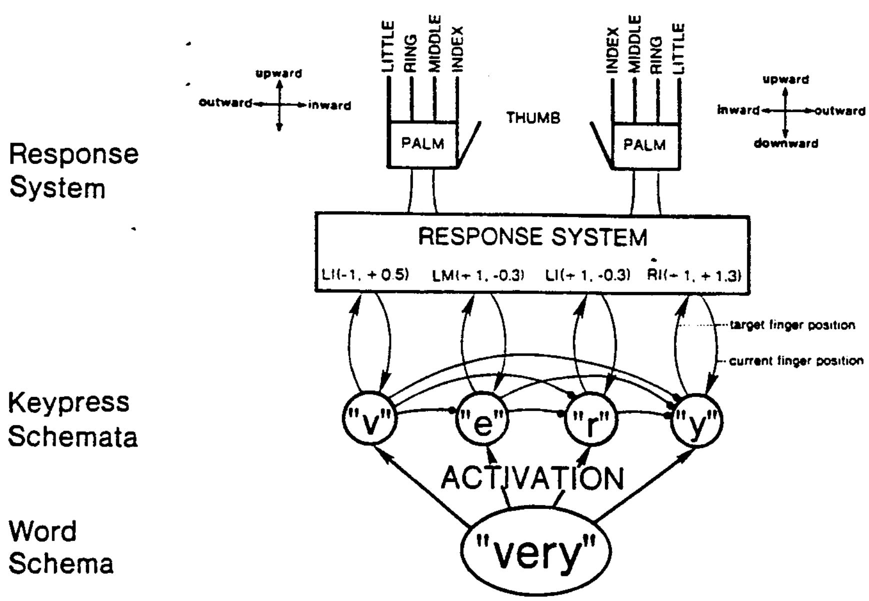


FIGURE 4. The interaction of activations in typing the word *very*. The *very* unit is activated from outside the model. It in turn activates the units for each of the component letters. Each letter unit specifies the target finger positions, specified in a keyboard coordinate system. L and R stand for the left and right hands, and I and M for the index and middle fingers. The letter units receive information about the current finger position from the response system. Each letter unit inhibits the activation of all letter units that follow it in the word: inhibitory connections are indicated by the lines with solid dots at their terminations. (From "Simulating a Skilled Typist: A Study of Skilled Motor Performance" by D. E. Rumelhart and D. A. Norman, 1982, *Cognitive Science, 6,* p. 12. Copyright 1982 by Ablex Publishing. Reprinted by permission.)

Because the *e* and *r* are simultaneously pulling the hand up, though, the *v* is typed primarily by moving the index finger, and there is little movement on the whole hand.

Once a finger is within a certain striking distance of the key to be typed, the actual pressing movement is triggered, and the keypress occurs. The keypress itself causes a strong inhibitory signal to be sent to the unit for the letter just typed, thereby removing this unit from the picture and allowing the unit for the next letter in the word to become the most strongly activated.

This mechanism provides a simple way for all of the letters to jointly determine the successive configurations the hand will enter into in the process of typing a word. This model has shown considerable success predicting the time between successive keystrokes as a function of the different keys involved. Given a little noise in the activation process, it can also account for some of the different kinds of errors that have been observed in transcription typing.

The typing model represents an illustration of the fact that serial behavior—a succession of key strokes—is not necessarily the result of an inherently serial processing mechanism. In this model, the sequential structure of typing emerges from the interaction of the excitatory and inhibitory influences among the processing units.

Reaching for an object without falling over. Similar mechanisms can be used to model the process of reaching for an object without losing one's balance while standing, as Hinton (1984) has shown. He considered a simple version of this task using a two-dimensional "person" with a foot, a lower leg, an upper leg, a trunk, an upper arm, and a lower arm. Each of these limbs is joined to the next at a joint which has a single degree of rotational freedom. The task posed to this person is to reach a target placed somewhere in front of it, without taking any steps and without falling down. This is a simplified version of the situation in which a real person has to reach out in front for an object placed somewhere in the plane that vertically bisects the body. The task is not as simple as it looks, since if we just swing an arm out in front of ourselves, it may shift our center of gravity so far forward that we will lose our balance. The problem, then, is to find a set of joint angles that simultaneously solves the two constraints on the task. First, the tip of the forearm must touch the object. Second, to keep from falling down, the person must keep its center of gravity over the foot.

To do this, Hinton assigned a single processor to each joint. On each computational cycle, each processor received information about how far the tip of the hand was from the target and where the center of gravity was with respect to the foot. Using these two pieces of information, each joint adjusted its angle so as to approach the goals of maintaining balance and bringing the tip closer to the target. After a number of iterations, the stick-person settled on postures that satisfied the goal of reaching the target and the goal of maintaining the center of gravity over the "feet."

Though the simulation was able to perform the task, eventually satisfying both goals at once, it had a number of inadequacies stemming from the fact that each joint processor attempted to achieve a solution in ignorance of what the other joints were attempting to do. This problem was overcome by using additional processors responsible for setting combinations of joint angles. Thus, a processor for flexion and extension of the leg would adjust the knee, hip, and ankle joints synergistically, while a processor for flexion and extension of the arm would adjust the shoulder and elbow together. With the addition of processors of this form, the number of iterations required to reach a solution was greatly reduced, and the form of the approach to the solution looked very natural. The sequence of configurations attained in one processing run is shown in Figure 5.

Explicit attempts to program a robot to cope with the problem of maintaining balance as it reaches for a desired target have revealed the difficulty of deriving explicitly the right combinations of actions for each possible starting state and goal state. This simple model illustrates that we may be wrong to seek such an explicit solution. We see here that a solution to the problem can emerge from the action of a number of simple processors each attempting to honor the constraints independently.

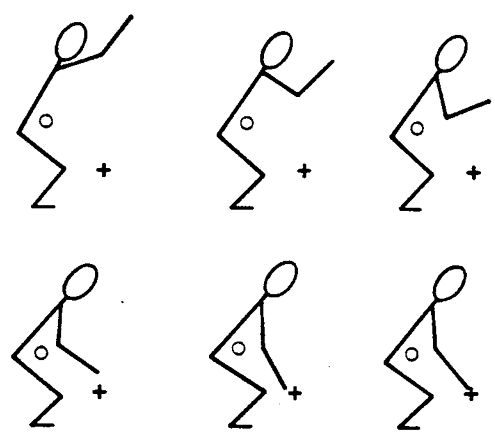


FIGURE 5. A sequence of configurations assumed by the stick "person" performing the reaching task described in the text, from Hinton (1984). The small circle represents the center of gravity of the whole stick-figure, and the cross represents the goal to be reached. The configuration is shown on every second iteration.

Perception

Stereoscopic vision. One early model using parallel distributed processing was the model of stereoscopic depth perception proposed by Marr and Poggio (1976). Their theory proposed to explain the perception of depth in random-dot stereograms (Figure 6) in terms of a simple distributed processing mechanism.

Random-dot stereograms present interesting challenges to mechanisms of depth perception. A stereogram consists of two random-dot patterns. In a simple stereogram such as the one shown here, one pattern is an exact copy of the other except that the pattern of dots in a region of one of the patterns is shifted horizontally with respect to the rest of the pattern. Each of the two patterns—corresponding to two retinal images—consists entirely of a pattern of random dots, so there is no information in either of the two views considered alone that can indicate the presence of different surfaces, let alone depth relations among those surfaces. Yet, when one of these dot patterns is projected to the left eye and the other to the right eye, an observer sees each region as a surface, with the shifted region hovering in front of or behind the other, depending on the direction of the shift.

FIGURE 6. Random-dot stereograms. The two patterns are identical except that the pattern of dots in the central region of the left pattern are shifted over with respect to those in the right. When viewed stereoscopically such that the left pattern projects to the left eye and the right pattern to the right eye, the shifted area appears to hover above the page. Some readers may be able to achieve this by converging to a distant point (e.g., a far wall) and then interposing the figure into the line of sight. (From *Vision*, p. 9, by D. Marr, 1982. San Francisco: Freeman. Copyright 1982 by W. H. Freeman & Co. Reprinted by permission.)

What kind of a mechanism might we propose to account for these facts? Marr and Poggio (1976) began by explicitly representing the two views in two arrays, as human observers might in two different retinal images. They noted that corresponding black dots at different perceived distances from the observer will be offset from each other by different amounts in the two views. The job of the model is to determine which points correspond. This task is, of course, made difficult by the fact that there will be a very large number of spurious correspondences of individual dots. The goal of the mechanism, then, is to find those correspondences that represent real correspondences in depth and suppress those that represent spurious correspondences.

To carry out this task, Marr and Poggio assigned a processing unit to each possible conjunction of a point in one image and a point in the other. Since the eyes are offset horizontally, the possible conjunctions occur at various offsets or disparities along the horizontal dimension. Thus, for each point in one eye, there was a set of processing units with one unit assigned to the conjunction of that point and the point at each horizontal offset from it in the other eye.

Each processing unit received activation whenever both of the points the unit stood for contained dots. So far, then, units for both real and spurious correspondences would be equally activated. To allow the mechanism to find the right correspondences, they pointed out two general principles about the visual world: (a) Each point in each view generally corresponds to one and only one point in the other view, and (b) neighboring points in space tend to be at nearly the same depth and therefore at about the same disparity in the two images. While there are discontinuities at the edges of things, over most of a two-dimensional view of the world there will be continuity. These principles are called the *uniqueness* and *continuity* constraints, respectively.

Marr and Poggio incorporated these principles into the interconnections between the processing units. The uniqueness constraint was captured by inhibitory connections among the units that stand for alternative correspondences of the same dot. The continuity principle was captured by excitatory connections among the units that stand for similar offsets of adjacent dots.

These additional connections allow the Marr and Poggio model to "solve" stereograms like the one shown in the figure. At first, when a pair of patterns is presented, the units for all possible correspondences of a dot in one eye with a dot in the other will be equally excited. However, the excitatory connections cause the units for the correct conjunctions to receive more excitation than units for spurious conjunctions, and the inhibitory connections allow the units for the correct conjunctions to turn off the units for the spurious connections. Thus, the model tends to settle down into a stable state in which only the correct correspondence of each dot remains active.

There are a number of reasons why Marr and Poggio (1979) modified this model (see Marr, 1982, for a discussion), but the basic mechanisms of mutual excitation between units that are mutually consistent and mutual inhibition between units that are mutually incompatible provide a natural mechanism for settling on the right conjunctions of points and rejecting spurious ones. The model also illustrates how general principles or rules such as the uniqueness and continuity principles may be embodied in the connections between processing units, and how behavior in accordance with these principles can emerge from the interactions determined by the pattern of these interconnections.

Perceptual completion of familiar patterns. Perception, of course, is influenced by familiarity. It is a well-known fact that we often misperceive unfamiliar objects as more familiar ones and that we can get by with less time or with lower-quality information in perceiving familiar items than we need for perceiving unfamiliar items. Not only does familiarity help us determine what the higher-level structures are when the lower-level information is ambiguous; it also allows us to fill in missing lower-level information within familiar higher-order patterns. The well-known *phonemic restoration effect* is a case in point. In this phenomenon, perceivers hear sounds that have been cut out of words as if they had actually been present. For example, Warren (1970) presented *legi#lature* to subjects, with a click in the location marked by the #. Not only did subjects correctly identify the word legislature; they also heard the missing /s/ just as though it had been presented. They had great difficulty localizing the click, which they tended to hear as a disembodied sound. Similar phenomena have been observed in visual perception of words since the work of Pillsbury (1897).

Two of us have proposed a model describing the role of familiarity in perception based on excitatory and inhibitory interactions among units standing for various hypotheses about the input at different levels of abstraction (McClelland & Rumelhart, 1981; Rumelhart & McClelland, 1982). The model has been applied in detail to the role of familiarity in the perception of letters in visually presented words, and has proved to provide a very close account of the results of a large number of experiments.

The model assumes that there are units that act as detectors for the visual features which distinguish letters, with one set of units assigned to detect the features in each of the different letter-positions in the word. For four-letter words, then, there are four such sets of detectors. There are also four sets of detectors for the letters themselves and a set of detectors for the words.

In the model, each unit has an activation value, corresponding roughly to the strength of the hypothesis that what that unit stands for is present in the perceptual input. The model honors the following important relations which hold between these "hypotheses" or activations: First, to the extent that two hypotheses are mutually consistent, they should support each other. Thus, units that are mutually consistent, in the way that the letter *T* in the first position is consistent with the word *TAKE*, tend to excite each other. Second, to the extent that two hypotheses are mutually inconsistent, they should weaken each other. Actually, we can distinguish two kinds of inconsistency: The first kind might be called between-level inconsistency. For example, the hypothesis that a word begins with a *T* is inconsistent with the hypothesis that the word is *MOVE*. The second might be called mutual exclusion. For example, the hypothesis that a word begins with *T* excludes the hypothesis that it begins with *R* since a word can only begin with one letter. Both kinds of inconsistencies operate in the word perception model to reduce the activations of units. Thus, the letter units in each position compete with all other letter units in the same position, and the word units compete with each other. This type of inhibitory interaction is often called *competitive inhibition*. In addition, there are inhibitory interactions between incompatible units on different levels. This type of inhibitory interaction is simply called *between-level inhibition*.

The set of excitatory and inhibitory interactions between units can be diagrammed by drawing excitatory and inhibitory links between them. The whole picture is too complex to draw, so we illustrate only with a fragment: Some of the interactions between some of the units in this model are illustrated in Figure 7.

Let us consider what happens in a system like this when a familiar stimulus is presented under degraded conditions. For example, consider the display shown in Figure 8. This display consists of the letters *W, O,* and *R*, completely visible, and enough of a fourth letter to rule out all letters other than *R* and *K*. Before onset of the display, the activations of the units are set at or below 0. When the display is presented, detectors for the features present in each position become active (i.e., their activations grow above 0). At this point, they begin to excite and inhibit the corresponding detectors for letters. In the first three positions, *W, O,* and *R* are unambiguously activated, so we will focus our attention on the fourth position where *R* and *K* are both equally consistent with the active features. Here, the activations of the detectors for *R* and *K* start out growing together, as the feature detectors below them become activated. As these detectors become active, they and the active letter detectors for *W, O,* and *R* in the other positions start to activate detectors for words which have these letters in

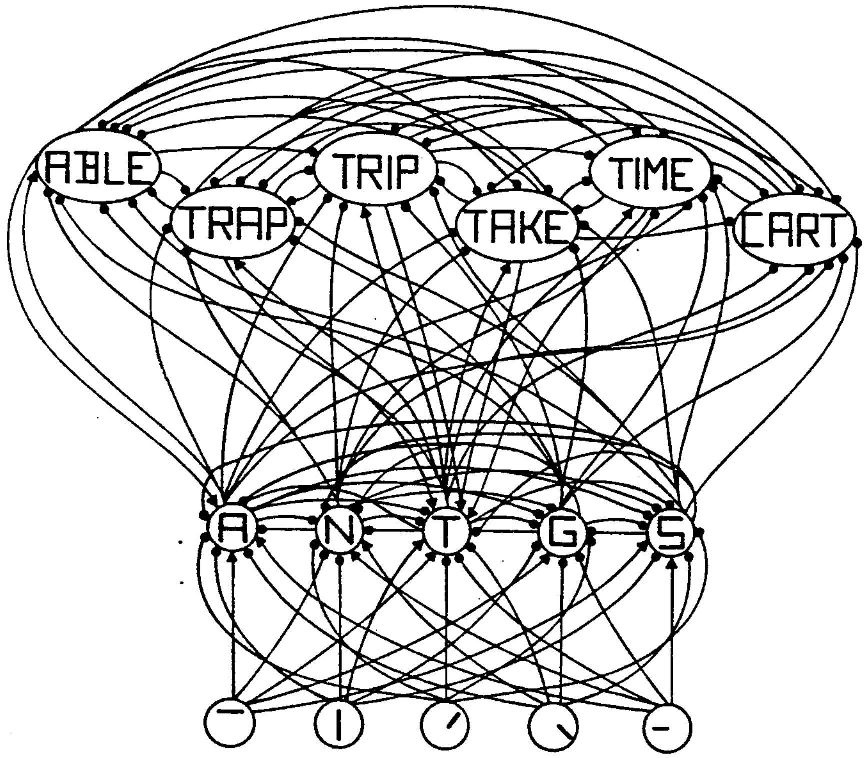


FIGURE 7. The unit for the letter *T* in the first position of a four-letter array and some of its neighbors. Note that the feature and letter units stand only for the first position; in a complete picture of the units needed from processing four-letter displays, there would be four full sets of feature detectors and four full sets of letter detectors. (From "An Interactive Activation Model of Context Effects in Letter Perception: Part 1. An Account of Basic Findings" by J. L. McClelland and D. E. Rumelhart, 1981, *Psychological Review, 88,* p. 380. Copyright 1981 by the American Psychological Association. Reprinted by permission.)

them and to inhibit detectors for words which do not have these letters. A number of words are partially consistent with the active letters, and receive some net excitation from the letter level, but only the word *WORK* matches one of the active letters in all four positions. As a result, *WORK* becomes more active than any other word and inhibits the other words, thereby successfully dominating the pattern of activation among the word units. As it grows in strength, it sends feedback to the letter level, reinforcing the activations of the *W, O, R,* and *K* in the corresponding positions. In the fourth position, this feedback gives *K* the upper hand over *R*, and eventually the stronger activation of the

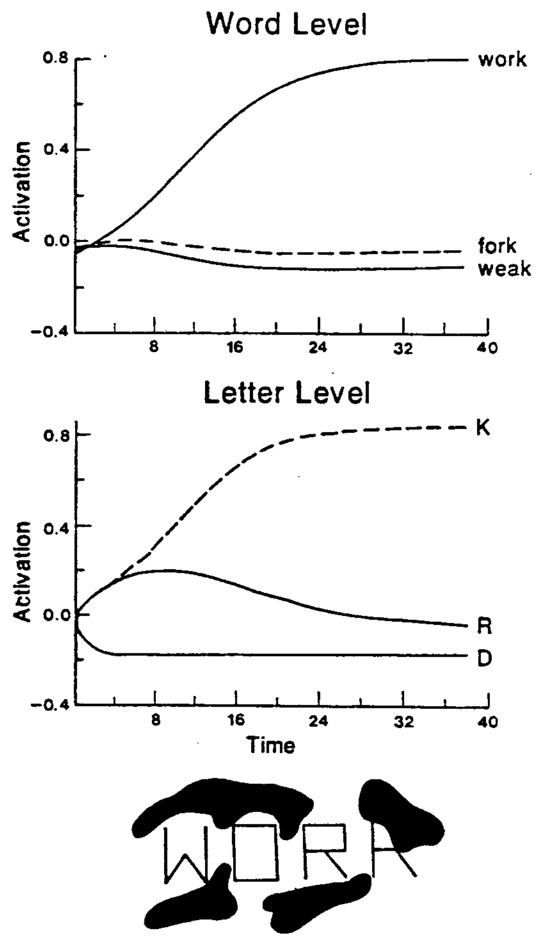


FIGURE 8. A possible display which might be presented to the interactive activation model of word recognition, and the resulting activations of selected letter and word units. The letter units are for the letters indicated in the fourth position of a four-letter display.

K detector allows it to dominate the pattern of activation, suppressing the *R* detector completely.

This example illustrates how PDP models can allow knowledge about what letters go together to form words to work together with natural constraints on the task (i.e., that there should only be one letter in one place at one time), to produce perceptual completion in a simple and direct way.

Completion of novel patterns. However, the perceptual intelligence of human perceivers far exceeds the ability to recognize familiar patterns and fill in missing portions. We also show facilitation in the perception of letters in unfamiliar letter strings which are word-like but not themselves actually familiar.

One way of accounting for such performances is to imagine that the perceiver possesses, in addition to detectors for familiar words, sets of detectors for regular subword units such as familiar letter clusters, or that they use abstract rules, specifying which classes of letters can go with which others in different contexts. It turns out, however, that the model we have already described needs no such additional structure to produce perceptual facilitation for word-like letter strings; to this extent it acts as if it "knows" the orthographic structure of English. We illustrate this feature of the model with the example shown in Figure 9, where the nonword *YEAD* is shown in degraded form so that the second letter is incompletely visible. Given the information about this letter, considered alone, either *E* or *F* would be possible in the second position. Yet our model will tend to complete this letter as an *E*.

The reason for this behavior is that, when *YEAD* is shown, a number of words are partially activated. There is no word consistent with *Y, E* or *F, A,* and *D*, but there are words which match *YEA_* (*YEAR*, for example) and others which match *_EAD* (*BEAD, DEAD, HEAD,* and *READ*, for example). These and other near misses are partially activated as a result of the pattern of activation at the letter level. While they compete with each other, none of these words gets strongly enough activated to completely suppress all the others. Instead, these units act as a group to reinforce particularly the letters *E* and *A*. There are no close partial matches which include the letter *F* in the second position, so this letter receives no feedback support. As a result, *E* comes to dominate, and eventually suppress, the *F* in the second position.

The fact that the word perception model exhibits perceptual facilitation to pronounceable nonwords as well as words illustrates once again how behavior in accordance with general principles or rules can emerge from the interactions of simple processing elements. Of course, the behavior of the word perception model does not implement exactly any of the systems of orthographic rules that have been proposed by linguists (Chomsky & Halle, 1968; Venesky, 1970) or psychologists (Spoehr & Smith, 1975). In this regard, it only approximates such rule-based descriptions of perceptual processing. However, rule systems such as Chomsky and Halle's or Venesky's appear to be only approximately honored in human performance as well (Smith & Baker, 1976). Indeed, some of the discrepancies between human performance data and rule systems occur in exactly the ways that we would predict from the word perception model (Rumelhart & McClelland, 1982). This illustrates the possibility that PDP models may provide more accurate accounts of the details of human performance than models