Continuous Symbol Systems
The Logic of Connectionism

Bruce MacLennan
Computer Science Department
University of Tennessee, Knoxville

February 28, 1992

Abstract

It has been long assumed that knowledge and thought are most naturally represented as *discrete symbol systems* (calculi). Thus a major contribution of connectionism is that it provides an alternative model of knowledge and cognition that avoids many of the limitations of the traditional approach. But what idea serves for connectionism the same unifying role that the idea of a calculus served for the traditional theories? We claim it is the idea of a *continuous symbol system*.

This paper presents a preliminary formulation of continuous symbol systems and indicates how they may aid the understanding and development of connectionist theories. It begins with a brief phenomenological analysis of the discrete and continuous; the aim of this analysis is to directly contrast the two kinds of symbols systems and identify their distinguishing characteristics. Next, based on the phenomenological analysis and on other observations of existing continuous symbol systems and connectionist models, I sketch a mathematical characterization of these systems. Finally the paper turns to some applications of the theory and to its implications for knowledge representation and the theory of computation in a connectionist context. Specific problems addressed include decomposition of connectionist spaces, representation of recursive structures, properties of connectionist categories, and decidability in continuous formal systems.

Our present knowledge of human perception leaves no doubt as to the general form of any theory which is to do justice to such knowledge: a theory of perception must be a *field theory*. By this we mean that neural functions and processes with which the perceptual facts are associated in each case are located in a continuous medium; and that the events in one part of this medium influence the events in other regions in a way that depends directly of the properties of both in their relation to each other.

— W. Köhler, *Dynamics in Psychology* (p. 55)

Nothing is more practical than a good theory.

— Kurt Lewin (Marrow, 1969)

The disadvantage of regarding things in separate parts is that when one begins to cut up and analyze, each one tries to be exhaustive. The disadvantage of trying to be exhaustive is that it is consciously (mechanically) exhaustive... Only one who can imagine the formless in the formed can arrive at the truth.

— Chuang-tzu (Soshi)

Symbolic representation of qualitative entities is doomed to its rightful place of minor importance in a world where flowers and beautiful women abound.

— Einstein, "Hyperbolic Aesthetic" (1937)

1 Need for a Theory of Continuous Symbol Systems

1.1 Human Symbolic Cognition

It is now widely recognized that human symbolic cognition, by which we mean the use of language, logic and explicit reasoning, is much more flexible than symbolic AI systems. If an expert system has too few rules, then it exhibits *brittle* behavior, failing in catastrophic ways when faced with novel situations or minor exceptions to the rules. On the other hand, trying to anticipate all the situations and exceptions that may occur leads to a proliferation of rules and an exponential explosion in the machine resources required. Human cognition does not have these limitations. When people reason explicitly, their use of categories is sensitive to the context of the problem, and their inferential processes are generated from and constrained by relevance to the situation. This seems to be a natural result of the implementation of these processes, rather than a result of special context rules or relevance rules. Human language use has similar characteristics: it is flexible, context-sensitive and controlled by relevance. Further, flexibility and "softness" is the natural state of both reason and language; "hard" logic and precise language use are skills that are not easily acquired, and they are special tools used by the expert when they are called for, but not otherwise. How can we achieve similar flexibility in connectionist symbol processing?

It is not sufficient to replace categories with hard boundaries by categories with fuzzy boundaries (as is done in fuzzy set theory). Although fuzzy categories do eliminate some sources of brittleness, they do not address the complex processes by which categories may be sensitive to the global context. To achieve the flexibility of human cognition and true context-sensitive symbol use, a more radical reinterpretation of symbolic processing is necessary. We must see context-sensitive holistic processes as the normal mode of operation, and see context-free discrete symbol manipulation as a specialized modification of this norm.[1]

These observations have practical implications, for they mean human-like flexibility and competence is unlikely to result from a simple hybrid of neural network technology and expert systems technology. Wherever it appears in a hybrid system, discrete, formal, context-free symbol manipulation will be a source of brittleness and other forms of unskillful behavior (see also Sun, Chapter **??** in this volume). If we want implementations of symbolic cognition that exhibit the flexibility and competence of people's, then they must be built on a foundation that is fundamentally continuous, holistic and context-sensitive.

[1]We use the term 'holistic' in spite of some of its unfortunate connotations; it is the only term that correctly denotes systems whose structures are misrepresented by being analyzed into independent parts.

1.2 Emergence of the Discrete

Human symbolic cognition is built upon a *subsymbolic substrate*, which is continuous in its principles of operation, and is the ultimate source of the flexibility of human symbol use. To see this, observe that the basic neural processes are best described as continuous, since information is represented by continuous quantities such as spiking frequency and membrane potential. Most neurons seem to operate as low precision analog devices. Furthermore, the large number of neurons found in the brains of the higher animals implies that functional areas can often be viewed as *spatially* continuous. That is, we can view such an area as a continuum of neurons, rather than as a large number of discrete neurons. This view is even more appropriate if, as some suggest, the basic computational unit is not the neuron, but the synapse (Shepherd, 1978). Thus the basic neural processes are both temporally and spatially continuous. This is important from a theoretical perspective, because it means that powerful mathematical tools may be brought to bear on the problem of mental representation.

In addition, many of the most basic cognitive processes, such as perception, association, sensory-motor coordination, and judgement of similarity, are by their nature continuous. We share these faculties with the lower animals, and we observe that they show the same flexibility in their use as we do. Discreteness is most apparent when we come to higher cognitive processes, such as language use and explicit reason, but these faculties partake of the flexibility of the underlying continuous processes.[2]

These considerations suggest that a critical research goal is to understand the processes by which discrete and approximately discrete symbols can emerge from continuous processes. Of course this is an inversion of the usual situation in computer science and even logic, where discrete symbols are taken as given, and continuous quantities are approximated by discrete structures. We must understand how the discrete structures found in logic and language can emerge from continuous representations and processes.

[2]Nalimov (1981, Ch. 8) uses continuous *semantic fields* as a basis for discrete language and thought. Lakoff (1988) discusses the grounding of linguistic structures in continuous sensory-motor processes, and the relation between cognitive linguistics and connectionism.

1.3 Goals

Our goal is to develop a theoretical framework for connectionist knowledge representation that fills a role analogous to the theory of formal systems in symbolic knowledge representation. The properties we expect of this theory include formality, idealization, qualitative inference, and elucidation of the emergence of the discrete from the continuous. I discuss each in turn.

First observe that 'formal' is used in two distinct but related ways. In the first case, *form* is contrasted with *meaning*, or equivalently, syntax with semantics. In this sense a system is formal if its inferential processes depend only on the "shapes" of symbols, not on their meanings. Therefore, we distinguish *formal systems*, which are purely syntactic, from *symbol systems*, in which the symbols have meanings; a symbol system is a formal system together with an *interpretation* or a *semantics*, which assigns the meanings.

In the second case *form* is contrasted with *matter* (as in Plato), and *formal* means that the system's information processing capacities depend only on abstract relationships rather than on their material embodiment. It is in this sense that computer programs are formal, since they define the same computations whether they are implemented electronically or in some other way. It is this kind of formality that allows calculi (discrete formal systems) to be implemented in any medium that is sufficiently close to the discrete ideal. Analogously we expect a theory of *continuous formal systems* to be independent of material embodiment and to depend only on idealized properties of continuous media.

Both notions of formality are important for connectionist theories of knowledge representation. First, by their being purely syntactic we are confident that our definitions of cognitive processes do not appeal to homunculi.[3] Second, by their being abstract we know that our theories have captured the essential characteristics of cognition, as opposed to the accidents of its biological or electronic implementation. The syntactic kind of formality ensures that the theory is self-contained; the abstract kind ensures that it contains no more than is necessary.

[3]This approach in effect reduces semantics to syntax, which seems necessary to understanding the mechanisms of cognition. This contentious issue cannot be addressed further here.

It will be worthwhile to say a few words about "idealization." The familiar theory of discrete formal systems makes a number of *idealizing assumptions* (detailed below, Section 2.2), which are only approximately realized in physical implementations. An example of an idealizing assumption is that there are only two atomic symbols, '0' and '1', and that there is nothing "between" them. In reality these symbols might be represented by two voltage levels between which there is a continuum of levels. However, many implementations are a good approximation to the ideal, and this is the reason that discrete formal systems provide a useful theoretical framework for understanding digital computers. We expect an analogous situation in the theory of continuous formal systems. We will make certain idealizing assumptions, such as that all functions are continuous, and the success of our theory will depend on the extent to which these assumptions approximate well the physical embodiments of connectionist knowledge representation. For example, we will assume that there is always an analog value between two given analog values, even though in some implementations analog values might be represented by electric charge, which is quantized.

Connectionist models are typically continuous nonlinear dynamical systems with very large numbers of variables. Analytic prediction of the behavior of these systems is usually impossible, so it is often necessary to resort to simulation. However, such systems may be very sensitive to inaccuracies of the simulation. These considerations suggest that we need qualitative tools for understanding connectionist models, since qualitative prediction may be possible even when detailed quantitative analysis is not. For example, the existence of a Lyapunov function allows us to predict that the system will approach an asymptotically stable equilibrium, even though we may not be able to describe its exact trajectory. More generally, we expect the theory of continuous formal systems to be a *topological* theory rather than a *numerical* theory.

As argued above, a central problem of connectionist knowledge representation is the emergence of symbolic cognitive processes from subsymbolic processes. Therefore, a principal goal of the theory of continuous formal systems is that it should elucidate the relation between continuous and discrete knowledge representation, and in particular should show how continuous connectionist systems can approximate idealized discrete cognition. Some preliminary results will be found in

Section 5.2.

2 Phenomenological Analysis

2.1 Introduction

All mathematical theories are *idealizing*; that is, they select out certain properties in their domain and ignore the remainder so that they permit rigorous reasoning about the phenomena that depend on the selected properties. A mathematical theory is useful to the extent that it selects properties that are relevant to central phenomena of the domain, and to the extent that it ignores those that are peripheral. Unfortunately, when we are investigating a new domain of phenomena, it may not be obvious which phenomena are central and which peripheral; as a result idealization is problematic. In these cases a *phenomenological analysis* may help to identify the central phenomena and relevant properties.

This is the basic procedure we will follow: First we identify a domain of interest, such as connectionist systems or continuous symbol systems. The domain cannot be defined, because the identification of characteristic properties is the very problem to be solved; instead the domain must be *indicated*, largely through examples. Once the domain has been grasped we look for *invariances*, properties that hold always or for the most part. These invariances are the elements around which a mathematical theory can be constructed.

Since this kind of phenomenological analysis will be unfamiliar to many readers, we illustrate it first for the familiar *discrete* symbol systems (calculi), before applying it to the less familiar *continuous* symbol systems (image systems). This twofold analysis will also bring into the foreground the similarities and differences between the two kinds of systems.

2.2 Discrete Symbol Systems

2.2.1 Indication of the Domain

As discussed above, our phenomenological analysis begins by indicating the domain of phenomena to be analyzed. In this case we are aided by the fact that (discrete) formal systems, digital systems and calculi are all recognized categories. Therefore the identification of invariances can begin with an analysis of the *reasons* that people find these to be useful categories; these are the *pragmatic* invariances. First, however, we indicate the domain by asking, "What sorts of things are *seen as* calculi (discrete symbol systems)?"

The most familiar and characteristic example of a discrete symbol system is *written language*, and in this example we can see many of their invariances. First, discrete symbols have a hierarchical constituent structure: sentences composed of words, and words of letters. Second, the lowest level components, the letters, are considered *atomic* (indivisible), and these atomic components (*tokens*) belong to a finite number of *types*. Finally, sentences are *finite* assemblages of tokens obeying a *finite* number of syntactic rules.

Real written natural language is more complicated than implied here; for example real languages may not be characterized by a finite number of syntactic rules. Thus, although they are the main inspiration for discrete symbol systems, written languages may not be the best examples. Closer to the ideal are artificial languages such as the formulas of algebra and symbolic logic. Here the syntactic rules are finite and explicit, as are the rules for calculating with the symbols.

Less obvious, but equally familiar examples of discrete symbol systems are board games, such as checkers, chess, and go. Again, there is a finite set of rules defining the allowable configurations of tokens, and there are explicit rules defining the allowable "moves" (manipulations of the tokens).

The most complex and sophisticated discrete symbol systems are found in computer science, especially in artificial intelligence, where powerful knowledge representation languages permit the mechanization of some inferential processes (Fig. 1). The limitations of these systems has been a major motivation for the exploration of connectionist alternatives to conventional knowledge representation (Dreyfus, 1979; MacLennan, 1988a).

2.2.2 Pragmatic Invariances

Having indicated the domain of calculi (discrete symbol systems), we can begin the phenomenological analysis. Our goal is to find out *why* calculi are what they are. To discover this we must first ask how the phenomena *seen as* calculi are perceived. In this way we investigate

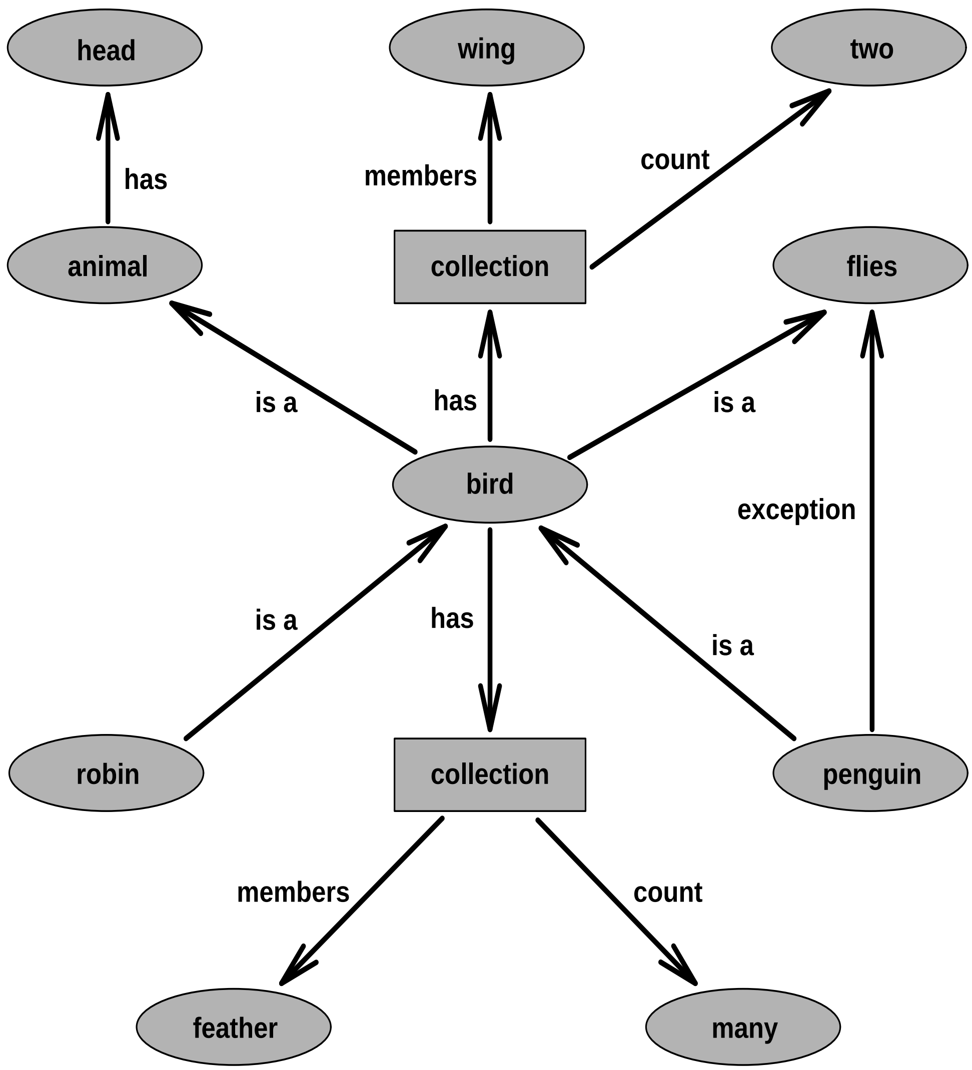

Figure 1: Example of a semantic net. Knowledge is represented by a set of atomic tokens connected by atomic relations. English-language labels make the nodes and links comprehensible to people, but are irrelevant to computer processing. In other words, `bird` and `has` could as well be `p0061` and `r0035`.

why is it important to people to recognize some phenomena as calculi. This will form a basis for identifying the other invariances of this domain.

The first characteristic of calculi is that they are *definite*. Ideally, we know exactly what we have got: what letter or symbol, what grammatical relation, what piece (white king, black rook), what board position, and so forth. Second, calculi are *reliable*. Errors, so long as they are not too large, do not affect the use of the calculus. For example, in transmitting binary information, there can be considerable variation in the signal representing a one, and it can still be decoded correctly, so long as it doesn't change so much that it looks more like a zero. Similarly, we can tolerate noise or other degradation in the form or position of printed symbols so long as it isn't so much as to make one symbol or spatial relation look like another. In general, all observers agree on the types of the tokens and the syntax of the formulas. Third, the discrete symbols are *reproducible*. Repeated reproduction does not result in cumulative error; the syntax of structures is not changed by copying.

Most of the characteristics of calculi result from their being *finitely specifiable*. That is, against an appropriate background of assumptions, we can completely describe a calculus, its syntax, semantics and rules of calculation, in a finite number of words. (Think of formal logic, board games and computer programs.)

These pragmatic invariances of calculi account for much of the success of digital computers and other digital technologies (such as digital audio). For some applications, however, other properties are more important, and it is in these applications that connectionist approaches are most promising.

2.2.3 Syntactic Invariances

Next I will outline the background assumptions we routinely make about the syntax of discrete formal systems. By making these assumptions explicit we will be better able to see the possibilities of other kinds of formal systems.

A syntactic type is one that can be determined by perception (for natural systems) or by a simple mechanism (for artificial systems). For examples, we may think of recognition of letters by a person, or recognition of bits by an electronic device. Types depend on the form

Figure 2: Discrete types are determined against a background of assumptions about what constitutes significant and insignificant variation in the tokens. For example, features of tokens that are assumed to be irrelevant to determining the type of the token might include not only size and font, but also marks considered "noise." These assumptions are usually unstated (i.e., in the background).

or "shape" of tokens, not on their meaning, but, as Fig. 2 illustrates, certain aspects of the shape are considered significant to a calculus while others (such as size or font) are not, and this set of assumptions varies from calculus to calculus. For mathematical purposes, '**A**' and '*A*' might be considered different types; for other purposes they would be insignificant variants of a single type. Further, we always assume for calculi that tokens can be correctly classified — an idealizing assumption that of course ignores the difficulties of real-world pattern classification.

We also assume that a calculus has a *finite* number of (atomic) syntactic types. This is implied by the condition that a calculus be finitely specified, since an infinite set of types could be finitely specified only by giving some general rule for their generation, in which case they are not atomic, but defined in terms of some more primitive types.

Finally, we assume that in a calculus the tokens can be unambiguously separated from the background, that is, that we always know whether or not a token is actually present (Fig. 3). This again is an idealization, since in real-world symbol processing the separation of signal from noise may be a difficult problem.

We have considered the assumed properties of the atomic tokens and types of calculi; now we turn to the relations by which they are

Figure 3: Discrete formal systems assume that it can be unambiguously determined whether or not a token is present, that is, it is assumed that a token can always be separated from the background. This *idealizing assumption* ignores the complexities of real-world signal detection. In this case, is the second number 35 or 3.5? Is there a decimal point after the 6? Is the symbol before the slash ‘1’, ‘i’ or ‘l’? Is the second operator a minus sign or an equal sign?

Figure 4: Discrete relations. Certain features of the arrangement are considered relevant to syntactic relations, other are not. For example, (a) one symbol being to the right of another may be significant, but the actual distance may not be. (b) Analogously, one symbol being a superscript of another is what is significant here.

assembled into composite symbol structures.

A *syntactic relation* is one that can also be detected by a simple perceptual or mechanical process. (Thus types are single-place perceptual predicates, whereas relations are multiplace perceptual predicates.) Once again, this classification process is assumed to be perfect, so we can always determine whether or not a relation holds (Fig. 4). Also, in a given calculus some characteristics are considered significant to the relationships (e.g. vertical displacement) while other are not (e.g., horizontal displacement; see Fig. 4).

Finite specifiability again dictates that there be a finite number of syntactic relationships (otherwise they are not primitive and can be specified in terms of more basic relations). The formation rules of most calculi can be applied recursively to build up composite formulas of arbitrary size, but the resulting formulas are required to be finite in size (number of tokens). Thus the formation rules can be applied only a finite number of times.

Most calculi are not static, that is, in addition to defining a set of symbolic structures, they define syntactic processes (*rules of calculation*) by which these structures are transformed. Finite specifiability determines many of the properties of processes. First, the number of rules is assumed to be finite. Second, the rules are assumed to be applied in discrete steps, so that in a finite amount of time only a finite number of rules may be applied. Further, the rules must be syntactic (*formal*), which means that determining the applicability of a rule depends only on the types of the tokens and their syntactic relations. Finally, application of a rule generates only a finite number of tokens and produces or changes only a finite number of relations. Thus rules are finite in their effects.

A very important property of calculi may be called *syntactic independence*: the actual syntactic types and relations used are arbitrary; they can be replaced by others with the same formal properties. This is the basis of digital computation, in which physical properties (charge, current, magnetic flux) replace other, sometimes abstract, properties (being an 'A', being immediately to the left of, etc.).

2.2.4 Semantic Invariances

Two assumptions are typically made about the semantics of discrete symbol systems. First, the operation of a calculus is *formal* or *syntactic*, that is, it is independent of any meaning that may be attached to the formulas. As far as calculation is concerned, the symbols have no meaning. Second, the semantics is *compositional*. That is, formulas are interpreted — given a meaning — recursively, by attaching meaning to the atomic types and to the syntactic rules of composition. Composite formulas are thus interpreted implicitly and there is a regular (finitely specifiable) relation between formulas and their meanings.

2.2.5 Idealization

Finally, it is important to observe that real (physical) discrete symbol systems are only approximations to this ideal (although often quite good approximations). For example, types may be confused ('O' vs. '0', '1' vs. 'l'). Manuscripts and even digital signals do get corrupted. Signal is sometimes taken for noise, and vice versa. In the theoretical characterization of discrete symbol systems we ignore this practical fuzziness and intrusion of the continuous into the idealized world of the discrete.

2.3 Continuous Symbol Systems

2.3.1 Indication of the Domain

Now that we have had some practice with phenomenological analysis in a familiar domain, we turn to *continuous* symbol systems, which have been investigated much less.

Spoken language provides the most familiar example of a continuous symbol system. Here we find significance conveyed by continuously variable and continuously varying parameters, including loudness, pitch, tone, tempo and rhythm.[4] The nonspoken components of everyday communication, such as body language, are also examples, and they remind us of gestural languages such as American Sign Language (ASL). Another informative example is the language of musical conducting: it clearly has a grammar, yet its communicative efficacy depends on continuous variation in a number of dimensions. Indeed, we can see that music and visual arts derive much of their communicative power from continuous variation (Arnheim, 1971; Goodman, 1966, 1968; Nalimov, 1981).

Of course all these continuous languages also have discrete elements, but the important point here is that they have a significant admixture of the continuous that is treated as a valuable *representational resource*, rather than as an interfering source of noise and error.

Many familiar devices are controlled through continuous symbol systems: musical instruments, of course, but also automobiles and aircraft, cameras and stereos. The steering wheel and brake both make significant use of continuous variation: how much to turn and how quickly to stop. The piano keyboard may seem to be a clear case of a discrete symbol system, but it permits continuous variation in both the time and intensity of impact.[5] We observe also that many devices, including stereos and lights, are controlled by rheostats.[6] Finally, the

[4]Ihde (1986, Ch. 1) provides an insightful analysis of the phenomenology of voice, including the contrast between the continuous and the discrete as exemplified by speech and writing.

[5]A commonplace of piano instruction is to stress *legato* playing, thus making the sound more continuous (e.g. Lhevinne, 1972, pp. 37–39). Conversely, violinists are encouraged to stress articulation. "Play the piano as though it were a violin, and the violin as though it were a piano" expresses the necessity of balancing the discrete and continuous.

[6]Apparently reflecting the view that "if it's digital it must be better," some stereos

mouse and high-resolution screens have made even our interaction with computers more continuous. Continuous symbol systems are an ubiquitous and natural aspect of our interaction with the world.

Finally, we mention briefly the most obvious examples of continuous symbol systems. Analog recording devices (both audio and video) provide examples of analog representational systems with limited computational ability. Analog computers are better examples of continuous information processing. Of course, analog computers have been out of favor for several decades, and analog audio and video equipment seem headed that way. There are good technological reasons for this move away from analog technology, and some of them have been already mentioned (Section 2.2); nevertheless, analog representational and computational systems have their own advantages, to which we now turn.

2.3.2 Pragmatic Invariances

In performing a phenomenological analysis of continuous symbol systems, we face the problem that they have not been generally recognized or studied *as a class*. (Indeed that is a principal goal of this paper.) Therefore we cannot *analyze* the class of continuous symbol systems, asking why certain things are seen as continuous symbol systems, because in fact they generally *aren't* so seen. Instead, we must take a *synthetic* approach, identifying common reasons for the use of continuous symbol systems, and thereby *creating* a category of continuous symbol systems. By way of analogy, we are not here trying to explain why some figure stands out from the background. Rather we are pointing out features of the background, with the intent that the set of features become figural. Once I've pointed out the face in the clouds, then you can see it too.

What then are the reasons for using continuous symbol systems? First, they are *flexible*. This means that, in the simplest terms, there is always another choice between too much and too little, there is always an opportunity for adjustment. Aristotle recognized the utility of such flexibility to biological systems in his doctrine of the "relative mean" (*Eth. Nic.* II. vi. 4–17): there is a "right amount," between excess and deficiency, that is appropriate for a given organism at a given time.

now have digital volume controls that have a discrete set of positions. It is often observed that the volume one wants is between the allowed positions.

In other words, optimal operating points are in principle achievable by continuous variation between excess and deficiency.

A second invariance of continuous symbol systems is that they are *robust*. Small errors (noise or malfunctions) generally lead to small effects; such systems degrade gracefully. The general absence of this property from discrete systems is in fact the root of the "software crisis" that plagues digital computer programming. In traditional engineering disciplines design is simplified by approximation, since continuity permits low-level effects to be ignored. Software engineering does not have this characteristic, since even one incorrect bit can lead to the catastrophic failure of a software system. In discrete symbol systems we typically have no bound on the effects of even the smallest changes.[7]

Third, because continuous structures can change gradually over time, they can be more easily made *adaptive*. In contrast, for a discrete symbol system to adapt, it is necessary to add or delete rules, which results in abrupt behavioral changes. A discrete symbol system cannot change its behavior gradually, but adaptability is common in continuous symbol systems.

Finally, we note that continuous symbol systems may have significant advantages in the time-critical situations often faced by animals and machines. For example, a continuous process converging to an asymptotically stable equilibrium permits the use of preliminary results, if they are needed before convergence occurs. Here we use a partial result when the "correct" answer cannot be obtained in time. Continuity also permits extrapolating to likely future states, thus allowing limited anticipation, which can improve the efficiency of future computations.

Continuous symbol systems no doubt have other pragmatic invariances, but the ones listed above will serve to contrast them with discrete symbol systems.

2.3.3 Syntactic Invariances

There is nothing in the definition of the word 'symbol' that requires symbols to be discrete; for example, we find 'symbol' defined as "something that represents something else by association, resemblance, or

[7]Of course specific discrete systems can be designed that are insensitive to fixed errors; an example is an error-correcting code.

convention; especially, a material object used to represent something invisible" (Morris, 1981, s.v. symbol). That is why we have spoken of *discrete* symbol systems versus *continuous* symbol systems.[8] Nevertheless, the word connotes discreteness and atomicity, a tendency reinforced by terms such as 'symbolic AI'. Therefore we prefer to speak of *images* rather than continuous symbols; this is consistent with the term's use in cognitive science, as well as in common usage:[9]

> A reproduction of the appearance of someone or something... A mental picture of something not real or present... A representation to the mind by speech or writing. (Morris, 1981, s.v. image)

The images of a continuous formal system are drawn from one or more *image spaces* or *continua*.

The *syntax* of continuous symbol systems is concerned with the formal properties of images, which can be described by continuous formal systems. The *semantics* of continuous symbol systems is concerned with the representative properties of images.

Semantics is considered in the next section; in this section we will identify some of the syntactic invariances of continuous symbol systems. These are the properties that we will want to capture in our theory of continuous *formal* systems. Thus we begin our investigation with the formal or syntactic properties of *uninterpreted* images. In the course of identifying invariances it will be helpful to look back at the examples we have collected of continuous symbol systems (Section 2.3.1). However, to convince ourselves that they are genuine invariances, we must use the procedure of *phenomenological variation*, that is, exploring the the range of systems (phenomena) in the indicated domain (Ihde, 1977). Unfortunately, space limitations force us to leave this process to the reader's imagination.

Our examples of continuous symbol systems show that similarity of images is a matter of degree. Further, the flexibility of continuous

[8]We have previously suggested *simulacrum* as a term for the continuous analog of the discrete *calculus*, that is, for what is here called a *continuous symbol system* (MacLennan, 1991b).

[9]A comprehensive terminology is sadly lacking. C. S. Peirce did pioneering work in this area, but his terms are not widely known, and would be confusing in this context. What we are calling a *symbol* seems to correspond to Peirce's *icon*, which has subtypes which he calls *images, diagrams* and *metaphors* (*Coll. Papers*, 2.274–304; Buchler, 1955, pp. 99–107).

symbol systems results from the fact that for any two images there is always a third image that is closer to either of the first two. For linear (one-dimensional) continua we may say that there is always an image between any two other images.

It is often convenient to assume that similarity is measured by a metric, but this is not always the case, and the choice of metric may be problematic. For example, the L_2 (Euclidean) metric is often used to measure the distance between images, but this is more through habit and mathematical convenience than principled choice. An L_p metric ($p > 2$) would accord greater significance to differences between the images, with L_∞ (maximum deviation) being the extreme case; conversely, L_1 tends to devalue differences. For another example, it is not obvious how differences in orientation or size should be combined with differences in color.

Processes in discrete formal systems are sensitive to the *types* of the tokens, and analogously we expect continuous formal systems to be sensitive to the *forms* of images, that is, to their syntactic type or category. The nature of these syntactic categories is different from those in discrete formal systems, since the pragmatic invariances of robustness and adaptability both imply that infinitesmal changes of an image not result in discontinuous changes of behavior. Therefore, category membership must vary continuously with changes in the images, which means that all syntactic categories must be in some way *fuzzy*. (See also Section 5.2.)

Next we consider the syntactic relationships that may obtain in continuous formal systems. Discrete formulas are constructed from atomic parts. In continuous systems, in contrast, the images are usually given as wholes, and their division into parts is problematic. This is illustrated in Fig. 5. What are the "parts" of the image of the frog? Since this image is represented as a bit map, the obvious answer is that the elementary parts are the pixels illustrated in the upper-right corner. This might be appropriate for some purposes, but for others a different decomposition would be more appropriate; we illustrate analyses into anatomical "parts," elementary splines, elementary polygons, and a stick-figure representation. There is no unique decomposition, as there would be for a formula in an (unambiguous) discrete formal system. Furthermore, in many cases there is no natural end to the analysis. For example, the anatomical analysis can be continued to arbitrarily smaller parts of the image.

Figure 5: Decomposition of images. Images do not have a unique decomposition into elementary "parts." The kind and degree of decomposition depends on the purpose to which it will be put. (a) The original image. Example decompositions: (b) pixels (only the eye is shown); (c) anatomical components; (d) elementary splines (of thresholded image); (e) elementary polygons (of thresholded image); (f) stick-figure analysis.

Finally we turn to computation in continuous formal systems. The adaptability and time-criticality invariances suggest that processes in continuous formal systems be assumed to progress continuously in time. Thus we take images to be transformed continuously (although we often use discrete-time approximations). See Fig. 6. Also note that there may be a number of paths by which one image may be reached from another by a process of continuous transformation.

The foregoing are the syntactic invariances that we will attempt to capture in a mathematical theory of continuous formal systems.

2.3.4 Semantic Invariances

We expect the semantics of continuous symbol systems to be an interesting topic of study, for corresponding to the *compositionality* of discrete semantics we have the *continuity* of continuous semantics; in other words, the pragmatic invariances all require a continuous function mapping images onto their interpretations (meanings). One implication of this requirement is that although continuous symbol systems may be completely formal, they are nevertheless not completely independent of their interpretations. This is familiar from the idea of analog computing: there must be some *analogy* between the symbol (image) and what it represents.

2.3.5 Idealization

As for discrete symbol systems, our characterization of continuous symbol systems is idealized. Apparently continuous processes may in fact be discrete (e.g. accumulation of charge in terms of electrons); images may be composed of discrete parts (atoms, electrons, silver grains); processes may progress in tiny finite steps. In perception there are "just noticeable differences." These all may be viewed as incursions of the discrete into the continuous. In both the discrete and continuous case, the relevant questions are: Which model is most useful? Which idealization does less violence to the phenomena?

Figure 6: Continuous transformation of images. Any image in the space can be transformed into any other by one or more processes of continuous transformation.

3 Postulates of Continuous Symbol Systems

Based on the foregoing phenomenological analysis we can now propose a candidate set of properties possessed by any continuous symbol system. In later sections we explore particular classes of such systems that have properties in addition to those enumerated here.

We begin with the syntax of continuous symbol systems. We have seen that in discrete symbol systems tokens are either of the same type or they are not, whereas in continuous symbol systems similarity is a matter of degree. It is generally unproblematic to assume that this degree of similarity is quantifiable and that the quantification has the properties of a *metric*, that is, a measure of distance, which is a binary function ρ from a space X to the real numbers, $\rho : X \times X \rightarrow \mathbf{R}$, that satisfies these identities:

$$\begin{array}{ll} \rho(x,x) = 0 & \text{Self-identity,} \\ \rho(x,y) = \rho(y,x) & \text{Symmetry,} \\ \rho(x,y) + \rho(y,z) \geq \rho(x,z) & \text{Triangle Inequality.} \end{array}$$

Therefore, unless stated otherwise, we assume that an image space is a metric space.[10]

The second syntactic invariance we address is that any image in the space may be continuously transformed into any other, which is expressed mathematically as follows. Let $a, b \in X$ be any two images in the space. We require that there be a continuous, one-to-one function $P : \mathbf{R} \rightarrow X$ such that for some $t_o, t_f \in \mathbf{R}$ (think of them as times) we have $P(t_o) = a$ and $P(t_f) = b$. The function P represents a continuous transformation of a into b. Without loss of generality we require $t_o = 0$ and $t_f = 1$. In topological terms P is a *path* or *arc* from a to b, and since we require there to be a path between any two images, the space is *path-connected* (*arcwise-connected*).[11]

Another invariance of continuous symbol systems is that for any two images a, b we can always find a third c closer to either (but not necessarily to both); that is, we can pick a c such that $\rho(a,c) < \rho(a,b)$

[10]Section 4.1.4 considers cases in which this assumption may be too strong.

[11]More carefully, P is a homeomorphism because $[0,1]$ is compact and X is Hausdorff (since all metric spaces are Hausdorff). Therefore P is a homeomorphism between $[0,1]$ and $P[0,1]$, which makes it a path (Moore, 1964, pp. 68, 71, 161).

and we can pick a c' such that $\rho(b, c') < \rho(a, b)$. However, we do not have to postulate this property, since it follows from image spaces being path connected: For example, since $c = P(t) \to b$ as $t \to 1$, c will eventually be in an arbitrarily small neighborhood of b, and thus can be made as close as we like to b.

A path-connected space is also connected in the more general sense, that is, it is not a union of separated sets. There is good evidence that image spaces are connected metric spaces, including:

1. A nontrivial[12] connected metric space has at least the cardinality of the real numbers (Hausdorff, 1957, p. 175).
2. In topology a *continuum* is defined to be a closed connected metric space (Hausdorff, 1957, p. 173).[13]
3. A finite or countable set is totally disconnected; therefore discrete symbol systems are totally disconnected (Hausdorff, 1957, p. 175).

On the other hand, we believe that image spaces must satisfy the stronger condition of being *path*-connected, since otherwise images would not necessarily be reachable by a *finite* process of continuous transformation.

The foregoing considerations lead us to propose:

Postulate 1 *Image spaces are path-connected metric spaces.*

Next we turn to a complex of properties that is at the heart of the continuity of image spaces: completeness and separability. A space is *complete* if all its Cauchy sequences have limits in the space. A space is *separable* if it has a countable dense subset, roughly, if all its images can be approximated by sequences of images with rational coordinates. The phenomenological analysis does not seem to require either property, and one can imagine image spaces that don't satisfy one or the other (for example, a disk missing its central point is not complete). Nevertheless, we tentatively assume both properties, because they are mathematically important and because most image spaces satisfy them. In particular, if a metric space is *compact* (per-

[12] Here 'nontrivial' means that it has more than one point.

[13] Sometimes a continuum is defined as a nontrivial compact connected space; the issue of compactness is addressed later.

haps the closest analog to the finiteness and definiteness of discrete formal systems), then it is both separable and complete.[14] Therefore:

Postulate 2 *Image spaces are separable and complete.*

Next we characterize mathematically the syntactic categories and syntactic relations of continuous symbol systems. From our phenomenological analysis (Section 2.2.3) we know that formal properties must vary continuously with changes in the images. Thus we propose:

Postulate 3 *Maps on image spaces are continuous.*

We take this to be the case for the syntactic maps upon which formal relations depend, but also for semantic maps between image spaces and their domains of interpretation (which thus must be continua).[15]

Finally, our phenomenological analysis has shown that processes in continuous formal system proceed continuously in time, and that infinitesmal changes in the state image do not result in behavioral discontinuities. This is formalized as follows.

In mathematical terms, a *process* is a continuous function $p : X \times \mathbf{R} \rightarrow X$ satisfying $p(s, 0) = s$ and $p[p(s, t_1), t_2] = p(s, t_1 + t_2)$ (the group property). Here X is the *state space* of the process and $p(s, t)$ is the state of the process time t after starting in state s. In addition, we allow the possibility that some processes are defined over only an interval of time, bounded or unbounded. For a fixed s, the function $m(t) = p(s, t)$ is called a *motion* and defines a *continuous curve* (Moore, 1964, p. 156). A *trajectory* is the set of images produced by a motion over an interval I of time, $m[I]$.

Given these definitions we propose:

Postulate 4 *A process in a continuous formal system is a continuous function of time and process state.*

[14] Indeed, many mathematicians define a continuum to be a nontrivial connected *compact* metric space.

[15] One consequence of this semantic rule is that exact interpretations of continuous symbol systems are inherently continuous; discrete interpretations can only be approximated. The situation is analogous for discrete symbol systems, the interpretations of which are inherently discrete, and for which continuous interpretations can only be approximated. We take this to be the import of the Löwenheim-Skolem Theorem, which states that any formal system (with a countable number of symbols) has a countable model. Therefore, any axiomatization of the real continuum cannot exclude countable, discrete interpretations (the Löwenheim-Skolem Paradox).

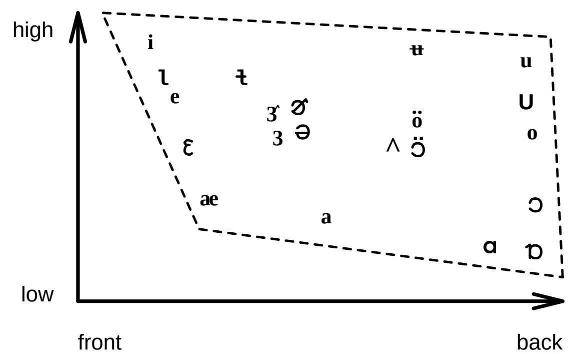

Figure 7: Continuous space of low dimension. The diagram shows the approximate location of the tongue when articulating the indicated vowels.

We take these postulates to be satisfied by any continuous symbol system. In the remainder of this paper we present a number of conclusions that can be drawn from these postulates as well as additional results for more specific classes of continuous symbol systems.

4 Connectionist Spaces

4.1 Topology

Although we have identified path-connected metric spaces with the general class of connectionist spaces, many of the latter have additional useful structure. Therefore, in this section we consider several important classes of connectionist spaces.

4.1.1 Finite-dimensional Euclidean Spaces

One common class of connectionist spaces is the class of continua that was the historically first to be recognized: finite-dimensional Euclidean spaces, E^n. This is the natural choice when images are defined by a few real parameters (Fig. 7), and it has been the mathematical context of most neural network theory.

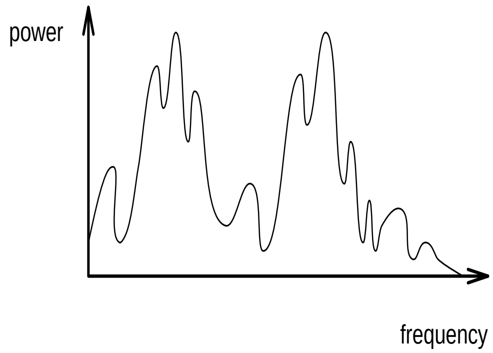

Figure 8: Continuous space of high or infinite dimension. The diagram represents the instantaneous power spectrum of a sound. Even though the number of dimensions may be finite, it is so large that the image is most usefully considered a continuous function (infinite-dimensional).

4.1.2 Hilbert Spaces

The use of finite-dimensional Euclidean spaces is less obvious when the number of dimensions is very large. Consider auditory images represented by instantaneous power spectra (Fig. 8). We can, of course, view these as members of a finite-dimensional space E^n, but in the case of human auditory images $n \approx 20\,000$ (Shepherd, 1988, p. 315). The situation is even worse for visual images (Fig. 9), where $n \approx 1.3 \times 10^8$ (the number of receptors; even the number of ganglion cells $\approx 10^6$) (McFarland, 1987, p. 588). Although from a mathematical standpoint $n = 1.3 \times 10^8$ is just as finite as $n = 2$, there are practical differences. At very least, it seems more natural to think of these images as continuous functions; their discreteness in fact is a physical detail that can often be ignored, like the discreteness of fluids in hydrodynamics.

For this reason we suggest that many connectionist spaces are best treated as *infinite*-dimensional Euclidean spaces, in other words, Hilbert spaces (see also Pribram, Chapter **??** in this volume).[16] Certainly they already provide the context for much of the theoretical work in vision, signal processing and image analysis, but in neural network research, finite dimensional vectors are still the norm. We have argued elsewhere that the most interesting neural networks —

[16]To be precise: The set of infinite-dimensional vectors of finite length are a Hilbert space, namely l_2.

Figure 9: Continuous space of very high dimension. A visual image such as this may have a dimension of 10^6 or even 10^8. It is much more reasonable to consider it a continuous function (infinite dimensional).

those with a *large* number of neurons — are best treated as having an *infinite* number of neurons (MacLennan, 1987a; 1987b; 1989a; 1989b; 1990). Such an approach abstracts away from the details of the neural fabric, which works well so long as the number of neurons is large enough.[17]

Another argument for Hilbert spaces is that they are — mathematically — where the continuous meets the discrete. The reason is the Riesz-Fischer Theorem, which is certainly one of the most profound in mathematics: L_2 is isomorphic and isometric to l_2, the set of square-integrable functions is isomorphic and isometric to the set of square-summable sequences of reals. This is the basis for expansions of functions as infinite series, including the Fourier. In particular, this shows how a discontinuous function, such as a step function, can emerge from a superposition of continuous functions, such as sinusoids. Thus we may hope that Hilbert spaces may provide a mathematical context for understanding the emergence of discrete (or nearly discrete) symbols from the subsymbolic continuum. Further evidence of the relevance of Hilbert spaces to the relation between continuous and discrete representations can be found in MacLennan (1991a).[18]

---

[17]Pribram (1991, 1992) has also argued for the use of Hilbert spaces as a framework in which to define the "neural wave equation."

[18]Indeed, the central importance of Hilbert spaces is shown by a theorem of Urysohn's which states that any metric space with a countable base is homeomorphic to some subset

4.1.3 Metric Spaces

Even with a low-dimensional space like that in Fig. 7, there is little reason to suppose that the Euclidean metric is always appropriate. For example, it is quite possible that perceptual similarity is more sensitive to high/low position than to front/back position, or vice versa. Further, this sensitivity difference might vary over the space. In other words, there is no guarantee that the equal-similarity contours around a sound are the circles that are defined by the Euclidean metric; they could be ellipses or even less regular curves. Thus, connectionist spaces need not be *isotropic* (the same in all directions). Isotropy is even less likely when we consider higher dimensional spaces such those indicated in Figs. 8 and 9, since similarity is unlikely to be equally sensitive to differences throughout the function's domain.

One simple improvement is to attach a weight function to the Euclidean metric. This allows differing sensitivities across the image, so that, for example, visual similarity may depend more on the centers of images than their peripheries, and auditory similarity may depend more on the midband (if that is what we want).

Since the provision of a fixed weight function still restricts the set of possible metrics more than we would like, a still more general notion of distance if often useful. In fact, we often do not know or care how the similarity of images depends on their microfeatures. In this case it is better to assume only that some quantifiable measure of distance holds among the images, that is, that they belong to a *metric space*. However, as we saw before (Section 3), it is also necessary to assume that the space is path-connected.

4.1.4 Semimetric Spaces

In some cases even a metric space may imply too much structure; certainly the triangle inequality (see Section 3) is problematic for some cognitive images. Fortunately, some results are obtainable even for *semimetric spaces*, which have a distance measure that need not satisfy the triangle inequality (MacLennan, 1988a). Nevertheless, in this paper we assume that all connectionist spaces are metric spaces.

---

of the Hilbert space E^{∞} (Nemytskii & Stepanov, 1989, p. 324). Note however that the homeomorphism need not preserve the metric.

4.2 Finite Decomposition of Spaces

We have seen (Section 2) that discrete and continuous formal systems differ in what is taken as *unproblematic givens*. In a discrete system, the atomic components are given, and these are assembled into more complex formulas through the use of syntactic relationships. If these relations are unambiguous, as is usually the case, then any formula can be decomposed in a unique way into its atomic components. On the other hand, in continuous formal systems, whole images are usually the unproblematic givens. Further, it is normal that there are many competing decompositions into lower-level images, and the appropriate decomposition often depends on the use to which it will be put. Finally, there is often no "bottom" to the decomposition; that is, there is no natural notion of atomic components.

It is important to realize that this problem is not just theoretical; it pervades empirical investigations into the "representational primitives" of sensory and motor systems. This is apparent in early vision research, where there is ongoing debate about whether images in the visual cortex have as elementary components oriented edges, wavelets, radial basis functions, two-dimensional Gabor functions, three-dimensional Gabor functions, etc. (MacLennan, 1991a). Sometimes it is not even obvious what are the representational alternatives, and techniques such as *multidimensional scaling* have been used in an attempt to find *possible* decompositions of a space into subspaces (Shepard, 1980). The continuous formal system viewpoint suggests that in some cases images may be processed holistically, that is, without decomposition, and in other cases by simultaneously using several incompatible decompositions.

Before discussing the mathematical decomposition of continua, it is necessary to say a few words about the suggestive but misleading terms *analytic* and *synthetic*. Perhaps because of their association with analytic and synthetic cognitive styles (e.g., Churchland, 1986, p. 199; Gregory, 1987, p. 744; Vernon, 1962, pp. 221–224), there is a natural tendency to consider discrete systems analytic and continuous systems synthetic. Unfortunately, there is another perspective on these terms that would use them in exactly the opposite way, for we have seen that discrete formulas are usually seen as being built up — synthesized — from atomic components, and continuous images are seen as being decomposed — analyzed — into simpler images. From this viewpoint, discrete systems are synthetic and continuous systems analytic.

Perhaps we can understand this paradox as follows. Since in a discrete symbol system the decomposition of a formula into its constituents is generally unproblematic, it it natural for discrete processes to be defined in terms of processes operating on the constituents, which is consistent with one definition of *analytic*: "Reasoning from a perception of the parts and interrelationships of a subject" (Morris, 1981, s.v. analytic). In other words, the process operates on the *analysis*, the separation of the whole into its constituents (Morris, 1981, s.v. analysis). Conversely, since in continuous systems decomposition is often problematic, it is natural for these processes to operate directly on the *synthesis*, the coherent whole resulting from a combination of elements (Morris, 1981, s.vv. synthetic, synthesis). Although this is perhaps the explanation of the paradoxical use of these terms, for the sake of clarity I avoid them whenever possible.

In the simplest case a set of images can be expressed as a Cartesian product of two or more other sets of images, $X = Y \times Z$. We have an example of this in Fig. 7, which shows the decomposition of the set of vowel sounds into the sets of front/back position and high/low position. In general, such decompositions are not obvious, and may require extensive experiments for their discovery.

Decomposition of a space involves more than simply expressing its set of images as a Cartesian product of other sets of images, for we must also show how the topology of the composite space results from the topologies of the constituent spaces. For example, in Fig. 7 we need to know how similarity or distance between vowels relates to similarity or distance in each of the two dimensions of tongue position. We cannot assume the obvious Euclidean relationship, $\rho^2[(x_1, x_2), (y_1, y_2)] = \rho_1^2(x_1, y_1) + \rho_2^2(x_2, y_2)$. At very least, the component distances might have different weights,

$$\rho^2[(x_1, x_2), (y_1, y_2)] = w_1\rho_1^2(x_1, y_1) + w_2\rho_2^2(x_2, y_2).$$

Further, the decomposition need not even be Euclidean (l_2), for we could have a different l_p ($p \neq 2$) decomposition rule:

$$\rho^p[(x_1, x_2), (y_1, y_2)] = w_1\rho_1^p(x_1, y_1) + w_2\rho_2^p(x_2, y_2),$$

or even an l_∞ decomposition:

$$\rho[(x_1, x_2), (y_1, y_2)] = \max[w_1\rho_1(x_1, y_1), w_2\rho_2(x_2, y_2)].$$

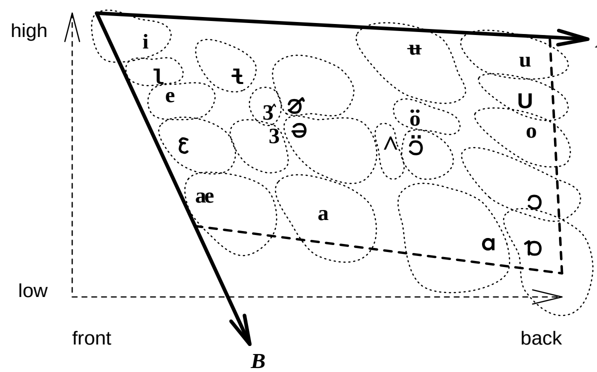

Figure 10: Alternate decomposition of a low dimensional space. The diagram indicates how a different set of axes might better decompose the two-dimensional metric (indicated by equal-distance contours). (Contours are for the sake of the example and do not represent the actual range of the vowels.)

No doubt more complex decompositions may also occur, and for some metric spaces there may be *no* practical decomposition.

It is also possible that we might find that a *different* choice of axes effects a better decomposition of the metric (which is the point of multidimensional scaling and many other statistical techniques). Figure 10 shows a case in which an alternative decomposition is more consistent with the equal-similarity contours around a set of points in vowel space.

4.3 Recursive Decomposition of Spaces

Recursive structures, which allow finite but unlimited nesting, such as trees, have been a mainstay of the analysis of linguistic structures since at least the time of Chomsky. The very flexibility of recursive structures has brought with it a problem: actual linguistic performance does have its limits, whereas the theory demands no limits. This has led to a central dogma, the *competence/performance distinction*, that is, our theoretical (but never observed) linguistic competence is distinguished from our actual linguistic performance. Linguists in the Chomskian tradition have tended to concentrate on the *ideal* — competence, and have mostly ignored the *real* — performance.

In contrast, from its beginnings connectionism has taken performance into account; witness the "100 step rule." Thus we may hope that connectionism will provide a model of recursive nesting that effects a better reconciliation between competence and performance. This may follow from the theory of continuous formal systems, as I indicate below.

Consider a simple example, a space X of binary trees. We must have a construction operation $\oplus : X \times X \to X$ that joins two binary trees $x, y \in X$ into a larger tree $x \oplus y \in X$. Conversely, we also need operations to extract the left and right subtrees of a composite tree, $\mathsf{left}(x \oplus y) = x$, $\mathsf{right}(x \oplus y) = y$. Finally, we require the operations $\oplus$, **left** and **right** to all be continuous, since that is an axiom of continuous formal system theory. This means that there is a *homeomorphism* (one-to-one, continuous map) between the space X of binary trees and its possible decompositions: the space L of *leaves* and the space $X \times X$ of pairs of trees. Thus,

$$X \cong L \sqcup X \times X,$$

where $\cong$ represents homeomorphism and $\sqcup$ represents disjoint union.

With this background, we can now address the question of the *continuous* recursive decomposition of a space. Our first result is that this is *impossible for finite-dimensional Euclidean spaces*, since Brouwer's theorem of the Invariance of Dimensionality shows that Euclidean spaces E^m and E^n are not homeomorphic if $m \neq n$ (Hausdorff, 1957, p. 232); indeed even subsets of these spaces cannot be homeomorphic (provided the subset of the higher dimensional space has interior points). In other words, finite-dimensional Euclidean spaces are characterized by their dimension. Therefore $E^n \times E^n \cong E^{2n}$ and is not homeomorphic to any subspace of E^n. Thus arbitrary trees or sequences cannot be represented continuously in finite-dimensional Euclidean spaces; we must turn to richer spaces.

We have already seen that images are often conveniently represented as continuous functions, which belong to *infinite*-dimensional Euclidean spaces, so they are the next candidates we consider.

We also observe that many infinite spaces are homeomorphic to two or more disjoint subspaces of themselves. For example, the unit interval $[0, 1]$ is homeomorphic to the one-third intervals $[0, 1/3]$ and $[2/3, 1]$ as well as to many others. The unit square $[0, 1]^2$ can also be embedded in itself in many different ways. Self-embeddable spaces such as these suggest one mechanism for the continuous recursive composition and decomposition of spaces.

Suppose images are represented by continuous functions $f : \Omega \to Y$ in some function space $\Phi(\Omega)$. Most of the domains Ω in which we are interested can be homeomorphically embedded in themselves in two or more different ways, so suppose $h : \Omega \to h[\Omega]$, $h' : \Omega \to h'[\Omega]$ are homeomorphisms such that $h[\Omega]$ and $h'[\Omega]$ are separated subsets of Ω. We now define the construction of f and g, $f \oplus g$, to be any continuous $c : \Omega \to Y$ such that $f = c \circ h$ and $g = c \circ h'$, where $\circ$ indicates composition. The selection operations are simply $\mathsf{left}(c) = c \circ h$ and $\mathsf{right}(c) = c \circ h'$.

To make these ideas clearer we present a concrete example, binary trees of one-dimensional images represented as continuous functions over $[0, 1]$. Let $\Omega = [0, 1]$; there are homeomorphisms h, h' such that $h[[0, 1]] = [0, 1/3]$ and $h'[[0, 1]] = [2/3, 1]$, namely $h(x) = x/3$, $h'(x) = (2 + x)/3$. For the construction $x \oplus y$ we take any continuous interpolation between $x(1/3)$ and $y(2/3)$; see Fig. 11. Clearly the same kind of construction could be used for images over the unit square. In this case we can map $[0, 1] \times [0, 1]$ into $[0, 1/3] \times [0, 1]$ and $[2/3, 1] \times [0, 1]$.

We consider now an alternative representation of continuous recursive spaces that makes use of the properties of Hilbert spaces. Suppose $f, g \in L_2(\Omega)$. Then we can represent them by generalized Fourier series: $f = \sum c_k e_k$, $g = \sum d_k e_k$, where $\{e_k\}$ is any orthonormal basis for $L_2(\Omega)$. Next represent the construction of f and g by the interleaved series: $f \oplus g = \sum c_k e_{2k} + d_k e_{2k+1}$ (Fig. 12). This operation is a homeomorphism; indeed, it is linear (versus bilinear) on $L_2(\Omega) \times L_2(\Omega)$, and even an isometry, since $||f \oplus g||^2 = ||f||^2 + ||g||^2$. Clearly the component selector functions **left** and **right** are easily defined.

We make several observation about the continuous recursive representations that we have defined. First, we can construct binary trees to an arbitrary (finite) depth, since if x, y and z are in the space X, then so are $x \oplus y$, $(x \oplus y) \oplus z$, etc. Similarly, if $x \in X$, then we can select its right and left components, $\mathsf{left}(x)$, $\mathsf{right}(x)$, *regardless of whether it resulted from a construction.* In this representation there are no atomic symbols (leaves); we can always "go deeper" in our analysis. This potentially bottomless recursive decomposition seems to agree with the properties of continuous images revealed by our phenomenological analysis. We also observe that a temporal sequence of

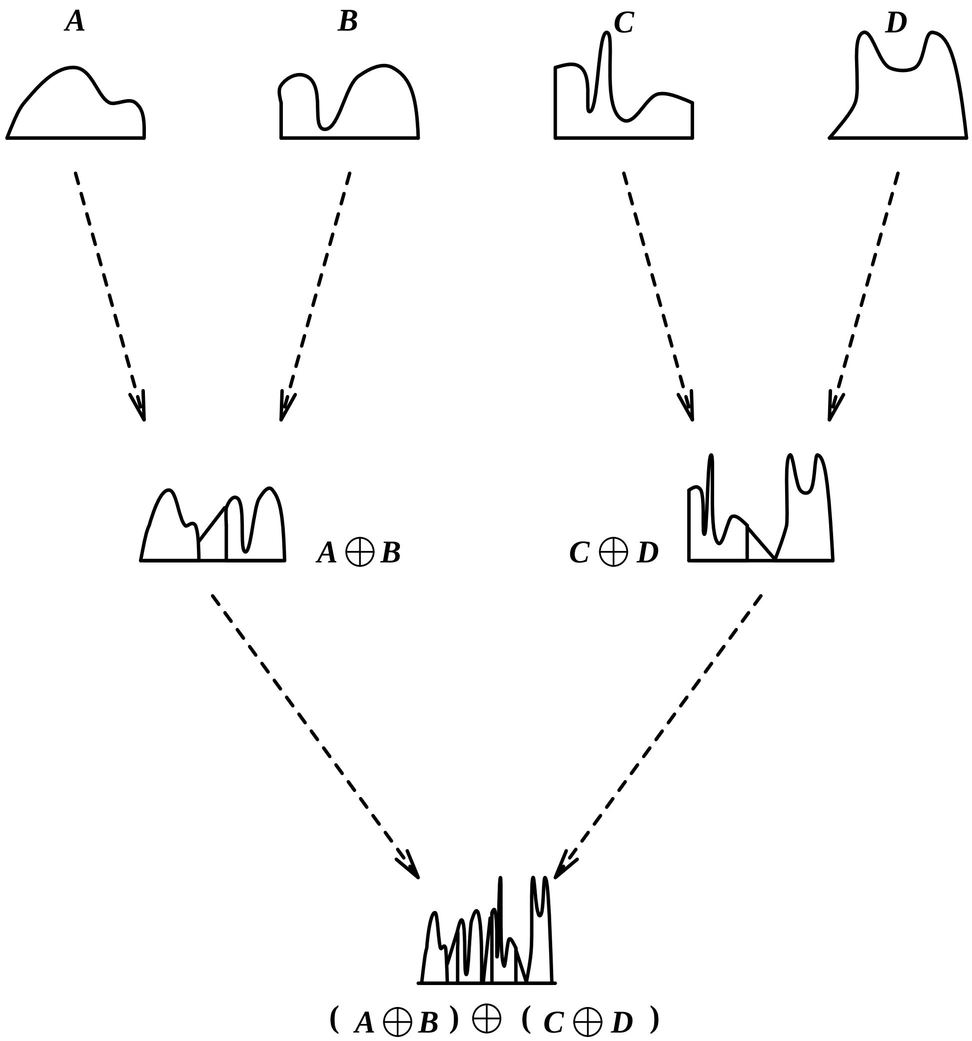

Figure 11: Recursive structure represented by functions over self-similar spaces. For the sake of example, we assume that the domain of all the functions is the interval $[0,1]$, which is homeomorphic to disjoint subintervals of itself, such as $[0, 1/2-\epsilon]$ and $[1/2+\epsilon, 1]$. Two functions A, B are combined by contraction, making $[0, 1/2-\epsilon]$ the domain of A and $[1/2+\epsilon, 1]$ that of B. Continuity is preserved by interpolation between $A(1/2-\epsilon)$ and $B(1/2+\epsilon)$. Depth of recursive nesting is limited only by the ability of the underlying medium to sustain higher gradients.

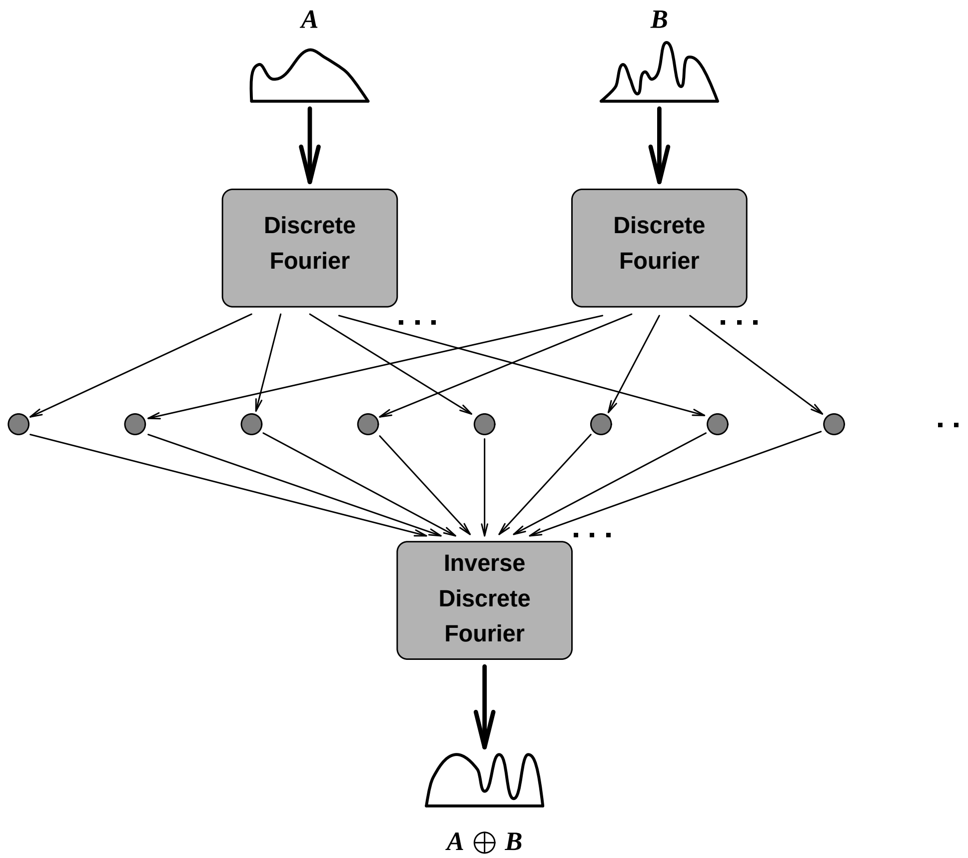

Figure 12: Recursive nesting in function spaces. Suppose $c_0, c_1, c_2, \ldots$ are the Fourier coefficients of A and $d_0, d_1, d_2, \ldots$ are the Fourier coefficients of B. Then the pair (A, B) is represented by the function whose coefficients are $c_0, d_0, c_1, d_1, c_2, d_2, \ldots$, the interleaved coefficients of A and B. Recursive nesting is limited only by the ability of the underlying medium to sustain higher frequencies. (Waveforms shown are merely schematic.)

images x_t can be recursively folded into a sequence s by the formula $s_{t+1} = x_t \oplus s_t$, which defines a right branching binary tree.

In both of the representations we have defined, successive composition pushes deeper information (for trees) or earlier information (for sequences) into higher frequency bands. Ideally, this doesn't matter; from a mathematical standpoint all bands are equally recoverable. Practically, however, physical media will not sustain arbitrarily high frequencies; also noise tends to be high frequency. Thus from a practical standpoint, components that are deeper or more in the past are progressively less recoverable. This seems to be a natural model of the competence/performance distinction, since noise and physical properties of the media limit performance to less than its theoretical competence. For example, arbitrarily large trees or sequences could be represented in the ideal continuous neural tissue. But real neural tissue, being composed of discrete neurons, places an upper limit on representable spatial frequency, and so on the size of trees or sequences.

5 Connectionist Maps

5.1 Mathematical Properties

Nonrecurrent neural networks implement a map between two spaces. For example, associative memories, filters, pattern classifiers and feature extractors can often be implemented without recurrent connections. The only restriction we have postulated on such maps is that they be mathematically continuous, and for this we can take whatever definition of continuity is most appropriate to the spaces being mapped. For example, if they are metric spaces, then $f : S \rightarrow T$ is continuous at a point x if and only if for all $\epsilon > 0$ there exists a $\delta > 0$ such that $\rho_T[f(x), f(y)] < \epsilon$ whenever $\rho_S(x, y) < \delta$. If the spaces are Euclidean, then we can use the Euclidean metric. If the spaces are not metric, then continuity must be defined in terms of open sets. Next we consider the consequences of this postulate.

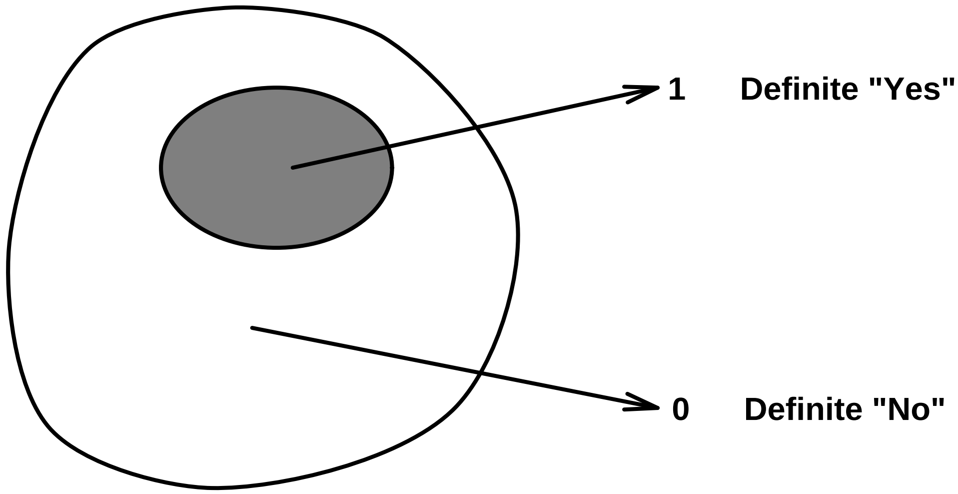

Figure 13: Requirements for exact classification in a continuous symbol system. We require a *continuous* function that is 1 on the category and 0 on the complement of the category. This is impossible in a continuum.

5.2 Categorization

5.2.1 Exact Categorization Impossible

Suppose we wish to divide a continuum S into two disjoint, mutually exclusive categories, A and non-A. Thus, for every image $x \in S$ we want a continuous map $f : S \to \{0,1\}$ such that $f(x) = 1$ if x is in category A and $f(x) = 0$ if it is not.[19] We may take this as a precise statement of the problem of exact categorization (Fig. 13).

Our first important result from the theory of continuous formal systems is that *exact categorization is impossible.* This is because it is easy to show that a space is connected if and only if it cannot be continuously mapped to a nontrivial discrete space (i.e. a space with more than one point) (Moore, 1964, p. 66). That is, a continuum cannot be discretized by a continuous map. Therefore we have:

Theorem 1 (Exact Categorization) *A continuous formal system cannot perform exact classification.*

Notice that this result is quite robust, since it follows from only two

[19]The use of 0 and 1 is not important; any discrete space with two elements would do.

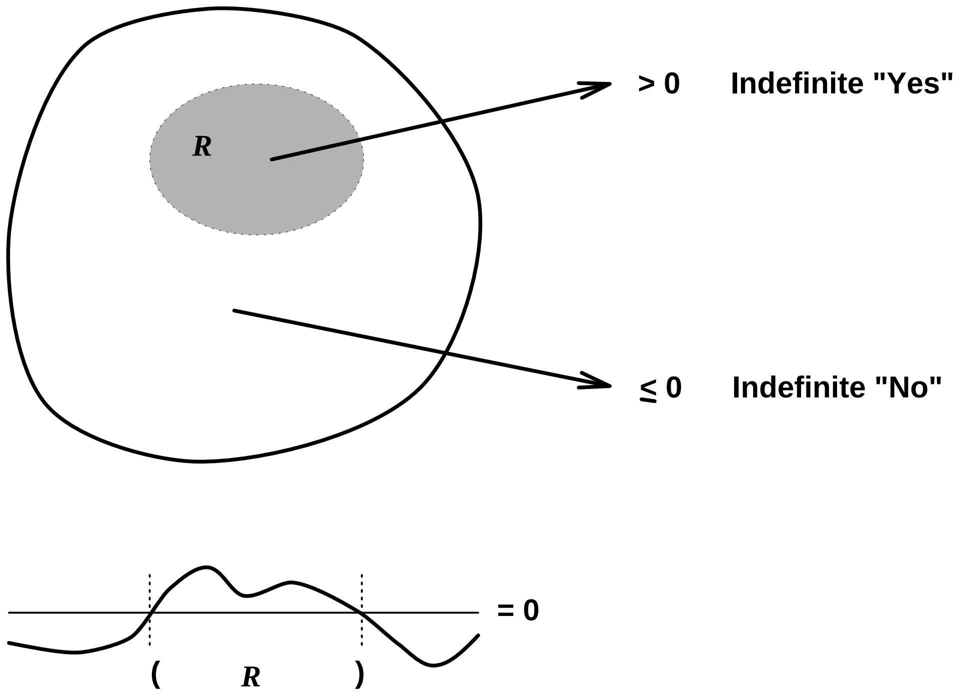

Figure 14: One form of classification permitted in continuous formal systems. We are permitted a function that is positive on the category (an open region) and nonnegative on its complement, but it must be continuous, so it will be arbitrarily close to zero near the boundary.

assumptions: connectionist spaces are connected, and connectionist maps are continuous.

5.2.2 Connectionist Categorization

Given that exact categorization is impossible, we must consider the kinds of categorization possible to connectionist systems. We find that various topological separation axioms correspond to various kinds of categorization; we consider several examples.

It is easy to show that for each open set in a metric space, there is a real-valued continuous function that is positive just on the set (Hausdorff, 1957, p. 129). Notice, however, that the boundary is fuzzy; hard thresholds are not possible (Fig. 14). As images approach the boundary, the categorization must leave the certain values (say ± 1)

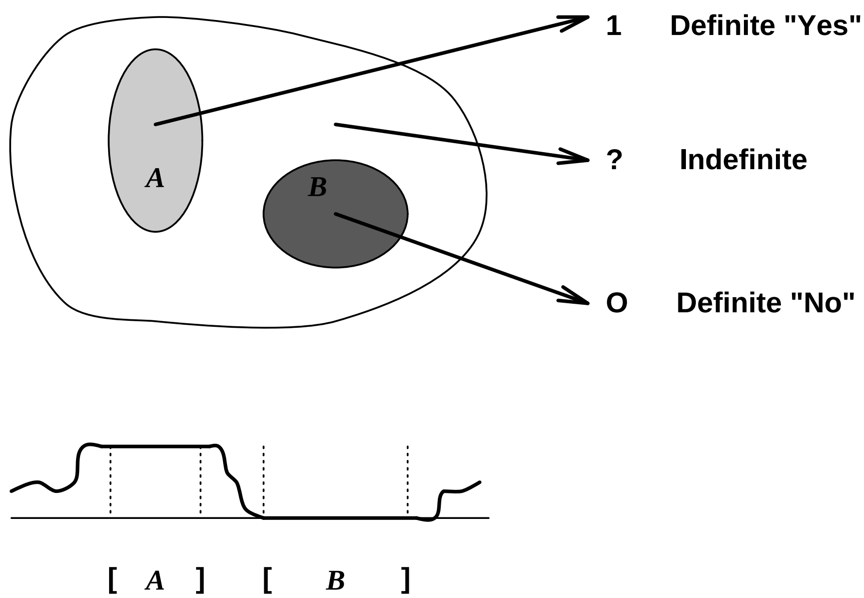

Figure 15: Classification by Urysohn's Lemma. We are permitted a closed region on which the classification is definitely "yes," and a closed region on which it is definitely "no," but there must be a nonempty region between the two where the classification is indefinite.

and approach uncertainty, 0. We can, of course, in principle make the uncertain area as small as we like, but the Exact Categorization Theorem says that we can never decrease it to zero.

In a *normal* topological space (such as a metric space) Urysohn's Lemma forms a basis for categorization (Moore, 1964, p. 122): For a pair of nonempty disjoint closed subsets of the space, there is a continuous map into $[0, 1]$ that is 1 on one subset and 0 on the other (Fig. 15). This captures the idea of two categories being mutually exclusive, but preserves their essential fuzziness. That is, we can have "definitely A"and "definitely B" provided the remainder of the space is indefinite (varies between the two). (There must be a remainder, since otherwise the space would be the union of separated sets, and hence disconnected.)

In *completely regular* spaces (such as metric spaces) we have the following categorization theorem (Moore, 1964, p. 132): For each point in the space and each neighborhood of the point, there is a continuous

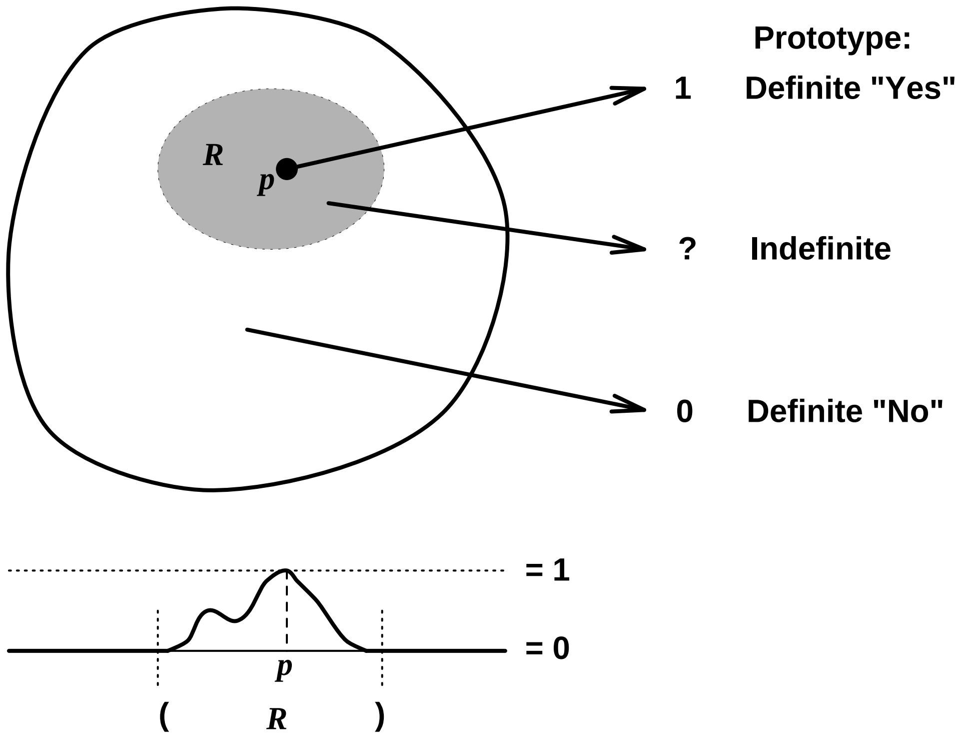

Figure 16: Classification relative to a prototype. We are permitted a function that is definitely "yes" for the prototype, and definitely "no" for images sufficiently far from the prototype, but it must vary continuously between these extremes.

map into $[0, 1]$ that is 1 at the point and 0 outside the neighborhood (Fig. 16). This too gives a kind of category to which some points definitely do not belong; however the category itself is defined relative to the point as exemplar. We have a sense of an image being "too far away" from the exemplar.

6 Connectionist Processes

6.1 Decidability

When we consider a new notion of computation, such as is provided by continuous formal systems, the question immediately arises of whether

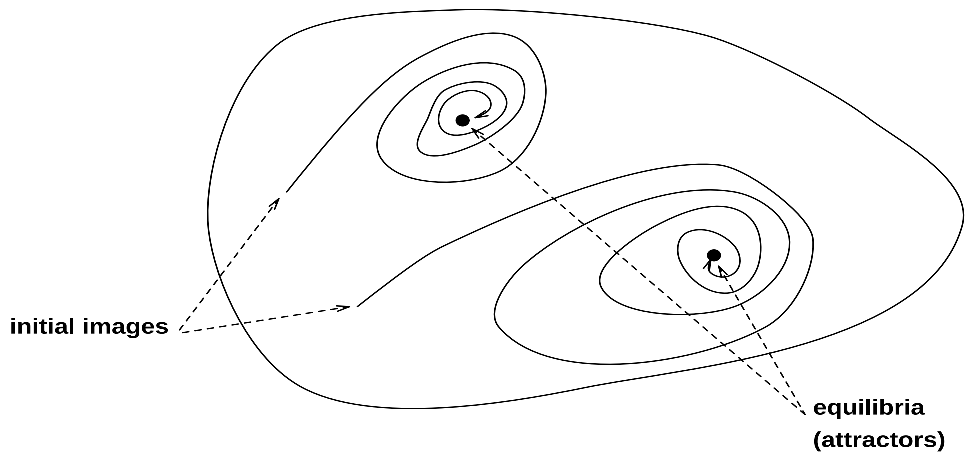

Figure 17: Decisions. In a continuous formal system a "decision" is an asymptotically stable equilibrium, a state which once approached will not be left.

they are subject to the same undecidability results as are discrete formal systems.[20]

However, before the question of decidability can even be addressed, we must ask what constitutes a decision in the context of continuous formal systems. Intuitively, making a decision is reaching a definite state that will not be later left (Fig. 17). (More accurately, the decision is required to be stable only so long as the context is stable. That is, reasoning is monotonic in an unchanging context, but a change of context may destabilize the decision.) In mathematical terms, *a continuous decision is an asymptotically stable equilibrium*: it remains so long as the context and initial conditions are fixed.[21] Then, determining if an image is decidable is accomplished by determining whether it is in the basin of attraction of some equilibrium. Conversely, an image is undecidable if it is outside all basins of attraction (Fig. 18).

Suppose we have a stable state with a corresponding basin of attraction D. Call an image undecidable if it is outside this basin. Thus

[20]Two other extensions of computation into the continuous realm are Blum (1989), Blum, Shub, & Smale (1988) and Stannett (1990). Their notions of decidability are somewhat different from ours.

[21]For some purposes a mere stable equilibrium can be considered a decision: in effect the system has settled into a set of possible results.

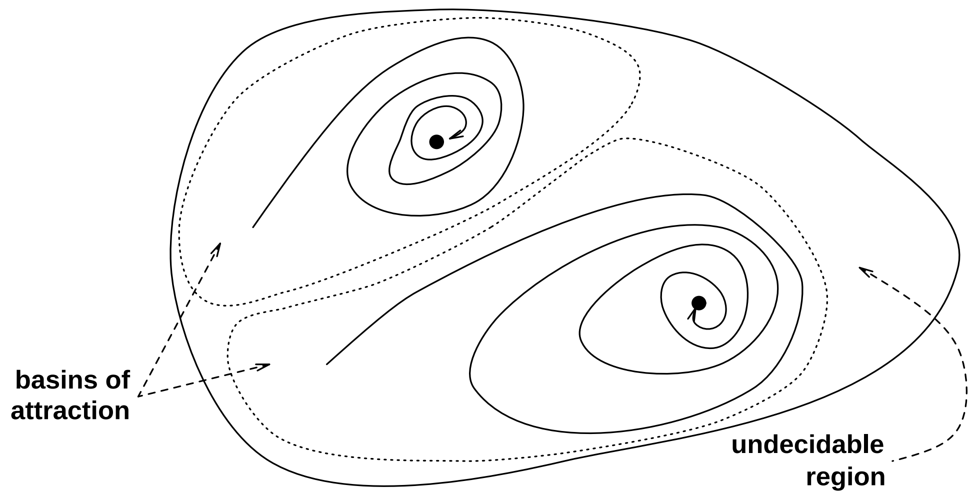

Figure 18: Decision basins. A "decision basin" is the basin of attraction of an asymptotically stable equilibrium. An *undecidable image* is one that is outside of all decision basins.

D represents the decidable images. Obviously we can define a function that is 1 on D and 0 on $X-D$. Thus, if we do not restrict ourselves to continuous formal systems and use instead our familiar discrete logic, then we can easily talk about the decidable and undecidable images. It is a sharp distinction: D vs. $X-D$.

On the other hand, from *within* a continuous formal systems, in order to categorize an image as decidable or undecidable, we need a *continuous* decision function $d \rightarrow \{0,1\}$ such that $d[D] = \{1\}$ and $d[X-D] = \{0\}$. But this is impossible for continua; it is just the exact categorization problem. As we saw, continuous categories are always fuzzy, and this includes the category 'decidable'. Notice, however, that this "undecidability result" comes from the fact that *in a continuous formal system it is impossible to ask a yes-or-no question.* Thus, from the perspective of continuous logic, *the decidability question is not even well-formed.*

What kinds of question *can* we ask a continuous formal system to decide? Here is one example. We can define a "definitely undecidable" set $U \subset X - D$ (proper subset). If D and U are disjoint closed sets, then we can define a continuous f such that $f[D] = \{1\}$ and $f[U] = \{0\}$, but there will remain a fuzzy region $X - D - U$ between

D and U.[22]

These decidability results have nowhere near the significance of the classical results of Gödel and Turing, but they do illustrate the fact that continuous formal systems bring with themselves an entirely new way of *asking* these questions. We hope that the future will bring deeper insights into decidability in both discrete and continuous formal systems.

7 Conclusions

We have argued that connectionist knowledge representation requires a theory of continuous symbol systems analogous to the theory of discrete symbol systems, which informs our understanding of traditional ("symbolic") knowledge representation. A phenomenological analysis exposed the differences between the two kinds of symbol systems, and revealed invariances that are important to capture in the mathematics. Based on this analysis we tentatively concluded that continuous symbol systems are characterized by path-connected metric spaces which are separable and complete, and by continuous maps and processes over those spaces.

Next we considered a number of connectionist spaces and concluded that many are profitably treated as Hilbert spaces. Problems associated with the decomposition of image spaces were addressed, including especially the possibility of recursive decomposition, which was shown to require function spaces (such as Hilbert spaces).

We found that the continuity of connectionist maps precludes exact classification, but does permit other kinds of classification that are more robust and less likely to lead to brittleness. Finally I argued that in the context of continuous formal systems, decisions are equivalent to attractors and that an image is decidable when it is in a basin of attraction. We discovered that yes-or-no decidability questions cannot be formulated in continuous symbol systems, and therefore that a theory of continuous decidability must take a different form from that for discrete systems.

A few words about future research. I do not see much need for a lot of additional work trying to establish which mathematical structure is the "right" formalization of continuous symbol systems. Sometimes

[22]This construction presumes that the space is normal, e.g., a metric space.

Hilbert spaces will be the best model, and other times connected metric spaces or finite-dimensional Euclidean spaces. This is analogous to discrete symbol systems, which are sometimes assumed to be deterministic and other times nondeterministic, sometimes assumed to have a finite number of types, other times a denumerably infinite number, and so forth. However, I do anticipate significant research in other areas, including continuous knowledge representation and an expanded theory of computability and decidability.

References

Arnheim, R. (1971). *Visual thinking.* Berkeley, CA: University of California Press.

Blum, L. (1989). *Lectures on a theory of computation and complexity over the reals (or an arbitrary ring)* (Report No. TR-89-065) Berkeley, CA: International Computer Science Institute.

Blum, L., Shub, M., & Smale, S. (1988). On a theory of computation and complexity over the real numbers: NP completeness, recursive functions and universal machines. *The Bulletin of the American Mathematical Society, 21*, 1–46.

Buchler, J. (Ed.). (1955). *Philosophical writings of Peirce.* New York, NY: Dover.

Churchland, P. S. (1986). *Neurophilosophy: Toward a unified science of the mind-brain.* Cambridge, MA: MIT Press.

Dreyfus, H. L. (1979). *What computers can't do: The limits of artificial intelligence* (rev. ed.). New York, NY: Harper & Row.

Goodman, N. (1966). *The structure of appearance* (2nd ed.). Indianapolis, IN: Bobbs-Merrill.

Goodman, N. (1968). *Languages of art: An approach to a theory of symbols.* Indianapolis, IN: Bobbs-Merrill.

Gregory, R. L. (Ed.). (1987). *The Oxford companion to the mind.* Oxford, UK: Oxford University Press.

Hausdorff, F. (1957). *Set theory* (John R. Aumann & al., Trans.). New York, NY: Chelsea.

Ihde, D. (1977). *Experimental phenomenology: An introduction.* New York, NY: Capricorn Books, G. P. Putnam's Sons.

Ihde, D. (1986). *Consequences of phenomenology.* Albany, NY: State University of New York Press.

Lakoff, G. (1988). A suggestion for a linguistics with connectionist foundations. In D. Touretzky, G. Hinton and T. Sejnowski (Eds.), *Proceedings of the 1988 Connectionist Models Summer School* (pp. 301–314). San Mateo, CA: Morgan Kaufmann Publishers.

Lhevinne, J. (1972). *Basic principles of pianoforte playing.* New York, NY: Dover.

MacLennan, B. J. (1987a). Technology-independent design of neurocomputers: The universal field computer. In M. Caudill & C. Butler (Eds.), *Proceedings, IEEE First International Conference on Neural Networks* (Vol. 3, pp. 39–49). New York, NY: Institute of Electrical and Electronic Engineers.

MacLennan, B. J. (1987b). *Field computation and nonpropositional knowledge* (Report No. NPS52-87-040). Monterey, CA: Naval Postgraduate School, Computer Science Department.

MacLennan, B. J. (1988a). Logic for the new AI. In J. H. Fetzer (Ed.), *Aspects of artificial intelligence* (pp. 163–192). Dordrecht, Holland: Kluwer Academic Publishers.

MacLennan, B. J. (1988b). Field computation: A model of massively parallel computation in electronic, optical, molecular and biological systems. Extended abstract in *Proceedings of AAAI Spring Symposium. Parallel Models of Intelligence: How Can Slow Components Think So Fast?* (pp. 180–183). Stanford, CA: American Association for Artificial Intelligence.

MacLennan, B. J. (1989a). *Continuous computation: Taking massive parallelism seriously* (Report No. CS-89-83). Knoxville, TN: University of Tennessee, Department of Computer Science. Also, poster presentation, *Los Alamos National Laboratory Center for Nonlinear Studies 9th Annual International Conference: Emergent Computation.* Los Alamos, NM.

MacLennan, B. J. (1989b). Outline of a theory of massively parallel analog computation. Abstract in *Proceedings, IEEE/INNS*

International Joint Conference on Neural Networks (Vol 2, p. 596). New York, NY: Institute of Electrical and Electronic Engineers. For full text see Report No. CS-89-84, Knoxville, TN: University of Tennessee, Department of Computer Science.

MacLennan, B. J. (1990a). *Field computation: A theoretical framework for massively parallel analog computation, parts I–IV* (Report No. CS-90-100). Knoxville, TN: University of Tennessee, Department of Computer Science.

MacLennan, B. J. (1991a). *Gabor representations of spatiotemporal visual images* (Report No. CS-91-144). Knoxville, TN: University of Tennessee, Computer Science Department. Also submitted for publication.

MacLennan, B. J. (1991b). *Characteristics of connectionist knowledge representation* (Report No. CS-91-147). Knoxville, TN: University of Tennessee, Computer Science Department. Also submitted for publication.

Marrow, A. J. (1969). *The practical theorist: The life and work of Kurt Lewin.* New York, NY: Basic Books.

McFarland, D. (Ed.). (1987). *The Oxford companion to animal behavior.* Oxford, UK: Oxford University Press.

Moore, T. O. (1964). *Elementary general topology.* Englewood Cliffs, NJ: Prentice-Hall.

Morris, W. (Ed.). (1981). *The American heritage dictionary of the English language.* Boston, MA: Houghton Mifflin Company.

Nalimov, V. V. (1981). *In the labyrinths of language: A mathematician's journey* (R. O. Colodny, Ed.). Philadelphia, PA: ISI Press.

Nemytskii, V. V., & Stepanov, V. V. (1989). *Qualitative theory of differential equations.* New York, NY: Dover.

Pribram, K. H. (1991). *Brain and perception: Holonomy and structure in figural processing.* Hillsdale: Lawrence Erlbaum.

Pribram, K. H. (1992). Brain and the structure of narrative. This volume.

Shepard, R. N. (1980). Multidimensional scaling, tree-fitting, and clustering. *Science, 210*, 390–398.

Shepherd, G. M. (1978). Microcircuits in the nervous system. *Scientific American, 238*, 92–103.

Shepherd, G. M. (1988). *Neurobiology* (2nd ed.). New York, NY: Oxford University Press.

Stannett, M. (1990). X-machines and the halting problem: Building a super-Turing machine. *Formal Aspects of Computing, 2*, 331–341.

Vernon, M. D. (1962). *The psychology of perception.* Baltimore, MD: Penguin Books.