Conceptual Hierarchies in Classical and Connectionist Architecture*

Alfred Kobsa[1]

TR-89-010

February, 1989

Representation systems for conceptual hierarchies have been used in the field of Artificial Intelligence for nearly two decades. They are based on symbolic representation structures and sequential processes operating upon these structures. Recently, a number of network structures have been developed in the field of Connectionism which are also claimed to be able to represent conceptual hierarchies. Processes in these networks operate in a parallel way and largely without a global control mechanism. This paper investigates the expressive power, interpretation, and inferential capabilities of these networks as compared to traditional representations of concept hierarchies in particular to KL-ONE, a standard representation language for conceptual hierarchies in the field of natural-language processing. Although the capabilities of current connectionist hierarchies fall short of traditional representations, three inference processes will be described which can be very easily and elegantly realized in a connectionist architecture whilst they are hard and cumbersome to implement in traditional knowledge representation systems.

[1]International Computer Science Institute, Berkeley, California, on leave from the University of Saarbruecken, West Germany.

*I would like to thank Joachim Diederich, Jerry Feldman, Mark Fanty, Susan Hollbach Weber, Christel Kemke, Mark Line, Lokendra Shastri, Robert Wilensky and Dekai Wu for comments on an earlier version of this paper or general discussions on the issues covered therein.

Conceptual Hierarchies in Classical and Connectionist Architecture*

Alfred Kobsa
International Computer Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704-1105

Abstract

Representation systems for conceptual hierarchies have been used in the field of Artificial Intelligence for nearly two decades. They are based on symbolic representation structures and sequential processes operating upon these structures. Recently, a number of network structures have been developed in the field of Connectionism which are also claimed to be able to represent conceptual hierarchies. Processes in these networks operate in a parallel way and largely without a global control mechanism. This paper investigates the expressive power, interpretation, and inferential capabilities of these networks as compared to traditional representations of concept hierarchies, in particular to KL-ONE, a standard representation language for conceptual hierarchies in the field of natural-language processing. Although the capabilities of current connectionist hierarchies fall short of traditional representations, three inference processes will be described which can be very easily and elegantly realized in a connectionist architecture whilst they are hard and cumbersome to implement in traditional knowledge representation systems.

*On leave from the University of Saarbrücken, West Germany. I would like to thank Joachim Diederich, Jerry Feldman, Mark Fanty, Susan Hollbach Weber, Christel Kemke, Mark Line, Lokendra Shastri, Robert Wilensky and Dekai Wu for comments on an earlier version of this paper or general discussions on the issues covered therein.

Contents

1 Introduction

A crucial prerequisite for the capability of an artificial intelligence system for processing natural language and reasoning about given facts is a knowledge representation (KR) scheme that includes conceptual knowledge, i.e. knowledge about the attributes of objects which are denoted by concepts. At least for avoiding redundancy, this knowledge is usually stored in the form of *concept hierarchies*, in which subconcepts inherit the information connected with their superconcepts if no local information exists.

In traditional artificial intelligence research, a great number of representation schemes have been developed for this purpose in the last two decades. These include in particular the so-called "frame" representations, semantic networks, predicate calculus, and (as an attempt to get the best out of all these whilst leaving out their deficiencies) the family of the KL-ONE languages, which have been investigated by Brachman (1978), Brachman & Schmolze (1985), Kaczmarek et al. (1986), v. Luck et al. (1987), Kobsa (1989) and several others. More recently, connectionist networks for the representation of conceptual hierarchies have been implemented as well (see e.g. Hinton 1981, Feldman & Ballard 1982, Cottrell 1985, Shastri & Feldman 1986, Derthick 1987a, b; Diederich 1988a, b, c; Shastri 1988a, b, forthcoming).[1] Unlike the case with traditional KR systems, processes in these networks operate in a parallel way and largely without a global control mechanism.

As will be shown, the expressive power of current connectionist systems for the representation of conceptual knowledge is weaker than that of representation systems developed in traditional artificial intelligence research. Nevertheless, it seems worthwhile to investigate the extent to which specific processes operating upon such conceptual representations can be more profitably realized in a parallel rather than a classical architecture. In this paper, three of these processes will be identified: classification in hierarchies which allow for the cancellation of attributes; (a rudimentary form of) Wilensky's (1983) concretion algorithm; and classification in concept hierarchies which include some sort of Rosch's (1973, 75) basic categories. Connectionist realizations of these processes on the basis of the Rochester Connectionist Simulator (Goddard et al. 1988) will be discussed.

2 Concept Representation in Traditional AI

In this section, a short survey of traditional concept representation systems will be given which is focused on the KL-ONE philosophy, a standard representation technique for conceptual knowledge in the field of natural-language processing. The SB-ONE system (Kobsa 1989) will be taken as a special representative of this representation language family, but any other member would do as well.

[1] Several more connectionist networks exist which are capable of representing conceptual knowledge, but cannot represent concept hierarchies. These will not be dealt with in this paper.

Most traditional concept representations comprise two levels of representation:

- the *general level*, which represents knowledge of a universal nature — in particular, knowledge about classes of objects; this part of a knowledge base is often addressed as the *terminological box* (T-Box) in KL-ONE jargon;
- the *individualized level*, which represents knowledge about individual facts and objects (the *individuals*) in the domain of discourse; this part of a knowledge base is often regarded as forming part of the so-called *assertional box* (A-box).

At both levels, the most important means of representation are *concepts* and *attribute descriptions*. General concepts correspond to unary predicates which can be applied to individuals of the situation to be represented. A relationship which normally exists between two individuals to which concepts gc_1 and gc_2 can be applied, respectively, is expressed through a general attribute description of one of these concepts. An attribute description of a concept gc_1 consists of a binary *role* predicate and the concept gc_2. This so-called *value-restriction concept* gc_2 restricts the possible values of the relation's second argument. The *number restriction* of an attribute description specifies how many of these attributes the individuals in the denotation of gc_1 possess in the minimal, maximal and default case. A *necessity marker* indicates in SB-ONE whether individuals to which gc_1 applies possess the respective attribute necessarily or only optionally.

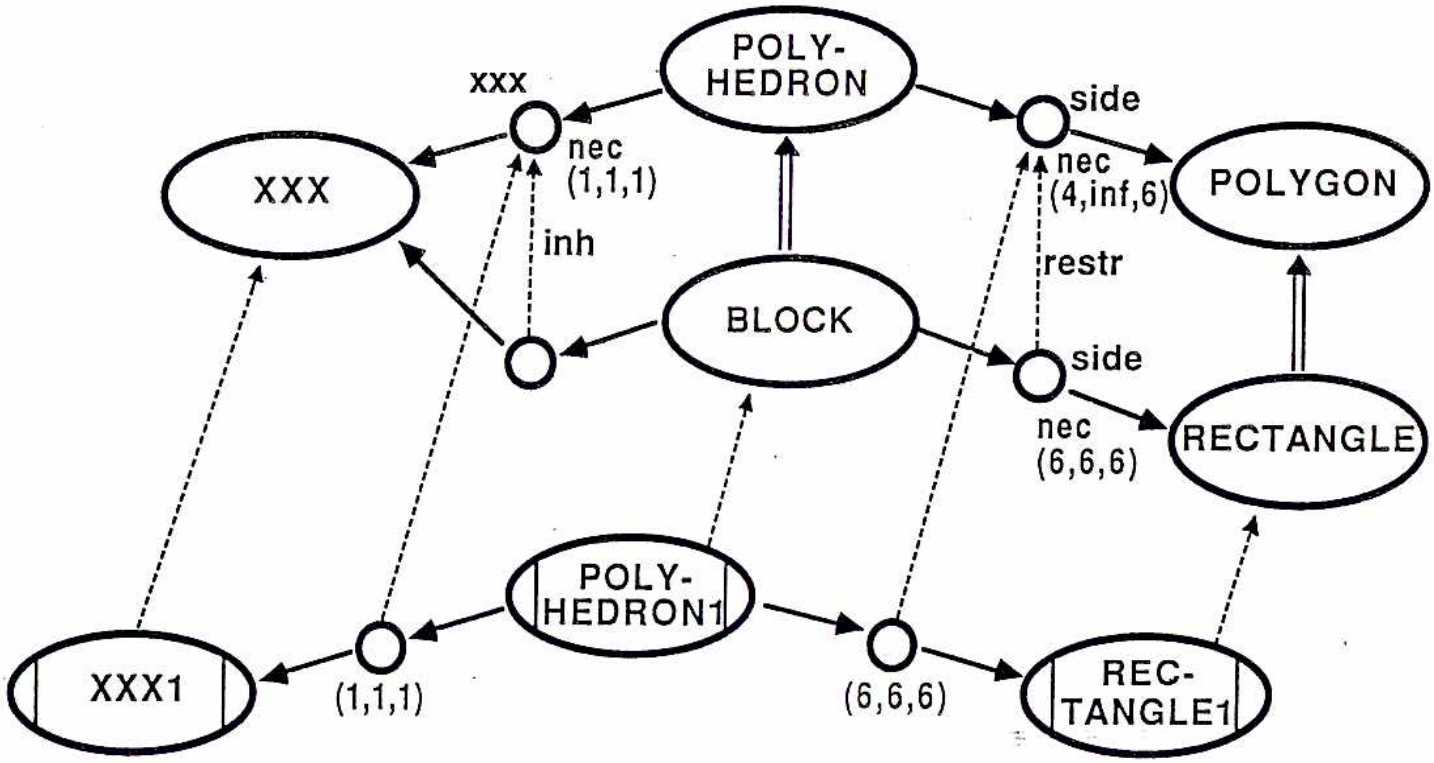

Figure 1: Example of a concept hierarchy in SB-ONE

Fig. 1 shows an example of an SB-ONE concept hierarchy in a graphical notation. The ovals and circles which do not possess lateral vertical chords represent general concepts and roles,

respectively. In the upper part of this level, the facts are represented that polyhedrons possess four to arbitrarily many (by default 6) sides, all of which are polygons, and exactly one xxx which is an XXX (as an arbitrary additional attribute). The lower part represents the fact that blocks possess exactly six sides all of which are rectangles, and exactly one xxx which is an XXX.

Individualized concepts in SB-ONE assert that, in the domain being represented, there exist individuals to which the corresponding predicates apply. In Fig. 1, the concepts 'POLYHEDRON', 'RECTANGLE' and 'XXX' as well as the attribute descriptions with role names 'side' and 'xxx' have been individualized (individualized concepts and roles carry lateral vertical chords). The whole structure expresses the fact that in the given situation there exist a polyhedron and six rectangles which form its sides, as well as an XXX which is its xxx.

Most of the recently proposed KL-ONE languages possess a precise mathematical interpretation (i.e. a so-called "semantics"). In the case of NIKL (Kaczmarek et al. 1986, Robins 1986), BACK (v. Luck et al. 1987), KRYPTON (Brachman et al. 1983) and related languages, it consists of a mapping from KR expressions into sets of individuals and two-placed relations between individuals. (In the case of SB-ONE, the mapping is into sets, relations, and an interpretation value out of {0,1}). Such an interpretation is given in order to make the meaning of representation structures as clear as possible, both for reasons of communication (everybody should interpret the same KR structure in the same way), and for a theoretical and practical evaluation of the behavior of processes operating upon the KR structure (i.e. for verifying whether processes operating upon a KR structure really always yield the desired result). Moreover, a precise interpretation makes it possible to clarify the capabilities and limits of a KR system, i.e. helps to determine what can and cannot be represented within the system.

Concepts can be ordered in a so-called *subsumption hierarchy*. A concept gc_1 subsumes a concept gc_2 if, semantically speaking, the extension of gc_2 is a subset of the extension of gc_1. On the syntactical level, this is parallelled by the *specialization* relation (denoted by the broad arrows in Fig. 1). A necessary condition for gc_2 to be a specialization of gc_1 is that each of gc_1's attribute descriptions is also possessed by gc_2, either in restricted form or unchanged ("inherited"). The SB-ONE relations 'restr' and 'inh', respectively, hold true in these cases. In languages of the KL-ONE family, concept hierarchies are always subsumption hierarchies, and attribute descriptions can only be inherited via this kind of hierarchy.

In Fig. 1, the concept 'BLOCK' is subsumed by the concept 'POLYHEDRON'. Specialization holds because the concept 'BLOCK' inherits from 'POLYHEDRON' the attribute description with role name 'xxx' without changes (cf. the 'inh' relation in Fig. 1), and the attribute description with role name 'side' in a restricted form (cf. the 'restr' relation; the number restriction interval and the value restriction concept become more specific). In KL-ONE-related languages, both 'POLYGON' and 'RECTANGLE' could also possess attribute descriptions of their own for which inheritance or restriction relations would have to apply.

The process of *classification* (see e.g. Schmolze & Lipkis 1983) determines for a concept gc_2 the most specific generalization, i.e. that concept gc_1 which subsumes gc_2 but does not subsume a

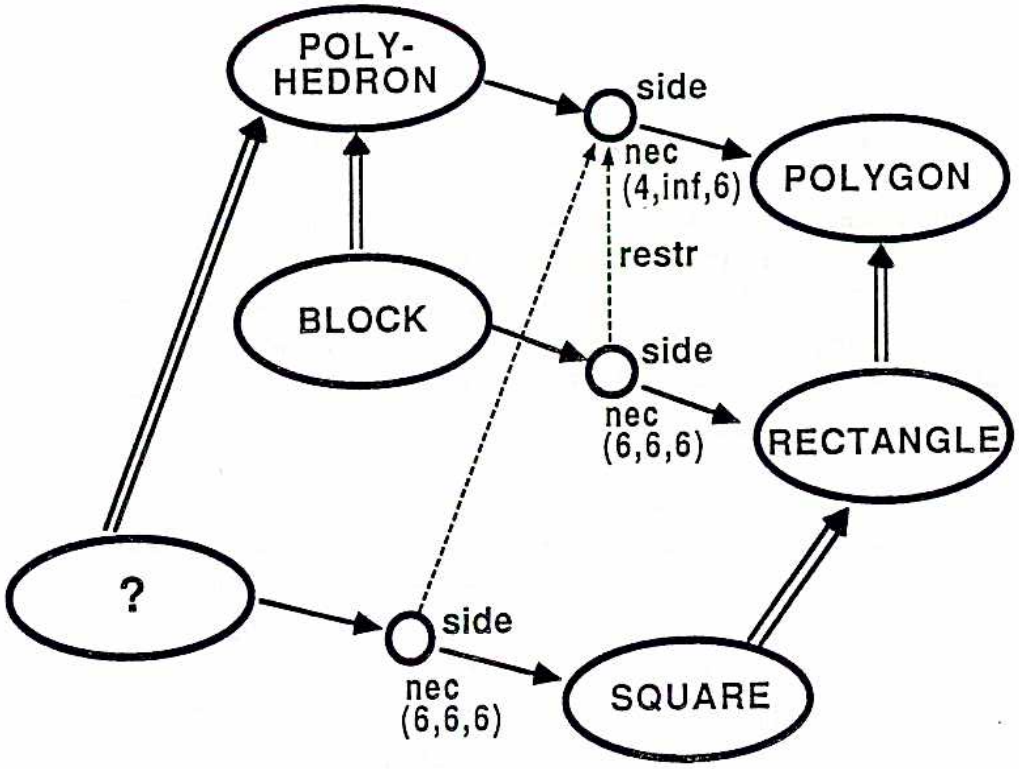

Figure 2: Classification in traditional concept hierarchies

concept gc_3 which itself subsumes gc_2 (there may also exist several such gc_1). The classification process operates exclusively on syntactical grounds (i.e. on the basis of the specialization relation), and should therefore be correct and complete with respect to subsumption. In the example of Fig. 2, the classifier would recognize that not only the concept 'POLYHEDRON', but also its specialization 'BLOCK' subsumes the concept marked with '?', since the attribute description of '?' is a restriction of that of 'BLOCK' because of the more specific value restriction.

The process of *realization* (Mark 1982) [or *recognition*, MacGregor 1988] does similar things for individualized concepts, namely to determine the most specific concept of which an individualized concept is an individualization. In Fig. 1, the realizer would recognize that the concept 'POLYHEDRON1' actually is not an individualization of 'POLYHEDRON', but more specifically of 'BLOCK'.

In addition to the representational elements mentioned, a great variety of other elements usually exist in classical conceptual representations. The SB-ONE language, for instance, also features inverse attribute descriptions, role value maps (for expressing identity relationships between role fillers), role differentiation (for forming partitions of the extensions of roles), as well as exhaustiveness and disjointness relations between subconcepts, among many others. Here we shall not deal with these advanced KR features, however, since they have hardly ever been realized in a connectionist architecture.

3 Connectionist Concept Hierarchies

In connectionist networks, knowledge about conceptual hierarchies is virtually always represented in a more or less "localized" way. This means that the conceptual information is not distributed over a whole network, but locally separable within the network. There exists a wide variety of connectionist conceptual representation systems which differ in expressive power and functionality. Nearly all of these distinguish between concepts and attribute descriptions, as is the case in traditional knowledge representation. However, no difference is often made between a general level and an individualized level, i.e. between generalized and individualized concepts and attribute descriptions. This reminds one of early frame representations such as FRL (Roberts & Goldstein 1977), where individualized frames were regarded as general frames with so much specific information associated with them that the cardinality of the extension was reduced to one.

Moreover, many connectionist representations do not distinguish between attribute descriptions and their value restrictions, neither on the general nor on the individualized level (if these levels are distinguished at all). In the cases of Figs. 1 and 2, this would mean, for instance, that the properties 'has-sides-which-are-polygons' and 'has-sides-which-are-rectangles' cannot be further decomposed, but are primitive. Therefore, then, these two properties have nothing to do with each other.

3.1 Example of a Connectionist Representation for Conceptual Hierarchies

For a first example of an implementation of a connectionist concept representation system, let us regard the network described in Diederich (1988a, b, c), which we will name 'DN' for short.[2] DN is partitioned into 4 spaces:

- the *concept space*, which contains the concepts
- the *attribute space*, which contains the attribute descriptions
- the *instance space*, which contains the *instances* (these correspond to the individualized concepts of SB-ONE)
- the *free space*, which is a repository for new concept sub-networks to be recruited in concept-learning processes (cf. Diederich 1988a, c)

For our purposes, it is sufficient to regard each concept, attribute description and instance as corresponding to exactly one node in the network. Actually, concepts in DN are 3-unit

[2]It should be pointed out that DN has primarily been developed for purposes of learning (Diederich 1988a, c), and less for the purposes of retrieval and classification with which we are mostly concerned here.

subnetworks as in Cottrell (1985), which allows one to represent the fact that some instance does not belong to some concept. Also, the sets of possible values of attributes actually form special competitive networks (so-called *winner-take-more* networks) which allow a value restriction to have more than one value. Since these features are not relevant for our purposes, we will ignore them for the rest of this paper. The free space will also be disregarded here.

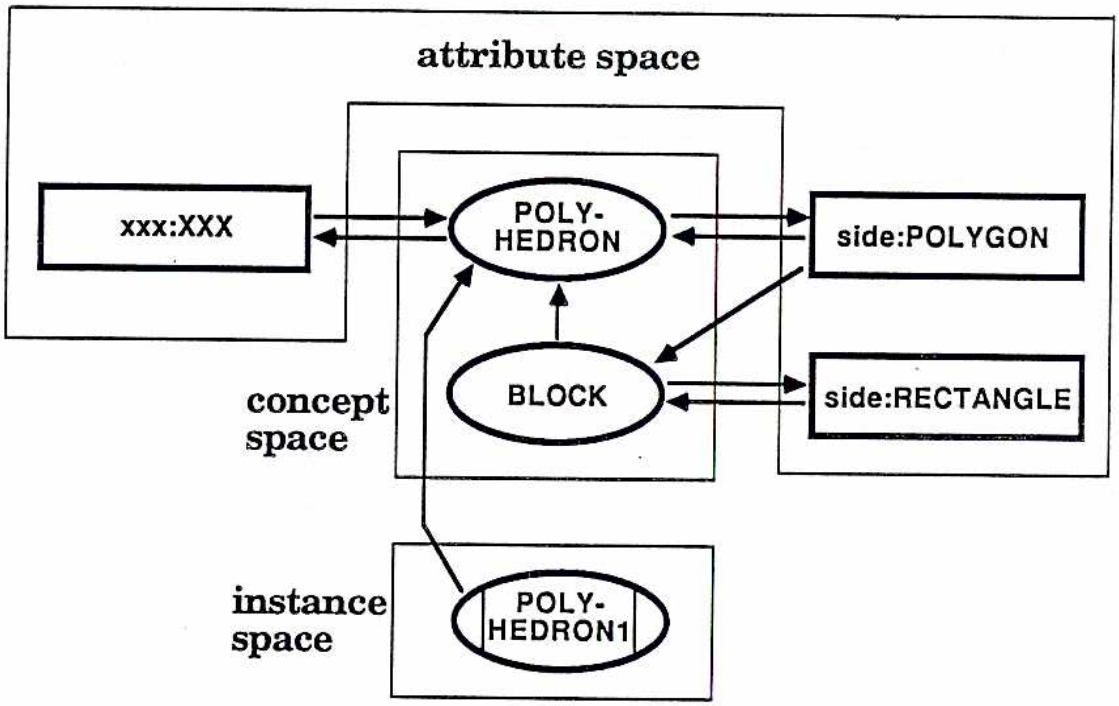

Figure 3: Fig. 1 in Diederich's architecture

Fig. 3 shows what elements of the knowledge base depicted in Fig. 1 can currently be represented in DN, and how this is performed:

- All concepts are linked to their attribute descriptions by excitatory links. The activation of a concept will thus activate all its attribute descriptions.
- Concepts may be ordered in a pre-defined hierarchy or heterarchy by establishing excitatory links from subconcepts to their direct superconcept(s). Activation of the subconcept will successively activate all direct and indirect superconcepts as well as their attribute descriptions. Hence, for determining the properties of a concept, it is not necessary to directly connect a concept with the attribute descriptions of its superconcepts, since they will be "inherited".
- All attribute descriptions are linked by excitatory links to the concepts which possess these attribute descriptions. Whenever an attribute description is linked both to a concept and its subconcepts, the weight of the latter links should be somewhat smaller than the former link. This guarantees that, when a set of attribute descriptions is clamped on, the topmost concept which possesses the most of these attribute descriptions will become the

most activated. It is this kind of inference which is called *classification* in the field of connectionism.

- The nodes for individualized concepts in the instance space are linked to concepts (whose attribute descriptions they inherit when clamped on) and possibly to additional attributes.

As can be seen, there exist a number of great differences between the KL-ONE-like knowledge base depicted in Fig. 1 and its connectionist reconstruction in DN. First of all, all attribute descriptions in DN are independent of each other. It is not possible to specify that one attribute description is a specialization of some other. Of course, since value restrictions are not distinguished, no hierarchy is possible for them either.

Also, instances in connectionist conceptual representations behave in principle like "normal" concepts. As is the case for concepts, the activation of instances is propagated to their attributes. In traditional knowledge representation systems, the "individualization" link between individualized concepts and general concepts has much more importance for practically all processes operating upon the knowledge base.

Consequently, connectionist "classification" in DN is also different from "classification" in traditional representations for concept hierarchies. Traditional classification takes attribute descriptions of value restriction concepts into account and determines subsumption relationships between concepts also on the basis of inferred subsumption relationships between these value restrictions, if necessary. And it often also draws complex deductions on the basis of information about number restrictions and disjointness (see e.g. Nebel 1988). Connectionist classification only determines the concept which possesses the most of the activated attribute descriptions (and need not do more since it is based on a simpler representation).

Moreover, no terminological distinction is usually made in the field of connectionism between the classification of concepts and the "classification" (i.e. "realization", in KL-ONE terms) of instances. Due to the fact that there is no distinction between general and individualized attribute descriptions both are performed in an identical way by clamping on the attributes which the instance of the concept possesses.

3.2 Other Connectionist Conceptual Hierarchies

3.2.1 Localist Representations

Shastri (1988a, b, forthcoming) has implemented a connectionist representation for conceptual hierarchies which overcomes some of the deficiencies mentioned (DN is currently also being enhanced with that goal in mind). He proposes the introduction of separate *property nodes* (which are identical in spirit to the "role nodes" in SB-ONE). A concept, a property and a property value are linked by two independent nodes, so-called *binder nodes*. These binder nodes

propagate activation only when they receive activation from both of their incoming links. In Fig. 4, the binder node b1 will propagate activation if both 'side' and POLYHEDRON are activated. This can be regarded as a determination of the value of property 'side' of concept 'POLYHEDRON'. When 'side' and 'POLYGON' are clamped on, binder node b2 will become activated and will itself activate the node 'POLYHEDRON'. If several properties and property values are activated, a kind of connectionist classification as described above will take place.

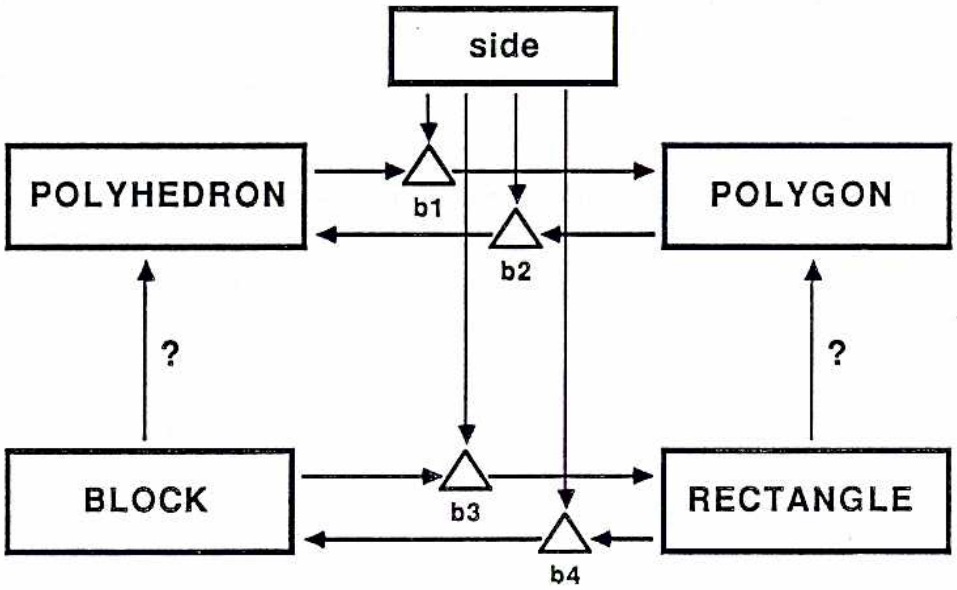

Figure 4: Detail of the general level of Fig. 2 in Shastri's architecture

Hollbach (1988a, b) extends Shastri's representation in that concepts can also be linked to *sets* of possible values. These values can be declared to be mutually disjoint.

Although Shastri also introduced "specialization links" between subordinated and superordinated concepts, and similar links between property values, these cannot be introduced between any two concepts and property values. For an example see Fig. 4: If one introduces the specialization links marked by '?', an activation of 'BLOCK' and 'side' will also activate 'POLYHEDRON' and yield two values for the 'side' property of 'BLOCK' (via binder nodes b1 and b3, respectively). Also, an activation of 'RECTANGLE' and 'side' will activate 'POLYGON' and thus also yield 'POLYHEDRON' as a second, less specific classification result. Various mechanisms, however, can be introduced to overcome these problems (e.g. by taking the delay of activation into account).

3.2.2 Semi-Localist Representations

Hinton (1981) presents a representation for conceptual hierarchies which is "local at a global scale but global [= distributed, A. K.] at a local scale" (Hinton et al. 1986, p. 79). Essentially, a triple of binary patterns is introduced for the representation of a concept-property-value relationship (cf. Fig. 5). This binary triple is used for teaching a network of perceptron-like units (Rosenblatt

1962). When two members of the triple are clamped on, the missing pattern (i.e. the concept, the property or the value pattern) will be completed.

verbal description			*representation pattern*
POLYHEDRON	side	POLYGON	1000 1 0100
BLOCK	side	RECTANGLE	1100 1 0110
?	side	SQUARE	1010 1 0111

Figure 5: Possible representation of Fig. 2 in Hinton's architecture

The binary patterns for subconcepts in Hinton's system possess '1's in all positions where they occur in the pattern of the superconcept, plus additional '1's which distinguish them from the patterns of all other concepts. The special input combination function used by the system guarantees that incomplete triples which contain the subconcept pattern are completed by using specific information concerning the subconcept, if such information was previously entered into the system, or otherwise by using information about the direct or indirect superconcepts. This means that attribute descriptions can be inherited, but also that inherited attribute descriptions can be overwritten by local exceptions. The same holds true for the relationship between concepts and their instances.[3] Fig. 5 shows a possible representation of the general level of Fig. 2 in Hinton's architecture.

Hinton uses an *implicit* encoding of subsumption relationships: the representation pattern of a subconcept *includes* the representation pattern of a superconcept. This kind of representation can be rather expensive: If one allows concepts to have more than one superconcept, the binary vector for concept patterns must be at least of length n, where n is the number or concepts to be represented. If only strict hierarchies are to be represented, the minimal pattern length is $\lceil \log_2(n+1) \rceil$, where '$\lceil\ \rceil$' is the integer ceiling function. It increases considerably for narrow and for balanced hierarchies, and may even also be equal to n in the extreme case where the hierarchy is just a linear path. Moreover, one should bear in mind that in both cases the distinguishability of concepts need not mean good retrieval. It is probably no coincidence that in the examples of Hinton (1981) the Hamming distance between patterns is always much greater than 1.

It seems possible, in principle, to introduce value restrictions instead of values at the general level, and to regard them as concepts between which subsumption relationships can be defined. This, of course, increases even more the minimal length of the concept pattern vectors. Classification is not possible: In Hinton's system, incomplete triples containing the property and the value will be completed by the supermost concepts which possess this attribute description. Hence, if n attribute descriptions are clamped on one after the other, possibly as many as n different concept patterns will subsequently result, which need not even be direct or indirect superconcepts. An additional mechanism would be necessary to find the most general subconcept of them.

[3] Actually, Hinton (1981) only discusses this latter relationship explicitly. His architecture, however, can also be used for the representation of subsumption relationships, which the author seems to have had in mind anyway.

Derthick (1987a, b) also uses a sort of "semi-localist" approach to conceptual representation: Features of an instance and of the concept to which the instance belongs are represented as binary vectors. Connections between features are represented in 5 modules (Hopfield networks) which are interrelated: The *subject module* represents an instance; the *subject-type* module represents the concepts to which the instance belongs; the *role-fillers module* represents the set of roles in which the features of the instance are fillers; the *role-filler-type-restrictions module* represents the set of value restrictions imposed on role fillers by the concept represented in the subject-type module; and the *role-filler-types* module represents the type of each instance in the role-fillers module. Subsumption relationships are only indirectly expressed, namely by subset relationships of the concept features. At the moment, the network is only used for retrieval purposes and for specific kinds of probability inferences.

4 More Differences between Traditional and Connectionist Concept Hierarchies

4.1 Spreading Activation vs. Connectionist Activation Propagation

In connectionist representations for conceptual hierarchies, search and inference processes are realized by an activation flow in connectionist networks. The activation of a node is propagated to those nodes with which the node is connected. In traditional architectures, so-called *spreading-activation* processes are sometimes employed (cf. Quillian 1968, Collins & Loftus 1975, Fahlmann 1979, 81; Charniak 1986), which at first sight seem to be very similar to activation propagation in the connectionist style. Although all sorts of intermediate systems can be defined, there are generally a number of differences between connectionist networks and traditional networks which employ spreading activation techniques (for others see Diederich, forthcoming):

- In connectionist networks, the activation flow is *"semantically uncontrolled"*: there exists only one type of link (e.g. a concept's link to its superconcept cannot be distinguished from a link to one of its attribute descriptions), and only the link weights can influence the propagation of activation. In traditional concept hierarchies, activation can often be selectively propagated on specific links only, depending on the task to be solved.[4]
- In connectionist networks, the propagation of activation stabilizes after a certain period, whereas spreading activation in traditional networks is generally aborted after a given time, usually by an external controller.
- The result of spreading activation is usually evaluated by an external *path evaluator*, which compares and analyzes paths of highly activated nodes and selects "interesting" and "meaningful" paths. A corresponding mechanism does not exist for connectionist networks. In

[4] In some connectionist systems (e.g. that of Shastri 1988a, b, forthcoming), however, links can be disabled when the network is to perform certain tasks. This seems to correspond to selective spreading activation.

some sense, the functionality of the path evaluator is encoded in the structure of these networks.

4.2 Epistemological Status of Attribute Descriptions

In many traditional knowledge representation schemes for conceptual hierarchies, attribute descriptions are regarded as necessary and sufficient conditions for a concept (this is particularly true for the languages of the KL-ONE family). This means (a) that all objects to which a concept applies possess all attributes described by the attribute descriptions of the concept, and (b) that a concept applies to an object only if the object possesses all of the attributes described by the concept's attribute descriptions. The connectionist representations presented seem to adhere only to the former conception: when concept nodes (and in Shastri's case their role nodes) are clamped on, all their attribute values become activated. Attribute descriptions, however, are not regarded as sufficient conditions since, when a set of attribute descriptions is clamped on, the topmost concept which possesses the most of these attributes will become the most activated. It is not necessary that this concept possesses *all* of these attributes, as must be the case in traditional knowledge representation.

4.3 Interpretation of Knowledge Representation Structures

Traditional conceptual representation languages usually possess a well-defined interpretation, mostly a mapping of language expressions into a domain (a structure consisting of a set of individuals and two-placed relations between individuals) or into a set of truth values. A set-theoretic or truth-functional justification can then be given for the results of the classification and realization processes. On the basis of such an interpretation, the computational complexity, decidability and completeness of these inference processes can also be determined (see e.g. Brachman & Levesque 1984, Nebel 1988, Patel-Schneider 1988).

No such interpretation can as yet be given to connectionist conceptual representations. Explanations for elements and processes of the implementation (such as that some node is supposed to stand for the concept 'elephant' or the set of elephants, or that a certain type of activation pattern is to determine whether or not some instance belongs to the set of elephants) can only be made verbally. First attempts are currently being made to reconstruct the behavior of connectionist models by formal means (see e.g. Cooper 1988, Shastri 1988b). No formal specification of the relationship between such a model and its reconstruction has been given, however. There also exist general attempts to relate connectionist models to symbolic models (see e.g. Smolensky 1988 and the open peer commentary). At least as far as the structure of conceptual hierarchies is concerned, however, there still seems to be a long way to go towards some sort of connectionist semantics comparable to the usual kinds of semantics of traditional knowledge representation schemes, or to some kind of "procedural semantics".

4.4 Discussion

The expressive power of current connectionist systems for the representation of conceptual knowledge is considerably weaker than that of representation systems developed in traditional artificial intelligence research: usually no difference is made between a general knowledge representation level and an individualized level (and hence no conceptual distinction is necessary between classification and realization); many connectionist representations do not distinguish between attributes and their values or value restriction concepts (hence the latter cannot form a concept hierarchy and cannot possess attributes of their own); and many representational elements of traditional representations (e.g. inverse attribute descriptions, role value maps, role differentiation, exhaustiveness or disjointness of subconcepts, etc.) have so far hardly been realized in a connectionist architecture.

Nevertheless, it seems worthwhile to investigate the extent to which specific processes which operate upon such conceptual representations can be more profitably realized in a parallel rather than a classical architecture. In the following sections, three processes will be identified which can be easily and elegantly integrated into a connectionist representation of concept hierarchies while they are hard and cumbersome to integrate into a traditional architecture. These processes are classification in hierarchies which allow for the cancellation of attributes, (a rudimentary form of) Wilensky's (1983) concretion algorithm, and classification in concept hierarchies which include some sort of Rosch's (1973, 75) basic categories. Connectionist realizations of these processes on the basis of the Rochester Connectionist Simulator (Goddard et al. 1988) will be discussed.

5 Cancellation of Attributes

Sometimes, when defining a concept hierarchy, one would like to see a concept B to be a subconcept of some concept A, apart from a specific attribute description of A which B should not possess. Another case is that the value restriction of an attribute description which B inherited from A should not be a subconcept of A's value restriction of this attribute description. For instance, it sometimes makes sense to regard a whale as being a fish, apart from the fact that it has lungs for breathing and not gills.

Several problems arise, however, if one introduces this kind of cancellation into a traditional concept hierarchy (cf. Brachman 1985). Remedies can be found for all of them, mostly by imposing additional restrictions on the use of cancellation (some of which are rather arbitrary, however). For instance, the maximum number of cancelled attribute descriptions of a concept can be limited (e.g. to one or two), the cancellation of already cancelled attribute descriptions can be prohibited, or the cancellation can be restricted to optional attribute descriptions only.

At least the following problem remains, however: When operating in a top-down fashion, a traditional classifier will prune away all branches in which attribute descriptions exist which

the new concept to be classified does not share. This is an incorrect procedure, however, if cancellation of attributes is allowed, since the problematic attribute might have been cancelled further down in the hierarchy. Thus, the traditional classifier ought to search the whole hierarchy. An alternative is provided by a connectionist architecture in which attributes are connected to all concepts which possess these attributes (as is the case with the network of Diederich 1988a, b, c). In this architecture, the appropriate concept will be found by the classifier even if attributes have been cancelled for a superordinate concept.

There obviously exists a trade-off between exhaustive search in concept hierarchies and full connectivity between attribute descriptions and concepts. Traditional classifiers avoid the former, cannot rely on the latter, and thus have problems with cancelled attribute descriptions. Connectionist networks possess the latter in any case and thus get classification in hierarchies with cancellation for free.

6 Concretion

Wilensky (1983), Wilensky et al. (1986), Norvig (1987), Wu (1987), Jacobs (1988) and others investigated the so-called *concretion mechanism*. Concretion is intuitively a very natural form of reasoning. The idea behind it is that a description of an instance may be positioned lower in the hierarchy than is justified by mere classification if there exists evidence gained from outside the system (e.g. through contextual information) that a specific lower concept could have been addressed by the description. That is, a concept can be positioned at a certain level in the hierarchy even if specific necessary attributes are missing. Default values are supplied for these attributes. Concretion thus is "a plausible inference, not a logical consequence of the taxonomy, and thus is beyond the scope of KL-ONE classification" (Norvig 1987, p. 114). Contextual information leads to a sort of priming which influences inference processes.

It is relatively easy to implement a rudimentary form of this kind of reasoning in a connectionist architecture if one assumes that contextual expectations will result in a small residual potential of the expected concept. This potential might be a residue of a preceding direct activation of the concept itself (e.g. if the concept has been addressed in an ongoing dialog), or a consequence of the activation of a context node with which the concept node is connected (Waltz & Pollack 1985, Hollbach 1988b).

For an example, assume that in Fig. 6 there exists such a residual potential for the concept 'ROYAL ELEPHANT'. When the instance marked with '?' is clamped on (which activates the 'leg:CYLINDER' and 'trunk:CYLINDER' attributes), "normal" connectionist classification would yield the strongest activation for the concept 'ELEPHANT' (see Section 3.1), and smaller activation for the concepts 'ROYAL ELEPHANT' and 'ANIMAL'. If the residual potential of 'ROYAL ELEPHANT' is strong enough to make up the difference between the classification-based activation of 'ROYAL ELEPHANT' and 'ELEPHANT', concretion will yield 'ROYAL ELEPHANT' as a result.

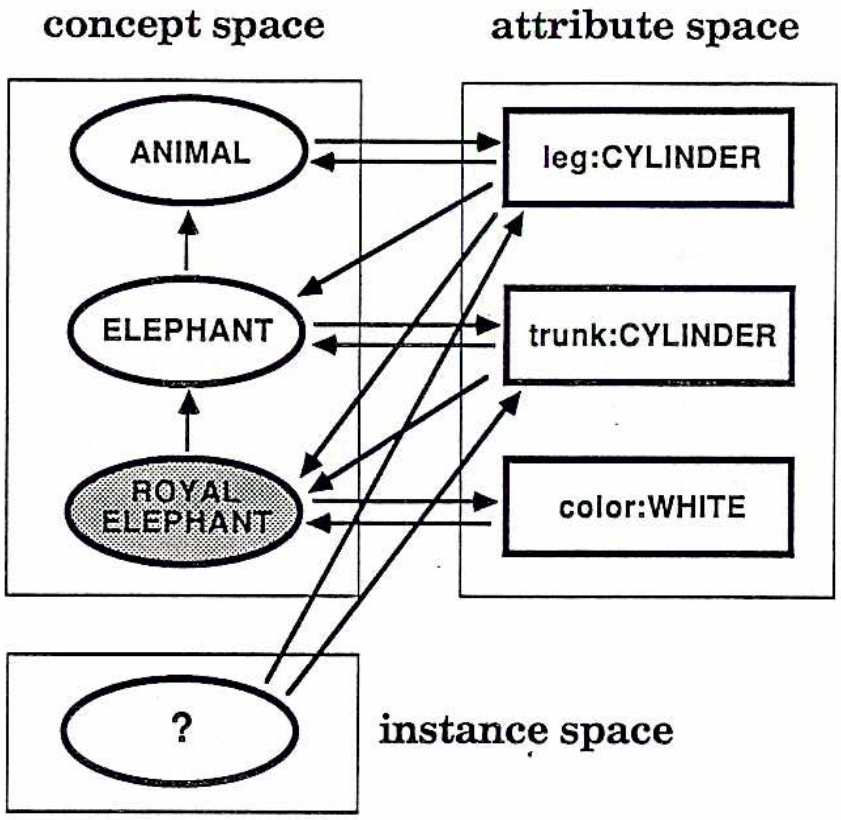

Figure 6: Classification enhanced by concretion

If the instance only had 'leg:CYLINDER' as an attribute, the classification-based activation of 'ROYAL ELEPHANT' would be smaller than in the first case, and the context-based activation of 'ROYAL ELEPHANT' would have to be stronger to overcome this difference. Thus there exists a kind of natural trade-off between the strength of expectation and the degree of match between the attributes of the instance and the expected concept.

The model is not restricted to the assumption that only one concept possesses a residual contextual potential: Assume that in Fig. 6 there exist several activated subconcepts of the concept 'ELEPHANT' (possibly at different levels in the hierarchy). The result of the concretion process will be that concept (or those concepts) for which the sum of residual potential and activation received from normal connectionist classification is greatest. Thus the model goes beyond the *least commitment principle* (Wilensky 1983, Wu 1987) since the result of concretion is not necessarily the supermost of the concepts which possess a residual potential (i.e. that activated concept which has the fewest attributes in addition to those determined for the new instance). Instead, the model also takes the possibly different residual potentials of these concepts into account.

The model may however fail to meet the *exhaustion principle* (Wilensky 1983, Wu 1987) which requires that the selected concept must (at least) possess all attributes of the instance. To see why this is the case, assume that the hierarchy in Fig. 6 is extended by an additional subconcept of 'ANIMAL', namely 'HORSE' (cf. Fig. 7). This concept thus possesses the attribute

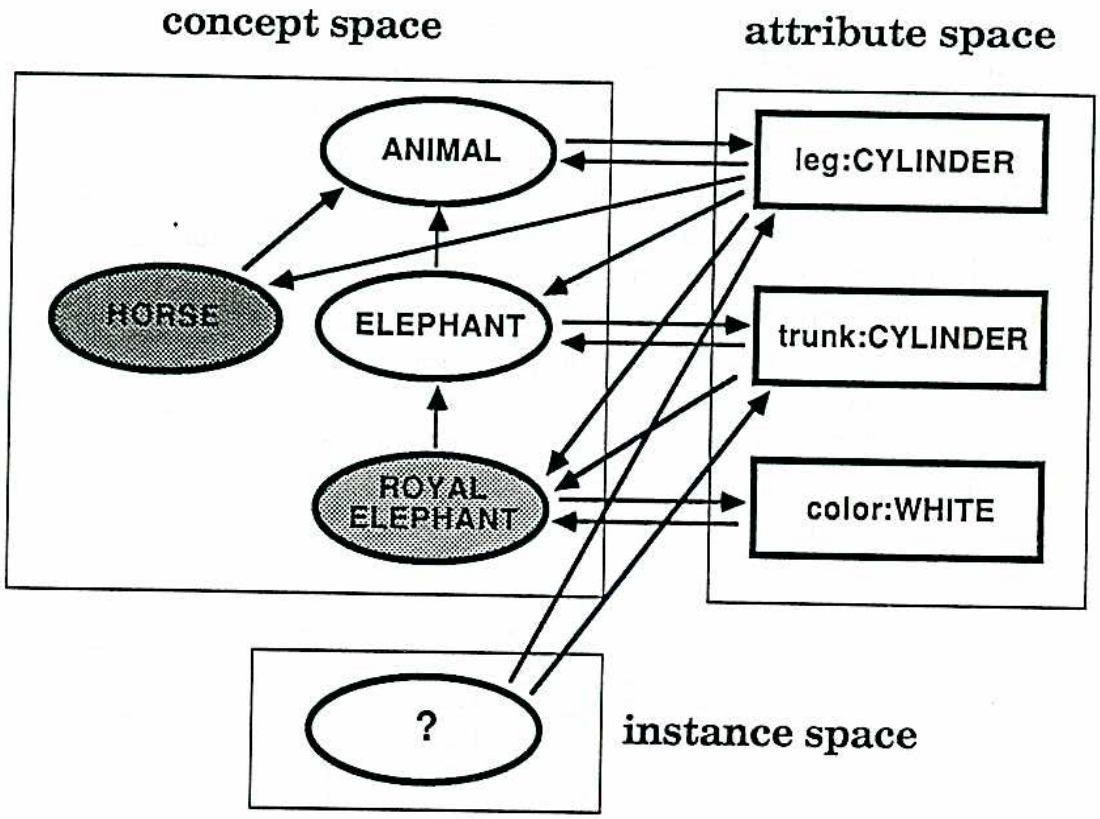

Figure 7: Misclassification due to contextual priming

'leg:CYLINDER', but not 'trunk:CYLINDER'. Normal connectionist classification would recognize an instance which possesses exactly these two attributes as being an ELEPHANT. This concept receives activation from both attributes, whereas the concept 'HORSE' is activated only by the former. Nevertheless, if the concept 'HORSE' possesses a very high residual potential, it might be selected by the connectionist concretion process even though it does not possess the attribute 'trunk:CYLINDER'. This "defect" is not surprising since connectionist inferences are not logical inferences, but constitute a sort of fuzzy reasoning in which plausible assumptions are as important as observed facts.

7 Basic Categories

Rosch (1973, 75, 78) has shown that there exist *basic level concepts* (or preferred concepts) in human classification: In a *forced naming task* (cf. Rosch et al. 1976), subjects who are shown the picture of a particular collie tend to respond that it is a *dog*, and not that it is a collie, a mammal, or an animal. In a *target recognition task*, subjects confirm that a pictured collie is a *dog* more quickly than they confirm that it is a collie, a mammal or an animal. These two results indicate that the concept 'dog' forms a basic level concept in this hierarchy.

A number of explanations have been given as to why certain concepts are preferred to others. Rosch & Mervis (1975) propose that basic level concepts appear to be the most general concepts for which the extensions still possess many common attributes. Gluck & Corter (1985) also take the (relative) cardinality of the extensions of concepts into account and introduce the notion of *category utility*: For basic concepts, the product of the relative probability of an individual which is in the extension of the concept and the expected number of correct predictions given such an instance is maximized. Other work (e.g. Tversky & Hemenway 1984, Hoffmann & Ziessler 1983, Corter et al. 1988) suggests that qualitative rather than quantitative differences in the attribute descriptions of concepts might be a reason why these concepts are basic (i.e. specific attribute descriptions - such as 'part' - play a more important role than others). Work by Barsalou (1987) on the context-dependency of typicality (i.e. which concepts are regarded as "typical" subconcepts of a superordinate concept) might even suggest a context-dependency of basic level concepts.

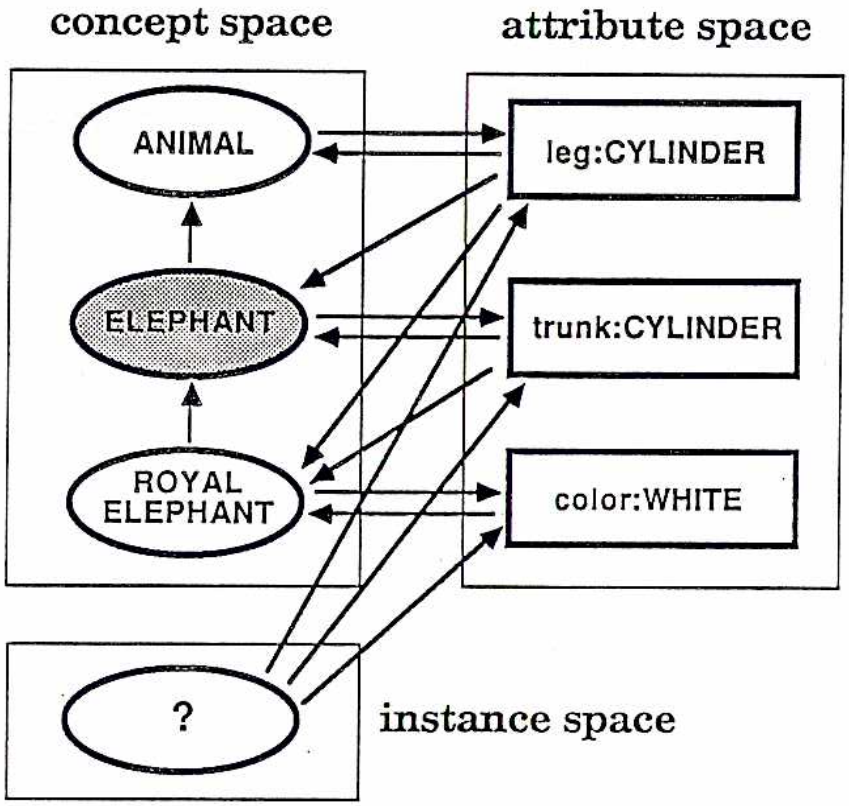

Figure 8: Classification in hierarchies with basic categories

We cannot deal here with the question of why certain concepts are preferred to others. We will simply presuppose that certain concepts possess some basic level potential (which might possibly be induced by the current context). Assume that in Fig. 8 the concept 'ELEPHANT' possesses such a basic potential. Note that, unlike in Fig. 6, the instance marked by '?' now possesses all three attribute descriptions. When this instance is clamped on, the concept 'ROYAL ELEPHANT' receives the highest activation from connectionist classification, and 'ELEPHANT' somewhat less. If the basic potential of the concept 'ELEPHANT' is strong enough, however, it

will override the difference in activation from classification, and yield 'ELEPHANT' as the most highly activated concept.

An interesting variant of this connectionist solution is to base the response to a categorization task not on the determination of the most strongly activated concept, but on a determination of that concept whose potential is the first to exceed a certain threshold value. Given appropriate weights of the links between attribute descriptions and concepts, as well as appropriate concept activation functions for determining the increase of concept potential on the basis of its current potential and the activation received from its activated attribute descriptions, basic level concepts will be the first to exceed this threshold potential (although they receive less activation from their attribute descriptions than more specific concepts, they have a better start due to their initial activation). Thus one can not only account for the preferences in the above-mentioned forced naming task, but also for the quicker performance in the target recognition task.

8 Summary

Although current connectionist realizations of conceptual hierarchies possess considerably less expressive power than their counterparts in the field of traditional knowledge representation, there exist a number of processes which can be very easily and elegantly realized in a connectionist architecture whilst they are hard or cumbersome to implement on the basis of traditional knowledge representation systems. Examples were given for classification in concept hierarchies which allow for the cancellation of attribute descriptions, classification enhanced by concretion, and classification in hierarchies which contain basic concepts. All of these processes constitute some kind of *fuzzy reasoning*, and it seems intuitive that connectionist architectures can carry out these tasks more easily than traditional systems.

All connectionist concept hierarchies with a minimal amount of expressive power currently possess a more or less local representation of knowledge (for a semi-localist one, see the models of Hinton 1981 and Derthick 1987a, b, which were discussed above). One can expect that if the expressive power is to be increased, even more of the uniformity of a network must be given up. For instance, when one allows in Shastri's representation (see Fig. 4) that POLYGONs may also have sides, then an activation of the concept 'POLYHEDRON' and the property 'side' results in an activation of the concept 'POLYGON', which in return (since 'side' is still on) results in an activation of the value of 'side' for 'POLYGON'. Special mechanisms (e.g. task-dependent deactivation of binder nodes) must be introduced to prevent such undesired effects (and, indeed, Shastri 1988a, b already utilizes such mechanisms in his system). Enhanced connectionist architectures for conceptual hierarchies will probably resemble traditional representations even more than they do today. The idea of hybrid systems which make the best out of both approaches no longer seems so futuristic, therefore.

9 References

Barsalou, L. W. (1987): The Instability of Graded Structure. In: U. Neisser, ed.: Concepts and Conceptual Development. Cambridge: Cambridge Univ. Press.

Brachman, R. J. (1978): A Structural Paradigm for Representing Knowledge. Report No. 3605, Bolt, Beranek & Newman, Cambridge, MA.

Brachman, R. J. and H. J. Levesque (1984): The Tractability of Subsumption in Frame-Based Description Languages. Proc. AAAI-84, Austin, TX, 34-37.

Brachman, R. J. (1985): I Lied About the Trees. AI Magazine, Summer 1985.

Brachman, R. J., R. E. Fikes and H. J. Levesque (1983): KRYPTON: Integrating Terminology and Assertion. Proc. AAAI-83, 31-35.

Brachman, R. J. and J. G. Schmolze (1985): An Overview of the KL-ONE Knowledge Representation System. Cognitive Science 9, 171-216.

Charniak, E. (1986): A Neat Theory of Marker Passing. Proceedings of the AAAI-86, Philadelphia, PA, 584-588.

Collins, A. M. and E. F. Loftus (1975): A Spreading-Activation Theory of Semantic Memory. Psychological Review 82, 407-428.

Cooper, P. (1988): Structure Recognition by Connectionist Relaxation: Formal Analysis. Proc. of the 7th Canadian Conference on Artificial Intelligence, Edmonton, Canada, 148-155.

Corter, J. E., M. A. Gluck and G. H. Bower (1988): Basic Levels in Hierarchically Structured Categories. Proc. of the 10th Annual Conference of the Cognitive Science Society, Montreal, Canada, 118-124.

Cottrell, G. W. (1985): Parallelism in Inheritance Hierarchies with Exceptions. Proc. IJCAI-85, Los Angeles, CA, 194-202.

Derthick, M. (1987a): A Connectionist Architecture for Representing and Reasoning about Structured Knowledge. Proc. of the 9th Annual Conference of the Cognitive Science Society, Seattle, WA, 131-142.

Derthick, M. (1987b): Counterfactual Reasoning with Direct Models. Proc. AAAI-87, 346-351.

Diederich, J. (1988a): Connectionist Recruitment Learning. Proc. of the 8th European Conference on Artificial Intelligence, Munich, West Germany, 351-356.

Diederich, J. (1988b): Knowledge Representation in a Structured Connectionist System. Unpub-

lished Draft, International Computer Science Institute, Berkeley, CA.

Diederich, J. (1988c): Knowledge-Intensive Recruitment Learning. TR-88-010, International Computer Science Institute, Berkeley, CA.

Diederich, J. (forthcoming) Spreading Activation and Marker-Propagation in Natural Language Processing. To appear in 'Theoretical Linguistics'.

Fahlman, S. E. (1979): NETL: A System for Representing and Using Real-World Knowledge. Cambridge, MA: MIT Press.

Fahlman, S. E. (1981): Representing Implicit Knowledge. In: G. E. Hinton and J. A. Anderson, eds.: Parallel Models of Associative Memory. Hillsdale, NJ: Lawrence Erlbaum.

Feldman, J. A. and D. H. Ballard (1982): Connectionist Models and Their Properties. Cognitive Science 6, 205-254.

Gluck, M. and J. Corter (1985): Information, Uncertainty, and the Utility of Categories. Proc. of the Seventh Annual Conference of the Cognitive Science Society, Irvine, CA, 283-287.

Goddard, N. H., K. J. Lynne and T. Mintz (1988): The Rochester Connectionist Simulator. TR 233, Computer Science Dept., Univ. of Rochester, Rochester, NY.

Hinton, G. E. (1981): Implementing Semantic Networks in Parallel Hardware. In: G. E. Hinton and J. A. Anderson: Parallel Models of Associative Memory. Hillsdale, NJ: Lawrence Erlbaum.

Hinton, G. E., J. L. McClelland and D. E. Rumelhart (1986): Distributed Representations. In: D. E. Rumelhart, J. L. McClelland and the PDP Research Group: Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol. I: Foundations. Cambridge, MA: MIT Press.

Hoffmann, J. and C. Ziessler (1983): Evaluating an Adaptive Network of Human Learning. Journal of Memory and Language 27, 166-195.

Hollbach, S. C. (1988a): Conceptual Representation in Connectionist Models. Technical Report, Computer Science Dept., Univ. of Rochester, Rochester, NY.

Hollbach, S. C. (1988b): Direct Inferences in a Connectionist Knowledge Structure. Proc. of the 10th Annual Conference of the Cognitive Science Society, Montreal, Canada, 608-614.

Jacobs, P. S. (1988): Concretion: Assumption-Based Understanding. In: Proc. COLING-88, Budapest, Hungary, 270-274.

Kaczmarek, T., R. Bates and G. Robins (1986): Recent Developments in NIKL. Proc. AAAI-86, 978-985.

Kobsa, A. (1989a): The SB-ONE Knowledge Representation Workbench: Extended Abstract. In: Preprints of the Workshop on Formal Aspects of Semantic Networks, Santa Catalina Island, CA, Feb. 1989.

Kobsa, A. (1989b): The SB-ONE Knowledge Representation Workbench. Technical Report, SFB 314: AI - Knowledge-Based Systems, Dept. of Computer Science, Univ. of Saarbrücken, W. Germany (forthcoming).

MacGregor, R. M. (1988): A Deductive Pattern Matcher. Proc. AAAI-88, St. Paul, MN, 403-408.

Mark, W. (1982): Realization. Proceedings of the 1981 KL-ONE Workshop. Report No. 4842, Bolt Beranek and Newman, Cambridge, MA, 78-89.

Nebel, B. (1988): Computational Complexity of Terminological Reasoning in BACK. Artificial Intelligence 34, 371-383.

Norvig, P. (1987): A Unified Theory of Inference for Text Understanding. Ph.D. Thesis and Report No. UCB/CSD 87/339, Computer Science Division, University of California at Berkeley, Berkeley, CA.

Patel-Schneider, P. (1988): Undecidability of Subsumption in NIKL. TR 75, Schlumberger Palo Alto Research, Palo Alto, CA.

Quillian, R. (1968): Semantic Memory. In: M. Minsky, ed.: Semantic Information Processing. Cambridge, MA: MIT Press.

Roberts, R. B. and I. P. Goldstein (1977): The FRL Primer. Memo No. 408, AI Lab, MIT, Cambridge, MA.

Robins, G. (1986): The NIKL Manual. Technical Report, Information Science Institute, Marina del Rey, CA.

Rosch (Heider), E. (1973): Natural Categories. Cognitive Psychology 4, 328-350.

Rosch, E. (1975): Cognitive Representations of Semantic Categories. Journal of Experimental Psychology (General) 104, 192-233.

Rosch, E. and C. Mervis (1975): Family Resemblances: Studies in the Internal Structure of Categories. Cognitive Psychology 7, 573-603.

Rosch, E., C. Mervis, W. Gray, D. Johnson and P. Boyer-Braem (1976): Basic Objects in Natural Categories. Cognitive Psychology 8, 382-439.

Rosch, E. (1978): Principles of Categorization. In: E. Rosch and B. Lloyd, eds.: Cognition and Categorization. Hillsdale, NJ: Lawrence Erlbaum.

Rosenblatt, F. (1962): Principles of Neurodynamics. New York: Spartan.

Schmolze, J. G. and T. A. Lipkis (1983): Classification in the KL-ONE Knowledge Representation System. Proc. IJCAI-83, 330-332.

Shastri, L. and J. A. Feldman (1986): Neural Nets, Routines and Semantic Networks. In: N. E. Sharkey, ed.: Advances in Cognitive Science 1. Chichester: Ellis Horwood.

Shastri, L. (1988a): A Connectionist Approach to Knowledge Representation and Limited Inference. To appear in 'Cognitive Science' 12(3).

Shastri, L. (1988b): Semantic Networks: An Evidential Formalization and its Connectionist Realization. London: Pitman.

Shastri, L. (forthcoming): Default Reasoning in Semantic Networks: an Evidential Formalization. To appear in 'Artificial Intelligence'.

Smolensky, P. (1988): On the Proper Treatment of Connectionism. Behavioral and Brain Sciences 11, 1-23.

Tversky, B. and K. Hemenway (1984): Objects, Parts and Categories. Journal of Experimental Psychology (General) 113, 169-193.

Waltz, D. L. and J. B. Pollack (1985): Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation. Cognitive Science 9, 51-74.

v. Luck, K., B. Nebel, C. Peltason, A. Schmiedel (1987): The Anatomy of the BACK System. KIT-Report 41, Dept. of Computer Science, Technical University of Berlin, Berlin, W. Germany.

Wilensky, R. (1983): Planning and Understanding. Reading, MA: Addison-Wesley.

Wilensky, R., J. Mayfield, A. Albert, D. Chin, C. Cox, M. Luria, J. Martin and D. Wu (1986): UC - A Progress Report. Report UCB/CSD 87/303, Computer Science Division, University of California at Berkeley, Berkeley, CA.

Wu, D. (1987): Concretion Inferences in Natural-Language Understanding. In: K. Morik, ed.: GWAI-87: 11th German Workshop on Artificial Intelligence. Berlin: Springer.