ELSEVIER

Artificial Intelligence 75 (1995) 241-295

Artificial Intelligence

# Robust reasoning: integrating rule-based and similarity-based reasoning

Ron Sun *

*Department of Computer Science, University of Alabama, Tuscaloosa, AL 35487, USA*

Received February 1993; revised March 1994

**Abstract**

The paper attempts to account for common patterns in commonsense reasoning through integrating rule-based reasoning and similarity-based reasoning as embodied in connectionist models. Reasoning examples are analyzed and a diverse range of patterns is identified. A principled synthesis based on simple rules and similarities is performed, which unifies these patterns that were before difficult to be accounted for without specialized mechanisms individually. A two-level connectionist architecture with dual representations is proposed as a computational mechanism for carrying out the theory. It is shown in detail how the common patterns can be generated by this mechanism. Finally, it is argued that the brittleness problem of rule-based models can be remedied in a principled way, with the theory proposed here. This work demonstrates that combining rules and similarities can result in more robust reasoning models, and many seemingly disparate patterns of commonsense reasoning are actually different manifestations of the same underlying process and can be generated using the integrated architecture, which captures the underlying process to a large extent.

## 1. Introduction

### *1.1. Patterns in reasoning*

Commonsense reasoning is one of the main problems in artificial intelligence. Commonsense reasoning is somewhat structured yet flexible, and usually reliable but sometimes fallible [11,29,61]. It has been extremely difficult for AI programs to capture such commonsense knowledge and reasoning in all its power and flexibility. Even the very concept, commonsense reasoning, is difficult to characterize: we cannot define

* E-mail: rsun@cs.ua.edu.


*SSDI* 0004-3702(94)00028-Y

what commonsense reasoning is, just as it is hard to define what intelligence is, or what knowledge is. Roughly speaking, however, commonsense reasoning can be taken, at least for the kind of commonsense reasoning explored in this work, as referring to informal kinds of reasoning in everyday life regarding mundane issues, where speed is oftentimes more critical than accuracy.

The study of commonsense reasoning as envisaged here is neither about the study of a particular domain, nor about idiosyncratic reasoning in any particular domain. It deals with commonsense reasoning *patterns*; that is, the recurrent, domain-independent basic forms of reasoning that are applicable across a wide range of domains (as we believe that such forms do exist.[1]) Allan Collins collected a large number of protocols of commonsense reasoning [8, 10] in the area of elementary geography and the like. Noticing the inadequacy of traditional logic in explaining the reasoning patterns exhibited by the protocols, he argues for alternative formalisms for patterns found in various commonsense reasoning tasks. Collins and Michalski [10] believe in the existence of common patterns (versus domain specific and/or ad hoc processes) that are widely applicable across domains (thus they actually developed a generalized logical formulation of them). This standpoint and the data on which it is based is also the starting point of studying commonsense reasoning in this work.

### *1.2. Rigor and flexibility*

One important issue in modeling commonsense reasoning is how we should handle the rigor and clarity in reasoning on one hand, and the flexible, approximate, and evidential character of the same process on the other hand. For example, we need clearly-defined structures to enable effective inferences and we need precise ways of encoding knowledge possessed by a cognitive agent: the exact prerequisites for an action, the precise outcome of a given situation, and so forth [10, 27, 35]. This imposes some necessary requirements on the type of models that are applicable. Simple backpropagation networks can be considered as unsuitable; for, although they can simulate any rule-based systems with the power of non-linearity of their node activation functions, they suffer from a few crucial shortcomings. For example, because of the complexity of non-linear signal propagation through multiple layers, they cannot keep track of their reasoning processes and produce explanations for their conclusions (although rule extraction has been attempted by e.g. [17]); they cannot distinguish (through introspection) the importance of conditions (i.e. which conditions are necessary or essential, and which are not); they do not have enough symbol manipulation capabilities, e.g. to fully account for compositionality and systematicity existing in symbolic systems [16].

On the other hand, commonsense reasoning data (such as Collins's protocols, to be detailed later) also shows that there is much flexibility in the reasoning of a cognitive agent. In order to capture this, we also need corresponding flexibility in our model. Specifically, we need means of evidential combination, with graded information (fuzzy

[1] The question of whether there exist any domain independent, recurrent common patterns in reasoning, especially in commonsense reasoning, is open (cf. [11, 29]). It is somewhat related to the debate between instance-based reasoning models and rule-based reasoning models (see e.g. [36, 42]).

and uncertain) and capable of accumulating confidence values incrementally (to be explained; [34,61]). We should also be able to deal with similarity and analogical reasoning [18,58]. (In this paper, the word *flexibility* is used throughout to denote these above aspects; see Section 7.3 for further discussions.) Traditional rule-based systems in their prevailing forms seem too cumbersome to handle various sorts of inexactness in that only several isolated kinds of inexactness can be accounted for and they have to be handled with separate mechanisms (cf. [11] regarding logic-based approaches and [10] regarding psychologically motivated work); to model similarity-based or analogical reasoning, they require special structures/mechanisms and/or complex search procedures [3,26]. Rule-based systems may have difficulties with other types of flexibility too.[2]

To satisfy the requirements on both sides, rigor and flexibility, we have to strike a balance between them carefully—this places a major constraint on developing an adequate theory to account for commonsense reasoning.

### *1.3. Connectionism versus rule-based reasoning*

A relevant issue that comes to mind in this connection is the debate of connectionism versus rule-based reasoning. It is quite clear now that connectionist models are capable of implementing rule-based reasoning in a variety of ways (e.g., [1,4,28,44,56], etc.), as well as other types of reasoning. However the following questions remain:

- Can connectionist models of rule-based reasoning do more in terms of accounting for robust human reasoning (as questioned in, for example, [42])?
- Can connectionist reasoning models reproduce directly those commonsense reasoning examples (such as those in [10]) that are difficult to be accounted for by rule-based systems in a simple and straightforward way?

Clearly, rules are efficient computational constructs for a compact, modular representation and direct, efficient reasoning. Rules are capable of rigorous descriptions of relevant knowledge. However, they traditionally lack certain flexibility, as pointed out before, which leads to particularly severe *brittleness* (or *rigidity*; more on this in Section 7.2). There are combined symbolic and numerical representations, for example, probabilistic reasoning (based on the probability theory, for treating uncertainty in rules [34]), fuzzy logic (for dealing with vague concepts [61]), and the certainty factor models (which are probabilistic in nature [6,22]). These models have certain flexibility and thus deal with the brittleness problem to a certain extent, but none of them to date can solve the problem very well, since each of them only deals with one or two aspects: fuzzy logic (as in [61]) does not deal well with cumulative evidence, the probabilistic approach (as in [34]) does not deal well with graded concepts, and so forth. None of them deals with similarity-based or analogical reasoning very well.

Connectionist models are inherently massively parallel and have the advantage of being robust, exhibiting generalization and fault tolerance. Connectionist reasoning and

[2] Researchers adopting a logic-based approach [11,29] may argue that a *normative* study of commonsense reasoning does *not* need to involve such flexibility. The main point of this research, however, is precisely the incorporation of such flexibility in a computationally efficient manner into reasoning systems, to make systems more realistic and *descriptive*.

learning are often said to be similarity-based: it usually involves utilizing previous similar training cases, either individually or collectively in a statistical way (cf. [2,37] for various analyses).

We will show, in this paper, how rule-based reasoning and similarity-based reasoning (as embodied in connectionist networks) can be integrated, and that this integration leads to a new theory of robust reasoning. This theory in its simple form can be implemented in a connectionist architecture.

### *1.4. The plan for this paper*

Based on a detailed analysis of examples and data (Section 2), a theory of robust reasoning is proposed that establishes a unified framework for diverse patterns in commonsense reasoning (Section 3). To implement the theory, an architecture combining essential characteristics of rule-based reasoning and connectionist similarity-based reasoning, CONSYDERR, is proposed (Section 4). In this architecture, a dual representational scheme is devised, which utilizes both localist representation (encoding rules) and distributed representation with features (for similarity matching). Some examples are examined in detail (Section 5). We explore the interaction between these two components, which enables the system to account for many difficult patterns (Section 6). Some discussions follow, especially on the problem of brittleness, which the theory of robust reasoning addresses (Section 7). Appendices contain some technical details of rules and logics in CONSYDERR (Appendix A) and some experiments (Appendix B).

## 2. Some examples

Let us look into a set of examples, most of which are protocols from Collins [8,10], though somewhat simplified. The goal here is not psychological data modeling, but a computational understanding.

(1) The first example shows uncertain, evidential reasoning:

> Q: Do you think they might grow rice in Florida?
> A: Yeah. I guess they could, if there were an adequate fresh water supply, certainly a nice, big, warm, flat area.

There is a rule in this example: if a place is big, warm, flat, and has an adequate fresh water supply, then it is a rice-growing area. The person answering the question deduced an uncertain conclusion based on partial knowledge, although a piece of crucial information (i.e. the presence of fresh water) is absent.

(2) The second example is as follows:

> Q: Is the Chaco the cattle country?
> A: It is like western Texas, so in some sense I guess it's cattle country.

Here because there is no known knowledge, an uncertain conclusion is drawn based on similarity with known knowledge.

(3) The third example is:

Q: Are there roses in England?
A: There are a lot of flowers in England. So I guess there are roses.

Here the deduction is based on *property inheritance*. Formally, England HORTICULTURE flower; rose IS-A flower; so England HORTICULTURE rose (to use the jargon of the inheritance theory; see [48]). The conclusion is only partially certain and is drawn because there is no information to the contrary (i.e. no *cancellation* of properties).

(4) The fourth example is:

Q: Is that [Llanos] where they grow coffee up there?
A: I don't think the savanna is used for growing coffee. The trouble is the savanna has a rainy season and you can't count on rain in general [for growing coffee].

This example shows a chain of rules in reasoning: Llanos is a savanna, savanna has a rainy season, and rainy seasons do not permit coffee growing.

(5) The fifth example is:

Q: Is Uruguay in the Andes Mountains?
A: It's a good guess to say that it's in the Andes Mountains because a lot of the countries [of South America] are.

Here there is no rule stating whether Uruguay is in the Andes or not. However, since most South American countries are in the Andes, the *default* is therefore *in the Andes*. Uruguay just "inherits" this default value (although incorrectly).

(6) The sixth example is:

Q: Can a goose quack?
A: No. A goose—well, it's like a duck, but it's not a duck. It can honk, but to say it can quack. No. I think its vocal cords are built differently. They have a beak and everything. But no, it can't quack.

More than one pattern is present here. One is based on similarity between geese and ducks, independent of knowledge regarding geese, yielding the conclusion that geese may be able to quack. Another pattern is a rule: since geese do not have vocal cords built for quacking, they cannot quack.

(7) The seventh example is:

Q: Is Florida moist?
A: The temperature is high there, so the water holding capacity of the air is high too. I think Florida is moist.

In this example the concepts involved are not all-or-nothing, but somehow graded, so the conclusion must be graded, in correspondence with the confidence values of known facts and rules.

(8) The eighth example is:

Q: Will high interest rates cause high inflation rates?
A: No. High interest rates will cause low money supply growth, which in turn causes low inflation rates.

This example shows a chaining of rules: High interest rates will cause low money supply growth, and low money supply growth will cause low inflation rates, so high interest rates will cause low inflation rates.

(9) The ninth example is:

> Q: What kind of vehicles are you going to buy?
> A: For carrying cargo, I have to buy a utility vehicle, but for carrying passengers, I have to buy a passenger vehicle. So I will buy a vehicle that is both a utility and a passenger vehicle. For example, a van.

This example shows the additive interaction of two rules: if carrying cargo, buy a utility vehicle, and if carrying passengers, buy a passenger vehicle. The result is the combination of the two rules: something that is both a utility vehicle and a passenger vehicle.

(10) The tenth example is:

> Q: Do women living in that [tropical] region have short life expectancy?
> A: Men living in tropical regions have short life expectancy, so probably women living in tropical regions have short life expectancy too.

This is another case of using similarity because of the lack of direct knowledge.

(11) The eleventh example is:

> Q: Are all South American countries in the tropical region?
> A: I think South American countries are in the tropical region, because Brazil is in the tropical region, Guyana is in the tropical region, Venezuela is in the tropical region, so on and so forth.

Although the conclusion is incorrect, this example illustrates *bottom-up inheritance* (a form of generalization). Since there is no knowledge directly associated with the superclass "South American countries" as to whether they are in the tropical or not, subclasses are looked at, and a conclusion is drawn based on the knowledge of the subclasses.

One can observe from these examples that (cf. [10]):

- The same patterns are present in many different situations (see e.g. Examples 2, 6, and 10).
- People are more or less certain about their conclusions depending on their certainty of information (including rules and facts; see e.g. Examples 1 and 7).
- People have some means of applying existing knowledge, and some means of performing similarity matching when there is no matching existing knowledge (many examples above indicate the existence of the two processes, individually or intermixed together).

A methodological note is in order here: in describing different patterns, we use rules if conditions are explicitly mentioned and manipulated; we use similarity matching if none of the relevant features or conditions is mentioned explicitly (it is thus suggestive of a holistic process); when in similarity matching, one concept is a superclass (or subclass) of the other, it is a case of inheritance.

## 3. A theory for accounting for the reasoning patterns

There are some important unanswered questions about commonsense reasoning. For example, what are the basic patterns in commonsense reasoning? What notions can be used to best characterize these patterns? What is the most fundamental problem underlying the difficulty in producing commonsense reasoning as in these data? We will try to answer (tentatively) some of these questions. Let us first establish some simple facts about commonsense reasoning.

### *3.1. Some basic forms*

#### *3.1.1. Rules*

First of all, there is the question of the proper form of knowledge representation for applying existing, directly-applicable knowledge (such as that in examples 1 and 7). Although there are many alternatives available, by all accounts, *rules* seem to be the best choice as an appropriate or even necessary form for expressing various kinds of knowledge, for a number of reasons. First of all, phenomenological evidence for the existence of rules in reasoning is mounting: Smith et al. [42] present eight criteria for the existence of rules in cognition; detailed experimental results are analyzed which show that the eight criteria can be satisfied by various data; so the conclusion is drawn that rules are an intrinsic part of cognition; Fodor and Pylyshyn [16] argue that linguistic and others processes require systematicity which only symbol manipulation and rule-based reasoning can provide; Pinker and Prince [35] show that phonological performance can be better modeled by utilizing rules, at least as a part of a mechanism.[3] Rule-based reasoning and symbolic manipulation provide some of the rigor and flexibility that are necessary in developing robust reasoning capabilities in commonsense reasoners.

Secondly, examining the examples discussed above, there are clear indications of the existence of rules; for example, in the Florida case, there is undoubtedly a rule with four conditions (big area, warm area, flat area, and fresh water supply) and one conclusion (rice-growing area). Conversely, examining all the examples in Section 2, although there are maybe more than one ways for encoding some knowledge, all directly applied knowledge can be captured (computationally) in rules rather naturally, as discussed earlier at length.

In addition, at the computational level, the following reasons support the use of rules:

- Rules are the most common form of knowledge representation, used widely in all kinds of AI systems.
- It has been convincingly argued that other knowledge representation schemes can be transformed into logic (rule) based schemes [7, 20, 30].
- Rules are precise but allow incorporation of confidence measures, uncertain knowledge, and plausible inference processes [34, 61].

[3] Some people may not agree with these opinions. I will not get into the controversy surrounding these arguments.

- Rules ensure modularity in representation, making the representation easy to construct and manipulate and making it easy to incorporate new knowledge and change existing ones (the detail of this aspect is not addressed in this paper).
- Representation with rules facilitates explanation generation (explaining inferential processes) and improves human comprehensibility in many other ways.

Next, back at a phenomenological level, we can see that commonsense reasoning processes are *evidential*, which means that the existing knowledge, or rules, are not deterministic, a priori, or transcendentally true. Rather, they are empirical, inexact, and uncertain. Based on the observations in Section 2, we further conjecture the following in rule representation:

- Concepts or propositions involved in reasoning processes are often graded, that is, not all-or-nothing but fuzzy, possibilistic, or probabilistic [14]. For example, "warm" is a fuzzy concept and there is no fixed boundary as to what is warm and what is not; similarly, the proposition "raining causes flooding" is a probabilistic rather than deterministic proposition. We can associate with each of those concepts and propositions a generic confidence measure that can be used to facilitate reasoning processes.
- As suggested by data (e.g. Example 6), different pieces of evidence are often weighted, that is, each of them may have more or less impact, depending on its importance or salience (see Osherson et al. [32] for additional evidence and arguments). We need a way of combining evidence from different sources with different weights, without incurring too much computational overhead (such as in probabilistic reasoning or Dempster–Shafer calculus [34,38]).
- The evidential combination process may be cumulative, or in other words, it tends to "add up" various pieces of evidence to reach a conclusion, with a confidence that is determined from the "sum" of the confidences of the different pieces of evidence. Knowing two conditions in a rule results in a larger confidence than knowing only one. For example, in the example regarding whether Florida is a rice-growing area, if we know all the four conditions, warm, flat, big and fresh water supply, we can make the conclusion with full confidence; when we know only three of the four conditions, we reach the same conclusion with less confidence. A cumulative evidential combination procedure is therefore necessary in rule representation.

### *3.1.2. Similarities*

The above data also clearly indicate the need for similarity matching and some form of analogy in reasoning: in situations where there is no directly applicable knowledge (such as in examples 2 and 6), one can find similar concepts, propositions or situations, within the current context, and come up with some plausible conclusions based on their similarity. The confidence in the conclusion can be determined based on the degree of the similarity. Phenomenologically, the comparison process that determines similarities is an intuitive, holistic and unstructured process, for the above protocols and examples do not indicate anything deliberative or analytical. This phenomenon is also recognized by e.g. Dreyfus and Dreyfus [13], Hinton [23], and Smolensky [43], based on theoretical and experimental observations. Computationally, however, similarity can be implemented as

rules [10] and thus only one process (rule application) is left. Such an approach creates two problems:

(1) one concept is similar to many other concepts, and thus too many rules will have to be added into a system to capture all of these similarities; this tends to make systems for any reasonably large domain unwieldy (to say the least) because of the existence of too many rules;

(2) it will be difficult then to distinguish between strong rule-governed reasoning and mere associations based on similarities, whereas the distinction is rather clear commonsensically.

It is evident that rule application and similarity matching are intrinsically mixed together; for example, in the protocol about geese, the application of a rule regarding vocal cords is intertwined with the similarity matching with ducks. By combining the two processes, many interesting inferences can be made with relative ease. In other words, it is the interaction between the two processes that creates these interesting reasoning patterns (for example, top-down inheritance, bottom-up inheritance, and cancellation; see Section 2). Therefore it is important to study their interaction and come up with a computational model within which the interaction can be utilized.

There is evidence suggesting that reasoning processes with similarity matching (comparison of analogous knowledge) are massively parallel and spontaneous (i.e. automatic) (see e.g. [60]), which should be taken into account in any theory of commonsense reasoning. This is also the case with rule application, which is oftentimes also parallel and spontaneous (see e.g. [13,25] for some arguments).

### 3.2. A synthesis: the theory of robust reasoning

#### 3.2.1. Some diverse patterns

We can summarize the patterns of commonsense reasoning in the examples (in Section 2) as follows (cf. [45,46]):

- Partial information (e.g., the first example), in which not all relevant information is known but a conclusion has to be drawn.
- Uncertain or fuzzy information (e.g., the first example again), in which information is not known exactly and with absolute certainty, but a plausible conclusion has to be drawn based on what is known.
- Similarity matching (e.g., the second example), in which rules describing similar but different situations are used due to the lack of exact matching rules (in case of novel input).
- Combinational rule interactions (e.g., the ninth example; see [49] for complete details regarding this aspect), in which conclusions and conditions of multiple rules combine to produce strengthened, weakened, or entirely new results, made possible by the lack of consistency and completeness resulting from a fragmented rule base.
- Top-down inheritance (e.g., the third example), in which information regarding superclasses is brought to bear on the subclasses.
- Bottom-up inheritance (e.g., the eleventh example), in which information regarding subclasses is brought to bear on the superclasses.

Although these patterns seem disparate, a theoretical synthesis below will show their commonalities.

*3.2.2. One underlying process*

*Some definitions* To perform a precise theoretical analysis, we need some definitions. A rule is defined here to be a structure consisting of some conditions and a conclusion; a numerical weight is associated with each condition. Whenever conditions are activated (to a degree commensurate with the confidence of the corresponding facts), the activation of the corresponding conclusion can be determined by multiplying the activation values of the conditions by the weights. This computation is commonly used (see [40, 50]) and adopted here for its intuitive appeal and simplicity (the justifications will be discussed in Section 6.1.1). We will denote a rule by $A \longrightarrow B$. And if $A$ is activated, the activation of $B$ due to $A$ is denoted as $A * (A \longrightarrow B)$.

Similarity can be defined here (a little simplistically) as a measure of the amount of overlap between the corresponding feature sets of the source and target concepts or propositions (taking into consideration the sizes of the source feature set; detailed analyses later). We will denote similarity by $A \sim B$. So if $A$ is activated, the activation of $B$ due to $A$ is denoted as $A * (A \sim B)$, that is, the activation of $A$ times the similarity measure between $A$ and $B$. (See [41] for some psychological evidence and arguments for a similar definition.)

*The analysis* To achieve a synthesis of the patterns identified in Section 3.2.1, we will analyze each of them on the basis of rules and similarities.

(1) When we have inexact information, the inexactness can be quantified with a confidence value, and the value can be used in reasoning. Given

$$A \longrightarrow B$$

if $A$ is activated to a degree commensurate with its confidence level, then

$$B = A * (A \longrightarrow B)$$

where $A$ and $B$ represent respective activations, $(A \longrightarrow B)$ represents the weight from $A$ to $B$ (i.e. the rule strength, a number), and "*" is multiplication.

(2) When we have incomplete information (that is, when we do not have all the requisite conditions to apply a rule), we can still deduce a conclusion, although with less confidence; for the confidence of a conclusion is determined based on the vector multiplication computation (i.e. the inner product, a simple extension of scalar multiplication). Suppose we have rules

$$A\ B\ C \longrightarrow D.$$

When given confidence values of $A$ and $B$ with $C$ unknown (zero activation), $D$ is deduced with less confidence than when given full confidence values of all of $A$, $B$ and $C$:

$$D = (A\ B\ 0) * (A\ B\ C \longrightarrow D)$$

where $(A\ B\ C \longrightarrow D)$ represents a vector of the three weights, and they are applied to the activation values of the conditions of the rule, $(A\ B\ C)$, to get the weighted-sum (the inner-product).

(3) The similarity matching situation (due to the lack of any exactly matching rule when encountering novel input) can be described as:

$$A \sim B,$$

$$B \longrightarrow C$$

and $A$ is activated (i.e. the activation $A \neq 0$); that is, we want to know about $A$ (e.g. Chaco), but there is no rule directly applicable beside a similarity with $B$ (e.g. Western-Texas). So we utilize the similarity between $A$ and $B$, and the knowledge $C$ associated with $B$ (e.g. cattle-country):

$$B = A * (A \sim B)$$

where $A$ represents the activation of the concept, $(A \sim B)$ represents the similarity between $A$ and $B$, and "$*$" is multiplication; so

$$C = B * (B \longrightarrow C) = A * (A \sim B) * (B \longrightarrow C)$$

where $(B \longrightarrow C)$ represents the weight (the rule strength) from $B$ to C.[4]

(4) For top-down inheritance, suppose $A$ is a subclass of $B$, A's property value is unknown, $B$ has a property value C, and we want to know the corresponding property value of A, that is,[5]

$$A \sim B,$$

$$B \longrightarrow C.$$

When $A$ is activated, $C$ will be activated accordingly, i.e.

$$C = A * (A \sim B) * (B \longrightarrow C).$$

(5) For bottom-up inheritance, suppose $B$ is a superclass of A, $B$'s property value is unknown, and $A$ has a property value $D$, and we want to know the corresponding property value of $B$, that is,[6]

$$B \sim A,$$

$$A \longrightarrow D.$$

When $B$ is activated, $D$ will be activated accordingly, i.e.

$$D = B * (B \sim A) * (A \longrightarrow D).$$

[4] Similarity matching is oftentimes context-sensitive; that is, similarity comparisons may have to be in relation to a particular context. See discussions in Section 7.

[5] The super/sub-class relation is a special case of the similarity relation. It is stronger than the general case (see [49]).

[6] It is generally the case that the bottom-up inheritance (a form of induction) is less reliable than the top-down case (a form of deduction), although they both can be based on similarity.

(6) For cancellation of inheritance, suppose $A$ is a superclass (or subclass) of $B$, $A$ has a property value $D$, and $B$ has a property value C, then

$$A \sim B,$$

$$B \longrightarrow C,$$

$$A \longrightarrow D.$$

When $A$ is activated, $D$ will be activated more than $C$.

(7) In case of rule interaction, we can describe the situation as

$$A \longrightarrow C,$$

$$B \longrightarrow D,$$

$$C \sim D.$$

When $A$ and $B$ both are activated, the interaction of $C$ and $D$ might result in something else being strongly activated, depending on their mutual similarity (see example 9).

The above synthesis of these different patterns provides on one hand the rigor and precision (in the sense explained in Section 1), and on the other hand the flexibility, which underlies most patterns of commonsense reasoning. A proper balance and mixture of the two in a theory of commonsense reasoning is our main goal as stated in Section 1.

## 4. A connectionist architecture

Given the theoretical synthesis of all the patterns, a unifying mechanism for rule applications and similarity matching is needed for actually carrying out the proposed theory.

### *4.1. Connectionist models of reasoning*

Let us examine some relevant existing models first.

#### *4.1.1. Connectionist rule-based reasoning*

There are a number of connectionist models of rule-based reasoning, which are of interest because they might provide a unifying mechanism that we are looking for. Connectionism has made some claims about rule-based behavior and reasoning. One point of view is that rule-like behavior is the result of complex interactions of network components, in deterministic or statistical ways, and therefore there is no fundamental difference between rules and non-rules. The point of view has been rebutted by many researchers working within the "symbolic" paradigm (see [16,35]). Another viewpoint held by many connectionists advocates structuring networks for direct rule-based reasoning [51,52]. Let us examine some of these models.

Touretzky and Hinton [56] present the first work in implementing rule-based reasoning in connectionist networks. They basically emulate the structure of a symbolic rule-based (production) system, with separate modules for working memory, rules, and facts; an elaborate pull-out network is designed to match working memory data against rules (when there is no exact match, the best match wins) and to decide which matching rule is to fire. The mechanism executes one rule at a time and there is no backtracking. The resulting system is the equivalent of a simple sequential symbolic rule-based system.

Derthick [12] presents a connectionist knowledge representation system in which inferences are carried out in parallel by constraints satisfaction through minimizing energy. By constructing the energy function in a way that captures the underlying relations between logical formulas, this process can produce an optimal (or near optimal) interpretation of the inputs. However, not all logical relationships between sentences and not all inference modes can be captured in energy functions. As a matter of fact, most of the commonsense reasoning, as shown by the protocols and examples (see Section 2), cannot be accomplished by energy minimization processes in any obvious way. In addition, the Boltzmann machine (used for energy minimization) is extremely slow, and yet does not guarantee the best results.

Recent work on connectionist models of rule-based reasoning includes [1,28,44]; they all have a connectionist network that uses local representation and can perform rigorous logical reasoning (including variable binding); their internal mechanisms are somewhat different. However, none of them deals with similarity matching. See [50] for detailed analyses.

Although almost all of these aforementioned connectionist rule-based systems are interesting in some way, they do not deal with inexactness and similarity matching sufficiently; therefore, they are unable to do more than what a typical symbolic rule-based system is capable of, besides parallelism. We need a principled way of integrating rule-based components and similarity-based components. The integration could on one hand add continuity (embodied in similarity-based connectionist networks) to a discretized rule-based system, so that it can better model continuous thought processes, and on the other hand, it could add structures (namely rules, as well as variables and bindings; see Section 7.1) to a structureless, associationistic connectionist network, giving it the rigor, precision and directedness necessary for modeling some cognitive tasks (cf. [1]).

#### *4.1.2. Connectionist case-based reasoning*

Case-based reasoning (CBR) is another interesting approach in AI that utilizes similarity (among cases) extensively. It is claimed [36] that "case-based reasoning is the essence of how human reasoning works". However, I believe that, although cases are important, rules (abstract knowledge) are also fundamental cognitive mechanisms. For example, when there is a rule: **if** a place is warm, flat and with enough fresh water supply, **then** it can be a rice-growing area, we only need to apply such a rule in deciding whether an area is a rice-growing area; only when there is no such a rule, or the conclusion of the rule is indecisive, may we apply analogous knowledge (cases). In existing case-based reasoning systems, various forms of rules are used, which also suggests the primary role of rules. I also believe that similarity matching should be done in

a massively parallel fashion (see [54,60] for similar points) and spontaneously (automatically), without incurring huge computational overhead. Phenomenological analyses also suggest similarity matching is carried out at a lower level, the subconceptual level, and matching is an intuitive, holistic process (as explained before; see [13]). Thus, simpler, massively parallel methods of matching are preferred in carrying out case-based reasoning.

Barnden and Srinivas [5] describe an attempt at utilizing case-based reasoning in connectionist networks. The system has some configuration matrices (CMs), each of which contains a case for short-term processing; a small set of gateway CMs provides the interface between short-term processing and long-term memory; cases in non-gateway CMs compete to have their contents copied into gateway CMs, where it can cause cases similar to it to be retrieved from long-term memory, and some symbol substitutions take place to adapt the cases. It is an interesting idea, though the complex structures, gateways, and copying/substitution mechanisms seem to hamper parallelism.

### *4.2. A unifying mechanism*

The above reviewed work, though not directly applicable, does suggest some useful ideas for a computational mechanism. I will outline below a unifying mechanism for carrying out the basic processes of rule-based reasoning and similarity-based reasoning. This mechanism, a connectionist architecture, consists of two levels: the first level, called CL, utilizes local representation, that is, one node for each domain concept; the other level, called CD, is more fine-grained, utilizing distributed representation with fine-grained feature (interpretable or uninterpretable) into which all domain concepts can be decomposed. By dividing the architecture into two levels, we can utilize the interaction between the two representations of different granularity to make the architecture more effective and computationally more efficient. Fig. 1 shows a sketch of the architecture. We will call this architecture CONSYDERR, which stands for *a CONnectionist SYstem with Dual representation for Evidential Robust Reasoning.*

The reason for explicit localist representation in CL is the need to explicitly implement rule applications in a connectionist network. We assume that this network carries out reasoning at the *conceptual level*; so the representation has to be explicit and individuated, in order for the concepts and reasoning processes to be consciously accessible and linguistically expressible, without extra matching networks or decoding networks (which is the case with distributed representations); we want explicitness, so that activated concepts can be easily identified, reasoning processes can be traced, and explanations can be generated (as mentioned in Section 1). This leads directly to the idea of local representation: each concept or proposition in a domain is represented by a single node in a network of nodes [51]; rules are then implemented by links between nodes representing conditions and nodes representing conclusions.

With rules being represented by links between nodes, reasoning can be done with only local computation—rule activations can be calculated within one or a few nodes, in place, without the need to perform indexing, retrieval, reorganization and updating of large databases of facts. Although there are various heuristics for reducing search time in large knowledge bases, none of them succeeded significantly. The overhead of

selecting data and moving data around can be avoided by directly connecting together those pairs of facts that are related by rules so that only local computation for passing on activations is necessary. This design thus has a computational advantage over more traditional rule-based systems. This design also has an advantage cognitively: the speed in human commonsense reasoning can be better matched [9].

Commonsense reasoning by nature is evidential, cumulative, and graded (as has been discussed before), which rule encoding in CL has to deal with. One possibility is to use graded and continuous activations (representing confidence values or certainty values) and weighted-sum combination functions (representing evidential combination in rule firing) for this purpose; thus rule encoding coincides with the operation of typical connectionist models and consequently has an easy implementation in a connectionist network. This idea will be examined and evaluated later (see also [50]); extensions into strict logical conjunctions or other complex combinations are also possible when needed.

To carry out similarity matching, a distributed representation (based on features) is needed, which must be at a different level because of its distributed character; thus the CD level comes into play. The nature of distributed feature representation decides that it is *similarity-based*: the amount of overlap between two sets of nodes representing two different concepts is proportional to the degree of similarity between these two concepts or propositions. On the other hand, the links in CD are the replications of the links in CL (which represent rules); that is, if there is a link between two nodes (or the two concepts they represent) at the CL level, then we will add a link between each node in the feature set of the first concept and each node in the feature set of the second concept, replicating diffusely the original link at CL (having the full cross-product connection between the two feature sets). This corresponds to the idea of incorporating analytical knowledge (which can be represented by a localist network) into intuition (which can be partially captured by a similarity-based distributed connectionist network; see [13]), although we will not discuss the learning process per se here.

For more flexibility, interactions between the two levels have a top-down path and a bottom-up path separately. Also, the operation of the system is in cycles; in other words, the interaction of the two levels is not ever-present, and each part is independent to a certain extent. One cycle can be divided into three phases: the top-down phase, the settling phase, and the bottom-up phase, in which top-down flows only occur during the top-down phase and bottom-up flows only occur during the bottom-up phase; in other words, the two levels have *phasic* interactions [49]. The computational utility of cycles (and the two-level structure on which they operate), as will become clear later on, is in avoiding time and space complexity as in e.g. Hopfield networks and other common types of connectionist networks [4], which requires ever-present global connectivity and thus results in slow settling. This structure may also be partially justified cognitively by the phenomenon of multiple streams of thoughts (see e.g. [31]).

### *4.3. A precise description of the mechanism*

#### *4.3.1. The equations*

A set of equations for describing the computation during the three phases is:

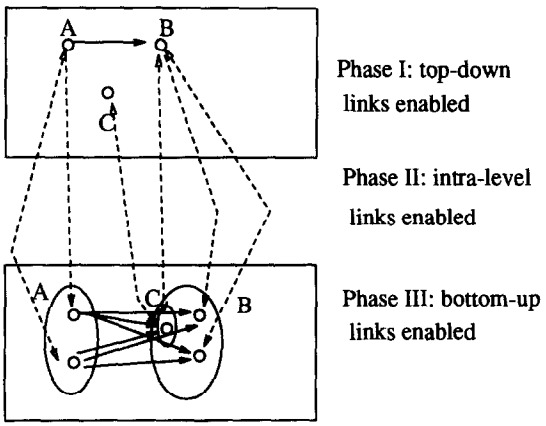



Fig. 1. The architecture. The top level is CL and the bottom level is CD. $A$, $B$ and $C$ are concepts, and the link between $A$ and $B$ represents a rule $A \longrightarrow B$. For example, $A$ = Chaco, $B$ = Cattle-country, and $C$ = Western-Texas.

For the top-down phase,

$$x_i(t+1) = \max_{x_i \in F_A} f(A(t))$$

where $A$ is any node in CL, $F_A$ is the set of nodes in CD that correspond to (i.e. represent) $A$, and $f$ is a monotonic increasing function; that is, a node in CD receives activation from the corresponding node in CL, and chooses the largest value; $t$ denotes the time period.

For the settling phase,

$$\Delta A(t+1) = \alpha \sum_i W_i I_i(t) - \beta A(t)$$

and

$$\Delta x_j(t+1) = \mu \sum_i w_i i_i(t) - \nu x_j(t)$$

where $W_i, w_i$ are link weights (that represent strengths or weights in rules, as discussed later), and $I_i, i_i$ are the activation of related facts (conditions or logical antecedents); that is, in this case, each node receives activations from other nodes at the same level and does a weighted-sum for incrementing its own activation. $\alpha, \beta, \mu, \nu$ are parameters controlling the network dynamics: $\alpha$ and $\mu$ are rates of changes, and $\beta$ and $\nu$ are decay rates.

For the bottom-up phase,

$$B(t+1) = \max(B(t), \sum_{x_i \in F_B} g(x_i(t)))$$

where $B$ is any node in CL, $F_B$ denotes the set of CD nodes representing $B$, and $g$ is a monotonic increasing function; that is, a CL node receives and sums up

activations from its corresponding set of CD nodes, and chooses the result as its activation if it is greater than its original activation (in case there are negative activations, absolute values are used in comparisons).

To simplify the matter, in this paper, we set $\alpha = \beta = \mu = \nu = 1$. So, derived from the above equations, the equilibrium state equations for the settling phase are,

$$A = \sum W_i * I_i,$$

$$x_i = \sum w_i * i_i$$

where $I_i$'s and $i_i$'s are final converged inputs to the two nodes, respectively; these inputs are from other nodes that are linked to $A$ and $x_j$; they eventually converge to some constant values, when external inputs are clamped to some nodes in the network and the network is given enough time to settle. The equilibrium equations for this architecture amount to a simple weighted-sum computation.[7]

Similarly, to simplify the top-down phase equation, we adopt a top-down weight $td$ and a bottom-up weight $bu$; we set $f(A) = td_A * A$, so that only a parameter $td$ needs to be determined. To simplify the bottom-up phase equation, we set $g(x_i) = bu_B * x_i$, where $x_i \in F_B$; this reduces the function to one parameter $bu$. So we have:

- For the top-down phase,

$$x_i(t+1) = \max_{x_i \in F_A} tu_A * A(t).$$

- For the bottom-up phase,

$$B(t+1) = \max(B(t), \sum_{x_i \in F_B} bu_B * x_i(t)).$$

*4.3.2. The parameters*

We need to specify the following parameters: weights for links between a node in CL and a node in CD (i.e. inter-level links, both top-down and bottom-up), denoted as $td_A$ and $bu_A$ (for any CL node $A$), and weights for links between two nodes in CD, denoted as $lw_{AB}$, where $A$ is the CL node that corresponds to the originating CD node and $B$ is the CL node that corresponds to the receiving CD node. (In contrast, links between two nodes in CL, and their associated weights, are taken as given, which represent rules.) We use the following parameter values (see [49]; detailed verifications later):

$$td_A = 1,$$

$$lw_{AB} = \frac{r}{f(|F_A|)},$$

$$bu_B = \frac{1}{g(|F_B|)}$$

[7] Each node in the system can have one or more sites [44], each of which computes the weighted-sum (or any other similar functions whenever needed) of the inputs for one rule. The maximum of the values computed by all the sites is taken to be the activation value of the node. See Appendix A for details.

where $f$ and $g$ are monotonic increasing functions that are slower than but close to linear functions, and $f$ is much closer to linear functions than $g$; $|F_A|$ denotes the size of the feature set of $A$; $A$ is the CL node where the rule link originates, and $B$ is the CL node where the rule link terminates; $r$ is the rule strength between $A$ and $B$ (in CL). The value range of the activation is $[-1, 1]$.

The above parameters lead to a similarity measure (which is easily obtained from the above equations):

$$(A \sim B) \approx \frac{|F_A \cap F_B|}{|F_B|}.$$

*4.3.3. Elaborations*

Applying the above cycle, first some nodes in CL get activated by external inputs (and clamped). Then the top-down phase will activate (and clamp) the CD nodes corresponding to the active CL nodes. In the settling phase, links representing rules related to those activated nodes take effect in both CL and CD. Because of similarities, concepts may have overlapping CD representations, so some of the CD representations will be partially activated if a concept *similar* to them is activated in CD. Finally in the bottom-up phase, fully or partially activated CD representations will percolate up to activate the corresponding nodes in CL. The result can be read off from CL.

The massive parallelism and spontaneity in the above specified architecture are evident: activations are propagated, in a massively parallel and spontaneous fashion, from all pre-link nodes to all post-link nodes; each node receives inputs as soon as it can, and therefore fires as soon as it can, ensuring a maximum degree of parallelism in terms of rule application. For similarity matching, all similar concepts are activated (in their CD representations) immediately and simultaneously once an original concept is activated, and matched automatically with the original one (through top-down and bottom-up flows); thus the architecture is extremely efficient by employing the two-level structure.

## 5. Analyzing some examples

Some examples will help to illustrate the architecture. More extensive experiments can be found in Appendix B.

### *5.1. The first example*

Let us look at the following example:

> Q: Do you think they might grow rice in Florida?
> A: Yeah. I guess they could, if there were an adequate fresh water supply, certainly a nice, big, warm, flat area.

The rule used is:

> *big-area warm-area flat-area fresh-water-supply* $\longrightarrow$ *rice-growing-area.*

This can be handled by CONSYDERR: Each node in CL represents a concept, including "big-area", "warm-area", "flat-area", "fresh-water-supply", and "rice-growing-area". The rules are represented by links between nodes. The weights on the links reflect degrees of confidence in the respective implications, as well as positiveness/negativeness of the implications. The reasoning process is as follows: First three out of the four conditions are activated to certain degrees (which reflect the corresponding confidence in these facts), then they send their activation to the node representing the conclusion ("rice-growing-area") and activate that node. Because of one missing condition, the activation, calculated with weighted-sum, will be less than one, but still greater than zero. Therefore we conclude that it *might* be a rice-growing area. The equilibrium state equations are as follows:
In CL,

$$E = A * w_{AE} + B * w_{BE} + C * w_{CE} + D * w_{DE}$$

where $A$ = *big-area*, $B$ = *warm-area*, $C$ = *flat-area*, $D$ = *fresh-water-supply*, and $E$ = *rice-growing-area*; $w$'s are the rule weights. In CD,

$$E = A * \frac{|F_A|}{f(|F_A|)} * \frac{|F_E|}{g(|F_E|)} * w_{AE} + B * \frac{|F_B|}{f(|F_B|)} * \frac{|F_E|}{g(|F_E|)} * w_{BE} + C * \frac{|F_C|}{f(|F_C|)} * \frac{|F_E|}{g(|F_E|)} * w_{CE} + D * \frac{|F_D|}{f(|F_D|)} * \frac{|F_E|}{g(|F_E|)} * w_{DE}.$$

Overall,

$$E = A * w_{AE} + B * w_{BE} + C * w_{CE} + D * w_{DE}.$$

### 5.2. *The second example*

Another example is as follows:

Q: Is the Chaco the cattle country?
A: It is like western Texas, so in some sense I guess it's cattle country.

Here, because there is no direct knowledge regarding Chaco, an uncertain conclusion is drawn based on similarity. The knowledge is expressed in a rule:

*Western-Texas* $\longrightarrow$ *cattle-country*

represented in CL by the two nodes, one for "Western-Texas" and the other for "cattle-country", and the link between the two nodes. The similarity between the two areas

*Chaco* $\sim$ *Western-Texas*

is implemented through feature overlapping in CD. And the CL links are diffusely replicated in CD. The reasoning process is as follows: first the node for Chaco is activated; in the top-down phase the CD representation of Chaco is activated and because of shared features, the CD representation of Western-Texas is activated partially to a degree proportional to the similarity measure; then in the settling phase, the links representing

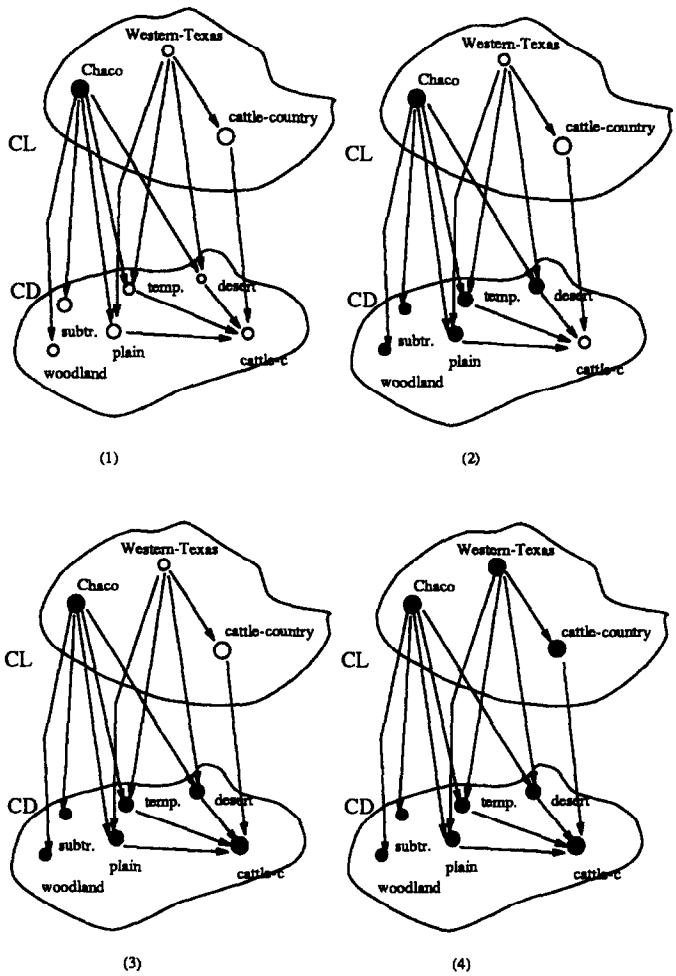



Fig. 2. The reasoning process for protocol 2. Black circles represent activated nodes. (1) Receiving inputs, (2) Top-down, (3) Settling (rule application), and (4) Bottom-up.

rules take effect in CD, so the CD representation of cattle-country is partially activated; finally in the bottom-up phase, the partially activated CD representation of cattle-country percolates up to activate the node representing cattle-country in CL. See Fig. 2.

The equation (combining CL and CD) is as follows:

$$C = \frac{|F_A \cap F_B|}{f(|F_B|)} * \frac{|F_C|}{g(|F_C|)} * w_{BC} * A$$

where $A$ stands for *Chaco*, $B$ for *Western-Texas*, and $C$ for *cattle-country*.

### 5.3. The third example

The following example involves inheritance, handled by a combination of both rule application and similarity matching.

Q: Are there roses in England?
A: There are a lot of flowers in England. So I guess there are roses.

It can be described by

$$\textit{England} \longrightarrow \textit{flower},$$

$$\textit{flower} \sim \textit{rose}$$

and we have $\textit{flower} \supset \textit{rose}$ and, in turn, $F_{\textit{flower}} \subset F_{\textit{rose}}$. The same way as before, this can be implemented in CONSYDERR with the two-level dual representation and their interaction. See Fig. 3 for details of the reasoning process. Overall, CONSYDERR deals successfully with inheritance (see [48] for complete treatments of inheritance).

### 5.4. A further point

In the above examples, the CONSYDERR network contains a large number of relevant and irrelevant nodes and links. How do we find the right links and nodes for certain reasoning in this myriad? The above discussion shows that the system is massively parallel and reasoning is spontaneous, so the problems of selecting from moment to moment a particular path to pursue and performing backtracking when reaching dead end are no longer existent. Because of the *massive parallelism* in the architecture, we can perform simple forward-chaining reasoning [8] in a parallel fashion efficiently, without the need for backtracking and without the need for the more difficult backward-chaining as some other reasoning systems do (cf. [21]). The results will be activated spontaneously following the activation of initial conditions (along with other information). See Appendix B for further examples.

## 6. Evaluations

### 6.1. Some preliminary evaluations

We will first evaluate the basic representational forms of CONSYDERR.

#### 6.1.1. Rule encoding

We will briefly examine the rule encoding scheme used in CONSYDERR (that is, *FEL*, or *Fuzzy Evidential Logic*).

Facts and rules are basic elements in this scheme. A fact (an atom or its negation) is represented by a node having a value between -1 and 1 (continuously), which is

[8] Goal components can be added to rules, so that reasoning will be more "goal-directed" (see [53] for details). Other devices of this sort are also possible.

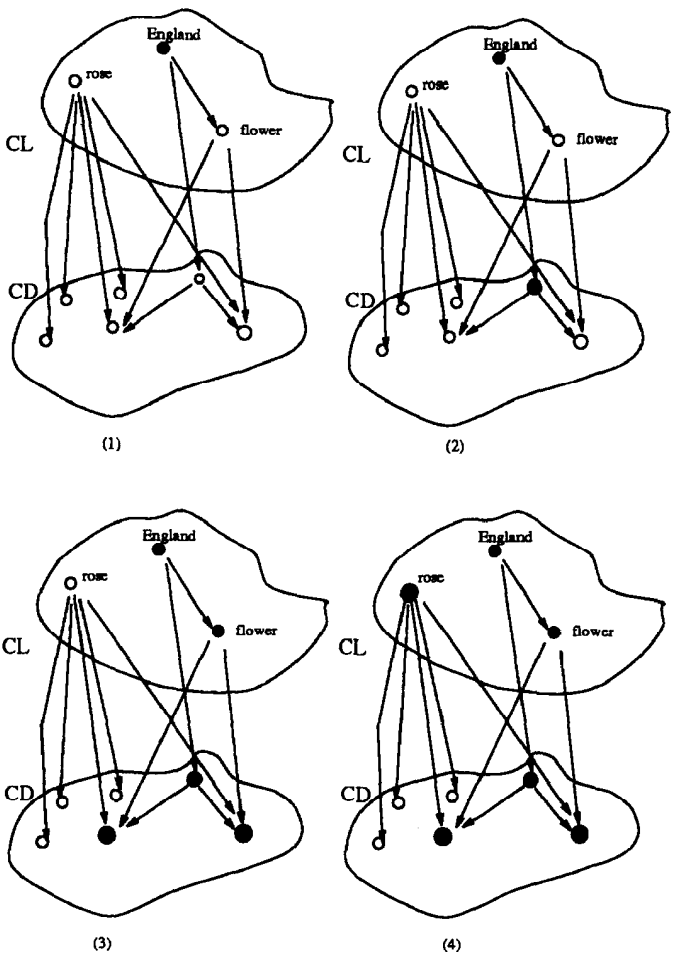



Fig. 3. The reasoning process for protocol 3. Black circles represent activated nodes. (1) Receiving inputs, (2) Top-down, (3) Settling (rule application), and (4) Bottom-up.

a generic confidence measure of the fact [49,53]. The value of an atom is related to the value of its negation by $A = -(\neg A)$. A rule is a structure composed of two parts: a left-hand side (LHS), which consists of one or more facts (conditions), and a right-hand side (RHS), which consists of one fact (the conclusion); it is encoded by having one link from each fact in the LHS to the RHS (as specified before). When facts in LHS get assigned values, the fact in RHS can be assigned a value according to a weighting scheme, the simplest form of which is the weighted-sum (possibly thresholded) operation, where each weight represent the relative importance of the corresponding fact. When the value of a fact in LHS is unknown, a zero is assumed as its value, which means no information available regarding its truth, so that reasoning can continue despite the missing information. A set of such rules connected together forms a network (a CL or a CD). A rule set is said to be hierarchical, if it maps onto

a feedforward network.

To evaluate this scheme, we can show that, as a special case, this rule encoding scheme can implement Horn clause logic [47]. Horn clause logic is a logic in which all formulas are in the forms of $P$ or $P_1 P_2 \ldots P_n \longrightarrow Q$, where P's and Q are propositions. We can specify a special case of FEL, in which values associated with facts are binary, each weight is positive, total weights of each rule sum to 1, and all thresholds are set to 1. Then it is sound and complete with respect to Horn clause logic (that is, the two are equivalent to each other). The basic idea is simple: we make the threshold of every rule to be 1, and assign the same weight to each fact in the LHS of a rule that equals 1 divided by the number of facts in the LHS. Thus the fact in RHS of a rule is activated if and only if all of the facts in the LHS of the rule is activated fully. The activation of RHS is either one or zero. So whatever is activated in FEL is true in corresponding Horn clause logic, and vice versa. (Therefore, LHS of an FEL rule is a logical conjunction in this case.) The details are in Appendix A.

To further ascertain its logical capability, we can show that FEL can simulate a modal logic—Shoham's Causal Theory [39] (see Appendix A). FEL can not only handle Shoham's logic as a special case, but also serve as an extension to it for better accounting for commonsense causal knowledge (see [50]).

*6.1.2. Similarity measures*

Similarity matching has the following characteristics (for general discussions of similarity measures, see [33,54,57]):

- The degree of similarity from concept $A$ to concept $B$ ($A \sim B$) [9] must depend on the amount of overlapping of the two corresponding feature sets, when everything else is equal. For example, suppose $A$ is river-valley, lowland, temperate area, $A'$ is river-valley, lowland, tropical area, and $B$ is temperate lowland; obviously $A$ matches $B$ better than $A'$ matches $B$.
- The degree of similarity from concept $A$ to concept $B$ must depend on the size of the feature set of $B$, when everything else is equal, because a larger feature set of $B$ means there are a lot other features in $B$ that do not match those of $A$. For example, suppose $B$ is temperate plain and $A$ is temperate area; $A$ matches $B$ quite well. Suppose $B'$ is temperate prairie plain; $A$ matches $B'$ less well.
- The size of the feature set of $A$ is not important in determining the similarity from $A$ to $B$. For example, suppose $B$ is temperate plain and $A$ is temperate area, and further suppose A' is temperate rainy area; the match from $A$ to $B$ is no less strong than the match from A' to $B$, because one extra feature in $A$ does not make $B$'s properties more or less true for $A$. What is important is what gets matched in $A$, not what does not get matched.

Here *similarity* refers specifically to "reasoning based on similarity matching". These above three considerations lead to the following formula:

[9] Here similarity from $A$ to $B$ means that, when there is no direct knowledge about $A$ available, then we will go to $B$, which is similar to $A$, to find plausible but potentially fallible answers.

$$(A \sim B) = \frac{f_1(|F_A \cap F_B|)}{f_2(|F_B|)}$$

where $f_1$ and $f_2$ can be any monotonic increasing functions, including identity functions. Therefore, the similarity measure we used in CONSYDERR is a special case of it (cf. [41]):

$$(A \sim B) = \frac{|F_A \cap F_B|}{|F_B|}.$$

Another consideration is whether similarity should be transitive, counter-transitive, or neither: that is, if $A$ is similar to $B$ and $B$ is similar to C, should $A$ be similar to $C$? There are two possibilities: (1) There is something in common to all three $A$, $B$ and $C$. In this case, $A$, $B$ and $C$ are pair-wise similar. So $A$ *is* similar to $C$. (2) There is something in common between $A$ and $B$ and something else in common between $B$ and $C$. In this case, $A$ is similar to $B$ and $B$ is similar to $C$, but $A$ is *not* similar to $C$. Therefore, similarity should not be made to be either transitive or counter-transitive in implementation (see [57]). The similarity matching in CONSYDERR fits the criterion, since it involves only one top-down/bottom-up cycle (and thus one similarity matching) at a time. Similarly, since $A \sim B$ is a numerical measure, which is not necessarily commutative, i.e. $A \sim B \neq B \sim A$, similarity measures should not be made symmetric [57].[10] Again, the similarity measure in CONSYDERR as shown above fits the criterion.

### 6.2. *Some other considerations*

We will identify some considerations necessary for further evaluations.

#### 6.2.1. *Mixed rules and similarities*

We need to deal with mixed rule application and similarity matching situations. For example, in one case, suppose $A \sim B$, $B \longrightarrow C$; we want to make sure that if $A$ is activated, then the activation of $B$ is $A * (A \sim B)$, and the activation of $C$ is $A * (A \sim B) * (B \longrightarrow C)$. One instance of this is the following: Western Texas is similar to Chaco, Chaco is a cattle country, so Western Texas is probably a cattle country (this is similar to case-based reasoning in a way). Other cases include:

- $A \longrightarrow B$, $B \sim C$;
- $A \longrightarrow B$, $B \sim C$, $C \longrightarrow D$;
- $A \longrightarrow B$, $B \longrightarrow C$, $C \sim D$;
- $A \longrightarrow B$, $B \longrightarrow C$, $C \longrightarrow D$;

and other various combinations. In all of these cases, we want an intuitively correct result analogous to the first case. These cases will be discussed in the following subsections.

[10] It should be noted that there is really no context-free, universally applicable similarity measures. The above considerations are generalized from the examples and are suitable for some large classes of problems, including all the tasks we apply the measure to, but are not claimed to be universally correct. There are, however, generic forms of similarity measures [57] that can be incorporated into the architecture to make it generic.

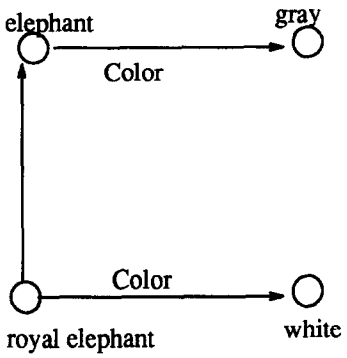



Fig. 4. Inheritance graph.

*6.2.2. Inheritance scenarios*

We need to deal with inheritance and cancellation identified earlier, which is an important aspect of commonsense reasoning earlier. The reason for giving special consideration to this aspect, which seems to be just a mixture of rule application and similarity matching, is that inheritance and cancellation involves competition of mutually conflicting concepts (i.e. property values), which gives rise to some subtle requirements regarding system parameters (for general discussions of inheritance, see [55]).

The inheritance problem can be formulated as inheritance of properties, that is, we express an inheritance problem in an "inheritance graph", which is a directed acyclic graph in which nodes represent concepts and two types of links, *is-a* and *has-property-value*, are used to connect pairs of nodes. See Fig. 4 for an example. This is slightly different from the problem formulation in [55] (for details, see [48]). This "inheritance graph" will be expressed as rules and similarity (and eventually transformed into the CONSYDERR representation), to solve the problem in CONSYDERR. For example, the situation in Fig. 4 can be expressed as rules

$$elephant \longrightarrow color\text{-}gray,$$

$$royal\text{-}elephant \longrightarrow color\text{-}white$$

and similarity

$$royal\text{-}elephant \sim elephant,$$

*is-a* links (such as in "royal elephants are elephants") are implemented implicitly through the containment relations between the corresponding feature sets of the two concepts or propositions; that is, the feature set of "royal elephant" is a superset of the feature set of "elephant", because "elephant" is a more general concept and hence has less features (less specificity) associated with it; "royal elephant" is a subclass of "elephant" and hence contain all the features of "elephant" plus some others that are unique to it. The other type of links (*has-property-value*) are implemented explicitly as rules as shown above, in which LHS of a rule is a concept node and RHS is a property-value pair (for example, $elephant \longrightarrow color\text{-}gray$). (We treat a property-value pair as a concept in CONSYDERR.)

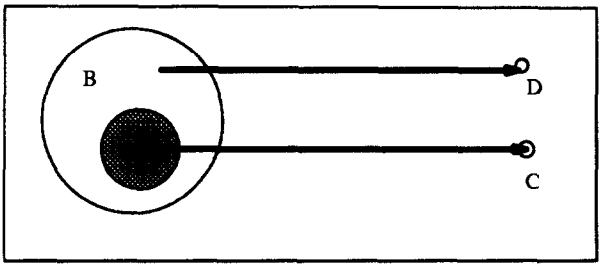



Fig. 5. Inheritance case I. The interrelation of the feature sets in CD.

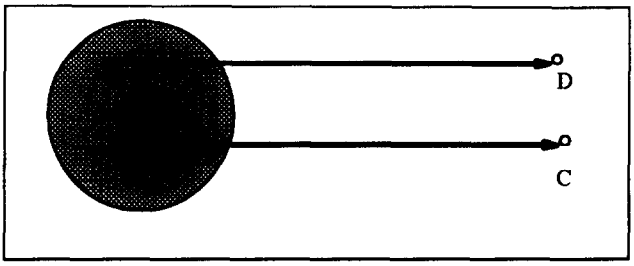



Fig. 6. Inheritance case II. The interrelation of the feature sets in CD.

Let $A$ be a superclass of $B$, therefore the feature set of $A$ is a subset of that of $B$ (i.e. $F_A \subset F_B$). Suppose $A$ has a property value C, and $B$ has no specified property value. If $B$ is activated, then $C$ should be activated to a certain extent, from (top-down) inheritance (remember $A$ is a superclass of $B$). On the other hand, supposing $B$ has a property value $D$ and $A$ has no specified property value; if $A$ is activated, $D$ should be activated too, from the percolation of property values (bottom-up inheritance) from $B$, a subset of A, to A.

What is most difficult to deal with is the cancellation of property inheritance. As in Fig. 5, $A$ has a property value $C$ and $B$ has a property value $D \neq C$. If $A$ is activated, $C$ should win over $D$ (remember that $A$ is a superclass of $B$, and therefore the feature set of $A$ is a subset of that of $B$). Conversely, as in Fig. 6, $A$ has a property value $C$ and $B$ has a property value $D \neq C$. If $B$ is activated, $D$ should win over $C$. These cases of inheritance will be handled in the following subsections.

### 6.3. *Essential properties—explaining the patterns*

We are now ready to see how CONSYDERR fulfill the requirement regarding the seven patterns identified in Section 3.2.2 (though only simple cases will be handled here).

(1) Given inexact information for a concept $A$, a node representing the concept can be partially activated, to a degree commensurate with its inexactness. Suppose there is a rule in the form of

$$A \longrightarrow B$$

and the weight on the link between $A$ and $B$ is $w_{AB}$. Due to the rule application, the activation of $B$, according to the CONSYDERR equations, is as follows:
From CD

$$B = A * \frac{|F_B|}{g(|F_B|)} * \frac{|F_A| * w_{AB}}{f(|F_A|)}.$$

From CL

$$B = A * w_{AB}.$$

So overall,

$$B \approx A * w_{AB}$$

i.e.

$$B = A * (A \longrightarrow B).$$

When there are multiple conditions in the rule, such as

$$A\ B\ C \longrightarrow D$$

with weights $w_{AD}, w_{BD}$, and $w_{CD}$, the activation of the conclusion $D$ is

$$D = A * w_{AD} + B * w_{BD} + D * w_{CD}.$$

Or

$$D = (A\ B\ C) * (A\ B\ C \longrightarrow D)$$

given the partial activations (inexact and uncertain information) of $A$, $B$, and $C$.

(2) When the given information is incomplete, i.e. not all conditions in a rule are known, the above weighted-sum computation can still be applied to reach a (weaker) conclusion. Suppose

$$A\ B\ C \longrightarrow D$$

with weights $w_{AD}, w_{BD}$, and $w_{CD}$, and $B$ and $C$ are known (e.g. $B = 1$ and $C = 1$), but $A$ is unknown ($A = 0$). The conclusion $D$ can still be reached, according to the CONSYDERR equations:

$$\begin{aligned} D &= A * w_{AD} + B * w_{BD} + D * w_{CD} \\ &= w_{BD} * B + w_{CD} * D \\ &= w_{BD} + w_{CD} \end{aligned}$$

i.e.

$$\begin{aligned} D &= (A\ B\ C) * (A\ B\ C \longrightarrow D) \\ &= (0\ 1\ 1) * (A\ B\ C \longrightarrow D). \end{aligned}$$

The situation in which the known conditions (such as $B$ and $C$ above) are inexact can be handled similarly, combining this method with the previous.

(3) When there is no rule applicable (in case of novel input), similarity matching is utilized to reach a conclusion. Given $A$, and the knowledge

$$A \sim B,$$

$$B \longrightarrow C,$$

that is, $A$ and $B$ share features (in CD) and there is a rule from $B$ to $C$ (in CL and CD), we apply the equations: from CD

$$B = A * \frac{|F_A \cap F_B|}{g(|F_A|)}$$
$$\approx A * s_{AB},$$
$$C = B * \frac{|F_B|}{f(|F_B|)} * w_{BC},$$

that is,

$$C \approx B * w_{BC} \approx A * s_{AB} * w_{BC}.$$

So

$$C = A * (A \sim B) * (B \longrightarrow C).$$

Other cases can be handled similarly, such as when $A$ is similar to a number of other concepts and those concepts are linked to many other concepts via rules (handled by straightforward extensions of the above method), or when the rules used have a number of other (unknown) conditions (handled by incorporating the previous method).

(4) Top-down inheritance can be described as

$$A \sim B,$$

$$B \longrightarrow C,$$

in which $A$ is a subclass of $B$ (so $F_A \supset F_B$), and $C$ is a property-value (such as *color-red*) of $B$. It is obvious that this is a special case of point 3, so it can be handled exactly the same way. We should note, however, in this case

$$s_{AB} = \frac{|F_A \cap F_B|}{g(|F_B|)} = \frac{|F_B|}{g(|F_B|)} \approx 1,$$

because $F_A \supset F_B$. And if we assume $w_{BC} = 1$ (the same for all property-value rules), the resulting activation of $C$ (representing the confidence that $A$ has property-value $C$) is:

$$C \approx B * w_{BC} \approx A * s_{AB} * w_{BC} \approx A.$$

So

$$C = A * (A \sim B) * (B \longrightarrow C) = A.$$

(5) The bottom-up inheritance is similar:

$$B \sim A,$$

$$A \longrightarrow D,$$

in which $B$ is a superclass of $A$ (so $F_B \subset F_A$), and $D$ is a property-value of A. This is also a special case of Case 3, so it can be handled the same way. In this case, we have

$$s_{BA} = \frac{|F_B \cap F_A|}{g(|F_A|)} \approx \frac{|F_B|}{|F_A|} < 1,$$

because $F_B \subset F_A$. If we assume $w_{AD} = 1$ (the same for all property-value rules),

$$D \approx A * w_{AD} \approx B * s_{BA} * w_{AD} < B,$$

which represents the partial confidence based on the evidence from the subclass $B$. That is to say,

$$D = B * (B \sim A) * (A \longrightarrow D) < B.$$

(6) Cancellation of top-down inheritance is as follows: suppose $A$ is a subclass of $B$, $A$ has a property value $D$, and $B$ has a property value $C \neq D$ (assuming $C$ and $D$ have one feature node each). Then

$$A \sim B,$$

$$B \longrightarrow C,$$

$$A \longrightarrow D.$$

When $A$ is activated, $D$ should be activated more than $C$ to cancel the inherited property-value $C$. According to the equations in Section 4.1, assuming $w_{AD} = w_{BC} = 1$,

$$\begin{aligned} D &= A * |F_A| \frac{A_{AD}}{f(|F_A|)} \\ &= A * \frac{|F_A|}{f(|F_A|)}, \end{aligned}$$

$$\begin{aligned} C &= A * |F_A \cap F_B| \frac{w_{BC}}{f(|F_B|)} \\ &= A * w_{BC} * \frac{|F_A \cap F_B|}{f(|F_B|)} \\ &= A * w_{BC} * \frac{|F_B|}{f(|F_B|)} \\ &= A * \frac{|F_B|}{f(|F_B|)} \end{aligned}$$

where $F_A \supset F_B$ and $|F_A| > |F_B|$. Since $f$ is a monotonic increasing function, slower than linear, $D > C$ (by a small margin).

(7) Cancellation of bottom-up inheritance is similar: suppose $B$ is a superclass of $A$, $B$ has a property value $C$, and $A$ has a property value $D \neq C$. Then

$$B \sim A,$$

$$A \longrightarrow D,$$

$$B \longrightarrow C.$$

When $B$ is activated, $C$ should be activated more strongly than $D$. According to the equations in Section 4.1, assuming again $w_{AD} = w_{BC} = 1$,

$$\begin{aligned} C &= B * |F_B| \frac{w_{BC}}{f(|F_B|)} \\ &= B * \frac{|F_B|}{f(|F_B|)}, \end{aligned}$$

$$\begin{aligned} D &= B * |F_B \cap F_A| \frac{w_{AD}}{f(|F_A|)} \\ &= B * w_{AD} * \frac{|F_B \cap F_A|}{f(|F_A|)} \\ &= B * w_{AD} * \frac{|F_B|}{f(|F_A|)} \\ &= B * \frac{|F_B|}{f(|F_A|)} \end{aligned}$$

where $F_B \subset F_A$ and $|F_B| < |F_A|$, and $f$ is slower than but close to linear. So we have $C > D$.

(8) Rule interaction can be described as

$$A \longrightarrow C,$$

$$B \longrightarrow D,$$

$$C \sim D.$$

When $A$ and $B$ both are activated, the interaction of $C$ and $D$ might result in something else being strongly activated. Assuming $F_E = F_C \cup F_D$ and $F_C \cap F_D = \emptyset$,

$$E = |F_E - F_D| \frac{1}{g(|F_E|)} C + |F_D| \frac{1}{g(|F_E|)} D.$$

If the activations $C = D$,

$$E = D(|F_E - F_D| + |F_D|) \frac{1}{g(|F_E|)}$$

$$= D * |F_E| * \frac{1}{g(|F_E|)}.$$

So we have activations $E > D$ and $E > C$. Other cases can be dealt with similarly, such as $F_E = F_C \cap F_D$ or $F_E = F_C - F_D$.

### 6.4. Some further properties

#### 6.4.1. More on mixed rules and similarities

As mentioned in Section 6.2, the following cases should also be dealt with correctly:

(1) For $A \longrightarrow B$, $B \sim C$, if $A$ is activated, then according to the CONSYDERR equations,

$$\begin{aligned} C &= \sum_{F_A} A \frac{w_{AB}}{f(F_A)} \sum_{F_B \cap F_C} \frac{1}{g(F_C)} \\ &= A * w_{AB} * \frac{|F_A|}{f(F_A)} \frac{|F_B \cap F_C|}{g(F_C)} \\ &\approx A * w_{AB} * s_{BC} \end{aligned}$$

that is,

$$C = A * (A \longrightarrow B) * (B \sim C).$$

(2) For $A \longrightarrow B$, $B \longrightarrow C$, if $A$ is activated, then according to the CONSYDERR equations,

$$\begin{aligned} C &= \sum_{F_A} A \frac{w_{AB}}{f(F_A)} \sum_{F_B} \frac{w_{BC}}{f(F_B)} \sum_{F_C} \frac{1}{g(F_C)} \\ &\approx A * w_{AB} * w_{BC} \end{aligned}$$

that is,

$$C = A * (A \longrightarrow B) * (B \longrightarrow C).$$

(3) For $A \longrightarrow B$, $B \longrightarrow C$, $C \longrightarrow D$, if $A$ is activated, then according to the CONSYDERR equations,

$$\begin{aligned} D &= \sum_{F_A} A \frac{w_{AB}}{f(F_A)} \sum_{F_B} \frac{w_{BC}}{f(F_B)} \sum_{F_C} \frac{w_{CD}}{f(F_C)} \sum_{F_D} \frac{1}{g(F_D)} \\ &\approx A * w_{AB} * w_{BC} * w_{CD} \end{aligned}$$

that is,

$$D = A * (A \longrightarrow B) * (B \longrightarrow C) * (C \longrightarrow D).$$

(4) For $A \sim B$, $B \longrightarrow C$, $C \longrightarrow D$, if $A$ is activated, then

$$D = \sum_{F_A \cap F_B} A \frac{w_{BC}}{f(F_B)} \sum_{F_C} \frac{w_{CD}}{f(F_C)} \sum_{F_D} \frac{1}{g(F_D)}$$
$$\approx A * s_{AB} * w_{BC} * w_{CD}$$

that is,

$$D = A * (A \sim B) * (B \longrightarrow C) * (C \longrightarrow D).$$

(5) For $A \longrightarrow B$, $B \sim C$, $C \longrightarrow D$, if $A$ is activated, then

$$D = \sum_{F_A} \frac{w_{AB}}{f(F_A)} \sum_{F_B \cap F_C} \frac{w_{CD}}{f(F_B)} \sum_{F_D} \frac{1}{g(F_D)}$$
$$\approx A * w_{AB} * s_{BC} * w_{CD}$$

that is,

$$D = A * (A \longrightarrow B) * (B \sim C) * (C \longrightarrow D).$$

(6) For $A \longrightarrow B$, $B \longrightarrow C$, $C \sim D$, if $A$ is activated, then

$$D = \sum_{F_A} \frac{w_{AB}}{f(F_A)} \sum_{F_B} \frac{w_{BC}}{f(F_B)} \sum_{F_C \cap F_D} \frac{1}{g(F_D)}$$
$$\approx A * w_{AB} * w_{BC} * s_{CD}$$

that is,

$$D = A * (A \longrightarrow B) * (B \longrightarrow C) * (C \sim D).$$

(7) For $A \sim B$, $B \longrightarrow C$, $C \sim D$, if $A$ is activated, then

$$D = \sum_{F_A \cap F_B} \frac{w_{BC}}{f(F_B)} \sum_{F_C \cap F_D} \frac{1}{g(F_D)}$$
$$\approx A * s_{AB} * w_{BC} * s_{CD}$$

that is,

$$D = A * (A \sim B) * (B \longrightarrow C) * (C \sim D).$$

In all of the above cases, the network dynamics results in the correct node activation for each node involved, according to the appropriate similarity measures and rule application requirements. Although we discussed rules with only a single premise, rules with multiple premises are straightforward extensions of these cases (see [50]).

It is easy to notice there is no case where there are two or more consecutive similarity relations. This is because we do not want similarity to propagate. As mentioned before, CONSYDERR satisfies this requirement by allowing one cycle only in similarity matching.

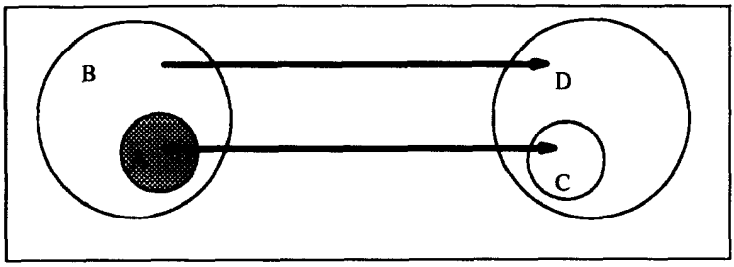



Fig. 7. Inheritance case III.

*6.4.2. More on inheritance*

There are the following two cases of inheritance (from [55]) that are more complicated but need to be handled correctly. As before, assume $A$ is a superclass of $B$, and thus $F_A \subset F_B$. Suppose $A$ has a property value $C$ and $B$ has a property value $D$ that is a subclass of $C$ (and therefore $F_D \supset F_C$). If $A$ is activated, $C$ should win over $D$. See Fig. 7. For example, $A$ is a particular region (say, the South), $B$ is a subarea of $A$ (say, the Southeast), $C$ is a fruit-growing area, and $D$ is an orange-growing area. So if $A$ is given, $C$ should have a higher confidence value than $D$.

In CONSYDERR, when $A$ is activated,

$$C = |F_C| * \frac{1}{g(|F_C|)} * |F_A| * \frac{1}{f(|F_A|)} * A$$
$$\approx A$$

and

$$D = A * \left(\frac{|F_C|}{g(|F_D|)} + \frac{|F_D - F_C|}{g(|F_D|)} \frac{|F_A|}{f(|F_B|)}\right)$$
$$< A$$

because $|F_C|/g(|F_D|) < 1$, $|F_D - F_C|/g(|F_D|) < 1$, and $|F_A|/f(|F_B|) < 1$. So $C > D$.

In the other case, as shown in Fig. 8, suppose $A$ has a property value $C$ and $B$ has a property value $D$ that is a subclass of $C$ (and therefore $F_D \supset F_C$). If $B$ is activated, $D$ should win over $C$. For example, $A$ is a particular region (say, the South), $B$ is a subarea of $A$ (say, the Southeast), $C$ is a fruit-growing area, and $D$ is an orange-growing area. So if $B$ is given, $D$ should have a higher confidence value than $C$.

In CONSYDERR,

$$C = |F_C| * \frac{1}{g(|F_C|)} * |F_B| * B * \frac{1}{f(|F_B|)}$$

and

$$D = |F_D| * \frac{1}{g(|F_D|)} * |F_B| * B * \frac{1}{f(|F_B|)}.$$

So we have $D > C$ (since $|F_D| > |F_C|$).

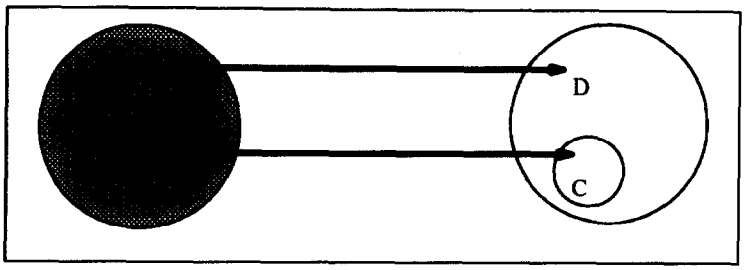



Fig. 8. Inheritance case IV.

### 6.5. *Are the two levels necessary?*

One objection to the CONSYDERR architecture is that CL is not really needed, since all of its work is also done by CD. While this is correct when comparing the two levels in isolation, this objection misses the point that the synergy resulting from the interaction between the two levels can only be generated by having two separate levels. Specifically, the existence of CL serves the following purposes:

- Inheritance is handled through top-down and bottom-up flows, and the bottom-up weight *bu* plays a crucial role in ensuring correct cancellation (see earlier discussions and see Appendix B). Without the two-level structure, we will not be able to handle inheritance with such an efficiency (in constant time; see [48] for detailed derivations; see Appendix B for experiments).
- In order to extract conceptual information from the distributed representation in CD, we need the correspondence between a concept and its features to be explicitly represented; the CL/CD structure is an economical way of accomplishing this.
- In order to understand/explain rule structures, an explicit representation of rule structures as in CL is necessary, since the corresponding rule representation in CD is distributed and thus does not help with the comprehensibility of rules (which is important for conceptual-level reasoning).

## 7. Discussions

### 7.1. *Some further extensions*

A simple extension is adding a thresholding mechanism to the inter-level interaction, the same as in the intra-level case, so that a CD node (or a CL node) is activated during the top-down (or bottom-up) phase, only if the top-down (or bottom-up) activations received exceed a threshold. This serves to make the inter-level interaction non-linear.

Note that the similarity measure used can be easily generalized by adding different weights, functions, and/or thresholds to the inter-level links and the nodes involved as described in Section 4.3.1; this can be useful for more complex situations that require more complex similarity measures (e.g. non-linear combinations of features). An intermediate level of "hidden" nodes can also be added to provide more complex mapping capabilities.

For further expanding the expressive and reasoning power of the architecture, we add variable binding capabilities (cf. [16,35]). Adding variables to connectionist models is a technically complicated endeavor and many issues need to be addressed; for example, how are variables represented? how are bindings passed along? what is the complexity of the solution? what is the logical capability of the solution mechanism? These issues have already been addressed in relation to CONSYDERR and the solution is fully covered in a set of papers (see [44,47,49,53]; cf. [1,28] for other solutions to the variable binding problem). The problem requires extensive technical treatments that have only marginal relevance to the theme of the present paper; because of the fact that a variety of solutions exist, this problem is no longer of high importance; considering these factors and the space limitation, I will not repeat the solution to the variable binding problem here.

Yet another issue is the mechanism for taking contexts into consideration [45], especially for context-sensitive relationships involving rules and similarities (such as *Chaco* $\sim$ *Western-Texas* in terms of *cattle-raising*; cf. [10]). The mechanism in CONSYDERR [49] is a feedforward network that takes current contexts (such as a query as in the earlier examples) as input and produces two types of modulation signals for modulating feature nodes: *enable* and *disable*. The disabled feature nodes will have activations equal to 0 and therefore will not participate in similarity matching. The actual disabling occurs at the inter-level links connecting these feature nodes to their corresponding concept nodes; so the inter-level weights are adjusted accordingly, e.g. from $1/g(|F_A|)$ to $1/g(|F'_A|)$, where $F'_A$ is a reduced feature set with some feature nodes disabled. In this way, similarity measures take contexts into account and produce more accurate results, e.g. comparing Chaco and Western-Texas with respect to features relevant to cattle-raising. See Fig. 9 (the full explanation is in [49]). This feedforward network can be structured in a way similar to CL; that is, each link can represent some *context rules*, which decide if a feature is relevant in a given context. More complex structures, such as a backpropagation network, are also permissible. [11]

### 7.2. Brittleness and robustness

The term *brittleness* has been around for quite a while for describing some fundamental flaws of existing rule-based approaches [24,59]. Though different authors have ascribed somewhat different meanings to the word, basically, "brittleness" (the opposite of robustness) suggests being easily broken: the slightest deviation in inputs from what is exactly known by a system can cause a complete breakdown of the system. Specifically, it can be qualified as the inability of a system to deal, in a systematic way within a unified framework, with some important aspects in reasoning, including the following aspects (which have been identified for a long time):

- partial information,
- uncertain or fuzzy information,

[11] The outstanding issues in this regard include learning of context rules [49], hierarchical structuring of contexts, and more elaborate interactions of contexts and reasoning, all of which must be handled within reasonable time and space constraints.

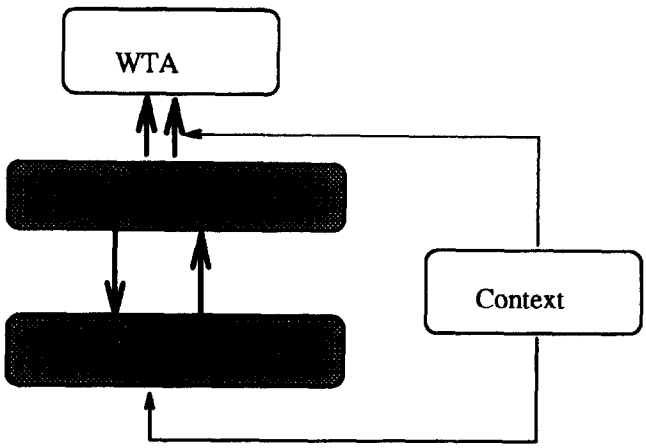



Fig. 9. The overall architecture with feature and result selections.

- similarity matching,
- rule interactions,
- generalization,
- inheritance (i.e. top-down inheritance),
- percolation (i.e. bottom-up inheritance),
- changing contexts and learning new rules.

They overlap substantially with the 7 patterns identified in Section 3.2.1, which are exemplified by the 11 examples.

The brittleness problem is pervasive. With the exception of some extremely specialized narrow domains, it shows up in all kinds of reasoning in various domains: for example, the brittleness problem exists in *decision-making*: when there are no precisely specified preconditions and actions (e.g. there exist different values, not just missing values), a typical rule-based system cannot proceed without further information regarding the situation or without additional mechanisms (cf. [21]), due to the lack of the aforementioned flexibility in such a system. To circumvent this brittleness problem, a brute force approach can be taken in such a system: every possible scenario of combinations of conditions and decisions is analyzed beforehand and structured into the system, which is not always possible, especially for large systems. The brittleness problem exists also in *diagnosis*, in real world *planning*, and in *control* with expert systems, and many other domains. The difficulty of conventional rule-based reasoning in dealing with all of these commonsense reasoning domains signifies the range and importance of the problem, so it is paramount to address this problem, in order to develop systems capable of robust reasoning.

The idea of robust reasoning is meant to capture the kind of reasoning that occurs in commonsense domains, and both the flexibility and the rigor associated with it; or more precisely, it is meant to be reasoning free from the problem of brittleness. The theory of robust reasoning is able to deal with the requisite flexibility: fuzzy, partial, or incomplete information, inexact knowledge or inexact matches between knowledge in store and situations at hand, generalization from known instances, similarity-based reasoning, and

other kinds of flexibility. Among all the aspects of the brittleness problem, learning new rules is a separate issue (quite different from the representational issues) and should be so dealt with. (Although the study of their interaction would be interesting, it is beyond the scope of this paper to cover them all.) While the remaining aspects of the problem still look like a disparate set of problems, they are all characterized in the present theory as rule-based reasoning coupled with similarity-based reasoning. Thus the present theory provides a promising avenue towards robust reasoning.

The significance lies in the fact that, based on this theory, a connectionist architecture with a simple two-level structure can deal with brittleness (to a certain extent) and solve a certain range of representation and reasoning problems effectively and computationally efficiently in a massively parallel manner; notably, the task is accomplished within a unified framework.[12]

### *7.3. The connection with Collins and Michalski's analysis*

Collins and Michalski [10] present an interesting analysis of patterns in commonsense reasoning. I will not attempt to discuss the theory completely, except to point out some connections to this work. The six categories in [10] are of particular interest:

(1) derivation from mutual implication,
(2) derivation from mutual dependence,
(3) generalization-based transformation,
(4) specialization-based transformation,
(5) similarity-based transformation,
(6) dissimilarity-based transformation.

According to the present analysis, they can be reduced as follows: the first two categories have little difference (computationally) and both can, to a large extent, be dealt with by rule application (with contextual effects handled by the context module as discussed before); the rest of the categories are similarity-related, that is, generalization and specialization are special cases of similarity, and therefore they can be dealt with by similarity matching. Dis-similarity based inference is not covered here.

In Collins and Michalski's theory, the confidence of the conclusions reached depends on a number of parameters: conditional likelihood, degree of certainty, degree of typicality of a subset within a set, degree of similarity of one set to another, frequency of the referent in the domain of the descriptor, dominance of a subset in a set, multiplicity of the referent, and finally, multiplicity of the argument. According to the present analysis, a smaller set of parameters can be identified: rule weights and similarity measures. These two parameters can subsume the above parameters used by Collins and Michalski: the first two, conditional likelihood and degrees of certainty, can be easily captured by rule weights [50]; the rest can be accounted for by similarity measures or a combination of rule weights and similarity measures.

Finally, although the theory of Collins and Michalski [10] is consistent and justified, the existing implementations of it seem to be a juxtaposition of different techniques

[12] However, it is not a complete solution to all the problems within the scope but a solution to a subset of the most common reasoning problems that can be solved with a simple computational mechanism.

within a rule-based framework. A problem with such implementations, I believe, is that they still suffer from the brittleness problem: similarity is handled by rules, and one rule is used for each pair of similar concepts; therefore, there may be too many rules around, because there are too many things similar to any one particular thing, in many different ways and under many different contexts, as pointed out before; such systems will not be free from brittleness in any reasonably large domain, considering the difficulty of this brute force method of putting every pair of similar things into rules. The computational complexity is tremendous, given the number of rules needed. A more integrated mechanism capable of dealing with similarity matching and other types of flexible reasoning in a massively parallel fashion (also cf. [18]) is preferred.

### *7.4. Limitations and further work*

We shall further refine the theory and the CONSYDERR architecture proposed here. There are several aspects worth pursuing further. One aspect that may be of great importance for further development is automatically developing distributed representation, which is currently underway. It has been suggested that this can be done through grounding high-level processes into low-level processes (i.e. symbol grounding [19]), and through mapping syntactic representation into semantic representation by learning algorithms [15]. Features developed in this way (which may not be conceptually interpretable) may better capture the similarity between concepts involved in given situations, and may lead to more accurate models of commonsense reasoning based on similarity matching.

In addition, it is important to study in greater detail the interaction between the feature representation and the more explicit local representation, especially with regard to the exploration of the synergy which interaction generates.

Backward chaining and goal-directed inference are not treated in this paper. Nevertheless, they are important issues and should be addressed in subsequent work.

More experimental work shall be carried out as the next step in this research, which shall include examinations of system dynamics with complex rule structures and feature structures, detailed verifications with quantitative data, and explorations of other representational and reasoning types, such as temporal reasoning, backward chaining, and recursive rule structures.

## 8. Concluding remarks

The theory advanced here is meant to be an *integrated* model that can deal effectively with a (seemingly disparate) set of important problems in commonsense reasoning (i.e. the basic elements of commonsense reasoning): rule application, evidential combination, similarity matching, inheritance in both directions, and so on. The key point is that these problems are handled in a single unified framework that has no special provision for any single one of these problems. Through data analyses, these problems are reduced to a single process; thus a theory of robust reasoning is proposed, combining the rigor and flexibility needed in commonsense reasoning. This theory is carried out by CON-

SYDERR, a connectionist architecture, which thus serves as a unifying mechanism. CONSYDERR integrates rule-based reasoning into connectionist networks and couples localist networks with similarity-based distributed representation.

## Appendix A. FEL rules in CONSYDERR

FEL stands for *Fuzzy Evidential Logic*. Here the word *fuzzy* is used to refer to the gradedness or continuous inexactness of a concept or a proposition, not restricted to Zadeh's definition of fuzziness based on linguistic variables.

**Definition A.1.** A *fact* is a propositional atom or its negation, represented by a letter (with or without a negation symbol) and having a value between $l$ and $u$. The value of an atom is related to the value of its negation by a specific method, so that knowing the value of an atom results in immediately knowing the value of its negation, or vice versa.

**Definition A.2.** A *rule* is a structure composed of two parts: a left-hand side (LHS), which consists of one or more facts, and a right-hand side (RHS), which consists of one fact. When facts in LHS get assigned values, the fact in RHS can be assigned a value according to a *weighting scheme*.

**Definition A.3.** A *weighting scheme* is a way of assigning a weight to each fact in LHS of a rule, with the total weights less than or equal to 1, and of determining the value of the fact in RHS of a rule by thresholded (if thresholds are used) weighted-sum of the values of the facts in LHS (or inner-products of weight vectors and vectors of values of LHS facts). When the range of values is continuous, then the weighted-sum is passed on if its absolute value is greater than the threshold, or 0 if otherwise. When the range of values is binary (or bipolar), then the result will be one or the other depending on whether the weighted-sum (or the absolute value of it) is greater than the threshold or not (usually the result will be 1 if the weighted-sum is greater than the threshold, 0 or $-1$ if otherwise).

**Definition A.4.** A *theory* is a 4-tuple: $< A, R, W, T >$, where $A$ is a set of facts, $R$ is a set of rules, $W$ is a weighting scheme for $R$, and $T$ is a set of thresholds each of which is associated with one element in $R$.

**Definition A.5.** A *Conclusion* in FEL is a value associated with a fact, calculated from rules and facts by doing the following:

- for each rule having that conclusion in its RHS, obtain conclusions of all facts in its LHS (if any fact is unobtainable, assume it to be zero); and then calculate the value of the conclusion in question using the weighting scheme;
- take the MAX of all these values associated with that conclusion calculated from different rules or given in initial input.

**Definition A.6.** A rule set is *hierarchical*, if the graph depicting the rule set is acyclic; the graph is constructed by drawing a unidirectional link from each fact (atom) in LHS of a rule to the fact (atom) in RHS of a rule.

**Definition A.7.** A *fuzzy evidential logic* (FEL) is a 6-tuple: $< A, R, W, T, I, C >$, where $A$ is a set of facts (the values of which are assumed to be zero initially), $R$ is a set of rules, $W$ is a weighting scheme for $R$, $T$ is a set of thresholds each of which is for one rule, $I$ is a set of elements of the form $(f, v)$ (where $f$ is a fact, and $v$ is a value associated with $f$), and $C$ is a procedure for deriving conclusions (i.e. computing values of facts in RHS of a rule in $R$, based on the initial condition $I$).

**Definition A.8.** $FEL_1$ is FEL when the range of values is restricted to between 0 and 1, and the way the value of a fact is related to the value of its negation is:

$$A = 1 - \neg A$$

for any fact $A$.

**Definition A.9.** $FEL_2$ is FEL when the range of values is restricted to between $-1$ and 1, and the way the value of a fact related to the value of its negation is:

$$A = -\neg A$$

for any fact $A$.

This range corresponds to the range of node values in CONSYDERR.

**Definition A.10.** An *element* is a structure that represents one and only one fact and has multiple sites each of which receives a group of links that represents one single rule (i.e. links from facts in LHS of the same rule that has the fact which this element represents in its RHS.

**Definition A.11.** An *implementation* of FEL is a network of elements connected via links, where each node represents an atom and its negation (there is a one-to-one mapping between an atom and a node) and links represent rules, going from nodes representing facts in LHS of a rule to nodes representing facts in RHS of a rule.

**Theorem A.12.** *There is an isomorphic one-to-one mapping between FEL and CL (or CD).*

Proofs can be found in [49].

**Definition A.13.** *Horn clause logic* is a logic in which all formulas are in the forms of

$$P$$

or

$$P_1 P_2 \ldots P_n \longrightarrow Q$$

where $P$, the $P_i$'s and $Q$ are propositions.

**Definition A.14.** A *binary FEL* is a FEL (either $FEL_1$ or $FEL_2$) in which values associated with facts are binary (or bipolar), total weights of each rule sum to 1, and all thresholds are set to 1.

**Theorem A.15.** *The binary FEL is sound and complete with respect to Horn clause logic.*

**Proof.** The inference rule for FEL can be defined as a variant of forward chaining (according to the definition of *conclusion*). Let $K$ be a set of FEL facts and their values, in the form of pairs $(f, v)$, where $f$ is a fact and $v$ is its value, a real number between $-1$ and 1 representing the confidence that $f$ is true. Assume that all facts are uniquely represented in $K$ (though their confidence values may be zero). The inference rule will simply add FEL facts to $K$ until no new ones can be added:

(1) Given the FEL rule: $A_1 \ldots A_r \longrightarrow B$ $(w_1, \ldots, w_r)$. If $(A_1, v_1), \ldots, (A_r, v_r)$ are in $K$, then let $v' = w_1 * v_1 + \cdots + w_r * v_r$ and assume $(B, v)$ be in $K$. If $|v'| \geqslant \theta$ and $|v'| > |v|$, then replace $(B, v)$ by $(B, v')$.

(2) Given the FEL fact: $(B, 1)$ (the known fact), we simply replace $(B, v)$ in $K$ by $(B, 1)$.

In case of binary FEL, we have $\theta = 1$. Also, if we know that the value of $B$ is 1, then we know that the value of $\neg B$ is $-1$, vice versa.

Given a Horn clause theory $H$, one can produce a corresponding binary FEL theory $F$ as follows: each Horn clause $A_1, \ldots, A_r \longrightarrow B$ is transformed into a FEL rule by associating a weight $w_i$ with each fact $A_i$, in such a way that $w_i > 0$ and their sum is 1. Conversely, given a binary FEL theory $F$, one can also produce a corresponding Horn clause theory $H$ as follows: each FEL rule $A_1, \ldots, A_r \longrightarrow B$ $(w_1, w_2, \ldots, w_r)$ is transformed into a Horn rule directly, by ignoring weights $w_i$'s, since according to the definition of binary FEL, we always have $w_i > 0$ and their sum is 1.

Assume that all facts in $K$ initially have values 0. Then $(B, 1)$ is a FEL conclusion of $F$ iff $b$ is a logical consequence of $H$. To see this, observe that a FEL fact $(B, 1)$ is added to $K$ if and only if all of the facts in the LHS of the rule have values 1. Thus, $(B, 1)$ is only introduced to $K$ if there exists facts $(A_1,1), \ldots, (A_r,1)$ in $K$. It follows that the FEL inference rule behaves exactly like the forward chaining operator for Horn clause logic when we restrict our attention to facts of value 1. $B$ can be inferred by forward chaining in $H$, iff $(B, 1)$ can be inferred by the FEL inference rule. Therefore, because the Horn clause forward-chaining inference rule is sound and complete, binary FEL is sound and complete with respect to Horn clause Logic. □

**Definition A.16.** A *causal theory* is a set of formulas of the following form

$$\wedge_i \Box\, n_i A_i \wedge_j \Diamond n_j B_j \longrightarrow \Box C$$

where $n_i$'s are either $\neg$ or nothing, $\wedge$ and $\Diamond$ are two modal operators, $n_i A_i$'s are necessary conditions (causes), and $n_j B_i$'s are possible conditions (enabling conditions). $C$ is concluded iff all $n_i A_i$'s are true and all $n_j B_j$'s are not known to be false.

The basic idea is that not all conditions are the same: some conditions (necessary conditions, with $\Box$) are more important than the others, because they are the main factors in determining the causal outcome, while other conditions (possible conditions, with $\diamond$) can be assumed true, since they usually hold. According to the logic, as long as we know that all necessary conditions are true, and no possible conditions are known to be false, then we deduce the conclusion (tentatively).[13]

We need to find a scheme that can equate FEL to Causal Theory. First we need to find a mapping that links truth values in Causal Theory to numerical values in FEL, then we can proceed to find a weighting scheme to enable FEL to simulate Causal Theory. To find a full correspondence between FEL and Causal Theory, we also need a proof procedure in FEL that directly corresponds to a proof procedure in CT and thus enables the derivation of all true formulas (theorems). When all these details are taken care of (see [49]), we have:

**Theorem A.17.** *For every hierarchical, non-temporal Causal Theory, there is a FEL such that CT: $C \models A$ iff FEL: $C' \models$ 'A = 1', where $\models$ denotes derivability, C is a set of initially known conditions for Causal Theory CT, and $C'$ is the initial condition for FEL mapped over from C in CT.*

## Appendix B. Experiments with CONSYDERR

### *B.1. GIRO*

We construct and study the system GIRO (*Geographical Information Reasoning and Organization*), which stores fairly large amount of knowledge.

The knowledge representation in GIRO utilizes the two-level idea in CONSYDERR by dividing the geographical knowledge represented in the system into two categories: concepts, which include basic geographic regions and regional characterizations (such as "cattle-country"), and features, which include primitive physical descriptions of regions, such as "highland", "mountainous", and "tropical". Concepts are represented in CL, and features are represented in CD. Each geographic region represented in CL is connected to its corresponding features in CD, and because of the fact that features are shared by similar concepts, the CD representation is similarity-based. Each region is connected to other concepts in CL by links, if this knowledge is available to the system (e.g. *Western-Texas* $\longrightarrow$ *cattle-country*).

Since in Collins' experiments [10] there was no systematic determination of the subjects' knowledge, we have to find other means of obtaining knowledge systematically. The large amount of knowledge stored in the system is extracted entirely from encyclopedias [49], The knowledge extracted is in the form of (1) regions, (2) their agricultural characterizations, and (3) their physical features. Each region is linked to

[13] This is a non-temporal version of CT [39]; that is, we strip away all temporal notations and treat the same proposition in the original definition with different temporal intervals as different propositions (and represent them with different letters).

| | |
|---|---|
| "cotton-producing-area" | "soybean-growing-area" |
| "coffee-growing-area" | "rubber-producing-area" |
| "wine-producing-area" | "sheep-country" |
| "potato-growing-area" | "producing-banana" |
| "rubber-producing-area" | "producing-tropical-fruits" |
| "goats-area" | "corn-growing-area" |
| "rice-growing-area" | "sugar-producing-area" |
| "wheat-growing-area" | "fruit-veg-growing-area" |

Fig. B.1. Regional characterizations included in the system.

| | | | | |
|---|---|---|---|---|
| temperate | arctic | woodland | plain | Mediterranean |
| plateau | Mts | coastal-land | lake | tropical |
| lowland | hill | river-valley-basin | swamp | rainforest |
| evergreen | deciduous | highland | upland | sparsely-populated |
| densely-populated | fertile | infertile | flood | prairie |
| dependable-rainfall | scrub | farming | rugged | subtropical |
| rainy | savanna | dry-arid | grassland | desert |

Fig. B.2. Geographical features included in the system.

its agricultural characterizations by rules (in CL), its physical features are used in its distributed representation in CD. (For details of how knowledge is divided into features and concepts, see [49].) Since such knowledge is well-documented in encyclopedias, knowledge extraction is straightforward. The extraction process is systematic, covering all the geographical regions in South America, to make sure the knowledge obtained is not arbitrary (the process can be automated; see [49]). There is *no* arbitrary hand-tuning of weights either—all weights are set at 1. Fig. B.1 lists all the concepts used for agricultural characterization of regions. Fig. B.2 lists all the features used in CD.

The system reasons about agricultural characterizations of regions by means of rules and similarity matching. When there is direct knowledge associated with an area (in the form of rules), it is applied, and a strong conclusion is reached; when there is no such knowledge (i.e. when a novel input is encountered), similarity matching (through feature overlap in CD) is used to apply indirect knowledge (associated with similar concepts) and a plausible conclusion is reached.

To reason about "Brazil-north", which has features such as "tropical rainforest hilly plateau", we start by giving GIRO a query: What is the main agricultural product of "Brazil-north"? which amounts to activating the node representing "Brazil-north". The output from GIRO is as follows:

```
(consyderr 0)

  TITLE:  GEOGRAPHY
  focusing on context AGRICULTURE: remove feature NIL
  setup done
  starting running
  top down
  cl propagating
  cd propagating
  bottom up
```

```
the average activation is 0.1213409896658248
( 2, ''cattle-country'', 0.1249998807907104 )
  ( 10, ''fruit-veg-growing-area'', 0.1249998807907104 )
  ( 12, ''producing-banana'', 0.1249998807907104 )
  ( 13, ''producing-tropical-fruits'', 0.1249998807907104 )
  ( 20, ''rubber-producing-area'', 0.9999990463256836 )
  ( 29, ''c-Peru'', 0.125 )
  ( 32, ''Bolivia-orient-rainforest'', 0.125 )
  ( 40, ''Guyana-pgs'', 0.125 )
  ( 41, ''Guyana-hilly-country-forest'', 0.1666666666666667 )
  ( 42, ''Guyana-hilly-country-savanna'', 0.125 )
  ( 45, ''Brazil-cw'', 0.125 )
  ( 50, ''Brazil-n'', 1 )
  ( 60, ''Columbia-basin'', 0.1666666666666667 )
  ( 61, ''Ecuador-coast'', 0.125 )
  ( 66, ''Suriname-plateau'', 0.125 )
```

The result shows that it is a rubber-producing area for sure (with confidence value equal to 0.999999), and it is similar, to a small extent, to "Guyana-hilly-country-forest" (with confidence value 0.167) and "Colombia-basin" (with confidence value 0.167), and there is a small chance that it produces bananas (with confidence value 0.125) and tropical fruits (with confidence value 0.125), etc.[14] Despite the appearance of mere information retrieval, those conclusions actually result from rule-based and similarity-based reasoning from the top-down, bottom-up, and settling information flows. For example, tracing the reasoning process of the system, we see that "producing-banana" (0.125) above is obtained based on the following reasoning: "Ecuador-coast" produces bananas, and "Brazil-north" is similar to "Ecuador-coast" to a small extent, so there is a small chance that "Brazil-north" might produce bananas. The conclusion of "rubber-producing-area", on the other hand, is obtained based on a straightforward rule application: *Brazil-north* $\longrightarrow$ *rubber-producing-area.* Other outcomes are obtained in similar fashions. See Fig. B.3.

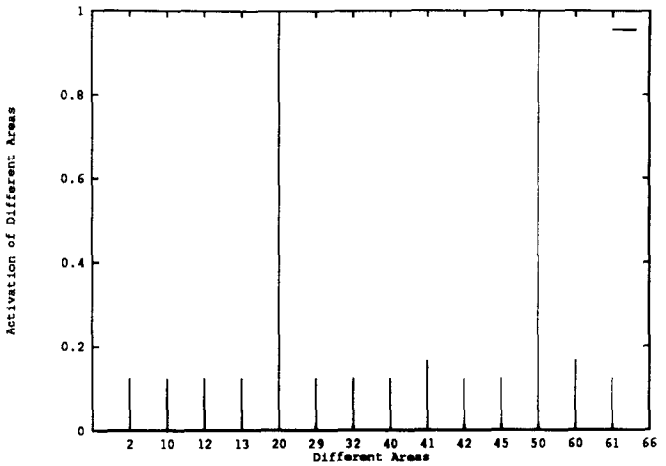
Another example is as follows: we give GIRO a query: What is the main agricultural product of Ecuador coast? by activating the node representing "Ecuador coast". The output from GIRO is as follows:

```
(consyderr 0)

 TITLE:  GEOGRAPHY
 focusing on context AGRICULTURE: remove feature NIL
   setup done
   starting running
   top down
   cl propagating
   cd propagating
   bottom up

  the average activation is 0.1433035089856102
```

[14] If we want to choose one answer out of many, we can simply use a winner-take-all network on top of this, but this is not an intrinsic part of GIRO and is not needed in this case.



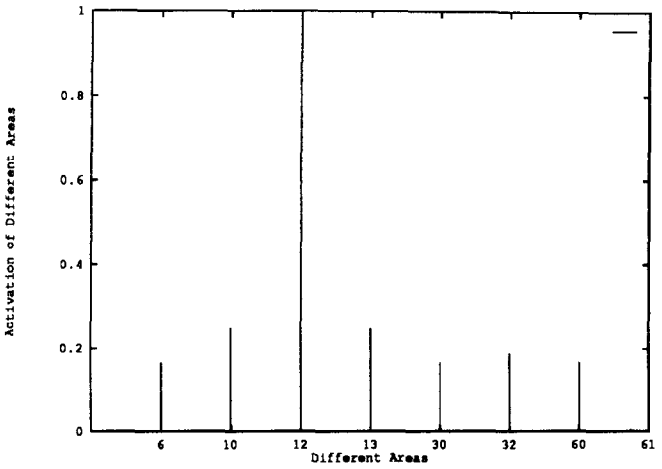

| | | |
|---|---|---|
| 2 | "cattle-country" | 0.1249998807907104 |
| 10 | "fruit-veg-growing-area" | 0.1249998807907104 |
| 12 | "producing-banana" | 0.1249998807907104 |
| 13 | "producing-tropical-fruits" | 0.1249998807907104 |
| 20 | "rubber-producing-area" | 0.9999990463256836 |
| 29 | "c-Peru" | 0.125 |
| 32 | "Bolivia-orient-rainforest" | 0.125 |
| 40 | "Guyana-pgs" | 0.125 |
| 41 | "Guyana-hilly-country-forest" | 0.1666666666666667 |
| 42 | "Guyana-hilly-country-savanna" | 0.125 |
| 45 | "Brazil-cw" | 0.125 |
| 50 | "Brazil-n" | 1 |
| 60 | "Columbia-basin" | 0.1666666666666667 |
| 61 | "Ecuador-coast" | 0.125 |
| 66 | "Suriname-plateau" | 0.125 |

Fig. B.3. Output from GIRO: case 1.

```
( 6, ''Uruguay-coastal'', 0.1666666666666667 )
  ( 10, ''fruit-veg-growing-area'', 0.2499997615814209 )
  ( 12, ''producing-banana'', 0.9999990463256836 )
  ( 13, ''producing-tropical-fruits'', 0.2499997615814209 )
  ( 30, ''e-Peru'', 0.1666666666666667 )
    ( 32, ''Bolivia-orient-rainforest'', 0.1875 )
    ( 60, ''Columbia-basin'', 0.1666666666666667 )
    ( 61, ''Ecuador-coast'', 1 )
```

The result indicates that the area produces banana (with confidence value equal to 0.99999), from applying a rule: *Ecuador-coast* $\longrightarrow$ *producing-banana*. And it is very likely producing tropical fruits and other fruits/vegetables (with confidence value equal to 0.25), inferred from mixed similarity matching and rule application: *Ecuador-coast* $\longrightarrow$ *producing-banana*, and *producing-banana* $\sim$ *producing-tropical-fruits*, so we have *producing-tropical-fruits* as one of the conclusions. It is similar, in some way, to "Uruguay-coastal", "eastern-Peru" and "Columbia-basin", due to similarity matching (or features overlaps) between "Ecuador-coast" and each of these regions. See Fig. B.4.

As yet another example, we give GIRO a query: Does "Brazil-south" produce cattle?




| | | |
|---|---|---|
| 6 | "Uruguay-coastal" | 0.1666666666666667 |
| 10 | "fruit-veg-growing-area" | 0.2499997615814209 |
| 12 | "producing-banana" | 0.9999990463256836 |
| 13 | "producing-tropical-fruits" | 0.2499997615814209 |
| 30 | "e-Peru", | 0.1666666666666667 |
| 32 | "Bolivia-orient-rainforest", | 0.1875 |
| 60 | "Columbia-basin" | 0.1666666666666667 |
| 61 | "Ecuador-coast" | 1 |

Fig. B.4. Output from GIRO: case 2.

by activating the node representing "Brazil-south" and looking for "cattle" in the results. The output from GIRO is as follows:

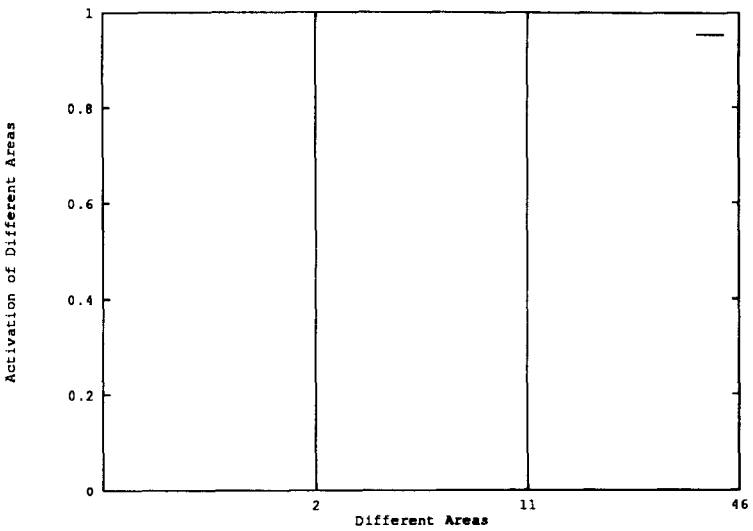
```
(consyderr 0)

  TITLE:  GEOGRAPHY
  focusing on context AGRICULTURE: remove feature NIL
  setup done
  starting running
  top down
  cl propagating''
  cd propagating ''
  bottom up

  the average activation is 0.1492545754568917
  ( 2, ''cattle-country'', 0.9999990463256836 )
    ( 11, ''sheep-country'', 0.9999990463256836 )
    ( 46, ''Brazil-s'', 1 )

    The Result ----
  ( 2, ''cattle-country'', 0.9999990463256836 ) ---end of results
```

The result indicates that the area does produce cattle and sheep (due to respective rules that lead to such results). Nothing else in the network fires strongly or distinguishably in this case. See Fig. B.5.




2 "cattle-country" 0.9999990463256836
11 "sheep-country" 0.9999990463256836
46 "Brazil-s" 1

Fig. B.5. Output from GIRO: case 3.

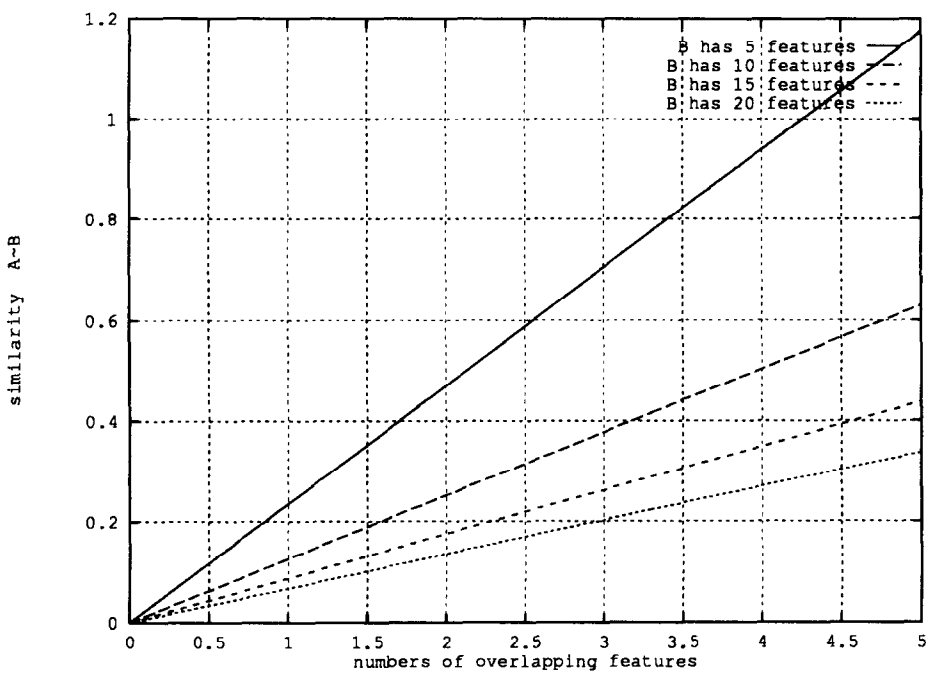



Fig. B.6. Similarity matching in CONSYDERR when $A$ has 20 features and $B$ has small variable numbers of features, where $S_{AB} = |F_A \cap F_B|/|F_B|^{9/10}$.

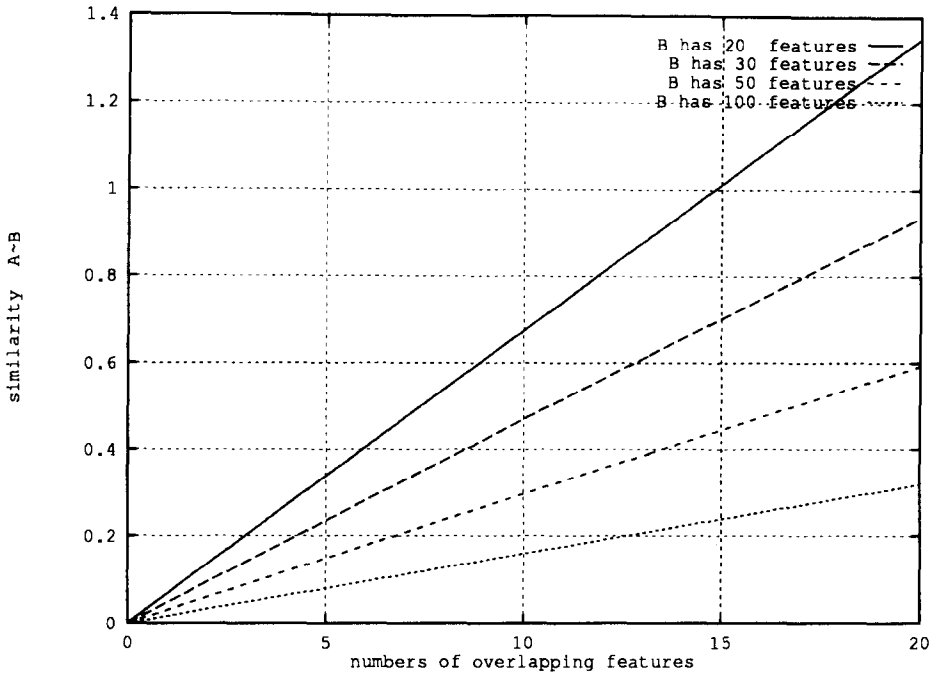



Fig. B.7. Similarity matching in CONSYDERR when $A$ has 20 features and $B$ has large variable numbers of features, where $S_{AB} = |F_A \cap F_B|/|F_B|^{9/10}$.

### *B.2. Systematic experiments*

Systematic experiments help to explore the whole spectrum of the behavior of CONSYDERR when it is applied to a wide range of situations.

#### *B.2.1. Similarity matching*

To test similarity matching (the ability to deal with novel input) in CONSYDERR, we collected the data in Fig. B.6 and Fig. B.7. $A$ is a concept with 20 features in CD. $B$ is another concept, with varying number of features in CD. When $A$ is activated, based on the similarity between $A$ and $B$, $B$ will be activated to a certain degree, that is, $A * s_{AB}$. The two figures demonstrate the difference in the final similarity matching outcome (the activation of $B$) when one or more of the following three parameters change: $|F_A|$ (the number of features for $A$), $|F_B|$ (the number of features for $B$), and $|F_A \cap F_B|$ (the number of overlapping features). Notice that the results show that each curve is almost straight, while in fact a slower than linear (but very close to linear) function is used concerning $|F_B|$ (as an approximation to the linear function). When the number of $|F_B|$ grows large, the addition of features in $B$ has less effect on the similarity matching outcome. The number of $|F_A|$ has basically no effect on the similarity matching outcome.

The figures demonstrate what happens when a novel input is encountered (here $A$ is a novel input and $B$ is an existing concept that shares some features with A). If we assume

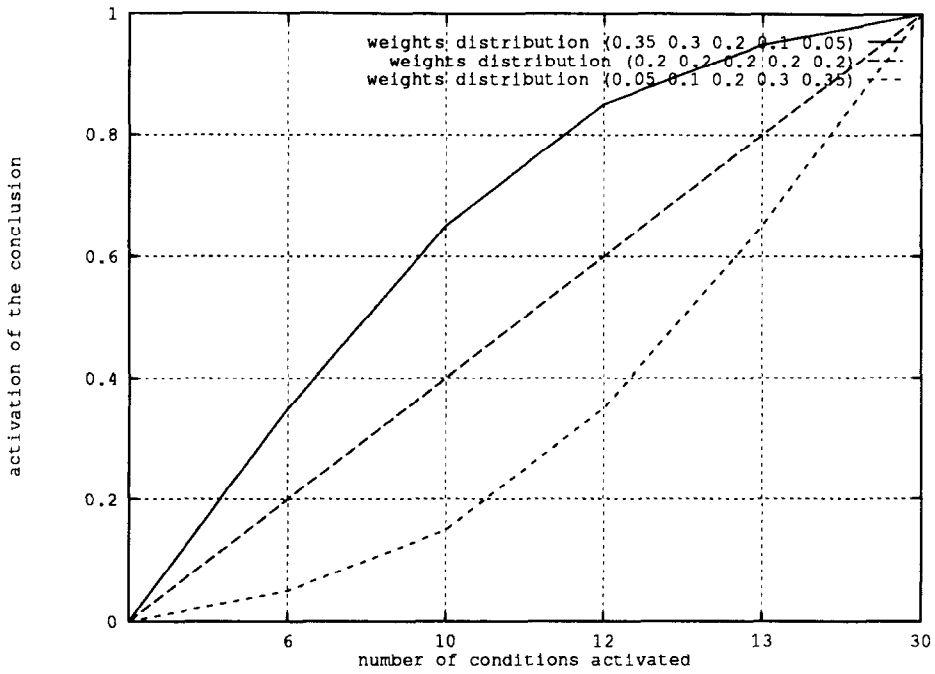



Fig. B.8. Rule application in CONSYDERR. Activations of the conclusion relative to the number of conditions activated, with different weight distributions.

that the novel input $A$ is activated to its full extent (having activation 1), the figures shows the activation of a similar concept $B$, where $B$ has different numbers of features and shares different numbers of features with A. The activation of $B$ (which is similar to the novel input $A$) brings to bear on $A$ the knowledge associated with $B$, so that the system will not break down due to the lack of exactly matching rules as in typical rule-based systems. As in case-based reasoning, when dealing with a novel situation, we utilize existing similar cases and adapt previous solutions to the new situation, although in this example we only deal with very simple and highly abstract cases.

### *B.2.2. Rule application*

To test rule application in CONSYDERR, we collected activation data in Fig. B.8. Note that the figure shows the different activations of the conclusion $B$ with different numbers of conditions, $A_1, A_2, A_3, A_4$, and $A_5$, activated (and to different degrees), in a fixed order but with different weight distributions.

What the figure demonstrates is that rules in CONSYDERR can be activated partially with a partial satisfaction of conditions by input. Depending on the weight distribution, a rule with a certain portion of its conditions activated can partially reach its conclusion, with varying degrees of confidence as shown in the figure as the activations of the conclusion.

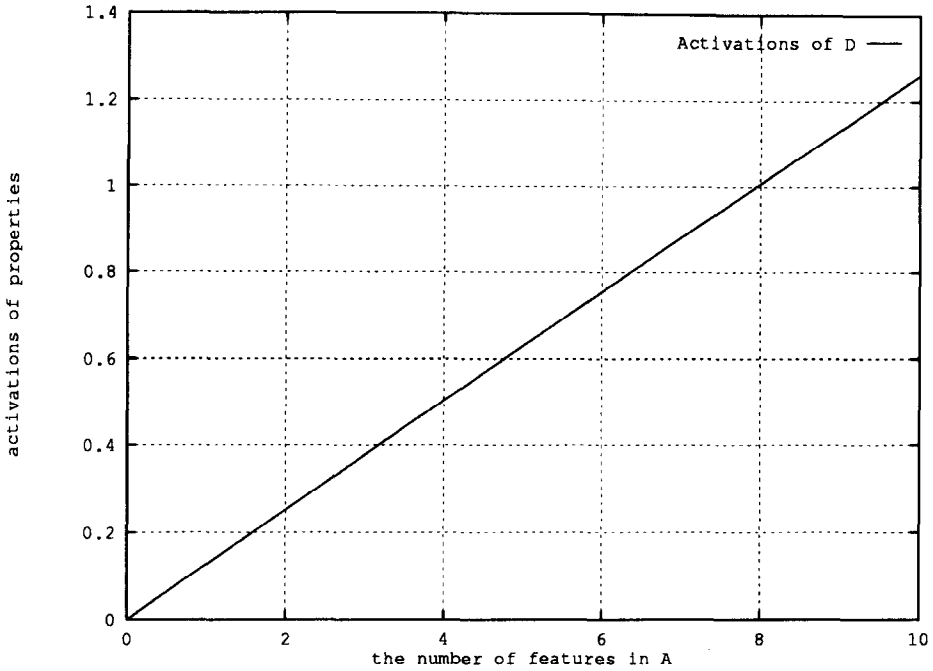



Fig. B.9. Inheritance in CONSYDERR. Activations of property-values relative to the number of features in A, when $A$ is fully activated; where $A \supset B$, $F_A \subset F_B$, $|F_B| = 10$, and $B \longrightarrow D$; the rule weight is equal to 1.

### *B.2.3. Inheritance*

To test inheritance in CONSYDERR, we can examine Figs. B.9, B.10, B.11, and B.12. These figures are for bottom-up inheritance, top-down inheritance, bottom-up inheritance with cancellation, and top-down inheritance with cancellation, respectively. We assume $A \supset B$, $F_A \subset F_B$, $|F_B| = 10$, and either $A \longrightarrow C$ (in case of bottom-up inheritance) or $B \longrightarrow D$ (in case of top-down inheritance) exists, or both (in case of cancellation), with the weights equal to 1. when $A$ is fully activated, the figures show the activations of the property-values ($B$ or $D$, or both). In these figures, we assume that $g(x) = x^{9/10}$ (used in *bu*; so that cancellation will work correctly), and $f(x) = x^{999/1000}$ (used in *lw*). we only look at cases where each property value is represented by a distinct single feature node in CD. Other cases are similar.

These figures demonstrate that CONSYDERR handles inheritance properly. When there is inheritance but no cancellation, the situation is handled the same way as in case of simple similarity matching: the property value of a similar concept is activated. When there is cancellation, the canceled property value always has a weaker activation than the right property value, as guaranteed by the selection of the values of the structural parameters (namely, *td*, *bu*, and *lw*; see [49] for derivations). Note that some activation values may be slightly higher than 1, in order for the comparison of conflicting property values (e.g. *color-red* versus *color-white*) to take place. This is not a problem: a winner-take-all network can be place on top of pairs of conflicting concepts to decide the final activations (0 or 1) for these concepts.

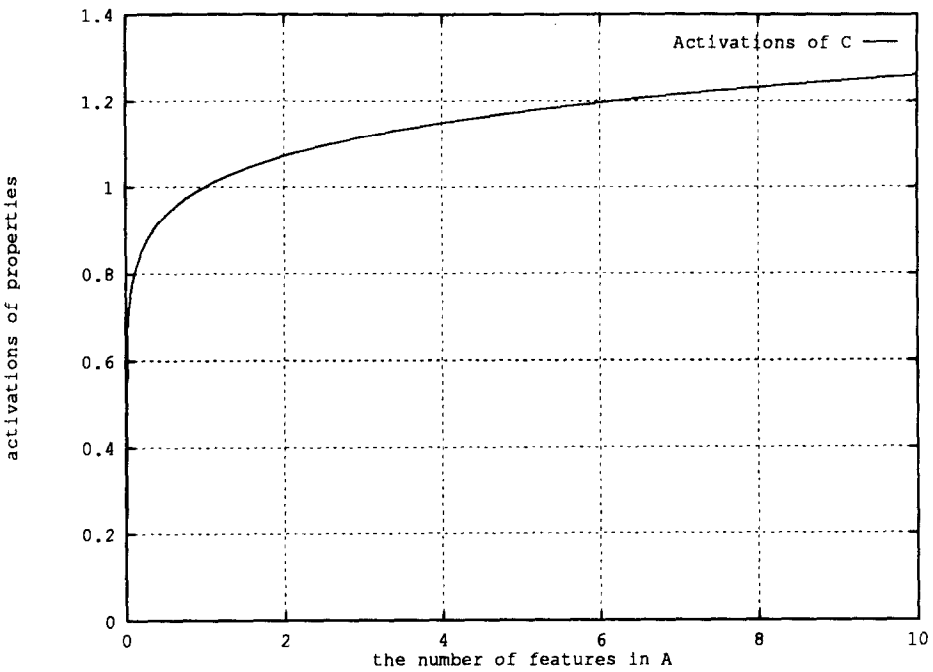



Fig. B.10. Inheritance in CONSYDERR. Activations of property-values relative to the number of features in A, when $B$ is fully activated; where $A \supset B$, $F_A \subset F_B$, $|F_B| = 10$, and $A \longrightarrow C$; the rule weight is equal to 1.

### *B.3. Two-level versus one-level*

The nagging question of whether the two-level structure is necessary can be partially answered experimentally. As argued before, one reason for the two-level dual representation architecture is to enable the correct handling of inheritance and cancellation. Fig. B.13 shows the difference in the handling of inheritance and cancellation. When there is no CL in the architecture (and therefore there is no bottom-up weight $bu$), the activation of a property-value concept is calculated by summing the activations of all the feature nodes (in CD) of a property-value concept, while with CL, the activation of each property-value is taken to be the activation of the corresponding node in CL. The figure demonstrates that, without CL, cancellation cannot be accomplished correctly, since the two competing concepts (property-values $C$ and $D$) will have the same activation (at exactly 1). There is no obvious way of remedying this problem beside having the CL/CD two levels. With the CL/CD two-level structure, due to the bottom-up weight $bu = 1/|F_A|^{9/10}$ (where $F_A$ is the size of the feature set of the corresponding concept), $C$ is always activated less strongly than $D$.

**Acknowledgements**

I wish to thank Dave Waltz, James Pustejovsky, Tim Hickey, and Pattie Maes for many discussions, comments and suggestions. I am grateful to Larry Bookman and

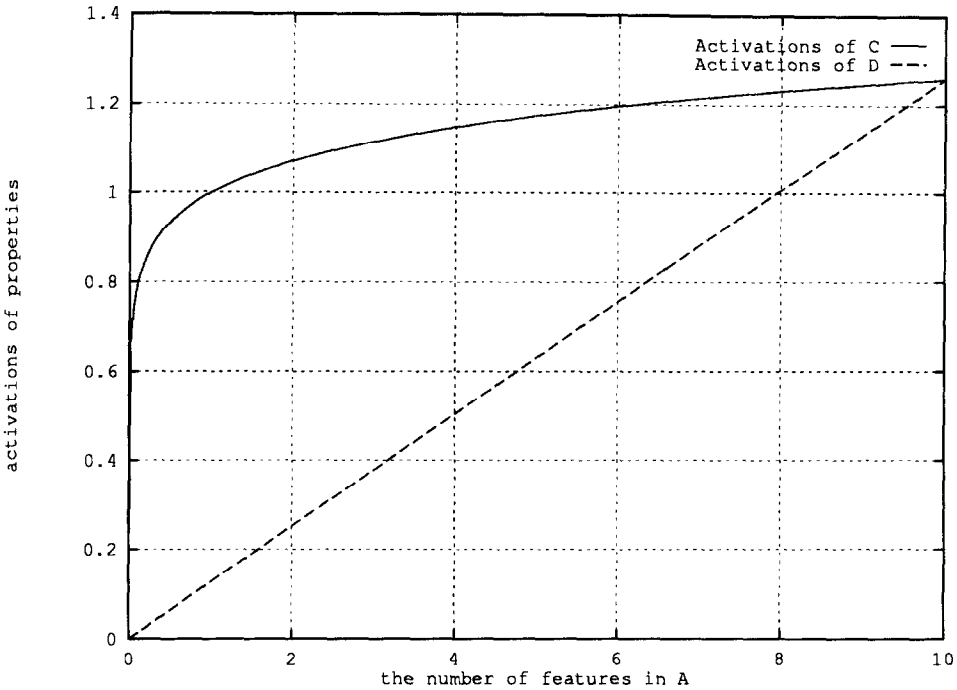



Fig. B.11. Inheritance in CONSYDERR. Activations of property-values relative to the number of features in A, when $A$ is fully activated; where $A \supset B$, $F_A \subset F_B$, $|F_B| = 10$, and $A \longrightarrow C$, $B \longrightarrow D$; the rule weights are equal to 1.

Steven Sloman for their detailed comments on the draft. I wish to thank Allan Collins for providing me with the data. Thanks also go to the anonymous reviewers for their comments.

## References

[1] V. Ajjanagadde and L. Shastri, Efficient inference with multi-place predicates and variables in a connectionist system, in: *Proceedings Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI (1989) 396–403.

[2] J. Anderson and E. Rosenfeld, eds., *Neurocomputing* (MIT Press, Cambridge, MA, 1988).

[3] J. Anderson and R. Thompson, Use of analogy in a production system architecture, in: S. Vosniadou and A. Ortony, eds., *Similarity and Analogical Reasoning* (Cambridge University Press, Cambridge, England, 1989).

[4] J. Barnden, The right of free association: relative-position encoding for connectionist data structures, in: *Proceedings Tenth Annual Conference of the Cognitive Science Society*, Montreal, Que. (1988) 503–509.

[5] J. Barnden and K. Srinivas, Overcoming rule-based rigidity and connectionist limitations through massively parallel case-based reasoning, *Int. J. Man-Mach. Stud.*, to appear.

[6] B. Buchanan and E. Shortliffe, eds., *Rule-Based Reasoning: the Mycin Experiment* (Addison-Wesley, Reading, MA, 1989).

[7] J. Chomsky, Rules and representation, *Behav. Brain Sci.* (1980) 1–16

[8] A. Collins, Fragments of a theory of human plausible reasoning, in: D. Waltz, ed., *Theoretical Issues in Natural Language Processing II* (University of Illinois, Urbana, IL, 1978) 194–201.

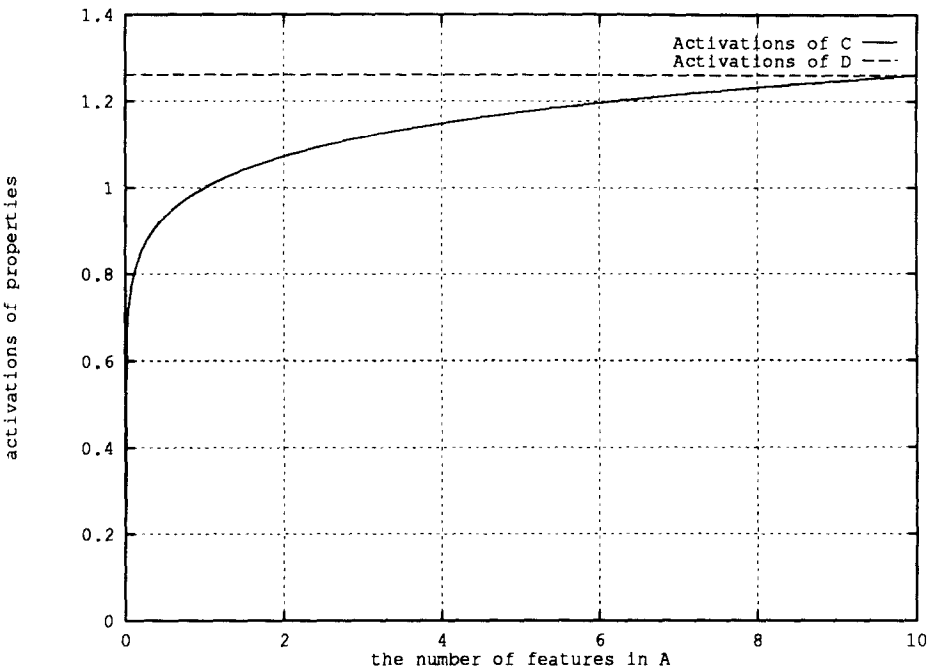



Fig. B.12. Inheritance in CONSYDERR. Activations of property-values relative to the number of features in A, when $B$ is fully activated; where $A \supset B$, $F_A \subset F_B$, $|F_B| = 10$, $A \longrightarrow C$, and $B \longrightarrow D$; the rule weights are equal to 1.

[9] A. Collins and J. Loftus, Spreading activation theory of semantic processing, *Psychol. Rev.* **82** (1975) 407–428.
[10] A. Collins and R. Michalski, The logic of plausible reasoning: a core theory, *Cogn. Sci.* **13** (1) (1989) 1–49.
[11] E. Davis, *Representations of Commonsense Knowledge* (Morgan Kaufmann, San Mateo, CA, 1990).
[12] M. Derthick, Mundane reasoning by parallel constraint satisfaction, Tech. Report TR CMU-CS-88-182, Carnegie-Mellon University, Pittsburg, PA (1988).
[13] H. Dreyfus and S. Dreyfus, *Mind Over Machine* (The Free Press, New York, 1987).
[14] D. Dubois and H. Prade, An introduction to possibilistic and fuzzy logics, in: P. Smets et al., eds. *Non-Standard Logics for Automated Reasoning* (Academic Press, San Diego, CA, 1988).
[15] M. Dyer, Distributed symbol formation and processing in connectionist networks, *J. Expt. Theor. Artif. Intell.* **2** (1990) 215–239.
[16] J. Fodor and Z. Pylyshyn, Connectionism and cognitive architecture: a critical analysis, in: S. Pinker and J. Mehler, eds., *Connections and Symbols* (MIT Press, Cambridge, MA, 1988).
[17] L. Fu, Rule learning by searching on adapted nets, in: *Proceedings AAAI-91*, Anaheim, CA (1991) 590–595.
[18] A. Golding and P. Rosenbloom, Improving rule-based systems through case-based reasoning, *Proceedings AAAI-91*, Anaheim, CA (1991) 22–27.
[19] S. Harnad, The symbol grounding problem, *Physica D* **42** (1–3) (1990) 335–346.
[20] P.J. Hayes, In defence of logic, in: *Proceedings IJCAI-77*, Cambridge, MA (1977) 559–565.
[21] F. Hayes-Roth, D.A. Waterman and D.B. Lenat, eds., *Building Expert Systems* (Addison-Wesley, Reading, MA, 1983).
[22] D. Heckerman, Probabilistic interpretation for MYCIN's certainty factors, in: L.N. Kanal and J.F. Lemmer, eds., *Uncertainty in Artificial Intelligence* (Elsevier Science Publishers, New York, 1985) 167–196.

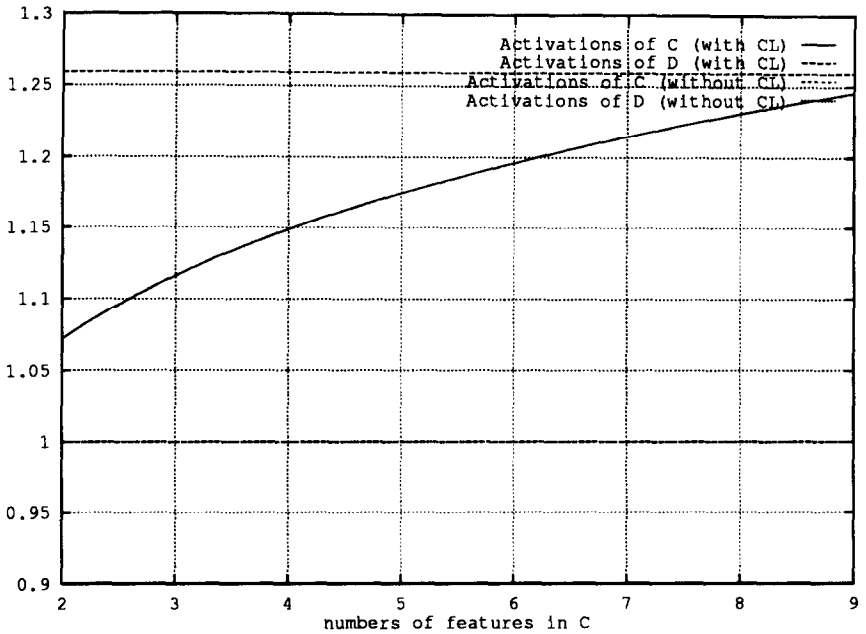



Fig. B.13. One-level versus two-level: inheritance and cancellation. $A \supset B$, $F_A \subset F_B$. $A \longrightarrow C$. $B \longrightarrow D$. All the weights are equal to 1. $F_C \subset F_D$. Assume $B$ is fully activated. $D$ should win over $C$. $|F_D|$ is fixed at 10, while $|F_C|$ varies from 2 to 9.

[23] G. Hinton, Mapping part-whole hierarchies into connectionist networks, *Artif. Intell.* **46** (1990) 47–76.
[24] J. Holland, Escaping brittleness, in: R. Michalski, J. Carbonell and T. Mitchell eds., *Mach. Learn.* **2** (Morgan Kaufmann, San Mateo, CA, 1986) 593–625.
[25] J. Holland, N. Nisbitt, T. Thagard and J. Holyoak, *Induction: A Theory of Learning and Development* (MIT Press, Cambridge, MA, 1986).
[26] K. Holyoak and P. Thagard, A computational model of analogical problem solving, in: S. Vosniadou and A. Ortony, eds., *Similarity and Analogical Reasoning* (Cambridge University Press, New York, 1989).
[27] F. Keil, *Concepts, Kinds, and Cognitive Development* (MIT Press, Cambridge, MA, 1989).
[28] T. Lange and M. Dyer, Frame selection in a connectionist model, in: *Proceedings Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI (1989) 706–713.
[29] J. McCarthy, Programs with common sense, in: M. Minsky, ed., *Semantic Information Processing* (MIT Press, Cambridge, MA, 1968).
[30] N. Nilsson, *Principles of Artificial Intelligence* (Tiogo, San Mateo, CA, 1980).
[31] L. Norman, *Human Information Processing* (Academic Press, San Diego, CA, 1977).
[32] D. Osherson, E. Smith, and E. Shafir, Some origins of belief, Tech. Report 3, Cognitive Science and Machine Intelligence Lab., University of Michigan, Ann Arbor, MI (1987).
[33] D. Osherson, J. Stern, O. Wilkie, M. Stob and E. Smith, Default probability, *Cogn. Sci.* **15** (1991) 251–269.
[34] J. Pearl, *Probabilistic Reasoning in Intelligent Systems* (Morgan Kaufmann, San Mateo, CA, 1988).
[35] S. Pinker and A. Prince, On language and connectionism, in: S. Pinker and J. Mehler, eds., *Connections and Symbols* (MIT Press, Cambridge, MA, 1988).
[36] C. Riesbeck and R. Schank, *Inside Case-based Reasoning* (Lawrence Erlbaum, Hillsdale, NJ, 1989).
[37] D. Rumelhart, J. McClelland and the PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructures of Cognition* (MIT Press, Cambridge, MA, 1986).
[38] G. Shafer, *A Mathematical Theory of Evidence* (Princeton University Press, Princeton, NJ, 1974).

[39] Y. Shoham, Reasoning about change: time and causation from the standpoint of artificial intelligence, Ph.D. Dissertation, Computer Science Department, Yale University, New Heaven, CT (1987).
[40] T. Shultz, P. Zelazo and D. Engelberg, Managing uncertainty in rule-based reasoning, in: *Proceedings Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI (1989) 227–234.
[41] S. Sloman, Feature-based induction, *Cogn. Psychol.*, to appear.
[42] E. Smith, C. Langston and R. Nisbett, The case for rules in reasoning, *Cogn. Sci.* **16** (1992) 1–40.
[43] P. Smolensky, On the proper treatment of connectionism, *Behav. Brain Sci.* **11** (1988) 1–43.
[44] R. Sun, A discrete neural network model for conceptual representation and reasoning, in: *Proceedings Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI (1989) 916–923.
[45] R. Sun, Connectionist models of rule-based reasoning, in: *Proceedings Thirteenth Annual Conference of the Cognitive Science Society*, Chicago, IL (1991) 437–442.
[46] R. Sun, A connectionist model for commonsense reasoning incorporating rules and similarities, *Knowledge Acquisition* **4** (1992) 293–321.
[47] R. Sun, On variable binding in connectionist networks, *Connection Sci.* **4** (2) (1992) 93–124.
[48] R. Sun, An efficient feature-based connectionist inheritance scheme, *IEEE Trans. Syst. Man Cybern.* **23** (2) (1993) 512–522.
[49] R. Sun, *Integrating Rules and Connectionism for Robust Commonsense Reasoning* (Wiley, New York, 1994).
[50] R. Sun, A neural network model of causality, *IEEE Trans. Neural Networks* **5** (1994) 604–611.
[51] R. Sun, L. Bookman and S. Shekhar, eds. *Working Notes of the AAAI Workshop on Integrating Neural and Symbolic Processes* (AAAI, Menlo Park, CA, 1992).
[52] R. Sun and D. Waltz, Neural networks and human intelligence, *J. Math. Psychol.* **34** (4) 483–488. (1990).
[53] R. Sun and D. Waltz, Neurally inspired massively parallel model of rule-based reasoning, in: B. Soucek, ed., *Neural and Intelligent System Integration* (Wiley, New York, 1991) 341–381.
[54] P. Thagard and K. Holyoak, How to compute semantic similarity, in: *Proceedings DARPA Case-Based Reasoning Workshop*, Pensacola Beach, FL (1989) 85–88.
[55] D. Touretzky, *The Mathematics of Inheritance* (Morgan Kaufmann, San Mateo, CA, 1985).
[56] D. Touretzky and G. Hinton, Symbols among neurons, in: *Proceedings IJCAI-85*, Los Angeles, CA (1985) 238–243.
[57] A. Tversky, Features of similarity, *Psychol. Rev.* **84** (4) (1977) 327–352.
[58] S. Vosniadou and A. Ortony, eds., *Similarity and Analogical Reasoning* (Cambridge University Press, New York, 1989).
[59] D. Waltz, Connectionist models: not just a notational variant, not a panacea, in: D. Waltz, ed., *Theoretical Issues in Natural Language Processing* (Ablex, Norwood, NJ, 1988) 1–8.
[60] D. Waltz, Is indexing used for retrieval? in: *Proceedings DARPA Case-Based Reasoning Workshop*, Pensacola Beach, FL (1989) 41–45.
[61] L. Zadeh, Fuzzy Logic, *Computer* **21** (4) (1988) 83–93.