

# Local Theories of Inheritance

Krishnaprasad Thirunarayan \*

Department of Computer Science and Engineering

Wright State University, Dayton, OH 45435.

(513)-873-5109

tkprasad@cs.wright.edu

## Abstract

Inheritance networks are not expressive enough to capture all intuitions behind inheritance. Hence, a number of significantly different semantics have been proposed. Here, a family of *local* theories of inheritance is presented that are conceptually simple and computationally tractable. A general model-theoretic framework for specifying *direct* semantics to networks and clarifying relationships among them is developed. It is also shown that the local constraints specifying the semantics are satisfiable when the attention is restricted to *acyclic* networks. The notion of local semantics is generalized to what is called “ground” local semantics to further understand the relationship between the local theories and the path-based theories of inheritance. A polynomial-time set-based inheritance algorithms that conform to these inheritance theories is also presented.

---

\*This research was supported in part by the NSF grant IRI-9009587.

## 1 Introduction

In order to exhibit intelligent behaviour, a system needs to represent and utilize knowledge about the relevant domains. Hence, it is necessary to design representational languages that express this knowledge concisely and develop reasoning methods to derive facts implicit in these representations. The general task of knowledge representation and reasoning is computationally very hard. But a significant portion of common-sense reasoning can be factored out as inheritance reasoning that can be carried out quickly. This has been the primary motivation for the use of inheritance networks for knowledge representation, and the design of efficient special purpose algorithms to reason with them.

Any realistic representation of the real-world knowledge must necessarily allow representation of exceptions. For instance, typically, mammals are nonflyers. Bats are exceptional mammals because, normally, they can fly. But gravely sick bats do not fly. In general, properties inherited from a class,  $c$ , must dominate the properties inherited from any superclass of  $c$ , in case of a conflict.

The representation language should also allow expression of *preferential* inheritance of a property from a class over inheritance from another class. For instance, given that Pratt is a graduate teaching assistant, that is, Pratt is both a graduate student and a teaching assistant, one must be able to infer that Pratt gets salary and is required to pay taxes. Note that even though a graduate student typically does not get salary, and hence is not a tax payer, Pratt “inherits” its tax paying status because he is a teaching assistant.

There have been a number of proposals for the semantics of inheritance networks. This has been done either through a translation into a general logical formalism or by special path-based techniques. The former approach requires the specification of an *algorithmic* transformation of the network into a set of sentences in a logical language. The intuition about inheritance is captured *indirectly* through the semantics of the logic language. The user must know the logical formalism and the translation algorithm to understand the meaning of the network [5, 8, 12, 19, 20, 25, 32]. Furthermore, the “declarativeness” of these approaches is arguable, as it depends on the conceptual difficulty of the algorithmic part. Since most transformations are quite complex, it is often unclear how the different semantics relate to each other.

In the second approach, the semantics is given by characterizing sets of *inheritance paths* in the network [10, 13, 14, 28, 36, 39, 40].

In this paper we put forth a general framework for specifying model theory of inheritance networks *directly*, bypassing complex intermediate transformations. From the user’s perspective, this is desirable, as it makes comprehension of inheritance specifications easier. Furthermore, there seem to be equally well-motivated, yet significantly different interpretations of certain network topologies. This is, in part, due to the fact that inheritance networks are not sufficiently expressive. Our direct, model-theoretic approach enables careful study of the differences among some of these proposals.

The *local* theories of inheritance, we propose, are conceptually simple and computationally attractive [18]. Informally, a semantics of inheritance networks has the *locality* property if the meaning of a node depends only on the meaning assigned to its immediate neighbors and not on the meaning assigned to nodes arbitrarily far away. Locality accrues benefits such as compositionality. To understand the meaning of the whole network, one can understand the meaning of its constituent subnetworks and then compose their meaning appropriately using only the meaning of the “interface” nodes. The locality property enables a declarative understanding of the network. Furthermore, the changes in the meaning of a node propagate smoothly through the network. (We are tacitly assuming that the gluing process does not change the specificity relationships among the nodes in the subnetworks.) It also supports efficient parallel computation of the meaning of large networks. This is because, one can partition the task of computing the meaning of a network into smaller independent subtasks of computing the meanings of its subnetworks, and then combining these meanings to obtain the meaning of the whole network.

The paper is organized as follows. Section 2.1 specifies the syntax of preferential inheritance networks which are simple generalization of inheritance networks. Section 2.2 provides an informal description of our approach. We develop a model-theoretic framework for the declarative semantics of inheritance networks in Section 2.3 and define the local specificity we espouse in Section 2.4. In Section 2.5, we study some of the salient features of the proposed semantics. Section 2.6 defines and analyzes a model-theoretic notion of locality in the context of inheritance net-

works. In Section 2.7, we discuss the inheritance algorithm. Finally, we conclude in Section 3.

## 2 Direct Semantics of Inheritance Networks

We describe a general framework for specifying declarative semantics of preferential networks *directly* in terms of the network [18]. In particular, our approach captures “individual flow” intuitions about inheritance [13], which capture more adequately notion of inheritance than the “property flow view” [21, 39]. Pragmatically, the most important payoff of this approach is that inheritance algorithms based on our theories are *computationally tractable*.

### 2.1 Syntax of Preferential Inheritance Networks

Traditional inheritance networks are extended to preferential inheritance networks by imposing suitable orderings on the in-arcs into a node. Note that we order the *children* of each node rather than its *parents* because understanding of inheritance as the flow of individuals up the network to acquire properties captures more adequately our intuitions about inheritance than the view of properties flowing downwards [21]. This “specificity” ordering captures class-subclass relationship (as in the bats-example in Section 1) and supports representation of disambiguation information not directly inferrable from the topology (as in the GTA-example in Section 1).

**Definition 1** *A preferential inheritance network is an ordered directed acyclic graph consisting of a set of individual nodes  $\mathbf{I}$ , a set of property nodes  $\mathbf{P}$ , a set of positive arcs  $\mathbf{E}^+ \subseteq (\mathbf{I} \cup \mathbf{P}) \times \mathbf{P}$ , a set of negative arcs  $\mathbf{E}^- \subseteq (\mathbf{I} \cup \mathbf{P}) \times \mathbf{P}$ , and for each node  $p \in \mathbf{P}$ , a specificity relation  $\prec_p$  on its in-arcs.*

We use  $p, q$  and  $r$  to stand for any property node in  $\mathbf{P}$ , and  $i$  for any individual node in  $\mathbf{I}$ . Assume  $\mathbf{E}^+ \cap \mathbf{E}^- = \emptyset$ .

## 2.2 Motivational Examples

We illustrate our approach using the network in Figures 1 and 2 whose arcs represent the following facts:

- Clyde is an individual. Tweety is a penguin. Donald is a bird.
- Penguins are Birds. Typically, birds fly. Typically, penguins do not fly.

For our purposes, the state of the world is characterized by a description of the properties possessed by the individuals in the world. A *semantic structure* for the network is a triple  $(\mathcal{D}, \mathcal{I}, \mathcal{M})$ , where  $\mathcal{D}$  stands for the domain of discourse,  $\mathcal{I}$  stands for the individual meaning function for nodes, and  $\mathcal{M}$  stands for the set-based meaning function for property nodes.

The *domain* of discourse  $\mathcal{D}$  is a pair of sets  $(\mathcal{D}_i, \mathcal{D}_p)$ , where  $\mathcal{D}_i$  stands for a set of individuals and  $\mathcal{D}_p$  for a set of properties. The interpretation function  $\mathcal{I}$  assigns to each individual node in  $\mathbf{I}$  an individual from  $\mathcal{D}_i$  and to each property node in  $\mathbf{P}$  a property from  $\mathcal{D}_p$ . The interpretation function  $\mathcal{M}$  assigns to every property node  $p$  in  $\mathbf{P}$  a pair of subsets of  $\mathcal{D}_i$ , denoted as  $(p^+, p^-)$ , where  $p^+$  (resp.  $p^-$ ) contains individuals that are believed to possess property  $p$  (resp.  $\neg p$ ).  $p^+ \cap p^-$  contains individuals for which there is evidence to possess both  $p$  and  $\neg p$ . To accommodate generic statements,  $\mathcal{D}_i$  can be replaced by  $(\mathcal{D}_i \cup \mathcal{D}_p)$ .

The following assignment to the nodes of the network given in Figure 1(a) depicts a possible interpretation of the network.

- $\mathcal{D}_i = \{Clyde, Donald, Tweety\}$ .  
 $\mathcal{D}_p = \{penguins, birds, fly\}$ .
- $\mathcal{I}(Clyde) = Clyde$ .  
 $\mathcal{I}(Donald) = Donald$ .  
 $\mathcal{I}(Tweety) = Tweety$ .  
 $\mathcal{I}(penguins) = penguins$ .  
 $\mathcal{I}(birds) = birds$ .  
 $\mathcal{I}(fly) = fly$ .

- $\mathcal{M}(\text{penguins}) = (\{\}, \{\})$ .  
 $\mathcal{M}(\text{birds}) = (\{\}, \{\text{Clyde}\})$ .  
 $\mathcal{M}(\text{fly}) = (\{\text{Donald}, \text{Tweety}\}, \{\text{Clyde}\})$ .

However, this assignment does not seem to capture the intuitive meaning of the network. For instance, it is not known from the above assignment whether Tweety is a penguin or not, even though it can be gleaned from the given information that Tweety is a penguin. To rule out such an assignment as unsuitable, we make explicit constraints that the edges impose on the assignment of sets to nodes. In particular, whenever there is a positive (resp. negative) arc from an individual node  $n$  to a property node  $p$ , we require that  $n \in p^+$  (resp.  $n \in p^-$ ). Thus  $\text{Tweety} \in \text{penguins}^+$  and  $\text{Donald} \in \text{birds}^+$ .

Similarly, observe that penguins are birds, and hence, *Tweety* should also belong to  $\text{birds}^+$ . That is, we wish to rule out interpretations of the network in which an entity is a penguin, but is not a bird. To capture the meaning of a “strict” positive (resp. negative) arc from a property node  $p$  to a property node  $q$ , we add the constraint  $p^+ \subseteq q^+$  (resp.  $p^+ \subseteq q^-$ ). The following interpretation of the network satisfies the afore-mentioned constraints. (Also see Figure 1(b).)

- $\mathcal{M}(\text{penguins}) = (\{\text{Tweety}\}, \{\})$ .  
 $\mathcal{M}(\text{birds}) = (\{\text{Donald}, \text{Tweety}\}, \{\text{Clyde}\})$ .  
 $\mathcal{M}(\text{fly}) = (\{\text{Donald}, \text{Tweety}\}, \{\text{Clyde}, \text{Tweety}\})$ .

Even though, on the basis of the given facts, we intuitively conclude that Tweety does not fly, our formalization supports inconsistent conclusions about Tweety’s flight. This is because the arcs from node bird and node penguin into node fly have been incorrectly interpreted as strict, rather than as *defeasible*. Given the fact that penguins are a subclass of birds, the contradiction can be resolved in favor of Tweety’s inability to fly. These ideas can be formalized as follows: A positive (resp. negative) defeasible arc from a property node  $p$  to a property node  $q$  is translated as  $(p^+ - \text{knownexceptions}) \subseteq q^+$  (resp.  $(p^+ - \text{knownexceptions}) \subseteq q^-$ ), where the “known exceptions” to a rule is interpreted as *stronger conflicting conclusions*. For the example at hand this turns out to be:  $\text{penguins}^+ \subseteq \text{fly}^-$  and  $(\text{birds}^+ - \text{penguins}^+) \subseteq \text{fly}^+$ . Note that in going from strict arcs to

defeasible arcs the focus shifted from relating assignments to pairs of adjacent nodes, to relating assignments to a node and all of its neighboring nodes. This happens because *the context of a node is very important in interpreting the meaning of a defeasible arc incident on it*. The interpretations that satisfy these constraints are called *models* of the network. These capture states of the world consistent with the network. A possible model of the network in Figure 2(a) is:

- $\mathcal{M}(\text{penguins}) = (\{\text{Tweety}\}, \{\})$ .
- $\mathcal{M}(\text{birds}) = (\{\text{Donald}, \text{Tweety}\}, \{\text{Clyde}\})$ .
- $\mathcal{M}(\text{fly}) = (\{\text{Donald}\}, \{\text{Tweety}, \text{Clyde}\})$ .

From the given facts, it is also conceivable that the individual Clyde is a bird and that it flies. However, there is nothing in the input to suggest that or otherwise. To reflect this ignorance, and to avoid baseless pinning down of the characteristics of Clyde, we introduce the notion of a minimal model. In the absence of cycles in the network, this can be formalized by replacing  $\subseteq$  by  $=$  as shown:

- $(\text{penguins}^+, \text{penguins}^-) = (\{\text{Tweety}\}, \{\})$
- $(\text{birds}^+, \text{birds}^-) = ([\{\text{Donald}\} \cup \text{penguins}^+], \{\})$
- $(\text{fly}^+, \text{fly}^-) = ([\text{birds}^+ - \text{penguins}^+], \text{penguins}^+)$ .

The unique minimal model satisfying these constraints turns out to be:

- $\mathcal{M}(\text{penguins}) = (\{\text{Tweety}\}, \{\})$ .
- $\mathcal{M}(\text{birds}) = (\{\text{Donald}, \text{Tweety}\}, \{\})$ .
- $\mathcal{M}(\text{fly}) = (\{\text{Donald}\}, \{\text{Tweety}\})$ .

In words, Tweety is a penguin bird that cannot fly, while Donald is a flying bird. Also see Figure 2(b).

The afore-mentioned formalization captures the properties one can associate with the individuals Tweety, Donald etc. To determine the set of *properties* that can be associated with a “typical” penguin, or a “typical” bird, or a “typical” flying object, we can imagine associating with each property node a special individual node as its child. That is, asserting  $p \in p^+$  for each node  $p$ . Then, the properties

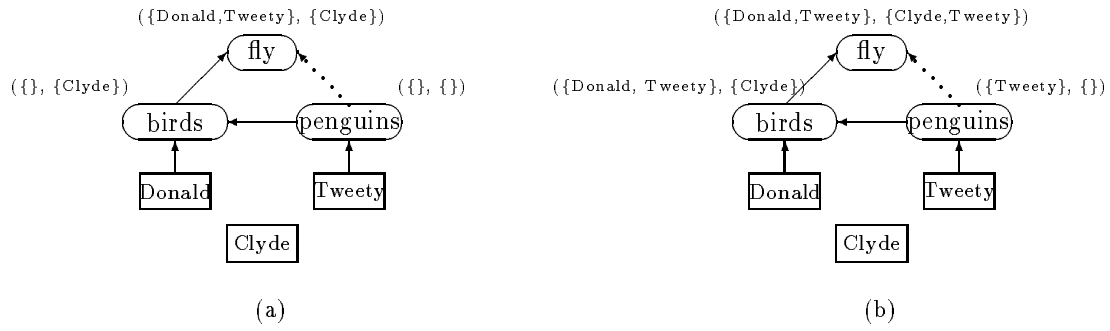


Figure 1: Interpretations of the Penguin Triangle

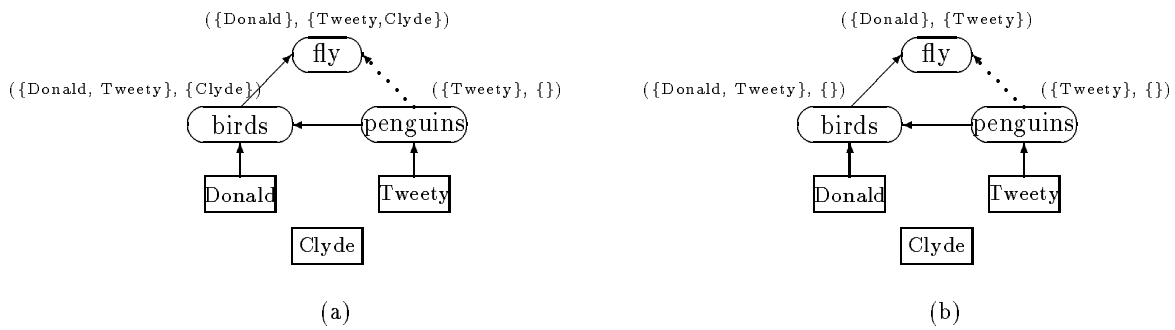


Figure 2: A Model and a Minimal Model

inherited by the new individual  $p$  corresponding to the property node  $p$  gives the collection of properties associated with a “typical” member of  $p$ . These inherited properties can also be interpreted as the set of generic statements supported by our network.

With this augmentation, the minimal model of our network becomes:

- $\mathcal{M}(\text{penguins}) = (\{\text{penguin}, \text{Tweety}\}, \{\})$ .
- $\mathcal{M}(\text{birds}) = (\{\text{bird}, \text{Donald}, \text{penguin}, \text{Tweety}\}, \{\})$ .
- $\mathcal{M}(\text{fly}) = (\{\text{fly}, \text{bird}, \text{Donald}\}, \{\text{penguin}, \text{Tweety}\})$ .

and the corresponding conclusions supported by our network are:

- Tweety is a penguin. Tweety is a bird. Tweety does not fly.
- Donald is a bird. Donald flies.
- Penguins are birds. Penguins do not fly. Birds fly.

Consider now the specificity relation used to disambiguate conflicting conclusions about Tweety’s flight. We argued that the inheritance of  $\neg\text{fly}$  via penguins dominates the inheritance of  $\text{fly}$  via birds. This is justified because, if, on the basis of the fact that  $x$  is a *penguins*,  $x$  is a *birds*, then we can regard *penguins* as providing more specific information than *birds*. So when we associate a property  $\neg\text{fly}$  with *penguins* we are willing to override any evidence for  $\text{fly}$  that rests on the “implicit” property *birds* of the members of *penguins*. This notion of specificity can be formalized by determining whether or not  $\text{penguins} \in \text{birds}^+$ , and if so, setting

$$\forall \text{ common parents of } \text{fly} : \langle \text{birds}, \text{fly} \rangle \prec_{\text{fly}} \langle \text{penguins}, \text{fly} \rangle.$$

Thus, we observe that the computation of inheritance relation and the specificity relation can be integrated.

We now consider another detailed example of specifying the semantics of monotonic inheritance [37] in our framework. (For simplicity, we do not specify reasoning with generic statements.) Here, each arc is interpreted as strict, that is, it stands for a statement that *does not* admit any exception. In the event of a conflict, the contradictory properties may be assigned the status of inconsistent (or ambiguous)

belief. In the Nixon diamond example the positive (resp. negative) arc from node quaker (resp. republican) to node pacifist is interpreted as an assertion that “every quaker is a pacifist” (resp. “every republican is not a pacifist ”). Under this interpretation, if Nixon is both a quaker and a republican then Nixon is inferred to be a pacifist because he is a quaker, and inferred not to be a pacifist because he is a republican. This results in an inconsistent belief about Nixon’s pacifism.

The constraints that capture the semantics of monotonic inheritance are:

- C1 Individual  $i$  has property  $p$  if there is a positive arc from node  $i$  to node  $p$ , or individual  $i$  has property  $q$  and there is a positive arc from node  $q$  to node  $p$ .

Formally, for each property  $p$  and individual  $i$ :

$$p^+(i) \text{ if } \mathbf{E}^+(i,p) \vee \exists q [ \mathbf{E}^+(q,p) \wedge q^+(i) ].$$

- C2 Individual  $i$  does not have property  $p$  if there is a negative arc from node  $i$  to node  $p$ , or individual  $i$  has property  $q$  and there is a negative arc from node  $q$  to node  $p$ , or individual  $i$  has property  $q$  and there is a negative arc from node  $p$  to node  $q$ , individual  $i$  has property  $\neg q$  and there is a positive arc from node  $p$  to node  $q$ .

Formally, for each property  $p$  and individual  $i$ :

$$p^-(i) \text{ if } \mathbf{E}^-(i,p) \vee \exists q [ ( \mathbf{E}^-(q,p) \wedge q^+(i) ) \vee ( \mathbf{E}^-(p,q) \wedge q^+(i) ) \vee ( \mathbf{E}^+(p,q) \wedge q^-(i) ) ].$$

Figure 3(a) depicts a network which represents the following facts:

- Richard is a quaker. Nixon is a republican.
- Every quaker is a pacifist. Every republican is a nonpacifist.

The following interpretation is a valid semantic structure for this network:

- $\mathcal{D}_i = \{R.Nixon, Carter\}$ .
- $\mathcal{D}_p = \{quaker, republican, pacifist\}$ .

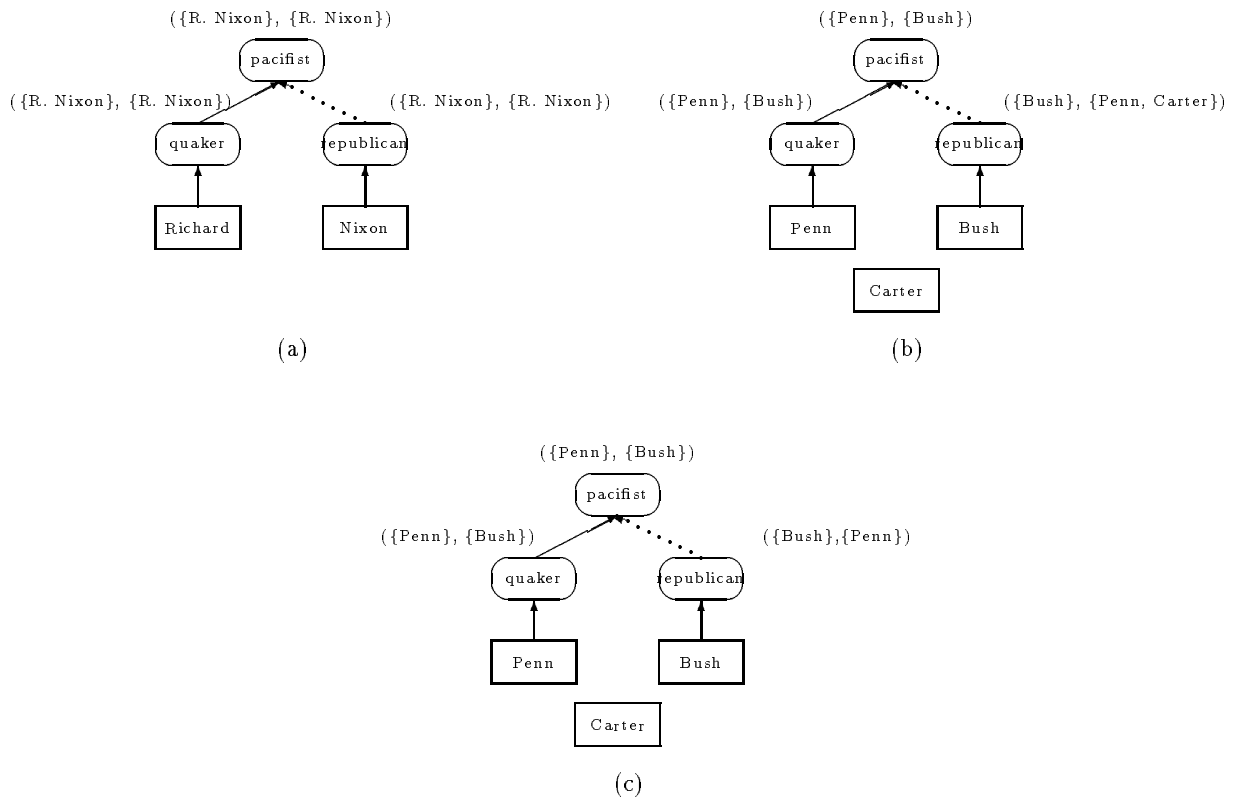


Figure 3: The monotonic inheritance models

- $\mathcal{I}(\text{Richard}) = R.Nixon.$   
 $\mathcal{I}(\text{Nixon}) = R.Nixon.$   
 $\mathcal{I}(\text{quaker}) = quaker.$   
 $\mathcal{I}(\text{republican}) = republican.$   
 $\mathcal{I}(\text{pacifist}) = pacifist.$
- $\mathcal{M}(\text{quaker}) = (\{R.Nixon\}, \{R.Nixon\}).$   
 $\mathcal{M}(\text{republican}) = (\{R.Nixon\}, \{R.Nixon\}).$   
 $\mathcal{M}(\text{pacifist}) = (\{R.Nixon\}, \{R.Nixon\}).$

Observe that both individual nodes Richard and Nixon are mapped to the same individual  $R.Nixon$ , and that there is no explicit reference to the individual  $Carter$ . It can be verified that this structure satisfies the constraints C1 and C2, and hence, it is a *model* of the network.

In particular,  $R.Nixon$  is believed to be a *quaker* (resp. *republican*) because of the direct positive arc from node Richard (resp. Nixon) to node quaker (resp. republican).  $R.Nixon$  is believed to be  $\neg$ *quaker* (resp.  $\neg$ *republican*) because of the positive (resp. negative) arc from the node quaker (resp. republican) to node pacifist and the fact that  $Richard\_Nixon$  is believed to be a *pacifist* (resp.  $\neg$ *pacifist*).

Figure 3(b) depicts a network which represents the following facts:

- Carter is an individual.
- Penn is a quaker. Bush is a republican.
- Every quaker is a pacifist. Every republican is a nonpacifist.

The following semantic structure is a *Herbrand model* for this network because it satisfies the network and also the unique-name hypothesis (that is, distinct individual (resp. property) nodes map to distinct individuals (resp. properties)):

- $\mathcal{D}_i = \{Penn, Bush, Carter\}.$   
 $\mathcal{D}_p = \{quaker, republican, pacifist\}.$
- $\mathcal{I}(\text{Penn}) = Penn.$   
 $\mathcal{I}(\text{Bush}) = Bush.$

$\mathcal{I}(\text{Carter}) = \text{Carter}.$   
 $\mathcal{I}(\text{quaker}) = \text{quaker}.$   
 $\mathcal{I}(\text{republican}) = \text{republican}.$   
 $\mathcal{I}(\text{pacifist}) = \text{pacifist}.$

- $\mathcal{M}(\text{quaker}) = (\{\text{Penn}\}, \{\text{Bush}\}).$
- $\mathcal{M}(\text{pacifist}) = (\{\text{Penn}\}, \{\text{Bush}\}).$
- $\mathcal{M}(\text{republican}) = (\{\text{Bush}\}, \{\text{Penn}, \text{Carter}\}).$

Even though the assumption that Carter is not a republican is consistent with the network, it seems redundant. To capture the intended meaning of the network, we formalize the notion of minimality by replacing **if** with **iff** in the constraints C1 and C2. The semantic structure in Figure 3(c) is the same as that in Figure 3(b), except that the meaning of node republican is changed as follows:

$\mathcal{M}(\text{republican}) = (\{\text{Bush}\}, \{\text{Penn}\}).$

This structure satisfies the minimality constraints, and hence, is a minimal model of the network.

To understand the differing interpretations of ambiguity, consider again the Nixon's diamond shown in Figure 3 where the arcs into node pacifist are reinterpreted as *defeasible* (rather than as strict).

- Quakers are normally pacifists.
- Republicans are normally not pacifists.

In this situation we can feign ignorance about Nixon's pacifism. This is legitimate, since we have two equally strong conflicting evidences in support of his pacifism and there is no way to choose one over the other to determine whether Nixon is a pacifist or not. This is the *skeptical approach* to ambiguity. On the other hand, it is equally reasonable to argue that in real life Nixon is either a pacifist or a non-pacifist, and to associate a set of two minimal models with this network. In one model, Nixon is a Pacifist, while in the other, he is not. This is the *credulous approach* to ambiguity. Formalization of all these aspects and options is the subject matter of the following sections.

### 2.3 Direct Semantics

We present formal specifications of a family of related inheritance theories for preferential inheritance networks.

**Definition 2** *A Herbrand semantic structure for a preferential network is a pair  $(\mathcal{D}, \mathcal{M})$ , where  $\mathcal{D}$  stands for the domain of discourse, and  $\mathcal{M}$  for the set-based meaning function. The domain  $\mathcal{D}$  is a pair  $(\mathcal{D}_i, \mathcal{D}_p)$ , where  $\mathcal{D}_i$  is a set of individuals corresponding to nodes in  $\mathbf{I}$ , and  $\mathcal{D}_p$  is a set of properties corresponding to nodes in  $\mathbf{P}$ . The meaning function  $\mathcal{M}$  associates with each property node  $p \in \mathbf{P}$ , a pair of subsets  $(p^+, p^-)$  of  $\mathcal{D}_i \cup \mathcal{D}_p$ , where  $p^+$  represents the set of individuals that possess property  $p$  union the set of properties by virtue of which an individual may possess  $p$ ; and  $p^-$  represents the set of individuals that possess  $\neg p$  union the set of properties by virtue of which an individual may possess  $\neg p$ .*

We use the letters  $n, m$  to denote individual nodes in  $\mathbf{I}$  and  $n, m$  to denote the respective individuals in  $\mathcal{D}_i$ . The letters  $p, q, \dots$  will be used to denote property nodes in  $\mathbf{P}$  and  $p, q, \dots$  to denote the respective properties in  $\mathcal{D}_p$ . We assume, for notational convenience, the equivalences:  $p^+(n) \equiv n \in p^+$  (resp.  $p^-, p^*$ ) and  $\neg p^+(n) \equiv n \notin p^+$  (resp.  $p^-, p^*$ ).

The different approaches to inheritance differ in the specification of the constraints for a semantic structure to be a model. Informally, individual  $n$  should inherit property  $p$  if the maximal evidence in support of the inheritance is through a positive arc to node  $p$ ; and individual  $n$  should inherit property  $\neg p$  if the maximal evidence is through a negative arc to node  $p$ . There is maximal evidence in support of  $\neg p(n)$  (resp.  $p(n)$ ) if there is a negative (resp. positive) arc from  $n$  to  $p$ , or if there is no positive (resp. negative) arc from  $n$  to  $p$ , but there is an “undefeated” negative (resp. positive) arc from  $t$  to  $p$  and  $n$  inherits  $t$ . The negative (resp. positive) arc from  $t$  to  $p$  is “defeated”, if there is a node  $s$  that has a positive (resp. negative) arc to  $p$  such that  $n$  inherits  $s$ , and  $s$  provides more specific information than  $t$ .

In a skeptical interpretation of ambiguity, individual  $n$  is ambiguous about inheriting  $p$  or  $\neg p$ , whenever there are equally strong or incomparable evidences for  $p$  and  $\neg p$ . However, in a credulous interpretation of ambiguity,  $n$  inherits  $p$  in some of the models and inherits  $\neg p$  in the others.

The statement that “ $\exists$  maximal evidence for  $\neg p(n)$ ” can be formally expressed as:

$$\mathbf{E}^-(n, p) \vee \neg \mathbf{E}^+(n, p) \wedge \exists t [ \mathbf{E}^-(t, p) \wedge t^+(n) \wedge \neg \exists s ( [ \mathbf{E}^+(s, p) \wedge s^+(n) ] \wedge \langle t, p \rangle \prec_p \langle s, p \rangle ) ],$$

and the statement that “ $\exists$  maximal evidence for  $p(n)$ ” can be formally expressed as:

$$\mathbf{E}^+(n, p) \vee \neg \mathbf{E}^-(n, p) \wedge \exists t [ \mathbf{E}^+(t, p) \wedge t^+(n) \wedge \neg \exists s ( [ \mathbf{E}^-(s, p) \wedge s^+(n) ] \wedge \langle t, p \rangle \prec_p \langle s, p \rangle ) ].$$

### 2.3.1 Skeptical Semantics

The following constraints formalize a skeptical semantics of inheritance networks that resembles the meaning assigned to networks by the evidence-based theory [19, 22]. In particular, they specify when an individual  $i$  can possess a property  $p$ .

For each property  $p$  and individual  $n$  :

$$p^-(n) \quad \mathbf{if} \quad \text{“}\exists \text{ maximal evidence for } \neg p(n)\text{”}. \quad (1)$$

$$p^+(n) \quad \mathbf{if} \quad \text{“}\exists \text{ maximal evidence for } p(n)\text{”}. \quad (2)$$

$$p^*(n) \equiv p^+(n) \wedge p^-(n).$$

The constraints for a property  $q$  to provide more specific information than property  $p$  are similar. However, observe that the preferential networks cannot represent complementary inheritance [22].

### 2.3.2 Minimal Models

We may select “preferred” models, by capturing the idea of minimality, by replacing **if** with **iff** in (1) and (2). Note that checking whether a semantic structure is a model or a minimal model is *local*, i.e., for each node, it depends only on its children.

### 2.3.3 An Alternative Skeptical Semantics

The skeptical theory <sup>[13]</sup> treats “no information” and “ambiguous information” on a par. This interpretation of the ambiguity is obtained from the above semantics by associating with each node a pair of sets  $(p_h^+, p_h^-)$ , where

$$p_h^+ = p^+ - p^* \text{ and } p_h^- = p^- - p^*.$$

### 2.3.4 Probabilistic Skeptical Semantics

A skeptical semantics that is related to inheritance theories <sup>[8, 10]</sup> can be specified as follows:

For each property  $p$  and individual  $n$  :

$$p^-(n) \text{ iff } [ \text{“}\exists \text{ maximal evidence for } \neg p(n)\text{”} \vee ( \neg p^+(n) \wedge \exists t ( \mathbf{E}^+(p, t) \wedge t^-(n) \vee \mathbf{E}^-(p, t) \wedge t^+(n) ) ) ]. \quad (3)$$

$$p^+(n) \text{ iff } \text{“}\exists \text{ maximal evidence for } p(n)\text{”}. \quad (4)$$

$$p^*(n) \equiv p^+(n) \wedge p^-(n).$$

The meaning of each property node is a pair of sets  $(p_g^+, p_g^-)$ , where

$$p_g^+ = p^+ - p^* \text{ and } p_g^- = p^- - p^*.$$

The specificity relation described in Section 2.4 is similar to the one obtained by restricting the notion of *defeat* for paths <sup>[10]</sup> to just links <sup>[11]</sup>. Furthermore, the conclusions obtained using the “contrapositive” component are similar to those sanctioned by the causal semantics of inheritance networks <sup>[8]</sup>.

### 2.3.5 Credulous Semantics

The following constraints formalize a credulous semantics of inheritance networks.

For each property  $p$  and individual  $n$  :

$$p^-(n) \text{ if } \neg p^+(n) \wedge \text{“}\exists \text{ maximal evidence for } \neg p(n)\text{”}. \quad (5)$$

$$p^+(n) \text{ if } \neg p^-(n) \wedge \text{“}\exists \text{ maximal evidence for } p(n)\text{”}. \quad (6)$$

$$p^+ \cap p^- = \emptyset. \quad (7)$$

Observe that credulous semantics differs from the skeptical one essentially in the first conjunct of (5) and (6) (cf. (1) and (2)). Note that checking whether a semantic structure is a model is again *local*.

The constraints for a property  $q$  to provide more specific information than property  $p$  are similar.

### 2.3.6 Minimal models

We may filter out “extraneous” models by capturing minimality, similarly to the skeptical case by replacing **if** with **iff** in (5) and (6). In this case, (7) would follow from the modified (5) and (6).

### 2.3.7 An Alternative Credulous Semantics

In addition to the conclusions sanctioned by the above approach, we may strengthen negative conclusions by taking recourse to contrapositive reasoning [20], in which if we know that “p’s are q’s” (resp. “p’s are  $\neg$ q’s”), then we also assume that “ $\neg$ q’s are  $\neg$ p’s” (resp. “q’s are  $\neg$ p’s”).

We specify the credulous semantics that incorporates contrapositive reasoning by modifying the constraint (5) of Section 2.3.5 and augmenting it with two disjuncts as follows:

$$\begin{aligned}
 & p^-(n) \quad \mathbf{if} \\
 & \neg p^+(n) \wedge [ \text{“}\exists \text{ maximal evidence for } \neg p(n)\text{”} \\
 & \vee \exists t ( \mathbf{E}^+(p, t) \wedge t^-(n) \vee \mathbf{E}^-(p, t) \wedge t^+(n) ) ].
 \end{aligned} \tag{8}$$

Furthermore, the minimality constraint is obtained by changing **if** to **iff**. Note that checking whether a semantic structure is a model or a minimal model is still *local*.

### 2.3.8 Generic Statements

The meaning of a node can be defined either as a set of individuals or as a set of properties. To determine the set of *properties* that may be associated with a “typical” individual of a property (resp. class) by virtue of it being a member of that property, imagine associating with each property (resp. class) node a special

individual node as its child. This is achieved by extending the domain of discourse from the set of constants  $D_i$  representing individuals in  $\mathbf{I}$  to the set of constants  $D_i \cup D_p$  representing all nodes in  $\mathbf{I} \cup \mathbf{P}$ , and adding the constraint  $p \in p^+$ , for each node  $p$  in  $\mathbf{P}$ . The properties inherited by the constant  $p$  corresponding to node  $p$  gives the collection of properties associated with a “typical” member of  $p$ . These inherited properties can also be interpreted as the set of generic statements supported by our network.

## 2.4 The Specificity Relation

We view the specificity relation  $\prec_p$  on the in-arcs into a property node  $p$  as specifying the relative strength of inheritance of property  $p$  through them. The specificity relation itself consists of two components: the inferred specificity constraints and the input preferential constraints. (In production system terminology, the inferred specificity is analogous to *meta-knowledge*, while the input preference is analogous to *meta-rule* [26].) The former constraints correspond to class-subclass information normally implicit in the topology of the network, while the latter constraints are explicit disambiguation information input by the representer. (These information can also be derived from the probabilistic analysis [8, 27].) These specificity relations are used to resolve conflicting inheritance. That is, if an individual  $i$  can potentially inherit  $p$  via the positive arc  $\langle q, p \rangle$  and  $\neg p$  via the negative arc  $\langle r, p \rangle$ , then the specificity relation between  $\langle q, p \rangle$  and  $\langle r, p \rangle$  is used to determine which property inheritance prevails over the other.

We briefly describe the inferred specificity constraints for different categories of networks. We regard an inheritance network as *well-formed*, if, for each property node  $p$ , the specificity relation  $\prec_p$  associated with its in-arcs satisfies the inferred constraints. In the sequel, we deal only with well-formed networks.

1. For tree-structured hierarchies, the specificity relation  $\prec_p$  is “static”. It must satisfy the following conditions:
  - (a) An explicitly stated assertion “ $i$  is a  $p$ ” must have priority over the default inheritance of  $\neg p$  by  $i$  through  $r$ . (See Figure 4(a).) That is, for arcs  $\langle i, p \rangle$  and  $\langle r, p \rangle$ , we have  $\langle r, p \rangle \prec_p \langle i, p \rangle$ .

- (b) For all individuals in  $q$ , inheritance of  $p$  (resp.  $\neg p$ ) through a subclass  $q$  must take precedence over inheritance of  $\neg p$  (resp.  $p$ ) through the class  $r$ . (See Figure 4(b).) That is, for any pair of arcs  $\langle q, p \rangle$  and  $\langle r, p \rangle$ , such that there is a directed path from  $q$  to  $r$ , it is the case that  $\langle r, p \rangle \prec_p \langle q, p \rangle$ .
2. For “acyclic” inheritance networks, the specificity relationship is “dynamic” [38], and can be obtained by generalizing Condition 1b. That is, it must satisfy the following conditions:

- (a) A fact always overrides a default conclusion. Hence,

$$\text{if } \mathbf{E}(i, r) \text{ then for all children } q : \langle q, r \rangle \prec_r \langle i, r \rangle$$

- (b) The notion of “local” specificity we can espouse is as follows: If, on the basis of the fact that  $x$  is a  $q$ , we can infer that  $x$  is a  $p$ , then we regard  $q$  as providing more specific information than  $p$ . The intuition is that if we associate a property  $r$  with  $q$ , then we are willing to override any evidence for  $\neg r$  that rests on the “implicit” property  $p$  of the members of  $q$ . This notion of specificity can be *computed locally by the inheritance algorithm* in a natural way. In particular,  $q \in p^+$  implies that  $q$  provides more specific information than  $p$  and hence,

$$\text{if } q \in p^+ \text{ then for all common parents } r : \langle p, r \rangle \prec_r \langle q, r \rangle.$$

Note that the apparent circularity in the afore-mentioned mutually recursive definition of specificity relation and the inheritance relation is in fact well-formed for acyclic networks as demonstrated in Appendix A.

3. The above definition of local specificity cannot capture the skeptical semantics [13] as illustrated by the following example brought to my notice by Karl Schlechta: In Figure 5, the individual  $a$  inherits  $\neg s$  from  $p$  in preference to  $s$  from  $r$  irrespective of whether or not there is a negative arc from  $a$  to  $q$ . This is because  $r^+(p)$  holds and hence,  $p$  is more specific than  $r$ . On the contrary, in the skeptical theory [13], the addition of the negative arc from  $a$  to  $q$  will revise “ $a$  inherits  $\neg s$ ” to “ $a$  is ambiguous about  $s$ ”. This is because, in the

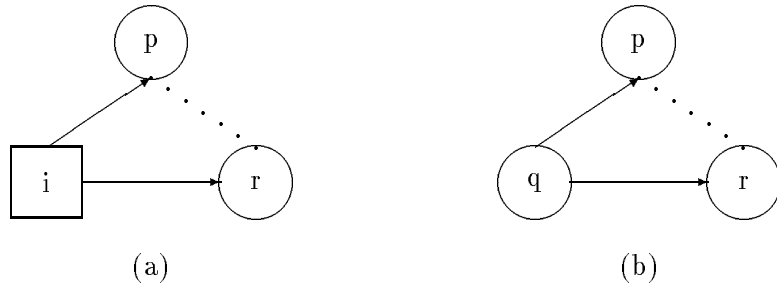


Figure 4: Constraints on pre-order  $\preceq_p$

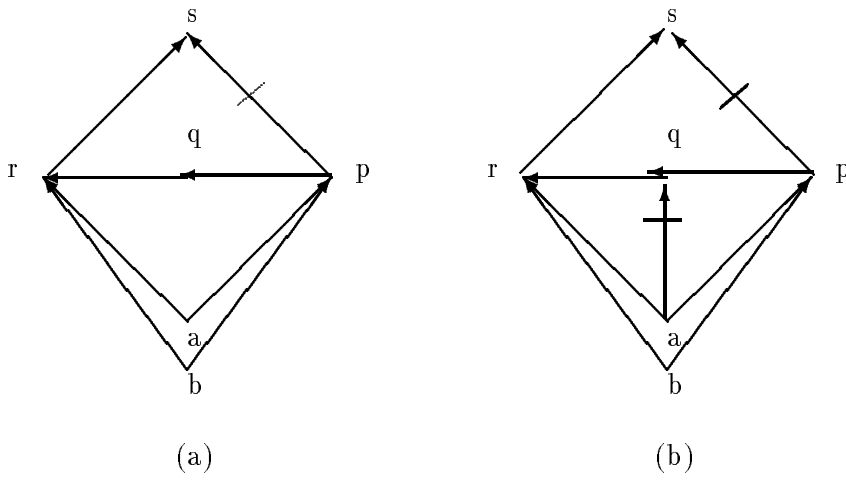


Figure 5: Specificity can be nonlocal.

former case, the arc from a to r is regarded as a *redundant* evidence, while, in the latter case, it is accorded the status of an *independent* evidence.

We explain briefly why the skeptical semantics <sup>[13]</sup> is what we call ground local. (See Section 2.6 for details.) For this purpose, we derive an ordering on the in-arcs of a property node  $p$  as a function of an individual node  $n$  from the reformulated definition of *defeasible preemption*<sup>[16]</sup>.

A word about the notation used.  $\pi(n, \tau_1, x, \tau_2, q)$  denotes a sequence of positive arcs from node  $n$  to node  $q$  via node  $x$ , and  $\pi(n, \sigma, q) \rightarrow p$  (resp.  $\pi(n, \sigma, q) \not\rightarrow p$ ) denotes a path from node  $n$  to node  $p$  ending with a positive (resp. negative) arc from node  $q$  to node  $p$ .  $\Gamma$  stands for an inheritance network and  $\Phi$  for its extension constructed so far.

A positive path  $\pi(n, \sigma, q) \rightarrow p$  is *preempted* in the context  $\langle \Gamma, \Phi \rangle$  iff there is a node  $x$  such that (i) either  $x = n$  or there is a path of the form  $\pi(n, \tau_1, x, \tau_2, q) \in \Phi$ , and (ii)  $x \not\rightarrow p \in \Gamma$ . This translates to the fact that inheritance of  $\neg p$  by  $n$  through the negative arc from node  $x$  to node  $p$  is stronger than the inheritance of  $p$  through the positive arc from node  $q$  to node  $p$ . Note that only the existence of the paths  $\sigma$ ,  $\tau_1$  and  $\tau_2$  is required.

A negative path  $\pi(n, \sigma, q) \not\rightarrow p$  is *preempted* in the context  $\langle \Gamma, \Phi \rangle$  iff there is a node  $x$  such that (i) either  $x = n$  or there is a path of the form  $\pi(n, \tau_1, x, \tau_2, q) \in \Phi$ , and (ii)  $x \rightarrow p \in \Gamma$ . This translates to the fact that inheritance of  $p$  by  $n$  through the positive arc from node  $x$  to node  $p$  is stronger than the inheritance of  $\neg p$  through the negative arc from node  $q$  to node  $p$ . Here again, the details of the paths  $\sigma$ ,  $\tau_1$  and  $\tau_2$  are unimportant.

These observations enable representation of the “skeptical” specificity relationship <sup>[13]</sup> by ordering in-arcs into node  $p$  as a function of the individual node  $n$  (rather than being independent of  $n$ ).

Similar translations of the inheritance networks to defeasible logic <sup>[3]</sup> and to stratified logic programs <sup>[9]</sup> have been carried out.

## 2.5 Characteristics of the Local Semantics

Here we assume, without loss of generality, that the specificity relation  $\prec_p$  for each property node  $p$  consists entirely of the inferred component. It is straight-forward to incorporate the preferential component.

### 2.5.1 Skeptical Semantics

It is not necessary for the set of constraints given in Section 2.3.1 be satisfiable for an arbitrary directed graph. In fact,

**Lemma 1** *There exist directed graphs that are unsatisfiable.*

However, even though arbitrary directed graphs can be unsatisfiable the restriction of *acyclicity* ensures that inheritance networks are satisfiable.

**Theorem 1** *Every preferential inheritance network has a minimal skeptical model.*

We can in fact strengthen this result to

**Theorem 2** *Every preferential inheritance network admits a unique minimum skeptical model.*

### 2.5.2 Credulous Semantics

We show the satisfiability of the constraints given in Section 2.3.6 and Section 2.3.7 for inheritance networks. That is,

**Theorem 3** *Every preferential inheritance network has a minimal credulous model.*

### 2.5.3 Local Specificity

We summarize certain important features of local specificity given in Section 2.4 arising as a consequence of defining the specificity relation and the inheritance relation mutually recursively.

The constraints to be satisfied for the inheritance of a property by an individual, and the constraints to be satisfied for a property to provide more specific information

than another property, are practically identical. So, if an inheritance network  $\Gamma$  is augmented with a new individual node  $i$  and a new positive arc  $\mathbf{E}^+(i, q)$ , then  $q$  is more specific than  $p$  iff  $i$  inherits  $p$ . Similarly,  $q$  is potentially disjoint from  $p$  iff  $i$  inherits  $\neg p$ . This provides an intuitive justification for our notion of specificity.

An individual  $c$  inherits a property  $p$  if there is a direct arc from  $c$  to  $p$ , or if  $c$  inherits  $p$  through a child of  $p$ . An individual  $c$  can be thought to inherit a property  $p$  via a node  $q$ , if  $c$  inherits  $q$ , and among all the children of  $p$  that  $c$  inherits,  $q$  is the most specific. Additionally, if individual nodes  $c$  and  $d$  inherit from nodes  $q$  and  $r$ , and  $q$  is more specific than  $r$ , then both  $c$  and  $d$  inherit from  $q$  in preference to  $r$ . The property inherited by  $c$  can differ from that inherited by  $d$  only if there is another “conflicting” node  $t$  which is not dominated by  $q$  for  $c$  or  $d$ . In particular,  $q$  being more specific than  $r$  does not depend on  $c$  or  $d$ .

All these observations follow directly from the semantics specification in Section 2.3 and the local specificity in Section 2.4.

## 2.6 Locality

The locality principle is at the heart of language definition techniques <sup>[1]</sup> and is also fundamental to efficient knowledge representation <sup>[30]</sup>. For instance, a propositional logic language can be specified using a context-free grammar, and its semantics defined by giving truth-tables for the boolean operators and a truth assignment to all proposition symbols in the language. This semantics specification exhibits locality property by definition. The meaning of an arbitrary formula depends only on the meanings of its “immediate” subformulae in the derivation tree, and does not depend on how that meaning is arrived at. The locality property of the semantics allows a “context-free” application of an inference rule. Given a Prolog rule and the fact that all the body literals in the rule are proven, the head of the rule can be asserted without any further deliberations about what else is present or provable from the program. Similarly, given the credibility of each MYCIN rule, the certainty factors associated with the body literals, and the combination function, the certainty of the head can be determined <sup>[30]</sup>.

In probabilistic reasoning <sup>[30]</sup>, conditional probabilities are not amenable to such “local” treatment. This is because a conditional probability  $P(p | q)$  does not

impose a definitive constraint on the probability of  $p$ , given the probability of  $q$ , in the presence of some additional evidence  $e$ . Determining the applicability of each arc of an inheritance network *independently* falls prey to the same problem because the context in which the arc appears has not been considered explicitly [8, 27]. However, belief propagation in causal networks requires only local computation. The value of a belief parameter associated with a node depends only on the corresponding parameter values associated with its children/parent nodes and the local conditional probability matrix. Similarly, it is possible to specify the contextual “support” and “defeat” information in an inheritance network by relating the meanings of a node and its neighbors using local constraints as shown in Section 2.3. Furthermore, the belief propagation algorithm requires a number of passes through the belief network for convergence to a stable assignment. In contrast, the local semantics of inheritance network can be computed in a monotonic fashion in just two passes, because arcs of the form  $\neg p$  to  $q$  (resp.  $\neg q$ ) are absent, as described in Appendix B.

In tree-structures, the meaning of a tree can be defined in terms of the meanings of its “immediate” subtrees, which are always disjoint from each other. In the case of DAG structures such a recursive decomposition is not always possible. However the concept of locality can still be defined using the notion of neighborhood as described below. (These ideas were first reported in [23]. They are reproduced here for the sake of completeness.)

### 2.6.1 Locality in Inheritance Networks

We can formalize the notion of *local* semantics in the context of inheritance networks by making precise the statement — *the meaning of a node is constrained only by the meaning assigned to the nodes in its immediate neighborhood.*

Let  $\Gamma$  be a *preferential inheritance network*. A node  $q$  is a *neighbor* of a node  $p$  if there is a direct arc between  $p$  and  $q$ . A *node-environment*  $\Gamma_p$  corresponding to the node  $p$  of an inheritance network  $\Gamma$  is the subnetwork of  $\Gamma$  consisting of the node  $p$  and all its neighbors. A *specificity relation* wrt a node  $p$  is a binary asymmetric relation among the arcs of the node-environment of  $p$ . We say that there exists an *isomorphism* between the node-environments  $\Gamma_p^1$  and  $\Gamma_q^2$  if there exists a bijection between the nodes of  $\Gamma_p^1$  and  $\Gamma_q^2$ , such that the *adjacency relation* among the nodes

and the *specificity relation* among the arcs of the node-environments are *preserved*. For our purposes, a semantics of inheritance networks is an assignment of sets to the nodes of the networks that satisfy a set of constraints  $\mathcal{C}$ . (See Section 2.3 for an example.) When the set of constraints  $\mathcal{C}$  is implicit in the discussion, we just say that the semantics of a network is an assignment that *satisfies* the network. A satisfying assignment to the network  $\Gamma$  is denoted by  $\mathcal{M}_\Gamma$ , and the meaning that  $\mathcal{M}_\Gamma$  assigns to a node  $p$  of  $\Gamma$  is denoted by  $\mathcal{M}_\Gamma(p)$ . The notation  $\mathcal{M}_\Gamma[\mathcal{M}_\Gamma(p) \leftarrow S]$  stands for the assignment of sets to the nodes of  $\Gamma$  obtained from  $\mathcal{M}_\Gamma$  by replacing  $\mathcal{M}_\Gamma(p)$  with  $S$ .

**Definition 3** *A semantics of inheritance networks exhibits the property of locality if for any pair of networks  $\Gamma^1$  and  $\Gamma^2$  containing isomorphic node-environments  $\Gamma_p^1$  and  $\Gamma_q^2$ , whenever the meaning assigned to all the neighbors of node  $p$  in  $\Gamma_p^1$  is identical to the meaning assigned to the “corresponding” neighbors of node  $q$  in  $\Gamma_q^2$ , it is the case that the assignment  $\mathcal{M}_\Gamma^2[\mathcal{M}_\Gamma^2(q) \leftarrow \mathcal{M}_\Gamma^1(p)]$  to the network  $\Gamma^2$  satisfies the network  $\Gamma^2$ .*

The theories of inheritance given in Section 2.3 have the locality property by definition. We further illustrate the concept of locality by showing that the credulous semantics [39] is *nonlocal*. Figure 6(1) and Figure 6(2) depict two networks, and their “upward” meanings [39]. (Note that this example has been chosen to illustrate the importance of a “suitable view” of the semantics to ascertain its locality.) Each node  $p$  is assigned a pair of sets of nodes  $(p^+, p^-)$ , where  $q \in p^+$  stands for  $q$  possesses  $p$  and  $q \in p^-$  for  $q$  possesses  $\neg p$ . Figure 6(3) depicts the application of the locality condition to  $p$ . The assignment to the network in Figure 6(3) does not satisfy the network according to Touretzky’s semantics because  $n$  and  $b$  do not inherit  $p$  similarly. This shows that the semantics [39] may be nonlocal.

But one may argue that the semantics may in fact be local if only one “looked at it in the right perspective”. For instance, one may consider the assignment of sets  $(p^+, p^-)$  to node  $p$  to mean the following:  $q \in p^+$  stands for  $p$  possesses  $q$  (instead of  $q$  possesses  $p$  as above) and  $q \in p^-$  for  $p$  possesses  $\neg q$  (instead of  $q$  possesses  $\neg p$  as above). In that case, one can verify that the satisfying “downward” assignments to the networks in Figure 6 do not violate locality. But it can be shown that this view

of the semantics is still nonlocal by considering the assignments to networks given in Figure 7. Figure 7(1) displays a model of the network, in which it is not known whether  $t$  is a  $w$ . The assignment in Figure 7(2) is a model of the modified network, in which  $t$  is a  $w$  by virtue of being a  $b$ . However, the assignment in Figure 7(3) obtained by the application of the locality condition is not a model according to the semantics [39].

Another consequence of the lack of locality property is the counter-intuitive interpretation of certain network topologies [33, 40]. Consider the following example [40]: *C's and D's are E's. B's are C's and D's. A's are B's and not D's.* According to [39], there is an ambiguity about whether *A's are E's*; while the semantics in Section 2.3 accords support to the conclusion — *A's are E's*, which is intuitively satisfactory.

It must also be mentioned that a translation is said to be local if incremental changes to the network results in incremental changes to its translation [32]. For instance, addition of an arc to a network results in the addition of a fact to the translation [32]. This is because complex meta-level axioms capture the computation of specificity relations from the assertions describing the network. On the other hand, our notion of locality is *semantic* in that it is based on the relationship among the meanings of adjacent nodes. We regard a semantics of inheritance networks as *local* if the meaning of a node in the network depends only on the meanings of its immediate neighbours (where the meaning of a node is tacitly assumed to be a pair of sets of nodes and not a pair of sets of paths).

It has also been assumed that the specificity relation used to disambiguate inheritance conflicts wrt a common property by an individual via a set of nodes, *depends only on the set of nodes, and not on the individual inheritance paths*. But one can argue that this is counter-intuitive, as illustrated below. Consider Figure 5(a) [38]. A  $p$  is a  $q$  and a  $q$  is an  $r$ , and hence,  $p$  is more specific than  $r$ . One can then view the arcs from  $a$  and  $b$  to  $r$  as being redundant. So, both  $a$  and  $b$ , inherit  $\neg s$  from  $p$ . On the other hand, in Figure 5(b) [38],  $a$  is not a  $q$ . So the arc from  $a$  to  $r$  can be interpreted as contributing an independent evidence for  $a$  to be an  $r$ . In this situation, there is an ambiguity about whether  $a$  is an  $s$ , because we cannot disambiguate the conflicting evidence for inheriting  $s$  through  $r$  and for inheriting

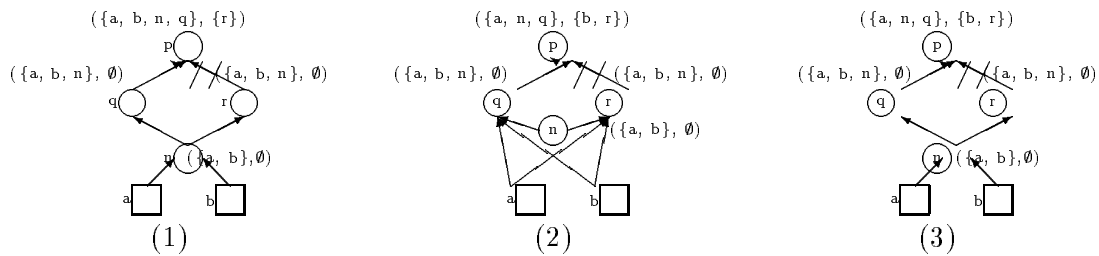


Figure 6: NonLocality of Touretzky's Semantics I

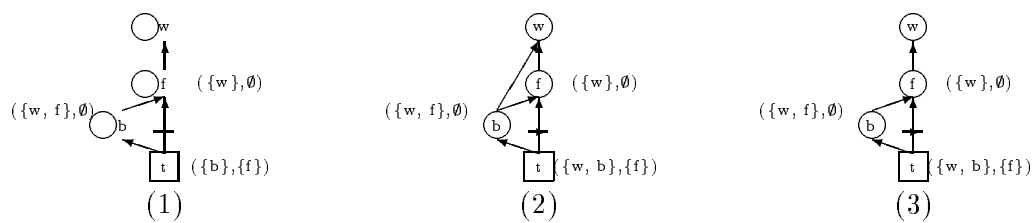


Figure 7: NonLocality of Touretzky's Semantics II

$\neg s$  through  $p$ . Thus, we see that even though both  $a$  and  $b$  possess properties  $p$  and  $r$ ,  $p$  is more specific than  $r$  wrt  $a$ , while  $p$  and  $r$  are unrelated wrt to  $b$ . This problem stems from the popular view that an explicit arc which is implicitly supported by the network is regarded as redundant, and not as contributing an additional independent support.

This example also shows that the semantics of heterogeneous networks [14] and homogeneous networks [13] are intrinsically *nonlocal* according to our definition. Furthermore, we observe that the specificity relation wrt node  $s$  for individuals  $a$  and  $b$  are different — node  $p$  is more specific than node  $r$  wrt node  $a$ , while node  $p$  and node  $r$  are unrelated wrt to node  $b$ . This motivates us to generalize the notion of locality to *ground locality* where the specificity relation is further parameterized by the “inheriting” node. This extension is similar in spirit to the generalization of the notion of stratification to local stratification in logic programming [31].

The *ground specificity relation* for node  $b$  wrt node  $p$  is a binary asymmetric relation among the arcs of the node-environment of  $p$  for  $b$ . We say that there exists a *ground isomorphism* between the node-environments  $\Gamma_p^1$  and  $\Gamma_q^2$  if there exists a bijection between the nodes of  $\Gamma_p^1$  and  $\Gamma_q^2$ , such that the *adjacency relation* among the nodes, and for all nodes  $b$ , the *ground specificity relation* among the arcs of the node-environments are *preserved*.

**Definition 4** *A semantics of inheritance networks exhibits the property of ground locality if for any pair of networks  $\Gamma^1$  and  $\Gamma^2$  containing ground isomorphic node-environments  $\Gamma_p^1$  and  $\Gamma_q^2$ , whenever the meaning assigned to all the neighbors of node  $p$  in  $\Gamma_p^1$  is identical to the meaning assigned to the “corresponding” neighbors of node  $q$  in  $\Gamma_q^2$ , it is the case that the assignment  $\mathcal{M}_\Gamma^2[\mathcal{M}_\Gamma^2(q) \leftarrow \mathcal{M}_\Gamma^1(p)]$  to the network  $\Gamma^2$  satisfies the network  $\Gamma^2$ .*

In conjunction with the specificity notion described in Section 2.4, the constraints in Section 2.3.3 capture the skeptical semantics [13]; and the constraints given in Section 2.3.6 capture the credulous semantics [15]. Thus, these theories are ground local according to our definition. Similarly, the family of inheritance theories [35, 36] are ground local. However, the semantics [39] is still *ground non-local* as it is impossible to define a satisfactory ordering on the in-arcs of node  $p$

in Figure 6(2), and on the out-arcs of node  $t$  in Figure 7(1). In Figure 6(2), the orderings  $\prec_p^a$  and  $\prec_p^b$  for the individuals  $a$  and  $b$  respectively cannot be independent because of coupling; while in Figure 7(1), a suitable ordering  $\prec_t^a$  cannot exist.

## 2.7 Inheritance Algorithms

In Appendix B, we give an inheritance algorithm to compute a minimal model of the network according to the specification in Section 2.3.7. The inheritance algorithms corresponding to other local theories given in Section 2.3 can be obtained by simple modifications to portions of this algorithm.

We have also designed special-purpose algorithms to compute

- whether or not an individual  $n$  inherits a property  $p$ .
- whether or not an individual  $n$  inherits a property  $\neg p$ .
- all properties possessed by an individual  $n$ .
- all individuals that possess a property  $p$ .

The general strategy for designing these special purpose algorithms is as follows: We take the whole inheritance network, and mark relevant portions of the network by looking at the form of the query. To compute all the properties of an individual  $i$ , we mark all the ancestors of  $i$ ; to compute all individuals that possess a property  $p$ , we mark all descendants of  $p$ . To determine whether an individual  $i$  possesses  $p$ , we temporarily mark all the ancestors of  $i$  first, and then assign permanent marks to those nodes that are descendants of  $p$  too. The marking procedure gets a little complex, when the inheritance algorithm supports contrapositive reasoning. The general inheritance algorithm is invoked only on this smaller marked network to answer the query.

Here, we briefly describe the computational complexity of these inheritance algorithms.

After a careful analysis, it is clear that a realistic characterization of their complexity should be given in terms of parameters like the total number of nodes  $n$ , the total number of edges  $e$ , the depth of the network  $h$ , the maximum number of

children  $c$ , the maximum number of descendants  $d$  etc. In general, the maximum size of the sets computed as the meaning of a node is  $\mathbf{O}(n)$ . But, in the absence of contrapositive reasoning, the maximum size of the sets that arise as the meaning of a node is only  $\mathbf{O}(d)$ . The worst-case time complexity for computing a minimal model can be obtained by observing the following facts: The cost of the initialization step is  $\mathbf{O}(n)$ ; that of degree calculation is  $\mathbf{O}(e)$ ; that of determining if  $q^+(r)$  holds of two children  $q$  and  $r$  of the same node  $p$  is  $\mathbf{O}(d)$ ; that of computing the specificity relation  $\prec_p$  at each node  $p$  is  $\mathbf{O}(c^2d)$ ; that of computing the meaning of a property node is  $\mathbf{O}(c^2d^2)$ . Thus, the time complexity of the inheritance algorithm is  $\mathbf{O}(n c^2d^2)$ . Similarly, one can argue that the parallel time complexity of the algorithm is  $\mathbf{O}(h c^2d^2)$ .

In contrast, computing credulous expansion according to [10, 39] is NP-hard. However, the skeptical semantics [13] and the inheritance theories [36] are polynomial-time computable. To be precise, the time complexity of the parallel marker passing algorithm [13] for determining whether  $p$  inherits  $q$  is proportional to the depth of the network and the number of conflicted nodes. (A direct comparison of this result with our inheritance algorithm is difficult because of the differences in the form of the query and the underlying machine architecture, which, for instance, permits concurrent writes and bit-vector representation of sets etc.) Furthermore, given a polynomial-time algorithm to determine specificity, a polynomial-time algorithm to compute *ideally skeptical semantics* (i.e., the intersection of all credulous extensions) can be obtained [17, 35].

### 3 Conclusions and Future Work

A satisfactory solution to the inheritance problem is necessary to build a large system that can represent and utilize human experience and expertise for problem-solving in diverse areas like medical diagnosis, intelligent query answering, planning, decision-making, and common-sense reasoning about the world in general. In this paper, we proposed a family of local inheritance theories that are conceptually simple and computationally efficient. The design space for inheritance reasoners [40] has been mapped out, and a number of choices available to the designers of such

systems considered. It is emphasized that tractability and efficiency aspects are important for the design of viable inheritance reasoners [34]. Because the language of inheritance network is not sufficiently expressive to incorporate the various intuitions about inheritance, we propose to investigate alternative annotations to the nodes and to the arcs of the network to accommodate these various interpretations of inheritance networks in a unified framework. Such a language, we believe, will benefit a knowledge engineer because it will permit him to decide what knowledge can be encoded in the system, and give him understandable formal guarantees about the quality of the conclusions that will be generated [2].

## References

- [1] A. V. Aho and J. D. Ullman, *Principles of Compiler Design*, Addison-Wesley Publishing Co., Inc., 1977.
- [2] F. Bacchus, A modest, but semantically well-founded, inheritance reasoner, In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1104-1109, 1989.
- [3] D. Billington, K. De Coster, and D. Nute, A modular translation from defeasible nets to defeasible logics, *Journal of Experimental and Theoretical Artificial Intelligence*, **2**, 151-177 (1990).
- [4] J. P. Delgrande, An approach to default reasoning based on a first-order conditional logic: Revised report, *Artificial Intelligence*, **36**, 63-90 (1988).
- [5] D. Etherington and R. Reiter, On inheritance hierarchies with exceptions, In *Proceedings of the Second National Conference on Artificial Intelligence*, 104-108, 1983.
- [6] S. E. Fahlman, *NETL: A System for Representing and Using Real-World Knowledge*, The MIT Press, Cambridge, MA, 1979.
- [7] H. Geffner and J. Pearl, A framework for reasoning with defaults, In *Proceedings of Society for Exact Philosophy Conference*, 1988.

- [8] H. Geffner, Default Reasoning: Causal and Conditional Theories, *Ph.D. Dissertation*, University of California at Los Angeles, 1989.
- [9] E. Gregoire, Skeptical theories of inheritance and nonmonotonic logics, In *Proceedings of the Fourth International Symposium on Methodologies for Intelligent Systems*, 430-438, 1989.
- [10] H. Geffner and T. Verma, Inheritance = Chaining + Defeat, In *Proceedings of the Fourth International Symposium on Methodologies for Intelligent Systems*, 411-418, 1989.
- [11] H. Geffner, *Personal Communication*.
- [12] B. Haugh, Tractable theories of multiple defeasible inheritance in ordinary nonmonotonic logics, In *Proceedings of the Seventh National Conference on Artificial Intelligence*, 421-426, 1988.
- [13] J. Horty, R. Thomason, and D. Touretzky, A skeptical theory of inheritance in nonmonotonic semantic networks, *Artificial Intelligence*, **42**, 311-348 (1990).
- [14] J. Horty and R. Thomason, Mixing strict and defeasible inheritance, In *Proceedings of the Seventh National Conference on Artificial Intelligence*, 427-432, 1988.
- [15] J. Horty, A credulous theory of mixed inheritance, In *Inheritance Hierarchies in Knowledge Representation*, M. Lenzerini, D. Nardi, and M. Simi (eds), John Wiley and Sons, 1990.
- [16] J. Horty, Some direct theories of nonmonotonic inheritance, In *Handbook of Logic in Artificial Intelligence and Logic Programming*, D. Gabbay, C. Hogger and J. A. Robinson (eds.), Oxford University Press, 1993.
- [17] H. A. Kautz and B. Selman, Hard problems for simple default logics, In *Proceedings of the First International Conference on Knowledge Representation and Reasoning*, 189-197, 1989.

- [18] T. Krishnaprasad, M. Kifer, and D. S. Warren, On the declarative semantics of inheritance networks, In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1093-1098, 1989.
- [19] T. Krishnaprasad and M. Kifer, An evidence-based framework for a theory of inheritance, In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1099-1103, 1989.
- [20] T. Krishnaprasad, M. Kifer, and D. S. Warren, A circumscriptive semantics of inheritance networks, In *Proceedings of the Fourth International Symposium on Methodologies for Intelligent Systems*, 448-457, 1989.
- [21] K. Thirunarayan, An analysis of property-flow view vs individual-flow view of inheritance, In *Proceedings of the Sixth International Symposium on Methodologies for Intelligent Systems*, 256-265, 1991.
- [22] K. Thirunarayan and M. Kifer, A theory of nonmonotonic inheritance based on annotated logic, *Artificial Intelligence*, **60**, 23-50 (1993).
- [23] K. Thirunarayan, Locality in inheritance networks, In: *Information Processing Letters*, **46(6)**, 263-268 (1993).
- [24] H. Levesque, A knowledge-level account of abduction, In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1061-1067, 1990.
- [25] J. McCarthy, Applications of circumscription to formalizing common-sense knowledge, *Artificial Intelligence*, **28**, 89-116 (1986).
- [26] M. E. Morris, Meta-level reasoning for conflict resolution in backward chaining, In *Principles of Expert Systems*, A. Gupta and B. E. Prasad (eds.), IEEE Press, 175-180, 1988.
- [27] E. Neufeld, Notes on "A clash of intuitions", *Artificial Intelligence*, **48**, 225-240 (1991).

- [28] L. Padgham, A model and representation for type information and its use in reasoning with defaults, In *Proceedings of the Seventh National Conference on Artificial Intelligence*, 409-414, 1988.
- [29] L. Padgham, Negative reasoning using inheritance, In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1086-1092, 1989.
- [30] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann Publishers Inc., San Mateo, 1988.
- [31] T. C. Przymusinski, On the declarative semantics of stratified deductive databases and logic Programs, In *Foundations of Deductive Databases and Logic Programming*, J. Minker (ed.), Morgan Kaufmann, 193-216, 1988.
- [32] H. Przymusinska and M. Gelfond, Formalization of inheritance reasoning in autoepistemic logic, *Fundamenta Informaticae*, **13(4)**, 403-444 (1990).
- [33] E. Sandewall, Nonmonotonic inference rules for multiple inheritance with exceptions, *Proceedings of the IEEE*, **74(10)**, 1345-1353, 1986.
- [34] B. Selman and H.J. Levesque, The tractability of path-based inheritance, In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1140-1145, 1989.
- [35] L. A. Stein, Skeptical inheritance: Computing the intersection of credulous extensions, In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1153-1158, 1989.
- [36] L. A. Stein, Resolving ambiguity in nonmonotonic inheritance hierarchies, *Artificial Intelligence*, **55**, 259-310 (1992).
- [37] R. Thomason, J. Horty, and D. Touretzky, A calculus for inheritance in monotonic semantic nets, in: Z. Ras and M. Zemankova (eds.), *Proceedings of the Second International Symposium on Methodologies for Intelligent Systems*, 280-287, 1987.

- [38] R. Thomason and J. Horty, Logics for inheritance theory, In *Non-Monotonic Reasoning*, M. Reinfrank, J. de Kleer, M. Ginsberg, and E. Sandewall (eds.), Springer-Verlag, 1989.
- [39] D. Touretzky, *The Mathematics of Inheritance Systems*, Morgan Kaufmann, Los Altos, 1986.
- [40] D. Touretzky, J. Horty, and R. Thomason, A Clash of intuitions: The current state of nonmonotonic multiple inheritance systems, In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 476-482, 1987.

## A Proofs of Theorems

**Definition 5** *The degree of a node in a preferential inheritance network is the length of a longest directed path from a leaf node to the node. That is, the degree of a leaf node is 0. The degree of a nonleaf node is one more than the maximum of the degrees of its children.*

**Definition 6** *The depth of a node in a preferential inheritance network is the length of a longest directed path from the node to a terminal node. (A terminal node is a property node with no out-arcs.) That is, the depth of a terminal node is 0. The depth of a nonterminal node is one more than the maximum of the depths of its parents.*

**Lemma 1** *There exist directed graphs that are unsatisfiable.*

**Proof:** Consider the cyclic network in Figure 8 from [15]. There is an arc from  $p$  to  $q$  so  $q^+(p)$  must hold. In the absence of any prospect for a conflict at nodes  $r$  and  $t$ ,  $p$  can propagate from node  $q$  to nodes  $r$  and  $t$ . Hence,  $r^+(p)$  and  $t^+(p)$  must hold. There is an arc from  $t$  to  $p$ , hence,  $p^+(t)$  also holds. Thus, according to our definition of specificity,  $p$  is more specific than  $t$  and  $t$  is more specific than  $p$ .

Now consider the individual node  $n$ .  $p^+(n)$  must hold because there is an arc from  $n$  to  $p$ . As both positive and negative arcs are incident on  $q$  there is potential for a conflict to occur. In particular, we want to determine whether or not  $n$  is a  $q$ .

We analyze the two possible scenerios — the case when  $t^+(n)$  holds and the case when  $t^+(n)$  does not hold. If  $t^+(n)$  holds then both  $q^+(n)$  and  $q^-(n)$  cannot hold. This is because, both  $\langle p, q \rangle \prec_q \langle t, q \rangle$  and  $\langle t, q \rangle \prec_q \langle p, q \rangle$  hold, and hence, there is neither a *maximal* support for  $q^+(n)$  nor a *maximal* support for  $q^-(n)$ . If  $q^+(n)$  does not hold, then there is *no* support for  $r^+(n)$ , and hence, there is *no* support for  $t^+(n)$ . Thus  $t^+(n)$  cannot hold which is a contradiction. But, if we assume the other possibility that  $t^+(n)$  does not hold, then there is clearly no chance of a conflict at  $q$  wrt  $n$ . Hence,  $q^+(n)$ ,  $r^+(n)$  and  $t^+(n)$  must all hold which is contrary to our assumption. So the network in Figure 8 is unsatisfiable according to our semantics. •

**Theorem 1** *Every preferential inheritance network has a minimal skeptical model.*

**Proof:** We prove that there exists an interpretation that satisfies the constraints given in Section 2.3.2. (One can modify this interpretation trivially to construct a model that satisfies constraints in Section 2.3.3.)

Let  $\mathcal{C}$  denote the set of constraints,  $P_i$  denote the set of property nodes with degree  $i$  or less, and  $\mathcal{C}_i$  the set of constraints corresponding to nodes in  $P_i$ . Recall that for every property node  $p$  in the network, there are two *iff*-constraints — corresponding to  $p^+$  and  $p^-$  respectively. We can impose a natural ordering on  $\mathcal{C}$ , based on the DAG structure of the network. In particular,  $\mathcal{C}$  can be stratified based on the degree of the property node associated with the constraint. We show, by induction on degree, that a model  $\mathcal{M}$  for  $\mathcal{C}$  can be constructed from the chain of partial models  $\mathcal{M}_0, \mathcal{M}_1, \dots$  that satisfy  $\mathcal{C}_0, \mathcal{C}_1, \dots$  respectively. In particular, the meaning assigned to nodes in  $P_i$  by  $\mathcal{M}_i$  and  $\mathcal{M}$  are identical. In what follows we assume that the local specificity relation required at any node is completely determined by the inheritance relation computed for its children thus far. The complete proof involves a straight-forward simultaneous induction.

**Basis:**  $\mathcal{C}_0$  is satisfied by the interpretation  $\mathcal{M}_0$  that maps every property node  $p$  of degree 0 to  $(\emptyset, \emptyset)$ . This is because the right hand side of the constraints corresponding to such nodes is identically **false**. In addition, for the same reason,  $\mathcal{M}$  must also assign the same meaning to nodes in  $\mathcal{P}_0$ .

**Induction Hypothesis:** Assume that  $\mathcal{C}_i$  is satisfied by the interpretation  $\mathcal{M}_i$  and the meaning assigned to nodes in  $P_i$  by  $\mathcal{M}_i$  and  $\mathcal{M}$  are identical.

**Induction Step:** We show that  $\mathcal{M}_i$  can be extended to  $\mathcal{M}_{i+1}$  that satisfies  $\mathcal{C}_{i+1}$ . First of all observe that the constraints in the set  $\mathcal{D}_{i+1} = \mathcal{C}_{i+1} - \mathcal{C}_i$ , correspond to property nodes of degree  $i + 1$ . The right hand side of the constraints in  $\mathcal{D}_i$  contain references only to nodes in  $P_i$  because inheritance networks are acyclic. This implies that meaning of each node  $p$  of degree  $i + 1$  can be completely determined using the constraints for  $p$  in  $\mathcal{D}_{i+1}$ , the partial model  $\mathcal{M}_i$ , and the local specificity information w.r.t. children of  $p$  (which is assumed to be known from the simultaneous induction hypothesis about the the meaning of nodes in  $P_i$ ). Thus the partial model  $\mathcal{M}_{i+1}$  extends  $\mathcal{M}_i$  by interpreting nodes in  $P_{i+1} - P_i$  such that they satisfy  $\mathcal{D}_{i+1}$ . Observe that these new assignments are not relevant to the satisfiability of the constraints in  $\mathcal{C}_i$ . Thus we see that a model  $\mathcal{M}$  satisfying  $\mathcal{C}$  can be constructed iteratively by determining  $\cup_i \mathcal{M}_i$  as indicated. •

**Theorem 2** *Every preferential inheritance network admits a unique minimum skeptical model.*

**Proof:** We show that the constraints in Section 2.3.2 can be satisfied in only one way, by contradiction.

Assume that there exist two distinct minimal models —  $\mathcal{M}_1$  and  $\mathcal{M}_2$  — satisfying the constraints in Section 2.3.2 for an inheritance network.  $\mathcal{M}_1$  and  $\mathcal{M}_2$  must agree on the meaning of nodes in  $\mathcal{P}_0$  because the right hand side of the constraints corresponding to such nodes is identically **false**. So, if  $\mathcal{M}_1$  and  $\mathcal{M}_2$  differ, then there must exist a smallest  $i$  such that they differ on a property node  $p$  in  $\mathcal{P}_i$  where  $i > 0$ , and they agree on all nodes in  $\cup_{j < i} \mathcal{P}_j$ . But this is a contradiction because the meaning of the nodes in  $\mathcal{P}_i$  is completely determined by the meaning of the nodes in  $\cup_{j < i} \mathcal{P}_j$  and the local constraints in  $\mathcal{D}_i$ . Hence every inheritance network has a unique minimum skeptical model. •

**Theorem 3** *Every preferential inheritance network has a minimal credulous model.*

**Proof:** We prove that there exists an interpretation that satisfies the constraints of Section 2.3.7. (This proof can be modified trivially to demonstrate the same for the constraints in Section 2.3.6.)

Analogously to the skeptical case, we try to impose a natural ordering on the set of constraints  $\mathcal{C}$  based on the DAG structure of the network. However this is more complicated in the credulous case because, the body of a constraint can refer to the meaning of the node and its parents in addition to that of the children. So, for the purpose of stratifying the constraints, we associate with every property node  $p$  three predicates/sets —  $p_f^+$ ,  $p_f^-$ , and  $p_b^-$ . Intuitively,  $p_f^+$  (resp.  $p_f^-$ ) is that component of  $p^+$  (resp.  $p^-$ ) which is determined by the children of  $p$ , while  $p_b^-$  is that component of  $p^-$  which is determined by the parents of  $p$ . (The latter component is due to contrapositive reasoning.) The set  $S_d$  denotes the set of predicates of the form  $p_f^+$ ,  $p_f^-$  corresponding to property nodes with degree  $d$ , and the set  $S_{\max\text{degree}+h}$  denotes the set of predicates of the form  $p_b^-$  corresponding to property nodes of depth  $h$ . The constraints associated with a property node  $p$  of degree  $d + 1$  and depth  $h + 1$  can be expressed abstractly as follows where **cond** is some condition that depends on the bracketed predicates:

$$p_f^+(n) \text{ iff } \neg p_f^-(n) \wedge \text{cond}[\cup_{j \leq d} S_j](n).$$

$$p_f^-(n) \text{ iff } \neg p_f^+(n) \wedge \text{cond}[\cup_{j \leq d} S_j](n).$$

$$\neg[p_f^+(n) \wedge p_f^-(n)]$$

$$p_b^-(n) \text{ iff } \text{cond}[\cup_{j \leq d} S_j \cup \cup_{j \leq h} S_{(\max\text{degree}+j)}](n).$$

$$\neg[p_f^+(n) \wedge p_b^-(n)]$$

Let  $\mathcal{C}_i$  correspond to the constraints associated with the components in  $\cup_{j \leq i} S_j$ .  $\mathcal{C}$  can be stratified using the degree and the depth of the node corresponding to the left hand side of a constraint as follows, where  $\mathcal{D}_{i+1} = \mathcal{C}_{i+1} - \mathcal{C}_i$ :

$$\mathcal{D}_0 \ll \mathcal{D}_1 \ll \dots \ll \mathcal{D}_{\max\text{degree}+\max\text{depth}}$$

We show, by induction on the stratification level of a constraint, that a model  $\mathcal{M}$  for  $\mathcal{C}$  can be constructed from a chain of partial models  $\mathcal{M}_0, \mathcal{M}_1, \dots, \mathcal{M}_{\max\text{degree} + \max\text{depth}}$  that satisfy  $\mathcal{C}_0, \mathcal{C}_1, \dots, \mathcal{C}_{\max\text{depth}+\max\text{depth}}$  respectively. In particular, we see that the components of the meaning assigned to a node by  $\mathcal{M}$  can be obtained directly from the corresponding components assigned by  $\mathcal{M}_i$ .

**Basis:**  $\mathcal{C}_0$  is satisfied by the interpretation  $\mathcal{M}_0$  that maps each predicate/set in  $S_0$  to  $\emptyset$ . This is because the right hand side of the corresponding constraint is identically false.

**Induction Hypothesis:** Assume that  $\mathcal{C}_i$  is satisfied by the interpretation  $\mathcal{M}_i$ .

**Induction Step:** We show that  $\mathcal{M}_i$  can be extended to  $\mathcal{M}_{i+1}$  that satisfies  $\mathcal{C}_{i+1}$ .

First of all, let  $i + 1 \leq \text{maxdegree}$ . Observe that the right hand side of the constraints in  $\mathcal{D}_{i+1}$  contains an explicit reference to  $p_f^\pm$  and the **cond**[] refers to the predicates in  $\cup_{j \leq i} S_j$ . By induction hypothesis, the meaning of **cond**[] is fixed. If **cond**[( $n$ )] corresponding to  $p_f^+(n)$  (resp.  $p_f^-(n)$ ) is true then  $p_f^+(n)$  (resp.  $p_f^-(n)$ ) can hold. But the disjointness condition dictates that if both  $p_f^+(n)$  and  $p_f^-(n)$  can hold, then only one of  $p_f^+(n)$  or  $p_f^-(n)$  may hold — the choice between them being arbitrary. This implies that meaning of each node  $p$  of degree  $i + 1$  can be completely determined using the constraints for  $p$  in  $\mathcal{D}_{i+1}$ , the partial model  $\mathcal{M}_i$  and the local specificity information w.r.t. children of  $p$  (which is assumed from the simultaneous induction hypothesis about the the meaning of the children.) Thus the partial model  $\mathcal{M}_{i+1}$  extends  $\mathcal{M}_i$  by interpreting predicates in  $S_i$  as described. Note also that these new assignments are not relevant to the satisfiability of the constraints in  $\mathcal{C}_i$ , and hence,  $\mathcal{M}_{i+1}$  is a model for  $\mathcal{C}_{i+1}$ .

Now consider the remaining cases when  $i + 1 \geq \text{maxdegree}$ . Observe that the right hand side of the constraints in  $\mathcal{D}_{i+1}$  contains the **cond**[] that refers only to predicates in  $\cup_{j \leq i} S_j$ . By induction hypothesis, the meaning of **cond**[] is fixed and determines the predicates in  $S_{i+1}$ . Also note that whenever  $p_f^+(n)$  holds **cond**[( $n$ )] does not. So, the disjointness condition is automatically satisfied. Thus the interpretation  $\mathcal{M}_{i+1}$  extends  $\mathcal{M}_i$  to a model of  $\mathcal{C}_{i+1}$ .

So a model  $\mathcal{M}$  of  $\mathcal{C}$  can be constructed iteratively by determining  $\cup_i \mathcal{M}_i$  as indicated. •

A study of these proofs shows that Theorem 3 implies Theorem 1. In particular, the skeptical model  $\mathcal{M}_{\text{maxdegree}}$  is closely related to the credulous partial model  $\mathcal{M}_{\text{maxdegree}}$ .

## B Design of an Inheritance Algorithm

In Section 2.3, we gave a precise logical specification for when an individual  $n$  inherits a property  $p$ . We will now develop systematically, an algorithm to compute a minimal model of the network according to the specification in Section 2.3.7. The proof of correctness of this algorithm follows from the results in Appendix A.

To specify precisely the order in which the meaning of the nodes is computed, we make the following definition. An arc from an individual node to a property node is considered *strict*, while an arc from a property node to a property node is considered *defeasible*. A *path* is defined as a sequence of defeasible arcs in the network. The *degree* of a node is the length of the longest path into that node.

We describe the precise flow of individuals through the network acquiring properties now. There is both a forward (or upward) flow of individuals in the direction of the arrows, and a backward (or downward) flow of individuals in the reverse direction because of contraposition. The forward flow contains two components. One subcomponent is due to individuals inheriting properties positively through the positive arcs and the other subcomponent is due to individuals inheriting properties negatively through the negative arcs. The backward flow also contains two components, each contributing to the inheritance of properties negatively. One subcomponent is due to individuals flowing backwards through positive arcs by virtue of possessing a property negatively, and the other subcomponent is due to individuals flowing backward through negative arcs by virtue of possessing a property positively.

To be precise, the positive component of the meaning of a node  $p$ , i.e., the set  $p^+$ , is computed from the positive components of the meaning of its “positive” children minus the negative subcomponent of the meaning of  $p$ , i.e., a subset of  $p^-$  computed due to “more specific” classes. The negative component of the meaning of a node  $p$ , i.e., the set  $p^-$ , is the union of two subcomponents. The forward flow subcomponent of  $p^-$  is computed from the positive component of the meaning of all its “negative” children minus the positive subcomponent of  $p$ , i.e., the subset of  $p^+$  computed due to “more specific” classes. The backward flow subcomponent of  $p^-$  is computed from the positive component of the meaning of all its parents through the negative arcs, and from the negative component in the meaning of all

its parents through the positive arcs.

In what follows, we present a set-based algorithm for computing a minimal model of an inheritance network developed above. With each node  $n$  we associate a constant  $n$ , and with each property node  $p$  we associate a pair of sets  $(p_{\mathcal{E}}^+, p_{\mathcal{E}}^-)$ , where  $p_{\mathcal{E}}^+$  (resp.  $p_{\mathcal{E}}^-$ ) represents the set of individuals that inherit  $p$  (resp.  $\neg p$ ) in a model  $\mathcal{E}$ . Intuitively, the constant  $i$  associated with a node  $i \in \mathbf{I}$  represents an individual in the world, while the constant  $p$  associated with a node  $p \in \mathbf{P}$  represents a “hypothetical” typical individual of the collection that possesses property  $p$ . The computation is carried out for all individuals simultaneously. In case of an ambiguous network, one of several possible minimal models is chosen non-deterministically. (We use the keyword **pardo** explicitly in some of the **for**-loops, to indicate that the body of the loop can be executed for each value of the “index” variable, in parallel, simultaneously.)

We formalize the bottom-up processing of nodes in the ascending order of their degrees. The constant *max-degree* stands for the maximum degree of a node.

```

begin
  begin /* initialize-facts. */
    foreach property node  $p$  pardo  $(p^+, p^-) := (\{p\}, \emptyset)$ ;
    foreach positive arc  $\langle n, p \rangle$  do  $p^+ := p^+ \cup \{n\}$ ;
    foreach negative arc  $\langle n, p \rangle$  do  $p^- := p^- \cup \{n\}$ ;
  end;
  begin /* compute-forward-flow-component. */
    foreach degree  $d := 1$  to max-degree do
      foreach node  $p$  of degree  $d$  pardo
        compute-forward-flow-component-of-meaning-of- $p$ ;
    end;
  begin /* compute-backward-flow-component. */
    foreach degree  $d := \text{max-degree} - 1$  downto  $0$  do
      foreach node  $p$  of degree  $d$  pardo
        compute-backward-flow-component-of-meaning-of- $p$ ;
    end;
end;

```

We need some notation to refine this further. Recall that the arcs into each node are ordered. Let graph  $H_p$  stand for the diagram corresponding to the relation  $\prec_p$ . The nodes of  $H_p$  are children of  $p$ . There is an edge from a node  $r$  to a node  $q$  in  $H_p$  if either  $q^+(r)$  or  $\langle q, p \rangle \prec_p \langle r, p \rangle$ . If  $q^+(r)$  is true then  $r$  is more specific than  $q$ . Also, we refer to a node in relation to another node w.r.t.  $H_p$ , instead of the whole network, by prefixing a term with an “ $H_p$ ”. For instance, an edge in  $H_p$  will be referred to as an  $H_p$ -edge, the degree of  $q$  in  $H_p$  as the  $H_p$ -degree etc. Let  $p_q^+$  (resp.  $p_q^-$ ) denote the contribution of node  $q$  towards  $p^+$  (resp.  $p^-$ ) and  $p^+$  (resp.  $p^-$ ) denote the set of individuals that can possibly inherit  $p$  (resp.  $\neg p$ ). The set of individuals that inherit  $p$  ambiguously is  $p^+ \cap p^-$ . To choose a specific minimal model, we partition the set  $p^+ \cap p^-$  into two sets  $p_a^+$  and  $p_a^-$ . The set  $p_a^+$  (resp.  $p_a^-$ ) stands for the set of those individuals that inherit  $p$  ambiguously, but in the chosen model  $\mathcal{E}$  they inherit  $p$  (resp.  $\neg p$ ).

```

begin /* compute-forward-flow-component-of-meaning-of-p. */
  foreach  $H_p$ -degree  $d := 0$  to  $H_p$ -max-degree do
    begin
      foreach  $q$  of degree  $d$  and positive arc  $\langle q, p \rangle$  pardo
        begin /* compute-contribution-of-q-to- $p^+$ . */  $p_q^+ := q^+$ ;
          foreach negative arc  $\langle r, p \rangle$  such that  $\langle r, q \rangle$  is an  $H_p$ -edge do
             $p_q^+ := p_q^+ - r_{\mathcal{E}}^+$ ;
          end;
        foreach  $r$  of degree  $d$  and negative arc  $\langle r, p \rangle$  pardo
          begin /* compute-contribution-of-r-to- $p^-$ . */  $p_r^- := r^+$ ;
            foreach positive arc  $\langle q, p \rangle$  such that  $\langle q, r \rangle$  is an  $H_p$ -edge do
               $p_r^- := p_r^- - q_{\mathcal{E}}^+$ ;
            end;
          /* compute-individuals-that-can-inherit-p */  $p^+ := p^+ \cup \bigcup_{q: \langle q, p \rangle \in \mathbf{E}^+} p_q^+$ ;
          /* compute-individuals-that-can-inherit- $\neg p$ ; */  $p^- := p^- \cup \bigcup_{r: \langle r, p \rangle \in \mathbf{E}^-} p_r^-$ ;
          /* choose a minimal model in case of ambiguity. */  $\text{partition}(p^+ \cap p^-, p_a^+, p_a^-)$ ;
          /* compute-individuals-that-may-inherit-p; */  $p_e^+ := p^+ - p_a^-$ ;
          /* compute-individuals-that-may-inherit- $\neg p$ ; */  $p_e^- := p^- - p_a^+$ ;
        end;
      end;
    end;
  end;

```

```

end;
begin /* “compute-backward-flow-component-of-meaning-of-p” */
  foreach positive arc  $\langle p, q \rangle$  do  $p_e^- := p_e^- \cup (q_{\mathcal{E}}^- - p_e^+)$ ;
  foreach negative arc  $\langle p, q \rangle$  do  $p_e^- := p_e^- \cup (q_{\mathcal{E}}^+ - p_e^+)$ ;
   $p_{\mathcal{E}}^+ := p_e^+$ ;
   $p_{\mathcal{E}}^- := p_e^-$ ;
end;

```

Inheritance algorithms to compute minimal models of other theories in Section 2.3 can be obtained by modifying this algorithm. For example, the theory in Section 2.3.6 can be obtained by deleting “compute-backward-flow-component-of-meaning-of-p”. Similarly, one can compute the skeptical semantics by modifying the procedures “compute-individuals-that-*may*-inherit-p” and “compute-individuals-that-*may*-inherit- $\neg p$ ” in the obvious way.

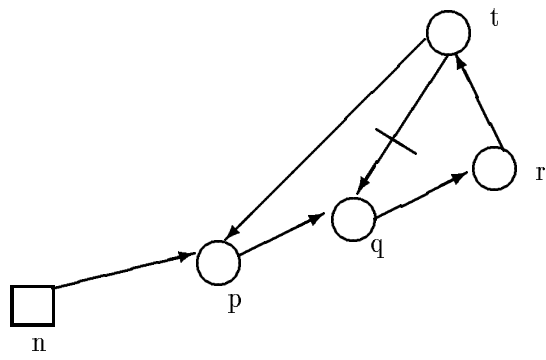


Figure 8: Unsatisfiable Cyclic Network