On the Complexity of Montonic Inheritance with Roles

by

Ramiro A. de T. Guerreira, Andrea S. Hemerly, Yoav Shoham

Department of Computer Science

Stanford University
Stanford, California 94305
On the Complexity of Monotonic Inheritance with Roles

Ramiro A. de T. Guerreiro
Rio Scientific Center
IBM Brazil
I? 0. Box 4624
2001, Rio de Janeiro, RJ

Andrea S. Hemery
Rio Scientific Center
IBM Brazil
P.O. Box 4624
20001, Rio de Janeiro, RJ

Yoav Shoham
Computer Science Dept
Stanford University
Stanford, CA 94305

ABSTRACT

We investigate the complexity of reasoning with monotonic inheritance hierarchies that contain, beside ISA edges, also ROLE (or FUNCTION) edges. A ROLE edge is an edge labelled with a name such as spouse-of or brother-of. We call such networks ISAR networks. Given a network with n vertices and m edges, we consider two problems: (P1) determining whether the network implies an isa relation between two particular nodes, and (P2) determining all isa relations implied by the network. As is well known, without ROLE edges the time complexity' of P1 is $O(m)$, and the time complexity of P2 is $O(n^3)$. Unfortunately, the results do not extend naturally to ISAR networks, except in a very restricted case. For general ISAR network we first give a polynomial algorithm by an easy reduction to propositional Horn theory. As the degree of the polynomial is quite high ($O(n^3)$ for P1, $O(nm^3)$ for P2), we then develop a more direct algorithm. For both P1 and P2 its complexity is $O(n^3 + m^3)$. Actually, a finer analysis of the algorithm reveals a complexity of $O(nr(\log r) + n^2r + n^3)$, where r is the number of different ROLE labels. One corollary is that if we fix the number of ROLE labels, the complexity of our algorithm drops back to $O(n^3)$.

1. INTRODUCTION

Inheritance systems are a common framework for representing knowledge, in both AI and the database community. In these systems objects are organized hierarchically, and properties of objects are inherited by those below them in the hierarchy. For example, if it is recorded in this knowledge base that mothers are parents and that parents are responsible people, it may be concluded that mothers too are responsible.

As is well known, an inheritance system may be represented by a directed graph. The vertices in the graph are all of the same kind, and they each represent a class of objects. Arcs, on the other hand, come in several varieties, and there has been less uniformity among the various inheritance schemes in this respect. Beside the basic ISA type of arc, denoting class inclusion and common to all systems, other types that have been mentioned are ROLES (or FUNCTIONs), RELATIONS, and IDENTITIES. In the past few years much attention has been paid to the issue of cancellation of inheritance, that is, to systems which allow an object to override some property that it would otherwise inherit from another object higher in the hierarchy. These systems have been called nonmonotonic (since the set of properties does not increase monotonically as one descends the hierarchy); in contrast, systems without cancellation have been called monotonic. Most recent research in inheritance systems has been concerned with the semantics of inheritance. In particular there have been several results relating cancellations to nonmonotonic logics (Etherington, 1987), (Touretzky, 1986), (Touretzky et al., 1987).

Our concern in this paper is different, as we look at the complexity of reasoning with inheritance networks. Consider a network with vertices V and edges E, and let |V| = n and |E| = m. As is well known, if all the edges are ISA edges (such simple networks have been called taxonomic) then in time $O(m)$ one can determine whether the network implies an ISA relation between two particular nodes, and in time $O(nm)$ (and therefore in time $O(n^3)$) one can find all the implied ISA relations in the graph. If E contains other types of edge or if cancellation is allowed then the problem becomes harder. We know of relatively few results in this direction, including ones by Touretzky (1986) and Borgida (1989). Some relevant results involving negative and positive links are found in Thomason (1986). There are also results involving RELATIONS and IDENTITIES in Thomason (1989). We know of no results on the particular problem we consider, which is to allow E to contain ROLES as well as ISA edges, and to prohibit cancellation; we call these ISAR networks. We preclude cancellation not because we consider it unimportant, but because we would like to understand the monotonic case first. As will be seen, it is by no means straightforward. The
problem lies in a new closure rule that is provided by the interaction between ROLES and ISA edges. Consider for example the following graph:

```
parent spouse-of
mother is e.of
```

Intuitively, since mothers are parents, spouses of mothers are also spouses of parents. In other words, an ISA relation is implied about the two right nodes.

Of course, this intuitive claim needs to be formalized, and we will indeed do that. We will then consider the complexity of determining the implied ISA relations in such a network. In a very restricted type of ISAR networks we will be able to salvage the $O(m)$ and $O(nm)$ results from the simple taxonomic casc. For general ISAR networks we will offer a slightly costlier $O(n^3 + m^3)$ algorithm to find all the implied ISA relations. Actually, a finer analysis of our general algorithm reveals a complexity of $O(n \log r + n^2r + n^3)$, where $r$ is the number of ROLE labels in the network (we distinguish ROLE labels which are distinct, like brother-of and spouse-of, and actual ROLE edges, in which ROLE labels may be repeated).

Note that we have $r \leq m$, but we do not have $m \leq n^2$, since, unlike ISA edges, we may have multiple ROLE edges between two nodes (the spouses of mothers are exactly the joint-tax-payers of mothers). Among other things, this finer analysis takes us back to $O(n^3)$ for an $r$ bounded by a constant. As this is close to the best known algorithm for simple taxonomic networks it seems unlikely that this result can be significantly improved.

The remainder of the article is organized as follows.

In section 2, we briefly define the semantics of ISAs and ROLES, and based on these we provide provably complete conditions for determining all the implicit ISAs entailed by a given ISAR network. In section 3, we formally define the graph theoretic problem. In section 4, we briefly recall the results on taxonomic hierarchies, all well known. In section 5, we finally turn to the complexity of reasoning with ISAR networks. In section 5.1, we extend the results of section 4 to a restricted kind of ISAR networks which we call “equi-multiple inheritance”-ISAR (EMI-ISAR) networks. We then turn to the general case. First, in section 5.2, we provide a polynomial algorithm which reduces the problem to that of determining entailment by a propositional Horn theory. The degree of the polynomial turns out to be quite high, and so, in section 5.3, we give another, more direct algorithm, whose complexity was discussed above. Finally, in section 6 we summarize our results, compare them to previous results of which we are aware, and point to some open questions.

2. THE SYNTAX AND SEMANTICS OF ISAR NETWORKS

In order to be able to define our problem we first present the syntax and semantics for monotonic ISAR networks. Their syntax is defined as follows.

**Definition 1:** Let $V$ and $L$ be two disjoint sets. An ISAR network is a triple $<V,E,L>$ where $E \subseteq V \times V, E \subseteq V \times L$ and it satisfies:

1) If $(a,b,p) \in E_r$ and $(a,c,p) \in E_r$ then $b = c$;
2) If $p \in L$ then there are $a \in V$ and $b \in V$ such that $(a,b,p) \in E_r$.

$V$ is the set of vertices, $L$ is the set of ROLE labels, $E_r$ is the set of ISA edges and $E_r$ is the set of ROLE edges.

The second condition in the above definition is not essential, but it guarantees that any ROLE label indeed labels at least one ROLE edge, which is convenient. We now define their semantics.

**Definition 2:** Let $N = <V,E,L>h$ be an ISAR network and $L$ the set of ROLE labels of $N$. A model for $N$ is a pair $<D,\psi>$ where $D$ is a set and $\psi$ is a (total) function on VUL such that:

1) If $a \in V$ then $\psi(a) \in D$;
2) If $p \in L$ then $\psi(p)$ is a partial function from $D$ to $D$;
3) If $(a,b) \in E$ then $\psi(a) \subseteq \psi(b)$;
4) If $(a,b,p) \in E_r$ then $\psi(b) = \psi(p)(\psi(a))$.

Next we define two isa relations, one semantic and one syntactic.

**Definition 3:** Let $N = <V,E,L>h$ be an ISAR network. The binary relation $isa_a$ on $V$ is defined by:

- $isa_a(a,b)$ iff for every model $<D,\psi>$ for $N$, it is the case that $\psi(a) \subseteq \psi(b)$.

We will denote the fact that $isa_a(a,b)$ holds by $N \models isa(a,b)$.

**Definition 4:** Let $N = <V,E,L>$ be an ISAR network. The binary relation $isa_2$ on $V$ is the smallest set satisfying:

1) If $(a,b) \in E$ or $a = b$ then $(a,b) \in isa_2$;
2) (Rule1) If $(a,b) \in isa_2$ and $(b,c) \in isa_2$ then $(a,c) \in isa_2$;
3) (Rule2) If $(a,b) \in isa_2$, $(a,c,p) \in E_r$ and $(b,d,p) \in E_r$ then $(c,d) \in isa_2$.

We will denote the fact that $isa_2(a,b)$ holds by...
\[ N \models \text{isa}(a,b). \]

The next theorem establishes that Isa, and \( \text{isa}_r \) are actually the same relation.

**Theorem 1:** (Soundness and Completeness) Let \( N = \langle V, E, E_\text{r} \rangle \) be an ISAR network. For every \( a, b \in V \) and \( x, y \in V \), if \( N \models \text{isa}(a,b) \) then \( N \models \text{isa}(a,b) \).

**Proof.** Note that if \( (a,b) \in E \), then \( N \models \text{isa}(a,b) \); and also that Rule 1 and Rule 2 are sound with respect to our semantics. \( \leftarrow \) We omit this part of the proof; it will be included in the long version of this paper.

Note that if \( E_\text{r} \) is empty then the ISAR network reduces to a simple taxonomic inheritance network.

### 3. FORMAL PROBLEM DEFINITION

Given the syntax and semantics of ISAR networks, we now formally define the two problems we will be addressing.

**P.** Input: an ISAR network \( N = \langle V, E, E_\text{r} \rangle \) and a pair of vertices \( x, y \in V \).

Output: ‘yes’ if \( N \models \text{isa}(x,y) \), ‘no’ otherwise.

**P.** Input: an ISAR network \( N = \langle V, E, E_\text{r} \rangle \).

Output: an ISAR network \( N = \langle V, E', E'_\text{r} \rangle \) such that \( E'_\text{r} = \{(x,y) : N \models \text{isa}(x,y)\} \).

If \( |V| = n \) and \( \text{COMP} \), is the time complexity of \( P \), then clearly we have \( \text{COMP} \leq n^5 \text{COMP}_1 \), since we solve \( P_2 \) by solving \( P_1 \) for each pair of nodes.

In the rest of this paper, the number of vertices, \( |V| \), will be \( n \), the number of edges, \( |E| + |E_\text{r}| \), will be \( m \) and the number of ROLE labels, \( |R| \), will be \( r \).

Note that \( r \leq m \) and \( m \leq n^2 \).

### 4. SIMPLE TAXONOMIC HIERARCHIES: A REVIEW

In this section we briefly review the well-known results for the case in which the network contains only ISA edges.

**Theorem 2:** There exists an \( O(n) \) algorithm for \( P_1 \).

**Proof.** Use, e.g., the depth-first search (DFS) algorithm for directed graphs (Aho et al., 1974).

In fact, DFS may be used to find in \( O(m) \) time all the nodes reachable from a given node. We therefore have the following:

**Corollary 1:** There exists an \( O(nm) \) algorithm for \( P_2 \).

**Proof.** Run a DFS from each node.

There is also the well known direct algorithm for \( P_1 \): **Theorem 3:** There exists an \( O(n^2) \) algorithm for \( P_1 \).

**Proof.** Use the dynamic programming algorithm of, e.g., (Aho et al., 1974).

In fact, there exists a theoretically even better algorithm for \( P_1 \), whose complexity is about \( O(n^7) \). However, this theoretical result has not been translated to a practical advantage.

We mention these results for two reasons. First, as these are the best known results for taxonomic networks (and of course the linear result for \( P_1 \), is provably optimal) they form a lower bound for what we might expect for ISAR networks, and are good reference points against which to test our results. Second, the details of the algorithms mentioned above provide good insight into the qualitative increase in difficulty of ISAR networks. In the next section we discuss the DFS algorithm, and why it can be extended only to a limited class of ISAR networks. The dynamic programming algorithm, on the other hand, does not extend at all as far as we can see. Briefly, it relies on the property that if a path is decomposed at any vertex then each component is itself a path; that is true for simple taxonomic hierarchies, but not for general ISAR networks.

### 5. ALGORITHMS FOR ISAR NETWORKS

We now address the two problems defined in section 3, \( P_1 \) and \( P_2 \), in the context of general ISAR networks. We start with a very efficient algorithm for a restricted class of ISAR networks. We then give an easy algorithm for the general case whose complexity, though polynomial, is uncomfortably high. Finally, we give a low polynomial algorithm for the general case.

#### 5.1 EMI-ISAR networks

The DFS algorithm for taxonomic hierarchies extends paths into the graph, backtracks chronologically when a path is blocked, and never traverses the same edge twice. In this section we extend the algorithm to ISAR networks, introducing two major modifications. First, paths are extended in a way that is more complicated than simply following ISA edges. Second, in order to guarantee that we do not lose completeness by not traversing edges more than once (which guarantees linearity) we will need to impose a strong restriction on the network. Given the space limitations on this paper, we will only illustrate the algorithm through an example. Consider the simple network in Figure 1a.
Now consider the query ‘isa(\{289\},R+)’. This query should succeed due to the path shown in Figure 1b, which consists of three types of edge traversal: going back on ROLE edges (e.g., $Z^+$ to $Z$), going up ISA edges (e.g., $Z$ to $Q$), and going forward on ROLE edges (e.g., $R$ to $R^+$). We will call these respectively left, up and right traversal. Left and up traversals have no preconditions. Right traversal has a precondition that it not immediately follow a left traversal, and that the last left traversal to precede it was along a ROLE with the same label. To implement this we maintain a stack as we develop a path: up traversal does not affect the stack, back traversal pushes the ROLE label onto the stack, right traversal pops the stack (and has the precondition mentioned above). Figures 1c-1j illustrate the stack at all the vertices along the path in Figure 1b.

**Lemma 1:** Let $N$ be an ISAR network. Then $N \models isa(x,y)$ iff there is a path of the sort described above that starts at $x$ with an empty stack and ends at $y$ with an empty stack.

The only question that remains is how to determine efficiently whether such a path exists. Unfortunately, in ISAR networks with multiple inheritance we will in general need to traverse some edges many times. A simple example exists already in Figure 1a: if the first path developed is $\{289\}\{17\}Z^+QR$, then at that point backtracking must occur. If we are not allowed to traverse the edge QR twice, then we will not discover the path $\{289\}\{17\}Z^+QRR^+R^+$, and thus miss a solution. In special case, however, it is safe to not traverse an edge twice:

**Definition 5:** The label of a path is the sequence of ROLE labels appearing in it, ignoring all ISA edges.

**Theorem 4:** In the case of EMI-ISAR networks there exists an $O(m)$ algorithm for $P_r$.

**Proof.** Develop paths of the sort described above in a depth-first fashion, backtracking chronologically, never traversing an edge twice.

In fact, just as in the simple taxonomic case, this extended DFS can be used to discover all nodes to which a path exists from a given node. We thus get the following:

**Corollary 2:** In the case of EMI-ISAR networks there exists an $O(nm)$ algorithm for $P_r$.

Note that our results hold also when the network contains cycles.

### 5.2 Reducing general ISAR networks to propositional Horn theory

We now start to look at the general case of ISAR networks. In this section we pursue an easy way out, namely to reduce the graph theoretic problem to the problem of deciding a query about a propositional Horn theory, which is known to be decidable in linear time (Dowling and Gallier, 1984). Unfortunately, the resulting datalog theory will not be linear in the size of the ISAR network.

Let $N = < V, E, E_r >$ be an ISAR network. We construct a Horn theory $Th(N)$ as follows. First, for each three vertices $x, y, z$ in $N$, we construct a clause

$$isa(x,y) \rightarrow isa(x,z) \land isa(z,y)$$

Then, for each four vertices $v, x, y, z$ in $N$ and each ROLE label $l$ we construct a clause

$$isa(x,y) \rightarrow isa(v,z) \land isa(l,v) \land isa(l,x) \land isa(l,y)$$

Finally, for every pair $(a,b)$ in $E$, we add a predicate $isa(a,b)$, and for every triple $(a,b,p)$ in $E_r$ we add a predicate $role(p,a,b)$.

**Theorem 5:** There exists an $O(n^4)$ (and thus $O(mn^4)$) algorithm for $P_r$.

**Proof.** From Theorem 1 we have that $N \models isa(x,y)$ iff $Th(N) \models isa(x,y)$. The latter can be decided in linear time in $Th(N)$. The number of clauses in $Th(N)$ is $O(n^4 + nm^4) = O(n^4)$.

**Corollary 3:** There exists an $O(n^4)$ (and thus $O(mn^4)$) algorithm for $P_r$. 

---

**Definition 6:** An ISAR network is an **equi-multiple inheritance-ISAR network** (EMI-ISAR network) if for any two nodes $x$ and $y$, all paths from $x$ to $y$ have the same label.
5.3 An efficient algorithm for general ISAR networks

The degree of the polynomial in the previous algorithm is a bit too high for comfort. We now offer a more direct algorithm whose complexity is much lower.

Definition 7: A directed AND/OR graph is one in which the set of edges emanating from each node is partitioned into sets, each set called an AND-set of that node (single edges are viewed as singleton sets). A path in such a graph is a rooted tree such that the set of edges in the tree emanating from each vertex forms an AND-set of that vertex in the AND/OR graph. Searching an AND/OR graph from a given vertex means starting with a path consisting of the node itself, and iteratively extending it.

Definition 8: Let \( N = \langle V, E, E_r \rangle \) be an ISAR network. The evidence graph of \( N \) is the AND/OR directed graph \( \text{EVID}(N) = \langle V, E' \rangle \) where

\[
E' = \{(i,k),(i,j)\} : \text{for some } p, (i,k,p) \text{ and } (j,l,p) \text{ are both in } E_r
\]

\[
\cup \{( (i,k),(j,l) ) , ( (i,k),(j,l) ) : i,j,k \in V \}.
\]

The first type of edge is shown pictorially below:

The intuition behind the construction is the following: an AND-set of a vertex \((i,j)\) in the evidence graph is evidence that \((i,j)\) is in the isa relation. More precisely, we have the following:

Definition 9: Let \( N = \langle V, E, E_r \rangle \) be an ISAR network. A path rooted at \((a,b)\) in \(\text{EVID}(N)\) is said to be grounded if \(a = b\) or for all terminal nodes \((k,l)\) in that path it is the case that \((k,l)\) is in \(E_r\).

Lemma 2: Let \( N = \langle V, E, E_r \rangle \) be an ISAR network and \(i,j\) in \(V\). Then \(N\rightrightarrows\text{isa}(i,j)\) if and only if \(N\rightrightarrows\text{isa}(i,j)\). By induction on the number of applications of Rule 1 and Rule2 (Definition 4) we have that if \(N\rightrightarrows\text{isa}(i,j)\), then there is a grounded path rooted at \((i,j)\) in \(\text{EVID}(N)\). By induction on the size of the path we may prove that if there is a grounded path rooted at \((i,j)\) in \(\text{EVID}(N)\) then \(N\rightrightarrows\text{isa}(i,j)\).

Lemma 3: It can be determined in time \(O(m'^3)\) simultaneously for all vertices in \(\text{EVID}(N)\) whether there is a grounded path rooted at them, where \(m'\) is the number of edges in \(\text{EVID}(N)\).

Proof. (outline) Conduct a breadth-first search (BFS) starting from all nodes \((i,j)\) such that \((i,j)\) is in \(E_r\), moving backwards on edges, and extend a path beyond a vertex only when at least one of its AND-sets has all its members originate in previously-reached nodes.

The last lemma points to the reason for constructing the evidence graph. We now note that \(m'\) is bounded by the complexity of generating \(\text{EVID}(N)\). To complete the story, then, it remains to estimate this complexity. We first show an easy bound, and then look more closely at the algorithm to improve the complexity.

Theorem 6: There exists an \(O(n^3 + m'^3)\) algorithm for \(P_r\).

Proof. The construction of the edges in \(\text{EVID}(N)\) that are due to the transitive closure is done in time \(O(n'^3)\). To construct the other edges, we look at all pairs of ROLE edges \((i,j)\) and \((k,l)\), and, if their ROLE labels agree, add to \(\text{EVID}(N)\) the edges \(((i,k),(j,l))\) and \(((j,l),(i,k))\). The total number of edge-pairs is \(O(m'^3)\). Thus the total complexity of the algorithm is \(O(n^3 + m'^3)\).

Recall that in ISAR networks there is no necessary relation between the number of vertices and the number of edges. However, if it happens that \(m = O(n^3)\), we have that the algorithm is of complexity \(O(n^6)\). We now improve on this by a more careful construction of the evidence graph.

Theorem 7: There exists an \(O(nr(log r) + n^2 r + n^3)\) algorithm for \(P_r\), where \(r\) is the number of different ROLE labels.

Proof. We create the first \(n^3\) edges as before. Then, rather than blindly compare all pairs of edges, we do the following:

1) Create a list for each vertex of all the ROLE edges emanating from it and their associated label. A typical list will have the form \((l_1,i_1),(l_2,i_2),\ldots\) (where \(i, i_1, i_2, \ldots\) are vertices, and \(l, l_1, l_2, \ldots\) are ROLE labels);
2) Sort each of these lists by the label component;
3) For each pair of vertices \((i,j)\), scan their lists in parallel to see which role labels they share. If you encounter the pair \((p,k)\) in \(i\)'s list and the pair \((p,l)\) in \(j\)'s list, add the edges \(((k,l),(i,j))\) and \(((l,k),(j,i))\).
Complexity of the steps:

1) $O(m)$;
2) $O(n r (\log r))$ (note that each list is of length $r$ at most);
3) $O(n^* r)$ (scanning the sorted lists is linear in their length, $r$, and there are $n^*$ pairs of vertices).

We also note that we have $m \leq n^* r$, and so the total complexity of creating the evidence graph is $O(n r (\log r) + n^* r + n^*)$.

**Corollary 4:** If the number of ROLE labels is bounded by a constant, there is an $O(n^3)$ algorithm for $P_2$.

We note that as this is realistically the lowest complexity known for transitive closure, we would not hope to improve on this.

### 6. SUMMARY AND DISCUSSION

We have offered new results on the complexity of reasoning with inheritance hierarchies with ROLES, or ISAR networks. We defined two problems, $P_1$ (determining whether a ISAR network implies an isa relation on two nodes) and $P_2$ (finding the closure of the isa relation). Let $n$ be the number of vertices of an ISAR network, $m$ the number of edges, and $r$ the number of distinct ROLE labels. To somewhat crudely summarize our results, we have the following.

<table>
<thead>
<tr>
<th></th>
<th>no ROLES</th>
<th>EMI-ISAR</th>
<th>Horn alg</th>
<th>direct alg</th>
<th>fixed labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td>$n$</td>
<td>$n$</td>
<td>$mn^4$</td>
<td>$m^2 + n^4$</td>
<td>$n^3$</td>
</tr>
<tr>
<td>$P_2$</td>
<td>$nm$</td>
<td>$nm$</td>
<td>$mn^4$</td>
<td>$m^2 + n^4$</td>
<td>$n^3$</td>
</tr>
</tbody>
</table>

The only results bearing directly on ISAR networks with which we are familiar are due to Borgida (1989). His results include NP-Hardness for networks with cancellation, and polynomial results for two other problems. We do not yet understand well the relation between his results and ours. There appear to be few other complexity results. We are aware of Touretzky’s (1986) polynomial algorithm for parallel networks with RELATIONS, but do not see an interaction with our work.

Our results leave open some interesting questions. Our general result for $P_2$ is somewhat worse than the $O(mn)$ of transitive closure; can it be improved? Another striking feature of our result is that in the general case we have identical results for $P_1$ and $P_2$, although at first glance it seems that $P_1$ is much easier. Actually, our experience with the problem leads us to conjecture that $P_1$ is not any easier, but it would be nice to have a result on that. Then there is a question about other ways to salvage the $O(n), O(nm)$ results from the simple taxonomic case: do there exist interesting classes of networks which permit that other than EMI-ISAR networks? Finally, what happens when we add other features to the network, such as RELATIONS or cancellation? We conjecture that at least in the latter case the problem in general becomes intractable, which seems to agree with Borgida’s result mentioned above.

**Acknowledgments.** The authors are grateful to Richmond H. Thomason, who described the problem, and for helpful suggestions on a first version of this paper.

**REFERENCES**


