Artificial Intelligence **39**, 2 (1989), pages 263–272.

Undecidability of Subsumption in NIKL

Peter F. Patel-Schneider[1]
AI Principles Research Department
AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill, New Jersey 07974

July 1988

[1]Much of the research reported in this note was performed at Schlumberger Palo Alto Research.

Abstract

Subsumption—determining whether one concept is more general than another—is known to be NP-hard for all reasonably expressive terminological logics, but, up to now, the decidability of subsumption for terminological logics used in current knowledge representation systems such as NIKL remained unknown. This paper shows that subsumption in the terminological logic of NIKL is undecidable and thus that there are no complete algorithms for subsumption or classification in NIKL.

1 Introduction

Terminological logics (also called frame-based description languages) formalize and extend the notions of frames. They are used to represent knowledge about the terminology used to describe the world. Terminological logics are found in many modern knowledge representation systems, such as KL-ONE [Brachman and Schmolze, 1985], KRYPTON [Brachman *et al.*, 1985], KL-TWO [Vilain, 1985], NIKL [Robins, 1986; Kaczmarek *et al.*, 1986], KANDOR [Patel-Schneider, 1984], BACK [Nebel and von Luck, 1988], and LOOM [MacGregor and Bates, 1987; MacGregor, 1988].

Although the syntax and expressive power of the terminological logics used in different knowledge representation systems vary considerably, these terminological logics all have the same basic ideas. They all allow the construction of structured concepts and roles; they all have a similar set of operators for constructing these concepts and roles; and they all have a formal semantics[1] which provides a precise meaning for concepts and roles.

Perhaps the most important point of similarity between all these terminological logics is that they define semantic relationships between concepts and roles which play an essential part in the operation of the system. These semantic relationships include whether one concept or role is necessarily more general than another, *i.e.* whether the first *subsumes* the other; whether two concepts or roles are necessarily disjoint; and whether one concept or role can have any instances. The operations of the system are then defined in terms of these semantic relationships. For example, all of the systems have a query operation which asks whether one concept or role subsumes another. The formal definition of subsumption in the logic provides a correctness criterion for the operation.

It turns out that most of the operations performed by such systems can be reduced to determining subsumption relationships. For example, determining whether a concept can have any instances is equivalent to determining whether it is subsumed by a concept which is known to necessarily not have any instances. Similarly classification—determining where a concept fits in a taxonomy of concepts—reduces to asking which of the concepts in the taxonomy it subsumes and which are subsumed by it.

[1] In the earlier systems, this formal semantics was devised after the system was built, but nowadays the semantics precedes the system.

Unfortunately, determining whether one concept subsumes another is an inherently computationally expensive operation. This came as quite a surprise, since the first subsumption algorithm (for KL-ONE), developed by Lipkis [1982], was a simple structural algorithm that ran in polynomial time, and was initially thought to be complete. However, when Schmolze and Israel [1983] developed a formal semantics for KL-ONE, the algorithm was found to be sound but not complete.

More recently, Levesque and Brachman [1987] have shown that subsumption in a very simple terminological logic is co-NP-complete. This logic is a subset of the terminological logics of KL-ONE, NIKL, and KL-TWO, and thus subsumption is NP-hard in these systems. Also, Nebel [1988] has shown that subsumption is NP-hard in a subset of the terminological logics of KANDOR, BACK, KL-ONE, NIKL, and KL-TWO. This intractability remains even when numbers are represented in unary notation, thus making subsumption in all these languages strongly NP-hard.

Although subsumption was known to be intractable in these languages, it was not known whether it was decidable or not. A very recent result by Schild [1988] shows that subsumption is undecidable in very expressive terminological logics, ones that include conjunction and negation of roles.[2] However, since the terminological logics used in most existing knowledge representation systems do not include these constructs, this does not imply that subsumption is undecidable in these less expressive terminological logics. The purpose of this paper is to show that subsumption is indeed undecidable for the terminological logic of NIKL (a much less expressive logic than the one shown undecidable by Schlid), and hence in knowledge representation systems that use terminological logics of similar power.

2 Formal Definitions

Before this undecidability result can be shown, a formal definition of the terminological logic in question is needed.

This terminological logic has the following syntax:

<concept> ::= <atomic concept> |

[2]Shild's proof is by reduction to Turing machines and is much more complicated than the proof here.

```
                 (and <concept>+) |
                 (some <role>) |
                 (atmost 1 <role>) |
                 (all <role chain> <role chain>)
<role> ::= <atomic role> |
           (restrict <role> <concept>) |
           (inverse <role>)
<role chain> ::= (self) |
                 <role> |
                 (compose <role> <role>+)
```

This syntax is slightly different from the syntax of NIKL, as given in [Schmolze, 1989]. However, the expressive power of this logic is a subset of the expressive power of NIKL, as all the constructs in this logic can be created in NIKL, even using arbitrary roles in role chains, which can be obtained in NIKL by naming the role and using the name in the role chain.

The semantics of the logic is defined as follows:

Definition 1 *A semantic structure, s, is a pair, $\langle D, \mathcal{E} \rangle$, where D is a non-empty set and $\mathcal{E}$ is a mapping from concepts and roles to their extension. The extension of a concept is a subset of D—the set of domain elements that belong to the concept. Similarly, the extension of a role is a subset of $D \times D$.*

The extension of non-atomic concepts and roles has to meet certain properties, namely

$d \in \mathcal{E}[(\mathsf{and}\ \mathsf{C}_1 \ \ldots \mathsf{C}_n)]$	*iff*	*for each* i, $d \in \mathcal{E}[\mathsf{C}_i]$
$d \in \mathcal{E}[(\mathsf{some}\ \mathsf{R})]$	*iff*	$\exists e \ \langle d, e \rangle \in \mathcal{E}[\mathsf{R}]$
$d \in \mathcal{E}[(\mathsf{atmost}\ 1\ \mathsf{R})]$	*iff*	$\lvert \{e : \langle d, e \rangle \in \mathcal{E}[\mathsf{R}]\} \rvert \leq 1$
$d \in \mathcal{E}[(\mathsf{all}\ \mathsf{R}_1\ \mathsf{R}_2)]$	*iff*	$\forall e \ \ \langle d, e \rangle \notin \mathcal{E}[\mathsf{R}_1]$ *or* $\langle d, e \rangle \in \mathcal{E}[\mathsf{R}_2]$
$\langle d, e \rangle \in \mathcal{E}[(\mathsf{restrict}\ \mathsf{R}\ \mathsf{C})]$	*iff*	$\langle d, e \rangle \in \mathcal{E}[\mathsf{R}]$ *and* $e \in \mathcal{E}[\mathsf{C}]$
$\langle d, e \rangle \in \mathcal{E}[(\mathsf{inverse}\ \mathsf{R})]$	*iff*	$\langle e, d \rangle \in \mathcal{E}[\mathsf{R}]$
$\langle d, e \rangle \in \mathcal{E}[(\mathsf{self})]$	*iff*	$d = e$
$\langle d, e \rangle \in \mathcal{E}[(\mathsf{compose}\ \mathsf{R}_1 \ \ldots \mathsf{R}_n)]$	*iff*	$\exists z_1, \ldots, z_{n+1} \ \ z_1 = d, z_{n+1} = e$, *and for each* i, $\langle z_i, z_{i+1} \rangle \in \mathcal{E}[\mathsf{R}_i]$

This semantics is compatible with the definitions in [Schmolze, 1989].

Now subsumption can be defined:

Definition 2 *For any two concepts,* C *and* C', C *is subsumed by* C' *(*$\mathsf{C} \Rightarrow \mathsf{C}'$*) iff for any semantic structure,* $s = \langle D, \mathcal{E} \rangle$, $\mathcal{E}[\mathsf{C}] \subseteq \mathcal{E}[\mathsf{C}']$.

that is, one concept is subsumed by a second when all individuals that are instances of the first must also be instances of the second.

3 Undecidability

Subsumption is shown to be undecidable in this logic via a reduction to the Post correspondence problem for the alphabet $\{0, 1\}$. The following definition of the Post correspondence problem is taken from [Lewis, 1979, p. 55].[3]

Definition 3 *A* correspondence system *is a finite subset $\mathcal{P}$ of $\Sigma^+ \times \Sigma^+$ for some finite alphabet Σ; i.e., a finite set of pairs of nonempty strings. A* presolution *of $\mathcal{P}$ is a pair of strings $\langle \alpha_1 \dots \alpha_k, \beta_1 \dots \beta_k \rangle$ such that $\langle \alpha_i, \beta_i \rangle \in \mathcal{P}$ for $i = 1, \dots, k$. This presolution is a* solution *of $\mathcal{P}$ provided that $k > 0$ and $\alpha_1 \dots \alpha_k = \beta_1 \dots \beta_k$. The* Post correspondence problem *is to determine, given a correspondence system $\mathcal{P}$, whether or not $\mathcal{P}$ has a solution, for a fixed alphabet Σ.*

Theorem 1 *The Post correspondence problem is undecidable for the alphabet $\Sigma = \{0, 1\}$.*
Proof: See [Lewis, 1979], pp. 55–58.

The outline of the proof of undecidability of subsumption in this logic is as follows: Given a correspondence system $\mathcal{P}$, construct a concept $\mathsf{C}_{\mathcal{P}}$ such that there is a solution to $\mathcal{P}$ if and only if C is subsumed by $\mathsf{C}_{\mathcal{P}}$, where C is an atomic concept not appearing in $\mathsf{C}_{\mathcal{P}}$. If this can be done then any algorithm for subsumption in this logic would give rise to an algorithm for the Post correspondence problem; since no such algorithm exists for the Post correspondence problem, subsumption for this logic would then be undecidable.

Several atomic roles and concepts are used in the construction. The atomic role T plays the role of a "universal role". The extension of the atomic concept A is the domain element that represents the empty string. The atomic role F_a, for $a \in \{0, 1\}$, has the property that $\langle d, e \rangle \in \mathcal{E}[\mathsf{F}_a]$ iff e represents the concatenation of the string represented by d with a. The

[3] The following notation is used: If Σ is a set of characters, then Σ^+ is the set of finite, non-empty strings over Σ. If $\sigma \in \Sigma^+$, then $|\sigma|$ is the number of characters in σ, and σ^i is the ith character of σ.

atomic role P has the property that $\langle d, e\rangle \in \mathcal{E}[\mathsf{P}]$ iff there is a presolution of $\mathcal{P}$ with the strings represented by d and e.

The construction is much easier to understand if the following semantically meaningful abbreviations are used.

Definition 4 (Syntactic Abbreviations)

1. U $\stackrel{\text{def}}{=}$ (and (all T (inverse T))
(all (compose T T) T)
(some T)
(all T (restrict T (all F_0 T)))
(all T (restrict T (all F_1 T))))

2. (unique C) $\stackrel{\text{def}}{=}$ (and (atmost 1 (restrict T C))
(some (restrict T C)))

3. (function R) $\stackrel{\text{def}}{=}$ (all T (restrict T (and (atmost 1 R) (some R))))

4. (exists-self R) $\stackrel{\text{def}}{=}$ (some (restrict T (all (self) R)))

5. (impl C_1 C_2) $\stackrel{\text{def}}{=}$ (all (restrict T C_1) (restrict T C_2))

6. (impl R_1 R_2) $\stackrel{\text{def}}{=}$ (all T (restrict T (all R_1 R_2)))

7. $\mathsf{I}_{\alpha,\beta}$ $\stackrel{\text{def}}{=}$
(some (restrict T
(and A (some (restrict F_{α^1} ...
(some (restrict $\mathsf{F}_{\alpha^{|\alpha|}}$
(some (restrict P
(some (restrict (inverse $\mathsf{F}_{\beta^{|\beta|}}$) ...
(some (restrict (inverse F_{β^1}) A)) ...)))))) ...)))))

8. $\mathsf{J}_{\alpha,\beta}$ $\stackrel{\text{def}}{=}$
(impl P (compose F_{α^1} ... $\mathsf{F}_{\alpha^{|\alpha|}}$ P (inverse $\mathsf{F}_{\beta^{|\beta|}}$) ... (inverse F_{β^1})))

where α, β are elements of $\{0,1\}^+$.

The first abbreviation states certain properties about T, which must hold for a semantic structure to be of interest. These properties are that T must be symmetric, transitive, and non-empty, as well as being a superset of F_0 and F_1.

The remaining abbreviations can be be understood in a semantic structure where T is a universal relation. (It is not the case that T will always be a true universal relation, but in all cases of interest a subset of the domain where T is a universal relation will be used. The "domain" in the following statements refers to this subset of the entire domain.) If this is the case then

1. the extension of (unique C) is the entire domain iff the extension of C is a singleton,

2. the extension of (function R) is the entire domain iff the extension of R forms a function,

3. the extension of (exists-self R) is the entire domain iff the extension of R includes $\langle e, e\rangle$ for some e in the domain,

4. the extension of (impl C_1 C_2) is the entire domain iff the extension of C_2 includes the extension of C_1,

5. the extension of (impl R_1 R_2) is the entire domain iff the extension of R_2 includes the extension of R_1,

6. the extension of $\mathsf{I}_{\alpha,\beta}$ is the entire domain iff
$$\begin{aligned}\exists e, f, g, h \;\; & e \in \mathcal{E}[\mathsf{A}] \wedge f \in \mathcal{E}[\mathsf{A}] \wedge \\ & \langle e, g\rangle \in \mathcal{E}[(\mathsf{compose}\ \mathsf{F}_{\alpha^1} \ldots \mathsf{F}_{\alpha^{|\alpha|}})] \wedge \\ & \langle f, h\rangle \in \mathcal{E}[(\mathsf{compose}\ \mathsf{F}_{\beta^1} \ldots \mathsf{F}_{\beta^{|\beta|}})] \wedge \\ & \langle g, h\rangle \in \mathcal{E}[\mathsf{P}],\end{aligned}$$
and

7. the extension of $\mathsf{J}_{\alpha,\beta}$ is the entire domain iff
$$\begin{aligned}\forall e, f\, \exists g, h\, \langle e, f\rangle \in \mathcal{E}[\mathsf{P}] \rightarrow \;\; & \langle e, g\rangle \in \mathcal{E}[(\mathsf{compose}\ \mathsf{F}_{\alpha^1} \ldots \mathsf{F}_{\alpha^{|\alpha|}})] \wedge \\ & \langle f, h\rangle \in \mathcal{E}[(\mathsf{compose}\ \mathsf{F}_{\beta^1} \ldots \mathsf{F}_{\beta^{|\beta|}})] \wedge \\ & \langle g, h\rangle \in \mathcal{E}[\mathsf{P}].\end{aligned}$$

Now the full construction is defined:

Definition 5 *Let $\mathcal{P}$ be a correspondence system over the alphabet $\Sigma = \{0,1\}$, $\mathcal{P} = \{\langle\alpha_1, \beta_1\rangle, \ldots, \langle\alpha_n, \beta_n\rangle\}$. Then the concept $\mathsf{C}_{\mathcal{P}}$ is defined as*

$$\begin{array}{l}(\mathsf{impl\ (and\ U\ (unique\ A)\ (function\ F_0)\ (function\ F_1)} \\ \qquad\qquad \mathsf{I}_{\alpha_1,\beta_1}\ \ldots \mathsf{I}_{\alpha_n,\beta_n}\ \mathsf{J}_{\alpha_1,\beta_1}\ \ldots \mathsf{J}_{\alpha_n,\beta_n}) \\ \qquad \mathsf{(exists\text{-}self\ P))}\end{array}$$

The idea behind the construction is that the extension of $\mathsf{C}_{\mathcal{P}}$ is the entire domain iff whenever certain conditions are satisfied then the extension of P includes $\langle e, e\rangle$ for some e in the domain. These conditions are that T is a "universal" role, that the extension of A is a singleton, that the extension of F_0 and F_1 form functions, and that if the strings represented by d and e form a presolution of $\mathcal{P}$ then $\langle d, e\rangle$ is in the extension of P. This equivalence is used to show that $\mathsf{C} \Rightarrow \mathsf{C}_{\mathcal{P}}$, where C is an atomic concept not occurring in $\mathsf{C}_{\mathcal{P}}$, iff $\mathcal{P}$ has a solution, and thus that subsumption in the logic in undecidable.

Theorem 2 *Subsumption in the terminological logic given here is undecidable.*

Proof: Let $\mathcal{P}$ be a correspondence system, let $\mathsf{C}_{\mathcal{P}}$ be as above, and let C be an atomic concept not appearing in $\mathsf{C}_{\mathcal{P}}$.

Suppose $\mathsf{C} \Rightarrow \mathsf{C}_{\mathcal{P}}$.
Let $s = \langle D, \mathcal{E}\rangle$ be a semantic structure with $D = \{0,1\}^*$ for which

$\mathcal{E}[\mathsf{C}] = D$,
$\mathcal{E}[\mathsf{T}] = D \times D$,
$d \in \mathcal{E}[\mathsf{A}]$ iff d is the empty string (written λ),
$\langle d, e\rangle \in \mathcal{E}[\mathsf{F}_0]$ iff $e = d \cdot 0$,
$\langle d, e\rangle \in \mathcal{E}[\mathsf{F}_1]$ iff $e = d \cdot 1$, and
$\langle d, e\rangle \in \mathcal{E}[\mathsf{P}]$ iff $\langle d, e\rangle$ is a presolution of $\mathcal{P}$.

Now a simple inspection reveals that

$\mathcal{E}[\mathsf{U}] = D$,
$\mathcal{E}[(\mathsf{unique\ A})] = D$,
$\mathcal{E}[(\mathsf{function\ F_0})] = D$,
$\mathcal{E}[(\mathsf{function\ F_1})] = D$,
$\mathcal{E}[\mathsf{I}_{\alpha_i,\beta_i}] = D$, for $1 \le i \le n$, and
$\mathcal{E}[\mathsf{J}_{\alpha_i,\beta_i}] = D$, for $1 \le i \le n$.

Therefore $\mathcal{E}[(\textsf{exists-self P})] = D$, since $\mathcal{E}[\textsf{C}_\mathcal{P}] = D$. Thus there exists $e \in D$ such that $e \in \mathcal{E}[(\textsf{all (self) P})]$ and thus $\langle e, e\rangle \in \mathcal{E}[\textsf{P}]$. Therefore $\mathcal{P}$ has a solution.

Suppose $\mathcal{P}$ has a solution.

Let $s = \langle D, \mathcal{E}\rangle$ be an arbitrary semantic structure. Consider $d \in D$ for which

$$\begin{array}{ll} d \in \mathcal{E}[\textsf{U}], & \\ d \in \mathcal{E}[(\textsf{unique A})], & \\ d \in \mathcal{E}[(\textsf{function F}_0)], & \\ d \in \mathcal{E}[(\textsf{function F}_1)], & \\ d \in \mathcal{E}[\textsf{I}_{\alpha_i,\beta_i}], \text{ for } 1 \leq i \leq n, & \text{and} \\ d \in \mathcal{E}[\textsf{J}_{\alpha_i,\beta_i}], \text{ for } 1 \leq i \leq n. & \end{array}$$

Let $D' = \{e \in D : \langle d, e\rangle \in \mathcal{E}[\textsf{T}]\}$. Then there is a unique $e \in D'$ such that $e \in \mathcal{E}[\textsf{A}]$. Also, for all $e \in D'$ there is a unique $e' \in D'$ such that $\langle e, e'\rangle \in \mathcal{E}[\textsf{F}_0]$ and a unique $e'' \in D'$ such that $\langle e, e''\rangle \in \mathcal{E}[\textsf{F}_1]$.

For $\alpha \in \{0,1\}^*$, let $\overline{\alpha}$ be defined inductively as:

- $\overline{\lambda}$ is the unique $e' \in D'$ such that $e' \in \mathcal{E}[\textsf{A}]$,
- $\overline{\alpha \cdot 0}$ is the unique $e' \in D'$ such that $\langle \overline{\alpha}, e'\rangle \in \mathcal{E}[\textsf{F}_0]$, and
- $\overline{\alpha \cdot 1}$ is the unique $e' \in D'$ such that $\langle \overline{\alpha}, e'\rangle \in \mathcal{E}[\textsf{F}_1]$.

Similarly, for $\alpha \in \{0,1\}^+$ and $e \in D'$, let $e\overline{\alpha}$ be defined inductively as:

- $e\overline{0}$ is the unique $e' \in D'$ such that $\langle e, e'\rangle \in \mathcal{E}[\textsf{F}_0]$,
- $e\overline{1}$ is the unique $e' \in D'$ such that $\langle e, e'\rangle \in \mathcal{E}[\textsf{F}_1]$,
- $e\overline{\alpha \cdot 0}$ is the unique $e' \in D'$ such that $\langle e\overline{\alpha}, e'\rangle \in \mathcal{E}[\textsf{F}_0]$, and
- $e\overline{\alpha \cdot 1}$ is the unique $e' \in D'$ such that $\langle e\overline{\alpha}, e'\rangle \in \mathcal{E}[\textsf{F}_1]$.

Then $\langle \overline{\alpha_i}, \overline{\beta_i}\rangle \in \mathcal{E}[\textsf{P}]$, for $1 \leq i \leq n$, because $d \in \mathcal{E}[\textsf{I}_{\alpha_i,\beta_i}]$. Also, $\langle e, e'\rangle \in \mathcal{E}[\textsf{P}]$ implies $\langle e\overline{\alpha_i}, e'\overline{\beta_i}\rangle \in \mathcal{E}[\textsf{P}]$, for $1 \leq i \leq n$, because $d \in \mathcal{E}[\textsf{J}_{\alpha_i,\beta_i}]$. Therefore, if $\langle \alpha, \beta\rangle$ is a presolution of $\mathcal{P}$ then $\langle \overline{\alpha}, \overline{\beta}\rangle \in \mathcal{E}[\textsf{P}]$. Since $\mathcal{P}$ has a solution, there exists $\alpha \in \{0,1\}^+$ such that $\langle \alpha, \alpha\rangle$ is a presolution of $\mathcal{P}$. Thus $\langle \overline{\alpha}, \overline{\alpha}\rangle \in \mathcal{E}[\textsf{P}]$, and, since $\overline{\alpha} \in D'$ for all $\alpha \in \{0,1\}^*$, $d \in \mathcal{E}[(\textsf{exists-self P})]$.

Therefore, for all $d \in D$, if $d \in \mathcal{E}[\mathsf{U}]$, $d \in \mathcal{E}[(\mathsf{unique\ A})]$, $d \in \mathcal{E}[(\mathsf{function\ F_0})]$, $d \in \mathcal{E}[(\mathsf{function\ F_1})]$, $d \in \mathcal{E}[\mathsf{I}_{\alpha_i,\beta_i}]$, for $1 \leq i \leq n$, and $d \in \mathcal{E}[\mathsf{J}_{\alpha_i,\beta_i}]$, for $1 \leq i \leq n$, then $d \in \mathcal{E}[(\mathsf{exists\text{-}self\ P})]$.

Thus for all $d \in D$, for any $d' \in D$, if

$$d' \in \mathcal{E}[(\mathsf{and\ U\ (unique\ A)\ (function\ F_0)\ (function\ F_1)} \\ \mathsf{I}_{\alpha_1,\beta_1} \ldots \mathsf{I}_{\alpha_n,\beta_n}\ \mathsf{J}_{\alpha_1,\beta_1} \ldots \mathsf{J}_{\alpha_n,\beta_n})\ ]$$

and $\langle d, d' \rangle \in \mathcal{E}[\mathsf{T}]$, then $d' \in \mathcal{E}[(\mathsf{exists\text{-}self\ P})]$ (and $\langle d, d' \rangle \in \mathcal{E}[\mathsf{T}]$). Therefore, recalling the definition of $\mathsf{C}_\mathcal{P}$, $d \in \mathcal{E}[\mathsf{C}_\mathcal{P}]$, i.e., $\mathcal{E}[\mathsf{C}_\mathcal{P}] = D$, and thus $\mathsf{C} \Rightarrow \mathsf{C}_\mathcal{P}$.

Therefore $\mathsf{C} \Rightarrow \mathsf{C}_\mathcal{P}$ iff $\mathcal{P}$ has a solution, and thus, since the Post correspondence problem is undecidable, so too is subsumption in this terminological logic. ∎

Since this logic is a subset of the terminological logics of NIKL and LOOM, these terminological logics are also undecidable. A simple translation to first-order logic can be used to show that subsumption is semi-decidable in this logic and also in the terminological logics of NIKL and LOOM.

There are several other subsets of the terminological logic of NIKL that are also undecidable. For example, the inverse role construct can be eliminated by adding

(all T (restrict T (atmost 1 G_0)))
(all T (restrict T (atmost 1 G_1)))
(all T (restrict T (all (self) (compose F_0 G_0))))
(all T (restrict T (all (self) (compose F_1 G_1))))

to the conjuncts in $\mathsf{C}_\mathcal{P}$, and replacing (inverse F_0) with G_0 and (inverse F_1) with G_1.[4] A proof that this logic is undecidable is given in Appendix A.

4 Conclusion

This result does not lessen the utility of terminological logics, especially since knowledge representation systems using terminological logics have retreated

[4]Manfred Schmidt-Schauss, in work performed while this note was in press, has a proof that an even smaller subset is undecidable.

to incomplete subsumption and classification algorithms as a result of the intractability of subsumption in these logics. In particular, the classification algorithm of NIKL does nothing with the information that one role is the inverse of another and also does not discover subsumptions such as (all R C_2) subsuming (and (all R C_1) (all (restrict R C_1) C_2)).

The meaning of this result is that no complete algorithm exists for subsumption in the terminological logic of NIKL, and for terminological logics incorporating the terminological logic shown to be undecidable here. Therefore attention must be transferred from finding complete subsumption and classification algorithms to providing better, and better described, partial subsumption and classification algorithms.

Acknowledgments

Some of the constructions in this paper are based on constructions in the paper by Klaus Schild.

References

[Brachman and Schmolze, 1985] Ronald J. Brachman and James G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2):171–216, April–June 1985.

[Brachman *et al.*, 1985] Ronald J. Brachman, Victoria Pigman Gilbert, and Hector J. Levesque. An essential hybrid reasoning system: Knowledge and symbol level accounts of KRYPTON. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 532–539, Los Angeles, California, August 1985. International Joint Committee on Artificial Intelligence.

[Kaczmarek *et al.*, 1986] Thomas S. Kaczmarek, Raymond Bates, and Gabriel Robbins. Recent developments in NIKL. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 978–985, Philadelphia, Pennsylvania, August 1986. American Association for Artificial Intelligence.

[Levesque and Brachman, 1987] Hector J. Levesque and Ronald J. Brachman. Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence*, 3(2):78–93, May 1987.

[Lewis, 1979] Harry R. Lewis. *Unsolvable Classes of Quantificational Formulas.* Addison-Wesley, Reading, Massachusetts, 1979.

[Lipkis, 1982] Thomas Lipkis. A KL-ONE classifier. In James G. Schmolze and Ronald J. Brachman, editors, *Proceedings of the 1981 KL-One Workshop*, pages 126–143, Jackson, New Hampshire, June 1982. Bolt Beranek and Newman Inc.

[MacGregor and Bates, 1987] Robert M. MacGregor and Raymond Bates. The Loom knowledge representation language. Technical Report ISI/RS-87-188, Information Sciences Institute, University of Southern California, May 1987.

[MacGregor, 1988] Robert M. MacGregor. A deductive pattern matcher. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 403–408, St. Paul, Minnesota, August 1988. American Association for Artificial Intelligence.

[Nebel and von Luck, 1988] Bernhard Nebel and Kai von Luck. Hybrid reasoning in BACK. In Zbignew W. Ras and Lorenza Saitta, editors, *Methodologies for Intelligent Systems*, volume 3, pages 260–269. North-Holland, New York, 1988.

[Nebel, 1988] Bernhard Nebel. Computational complexity of terminological reasoning in BACK. *Artificial Intelligence*, 34(3):371–383, April 1988.

[Patel-Schneider, 1984] Peter F. Patel-Schneider. Small can be beautiful in knowledge representation. In *Proceedings of the IEEE Workshop on Principles of Knowledge-Based Systems*, pages 11–16, Denver, Colorado, December 1984. IEEE Computer Society.

[Robins, 1986] Gabriel Robins. The NIKL manual. The Knowledge Representation Project, Information Sciences Institute, University of Southern California, April 1986.

[Schild, 1988] Klaus Schild. Undecidability of subsumption in $\mathcal{U}$. KIT-Report 67, Fachbereich Informatik, Technische Universität Berlin, 1988.

[Schmolze and Israel, 1983] James G. Schmolze and David J. Israel. KL-ONE: Semantics and classification. Technical Report 5421, BBN Laboratories, 1983. Part of a collection entitled "Research in Knowledge Representation for Natural Language Understanding—Annual Report, 1 September 1982–31 August 1983".

[Schmolze, 1989] James G. Schmolze. The language and semantics of NIKL. Technical Report 89–4, Department of Computer Science, Tufts University, September 1989.

[Vilain, 1985] Marc Vilain. The restricted language architecture of a hybrid representation system. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 547–551, Los Angeles, California, August 1985. International Joint Committee on Artificial Intelligence.

A Undecidability of Logic without Inverse

This appendix contains a proof that subsumption is undecidable in the terminological logic given here even if role inverses are not included. The basic idea behind the proof is to replace role inverses by a new role which is constrained to be the inverse of the original role.

Definition 6 *Let $\mathcal{P}$ be a correspondence system over the alphabet $\Sigma = \{0, 1\}$, $\mathcal{P} = \{\langle\alpha_1, \beta_1\rangle, \ldots, \langle\alpha_n, \beta_n\rangle\}$. Then the concept $\mathsf{C}'_\mathcal{P}$ is defined as*

$$\begin{array}{l}\mathsf{(impl\ (and\ U\ (unique\ A)\ (function\ F_0)\ (function\ F_1)}\\ \qquad\mathsf{(all\ T\ (restrict\ T\ (atmost\ 1\ G_0)))}\\ \qquad\mathsf{(all\ T\ (restrict\ T\ (atmost\ 1\ G_1)))}\\ \qquad\mathsf{(all\ T\ (restrict\ T\ (all\ (self)\ \ (compose\ F_0\ G_0))))}\\ \qquad\mathsf{(all\ T\ (restrict\ T\ (all\ (self)\ \ (compose\ F_1\ G_1))))}\\ \qquad\mathsf{I}'_{\alpha_1,\beta_1}\ \ldots\mathsf{I}'_{\alpha_n,\beta_n}\ \mathsf{J}'_{\alpha_1,\beta_1}\ \ldots\mathsf{J}'_{\alpha_n,\beta_n})\\ \quad\mathsf{(exists\text{-}self\ P)),}\end{array}$$

where

1. $\mathsf{I}'_{\alpha,\beta} \stackrel{\text{def}}{=}$

(some (restrict T
    (and A (some (restrict F_{α^1} ...
      (some (restrict $\mathsf{F}_{\alpha^{|\alpha|}}$
        (some (restrict P
          (some (restrict $\mathsf{G}_{\beta^{|\beta|}}$ ...
            (some (restrict G_{β^1} A)) ...))))))...))))),

and

2. $\mathsf{J}'_{\alpha,\beta} \stackrel{\text{def}}{=}$ (impl P (compose $\mathsf{F}_{\alpha^1} \ldots \mathsf{F}_{\alpha^{|\alpha|}}$ P $\mathsf{G}_{\beta^{|\beta|}} \ldots \mathsf{G}_{\beta^1}$)).

Theorem 3 *Subsumption in the terminological logic given here is undecidable, even if role inverses are not included.*

Proof: Let $\mathcal{P}$ be a correspondence system, let $\mathsf{C}'_{\mathcal{P}}$ be as above, and let C be an atomic concept not appearing in $\mathsf{C}'_{\mathcal{P}}$.

Suppose $\mathsf{C} \Rightarrow \mathsf{C}'_{\mathcal{P}}$.
Then let $s = \langle D, \mathcal{E} \rangle$ be a semantic structure for which

$\mathcal{E}[\mathsf{C}] = D$,
$\mathcal{E}[\mathsf{T}] = D \times D$,
$d \in \mathcal{E}[\mathsf{A}]$ iff d is the empty string (written λ),
$\langle d, e \rangle \in \mathcal{E}[\mathsf{F}_0]$ iff $e = d \cdot 0$,
$\langle d, e \rangle \in \mathcal{E}[\mathsf{F}_1]$ iff $e = d \cdot 1$,
$\langle d, e \rangle \in \mathcal{E}[\mathsf{G}_0]$ iff $\langle e, d \rangle \in \mathcal{E}[\mathsf{F}_0]$,
$\langle d, e \rangle \in \mathcal{E}[\mathsf{G}_1]$ iff $\langle e, d \rangle \in \mathcal{E}[\mathsf{F}_1]$, and
$\langle d, e \rangle \in \mathcal{E}[\mathsf{P}]$ iff $\langle d, e \rangle$ is a presolution of $\mathcal{P}$.

Now a simple inspection reveals that

$\mathcal{E}[\mathsf{U}] = D$,
$\mathcal{E}[$(unique A)$] = D$,
$\mathcal{E}[$(function F_0)$] = D$,
$\mathcal{E}[$(function F_1)$] = D$,
$\mathcal{E}[$(all T (restrict T (atmost 1 G_0)))$] = D$,
$\mathcal{E}[$(all T (restrict T (atmost 1 G_1)))$] = D$,
$\mathcal{E}[$(all T (restrict T (all (self) (compose F_0 G_0))))$] = D$,
$\mathcal{E}[$(all T (restrict T (all (self) (compose F_1 G_1))))$] = D$,
$\mathcal{E}[\mathsf{I}'_{\alpha_i,\beta_i}] = D$, for $1 \leq i \leq n$, and
$\mathcal{E}[\mathsf{J}'_{\alpha_i,\beta_i}] = D$, for $1 \leq i \leq n$.

Therefore $\mathcal{E}[(\textsf{exists-self P})] = D$, since $\mathcal{E}[\mathsf{C}'_{\mathcal{P}}] = D$. Thus there exists $e \in D$ such that $e \in \mathcal{E}[(\textsf{all (self) P})]$ and thus $\langle e, e\rangle \in \mathcal{E}[\mathsf{P}]$. Therefore $\mathcal{P}$ has a solution.

Suppose $\mathcal{P}$ has a solution.

Then let $s = \langle D, \mathcal{E}\rangle$ be an arbitrary semantic structure. Consider $d \in D$ for which

$d \in \mathcal{E}[\mathsf{U}]$,
$d \in \mathcal{E}[(\textsf{unique A})]$,
$d \in \mathcal{E}[(\textsf{function F}_0)]$,
$d \in \mathcal{E}[(\textsf{function F}_1)]$,
$d \in \mathcal{E}[(\textsf{all T (restrict T (atmost 1 G}_0)))]$,
$d \in \mathcal{E}[(\textsf{all T (restrict T (atmost 1 G}_1)))]$,
$d \in \mathcal{E}[(\textsf{all T (restrict T (all (self) (compose F}_0\ \textsf{G}_0))))]$,
$d \in \mathcal{E}[(\textsf{all T (restrict T (all (self) (compose F}_1\ \textsf{G}_1))))]$,
$d \in \mathcal{E}[\mathsf{I}'_{\alpha_i,\beta_i}]$, for $1 \leq i \leq n$, and
$d \in \mathcal{E}[\mathsf{J}'_{\alpha_i,\beta_i}]$, for $1 \leq i \leq n$.

Let $D' = \{e \in D : \langle d, e\rangle \in \mathcal{E}[\mathsf{T}]\}$. Then, as in the main proof, there is a unique $e \in D'$ such that $e \in \mathcal{E}[\mathsf{A}]$. Also, for all $e \in D'$ there is a unique $e' \in D'$ such that $\langle e, e'\rangle \in \mathcal{E}[\mathsf{F}_0]$ and a unique $e'' \in D'$ such that $\langle e, e''\rangle \in \mathcal{E}[\mathsf{F}_1]$. Define α', for $\alpha \in \{0,1\}^*$, and $e\alpha'$, for $\alpha \in \{0,1\}^+$ and $e \in D'$, as in the main proof.

Since

$d \in \mathcal{E}[(\textsf{function F}_0)]$,
$d \in \mathcal{E}[(\textsf{all T (restrict T (atmost 1 G}_0)))]$, and
$d \in \mathcal{E}[(\textsf{all T (restrict T (all (self) (compose F}_0\ \textsf{G}_0))))]$,

therefore, for $e, e' \in D'$, $\langle e, e'\rangle \in \mathsf{F}_0$ iff $\langle e', e\rangle \in \mathsf{G}_0$. Similarly, for $e, e' \in D'$, $\langle e, e'\rangle \in \mathsf{F}_1$ iff $\langle e', e\rangle \in \mathsf{G}_1$.

Then $\langle \alpha'_i, \beta'_i\rangle \in \mathcal{E}[\mathsf{P}]$, for $1 \leq i \leq n$, because $d \in \mathcal{E}[\mathsf{I}'_{\alpha_i,\beta_i}]$, and $\langle e, e'\rangle \in \mathcal{E}[\mathsf{P}]$ implies $\langle e\alpha'_i, e'\beta'_i\rangle \in \mathcal{E}[\mathsf{P}]$, for $1 \leq i \leq n$, because $d \in \mathcal{E}[\mathsf{J}'_{\alpha_i,\beta_i}]$. Therefore, if $\langle \alpha, \beta\rangle$ is a presolution of $\mathcal{P}$ then $\langle \alpha', \beta'\rangle \in \mathcal{E}[\mathsf{P}]$. Since $\mathcal{P}$ has a solution, thus there exists $\alpha \in \{0,1\}^+$ such that $\langle \alpha, \alpha\rangle$ is a presolution of $\mathcal{P}$. Thus $\langle \alpha', \alpha'\rangle \in \mathcal{E}[\mathsf{P}]$, and, since $\alpha' \in D'$ for all $\alpha \in \{0,1\}^*$, thus $d \in \mathcal{E}[(\textsf{exists-self P})]$.

Therefore, as in the main proof, for all $d \in D$, $d \in \mathcal{E}[\mathsf{C}'_{\mathcal{P}}]$, and thus $\mathsf{C} \Rightarrow \mathsf{C}'_{\mathcal{P}}$.

Therefore $\mathsf{C} \Rightarrow \mathsf{C}'_{\mathcal{P}}$ iff $\mathcal{P}$ has a solution, and thus, since the Post correspondence problem is undecidable, so too is subsumption in the terminological logic without role inverses. ■