A note on encoding inverse roles and functional restrictions in $\mathcal{ALC}$ knowledge bases

Diego Calvanese, Giuseppe De Giacomo, Riccardo Rosati
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Salaria 113, 00198 Roma, Italy
{calvanese,degiacomo,rosati}@dis.uniroma1.it

1 Introduction

In this paper we show that it is possible to eliminate *inverse roles* and *functional restrictions* from $\mathcal{ALCFI}$ knowledge bases, while preserving the soundness and completeness of inference. Specifically, we present two polynomial encodings, the first from $\mathcal{ALCFI}$ knowledge bases into $\mathcal{ALCI}$ ones, and the second from $\mathcal{ALCI}$ knowledge bases into $\mathcal{ALC}$ ones. These encodings eliminate functional restrictions and inverse roles respectively, but add enough information so as not to destroy the meaning of concepts in the original knowledge base with respect to the reasoning tasks (in particular we will focus on logical implication).

The encodings presented here are derived from those in [De Giacomo and Lenzerini, 1994] and in [De Giacomo, 1996] (the latter in the context of Propositional Dynamic Logics) for much more expressive description logics, in which complex roles formed as regular expressions of atomic ones (including the reflexive-transitive closure) are allowed. Observe that, if we apply directly the encodings in [De Giacomo and Lenzerini, 1994; De Giacomo, 1996] to $\mathcal{ALCFI}$ knowledge bases, reflexive-transitive closure would be introduced to internalize axioms, and hence it would appear in the syntactic closure as well. As a consequence the resulting formula would not be expressible as an $\mathcal{ALCFI}$ knowledge base. However it can be shown that the parts not expressible as $\mathcal{ALCFI}$ assertions can be dropped without influencing the reasoning tasks. The encodings presented here are devised by making use of this result.

Encoding inverse roles and functional restrictions in $\mathcal{ALC}$ knowledge bases, on the one hand, is of practical interest, since it allows for basing the "core inference procedures" for logical implication in $\mathcal{ALCFI}$ on the inference procedures for logical implication in $\mathcal{ALC}$, which are typically more efficient (e.g. constraint systems [Buchheit *et al.*, 1993]) and for which implemented systems are already available (e.g. FACT [Horrocks, 1997]). On the other hand, such encodings are a simple illustration of a general technique for deriving reasoning procedures for expressive logics based on a (possibly polynomial) encoding of such logics into simpler ones. Intuitively, the technique is based on two main steps. Let the "Source Logic" be SL and the "Target Logic" be TL:

1. Identify a finite set of assertion schemas in the language of TL capturing those characteristics that distinguish SL from TL.
2. Devise a function that, given an SL knowledge base $\mathcal{K}$, returns a finite set of SL concepts whose interpretation uniquely determines that of $\mathcal{K}$, and which will be used to instantiate the assertion schemas in (1).

If both the cardinality of the sets in (1) and (2) and the size of their elements are polynomially bounded by the original concept, then so is the knowledge base we get. Such a technique has led to establish several decidability and complexity results, as well as reasoning procedures in DLs [De Giacomo and Lenzerini, 1994; Calvanese *et al.*, 1995; De Giacomo and Lenzerini, 1995; 1996].

2 The description logic $\mathcal{ALCFI}$

The description logic $\mathcal{ALCFI}$ has the following constructs:

$$\begin{array}{rcl} C & ::= & A \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \\ & & \forall Q.C \mid \exists Q.C \mid (\leq 1\, Q) \\ Q & ::= & R \mid R^- \end{array}$$

The semantics of the various constructs is the usual one (see e.g. [Donini *et al.*, 1996]). An $\mathcal{ALCFI}$ knowledge base $\mathcal{K}$ is a finite set of assertions of the form

$$C_1 \sqsubseteq C_2$$

where C_1 and C_2 are $\mathcal{ALCFI}$ concepts (without any restriction on cyclicity). The semantics of assertions is as usual. The reasoning service we are interested in is *logical implication* of the form $\mathcal{K} \models C_1 \sqsubseteq C_2$, that is, verifying if the assertion $C_1 \sqsubseteq C_2$ is satisfied in every interpretation that satisfies all assertions in $\mathcal{K}$. It is well-known that checking logical implication in $\mathcal{ALCFI}$ (as in $\mathcal{ALCI}$ and $\mathcal{ALC}$) is an EXPTIME-complete problem [De Giacomo and Lenzerini, 1994].

Given an $\mathcal{ALCFI}$ knowledge base $\mathcal{K}$, we call *syntactic closure* of $\mathcal{K}$ the set $CL(\mathcal{K})$ formed by all atomic concepts A, functional restrictions $(\leq 1\, Q)$, existential restrictions $\exists Q.C$, and universal restrictions $\forall Q.C$ in $\mathcal{K}$, and their negations. Both the number and the size of the formulae in $CL(\mathcal{K})$ are linearly bounded by the size of $\mathcal{K}$.

3 Eliminating functional restrictions

We now exhibit an encoding of $\mathcal{ALCFI}$ into $\mathcal{ALCI}$. Although such an encoding has a simple form, proving its correctness requires quite sophisticated manipulations on interpretations. In particular, we observe that $\mathcal{ALCFI}$ *does not have* the finite model property, while $\mathcal{ALCI}$ *does have* it. Hence filtration arguments, usual in modal logics, cannot be applied directly. We assume, without loss of generality, that $\mathcal{K}$ is in negation normal form (i.e. negations are pushed inside as much as possible).

Definition 1 Let $\mathcal{K}$ be an $\mathcal{ALCFI}$ knowledge base whose concepts are in negation normal form. We define the *$\mathcal{ALCI}$-counterpart* $\alpha(\mathcal{K})$ of $\mathcal{K}$ as the $\mathcal{ALCI}$ knowledge base $\alpha(\mathcal{K}) = \alpha_1(\mathcal{K}) \cup \alpha_2(\mathcal{K})$, where:

- $\alpha_1(\mathcal{K})$ is obtained from $\mathcal{K}$ by replacing each $(\leq 1\, Q)$ with a new atomic concept $A_{(\leq 1\, Q)}$, and each $\neg(\leq 1\, Q)$ with $(\exists Q.H_{(\leq 1\, Q)}) \sqcap (\exists Q.\neg H_{(\leq 1\, Q)})$, where $H_{(\leq 1\, Q)}$ is again a new atomic concept.
- $\alpha_2(\mathcal{K})$ is the set of assertions of the form:

$$A_{(\leq 1\, Q)} \sqcap \exists Q.C \quad \sqsubseteq \quad \forall Q.C$$

one for every $A_{(\leq 1\, Q)}$ occurring in $\alpha_1(\mathcal{K})$ and every $C \in CL(\alpha_1(\mathcal{K}))$. ∎

Intuitively, $\alpha_1(\mathcal{K})$ introduces the new concepts $A_{(\leq 1\, Q)}$ and $H_{(\leq 1\, Q)}$ in place of $(\leq 1\, Q)$, so that positive occurrences of $(\leq 1\, Q)$ are represented by the concept $A_{(\leq 1\, Q)}$, and negative occurrences are represented by $(\exists Q.H_{(\leq 1\, Q)}) \sqcap (\exists Q.\neg H_{(\leq 1\, Q)})$. Note that every instance of $(\exists Q.H_{(\leq 1\, Q)}) \sqcap (\exists Q.\neg H_{(\leq 1\, Q)})$ has at least two Q-successors. To understand the purpose of $\alpha_2(\mathcal{K})$ consider that the schema $A_{(\leq 1\, Q)} \sqcap \exists Q.\mathcal{C} \sqsubseteq \forall Q.\mathcal{C}$ (where $\mathcal{C}$ is to be replaced by every concept) characterizes exactly the functional restrictions. The set of assertions $\alpha_2(\mathcal{K})$ can be thought of as a finite instantiation of the schema above, with one instance for each concept in $CL(\alpha_1(\mathcal{K}))$. Imposing the validity of such finite instantiation is sufficient to guarantee that if $\alpha(\mathcal{K})$ has a model then it has a model which is a model of $\mathcal{K}$ as well, and vice-versa.

Theorem 2 *An $\mathcal{ALCFI}$ knowledge base $\mathcal{K}$ logically implies $A \sqsubseteq B$, where A and B are atomic concepts occurring in $\mathcal{K}$, if and only if its $\mathcal{ALCI}$-counterpart $\alpha(\mathcal{K})$ logically implies $A \sqsubseteq B$.*

We assume A and B atomic for convenience. This is not a limitation since they both can be put equivalent to complex concepts in $\mathcal{K}$.

4 Eliminating inverse roles

We now define the polynomial encoding β from $\mathcal{ALCI}$ knowledge bases into $\mathcal{ALC}$ knowledge bases.

Definition 3 Let $\mathcal{K}$ be an $\mathcal{ALCI}$ knowledge base. We define the *$\mathcal{ALC}$-counterpart* $\beta(\mathcal{K})$ of $\mathcal{K}$ as the set of assertions $\beta(\mathcal{K}) = \beta_1(\mathcal{K}) \cup \beta_2(\mathcal{K})$, where:

- $\beta_1(\mathcal{K})$ is obtained from $\mathcal{K}$ by replacing each occurrence of R^- with a new atomic role R^c, for every atomic role R occurring in $\mathcal{K}$.
- $\beta_2(\mathcal{K})$ is a set of assertions of the form:

$$C \quad \sqsubseteq \quad (\forall R.\exists R^c.C) \sqcap (\forall R^c.\exists R.C)$$

one for every $C \in CL(\beta_1(\mathcal{K}))$ and atomic role R occurring in $\mathcal{K}$. ∎

Intuitively, $\beta_1(\mathcal{K})$ replaces the inverse of atomic roles in $\mathcal{K}$ with new atomic roles. Each new role R^c is intended to represent R^- in $\beta_1(\mathcal{K})$. To understand the purpose of $\beta_2(\mathcal{K})$, consider that the assertion schema $\mathcal{C} \sqsubseteq (\forall R.\exists R^c.\mathcal{C}) \sqcap (\forall R^c.\exists R.\mathcal{C})$ (where $\mathcal{C}$ is to be replaced by every concept), characterizes R^c as the inverse of the role R. The set of assertions $\beta_2(\mathcal{K})$ can be thought of as a finite instantiation of the schema above, with one instance for each concept in $CL(\beta_1(\mathcal{K}))$. Imposing the validity of such finite instantiation is sufficient to guarantee that if $\beta(\mathcal{K})$ has a model then it has a model which is a model of the original knowledge base as well, and vice-versa.

Theorem 4 *An $\mathcal{ALCI}$ knowledge base $\mathcal{K}$ logically implies $A \sqsubseteq B$, where A and B are atomic concepts occurring in $\mathcal{K}$, if and only if its $\mathcal{ALC}$-counterpart $\beta(\mathcal{K})$ logically implies $A \sqsubseteq B$.*

5 Discussion

$\mathcal{ALCFI}$ knowledge bases are of special importance in database applications of DLs, such as capturing conceptual data models (e.g. the entity-relationship model), or representing interschema assertions in source integration [Calvanese *et al.*, 1998]. $\mathcal{ALCFI}$ is the simplest logic in which n-ary relations can be correctly represented (as reified concepts). In particular the logic $\mathcal{DLR}$ (a DL with built-in n-ary relations) on which the integration methodology in [Calvanese *et al.*, 1998] is based, is rephrasable in $\mathcal{ALCFI}$, if we omit number restrictions and hence renounce to express cardinality constraints. Implemented systems for such a logic currently do not exist. The presented encodings allow us to immediately extend available systems to deal with such a logic, thus giving us the possibility to start building prototypes for this class of applications.

References

[Buchheit *et al.*, 1993] Martin Buchheit, Francesco M. Donini, and Andrea Schaerf. Decidable reasoning in terminological knowledge representation systems. *J. of Artificial Intelligence Research*, 1:109–138, 1993.

[Calvanese *et al.*, 1995] Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. Structured objects: Modeling and reasoning. In *Proc. of the 4th Int. Conf. on Deductive and Object-Oriented Databases (DOOD-95)*, number 1013 in Lecture Notes in Computer Science, pages 229–246. Springer-Verlag, 1995.

[Calvanese *et al.*, 1998] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, Daniele Nardi, and Riccardo Rosati. Description logic framework for information integration. In *Proc. of the 6th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR-98)*, 1998.

[De Giacomo and Lenzerini, 1994] Giuseppe De Giacomo and Maurizio Lenzerini. Boosting the correspondence between description logics and propositional dynamic logics. In *Proc. of the 12th Nat. Conf. on Artificial Intelligence (AAAI-94)*, pages 205–212. AAAI Press/The MIT Press, 1994.

[De Giacomo and Lenzerini, 1995] Giuseppe De Giacomo and Maurizio Lenzerini. What's in an aggregate: Foundations for description logics with tuples and sets. In *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence (IJCAI-95)*, pages 801–807, 1995.

[De Giacomo and Lenzerini, 1996] Giuseppe De Giacomo and Maurizio Lenzerini. TBox and ABox reasoning in expressive description logics. In Luigia C. Aiello, John Doyle, and Stuart C. Shapiro, editors, *Proc. of the 5th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR-96)*, pages 316–327. Morgan Kaufmann, Los Altos, 1996.

[De Giacomo, 1996] Giuseppe De Giacomo. Eliminating "converse" from Converse PDL. *J. of Logic, Language and Information*, 5:193–208, 1996.

[Donini *et al.*, 1996] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Andrea Schaerf. Reasoning in description logics. In Gerhard Brewka, editor, *Principles of Knowledge Representation*, Studies in Logic, Language and Information, pages 193–238. CSLI Publications, 1996.

[Horrocks, 1997] Ian Horrocks. Optimisation techniques for expressive description logics. Technical Report UMCS-97-2-1, University of Manchester, Department of Computer Science, 1997.