

Discovery Using Heterogeneous Combined Logics

Joshua Taylor and Selmer Bringsjord

tayloj,selmer@rpi.edu

Computer and Cognitive Science Departments
Rensselaer Polytechnic Institute
Troy, NY USA

Introduction

Research in hybrid logic systems and, later, description logics, has revealed a tradeoff between the expressivity of a logical formalism, and the complexity of reasoning within that formalism. This is why, for instance, tractable inference procedures are known for certain classes of description logics and for (some) formalisms underlying knowledge representation on the Semantic Web.

Some traditions within artificial intelligence and knowledge representation have focused on more expressive knowledge representations. For some expressive representations, such as first-order logic, sound and complete proof calculi, such as the sequent calculus, have been developed. However, for the purposes of automated theorem proving, not all proof calculi are created equal. Engineering effective proof search strategies for some proof calculi is difficult, but other proof calculi, e.g., those based on resolution, do lend themselves to efficient proof search.

Recognizing the diversity of knowledge representation systems currently in existence, the different properties of proof calculi which may be employed over these systems, and the growing need to combine inferences made under multiple logical systems, we propose the development of formalisms to govern these interactions, and call this the study of *combined logics*.

Preliminaries

We review a few necessary preliminaries, and define the terms *heterogeneous* and *homogeneous* with respect to logical systems coupled with proof calculi.

Logical Systems

We use the common notion of *logical system*: a pair, $\langle \mathcal{L}, \vdash \rangle$, where \mathcal{L} is a logical language, and \vdash is a proof calculus.

A language \mathcal{L} defines the syntax of expressions within the system, specifying, perhaps, for first-order languages, a set of boolean connectives, predicate and function symbols, and quantifiers, or, for description logic languages, set of concept, role, and individual names, and concept and role constructors. \mathcal{L} must also define the semantic interpretation of expressions within the system.

A proof calculus \vdash specifies the inference rules and syntax of proofs, and a notion of validity. A proof calculus may or may not possess soundness and completeness.

Logical Genera

A number of relationships may hold between two logical systems, $L_1 = \langle \mathcal{L}_1, \vdash_1 \rangle$, and $L_2 = \langle \mathcal{L}_2, \vdash_2 \rangle$. Several are likely to be of interest, and we here define some relevant descriptive terms. Firstly, if the languages \mathcal{L}_1 and \mathcal{L}_2 are the same, then the logical systems are said to be *homoglottal*. If the two systems are not homoglottal, then they are *heteroglottal*.

In the case that L_1 and L_2 are homoglottal, any distinction between them originates in differences in \vdash_1 and \vdash_2 . One important property by which calculi are distinguished is the deductive closures they yield for a set of formulae, Φ . If their respective deductive closures are the same for any set Φ , then the logical systems L_1 and L_2 are said to be *homoglottal homogeneous* systems. A trivial result is that all logical systems produced by coupling a language \mathcal{L} with any sound and complete proof calculus are homoglottal and homogeneous. If two logical systems are not homogeneous, then they are *heterogeneous*.

Other formal relationships between the calculi of homoglottal systems can also be envisioned. For instance, we might define a relationship that holds between \vdash_1 and \vdash_2 if and only if for every set of formulae Φ , the deductive closure under \vdash_1 is a subset of the deductive closure under \vdash_2 . This would be the case, for instance, with inference algorithms for fragments of first-order logic that correspond to description logics and sound and complete calculi for first-order logic.

In the case that L_1 and L_2 are heteroglottal, then the comparison of \vdash_1 and \vdash_2 is more complicated. For instance, it could be the case that one of \mathcal{L}_1 and \mathcal{L}_2 is a sub-language of the other, in which case homogeneity and heterogeneity could be defined in a manner similar to the one present above (with some provision for the fact that not every set Φ of formulae under one language is actually a set of formulae under the other). More likely, however, is that the precise relationship of the systems will require knowledge of the particular semantics of the two systems. We make no claim to have fully formalized any such notions at this time.

Example

We consider two logical systems. The first is a description logic-based system of familial relationships. We do not elab-

orate on the proof calculus associated with this system. Decision procedures are known for certain classes of description logics, and we assert that the description logic of this system has such a decision procedure and associated calculus. A portion of the knowledgebase associated with the system follows.

$$Parent \sqsubseteq \exists hasChild.Person \quad (1)$$

$$Mother \equiv Parent \sqcap Woman \quad (2)$$

The second logical system has a traditional first-order syntax, semantics, and proof calculus. It is a formalization of a portion of tax code. The intention of the expression $[Dependant, Son, Daughter](x, y)$ is that x has y as a $[Dependant, Son, Daughter]$. $Eligible(x)$ indicates that x is eligible for a particular deduction. As with the first system, we will not examine the proof calculus of this system, except to say that it is sound and complete, and is in the natural deduction style. A portion of the knowledgebase that governs whether individuals are eligible for a deduction, based on whether they have a dependant, follows.

$$\forall x \exists y Dependant(x, y) \Rightarrow Eligible(x) \quad (3)$$

$$\forall x, y (Son(x, y) \vee Daughter(x, y)) \Rightarrow Dependant(x, y) \quad (4)$$

Within the description logic based system, it can be observed that since *Mother* is the intersection of the concepts *Parent* and *Woman*, and *Parent* is subsumed by the restriction of *hasChild* to *Person*, *Mother* must be subsumed by the same restriction. The decision procedures for various classes of description logics can confirm this observation, and so this result is provable within the first logical system.

Lemma 1. *The concept Mother is subsumed by the restriction of the role hasChild to the class Person. Symbolically, $Mother \sqsubseteq \exists hasChild.Person$.*

Within the second logical system, which is based on first-order logic and a natural deduction style proof calculus, an inference begins with the introduction of a new, unused name, say, a , in a new scope. The inference continues with the assumption of a disjunction, that either there exists a y that is the daughter of a , or that there exists a y that is the son of a . Then, as a proof by cases, it can be determined that a must have some dependent, and so a is eligible for the deduction. Exiting the scope of a yields a universal generalization, resulting in Lemma 2.

Lemma 2. *For any individual x , if there is another individual y , such that x has y as a dependant, then x is eligible for the deduction. Symbolically, $\forall x [\exists y Son(x, y) \wedge \exists y Daughter(x, y)] \Rightarrow Eligible(x)$.*

Combining Results

We now have two lemmas, Lemma 1 and Lemma 2. The description logic of the first system is equivalent to a fragment of first-order logic, and we may recognize an equivalence between Lemma 1 and the following sentence in a traditional first-order syntax.

$$\forall x Mother(x) \Rightarrow \exists y [Person(x) \wedge hasChild(x, y)] \quad (5)$$

In this form, it seems that (5) is semantically related to Lemma 2, although the two sentences are expressed in different languages, but those languages share a common (general) syntax and semantics. We can leverage our prior work in provability-based semantic interoperability (Taylor, Shilliday, and Bringsjord 2007; Shilliday, Taylor, and Bringsjord 2007), in mapping the first-order language with the vocabulary of the description logic system to the language of the second system is not a particularly difficult task.

Once such a mapping has been accomplished, it is not difficult to infer (6), stating that mothers are eligible for the deduction in question.

$$\forall x Mother(x) \Rightarrow Eligible(x) \quad (6)$$

Remarks

In this example, we have explored how different logical languages and proof calculi might be used in conjunction to obtain results which neither system could produce alone. We selected systems whose logical languages differ, but whose underlying semantics were very similar. This enabled us to gloss over translation from a description logic to a first-order logic. After the relevant information had been placed into systems using different vocabularies, but similar semantics, our existing work in provability-based semantic interoperability enabled us to perform inferences using multiple knowledge sources.

The choice of systems was not made in order to make this process easy, but rather to present a situation which is difficult, but probably achievable today, and so an ideal *starting point* for research in combined logics, particularly as they apply to wide-scale knowledge representation systems today. Future work would necessarily include examining: what types of relationships between logical systems can be exploited to make information sharing between systems that do not have such similar semantics feasible; what kinds of inference can be performed on shared knowledge; and planning methods to determine what inferences might be performed within single systems, and what inferences require information obtained from multiple systems.

References

- Shilliday, A.; Taylor, J.; and Bringsjord, S. 2007. Toward automated Provability-Based Semantic Interoperability between Ontologies for the Intelligence Community. In Hornsby, K. S., ed., *OIC-2007 Ontology for the Intelligence Community: Towards Effective Exploitation and Integration of Intelligence Resources*, 66–72. CEUR-WS.org.
- Taylor, J.; Shilliday, A.; and Bringsjord, S. 2007. Provability-Based Semantic Interoperability via Translation Graphs. In Hainaut, J. L., ed., *First International Workshop on Ontologies and Information Systems for the Semantic Web (ONISW 2007), ER Workshops 2007, LNCS 4802*, 180–189.