2-2013

Research Challenges and Opportunities in Knowledge Representation, Section 4.1.1 Hybrid KR

Natasha Noy
Stanford University

Deborah McGuinness
Rensselaer Polytechnic Institute

Yuliya Lierler
University of Nebraska at Omaha, ylierler@unomaha.edu

4.1.1 Hybrid KR

Written by Peter Patel-Schneider, Yulia Lierler

Using one particular language limits us to the problems that we can effectively represent (and reason with) in that language. Thus, if we want to gain some or all of the benefits of multiple languages, we must try to combine languages, for example combining description logics and temporal logics to represent changing ontologies, or combining rules and databases to permit simple reasoning over large numbers of facts. However, combining two languages often results in an increase in the complexity of reasoning. For example, combining description logics and rules in the obvious manner generally results in a language with undecidable reasoning, even though both components have decidable reasoning. Developing systematic means for combining (a) heterogeneous KR languages and (b) various reasoning techniques under one roof is by no means a solved issue.

There are some recent initial successes in this direction. For example, advances in satisfiability modulo theories (Barrett et al., 2009) and constraint answer set programming (Brewka et al., 2011) demonstrate a potential for this direction of research. For instance, constraint programming (Rossi et al., 2008) is an efficient tool for solving scheduling problems, whereas answer set programming (Brewka et al., 2011) is effective in addressing elaborate planning domains. Constraint answer set programming that unifies these two KR sub--fields is best for solving problems that require both scheduling and planning capabilities of underlying tools.

Similarly, Description Logic Rules combine description logics and rules but limits the scope of the rules to obtain decidable reasoning. In this way, we can obtain most of the benefits of the two (or more) languages, while still retaining the desirable features of the component languages. We need to perform this analysis for each combination of languages—a formidably difficult task.

We can also consider producing a loose combination, where the two languages exist mostly independently, with separate reasoners, communicating via some sort of lingua franca or common sub--language and using some sort of intermediary to translate between or otherwise control the separate reasoners. This sort of loose combination can also be used to combine several reasoners over the same language, where the reasoners handle, or are complete or effective on, different but potentially overlapping sub--languages. Scientists have used these loose combinations for quite some time, starting with blackboard systems and continuing up to modern performance systems like Watson (Ferrucci et al., 2010). Nevertheless, many problems remain in producing such combination systems, ranging from issues of allocating resources to issues related to characterization of the capabilities of the combination.