ENHANCING A HYBRID TABLEAUX ALGORITHM Alejandra Lorenzo
Matriculation n°: 7543

Thesis Abstract

Hybrid Logic is a formalism used to represent knowledge and enable inference on it. The term covers a number of logical systems living between modal and classical logic. As it offers a good trade-off between complexity and expressivity, its theory has been investigated for more than fifteen years. However theorem provers (and model builders) for this logic have only been developed relatively recently.

One of the most recent theorem provers for Hybrid Logic is HTab ([Hoffmann and Areces, 2003]). HTab is one of the few theorem provers (and model builders) for Hybrid Logic that support not only basic hybrid language but also the universal and difference modalities and the down-arrow binder. It was developed by the TALARIS team (INRIA), and is based on a prefixed tableaux method adapted from the terminating tableaux introduced by T. Bolander and P. Blackburn in [Bolander and Blackburn., 2007].

HTab's performance has been compared with a number of other theorems provers and the results are good. But although a number of optimizations have already been incorporated into HTab, a potentially interesting one has not, namely caching. Caching can be defined as the storing of intermediate results in order to avoid recomputing them again.

Caching has been applied to a family of logics closely related to Hybrid Logic, namely the Description Logic family. So it is natural to ask: will caching help in the case of Hybrid Logic, including Hybrid Logic with the global modality? The aim of this thesis is to find out whether this optimization will be useful in this setting or not. We will restrict our attention to basic Hybrid Logic and the logic enriched with the universal modality, as Hybrid Logics containing the down-arrow binder are known to be undecidable.

This thesis is organized into eight chapters. In the first chapter, we briefly introduce the two families of languages: Hybrid Logic and Description Logic, and provide the main definitions needed for the rest of the report. The aim of this section is to provide the theoretical background needed in order to understand the similarities and differences between these two families.

In Chapter 2 we present the tableau algorithm. After providing a brief history of tableaux in different logics, we introduce the specific algorithm covered in this thesis: the prefixed tableau calculus for Hybrid Logic. Finally we present HTab, an implementation of this calculus for Hybrid Logic.

Chapter 3 provides a general introduction to the caching optimization, and discusses the approaches for caching already existing for the case of Description Logics.

Chapters 4 to 6 are the core of this work, as they explain in more detail each of the caching approaches, and how they can — and cannot — be implemented in Hybrid Logic. Chapter 4 introduces the main ideas needed to carry out UNSAT caching for the Hybrid Logic H(@, A), and Chapter 5 discusses its implementation. In Chapter 6 we discuss two other methods: MIXED caching and Global caching. We show that MIXED caching cannot be used with Hybrid Logic, and discuss what GLOBAL caching for Hybrid Logic might look like.

In Chapter 7 we provide some testing examples, and some early results on the UNSAT caching implementation. This testing stage is extremely important in order to know effectively if the improvement was significant, or even successful. The testing was carried out using an automated testing environment developed at INRIA called GridTest.

Finally, Chapter 8 contains our conclusions: what we have learned during this work and what we would like to do next.