Splitting Epistemic Logic Programs

Pedro Cabalar
University of Corunna
Corunna, Spain
cabalar@udc.es

Jorge Fandinno
University of Toulouse
IRIT, CNRS, France
jorge.fandinno@irit.fr

Luis Fariñas del Cerro
University of Toulouse
IRIT, CNRS, France
farinas@irit.fr

Abstract

Epistemic logic programs constitute an extension of the stable models semantics to deal with new constructs called *subjective literals*. Informally speaking, a subjective literal allows checking whether some regular literal is true in all stable models or in some stable model. As it can be imagined, the associated semantics has proved to be non-trivial, as the truth of the subjective literal may interfere with the set of stable models it is supposed to query. As a consequence, no clear agreement has been reached and different semantic proposals have been made in the literature. Unfortunately, comparison among these proposals has been limited to a study of their effect on individual examples, rather than identifying general properties to be checked. In this paper, we propose an extension of the well-known splitting property for logic programs to the epistemic case. To this aim, we formally define when an arbitrary semantics satisfies the epistemic splitting property and examine some of the consequences that can be derived from that, including its relation to conformant planning and to epistemic constraints. Interestingly, we prove (through counterexamples) that most of the existing proposals fail to fulfill the epistemic splitting property, except the original semantics proposed by Gelfond in 1991.

Introduction

The language of *epistemic specifications*, proposed by Gelfond (1991), constituted an extension of disjunctive logic programming that introduced modal operators to quantify over the set of stable models (Gelfond and Lifschitz, 1988) of a program. These new constructs were later incorporated as an extension of the Answer Set Programming (ASP) paradigm in different solvers and implementations – see (Leclerc and Kahl, 2018b) for a recent survey. The new constructs, *subjective literals*, have the form $\mathbf{K}\, l$ and $\mathbf{M}\, l$ and allow respectively checking whether regular literal l is true in every stable model (cautious consequence) or in some stable model (brave consequence) of the program. In many cases, these subjective literals can be seen as simple queries, but what makes them really interesting is their use in rule bodies, what may obviously affect the set of stable models they are meant to quantify. This feature makes them suitable for modelling introspection but, at the same time, easily involves cyclic specifications whose intuitive behaviour is not always easy to define. For instance, the semantics of an epistemic logic program may yield alternative sets of stable models, each set being called a *world view*. Deciding the intuitive world views of a cyclic specification has motivated a wide debate in the literature. In fact, in (Gelfond, 1991) original semantics or in its extension (Truszczyński, 2011), some cyclic examples manifested self-supportedness, so Gelfond (2011) himself and, later on, other authors (Kahl, 2014; Kahl et al., 2015; Fariñas del Cerro, Herzig, and Su, 2015; Shen and Eiter, 2017; Son et al., 2017) proposed different variants trying to avoid unintended results, without reaching a clear agreement. Unfortunately, comparison among these variants was limited to studying their effect on a set of "test" examples. This methodology has proven to fall short in such an uncertain context: confidence in any proposal is always subject to the appearance of new counterintuitive examples. A much stronger method would be defining instead formal properties to be established, as this would cover complete families of examples and, hopefully, could help to reach an agreement on some language fragments. For instance, one would expect that, at least, the existing approaches agreed on their interpretation of acyclic specifications. Regretfully, as we will show later, this is not the case.

In this paper we propose a candidate property, we call *epistemic splitting*, that not only defines an intuitive behaviour for stratified epistemic specifications but also goes further, extending the splitting theorem (Lifschitz and Turner, 1994), well-known for standard logic programs, to the epistemic case. Informally speaking, we say that an epistemic logic program can be split if a part of the program (the *top*) only refers to the atoms of the other part (the *bottom*) through subjective literals. A given semantics satisfies epistemic splitting if, given any splitted program, it is possible to get its world views by first obtaining the world views of the bottom and then using the subjective literals