

Lazy Model Expansion: Interleaving Grounding with Search

Broes De Cat*

Marc Denecker

Maurice Bruynooghe

Bart Bogaerts

*Department of Computer Science, KU Leuven
Celestijnenlaan 200A, 3001 Heverlee, Belgium*

BROES.DECAT@CS.KULEUVEN.BE

MARC.DENECKER@CS.KULEUVEN.BE

MAURICE.BRUYNOOGHE@CS.KULEUVEN.BE

BART.BOGAERTS@CS.KULEUVEN.BE

Peter J. Stuckey

*Doug McDonnell Building, Building 168
University of Melbourne, 3010 Australia*

PJS+123@CS.MU.OZ.AU

Abstract

Finding satisfying assignments for the variables involved in a set of constraints can be cast as a (bounded) *model generation* problem: search for (bounded) models of a theory in some logic. The state-of-the-art approach for bounded model generation for rich knowledge representation languages, like Answer Set Programming (ASP), $\text{FO}(\cdot)$ and the CSP modeling language Zinc, is *ground-and-solve*: reduce the theory to a ground or propositional one and apply a search algorithm to the resulting theory.

An important bottleneck is the blowup of the size of the theory caused by the reduction phase. *Lazily grounding* the theory during search is a way to overcome this bottleneck. We present a theoretical framework and an implementation in the context of the $\text{FO}(\cdot)$ knowledge representation language. Instead of grounding all parts of a theory, *justifications* are derived for some parts of it. Given a partial assignment for the grounded part of the theory and valid justifications for the formulas of the non-grounded part, the justifications provide a recipe to construct a complete assignment that satisfies the non-grounded part. When a justification for a particular formula becomes invalid during search, a new one is derived; if that fails, the formula is split in a part to be grounded and a part that can be justified.

The theoretical framework captures existing approaches for tackling the grounding bottleneck such as lazy clause generation and grounding-on-the-fly, and presents a generalization of the 2-watched literal scheme. We present an algorithm for lazy model expansion and integrate it in a model generator for $\text{FO}(\text{ID})$, a language extending first-order logic with inductive definitions. The algorithm is implemented as part of the state-of-the-art $\text{FO}(\text{ID})$ Knowledge-Base System IDP. Experimental results illustrate the power and generality of the approach.

*. Broes De Cat is funded by the Agency for Innovation by Science and Technology in Flanders (IWT).