

Syntaxe et sémantique d'une extension d'ASP avec variables existentielles

Fabien Garreau¹

¹ LERIA, 2 boulevard Lavoisier 49000 Angers

¹ LIRMM, 34090 Montpellier

fgarreau@info.univ-angers.fr

Résumé

Les règles existentielles [1] et l'Answer Set Programming (ASP [2, 3]) sont deux langages utilisés pour représenter des connaissances sous formes de règles. Le premier est issu des graphes conceptuels et le second de la programmation logique. Nous souhaitons étendre l'ASP afin de couvrir une partie des logiques de descriptions [4], dites légères, représentées par les règles existentielles. Pour cela, une des caractéristiques principales est l'ajout de variables quantifiées existentiellement dans la tête de règle. Nous détaillons dans ce papier, la syntaxe et la sémantique d'une extension de l'ASP appelée \exists ASP. Le but étant d'avoir un langage capable d'interpréter les benchmarks des deux langages et de fournir des réponses adéquates. Le solveur ASPeRiX [5] sera ensuite modifié pour traiter les problèmes de ce nouveau langage.

Références

- [1] J-F. Baget, M. Leclère, M-L. Mugnier, and E. Salvat. On rules with existential variables : Walking the decidability line. *AIJ*, 2011.
- [2] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Proc. International Conference on Logic Programming*, 1988.
- [3] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 1991.
- [4] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. DL-lite : Tractable description logics for ontologies. In *Proc. AAAI*, 2005.
- [5] C. Lefèvre and P. Nicolas. A first order forward chaining approach for answer set computing. In *Proc. LPNMR*, 2009.