Answer Set Programming with Constraints using Lazy Grounding - Proofs

A. Dal Palù[1], A. Dovier[2], E. Pontelli[3], and G. Rossi[1]

[1] Dip. Matematica, Univ. Parma,
{alessandro.dalpalu|gianfranco.rossi}@unipr.it
[2] Dip. Matematica e Informatica, Univ. Udine, dovier@dimi.uniud.it
[3] Dept. Computer Science, New Mexico State Univ., epontell@cs.nmsu.edu

1 Correctness and Completeness of GASP computations

This paper shows the proofs omitted in the paper appeared in [1]. For details and complete notation descriptions, please refer to that paper.

1.1 Definitions recall

A GASP-program can be seen as a syntactic shorthand for an ASP program where any non-ground GASP-rule represents a family of ground ASP rules. Let $\mathcal{A}$ be a collection of propositional atoms. An *ASP rule* has the form:

$$p \leftarrow p_0, \ldots, p_n, \mathbf{not}\, p_{n+1}, \ldots, \mathbf{not}\, p_m$$

where $\{p, p_0, \ldots, p_n, p_{n+1}, \ldots, p_m\} \subseteq \mathcal{A}$. An ASP program P is a collection of ASP rules.

An ASP model for a program P can be described by a *3-interpretation* I, i.e., a pair $\langle I^+, I^- \rangle$ such $I^+ \cup I^- \subseteq \mathcal{A}$ and $I^+ \cap I^- = \emptyset$. I^+ denotes the atoms that are known to be true while I^- denotes those atoms that are known to be false.

Given an ASP program P and a 3-interpretation I, we denote with $P \cup I$ the program

$$P \cup I = (P \setminus \{r \in P \mid \mathit{head}(r) \in I^-\}) \cup I^+.$$

Intuitively, $P \cup I$ is the program P modified in such a way to guarantee that all elements in I^+ are true and all elements in I^- are false.

Definition 1 (GASP-computation). *A* GASP-computation *of a program P is a sequence of 3-interpretations $I_0, I_1, I_2, \ldots$ that satisfies the following properties:*
- $I_0 = \mathsf{wf}(P)$
- $I_i \subseteq I_{i+1}$ *for all* $i \geq 0$ *(*Persistence of Beliefs*)*
- *if* $I = \bigcup_{i=0}^{\infty} I_i$*, then* $\langle I^+, \mathcal{A} \setminus I^+ \rangle$ *is a model of* P *(*Convergence*)*
- *for each* $i \geq 0$ *there exists a rule* $a \leftarrow \mathit{body}$ *in* P *that is applicable w.r.t.* I_i *and* $I_{i+1} = \mathsf{wf}(P \cup I_i \cup \langle \mathit{body}^+, \mathit{body}^- \rangle)$ *(*Revision*)*
- *if* $a \in I^+_{i+1} \setminus I^+_i$ *then there is a rule* $a \leftarrow \mathit{body}$ *in* P *which is applicable w.r.t.* I_j*, for each* $j \geq i$ *(*Persistence of Reason*).*

1.2 Proofs

Theorem 1 (correctness). *Given a program P, if there exists a* GASP*-computation that converges to a 3-interpretation I, then I is an answer set of P.*

Proof Sketch. The proof of correctness can be derived from a simple rewriting of a GASP-computation to an ASP computation as defined in [**?**]. Each step from I_i to I_{i+1} requires a well-founded model computation, that can be captured as a sequence of steps in the simpler notion of ASP computation. □

The proof of completeness of the GASP-computation can be derived with simple modifications from the analogous proof for the completeness of the basic algorithm used by SMODELS [**?**]. First of all, we can show that the basic step which moves from one step of the computation I_i to the successive one I_{i+1} preserves answer sets w.r.t. the body of the rule being applied.

Lemma 1. *Let us consider a 3-interpretation I and let $a \leftarrow body$ be a rule applicable w.r.t. I. Then $I' = \mathsf{wf}(P \cup I \cup \langle body^+, body^- \rangle)$ satisfies the following properties*

- $I \subseteq I'$
- *if M is an answer set of P such that $I \cup \langle body^+, body^- \rangle \subseteq M$, then $I' \subseteq M$.*

This result is an immediate consequence of the properties of the well-founded model of a program. The next result justifies the existence of a computation starting from a consistent point in the computation. Let us refer to a A-GASP-computation as a GASP-computation whose starting point I_0 is A.

Lemma 2. *Let M be an answer set of P and let A be a partial 3-interpretation such that $\mathsf{wf}(A \cup P) \subseteq M$. There exists a $\mathsf{wf}(A \cup P)$-GASP-computation that converges to M.*

Proof Sketch: Let us denote with $Atoms(A) = A^+ \cup A^-$. We can prove this result by induction on the number $n = \mathcal{A} \setminus Atoms(A)$.

If $n = 0$ then this means $A = M$; in this case $\mathsf{wf}(P \cup A) = A = M$, thus there is a $\mathsf{wf}(P \cup A)$-GASP-computation (composed of the single step I_0).

Let us consider the induction step. Since $n > 0$, this means that there are some atoms in M and not in A. First of all observe that if $M^+ = A^+$, then $\mathsf{wf}(A \cup P) = M$, and the result is immediate (there is a one-step $\mathsf{wf}(A \cup P)$-GASP-computation).

Let us consider the case where $M^+ \neq A^+$, and let $a \in M^+ \setminus A^+$. Clearly, there must be a rule $a \leftarrow body$ such that $M \models body$. Note that $\mathsf{wf}(A \cup P \cup \langle body^+, body^- \rangle)$ is a subset of M. From the inductive hypothesis, we know that there is a $\mathsf{wf}(A \cup P \cup \langle body^+, body^- \rangle)$-$\mathsf{GASP}$-computation that converges to M. This can be extended to a computation that starts from $\mathsf{wf}(P \cup A)$ by adding an initial step that makes use of the rule $a \leftarrow body$. □

Theorem 2 (completeness). *Given a program P and an answer set I of P, there exists a* GASP*-computation that converges to I.*

Proof. Immediate from lemma 2 by considering $A = \emptyset$. □

References

1. A. Dal Palù, A. Dovier, G. Rossi, and E. Pontelli. Answer Set Programming with Constraints using Lazy Grounding. *ICLP 2009*, Proceedings of the International conference of Logic Programming.