A proof of Theorem 1

Authors

Let us present a proof sketch of Sketch Rewriting Theorem 1.

Theorem 1 (Sound and Complete Sketched Rewriting)**.** *A sketched ASP program* $(P, S, D, f, \mathbf{E}^+, \mathbf{E}^-)$ *has a satisfying substitution* θ *iff the rewritten ASP program* P' *has an answer set.*

Lemma 1 (Preferences do not affect the decision version of SkASP)**.** *A sketched ASP program* $(P, S, D, f, \mathbf{E}^+, \mathbf{E}^-)$ *has a satisfying substitution* θ' *(preferred) iff* $(P, S, D, \mathbb{1}, \mathbf{E}^+, \mathbf{E}^-)$ *has a satisfying substitution* θ*, where* $\mathbb{1}(x) \equiv 1$ *is the constant function, i.e., no preferences – all substitutions are equal.*

Proof.

($\Longleftarrow$) If θ does not exist, then θ' does not exist neither, since the set of preferred solutions is the subset of all solutions. If θ, there are two cases, either θ is preferred in which we are done,since $\theta' = \theta$, or θ is not preferred. If so, then by definition of "not preferred" there must exist θ^1 that dominates θ, by transitivity of dominance, there must be a chain of preferences over the substitutions: θ^n is preferred over θ^{n-1} and so on until θ^1 which is preferred over θ, such that θ^n is not dominated by any other substitution. Then, from the existence of θ follows the existence of a preferred solution.

($\Longrightarrow$) Since the set of preferred solutions is a subset of the set of all solutions, this direction is trivial. If θ' exists, then θ can be put equal to θ. If no preferred solution θ' exists, assume that there is a θ but θ is not preferred. Then, θ must be dominated by a solution θ'' which is preferred, contradiction. □

Lemma 2 (Single Positive Example Substitution)**.** *Let* $(P, S, D, \mathbb{1}, \mathbf{E}^+, \mathbf{E}^-)$ *be a SkASP,* P' *be a rewritten program,* i *be an index of a positive example* e_i *in* $\mathbf{E}^+$*, then the SkASP* $(P, S, D, \mathbb{1}, \{e_i\}, \emptyset)$ *has a satisfying assignment* θ *iff the program* $W = P'[E \mapsto i]$ *has an answer set (where* E *is the example variable, see example expansion).*

Proof.

Since $E \mapsto i$, no other example with index different from i can contribute to the inference or firing of any rules and can be ignored. Then, from the decision generating step we know that there is one-to-one mapping between decision variables and θ, therefore we construct θ as $\{s_i \mapsto d_i\}$ from *decision_*$s_i(d_i)$ and vice versa. Since each decision predicate has one and only one value, this is always one-to-one mapping. Furthermore, let us demonstrate that if a fact $t(x_1, \dots, x_n)$ is present in an answer set A of W, then there a fact $t(x_1, \dots, x_n)$ or the fact $t(i, x_1, \dots, x_n)$ in an AS A' of $P\theta \cup e_i$. To show it, note each body has *example*(i), which is satisfied by the index i, then we perform the following satisfiability preserving transformation: remove all other examples facts from W and all guards *example(i)* from all inference rules and all example index i arguments from the literals. It is satisfiability preserving, since no fact and no inference rule can be satisfied by any other example index, then inference rules are the same in W and in P' up to the sketched variables. Then since there is one-to-one correspondence, the semantics of any reified predicate s_i is equivalent to its substitution d_i in a rule, due to the definition of the reification rules of s_i for its domain d_i.

($\Longleftarrow$) Assume there is an AS A for W (with the transformation described above), then we construct θ out of it and demonstrate that it satisfies all constraints. Then, we know that no positive rule is fired over A

$$\leftarrow r, \textit{positive(i)}.$$

Then the constraint

$$\leftarrow r\theta.$$

has not fired, since for θ corresponding *decision_*$s_i(d_i)$ the rule r is not satisfied in A and consequently not satisfied in A' for which we have established correspondence with A.

($\Longrightarrow$) $P\theta \cup e_i$ has an AS A', then we construct *decision_*$s_i(d_i)$ out of θ. Assume that we performed the same satisfiability preserving transformation as above,then W has an answer set, since no of its positive rules is satisfied. Let us indicate why, any rule

$$\leftarrow r\theta.$$

in $P\theta \cup e_i$ is not satisfied in A', then the set of atoms A that consist of decision variables above, the ground example atoms and all facts inferred by the non-integrity rules in $P\theta \cup e_i$ is an AS of W, since no of the positive rules

$$\leftarrow r, \textit{positive(i)}.$$

in W is fired, otherwise due to equivalence in semantics between reified predicates with their substitution and correspondence between θ and decision atoms, the rule $\leftarrow r\theta$. would be satisfied. □

Lemma 3 (Single Negative Example Substitution)**.** *Let* $(P, S, D, \mathbb{1}, \mathbf{E}^+, \mathbf{E}^-)$ *be a SkASP,* P' *be a rewritten program,* i *be an index of a negative example* e_i *in* $\mathbf{E}^-$*, then the SkASP* $(P, S, D, \mathbb{1}, \emptyset, \{e_i\})$ *has no satisfying assignment* θ *iff the program* $W = P'[E \mapsto i]$ *has no answer set (where* E *is the example variable, see example expansion).*

Proof. Similar to the proof of Lemma 2. □

Let us indicate how Theorem 1 follows from Lemmas 1, 2, 3.

Proof. First, we replace the SkASP $(P, S, D, f, \mathbf{E}^+, \mathbf{E}^-)$ with $(P, S, D, \mathbb{1}, \mathbf{E}^+, \mathbf{E}^-)$ due to Lemma 1.

Then for each positive e_i we apply Lemma 2 and for each negative we apply Lemma 3 and obtain the program $T = \bigwedge_{e_i \in \mathbf{E}^+ \cup \mathbf{E}^-} (P'[E \mapsto e_i])e_i])$, such it has an AS iff there is a θ satisfying all positive and none of the negatives. □