Proofs of the theorems in *FastLAS: Scalable Inductive Logic Programming incorporating Domain-specific Optimisation Criteria*

Mark Law
Imperial College London, UK
mark.law09@imperial.ac.uk

Alessandra Russo
Imperial College London, UK
a.russo@imperial.ac.uk

Elisa Bertino
Purdue University, USA
bertino@purdue.edu

Krysia Broda
Imperial College London, UK
k.broda@imperial.ac.uk

Jorge Lobo
ICREA - Universitat Pompeu Fabra
jorge.lobo@upf.edu

Given a program P with exactly one answer set, for any atom a, $P \models a$ denotes that a is in the answer set of P. Note that for every background knowledge B and example e_{ctx}, $B \cup e_{ctx}$ has exactly one answer set (by the definition of an ILP_{LAS}^{OPL} task). Due to the restrictions on M_h's, any hypothesis H can only define atoms which do not occur in the body of any rule in $B \cup H \cup e_{ctx}$. This means (by the splitting set theorem (Lifschitz and Turner 1994)) that the answer sets of $B \cup H \cup e_{ctx}$ are the answer sets of $A \in AS(H \cup A')$, where A' is the unique answer set of $B \cup e_{ctx}$. As $H \cup A'$ is stratified, this means that for any H, $B \cup H \cup e_{ctx}$ has a unique answer set.

Lemma 1. *Let e be an example in E^+ and $H \subseteq S_M$. For any ground atom $\mathtt{a} \in e_{pi}^{inc} \cup e_{pi}^{exc}$, $B \cup H \cup e_{ctx} \models \mathtt{a}$ iff H contains at least one rule that is a subrule of a rule in $\mathcal{C}(T, a, e)$.*

Proof.
$B \cup H \cup e_{ctx} \models a$
$\Leftrightarrow ground(B) \cup ground(H) \cup ground(e_{ctx}) \models a$
$\Leftrightarrow$ a is in the unique answer set of $ground(H) \cup A'$, where A' is the unique answer set of $B \cup e_{ctx}$.
$\Leftrightarrow \exists R^g \in ground(H)$ st $head(R^g) = a$ and $body(R^g)$ is satisfied by the unique answer set of $B \cup e_{ctx}$.
$\Leftrightarrow \exists R \in H$ st $\exists R^g \in ground(R)$ st $head(R^g) = a$ and $body(R^g)$ is satisfied by the unique answer set of $B \cup e_{ctx}$.
$\Leftrightarrow \exists R \in H$ st $\exists R^g \in ground(R)$ st $head(R^g) = a$ and $body(R^g)$ is satisfied by the unique answer set of $B \cup e_{ctx}$ and there is no rule $R' \in S_M$ st R is a strict subrule of R' and $\exists R^g \in ground(R')$ st $head(R^g) = a$ and $body(R^g)$ is satisfied by the unique answer set of $B \cup e_{ctx}$.
$\Leftrightarrow \exists R \in H$ st R is a subrule of at least one rule in $\mathcal{C}(T, a, e)$ (by Definition 5 of the main paper).

□

Proposition 1. *Let e be an example in E^+ and $H \subseteq S_M$. $B \cup H$ accepts e if and only if (1) for each $a \in e^{inc}$ there is at least one rule in H that is a subrule of a rule in $\mathcal{C}(T, a, e)$; and (2) for each $a \in e^{exc}$, no rule in H is a subrule of any rule in $\mathcal{C}(T, a, e)$.*

Proof.
$B \cup H$ accepts e
$\Leftrightarrow \exists A \in AS(B \cup H \cup e_{ctx})$ st A extends e_{pi}.
$\Leftrightarrow$ the unique answer set of $B \cup H \cup e_{ctx}$ extends e_{pi}.
$\Leftrightarrow \forall a \in e_{pi}^{inc}$, $B \cup H \cup e_{ctx} \models a$ and $\forall a \in e_{pi}^{exc}$, $B \cup H \cup e_{ctx} \not\models a$.
$\Leftrightarrow \forall a \in e_{pi}^{inc}$, there is at least one rule in H that is a subrule of a rule in $\mathcal{C}(T, a, e)$ and $\forall a \in e_{pi}^{exc}$, there is no rule in H that is a subrule of a rule in $\mathcal{C}(T, a, e)$ (by Lemma 1).

□

Theorem 1. *Let $\mathcal{C}(T) = \{R \mid e \in E^+, R \in e_I\}$. T is satisfiable iff $\mathcal{C}(T)$ contains an inductive solution of T.*

Proof.

- Assume T is unsatisfiable. Then there is no $H \subseteq S_M$ st $B \cup H$ accepts every example in E^+. Hence, as $\mathcal{C}(T) \subseteq S_M$, there is no $H \subseteq \mathcal{C}(T)$ such that $B \cup H$ accepts every example in E^+.
- Assume T is satisfiable. Then $\exists H \subseteq S_M$ st H accepts every example in E^+. $\forall e \in E^+, \forall a \in e_{pi}^{inc}$ there is at least one rule R_a in $\mathcal{C}(T, a, e)$, and hence in $\mathcal{C}(T)$, st H contains at least one subrule of R_a. Let H^* be the set of all such R_a's. We now show that H^* is an inductive solution of T.
  For each $e \in E^+, \forall a \in e_{pi}^{inc}$, H^* contains at least one rule that is a subrule of a rule in $\mathcal{C}(T, a, e)$. Assume for contradiction that there is an $a \in e_{pi}^{exc}$ for some $e \in E^+$ st H^* contains at least one subrule of a rule in $\mathcal{C}(T, a, e)$. Then H would also contain a subrule of this rule, meaning that by Proposition 1 H would not be an inductive solution of T. Hence, for each $e \in E^+, \forall a \in e_{pi}^{exc}$, H^* contains no rule that is a subrule of a rule in $\mathcal{C}(T, a, e)$. Hence, by Proposition 1, $B \cup H^*$ accepts all examples in E^+. Hence, H^* is an inductive solution of T. Hence, $\mathcal{C}(T)$ contains an inductive solution of T.

□

We use the notation $\mathcal{M}_{opt}^i(P, R, T)$ for $i \in [1, 5]$ to refer to the five components of $\mathcal{M}_{opt}(P, R, T)$ defined in the main paper. For simplicity, we assume that the ASP decompositions are stratified. In fact, the main results hold even if this is not the case, but the following lemma must be modified slightly to take into account that instead of there being one answer set per subrule, there would be one answer set for each answer set of the ASP decomposition for each subrule.

Lemma 2. *Let $\mathcal{S}[P]$ be a scoring function and R be a rule. For each subrule R' of R, there is exactly one answer set A of $\bigcup_{i\in[1,4]} \mathcal{M}^i_{opt}(P,R,T)$ such that $\mathcal{M}^{-1}_{rule}(A) = R'$. Furthermore, A satisfies the following properties:*

1. *For each $e \in E^+$, $\mathtt{v(e_{id})} \in A$ iff R' is a subrule of at least one rule in e_V.*
2. *The score of A (at its only priority level) is equal to $\mathcal{S}^{rule}(R',T)$.*

Proof.

First note that the set of all answer sets of $\bigcup_{i\in[1,4]} \mathcal{M}^i_{opt}(P,R,T)$ can be defined as

$$\left\{ A_1 \cup A_2 \;\middle|\; \begin{array}{c} R' \text{ is a subrule of } R, \\ A_1 \in AS(P \cup \mathcal{M}(R')), \\ A_2 \in AS(\mathcal{M}^4_{opt}(P,R,T) \cup \mathcal{M}(R')) \end{array} \right\}$$

and as the two programs $P \cup \mathcal{M}(R')$ and $\mathcal{M}^4_{opt}(P,R,T) \cup \mathcal{M}(R')$ are both stratified, this means that for each subrule R' of R there is exactly one answer set of $\bigcup_{i\in[1,4]} \mathcal{M}^i_{opt}(P,R,T)$ st $\mathcal{M}^{-1}_{rule}(A) = R'$.

1. Let $e \in E^+$. $\mathtt{v(e_{id})} \in A$ if and only if $\mathtt{v(e_{id})}$ is in the unique answer set of $\mathcal{M}^4_{opt}(P,R,T) \cup \mathcal{M}(R')$, which is the case if and only if R' is a subrule of at least one rule in e_V.
2. As the language of the weak constraints is disjoint from the rest of $\bigcup_{i\in[1,4]} \mathcal{M}^i_{opt}(P,R,T)$, the score of A is equal to the score of the unique answer set of $P \cup \mathcal{M}(R') \cup \mathcal{M}^3_{opt}(P,R,T)$, which is equal to $\mathcal{S}[P]^{rule}(R',T)$.

□

Theorem 3. *Let $\mathcal{S}[P]$ be a scoring function and R be a rule. Let AS be the optimal answer sets of $\mathcal{M}_{opt}(P,R,T)$. $\{\mathcal{M}^{-1}_{rule}(A) \mid A \in AS\}$ is the set of all optimisations of R.*

Proof.

1. We first show that every answer set in AS corresponds to an optimisation of R. Let $A \in AS$. Then, due to the constraints in $\mathcal{M}^5_{opt}(P,R,T)$ and by Lemma 2, $\mathcal{M}^{-1}_{rule}(A)$ is a subrule of R and for each $e \in E^+$ such that $e_{pen} = \infty$, $\mathcal{M}^{-1}_{rule}(A)$ is not a subrule of any rule in e_V.
   It remains to show that there is no subrule R'' of R such that $\mathcal{S}[P]^{rule}(R'',T) < \mathcal{S}[P]^{rule}(\mathcal{M}^{-1}_{rule}(A),T)$ and such that for each $e \in E^+$ st $e_{pen} = \infty$, $\mathcal{M}^{-1}_{rule}(A)$ is not a subrule of any rule in e_V. Assume for contradition that there is such an R''. Due to Lemma 2, if this were the case then there would be an answer set A' of $\mathcal{M}_{opt}(P,R,T)$ such that $\mathcal{M}^{-1}_{rule}(A') = R''$. Furthermore, by Lemma 2, the score of A is equal to $\mathcal{S}[P]^{rule}(\mathcal{M}^{-1}_{rule}(A),T)$ and the score of A' is equal to $\mathcal{S}[P]^{rule}(\mathcal{M}^{-1}_{rule}(A'),T)$. Hence, if this were the case, A would not be in AS. Contradiction! Hence, $\mathcal{M}^{-1}_{rule}(A)$ is an optimisation of R.
2. It remains to show that for every optimisation R' of R, there is an answer set $A \in AS$ such that $\mathcal{M}^{-1}_{rule}(A) = R'$. Let R' be an optimisation of R. By Lemma 2, there is an answer set A of $\bigcup_{i\in[1,4]} \mathcal{M}^i_{opt}(P,R,T)$ such that: $\mathcal{M}^{-1}_{rule}(A) = R'$; for each $e \in E^+$, $\mathtt{v(e_{id})} \in A$ iff R' is a subrule of at least one rule in e_V; and the score of A is equal to $\mathcal{S}^{rule}(R',T)$. As R' is an optimisation of R, A cannot contain $\mathtt{v(e_{id})}$ for any e st $e_{pen} = \infty$. Hence, A must satisfy the constraints in $\mathcal{M}^5_{opt}(P,R,T)$, and must be an answer set of $\mathcal{M}_{opt}(P,R,T)$.
   It remains to show that $A \in AS$ (i.e. that A is an optimal answer set of $\mathcal{M}_{opt}(P,R,T)$). Assume for contradiction that this is not the case. AS must contain at least one answer set (as the program is clearly satisfiable because A is an answer set), and therefore must contain at least one answer set A' with a lower score than A. But, by part (1) of this proof, A' corresponds to an optimisation R'' of R, and by Lemma 2 the score of A' is equal to $\mathcal{S}[P]^{rule}(R'',T)$. Hence, R' cannot be an optimisation, as it has a higher score than another optimisation R''. Contradiction! Hence, $A \in AS$.

□

For any rule R, let v_R be the set of all $e \in E^+$ st R is a subrule of a rule in e_V.

Theorem 4. *Let $\mathcal{S}[P]$ be a scoring function. $\bigcup_{R\in\mathcal{G}(T)} opt(P,R,T)$ is OPT-sufficient w.r.t. $(\mathcal{S}[P] + \mathcal{S}_{pen})$.*

Proof. Let $O = \bigcup_{R\in\mathcal{G}(T)} opt(P,R,T)$ and assume for contradiction that O is not OPT-sufficient. Then T must be satisfiable. Let H^* be an optimal inductive solution of T w.r.t. $\mathcal{S}[P]$. There must be a rule $h \in H^*$ for which there is no rule in $R \in O$ st $\mathcal{S}[P]^{rule}(h,T) = \mathcal{S}[P]^{rule}(R,T)$, $c_h = c_R$ and $v_R \subseteq v_h$.

There must be a rule R' in $\mathcal{G}(T)$ such that $c_{R'} = c_h$ and h is a subrule of R'. $opt(P,R',T)$ must not contain any rule R'' such that $v_{R''} \subseteq v_h$. But if this were the case then in the final iteration of the while loop, by Lemma 2, there would be an answer set A of $\mathcal{M}_{opt}(P,R',T)$ st $\mathcal{M}^{-1}_{rule}(A) = h$ and A did not violate any of the constraints in CS (note that $\{e \in v_h \mid e_{pen} = \infty\} = \emptyset$, so A does not violate any constraint in $\mathcal{M}^5_{opt}(P,R',T)$). This is a contradiction, as it would mean that the while loop would not terminate after the final iteration. Hence, O is OPT-sufficient. □

Meta representation

For a mode declaration m, let m_{args} be the list of arguments in m and let m' be the atom constructed from m by replacing these arguments with the variables $V_1, \ldots V_{|m_{args}|}$, and m'' be the atom constructed from m by replacing these arguments with the variables $Val_1, \ldots Val_{|m_{args}|}$.

Definition 1. *Let T be the ILP_{LAS}^{OPL} task $\langle B, M, E^+\rangle$, $e \in E^+$ and* a *be a ground atom. $\mathcal{M}(T, a, e)$ is the program*[1] *consisting of the following components:*

1. *The fact* $\mathtt{const(t, c)}$ *for each constant* c *of type* t.
2. *The fact* $\mathtt{var(v)}$ *for* $\mathtt{v} \in \{\mathtt{v1}, \ldots, \mathtt{vn}\}$*, where n is the maximum number of variable in a single rule.*
3. *The fact* $\mathtt{type(t)}$ *for each type* t.
4. *For each atom* $\mathtt{m} \in M_h$*, the fact* $\mathtt{modeh(m)}$.
5. *For each atom* $\mathtt{m} \in M_b$*, the fact* $\mathtt{modebp(m)}$.
6. *For each literal* $\mathtt{not\ m} \in M_b$*, the fact* $\mathtt{modebn(m)}$.
7. *For each atom* $\mathtt{m} \in M_{atoms}$*, the rule* $\mathtt{possible\_instantiation(m, m')} \mathtt{:-} \mathtt{b_1}, \ldots, \mathtt{b_{|m_{args}|}}$*, where for each $i \in [1, |m_{args}|]$, b_i is* $\mathtt{const(type, V_i)}$ *if the ith element of* $\mathtt{m_{args}}$ *is equal to* $\mathtt{const(type)}$ *and is* $\mathtt{var(type, V_i)}$ *if the ith element of* $\mathtt{m_{args}}$ *is equal to* $\mathtt{var(type)}$.
8. *The rules:*

```
1 { head(I) :
      modeh(M),
      possible_instantiation(M, I) } 1.

0 { in_body(pos(I)) } 1 :-
  modebp(M),
  possible_instantiation(M, I).

0 { in_body(neg(I)) } 1 :-
  modebn(M),
  possible_instantiation(M, I).
```

9. *The rules:*

```
equal(C, C) :- const(_, C).
0 { equal(V, C) : const(_, C) } 1 :- var(V).
1 { var(T, V) : type(T) } 1.
```

10. *For each atom* $\mathtt{m} \in M_{atoms}$*, the rule* $\mathtt{equal(m', m'')} \mathtt{:-} \mathtt{b_1}, \ldots, \mathtt{b_{|m_{args}|}}$*, where for each $i \in [1, |m_{args}|]$, b_i is* $\mathtt{equal(V_i, Val_i)}$.
11. *The reification of $B \cup e_{ctx}$, using the predicate* in_as.
12. *The fact* to_prove(a).
13. *The rules:*

```
:- to_prove(A), head(H), not equal(H, A).
:- in_body(pos(I)), equal(I, A), not in_as(A).
:- in_body(neg(I)), equal(I, A), in_as(A).
```

For any meta-level answer set (of $\mathcal{M}(T, a, e)$), $\mathcal{M}^{-1}(A)$ denotes the rule constructed from the head and in_body atoms in A.

Proposition 2. *Let T be the ILP_{LAS}^{OPL} task $\langle B, M, E^+\rangle$, $e \in E^+$ and* a *be a ground atom. $\{\mathcal{M}^{-1}(A) | A \in AS(\mathcal{M}(T, a, e))\}$ is the set of rules $R \in S_M$ such that $\exists R^g \in ground(R)$ st $\mathtt{a} = head(R^g)$ and the unique answer set of $B \cup e_{ctx}$ satisfies $body(R^g)$.*

Proof. Clearly, the first 8 components of $\mathcal{M}(T, a, e)$ represent S_M, so $\{\mathcal{M}^{-1}(A) | A \in AS(\mathcal{M}^{1-8}(T, a, e))\} = S_M$. It remains to show that the remaining five components constrain this to the subset of S_M such that at least one ground instance has a in the head and a satisfied body. The equal predicate generates all assignments of the variables, and therefore all ground instances of the instantiations of the modes. So the answer sets of $\mathcal{M}^{1-10}(T, a, e)$ correspond to the ground instances of rules in S_M. The only answer set of $\mathcal{M}^{11}(T, a, e)$ corresponds to the unique answer set of $B \cup e_{ctx}$. The constraints in $\mathcal{M}^{13}(T, a, e)$ therefore rule out any answer set corresponding to a ground instance R^g st (1) $head(R^g) \neq a$, (2) $B \cup e_{ctx} \not\models body^+(R^g)$, or (3) $B \cup e_{ctx} \not\models body^-(R^g)$. Hence, the answer sets of $\mathcal{M}(T, a, e)$ correspond to the ground instances R^g of rules in S_M st $\mathtt{a} = head(R^g)$ and the unique answer set of $B \cup e_{ctx}$ satisfies $body(R^g)$. Hence, $\{\mathcal{M}^{-1}(A) | A \in AS(\mathcal{M}(T, a, e))\}$ is the set of rules $R \in S_M$ such that $\exists R^g \in ground(R)$ st $\mathtt{a} = head(R^g)$ and the unique answer set of $B \cup e_{ctx}$ satisfies $body(R^g)$. ☐

Corollary 1. *Let T be the ILP_{LAS}^{OPL} task $\langle B, M, E^+\rangle$, $e \in E^+$ and* a *be a ground atom. Let AS_{proj} be the set of projections of the answer sets of $\mathcal{M}(T, a, e)$ over the predicates* head *and* in_body*. Then*

$$\left\{ \mathcal{M}^{-1}(A) \middle| \begin{array}{c} A \in AS_{proj}, \\ \nexists A' \in AS_{proj} \text{ st } A \subset A' \end{array} \right\} = \mathcal{C}(T, a, e).$$

Solving representation

Definition 2. *Let T be the ILP_{LAS}^{OPL} task $\langle B, M, E^+\rangle$, let O be a hypothesis space and let $\mathcal{S}$ be a decomposible scoring function. $\mathcal{M}_{solve}(T, \mathcal{S}, O)$ is the program consisting of the following components:*

1. *For each $h \in O$, the rule* $\mathtt{0\{in\_h(h_{id})\}1.}$ *and the weak constraint* $:\sim \mathtt{in\_h(h_{id})}.[\mathcal{S}^{\mathtt{rule}}(\mathtt{h}, \mathtt{T})@1, \mathtt{in\_h(h_{id})}]$
2. *For each example $e \in E^+$ with a finite penalty, the weak constraints:*
   - *For each $a \in e_{pi}^{inc}$, the weak constraint:* $:\sim \bigwedge_{\mathtt{h} \in \mathtt{O} \cap \mathcal{C}(\mathtt{T,a,e})} \mathtt{not\ in\_h(h_{id})}.[\mathtt{e_{pen}}@1, \mathtt{noise(e_{id})}]$.
   - $\forall h \in O \cap e_V$*:* $:\sim \mathtt{in\_h(h_{id})}.[\mathtt{e_{pen}}@1, \mathtt{noise(e_{id})}]$
3. *For each example $e \in E^+$ with an infinite penalty, the constraints:*
   - *For each $a \in e_{pi}^{inc}$, the constraint:* $\mathtt{:-} \bigwedge_{\mathtt{h} \in \mathtt{O} \cap \mathcal{C}(\mathtt{T,a,e})} \mathtt{not\ in\_h(h_{id})}$.
   - $\forall h \in O \cap e_V$*:* $\mathtt{:-}\,\mathtt{in\_h(h_{id})}$.

Given a task T and a scoring function $\mathcal{S}$, we say that $H \subseteq O$ is an optimal inductive solution of T in O w.r.t. $\mathcal{S}$ if $H \in ILP_{LAS}^{OPL}(T)$ and $\nexists H' \subseteq O$ st $H' \in ILP_{LAS}^{OPL}(T)$ and $\mathcal{S}(H', T) < \mathcal{S}(H', T)$.

Theorem 5. *Let T be the ILP_{LAS}^{OPL} task $\langle B, M, E^+\rangle$, O be a hypothesis space and $\mathcal{S}$ be a decomposible scoring function. Let AS be the optimal answer sets of $\mathcal{M}_{solve}(T, \mathcal{S}, O)$. The set of all optimal inductive solutions of T in O w.r.t. $(\mathcal{S} + \mathcal{S}_{pen})$ is equal to $\{\{h \in O \mid \mathtt{in\_h(h_{id})} \in A\} | A \in AS\}$.*

[1] This is a slight simplification of the (equivalent) more efficient representation used in FastLAS

Proof. The answer sets of $\mathcal{M}_{solve}(T, \mathcal{S}, O)$ correspond to all subsets of $\{\texttt{in\_h}(\texttt{h}_{\texttt{id}}) | h \in O\}$ that satisfy the constraints in the third part of the program. These constraints are satisfied by an answer set A if and only if there is no example with an infinite penalty that is not covered by the hypothesis corresponding to A. Hence, $\{\{h \in O \mid \texttt{in\_h}(\texttt{h}_{\texttt{id}}) \in A\} | A \in AS(\mathcal{M}_{solve}(T, \mathcal{S}, O))\}$ is equal to the set of inductive solutions that are contained in O. The penalty of each answer set A (at priority 1) is equal to the score $(\mathcal{S} + \mathcal{S}_{pen})(\{h \in O \mid \texttt{in\_h}(\texttt{h}_{\texttt{id}}) \in A\}, T)$. Hence, the set of all optimal inductive solutions of T in O w.r.t. $(\mathcal{S} + \mathcal{S}_{pen})$ is equal to $\{\{h \in O \mid \texttt{in\_h}(\texttt{h}_{\texttt{id}}) \in A\} | A \in AS\}$. $\square$

Note that as FastLAS constructs an OPT-sufficient hypothesis space using the *opt* algoithm (which is correct, as proven by Theorem 4), and then uses $\mathcal{M}_{solve}(T, \mathcal{S}, O)$ to find an optimal solution of T in O w.r.t. $(\mathcal{S} + \mathcal{S}_{pen})$, Theorem 5 suffices to show that FastLAS is sound and complete w.r.t. the optimal solutions of any task w.r.t. any scoring function $(\mathcal{S} + \mathcal{S}_{pen})$ st $\mathcal{S}$ is decomposible.

References

Lifschitz, V., and Turner, H. 1994. Splitting a logic program. In *ICLP*, volume 94, 23–37.