Appendix A Proofs

A.1 Main Lemma

Let us now extend the correspondence between stable models defined in terms of the SM operator and infinitary logic (Truszczyński 2012) to the above two-sorted case. We also allow formulas to contain extensional predicate symbols, which are not considered in (Truszczyński 2012).

We use the following notation. By $\mathcal{Z}$, we denote the set of all numerals, and by $\mathcal{T}$, we denote the set of all precomputed terms. For an interpretation I and a list $\mathbf{p}$ of predicate symbols, by $I^{\mathbf{p}}$, we denote the set of precomputed atoms $p(t_1, \dots, t_k)$ satisfied by I where $p \in \mathbf{p}$.

Let $\mathbf{p}, \mathbf{q}$ be a partition of the predicate symbols in the signature. Then, the *grounding of a sentence F with respect to an interpretation I and a set of intensional predicate symbols* $\mathbf{p}$ (and extensional predicate symbols $\mathbf{q}$) is defined as follows:

- $gr^{\mathbf{p}}_I(\bot) = \bot$;
- for $p \in \mathbf{p}$, $gr^{\mathbf{p}}_I(p(t_1, \dots, t_k)) = p((t_1^I)^*, \dots, (t_k^I)^*)$;
- for $p \in \mathbf{q}$, $gr^{\mathbf{p}}_I(p(t_1, \dots, t_k)) = \top$ if $p((t_1^I)^*, \dots, (t_k^I)^*) \in I^{\mathbf{q}}$ and $gr^{\mathbf{p}}_I(p(t_1, \dots, t_k)) = \bot$ otherwise;
- $gr^{\mathbf{p}}_I(t_1 = t_2) = \top$ if $t_1^I = t_2^I$ and $\bot$ otherwise;
- $gr^{\mathbf{p}}_I(F \otimes G) = gr^{\mathbf{p}}_I(F) \otimes gr^{\mathbf{p}}_I(G)$ if $\otimes$ is $\wedge$, $\vee$, or $\rightarrow$;
- $gr^{\mathbf{p}}_I(\exists X\, F(X)) = \{gr^{\mathbf{p}}_I(F(u)) \mid u \in \mathcal{T}\}^{\vee}$ if X is a program variable;
- $gr^{\mathbf{p}}_I(\forall X\, F(X)) = \{gr^{\mathbf{p}}_I(F(u)) \mid u \in \mathcal{T}\}^{\wedge}$ if X is a program variable;
- $gr^{\mathbf{p}}_I(\exists X\, F(X)) = \{gr^{\mathbf{p}}_I(F(u)) \mid u \in \mathcal{Z}\}^{\vee}$ if X is an integer variable;
- $gr^{\mathbf{p}}_I(\forall X\, F(X)) = \{gr^{\mathbf{p}}_I(F(u)) \mid u \in \mathcal{Z}\}^{\wedge}$ if X is an integer variable.

For a theory Γ, we define $gr^{\mathbf{p}}_I(\Gamma) = \{gr^{\mathbf{p}}_I(F) \mid F \in \Gamma\}$.

Definition 1. *Let Γ be a theory and $\mathbf{p}$ be a list of predicate symbols. Then, an interpretation I is called an* INF-$\mathbf{p}$-stable model *of Γ if $I^{\mathbf{p}}$ is a stable model of $gr^{\mathbf{p}}_I(\Gamma)$ in the sense of Definition 1 in (Truszczyński 2012).*

Any term, formula, or theory over the two-sorted signature σ_Π can be seen as one-sorted if we do not assign sorts to variables. On the other hand, some one-sorted terms and formulas cannot be viewed as terms or formulas over σ_Π; for instance, the one-sorted term $X + Y$, where X and Y are program variables, is not a term over σ_Π. We will refer to terms, formulas, and theories over σ_Π as two-sorted. *One-sorted interpretations* are defined as usual in first-order logic, with integer and program variables ranging over the same domain. *One-sorted* $\mathbf{p}$*-stable models* and *one-sorted* INF-$\mathbf{p}$*-stable models* are defined as $\mathbf{p}$-stable models and INF-$\mathbf{p}$-stable models (see Section 4), respectively, but using one-sorted interpretations rather than two-sorted ones. We also say that two theories are *one-sorted-equivalent* if both theories have exactly the same one-sorted models. The following is a special case of Theorem 5 in (Truszczyński 2012) restricted to our one-sorted language.

Proposition 1. *Let Γ be a finite theory and $\mathbf{p}$ be the list of all predicate symbols in some signature σ_Π. Then, the one-sorted $\mathbf{p}$-stable models of Γ and its one-sorted* INF-$\mathbf{p}$*-stable models coincide.*

In the following, we extend this result to two-sorted stable models with extensional predicate symbols (see Proposition 2 below). This requires the following notation and auxiliary results. The expression $is\_int(t)$, where t is a one-sorted term, stands for the one-sorted formula $t + \overline{1} \neq t + \overline{2}$. Given a two-sorted sentence F, we write F^{us} to denote the one-sorted sentence resulting from restricting all quantifiers that bind integer variables in F to $is\_int(t)$. Formally, formula F^{us} is recursively defined as follows:

- $F^{\mathrm{us}} = F$ for any atomic formula F;
- $(F \otimes G)^{\mathrm{us}} = F^{\mathrm{us}} \otimes G^{\mathrm{us}}$ with $\otimes \in \{\wedge, \vee, \to\}$;
- $(\forall X\, F(X))^{\mathrm{us}} = \forall X\, F(X)^{\mathrm{us}}$;
- $(\exists X\, F(X))^{\mathrm{us}} = \exists X\, F(X)^{\mathrm{us}}$;
- $(\forall N\, F(N))^{\mathrm{us}} = \forall N\, (is\_int(N) \to F(N)^{\mathrm{us}})$;
- $(\exists N\, F(N))^{\mathrm{us}} = \exists N\, (is\_int(N) \wedge F(N)^{\mathrm{us}})$;

where X and N are program variables and integer variables, respectively. We also define $\Gamma^{\mathrm{us}} = \{F^{\mathrm{us}} \mid F \in \Gamma\}$.

Intuitively, the one-sorted models of Γ^{us} are in a one-to-one correspondence with the two-sorted models of Γ. This one-to-one correspondence is formalized as follows. The *generalized value* of a ground term is its value if it exists and a fixed (arbitrarily chosen) symbolic constant u otherwise. Given a two-sorted interpretation I, by I^{us}, we denote the one-sorted interpretation such that

- the universe of I^{us} is the set of all precomputed terms;
- I^{us} interprets each ground term as its generalized value;
- I^{us} interprets every predicate symbol in the same way as I.

Lemma 1. *Let F be a two-sorted sentence and I be a two-sorted interpretation. Then, $I \models F$ iff $I^{\mathrm{us}} \models F^{\mathrm{us}}$.*

Proof. By structural induction. In case that F is an atomic formula of the form $p(t_1, \ldots, t_n)$, it follows that $I^{\mathrm{us}} \models F^{\mathrm{us}}$ iff $(s_1^*, \ldots, s_n^*) \in p^I$, where each s_i is the generalized value of t_i, iff $((t_1^I)^*, \ldots, (t_n^I)^*) \in p^I$ iff $I \models F$. Note that, since F is a two-sorted sentence, the generalized value of t_i coincides with its value.

The only remaining relevant cases are quantifiers over integer variables. We show here the case of a universal quantifier. Let N be an integer variable. Then,

$$
\begin{aligned}
&I^{\mathrm{us}} \models (\forall N\, G(N))^{\mathrm{us}} \\
\text{iff } &I^{\mathrm{us}} \models \forall N\, (is\_int(N) \to G(N)^{\mathrm{us}}) \\
\text{iff } &I^{\mathrm{us}} \models is\_int(u) \to G(u)^{\mathrm{us}} \text{ for all } u \in \mathcal{T} \\
\text{iff } &I^{\mathrm{us}} \models G(u)^{\mathrm{us}} \text{ for all } u \in \mathcal{Z} \\
\text{iff } &I \models G(u) \text{ for all } u \in \mathcal{Z} && \text{(induction hypothesis)} \\
\text{iff } &I \models \forall N\, G(N).
\end{aligned}
$$

The case for the existential quantifier is analogous. □

We extend this result also to the stable models of a first-order formula. The following auxiliary result is useful for that purpose.

Lemma 2 (Lemma 5 in Ferraris et al. 2011)**.** *The formula $(\mathbf{u} \leq \mathbf{p}) \wedge (F^*(\mathbf{u}) \to F)$ is satisfied by all one-sorted interpretations and for any one-sorted formula F.*

Lemma 3. *Let F be a two-sorted sentence, I be a two-sorted interpretation, and* $\mathbf{p}$ *be a list of predicate symbols. Then,* $I \models \mathrm{SM}_{\mathbf{p}}[F]$ *iff* $I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{p}}[F^{\mathrm{us}}]$.

Proof. From Lemma 1, we get that $I \models \mathrm{SM}[F]$ iff $I^{\mathrm{us}} \models (\mathrm{SM}[F])^{\mathrm{us}}$. We show below that formula $(\mathbf{u} \leq \mathbf{p}) \wedge (F^*(\mathbf{u}))^{\mathrm{us}}$ is equivalent to $(\mathbf{u} \leq \mathbf{p}) \wedge (F^{\mathrm{us}}(\mathbf{u}))^*$. This immediately implies that $(\mathrm{SM}[F])^{\mathrm{us}}$ and $\mathrm{SM}[F^{\mathrm{us}}]$ are also equivalent. The proof follows by structural induction, and the only relevant cases are, again, quantifiers over integer variables. We show the case of a universal quantifier here. Let N be an integer variable and assume $(\mathbf{u} \leq \mathbf{p})$. Then,

$$\begin{aligned}
&((\forall N\, G(N, \mathbf{u}))^{\mathrm{us}})^* \\
&= (\forall N\, (is_int(N) \to G(N, \mathbf{u})^{\mathrm{us}}))^* \\
&= \forall N\, (is_int(N) \to G(N, \mathbf{u})^{\mathrm{us}})^* \\
&= \forall N\, ((is_int(N) \to G(N)^{\mathrm{us}}) \wedge (is_int(N)^* \to (G(N, \mathbf{u})^{\mathrm{us}})^*)) \\
&= \forall N\, ((is_int(N) \to G(N)^{\mathrm{us}}) \wedge (is_int(N) \to (G(N, \mathbf{u})^{\mathrm{us}})^*)) \\
&\Leftrightarrow \forall N\, (is_int(N) \to (G(N)^{\mathrm{us}} \wedge (G(N, \mathbf{u})^{\mathrm{us}})^*)) \\
&\Leftrightarrow \forall N\, (is_int(N) \to (G(N, \mathbf{u})^{\mathrm{us}})^*) && \text{(Lemma 2)} \\
&\Leftrightarrow \forall N\, (is_int(N) \to (G(N, \mathbf{u})^*)^{\mathrm{us}}) && \text{(induction hypothesis)} \\
&= (\forall N\, G(N, \mathbf{u})^*)^{\mathrm{us}}.
\end{aligned}$$

□

This correspondence can also be established in terms of groundings as follows. The expression $\Gamma \equiv_s \Gamma'$, where Γ and Γ' are two infinitary propositional theories, stands for *strong equivalence* in the sense of (Harrison et al. 2017, Section 3.1).

Lemma 4. *Let Γ be a finite two-sorted theory, I be a two-sorted interpretation, and* $\mathbf{p}$ *be a list of predicate symbols. Then,* $gr_I^{\mathbf{p}}(\Gamma) \equiv_s gr_{I^{\mathrm{us}}}^{\mathbf{p}}(\Gamma^{\mathrm{us}})$.

Proof. We show $gr_I^{\mathbf{p}}(F) \equiv_s gr_{I^{\mathrm{us}}}^{\mathbf{p}}(F^{\mathrm{us}})$ for a formula F, which implies $gr_I^{\mathbf{p}}(\Gamma) \equiv_s gr_{I^{\mathrm{us}}}^{\mathbf{p}}(\Gamma^{\mathrm{us}})$. We proceed by structural induction. The only relevant cases are quantifiers over integer variables. We show the case of a universal quantifier here. Let N be an integer variable and $G'(u)$ stand for $gr_{I^{\mathrm{us}}}^{\mathbf{p}}(G^{\mathrm{us}}(u))$. Then,

$$\begin{aligned}
gr_I^{\mathbf{p}}(\forall N\, G(N)) &= \{gr_I^{\mathbf{p}}(G(u)) \mid u \in \mathcal{Z}\}^{\wedge} \\
&\equiv_s \{G'(u) \mid u \in \mathcal{Z}\}^{\wedge} && \text{(induction hypothesis)} \\
&\equiv_s (\{\top \to G'(u) \mid u \in \mathcal{Z}\} \cup \{\bot \to G'(u) \mid u \in \mathcal{T} \setminus \mathcal{Z}\})^{\wedge} \\
&= gr_{I^{\mathrm{us}}}^{\mathbf{p}}(\forall N\, (is_int(N) \to G^{\mathrm{us}}(N))) \\
&= gr_{I^{\mathrm{us}}}^{\mathbf{p}}((\forall N\, G(N))^{\mathrm{us}}).
\end{aligned}$$

The case for the existential quantifier is analogous. □

Next, we combine these results to establish a correspondence between the stable models of a first-order theory and the stable models of its infinitary grounding.

Lemma 5. *Let Γ be a finite two-sorted theory, I be a two-sorted interpretation, and* $\mathbf{p}$ *be the list of all predicate symbols in the signature. Then, the* $\mathbf{p}$*-stable and the* INF-$\mathbf{p}$*-stable models of Γ coincide.*

Proof. We have

$$\begin{aligned}
& I \text{ is a } \mathbf{p}\text{-stable model of } \Gamma \\
\text{iff } & I^{\mathbf{us}} \text{ is a } \mathbf{p}\text{-stable model of } \Gamma^{\mathbf{us}} && \text{(Lemma 3)} \\
\text{iff } & I^{\mathbf{us}} \text{ is an INF-}\mathbf{p}\text{-stable model of } \Gamma^{\mathbf{us}} && \text{(Proposition 1)} \\
\text{iff } & (I^{\mathbf{us}})^{\mathbf{p}} \text{ is a stable model of } gr^{\mathbf{p}}_{I^{\mathbf{us}}}(\Gamma^{\mathbf{us}}) \\
\text{iff } & (I^{\mathbf{us}})^{\mathbf{p}} \text{ is a stable model of } gr^{\mathbf{p}}_{I}(\Gamma) && \text{(Lemma 4)} \\
\text{iff } & I^{\mathbf{p}} \text{ is a stable model of } gr^{\mathbf{p}}_{I}(\Gamma) \\
\text{iff } & I \text{ is an INF-}\mathbf{p}\text{-stable model of } \Gamma.
\end{aligned}$$

For the next-to-last equivalence, just note that $(I^{\mathbf{us}})^{\mathbf{p}} = I^{\mathbf{p}}$. □

Proposition 2. *For any finite two-sorted theory* Γ *and list of predicate symbols* $\mathbf{p}$*, its* $\mathbf{p}$*-stable models and its* INF-$\mathbf{p}$*-stable models coincide.*

Proof. Let $\mathbf{q}$ be the list of all extensional predicate symbols in Γ, that is, all predicate symbols in the signature that do not belong to $\mathbf{p}$, and let $\mathit{Choice}(\mathbf{q})$ be the set containing a choice sentence $\forall \mathbf{U}\,(p(\mathbf{U}) \vee \neg p(\mathbf{U}))$ for every predicate $p \in \mathbf{q}$ and where $\mathbf{U}$ is a list of distinct program variables. Let Γ' be the theory obtained by replacing each occurrence of $p(\mathbf{t})$ in Γ with $p \in \mathbf{q}$ by $\neg\neg p(\mathbf{t})$. Let $\Gamma_1 = \Gamma \cup \mathit{Choice}(\mathbf{q})$ and $\Gamma'_1 = \Gamma' \cup \mathit{Choice}(\mathbf{q})$. Given the choice sentences in Γ_1 and Γ'_1 for the predicate symbols in $\mathbf{q}$, the $\mathbf{pq}$-stable models of Γ_1 and Γ'_1 coincide. Then,

$$\begin{aligned}
& I \text{ is a } \mathbf{p}\text{-stable model of } \Gamma \\
\text{iff } & I \text{ is a } \mathbf{pq}\text{-stable model of } \Gamma_1 && \text{(Theorem 2 in Ferraris et al. 2011)} \\
\text{iff } & I \text{ is a } \mathbf{pq}\text{-stable model of } \Gamma'_1 \\
\text{iff } & I \text{ is an INF-}\mathbf{pq}\text{-stable model of } \Gamma'_1 && \text{(Lemma 5)} \\
\text{iff } & I \text{ is an INF-}\mathbf{p}\text{-stable model of } \Gamma. && \text{(see below)}
\end{aligned}$$

It remains to be shown that the INF-$\mathbf{pq}$-stable models of Γ'_1 coincide with the INF-$\mathbf{p}$-stable models of Γ. For this, note that

$$\begin{aligned}
[gr^{\mathbf{pq}}_I(\Gamma'_1)]^{I^{\mathbf{pq}}} &= [gr^{\mathbf{pq}}_I(\Gamma' \cup \mathit{Choice}(\mathbf{q}))]^{I^{\mathbf{pq}}} \\
&= [gr^{\mathbf{pq}}_I(\Gamma')]^{I^{\mathbf{pq}}} \cup [gr^{\mathbf{pq}}_I(\mathit{Choice}(\mathbf{q}))]^{I^{\mathbf{pq}}} \\
&\equiv [gr^{\mathbf{p}}_I(\Gamma')]^{I^{\mathbf{pq}}} \cup [gr^{\mathbf{pq}}_I(\mathit{Choice}(\mathbf{q}))]^{I^{\mathbf{pq}}} \\
&= [gr^{\mathbf{p}}_I(\Gamma')]^{I^{\mathbf{p}}} \cup [gr^{\mathbf{pq}}_I(\mathit{Choice}(\mathbf{q}))]^{I^{\mathbf{q}}} \\
&\equiv [gr^{\mathbf{p}}_I(\Gamma')]^{I^{\mathbf{p}}} \cup I^{\mathbf{q}}.
\end{aligned}$$

The first two equalities hold by definition. The third step holds because all predicate symbols in $\mathbf{q}$ occur in Γ' under the scope of negation. Note that, for $q \in \mathbf{q}$, it follows that

$$\begin{aligned}
[gr^{\mathbf{pq}}_I(\neg q(\mathbf{t}))]^{I^{\mathbf{pq}}} &= [\neg q((\mathbf{t}^I)^*)]^{I^{\mathbf{pq}}} = I^{\mathbf{pq}}(\neg q((\mathbf{t}^I)^*)) \\
&\equiv \neg I(q((\mathbf{t}^I)^*)) \\
&= [\neg I(q((\mathbf{t}^I)^*))]^{I^{\mathbf{pq}}} = [\neg gr^{\mathbf{p}}_I(q(\mathbf{t}))]^{I^{\mathbf{pq}}} = [gr^{\mathbf{p}}_I(\neg q(\mathbf{t}))]^{I^{\mathbf{pq}}},
\end{aligned}$$

where

$$I(q((\mathbf{t}^I)^*)) = \begin{cases} \top & \text{if } q((\mathbf{t}^I)^*) \in I^{\mathbf{q}}; \\ \bot & \text{otherwise} \end{cases}$$

$$I^{\mathbf{pq}}(\neg q((\mathbf{t}^I)^*)) = \begin{cases} \bot \equiv \neg I(q((\mathbf{t}^I)^*)) & \text{if } q((\mathbf{t}^I)^*) \in I^{\mathbf{q}}; \\ \neg\bot = \neg I(q((\mathbf{t}^I)^*)) & \text{otherwise.} \end{cases}$$

The fourth case is because no predicate in $\mathbf{q}$ occurs in $gr_I^{\mathbf{p}}(\Gamma')$. Recall that extensional predicate symbols are removed by grounding. Similarly, *Choice*$(\mathbf{q})$ only contains predicate symbols from $\mathbf{q}$.

We now prove the following equivalence:

$$[gr_I^{\mathbf{pq}}(\Gamma_1')]^{I^{\mathbf{pq}}} \equiv [gr_I^{\mathbf{p}}(\Gamma)]^{I^{\mathbf{p}}} \cup I^{\mathbf{q}} \tag{A1}$$

For this, note that

$$gr_I^{\mathbf{p}}(\Gamma') \equiv gr_I^{\mathbf{p}}(\Gamma)$$

holds because $gr_I^{\mathbf{p}}(\Gamma')$ is the result of replacing each occurrence of $p(\mathbf{t})$ in $gr_I^{\mathbf{p}}(\Gamma)$ by $\neg\neg p(\mathbf{t})$ with $p \in \mathbf{q}$. As a result, $gr_I^{\mathbf{p}}(\Gamma')$ is the outcome of replacing each occurrence of X in $gr_I^{\mathbf{p}}(\Gamma)$ (with $X \in \{\top, \bot\}$) by $\neg\neg X$. Consequently, equivalence (A1) is proven, and we get that any interpretation J satisfies that $J^{\mathbf{pq}} \models [gr_I^{\mathbf{pq}}(\Gamma_1')]^{I^{\mathbf{pq}}}$ iff $J^{\mathbf{p}} \models [gr_I^{\mathbf{p}}(\Gamma)]^{I^{\mathbf{p}}}$ and $J^{\mathbf{q}} \models I^{\mathbf{q}}$. Note that $J^{\mathbf{q}} \models I^{\mathbf{q}}$ iff $J^{\mathbf{q}} \supseteq I^{\mathbf{q}}$. Then,

I is an INF-**pq**-stable model of Γ_1'
iff $I^{\mathbf{pq}}$ is a stable model of $gr_I^{\mathbf{pq}}(\Gamma_1')$
iff $I^{\mathbf{pq}}$ is a model of $[gr_I^{\mathbf{pq}}(\Gamma_1')]^{I^{\mathbf{pq}}}$ and there is no model $J \subset I^{\mathbf{pq}}$ of $[gr_I^{\mathbf{pq}}(\Gamma_1')]^{I^{\mathbf{pq}}}$
iff $I^{\mathbf{p}}$ is a model of $[gr_I^{\mathbf{p}}(\Gamma)]^{I^{\mathbf{p}}}$ and there is no model $J \subset I^{\mathbf{pq}}$ of $[gr_I^{\mathbf{p}}(\Gamma)]^{I^{\mathbf{p}}} \cup I^{\mathbf{q}}$
iff $I^{\mathbf{p}}$ is a model of $[gr_I^{\mathbf{p}}(\Gamma)]^{I^{\mathbf{p}}}$ and there is no model $J' \subset I^{\mathbf{p}}$ of $[gr_I^{\mathbf{p}}(\Gamma)]^{I^{\mathbf{p}}}$
iff $I^{\mathbf{p}}$ is a stable model of $[gr_I^{\mathbf{p}}(\Gamma)]$
iff I is an INF-**p**-stable model of Γ. □

The following adaptation of Proposition 3 from (Lifschitz et al. 2019) to our notation is useful to prove the Main Lemma.

Proposition 3. *Any rule R and interpretation I satisfy $gr_I^{\mathbf{p}}(\tau^* R) \equiv_s \tau R$.*

Proof. By identifying the precomputed terms in $\tau^*\Pi^{\mathbf{prop}}$ with their names in I, we get $gr_I^{\mathbf{p}}(\tau^*\Pi) = \tau^*\Pi^{\mathbf{prop}}$, where $\tau^*\Pi^{\mathbf{prop}}$ is defined as in (Lifschitz et al. 2019, Section 5). □

Proof of the Main Lemma. Let Π be a program, let $\mathbf{p}$ be the list of all predicate symbols occurring in Π other than the comparison symbols, and let $\mathcal{I}$ be a set of precomputed atoms. By the choice of $\mathbf{p}$, we get that all predicate symbols in Π and none of the relations belong to $\mathbf{p}$ and, therefore, $\mathcal{I} = (\mathcal{I}^{\uparrow})^{\mathbf{p}}$. Then, from Proposition 2, it follows that $\mathcal{I}^{\uparrow}$ is a $\mathbf{p}$-stable model of $\tau^*\Pi$ iff $\mathcal{I}^{\uparrow}$ is an INF-$\mathbf{p}$-stable model of $\tau^*\Pi$ iff $\mathcal{I}$ is a $\subseteq$-minimal model of $[gr_{\mathcal{I}^{\uparrow}}^{\mathbf{p}}(\tau^*\Pi)]^{\mathcal{I}}$ iff $\mathcal{I}$ is a stable model of $[gr_{\mathcal{I}^{\uparrow}}^{\mathbf{p}}(\tau^*\Pi)]$ iff $\mathcal{I}$ is a stable model of $\tau\Pi$ (Proposition 3) iff $\mathcal{I}$ is a stable model of Π (by definition). □

A.2 Main Lemma for IO-Programs

We need the following terminology to extend the Main Lemma to io-programs. The models of formula $\exists \mathbf{H}\,(\mathrm{SM}_{\mathbf{p}}[F])^{\mathbf{h}}_{\mathbf{H}}$ are called the **p**-*stable models with private symbols* **h** of F, where **H** is a tuple of predicate variables of the same length as **h** and $F^{\mathbf{h}}_{\mathbf{H}}$ is the result of replacing all occurrences of constants from **h** by the corresponding variables from **H**. For a set Γ of first-order sentences, the **p**-*stable models with private symbols* **h** of Γ are the **p**-stable models with private symbols **h** of the conjunction of all formulas in Γ (Cabalar et al. 2020). We usually omit parentheses and write just $\exists \mathbf{H}\,\mathrm{SM}_{\mathbf{p}}[F]^{\mathbf{h}}_{\mathbf{H}}$ instead of $\exists \mathbf{H}\,(\mathrm{SM}_{\mathbf{p}}[F])^{\mathbf{h}}_{\mathbf{H}}$.

Main Lemma for IO-Programs. *Let $\Omega = (\Pi, PH, In, Out)$ be an io-program, let* **p** *be the list of all predicate symbols occurring in Π other than the comparison and input symbols, and let* **h** *be the list of all its private symbols. A set $\mathcal{I}$ of precomputed public atoms is an io-model of Ω for an input $(\mathbf{v}, \mathbf{i})$ iff $\mathcal{I}^{\mathbf{v}}$ is a* **p**-*stable model with private symbols* **h** *of $\tau^*\Pi$ and $\mathcal{I}^{in} = \mathbf{i}$.*

The following is a reformulation of the Splitting Theorem in (Ferraris et al. 2009) adapted to our notation, and it will be useful in proving the above result. We adopt the following terminology. An occurrence of a predicate symbol in a formula is called *negated* if it belongs to a subformula of the form $F \to \bot$ and *nonnegated* otherwise. An occurrence of a predicate symbol in a formula is called *positive* if the number of implications containing that occurrence in the antecedent is even. It is called *strictly positive* if that number is 0. A *rule* of a first-order formula F is a strictly positive occurrence of an implication in F. The *dependency graph* of a formula is a directed graph that

- has all intensional predicate symbols as vertices and
- has an edge from p to q if, for some rule $G \to H$ of F, formula G has a positive nonnegated occurrence of q and H has a strictly positive occurrence of p.

Proposition 4. *Let F and G be one-sorted first-order sentences and let* **p** *and* **q** *be two disjoint tuples of distinct predicate symbols such that*

- *each strongly connected component of the of the dependency graph of $F \wedge G$ is a subset either of* **p** *or* **q**,
- *all occurrences in F of symbols from* **q** *are negated, and*
- *all occurrences in G of symbols from* **p** *are negated.*

Then, $\mathrm{SM}_{\mathbf{pq}}[F \wedge G]$ *is equivalent to* $\mathrm{SM}_{\mathbf{p}}[F] \wedge \mathrm{SM}_{\mathbf{q}}[G]$.

This result can be straightforwardly lifted to the two-sorted language as follows.

Proposition 5. *Let F and G be two-sorted first-order sentences and let* **p** *and* **q** *be two disjoint tuples of distinct predicate symbols such that*

- *each strongly connected component of the dependency graph of $F \wedge G$ is a subset either of* **p** *or* **q**,
- *all occurrences in F of symbols from* **q** *are negated, and*
- *all occurrences in G of symbols from* **p** *are negated.*

Then, $\mathrm{SM}_{\mathbf{pq}}[F \wedge G]$ *is equivalent to* $\mathrm{SM}_{\mathbf{p}}[F] \wedge \mathrm{SM}_{\mathbf{q}}[G]$.

Proof. Let I be any interpretation. Then,

$$
\begin{aligned}
& I \models \mathrm{SM}_{\mathbf{pq}}[F \wedge G] & \\
\text{iff } & I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{pq}}[(F \wedge G)^{\mathrm{us}}] & \text{(Lemma 3)} \\
\text{iff } & I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{pq}}[F^{\mathrm{us}} \wedge G^{\mathrm{us}}] & \\
\text{iff } & I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{p}}[F^{\mathrm{us}}] \wedge \mathrm{SM}_{\mathbf{q}}[G^{\mathrm{us}}] & \text{(Proposition 4)} \\
\text{iff } & I \models \mathrm{SM}_{\mathbf{p}}[F] \wedge \mathrm{SM}_{\mathbf{q}}[G]. & \text{(Lemma 3)}
\end{aligned}
$$

Note that the dependency graphs of $F \wedge G$ and $F^{\mathrm{us}} \wedge G^{\mathrm{us}}$ are the same. □

Lemma 6. *Let* $\Omega = (\Pi, PH, In, Out)$ *be an io-program and let* $\mathbf{p}$ *be the list of all predicate symbols occurring in* Π *other than the comparison and input symbols. A set* $\mathcal{I}$ *of precomputed public atoms is a stable model of* $\Omega(\mathbf{v}, \mathbf{i})$ *iff* $\mathcal{I}^{\mathbf{v}}$ *is a model of* $\mathrm{SM}_{\mathbf{p}}[\tau^*\Pi]$ *and* $\mathcal{I}^{in} = \mathbf{i}$.

Proof. Recall that, from the Main Lemma stated in Section 4, we get that $\mathcal{I}$ is a stable model of $\Omega(\mathbf{v}, \mathbf{i})$ iff $\mathcal{I}^{\uparrow}$ is a $\mathbf{pq}$-stable model of $\tau^*(\Omega(\mathbf{v}, \mathbf{i}))$ iff $\mathcal{I}^{\uparrow}$ is a model of $\mathrm{SM}_{\mathbf{pq}}[\tau^*(\Omega(\mathbf{v}, \mathbf{i}))]$, where $\mathbf{q}$ is the list of all input symbols. Let us denote by $\Omega(\mathbf{v})$ the set of rules obtained from the rules of Ω by substituting the precomputed terms $v(c)$ for all occurrences of all placeholders c. Then, $\tau^*(\Omega(\mathbf{v}, \mathbf{i})) = \tau^*(\Omega(\mathbf{v}) \cup \mathbf{i}) = \tau^*(\Omega(\mathbf{v})) \cup \tau^*(\mathbf{i}) \equiv_s \tau^*(\Omega(\mathbf{v})) \cup \mathbf{i}$. Furthermore, since there are no occurrences of predicate symbols in $\mathbf{q}$ in the heads of the rules of $\Omega(\mathbf{v})$ nor of any predicate symbol in $\mathbf{p}$ in the head of the rules in $\mathbf{i}$, we get that each strongly connected component is a subset either of $\mathbf{p}$ or $\mathbf{q}$. From Proposition 5, this implies that

$$
\begin{aligned}
& \mathcal{I}^{\uparrow} \text{ is a model of } \mathrm{SM}_{\mathbf{pq}}[\tau^*(\Omega(\mathbf{v}, \mathbf{i}))] \\
\text{iff } & \mathcal{I}^{\uparrow} \text{ is a model of } \mathrm{SM}_{\mathbf{p}}[\tau^*(\Omega(\mathbf{v}))] \text{ and } \mathcal{I}^{\uparrow} \text{ is a model of } \mathrm{SM}_{\mathbf{q}}[\mathbf{i}] \\
\text{iff } & \mathcal{I}^{\mathbf{v}} \text{ is a model of } \mathrm{SM}_{\mathbf{p}}[\tau^*\Pi] \text{ and } \mathcal{I}^{in} = \mathbf{i}.
\end{aligned}
$$

For the second equivalence, note that $\mathcal{I}^{\mathbf{v}}$ is identical to $\mathcal{J}^{\uparrow}$, except that it interprets each placeholder c as $\mathbf{v}(c)$ and that $\Omega(\mathbf{v})$ is the result of replacing each placeholder c by $\mathbf{v}(c)$. □

Proof of the Main Lemma for IO-Programs. From left to right. Assume that $\mathcal{I}$ is an io-model of Ω for input $(\mathbf{v}, \mathbf{i})$. Let us show that $\mathcal{I}^{\mathbf{v}}$ is a $\mathbf{p}$-stable model with private symbols $\mathbf{h}$ of $\tau^*\Pi$ and $\mathcal{I}^{in} = \mathbf{i}$. By definition, the assumption implies that there is some stable model $\mathcal{J}$ of $\Omega(\mathbf{v}, \mathbf{i})$ such that $\mathcal{I}$ is the set of all public atoms of $\mathcal{J}$. From Lemma 6, this implies that $\mathcal{J}^{\mathbf{v}}$ is a model of $\mathrm{SM}_{\mathbf{p}}[\tau^*\Pi]$ and $\mathcal{J}^{in} = \mathbf{i}$ and, thus, that $\mathcal{I}^{\mathbf{v}}$ is a model of $\exists \mathbf{H}\, \mathrm{SM}_{\mathbf{p}}[(\tau^*\Pi)]^{\mathbf{h}}_{\mathbf{H}}$ and $\mathcal{I}^{in} = \mathbf{i}$. For this last step, recall that $\mathcal{I}$ and $\mathcal{J}$ agree on all public predicates. By definition, this means that $\mathcal{I}^{\mathbf{v}}$ is a $\mathbf{p}$-stable model with private symbols $\mathbf{h}$ of $\tau^*\Pi$ and $\mathcal{I}^{in} = \mathbf{i}$.

From right to left. Assume that $\mathcal{I}^{\mathbf{v}}$ is a $\mathbf{p}$-stable model with private symbols $\mathbf{h}$ of $\tau^*\Pi$. Let us show that $\mathcal{I}$ is an io-model of Ω for an input $(\mathbf{v}, \mathbf{i})$. By definition, the assumption implies that $\mathcal{I}^{\mathbf{v}}$ is a model of $\exists \mathbf{H}\, \mathrm{SM}_{\mathbf{p}}[(\tau^*\Pi)]^{\mathbf{h}}_{\mathbf{H}}$. This implies that there is some model J of $\mathrm{SM}_{\mathbf{p}}[(\tau^*\Pi)]$ such that $\mathcal{I}^{\mathbf{v}}$ and J agree on the interpretation of all public predicates. Let $\mathcal{J}$ be the set of precomputed atoms satisfied by J. Then, there are no occurrences of placeholders in $\mathcal{J}$ and, thus, we get that $\mathcal{J}^{\mathbf{v}} = \mathcal{J}^{\uparrow} = J$ and, thus, also that $\mathcal{J}^{\mathbf{v}}$ is a stable model of $\mathrm{SM}_{\mathbf{p}}[(\tau^*\Pi)]$. Recall that we also have $\mathcal{I}^{in} = \mathbf{i}$ and, since $\mathcal{I}$ and $\mathcal{J}$ contain the same public atoms, we get that $\mathcal{J}^{in} = \mathbf{i}$. From Lemma 6, these two facts together imply that $\mathcal{J}$ is a stable model of $\Omega(\mathbf{v}, \mathbf{i})$ and, therefore, that $\mathcal{I}$ is an io-model of Ω for an input $(\mathbf{v}, \mathbf{i})$. □

A.3 Theorem 1

In order to prove Theorem 1, we need the notions of Clark normal form, completion, and tight theories (Ferraris et al. 2011, Section 6). We adapt these notions to a two-sorted language here. A theory—one-sorted or two-sorted—is in *Clark normal form* relative to a list $\mathbf{p}$ of intensional predicates if it contains exactly one sentence of the form

$$\forall V_1 \dots V_n \, (G \to p(V_1, \dots, V_n)) \tag{A2}$$

for each intensional predicate symbol p/n in $\mathbf{p}$, where G is a formula and $V_1, \dots, V_n$ are distinct program variables. The *completion* of a theory Γ in Clark normal form, denoted by $\mathrm{COMP}_{\mathbf{p}}[\Gamma]$, is obtained by replacing each implication $\to$ by an equivalence $\leftrightarrow$ in all sentences of form (A2). The following is a special case of Theorem 10 in (Ferraris et al. 2011) adapted to our notation.

Proposition 6. *For any one-sorted sentence F in Clark normal form and list of predicates $\mathbf{p}$, the implication*

$$\mathrm{SM}_{\mathbf{p}}[F] \to \mathrm{COMP}_{\mathbf{p}}[F]$$

is satisfied by all one-sorted interpretations.

Then, we can easily extend this result to two-sorted interpretations as follows.

Proposition 7. *For any two-sorted sentence F in Clark normal form, list of predicates $\mathbf{p}$, and two-sorted interpretation $\mathcal{I}$, if $\mathcal{I}$ satisfies $\mathrm{SM}_{\mathbf{p}}[F]$, then it also satisfies $\mathrm{COMP}_{\mathbf{p}}[F]$.*

Proof. Let I be any two-sorted interpretation. From Lemma 1, we get that $I \models \mathrm{COMP}_{\mathbf{p}}[F]$ iff $I^{\mathrm{us}} \models (\mathrm{COMP}_{\mathbf{p}}[F])^{\mathrm{us}}$. Furthermore, we can see that $(\mathrm{COMP}_{\mathbf{p}}[F])^{\mathrm{us}} = \mathrm{COMP}_{\mathbf{p}}[F^{\mathrm{us}}]$, and thus, we get

$$I \models \mathrm{COMP}_{\mathbf{p}}[F] \text{ iff } I^{\mathrm{us}} \models \mathrm{COMP}_{\mathbf{p}}[F^{\mathrm{us}}].$$

Similarly, from Lemma 3, we get

$$I \models \mathrm{SM}_{\mathbf{p}}[\Gamma] \text{ iff } I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{p}}[\Gamma^{\mathrm{us}}].$$

Finally, from Proposition 6, we get

$$I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{p}}[\Gamma^{\mathrm{us}}] \text{ implies } I^{\mathrm{us}} \models \mathrm{COMP}_{\mathbf{p}}[\Gamma^{\mathrm{us}}].$$

Consequently, the result holds. □

Let us introduce the *Clark form* of a program without input and output. The *Clark definition* of p/n in Π is a formula of the form

$$\forall V_1 \dots V_n \left(\bigvee_{i=1}^{k} \exists \mathbf{U}_i \, F_i \to p(V_1, \dots, V_n) \right), \tag{A3}$$

where each F_i is the formula representation of rule R_i and rules $R_1, \dots, R_k$ constitute the definition of p/n in Π. By $\mathit{Cdef}(\Pi)$, we denote the theory containing the Clark definitions of all predicate symbols. We also define $\mathit{Clark}(\Pi) \stackrel{\text{def}}{=} \mathit{Cdef}(\Pi) \cup \Pi_C$, where Π_C is the set containing the formula representation of all constraints in Π.

About first-order formulas F and G, it is said that F is *strongly equivalent* to G if, for any formula H, any occurrence of F in H, and any list $\mathbf{p}$ of distinct predicate symbols, $\mathrm{SM}_{\mathbf{p}}[H]$ is equivalent to $\mathrm{SM}_{\mathbf{p}}[H']$, where H' is obtained from H by replacing the occurrence of F by G. About finite first-order theories Γ and Γ', we say that Γ is *strongly equivalent* to Γ' when the conjunction of all sentences in Γ is strongly equivalent to the conjunction of all sentences in Γ'. First-order theory Γ is strongly equivalent to Γ' iff Γ is equivalent to Γ' in quantified equilibrium logic (Ferraris et al. 2011, Theorem 8). Therefore, Γ is strongly equivalent to Γ' if Γ is equivalent to Γ' in intuitionistic logic.

Lemma 7. *Let* Π *be a program without constraints. Then,* $\tau^*\Pi$ *is strongly equivalent to* $\mathit{Cdef}(\Pi)$.

Proof. By definition, $\tau^*\Pi$ contains a formula of the form

$$\forall V_1 \dots V_n \mathbf{U}_i \,(F_i \to p(V_1, \dots, V_n)) \tag{A4}$$

for each rule R_i in Π. Note that (A4) is strongly equivalent to

$$\forall V_1 \dots V_n \,(\exists \mathbf{U}_i \, F_i \to p(V_1, \dots, V_n)).$$

Furthermore, since Π is finite, it follows that $\tau^*\Pi$ is finite too and, therefore, we get that $\tau^*\Pi$ is strongly equivalent to Γ, where Γ is the theory containing a formula of the form

$$\bigwedge_{i=1}^{k} \forall V_1 \dots V_n \,(\exists \mathbf{U}_i \, F_i \to p(V_1, \dots, V_n)) \tag{A5}$$

for each predicate symbol p/n. Finally, since (A3) and (A5) are strongly equivalent, we get that $\tau^*\Pi$ and $\mathit{Cdef}(\Pi)$ are also strongly equivalent. □

Lemma 8. *For any program* Π, $\tau^*\Pi$ *is strongly equivalent to* $\mathit{Clark}(\Pi)$.

Proof. Let Π_1, Π_2 be a partition of Π such that Π_2 contains all constraints and Π_1 all the remaining rules. Then, $\tau^*\Pi = \tau^*(\Pi_1 \cup \Pi_2) = \tau^*\Pi_1 \cup \tau^*\Pi_2 = \tau^*\Pi_1 \cup \Pi_C$. Now, the result follows directly from Lemma 7. □

Proof of Theorem 1. From the Main Lemma for IO-Programs, it follows that $\mathcal{I}$ is an io-model of Ω for an input $(\mathbf{v}, \mathbf{i})$ iff $\mathcal{I}^{\mathbf{v}}$ is a $\mathbf{p}$-stable model with private symbols $\mathbf{h}$ of $\tau^*\Pi$ and $\mathcal{I}^{in} = \mathbf{i}$. Furthermore,

$$\begin{aligned}
&\mathcal{I}^{\mathbf{v}} \text{ is a } \mathbf{p}\text{-stable model with private symbols } \mathbf{h} \text{ of } \tau^*\Pi \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{SM}_{\mathbf{p}}[\tau^*\Pi]^{\mathbf{h}}_{\mathbf{H}} \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{SM}_{\mathbf{p}}[\mathit{Clark}(\Pi)]^{\mathbf{h}}_{\mathbf{H}} && \text{(Lemma 8)} \\
\text{implies that } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{COMP}_{\mathbf{p}}[\mathit{Clark}(\Pi)]^{\mathbf{h}}_{\mathbf{H}} && \text{(Proposition 7)} \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{COMP}_{\mathbf{p}}[\tau^*\Pi]^{\mathbf{h}}_{\mathbf{H}} \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \mathrm{COMP}[\Omega].
\end{aligned}$$

□

A.4 Theorem 2

To prove Theorem 2, we need the following terminology. An occurrence of a predicate symbol in a formula is called *negated* if it belongs to a subformula of the form $F \to \bot$ and *nonnegated* otherwise. An occurrence of a predicate symbol in a formula is called *positive* if the number of implications containing that occurrence in the antecedent is even. The *dependency graph* of a theory in Clark normal form is a directed graph that

- has all intensional predicate symbols as vertices and
- has an edge from p to q if q has a positive nonnegated occurrence in G for some sentence of form (A2).

A theory is *tight* if its predicate dependency graph is acyclic. The following is a reformulation of Theorem 11 in (Ferraris et al. 2011) adapted to our notation.

Proposition 8. *For any finite, tight, one-sorted theory* Γ *in Clark normal form,* $\mathrm{SM}_{\mathbf{p}}[\Gamma]$ *is equivalent to* $\mathrm{COMP}_{\mathbf{p}}[\Gamma]$.

The following lifts this result to the case of two sorts.

Proposition 9. *For any finite, tight, two-sorted theory* Γ *in Clark normal form,* $\mathrm{SM}_{\mathbf{p}}[\Gamma]$ *is equivalent to* $\mathrm{COMP}_{\mathbf{p}}[\Gamma]$.

Proof. Let I be any two-sorted interpretation. From Lemma 1, we get that $I \models \mathrm{COMP}_{\mathbf{p}}[\Gamma]$ iff $I^{\mathrm{us}} \models (\mathrm{COMP}_{\mathbf{p}}[\Gamma])^{\mathrm{us}}$. Furthermore, it is easy to see that $(\mathrm{COMP}_{\mathbf{p}}[\Gamma])^{\mathrm{us}} = \mathrm{COMP}_{\mathbf{p}}[\Gamma^{\mathrm{us}}]$, and thus, we get

$$I \models \mathrm{COMP}_{\mathbf{p}}[\Gamma] \text{ iff } I^{\mathrm{us}} \models \mathrm{COMP}_{\mathbf{p}}[\Gamma^{\mathrm{us}}].$$

Similarly, from Lemma 3, we get

$$I \models \mathrm{SM}_{\mathbf{p}}[\Gamma] \text{ iff } I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{p}}[\Gamma^{\mathrm{us}}].$$

Finally, from Proposition 8, we get

$$I^{\mathrm{us}} \models \mathrm{COMP}_{\mathbf{p}}[\Gamma^{\mathrm{us}}] \text{ iff } I^{\mathrm{us}} \models \mathrm{SM}_{\mathbf{p}}[\Gamma^{\mathrm{us}}].$$

Consequently, the result holds. □

Proof of Theorem 2. From the Main Lemma for IO-Programs, it follows that $\mathcal{I}$ is an io-model of Ω for an input $(\mathbf{v}, \mathbf{i})$ iff $\mathcal{I}^{\mathbf{v}}$ is a $\mathbf{p}$-stable model with private symbols $\mathbf{h}$ of $\tau^*\Pi$ and $\mathcal{I}^{in} = \mathbf{i}$. Furthermore,

$$\begin{aligned}
&\mathcal{I}^{\mathbf{v}} \text{ is a } \mathbf{p}\text{-stable model with private symbols } \mathbf{h} \text{ of } \tau^*\Pi \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{SM}_{\mathbf{p}}[\tau^*\Pi]^{\mathbf{h}}_{\mathbf{H}} \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{SM}_{\mathbf{p}}[\mathit{Clark}(\Pi)] && \text{(Lemma 7)} \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{COMP}_{\mathbf{p}}[\mathit{Clark}(\Pi)]^{\mathbf{h}}_{\mathbf{H}} && \text{(Proposition 9)} \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \exists \mathbf{H}\, \mathrm{COMP}_{\mathbf{p}}[\tau^*\Pi]^{\mathbf{h}}_{\mathbf{H}} \\
\text{iff } &\mathcal{I}^{\mathbf{v}} \text{ is a model of } \mathrm{COMP}[\Omega].
\end{aligned}$$

Recall that Ω is tight, and this implies that $\mathit{Clark}(\tau^*\Pi)$ is also tight. Note that $\mathit{Clark}(\tau^*\Pi)$ contains a formula of form (A3) for every predicate symbol p/n and that the antecedent of this formula is a disjunction of the formula representations of the bodies of all rules defining p/n. Therefore, the dependency graph of $\mathit{Clark}(\tau^*\Pi)$ is identical to the dependency graph of Ω with the exception of the addition of nonpositive edges corresponding to choice rules. □

A.5 Theorem 3

A *predicate expression* is a lambda expression of the form

$$\lambda X_1 \dots X_n \, F(X_1, \dots, X_n), \tag{A6}$$

where $F(X_1, \dots, X_n)$ is a formula and $X_1, \dots, X_n$ are object variables. This formula may have free variables other than $X_1, \dots, X_n$, called the *parameters* of (A6). If E is (A6) and $t_1, \dots, t_n$ are terms, then $E(t_1, \dots, t_n)$ stands for the formula $F(t_1, \dots, t_n)$. If $G(P)$ is a formula containing a predicate constant or variable P and E is a predicate expression of the same arity as P, then $G(E)$ stands for the result of replacing each atom $P(t_1, \dots, t_n)$ in $G(P)$ by $E(t_1, \dots, t_n)$. For any predicate expression E, the formulas

$$\forall P \, G(P) \rightarrow A(E) \quad \text{and} \quad G(E) \rightarrow \exists P \, A(P)$$

are theorems of second-order logic.

Lemma 9. *Let* $\mathbf{P} = P_1, \dots, P_l$ *be a list of predicate variables and let* $\mathbf{P}_i = P_1, \dots, P_i$ *be a prefix of* $\mathbf{P}$. *Let* $F_1(\mathbf{P}_1), \dots, F_l(\mathbf{P}_l)$ *be formulas such that* $\mathbf{P}_i$ *contains all free predicate variables occurring in* F_i. *Let* F *and* G*, respectively, be the following two formulas:*

$$\exists \mathbf{P} \, (F_1(\mathbf{P}_1) \wedge \dots \wedge F_l(\mathbf{P}_l) \wedge F'(\mathbf{P})), \tag{A7}$$

$$\forall \mathbf{P} \, (F_1(\mathbf{P}_1) \wedge \dots \wedge F_l(\mathbf{P}_l) \rightarrow F'(\mathbf{P})). \tag{A8}$$

Then, $F \equiv G$.

Proof. If $l = 0$, then $\mathbf{P}$ is the empty tuple, and thus, both (A7) and (A8) stand just for the formula $F'(\mathbf{P})$, so the result holds. Otherwise, we proceed by induction. Note that (A7) and (A8) are, respectively, equivalent to

$$\exists P_1 \, (F_1(P_1) \wedge \exists \mathbf{P}'_l \, (F_2(P_1, \mathbf{P}'_2) \wedge \dots \wedge F_l(P_1, \mathbf{P}'_l) \wedge F'(P_1, \mathbf{P}'_l))), \tag{A9}$$

$$\forall P_1 \, (F_1(P_1) \rightarrow \forall \mathbf{P}'_l \, (F_2(P_1, \mathbf{P}'_2) \wedge \dots \wedge F_l(P_1, \mathbf{P}'_l) \rightarrow F'(P_1, \mathbf{P}'_l))). \tag{A10}$$

where $\mathbf{P}'_i = P_2, \dots, P_i$. That is, $\mathbf{P}_i = P_1, \mathbf{P}'_i$. Then, by induction hypothesis, we get that the following two formulas are equivalent:

$$\exists \mathbf{P}'_l \, (F_2(P_1, \mathbf{P}'_2) \wedge \dots \wedge F_l(P_1, \mathbf{P}'_l) \wedge F'(P_1, \mathbf{P}'_l)),$$
$$\forall \mathbf{P}'_l \, (F_2(P_1, \mathbf{P}'_2) \wedge \dots \wedge F_l(P_1, \mathbf{P}'_l) \rightarrow F'(P_1, \mathbf{P}'_l)).$$

Therefore, (A10) is equivalent to

$$\forall P_1 \, (F_1(P_1) \rightarrow \exists \mathbf{P}'_l \, (F_2(P_1, \mathbf{P}'_2) \wedge \dots \wedge F_l(P_1, \mathbf{P}'_l) \wedge F'(P_1, \mathbf{P}'_l))). \tag{A11}$$

Hence, it only remains to be shown that (A9) and (A11) are equivalent. Let E be the predicate expression $\lambda X_1 \dots X_n \, G(X_1, \dots, X_n)$ such that $H(E) = F_1(P_1)$, with $H(Q)$ being the following formula:

$$\forall V_1 \dots V_n \, (P_1(V_1, \dots, V_n) \leftrightarrow Q(V_1, \dots, V_n)).$$

Then,

$$\begin{aligned}
\text{(A9)} &\Leftrightarrow \exists P_1\,(F_1(P_1) \wedge \exists \mathbf{P}_2\,(F_2(E,\mathbf{P}'_2) \wedge \cdots \wedge F_l(E,\mathbf{P}'_l) \wedge F'(E,\mathbf{P}'_l)))\\
&\Leftrightarrow \exists P_1\,F_1(P_1) \wedge \exists \mathbf{P}_2\,(F_2(E,\mathbf{P}'_2) \wedge \cdots \wedge F_l(E,\mathbf{P}'_l) \wedge F'(E,\mathbf{P}'_l))\\
&\Leftrightarrow \top \wedge \exists \mathbf{P}_2\,(F_2(E,\mathbf{P}'_2) \wedge \cdots \wedge F_l(E,\mathbf{P}'_l) \wedge F'(E,\mathbf{P}'_l))\\
&\Leftrightarrow \exists \mathbf{P}_2\,(F_2(E,\mathbf{P}'_2) \wedge \cdots \wedge F_l(E,\mathbf{P}'_l) \wedge F'(E,\mathbf{P}'_l))\\
&\Leftrightarrow \forall P_1\,(F_1(P_1) \rightarrow \exists \mathbf{P}_2\,(F_2(E,\mathbf{P}'_2) \wedge \cdots \wedge F_l(E,\mathbf{P}'_l) \wedge F'(E,\mathbf{P}'_l)))\\
&\Leftrightarrow \text{(A11)},
\end{aligned}$$

and the result holds. For the second-to-last equivalence, note that $F_1(P_1)$ is satisfiable and that P_1 does not occur on the right-hand side of the implication. □

Proof of Theorem 3. Recall that io-program Ω uses private recursion if

- its predicate dependency graph has a cycle such that every vertex in it is a private symbol or
- it includes a choice rule with a private symbol in the head.

This implies that, for a program that does not use private recursion, there is a private predicate symbol that does not depend on any other private predicate symbol. Let us assume without loss of generality that this is the predicate symbol p_1/n_1. Then, there is a predicate symbol that does not depend on any other private predicate symbol other than p_1/n_1, which we assume to be the predicate symbol p_2/n_2, and so on. Therefore, we have an order on the private symbols $p_1/n_1, \ldots, p_l/n_l$ such that each predicate symbol p_i/n_i only depends on other predicate symbols that precede them in this order. Then, the completed definition $F_i(\mathbf{P}_i)$ of any private predicate symbol p_i/n_i can be written as

$$\forall V_1 \ldots V_{n_i}\,(P_i(V_1, \ldots, V_{n_i}) \leftrightarrow G_i(\mathbf{P}_{i-1})),$$

where $\mathbf{P}_{i-1} = P_1, \ldots, P_{i-1}$ contains all free predicate variables in G_i and where we assume that $G_1(\mathbf{P}_0)$ is a first-order formula. Then, (15) and (16) can be, respectively, rewritten as (A7) and (A8). The result follows then directly from Lemma 9. □