A Functional View of Strong Negation

Michael Bartholomew and Joohyung Lee

School of Computing, Informatics, and Decision Systems Engineering
Arizona State University, Tempe, USA

Abstract. The distinction between strong negation and default negation has been useful in answer set programming. We present an alternative account of strong negation, which lets us understand strong negation in terms of Boolean functions under the functional stable model semantics. More specifically, we show that, under complete interpretations, minimizing both positive and negative literals in the traditional answer set semantics is essentially the same as ensuring the uniqueness of function values under the functional stable model semantics. The same account lets us view Lifschitz's two-valued logic programs as a special case of the functional stable model semantics. In addition, we show how non-Boolean intensional functions can be eliminated in favor of Boolean intensional functions, and furthermore can be represented using strong negation, which provides a way to compute functional stable model semantics using existing ASP solvers.

1 Introduction

The distinction between default negation and strong negation has been useful in answer set programming. In particular, it yields an elegant solution to the frame problem. The fact that block b stays at the same location l by inertia can be described by the rule

$$\mathit{On}(b,l,t{+}1) \;\leftarrow\; \mathit{On}(b,l,t),\; \mathit{not}\;\sim\!\mathit{On}(b,l,t{+}1) \tag{1}$$

along with the rule that describes the uniqueness of location values [Lifschitz, 2002]:

$$\sim\!\mathit{On}(b,l_1,t) \;\leftarrow\; \mathit{On}(b,l,t),\; l \neq l_1\;. \tag{2}$$

Here '$\sim$' is the symbol for strong negation that represents explicit falsity while '*not*' is the symbol for default negation (negation as failure). Rule (1) asserts that without explicit evidence to the contrary, block b remains at location l. If we are given explicit conflicting information about the location of b at time $t{+}1$ then this conclusion will be defeated by rule (2), which asserts the uniqueness of location.

An alternative representation of inertia, which uses choice rules instead of strong negation, was presented by Bartholomew and Lee [2012]. Instead of rule (1), they use choice rules:

$$\{\mathit{On}(b,l,t{+}1)\} \;\leftarrow\; \mathit{On}(b,l,t)\;, \tag{3}$$

which states that "if b is at l at time t, then decide arbitrarily whether to assert that b is at l at time $t{+}1$." Instead of rule (2), they write weaker rules for describing the functional property of *On*:

$$\leftarrow\; \{\mathit{On}(b,l,t) : \mathit{Location}(l)\}0 \qquad \text{existence of location} \tag{4}$$

$$\leftarrow\; 2\{\mathit{On}(b,l,t) : \mathit{Location}(l)\} \qquad \text{uniqueness of location.} \tag{5}$$

In the absence of additional information about the location of block b at time $t+1$, asserting *On*$(b, l, t{+}1)$ is the only option, in view of the existence of location constraint (4). But if we are given conflicting information about the location of b at time $t{+}1$ then not asserting *On*$(b, l, t+1)$ is the only option, in view of the uniqueness of location constraint (5). [1]

The set of rules (3), (4), and (5) can be more succinctly represented in the language of [Bartholomew and Lee, 2012], which can express intensional functions. That is, the three rules can be replaced by one rule

$$\{Loc(b, t{+}1) = l\} \ \leftarrow \ Loc(b, t) = l \tag{6}$$

where *Loc* is an intensional function constant. (The rule reads, "if block b is at location l at time t, by default, the block is at l at time $t{+}1$.") In fact, Corollary 2 of [Bartholomew and Lee, 2012] tells us how to eliminate intensional functions in favor of intensional predicates, justifying the equivalence between (6) and the set of rules (3), (4), and (5). The translation allows us to compute the language of [Bartholomew and Lee, 2012] using existing ASP solvers, such as SMODELS and GRINGO, but not DLV because DLV currently does not handle choice rules. On the other hand, all these solvers allow strong negation, and can accept rules like (1) and (2).

The two representations of inertia involving intensional predicate *On* do not result in the same answer sets. In the first representation, which uses strong negation, each answer set contains only one atom of the form *On*(b, l, t) for each block b and each time t; for all other locations l', negative literals $\sim$*On*(b, l', t) belong to the answer set. On the other hand, such negative literals do not occur in the answer sets of a program that follows the second representation, which uses choice rules. The difference can be well explained by the difference between the symmetric and the asymmetric views of predicates that Lifschitz described in his message to Texas Action Group, titled "Choice Rules and the Belief-Based View of ASP": [2]

> The way I see it, in ASP programs we use predicates of two kinds, let's call them "symmetric" and "asymmetric." The fact that an object a does not have a property p is reflected by the presence of $\sim p(a)$ in the answer set if p is "symmetric," and by the absence of $p(a)$ if p is "asymmetric." In the second case, the strong negation of p is not used in the program at all.

According to these terminologies, predicate *On* is symmetric in the first representation above, and asymmetric in the second representation.

This paper contains several technical results that help us understand the relationship between these two views. In this regard, it helps us to understand strong negation as a way to express intensional Boolean functions.

- Our first result provides an alternative account of strong negation in terms of Boolean intensional functions. For instance, (1) can be identified with

$$On(b, l, t{+}1) = \text{TRUE} \ \leftarrow \ On(b, l, t) = \text{TRUE} \wedge \neg(On(b, l, t{+}1) = \text{FALSE}) \ , \tag{7}$$

  and (2) can be identified with

$$On(b, l_1, t) = \text{FALSE} \ \leftarrow \ On(b, l, t) = \text{TRUE} \wedge l \neq l_1 \ . \tag{8}$$

[1] The two rules can be combined into one: $\leftarrow \ not\ 1\{On(b, l, t) : Location(l)\}1$.

[2] http://www.cs.utexas.edu/users/vl/tag/choice_discussion

Under complete interpretations, we show that minimizing both positive and negative literals in the traditional answer set semantics is essentially the same as ensuring the uniqueness of function values under the functional stable model semantics. In this sense, the use of strong negation can be viewed as a mere disguise of using Boolean functions.

– We show how non-Boolean intensional functions can be eliminated in favor of Boolean functions. Combined with the result in the first bullet, this tells us a new way to turn the language of [Bartholomew and Lee, 2012] into traditional answer set programs with strong negation, so that system DLV, as well as SMODELS and GRINGO, can be used for computing the language of [Bartholomew and Lee, 2012]. For instance, it tells us how to turn (6) into the set of rules (1) and (2).
– Lifschitz [2012] recently proposed "two-valued logic programs," which modifies the traditional stable model semantics so that it can represent complete information without distinguishing between strong negation and default negation. Using our result that views strong negation in terms of Boolean functions, we show that two-valued logic programs are in fact a special case of the functional stable model semantics in which every function is Boolean.

The paper is organized as follows. In Section 2 we review the two versions of the stable model semantics, one that allows strong negation, but is limited to express intensional predicates only, and the other that allows both intensional predicates and intensional functions. As a special case of the latter we also present multi-valued propositional formulas under the stable model semantics. Section 3 shows how strong negation can be viewed in terms of Boolean functions. Section 4 shows how non-Boolean functions can be eliminated in favor of Boolean functions. Section 5 shows how Lifschitz's two-valued logic programs can be viewed as a special case of functional stable model semantics.

A long version with complete proofs is available at `http://peace.eas.asu.edu/joolee/papers/sneg-long.pdf`.

2 Preliminaries

2.1 Review: First-Order Stable Model Semantics and Strong Negation

This review follows [Ferraris *et al.*, 2011]. A *signature* is defined as in first-order logic, consisting of *function constants* and *predicate constants*. Function constants of arity 0 are also called *object constants*. We assume the following set of primitive propositional connectives and quantifiers: $\bot$ (falsity), $\wedge$, $\vee$, $\rightarrow$, $\forall$, $\exists$. The syntax of a formula is defined as in first-order logic. We understand $\neg F$ as an abbreviation of $F \rightarrow \bot$.

The stable models of a sentence F relative to a list of predicates $\mathbf{p} = (p_1, \dots, p_n)$ are defined via the *stable model operator with the intensional predicates* $\mathbf{p}$, denoted by SM$[F; \mathbf{p}]$. Let $\mathbf{u}$ be a list of distinct predicate variables $u_1, \dots, u_n$ of the same length as $\mathbf{p}$. By $\mathbf{u} = \mathbf{p}$ we denote the conjunction of the formulas $\forall \mathbf{x}(u_i(\mathbf{x}) \leftrightarrow p_i(\mathbf{x}))$, where $\mathbf{x}$ is a list of distinct object variables of the same length as the arity of p_i, for all $i = 1, \dots, n$. By $\mathbf{u} \leq \mathbf{p}$ we denote the conjunction of the formulas $\forall \mathbf{x}(u_i(\mathbf{x}) \rightarrow p_i(\mathbf{x}))$ for all $i = 1, \dots, n$, and $\mathbf{u} < \mathbf{p}$ stands for $(\mathbf{u} \leq \mathbf{p}) \wedge \neg(\mathbf{u} = \mathbf{p})$. For any first-order sentence F, expression SM$[F; \mathbf{p}]$ stands for the second-order sentence

$$F \wedge \neg \exists \mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge F^*(\mathbf{u})), \tag{9}$$

where $F^*(\mathbf{u})$ is defined recursively:

- $p_i(\mathbf{t})^* = u_i(\mathbf{t})$ for any list $\mathbf{t}$ of terms;
- $F^* = F$ for any atomic formula F (including $\bot$ and equality) that does not contain members of $\mathbf{p}$;
- $(F \wedge G)^* = F^* \wedge G^*$; $(F \vee G)^* = F^* \vee G^*$;
- $(F \rightarrow G)^* = (F^* \rightarrow G^*) \wedge (F \rightarrow G)$;
- $(\forall x F)^* = \forall x F^*$; $(\exists x F)^* = \exists x F^*$.

A model of a sentence F (in the sense of first-order logic) is called $\mathbf{p}$-*stable* if it satisfies SM$[F; \mathbf{p}]$. We will often simply write SM$[F]$ instead of SM$[F; \mathbf{p}]$ when $\mathbf{p}$ is the list of all predicate constants occurring in F, and call a model of SM$[F]$ simply a *stable* model of F.

The traditional stable models of a logic program Π are identical to the Herbrand stable models of the *FOL-representation* of Π (i.e., the conjunction of the universal closures of implications corresponding to the rules).

Ferraris *et al.* [2011] incorporate strong negation into the stable model semantics by distinguishing between intensional predicates of two kinds, *positive* and *negative*. Each negative intensional predicate has the form $\sim p$, where p is a positive intensional predicate and '$\sim$' is a symbol for strong negation. In this sense, syntactically $\sim$ is not a logical connective, as it can appear only as a part of a predicate constant. An interpretation of the underlying signature is *coherent* if it satisfies the formula $\neg \exists \mathbf{x}(p(\mathbf{x}) \wedge \sim p(\mathbf{x}))$, where $\mathbf{x}$ is a list of distinct object variables, for each negative predicate $\sim p$. We consider coherent interpretations only.

Example 1 *The following is a representation of the Blocks World in the syntax of logic programs:*

$$\begin{array}{rcll}
\bot & \leftarrow & \mathit{On}(b_1, b, t), \mathit{On}(b_2, b, t) & (b_1 \neq b_2) \\
\mathit{On}(b, l, t+1) & \leftarrow & \mathit{Move}(b, l, t) & \\
\bot & \leftarrow & \mathit{Move}(b, l, t), \mathit{On}(b_1, b, t) & \\
\bot & \leftarrow & \mathit{Move}(b, b_1, t), \mathit{Move}(b_1, l, t) & \\
\mathit{On}(b, l, 0) & \leftarrow & \mathit{not}\ \sim\mathit{On}(b, l, 0) & \\
\sim\mathit{On}(b, l, 0) & \leftarrow & \mathit{not}\ \mathit{On}(b, l, 0) & \\
\mathit{Move}(b, l, t) & \leftarrow & \mathit{not}\ \sim\mathit{Move}(b, l, t) & \\
\sim\mathit{Move}(b, l, t) & \leftarrow & \mathit{not}\ \mathit{Move}(b, l, t) & \\
\mathit{On}(b, l, t+1) & \leftarrow & \mathit{On}(b, l, t), \mathit{not}\ \sim\mathit{On}(b, l, t+1) & \\
\sim\mathit{On}(b, l, t) & \leftarrow & \mathit{On}(b, l_1, t) & (l \neq l_1)\ .
\end{array} \tag{10}$$

Here On and Move are predicate constants, b, b_1, b_2 are variables ranging over the blocks, l, l_1 are variables ranging over the locations (blocks and the table), and t is a variable ranging over the timepoints. The first rule states that only one block can be on a block. The next three rules describe the effect and preconditions of action Move. The next four rules describe that fluent On is initially exogenous, and action Move is exogenous at each time. The next rule describes the inertia, and the last rule asserts that a block can be in only one location.

2.2 Review: Functional Stable Model Semantics

Functional stable model semantics is defined by modifying the semantics in the previous section to allow "intensional" functions [Bartholomew and Lee, 2012]. For predicate symbols (constants or variables) u and c, we define $u \leq c$ as $\forall \mathbf{x}(u(\mathbf{x}) \rightarrow c(\mathbf{x}))$. We define $u = c$ as $\forall \mathbf{x}(u(\mathbf{x}) \leftrightarrow c(\mathbf{x}))$ if u and c are predicate symbols, and $\forall \mathbf{x}(u(\mathbf{x}) = c(\mathbf{x}))$ if they are function symbols.

Let $\mathbf{c}$ be a list of distinct predicate and function constants and let $\widehat{\mathbf{c}}$ be a list of distinct predicate and function variables corresponding to $\mathbf{c}$. We call members of $\mathbf{c}$ *intensional* constants. By $\mathbf{c}^{pred}$ we mean the list of the predicate constants in $\mathbf{c}$, and by $\widehat{\mathbf{c}}^{pred}$ the list of the corresponding predicate variables in $\widehat{\mathbf{c}}$. We define $\widehat{\mathbf{c}} < \mathbf{c}$ as $(\widehat{\mathbf{c}}^{pred} \leq \mathbf{c}^{pred}) \wedge \neg(\widehat{\mathbf{c}} = \mathbf{c})$ and $\mathrm{SM}[F;\ \mathbf{c}]$ as

$$F \wedge \neg \exists \widehat{\mathbf{c}}(\widehat{\mathbf{c}} < \mathbf{c} \wedge F^*(\widehat{\mathbf{c}})),$$

where $F^*(\widehat{\mathbf{c}})$ is defined the same as the one in Section 2.1 except for the base case:

– When F is an atomic formula, F^* is $F' \wedge F$, where F' is obtained from F by replacing all intensional (function and predicate) constants in it with the corresponding (function and predicate) variables.

If $\mathbf{c}$ contains predicate constants only, this definition of a stable model reduces to the one in [Ferraris *et al.*, 2011], also reviewed in Section 2.1.

According to [Bartholomew and Lee, 2012], a *choice formula* $\{F\}$ is an abbreviation of the formula $F \vee \neg F$, which is also strongly equivalent to $\neg\neg F \rightarrow F$. A formula $\{\mathbf{t} = \mathbf{t}'\}$, where $\mathbf{t}$ contains an intensional function constant and $\mathbf{t}'$ does not, represents that $\mathbf{t}$ takes the value $\mathbf{t}'$ by default.

Example 2 *For example, let F_1 be $\{f = 1\}$, which stands for $(f = 1) \vee \neg(f = 1)$, and I_1 be an interpretation such that $I_1(f) = 1$. Let's assume here that we consider only interpretations that maps numbers to themselves. I_1 is an f-stable model of F_1: to see this $F_1^*(\widehat{f})$ is equivalent to $((\widehat{f} = 1) \wedge (f = 1)) \vee \neg(f = 1)$,[3] which is further equivalent to $(\widehat{f} = 1)$ under the assumption I_1. It is not possible to satisfy this formula by assigning $\widehat{f}$ a different value from $I_1(f)$. On the other hand, I_2 such that $I_2(f) = 2$ is not f-stable since $F_1^*(\widehat{f})$ is equivalent to $\top$ under I_2, so that it is possible to satisfy this formula by assigning $\widehat{f}$ a different value from $I_2(f)$. If we let F_2 be $\{f = 1\} \wedge (f = 2)$, then I_2 is f-stable: $F_2^*(\widehat{f})$ is equivalent to $\widehat{f} = 2$ under I_2, so that $\widehat{f}$ has to map to 2 as well. This example illustrates the nonmonotonicity of the semantics.*

Example 3 *The Blocks World can be described in this language as follows. For readability, we write in a logic program like syntax:*

$$\begin{array}{rcl}
\bot & \leftarrow & \mathit{Loc}(b_1, t) = b \wedge \mathit{Loc}(b_2, t) = b \wedge (b_1 \neq b_2) \\
\mathit{Loc}(b, t{+}1) = l & \leftarrow & \mathit{Move}(b, l, t) \\
\bot & \leftarrow & \mathit{Move}(b, l, t) \wedge \mathit{Loc}(b_1, t) = b \\
\bot & \leftarrow & \mathit{Move}(b, b_1, t) \wedge \mathit{Move}(b_1, l, t) \\
\{\mathit{Loc}(b, 0) = l\} & & \\
\{\mathit{Move}(b, l, t)\} & & \\
\{\mathit{Loc}(b, t{+}1) = l\} & \leftarrow & \mathit{Loc}(b, t) = l\ .
\end{array}$$

Here Loc is a function constant. The last rule is a default formula that describes the commonsense law of inertia. The stable models of this program are the models of $\mathrm{SM}[F;\ \mathit{Loc}, \mathit{Move}]$*, where F is the FOL-representation of the program.*

[3] It holds that $(\neg F)^*$ is equivalent to $\neg F$.

2.3 Review: Stable Models of Multi-Valued Propositional Formulas

The following is a review of the stable model semantics of multi-valued propositional formulas from [Bartholomew and Lee, 2012], which can be viewed as a special case of functional stable model semantics in the previous section.

The syntax of multi-valued propositional formulas is given in [Ferraris *et al.*, 2011]. A *multi-valued propositional signature* is a set σ of symbols called *constants*, along with a nonempty finite set $Dom(c)$ of symbols, disjoint from σ, assigned to each constant c. We call $Dom(c)$ the *domain* of c. A *Boolean* constant is one whose domain is the set $\{\text{TRUE}, \text{FALSE}\}$. An *atom* of a signature σ is an expression of the form $c{=}v$ ("the value of c is v") where $c \in \sigma$ and $v \in Dom(c)$. A *(multi-valued propositional) formula* of σ is a propositional combination of atoms.

A *(multi-valued propositional) interpretation* of σ is a function that maps every element of σ to an element of its domain. An interpretation I *satisfies* an atom $c{=}v$ (symbolically, $I \models c{=}v$) if $I(c) = v$. The satisfaction relation is extended from atoms to arbitrary formulas according to the usual truth tables for the propositional connectives. I is a *model* of a formula if it satisfies the formula.

The reduct F^I of a multi-valued propositional formula F relative to a multi-valued propositional interpretation I is the formula obtained from F by replacing each maximal subformula that is not satisfied by I with $\bot$. Interpretation I is a *stable model* of F if I is the only interpretation satisfying F^I.

Example 4 *Similar to Example 2, consider the signature $\sigma = \{f\}$ such that $Dom(c) = \{1, 2, 3\}$. Let I_1 be an interpretation such that $I_1(c) = 1$, and I_2 is such that $I_2(c) = 2$. Recall that $\{f = 1\}$ is shorthand for $(f = 1) \vee \neg(f = 1)$. The reduct of this formula relative to I_1 is $(f{=}1) \vee \bot$, and I_1 is the only model of the reduct. On the other hand, the reduct of $\{f{=}1\}$ relative to I_2 is $(\bot \vee \neg\bot)$ and I_2 is not its unique model. Also, the reduct of $\{f{=}1\} \wedge (f{=}2)$ relative to I_1 is $(\bot \vee \neg\bot) \wedge \bot$ and I_1 is not a model. The reduct of $\{f{=}1\} \wedge (f{=}2)$ relative to I_2 is $(\bot \vee \neg\bot) \wedge (f{=}2)$, and I_2 is the only model of the reduct.*

2.4 Multi-Valued Propositional Formulas and FO Formulas

Multi-valued propositional formulas can be identified with a special case of first-order formulas as follows. Let F be a multi-valued propositional formula of signature σ. We identify σ with a first-order signature σ' that consists of

- all symbols from σ as object constants, and
- all symbols from $Dom(c)$ where c is in σ as object constants.

We may view multi-valued propositional interpretations of σ as a special case of first-order interpretations of σ'. We say that a first-order interpretation I of σ' *conforms* to σ if

- the universe of I is the union of $Dom(c)$ for all c in σ;
- $c^I \in Dom(c)$ for every c in σ;
- $v^I = v$ for every v in $Dom(c)$ where $c \in \sigma$.

Proposition 1 *For any multi-valued propositional formula F of signature σ such that $|Dom(c)| \geq 2$ for every $c \in \sigma$, an interpretation I of σ is a multi-valued propositional stable model of F iff I is an interpretation of σ' that conforms to σ and satisfies* $\text{SM}[F; \sigma]$.

3 Relating Strong Negation to Boolean Functions

3.1 Representing Strong Negation in Multi-Valued Propositional Formulas

Given a traditional propositional logic program Π of a signature σ [Gelfond and Lifschitz, 1991], we identify σ with the multi-valued propositional signature whose constants are the same symbols from σ and every constant is Boolean. By Π^{mv} we mean the multi-valued propositional formula that is obtained from Π by replacing negative literals of the form $\sim p$ with $p = \text{FALSE}$ and positive literals of the form p with $p = \text{TRUE}$.

We say that a set X of literals from σ is *complete* if, for each atom $a \in \sigma$, either a or $\sim a$ is in X. We identify a complete set of literals from σ with the corresponding multi-valued propositional interpretation.

Theorem 1 *A complete set of literals is an answer set of Π in the sense of [Gelfond and Lifschitz, 1991] iff it is a stable model of Π^{mv} in the sense of [Bartholomew and Lee, 2012].*

The theorem tells us that checking the minimality of positive and negative literals under the traditional stable model semantics is essentially the same as checking the uniqueness of corresponding function values under the stable model semantics from [Bartholomew and Lee, 2012].

Example 5 *Consider the program that describes a simple transition system consisting of two states depending on whether fluent* p *is true or false, and an action that makes* p *true.*

$$
\begin{array}{rcl}
p_0 & \leftarrow & \textit{not} \ \sim p_0 \\
\sim p_0 & \leftarrow & \textit{not} \ p_0 \\
 & & \\
a & \leftarrow & \textit{not} \ \sim a \\
\sim a & \leftarrow & \textit{not} \ a
\end{array}
\qquad\qquad
\begin{array}{rcl}
p_1 & \leftarrow & a \\
 & & \\
p_1 & \leftarrow & p_0, \textit{not} \ \sim p_1 \\
\sim p_1 & \leftarrow & \sim p_0, \textit{not} \ p_1 \ .
\end{array}
\tag{11}
$$

The program has four answer sets, each of which corresponds to one of the four edges of the transition system. For instance, $\{\sim p_0, a, p_1\}$ is an answer set. This program can be encoded in the input languages of GRINGO *and* DLV. *In the input language of* DLV, *which allows disjunctions in the head, the four rules in the first column can be succinctly replaced by*

$$
p_0 \vee \sim p_0 \qquad\qquad a \vee \sim a \ .
$$

According to Theorem 1, the stable models of this program are the same as the stable models of the following multi-valued propositional formula (written in a logic program style; '$\neg$' represents default negation):

$$
\begin{array}{rcl}
p_0 = \text{TRUE} & \leftarrow & \neg(p_0 = \text{FALSE}) \\
p_0 = \text{FALSE} & \leftarrow & \neg(p_0 = \text{TRUE}) \\
 & & \\
a = \text{TRUE} & \leftarrow & \neg(a = \text{FALSE}) \\
a = \text{FALSE} & \leftarrow & \neg(a = \text{TRUE})
\end{array}
\qquad
\begin{array}{rcl}
p_1 = \text{TRUE} & \leftarrow & a = \text{TRUE} \\
 & & \\
p_1 = \text{TRUE} & \leftarrow & p_0 = \text{TRUE} \wedge \neg(p_1 = \text{FALSE}) \\
p_1 = \text{FALSE} & \leftarrow & p_0 = \text{FALSE} \wedge \neg(p_1 = \text{TRUE})
\end{array}
$$

3.2 Relation among Strong Negation, Default Negation, Choice Rules and Boolean Functions

In certain cases, strong negation can be replaced by default negation, and furthermore the expression can be rewritten in terms of choice rules, which often yields a simpler representation. The following theorem, which extends the *Theorem on Double Negation* from [Ferraris *et al.*, 2009] to allow intensional functions, presents a condition under which equivalent transformations in classical logic preserve stable models.

Theorem 2 *Let F be a sentence, let $\mathbf{c}$ be a list of predicate and function constants, and let I be a (coherent) interpretation. Let F' be a formula obtained from F by replacing a subformula $\neg H$ with $\neg H'$ such that $I \models \widetilde{\forall}(H \leftrightarrow H')$. Then*

$$I \models \mathrm{SM}[F;\ \mathbf{c}] \textit{ iff } I \models \mathrm{SM}[F';\ \mathbf{c}]\ .$$

We say that an interpretation is *complete* on a predicate p if it satisfies $\forall \mathbf{x}(p(\mathbf{x}) \vee \sim p(\mathbf{x}))$.

Theorem 2, in particular, allows us to replace strong negation occurring in H with default negation when we consider complete interpretations I since $I \models \sim p(\mathbf{t})$ iff $I \models \neg p(\mathbf{t})$.

Example 5 continued *Each answer set of the first program in Example 5 is complete. In view of Theorem 2, the first two rules can be rewritten as $p_0 \leftarrow$ not not p_0 and $\sim p_0 \leftarrow$ not not $\sim p_0$, which can be further abbreviated as choice rules $\{p_0\}$ and $\{\sim p_0\}$. Consequently, the whole program can be rewritten using choice rules as*

$$\begin{array}{ccr}
\{p_0\} & \qquad & p_1 \leftarrow a \\
\{\sim p_0\} & & \\
 & & \{p_1\} \leftarrow p_0 \\
\{a\} & & \{\sim p_1\} \leftarrow \sim p_0\ . \\
\{\sim a\} & &
\end{array}$$

Similarly, since $I \models (p_0 = \mathrm{FALSE})$ iff $I \models \neg(p_0 = \mathrm{TRUE})$, in view of Theorem 2, the first rule of the second program in Example 5 can be rewritten as $p_0 = \mathrm{TRUE} \leftarrow \neg\neg(p_0 = \mathrm{TRUE})$ and further as $\{p_0 = \mathrm{TRUE}\}$. This transformation allows us to rewrite the whole program as

$$\begin{array}{ccr}
\{p_0 = B\} & \qquad & p_1 = \mathrm{TRUE} \leftarrow a = \mathrm{TRUE} \\
\{a = B\} & & \{p_1 = B\} \leftarrow p_0 = B
\end{array}$$

where B ranges over $\{\mathrm{TRUE}, \mathrm{FALSE}\}$. This program represents the transition system more succinctly than program (11).

3.3 Representing Strong Negation by Boolean Functions in the First-Order Case

Theorem 1 can be extended to the first-order case as follows.

Let f be a function constant. A first-order formula is called *f-plain* if each atomic formula

- does not contain f, or
- is of the form $f(\mathbf{t}) = u$ where $\mathbf{t}$ is a tuple of terms not containing f, and u is a term not containing f.

For example, $f = 1$ is f-plain, but each of $p(f)$, $g(f) = 1$, and $1 = f$ are not f-plain.

For a list $\mathbf{c}$ of predicate and function constants, we say that F is $\mathbf{c}$-plain if F is f-plain for each function constant f in $\mathbf{c}$. Roughly speaking, $\mathbf{c}$-plain formulas do not allow the functions in $\mathbf{c}$ to be nested in another predicate or function, and at most one function in $\mathbf{c}$ is allowed in each atomic formula. For example, $f = g$ is not (f, g)-plain, and neither is $f(g) = 1 \rightarrow g = 1$.

Let F be a formula possibly containing strong negation. Formula $F^{(p,\sim p)}_{b}$ is obtained from F as follows:

- in the signature of F, replace p and $\sim p$ with a new intensional function constant b of arity n, where n is the arity of p (or $\sim p$), and add two non-intensional object constants TRUE and FALSE;
- replace every occurrence of $\sim p(\mathbf{t})$, where $\mathbf{t}$ is a list of terms, with $b(\mathbf{t}) = \text{FALSE}$, and then replace every occurrence of $p(\mathbf{t})$ with $b(\mathbf{t}) = \text{TRUE}$.

By BC_b ("Boolean Constraint on b") we denote the conjunction of the following formulas, which enforces b to be a Boolean function:

$$\text{TRUE} \neq \text{FALSE}, \tag{12}$$

$$\neg\neg\forall \mathbf{x}(b(\mathbf{x}) = \text{TRUE} \vee b(\mathbf{x}) = \text{FALSE}). \tag{13}$$

where $\mathbf{x}$ is a list of distinct object variables.

Theorem 3 *Let $\mathbf{c}$ be a set of predicate and function constants, and let F be a $\mathbf{c}$-plain formula. Formulas*

$$\forall \mathbf{x}((p(\mathbf{x}) \leftrightarrow b(\mathbf{x}) = \text{TRUE}) \wedge (\sim p(\mathbf{x}) \leftrightarrow b(\mathbf{x}) = \text{FALSE})), \tag{14}$$

and BC_b entail

$$\text{SM}[F;\ p, \sim p, \mathbf{c}] \leftrightarrow \text{SM}[F^{(p,\sim p)}_{b};\ b, \mathbf{c}] .$$

If we drop the requirement that F be $\mathbf{c}$-plain, the statement does not hold as in the following example demonstrates.

Example 6 *Take $\mathbf{c}$ as (f, g) and let F be $p(f) \wedge \sim p(g)$. $F^{(p,\sim p)}_{b}$ is $b(f) = \text{TRUE} \wedge b(g) = \text{FALSE}$. Consider the interpretation I whose universe is $\{1, 2\}$ such that I contains $p(1), \sim p(2)$ and with the mappings $b^I(1) = \text{TRUE}, b^I(2) = \text{FALSE}, f^I = 1, g^I = 2$. I certainly satisfies BC_b and (14). I also satisfies $\text{SM}[F;\ p, \sim p, f, g]$ but does not satisfy $\text{SM}[F^{(p,\sim p)}_{b};\ b, f, g]$: we can let I be $\widehat{b}^I(1) = \text{FALSE}, \widehat{b}^I(2) = \text{TRUE}, \widehat{f}^I = 2, \widehat{g}^I = 1$ to satisfy both $(\widehat{b}, \widehat{f}, \widehat{g}) < (b, f, g)$ and $(F^{(p,\sim p)}_{b})^*(\widehat{b}, \widehat{f}, \widehat{g})$, which is*

$$b(f) = \text{TRUE} \wedge \widehat{b}(\widehat{f}) = \text{TRUE} \wedge b(g) = \text{FALSE} \wedge \widehat{b}(\widehat{g}) = \text{FALSE}.$$

Note that any interpretation that satisfies both (14) and BC_b is complete on p. Theorem 3 tells us that for any interpretation I that is complete on p, minimizing the extents of both p and $\sim p$ has the same effect as ensuring that the corresponding Boolean function b have a unique value.

The following corollary shows that there is a 1–1 correspondence between the stable models of F and the stable models of $F^{(p,\sim p)}_{b}$. For any interpretation I of the signature of F that is complete on p, by $I_b^{(p,\sim p)}$ we denote the interpretation of the signature of $F_b^{(p,\sim p)}$ obtained from I by replacing the relation p^I with function b^I such that

$$b^I(\xi_1,\dots,\xi_n) = \text{TRUE}^I \ \text{ if } \ p^I(\xi_1,\dots,\xi_n) = \text{TRUE};$$
$$b^I(\xi_1,\dots,\xi_n) = \text{FALSE}^I \ \text{ if } \ (\sim p)^I(\xi_1,\dots,\xi_n) = \text{TRUE} \; .$$

(Notice that we overloaded the symbols TRUE and FALSE: object constants on one hand, and truth values on the other hand.) Since I is complete on p and coherent, b^I is well-defined. We also require that $I_b^{(p,\sim p)}$ satisfy (12). Consequently, $I_b^{(p,\sim p)}$ satisfies BC_b.

Corollary 1 *Let* $\mathbf{c}$ *be a set of predicate and function constants, and let* F *be a* $\mathbf{c}$*-plain sentence.* (I) *An interpretation* I *of the signature of* F *that is complete on* p *is a model of* $\text{SM}[F;\ p, \sim p, \mathbf{c}]$ *iff* $I_b^{(p,\sim p)}$ *is a model of* $\text{SM}[F_b^{(p,\sim p)};\ b, \mathbf{c}]$. (II) *An interpretation* J *of the signature of* $F_b^{(p,\sim p)}$ *is a model of* $\text{SM}[F_b^{(p,\sim p)} \wedge BC_b;\ b, \mathbf{c}]$ *iff* $J = I_b^{(p,\sim p)}$ *for some model* I *of* $\text{SM}[F; p, \sim p, \mathbf{c}]$.

The other direction, eliminating Boolean intensional functions in favor of symmetric predicates, is similar as we show in the following.

Let F be a $(b, \mathbf{c})$-plain formula such that every atomic formula containing b has the form $b(\mathbf{t}) = \text{TRUE}$ or $b(\mathbf{t}) = \text{FALSE}$, where $\mathbf{t}$ is any list of terms. Formula $F^{\,b}_{(p,\sim p)}$ is obtained from F as follows:

- in the signature of F, replace b with predicate constants p and $\sim p$, whose arities are the same as that of b;
- replace every occurrence of $b(\mathbf{t}) = \text{TRUE}$, where $\mathbf{t}$ is any list of terms, with $p(\mathbf{t})$, and $b(\mathbf{t}) = \text{FALSE}$ with $\sim p(\mathbf{t})$.

Theorem 4 *Let* $\mathbf{c}$ *be a set of predicate and function constants, let* b *be a function constant, and let* F *be a* $(b, \mathbf{c})$*-plain formula such that every atomic formula containing* b *has the form* $b(\mathbf{t}) = \text{TRUE}$ *or* $b(\mathbf{t}) = \text{FALSE}$. *Formulas (14) and* BC_b *entail*

$$\text{SM}[F;\ b, \mathbf{c}] \leftrightarrow \text{SM}[F^{\,b}_{(p,\sim p)};\ p, \sim p, \mathbf{c}] \; .$$

The following corollary shows that there is a 1–1 correspondence between the stable models of F and the stable models of $F^{\,b}_{(p,\sim p)}$. For any interpretation I of the signature of F that satisfies BC_b, by $I^{\,b}_{(p,\sim p)}$ we denote the interpretation of the signature of $F^{\,b}_{(p,\sim p)}$ obtained from I by replacing the function b^I with predicate p^I such that

$$p^I(\xi_1,\dots,\xi_n) = \text{TRUE} \ \text{ iff } \ b^I(\xi_1,\dots,\xi_n) = \text{TRUE}^I \; ;$$
$$(\sim p)^I(\xi_1,\dots,\xi_n) = \text{TRUE} \ \text{ iff } \ b^I(\xi_1,\dots,\xi_n) = \text{FALSE}^I \; .$$

Corollary 2 *Let* $\mathbf{c}$ *be a set of predicate and function constants, let* b *be a function constant, and let* F *be a* $(b, \mathbf{c})$*-plain sentence such that every atomic formula containing* b *has the form* $b(\mathbf{t}) = \text{TRUE}$ *or* $b(\mathbf{t}) = \text{FALSE}$. (I) *A coherent interpretation* I *of the signature of* F *is a model of* $\text{SM}[F \wedge BC_b;\ b, \mathbf{c}]$ *iff* $I^{\,b}_{(p,\sim p)}$ *is a model of* $\text{SM}[F^{\,b}_{(p,\sim p)};\ p, \sim p, \mathbf{c}]$. (II) *An interpretation* J *of the signature of* $F^{\,b}_{(p,\sim p)}$ *is a model of* $\text{SM}[F^{\,b}_{(p,\sim p)}; p, \sim p, \mathbf{c}]$ *iff* $J = I^{\,b}_{(p,\sim p)}$ *for some model* I *of* $\text{SM}[F \wedge BC_b; b, \mathbf{c}]$.

An example of this corollary is shown in the next section.

4 Representing Non-Boolean Functions Using Strong Negation

In this section, we show how to eliminate non-Boolean intensional functions in favor of Boolean intensional functions. Combined with the method in the previous section, it gives us a systematic method of representing non-Boolean intensional functions using strong negation.

4.1 Eliminating non-Boolean Functions in favor of Boolean Functions

Let F be an f-plain formula. Formula F^f_b is obtained from F as follows:

- in the signature of F, replace f with a new boolean intensional function b of arity $n+1$ where n is the arity of f;
- replace each subformula $f(\mathbf{t}) = c$ with $b(\mathbf{t}, c) = \text{TRUE}$.

By UE_b, we denote the following formulas that preserve the functional property:

$$\forall \mathbf{x} yz(y \neq z \wedge b(\mathbf{x}, y) = \text{TRUE} \rightarrow b(\mathbf{x}, z) = \text{FALSE}), \tag{15}$$

$$\neg\neg\forall \mathbf{x} \exists y(b(\mathbf{x}, y) = \text{TRUE}), \tag{16}$$

where $\mathbf{x}$ is a n-tuple of variables and all variables in $\mathbf{x}$, y, and z are pairwise distinct.

Theorem 5 *For any f-plain formula F,*

$$\forall \mathbf{x} y\big((f(\mathbf{x}) = y \leftrightarrow b(\mathbf{x}, y) = \text{TRUE}) \wedge (f(\mathbf{x}) \neq y \leftrightarrow b(\mathbf{x}, y) = \text{FALSE})\big)$$

and $\exists xy(x \neq y)$ entail

$$\text{SM}[F;\ f, \mathbf{c}] \ \leftrightarrow\ \text{SM}[F^f_b \wedge UE_b;\ b, \mathbf{c}]\ .$$

By I^f_b, we denote the interpretation of the signature of F^f_b obtained from I by replacing the function f^I with the function b^I such that

$$\begin{array}{ll} b^I(\xi_1, \dots, \xi_n, \xi_{n+1}) = \text{TRUE}^I & \text{if } f^I(\xi_1, \dots, \xi_n) = \xi_{n+1} \\ b^I(\xi_1, \dots, \xi_n, \xi_{n+1}) = \text{FALSE}^I & \text{otherwise.} \end{array}$$

Corollary 3 *Let F be an f-plain sentence.* (I) *An interpretation I of the signature of F that satisfies $\exists xy(x \neq y)$ is a model of* $\text{SM}[F;\ f, \mathbf{c}]$ *iff I^f_b is a model of* $\text{SM}[F^f_b \wedge UE_b;\ b, \mathbf{c}]$. (II) *An interpretation J of the signature of F^f_b that satisfies $\exists xy(x \neq y)$ is a model of* $\text{SM}[F^f_b \wedge UE_b;\ b, \mathbf{c}]$ *iff $J = I^f_b$ for some model I of* $\text{SM}[F;\ f, \mathbf{c}]$.

Example 3 continued *In the program in Example 3, we eliminate non-Boolean function Loc in favor of Boolean function On as follows. The last two rules are UE_{On}.*

$$\begin{array}{rcl}
\bot & \leftarrow & On(b_1, b, t) = \text{TRUE} \wedge On(b_2, b, t) = \text{TRUE} \wedge b_1 \neq b_2 \\
On(b, l, t+1) = \text{TRUE} & \leftarrow & Move(b, l, t) \\
\bot & \leftarrow & Move(b, l, t) \wedge On(b_1, b, t) = \text{TRUE} \\
\bot & \leftarrow & Move(b, b_1, t) \wedge Move(b_1, l, t) \\
\{On(b, l, 0) = \text{TRUE}\} & & \\
\{Move(b, l, t)\} & & \\
\{On(b, l, t+1) = \text{TRUE}\} & \leftarrow & On(b, l, t) = \text{TRUE} \\
On(b, l, t) = \text{FALSE} & \leftarrow & On(b, l_1, t) = \text{TRUE} \wedge l \neq l_1 \\
\bot & \leftarrow & not\ \exists l(On(b, l, t) = \text{TRUE})\ .
\end{array}$$

For this program, it is not difficult to check that the last rule is redundant. Indeed, since the second to the last rule is the only rule that has $On(b,l,t) = \text{FALSE}$ *in the head, one can check that any model that does not satisfy* $\exists l(On(b,l,t) = \text{TRUE})$ *is not stable even if we drop the last rule.*

Corollary 2 tells us that this program can be represented by an answer set program containing strong negation (with the redundant rule dropped.)

$$\begin{array}{rcll}
\bot & \leftarrow & On(b_1,b,t), On(b_2,b,t) & (b_1 \neq b_2) \\
On(b,l,t+1) & \leftarrow & Move(b,l,t) & \\
\bot & \leftarrow & Move(b,l,t), On(b_1,b,t) & \\
\bot & \leftarrow & Move(b,b_1,t), Move(b_1,l,t) & \\
\{On(b,l,0)\} & & & \\
\{Move(b,l,t)\} & & & \\
\{On(b,l,t+1)\} & \leftarrow & On(b,l,t) & \\
\sim On(b,l,t) & \leftarrow & On(b,l_1,t) & (l \neq l_1) \; .
\end{array} \tag{17}$$

Let us compare this program with program (10)*. Similar to the explanation in Example 5 (continued), the 5th and the 7th rules of* (10) *can be represented using choice rules, which are the same as the 5th and the 6th rules of* (17)*. The 6th and the 7th rules of* (10) *is the closed world assumption. We can check that adding these rules to* (17) *extends the answer sets of* (10) *in a conservative way with the definition of the negative literals. This tells us that the answer sets of the two programs are in a 1-1 correspondence.*

As the example explains, non-Boolean functions can be represented using strong negation by composing the two methods, first eliminating non-Boolean functions in favor of Boolean functions as in Corollary 3 and then eliminating Boolean functions in favor of predicates as in Corollary 2. In the following we state this composition.

Let F be an f-plain formula where f is an intensional function constant. Formula F^f_p is obtained from F as follows:

- in the signature of F, replace f with two new intensional predicates p and $\sim p$ of arity $n+1$ where n is the arity of f;
- replace each subformula $f(\mathbf{t}) = c$ with $p(\mathbf{t}, c)$.

By UE_p, we denote the following formulas that preserve the functional property:

$$\forall \mathbf{x} yz(y \neq z \wedge p(\mathbf{x}, y) \rightarrow \sim p(\mathbf{x}, z)) \; ,$$

$$\neg\neg\forall \mathbf{x} \exists y(p(\mathbf{x}, y)) \; ,$$

where $\mathbf{x}$ is an n-tuple of variables and all variables in $\mathbf{x}, y, z$ are pairwise distinct.

Theorem 6 *For any* $(f, \mathbf{c})$*-plain formula* F*, formulas*

$$\forall \mathbf{x} y(f(\mathbf{x}) = y \leftrightarrow p(\mathbf{x}, y)), \;\; \forall \mathbf{x} y(f(\mathbf{x}) \neq y \leftrightarrow \sim p(\mathbf{x}, y)), \;\; \exists xy(x \neq y)$$

entail

$$\text{SM}[F;\; f, \mathbf{c}] \leftrightarrow \text{SM}[F^f_p \wedge UE_p;\; p, \sim p, \mathbf{c}] \; .$$

By $I^f_{(p,\sim p)}$, we denote the interpretation of the signature of $F^f_{(p,\sim p)}$ obtained from I by replacing the function f^I with the set p^I that consists of the tuples $\langle \xi_1, \dots, \xi_n, f^I(\xi_1, \dots, \xi_n)\rangle$ for all $\xi_1, \dots, \xi_n$ from the universe of I. We then also add the set $(\sim p)^I$ that consists of the tuples $\langle \xi_1, \dots, \xi_n, \xi_{n+1}\rangle$ for all $\xi_1, \dots, \xi_n, \xi_{n+1}$ from the universe of I that do not occur in the set p^I.

Corollary 4 *Let F be an $(f, \mathbf{c})$-plain sentence.* (I) *An interpretation I of the signature of F that satisfies $\exists xy(x \neq y)$ is a model of* $\mathrm{SM}[F;\ f, \mathbf{c}]$ *iff $I^f_{(p,\sim p)}$ is a model of* $\mathrm{SM}[F^f_p \wedge \mathit{UE}_p;\ p, \sim p, \mathbf{c}]$. (II) *An interpretation J of the signature of F^f_p that satisfies $\exists xy(x \neq y)$ is a model of* $\mathrm{SM}[F^f_p \wedge \mathit{UE}_p;\ p, \sim p, \mathbf{c}]$ *iff $J = I^f_{(p,\sim p)}$ for some model I of* $\mathrm{SM}[F;\ f, \mathbf{c}]$.

Theorem 6 and Corollary 4 are similar to Theorem 8 and Corollary 2 from [Bartholomew and Lee, 2012]. The main difference is that the latter statements refer to a constraint called UEC_p that is weaker than UE_p. For instance, the elimination method from [Bartholomew and Lee, 2012] turns the Blocks World in Example 3 into an almost the same program as (17) except that the last rule is turned into a constraint UEC_{On}:

$$\leftarrow\ \mathit{On}(b, l, t) \wedge \mathit{On}(b, l_1, t) \wedge l \neq l_1\ . \tag{18}$$

It is clear that the stable models of $F^f_p \wedge \mathit{UE}_p$ are under the symmetric view, and the stable models of $F^f_p \wedge \mathit{UEC}_p$ are under the asymmetric view. To see how replacing UE_{On} by UEC_{On} turns the symmetric view to the asymmetric view, first observe that adding (18) to program (17) does not affect the stable models of the program. Let's call this program Π. It is easy to see that Π is a conservative extension of the program that is obtained from Π by deleting the rule with $\sim\mathit{On}(b, l, t)$ in the head.

5 Relating to Lifschitz's Two-Valued Logic Programs

Lifschitz [2012] presented a high level logic program that does not contain explicit default negation, but can handle nonmonotonic reasoning in a similar style as in default logic. In this section we show how his formalism can be viewed as a special case of multi-valued propositional formulas under the stable model semantics in which every function is assumed to be Boolean.

5.1 Review: Two-Valued Logic Programs

Let σ be a signature in propositional logic. A *two-valued rule* is an expression of the form

$$L_0\ \leftarrow\ L_1, \dots, L_n : F \tag{19}$$

where $L_0, \dots, L_n$ are propositional literals formed from σ and F is a propositional formula of signature σ.

A *two-valued program* Π is a set of two-valued rules. An interpretation I is a function from σ to $\{\textsc{true}, \textsc{false}\}$. The *reduct* of a program Π relative to an interpretation I, denoted Π^I, is the set of rules $L_0\ \leftarrow\ L_1, \dots, L_n$ corresponding to the rules (19) of Π for which $I \models F$. Interpretation I is a stable model of Π if it is a minimal model of Π^I.

Example 7

$$a \leftarrow \;:\; a, \qquad \neg a \leftarrow : \neg a, \qquad b \leftarrow a \;:\; \top \tag{20}$$

The reduct of this program relative to $\{a, b\}$ consists of rules a and $b \leftarrow a$. Interpretation $\{a, b\}$ is the minimal model of the reduct, so that it is a stable model of the program.

As described in [Lifschitz, 2012], if F in every rule (19) has the form of conjunctions of literals, then two-valued logic program can be turned into a program with strong negation, when we consider complete answer sets only. For instance, program (20) can be turned into

$$a \leftarrow \textit{not} \sim a, \qquad \sim a \leftarrow \textit{not}\, a, \qquad b \leftarrow a\,.$$

This program has two answer sets, $\{a, b\}$ and $\sim a$, and only the complete answer set $\{a, b\}$ corresponds to the stable model found in Example 7.

5.2 Translation into SM with Boolean Functions

Given a two-valued logic program Π of a signature σ, we identify σ with the multi-valued propositional signature whose constants are from σ and the domain of every constant is Boolean values $\{\text{TRUE}, \text{FALSE}\}$. For any propositional formula G, $Tr(G)$ is obtained from G by replacing every negative literal $\sim A$ with $A = \text{FALSE}$ and every positive literal A with $A = \text{TRUE}$. By $tv2sm(\Pi)$ we denote the multi-valued propositional formula which is defined as the conjunction of

$$\neg\neg Tr(F) \wedge Tr(L_1) \wedge \cdots \wedge Tr(L_n) \rightarrow Tr(L_0)$$

for each rule (19) in Π.

For any interpretation I of σ, we obtain the multi-valued interpretation I' from I as follows. For each atom A in σ,

$$I'(A) = \begin{cases} \text{TRUE} & \text{if } I \models A \\ \text{FALSE} & \text{if } I \models \neg A \end{cases}$$

Theorem 7 *For any two-valued logic program Π, an interpretation I is a stable model of Π in the sense of [Lifschitz, 2012] iff I' is a stable model of $tv2sm(\Pi)$ in the sense of [Bartholomew and Lee, 2012].*

Example 7 continued For the program Π in Example 7, $tv2sm(\Pi)$ is the following multi-valued propositional formula:

$$\big(\neg\neg(a{=}\text{TRUE}) \rightarrow a{=}\text{TRUE}\big) \wedge \big(\neg\neg(a{=}\text{FALSE}) \rightarrow a{=}\text{FALSE}\big) \wedge \big(a{=}\text{TRUE} \rightarrow b{=}\text{TRUE}\big).$$

According to [Bartholomew and Lee, 2012], this too has only one stable model in which a and b are both mapped to TRUE, corresponding the unique stable model of Π according to Lifschitz.

Consider now that (19) contains variables. It is not difficult to see that $tv2sm(\Pi)$ can be straightforwardly extended to non-ground programs. This accounts for providing the semantics of the first-order extension of two-valued logic programs.

6 Conclusion

In this note, we showed that, under complete interpretations, symmetric predicates using strong negation can be alternatively expressed in terms of Boolean intensional functions in the language of [Bartholomew and Lee, 2012]. Bartholomew and Lee [2013] show that a similar result holds with functional stable model semantics by Cabalar [Cabalar, 2011], but without requiring complete interpretations. We expect that other results in this paper can be extended without requiring the complete interpretation assumption by using the semantics by Cabalar. On the other hand, this would require us to refer to the notion of *partial* satisfaction, which is more complicated than the standard notion of satisfaction.

System CPLUS2ASP [Casolary and Lee, 2011; Babb and Lee, 2013] turns action language $\mathcal{C}+$ into answer set programs containing asymmetric predicates. The translation in this paper that eliminates intensional functions in favor of symmetric predicates provides an alternative method of computing $\mathcal{C}+$ using ASP solvers.

Acknowledgements: We are grateful to Vladimir Lifschitz for bringing attention to this subject. We are also grateful to Gregory Gelfond for useful discussions related to this paper.

References

[Babb and Lee, 2013] Joseph Babb and Joohyung Lee. CPLUS2ASP: Computing action language C+ in answer set programming. In *Proceedings of International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, 2013. To appear.

[Bartholomew and Lee, 2012] Michael Bartholomew and Joohyung Lee. Stable models of formulas with intensional functions. In *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 2–12, 2012.

[Bartholomew and Lee, 2013] Michael Bartholomew and Joohyung Lee. On the stable model semantics for intensional functions. *TPLP*, 2013. To appear.

[Cabalar, 2011] Pedro Cabalar. Functional answer set programming. *TPLP*, 11(2-3):203–233, 2011.

[Casolary and Lee, 2011] Michael Casolary and Joohyung Lee. Representing the language of the causal calculator in answer set programming. In *ICLP (Technical Communications)*, pages 51–61, 2011.

[Ferraris *et al.*, 2009] Paolo Ferraris, Joohyung Lee, Vladimir Lifschitz, and Ravi Palla. Symmetric splitting in the general theory of stable models. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 797–803. AAAI Press, 2009.

[Ferraris *et al.*, 2011] Paolo Ferraris, Joohyung Lee, and Vladimir Lifschitz. Stable models and circumscription. *Artificial Intelligence*, 175:236–263, 2011.

[Gelfond and Lifschitz, 1991] Michael Gelfond and Vladimir Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385, 1991.

[Gelfond *et al.*, 1991] Michael Gelfond, Vladimir Lifschitz, and Arkady Rabinov. What are the limitations of the situation calculus? In Robert Boyer, editor, *Automated Reasoning: Essays in Honor of Woody Bledsoe*, pages 167–179. Kluwer, 1991.

[Lifschitz, 2002] Vladimir Lifschitz. Answer set programming and plan generation. *Artificial Intelligence*, 138:39–54, 2002.

[Lifschitz, 2012] Vladimir Lifschitz. Two-valued logic programs. In *ICLP (Technical Communications)*, pages 259–266, 2012.

A Proof of Proposition 1

In the following section, we use $m(I)$ to denote the first-order interpretation that is identified with the multi-valued propositional interpretation I so that $m(I)(c) = I(c)$ for every c in σ.

Lemma 1. *For any multi-valued propositional formula F of signature σ, a multi-valued propositional interpretation I satisfies F iff the first-order interpretation $m(I)$ that conforms to σ satisfies F.*

Proof. By induction on F. ∎

Proposition 1. *For any multi-valued propositional formula F of signature σ such that $|Dom(c)| \geq 2$ for every $c \in \sigma$, an interpretation I of σ is a multi-valued propositional stable model of F iff I is an interpretation of σ' that conforms to σ and satisfies* $\text{SM}[F; \sigma]$.

Proof. We consider the reduct-based definition of the functional stable model semantics from [Bartholomew and Lee, 2013] rather than the second order logic based definition. Since the notions of satisfaction coincides for any multi-valued propositional formula by Lemma 1, we have that $I \models G$ iff $m(I) \models G$ for any subformula G of F and therefore, F^I is identical to $F^{m(I)}$. Therefore, since it is the case by Lemma 1 that $I \models F$ iff $m(I) \models F$, only the second items in the reduct definitions remain to be shown to coincide. That is, it is sufficient to check that

there exists a multi-valued interpretation $J \neq I$ satisfying F^I

iff

there exists a first-order interpretation $K <^\sigma m(I)$ satisfying $F^{m(I)}$.

Left-to-right: Assume that there exists a multi-valued interpretation $J \neq I$ satisfying F^I. We will show how to create a first-order interpretation $K <^\sigma m(I)$ satisfying $F^{m(I)}$. Let $K = m(J)$ and it is easy to see that since $J \neq I$, then $K <^\sigma m(I)$. Further, since $J \models F^I$, by Lemma 1 it follows that $K \models F^I$ and finally since F^I is identical to $F^{m(I)}$, $K \models F^{m(I)}$.

Right-to-left: Assume that there exists a first-order interpretation $K <^\sigma m(I)$ satisfying $F^{m(I)}$. We will show how to create an interpretation $J \neq I$ satisfying F^I. Take J to be a multi-valued interpretation of σ defined as follows:

$$c^J = \begin{cases} c^I & \text{if } c^K = c^{m(I)} \\ \alpha_c(c^I) & \text{otherwise} \end{cases}$$

for every $c \in \sigma$ where α_c is a mapping from $Dom(c)$ to $Dom(c)$ such that $\forall x(\alpha_c(x) \neq x)$. Note that this mapping is only possible when $|Dom(c)| \geq 2$. Also note that since $K <^\sigma m(I)$, K and $m(I)$ differ only on symbols from σ; allowing the two to differ on the non-intensional constants in $Dom(c)$ would no longer preserve the coincidence.

We will show that $J \models F^I$ iff $K \models F^I$ (and thus conclude that $J \models F^I$) by induction on F (recall F^I is identical to $F^{m(I)}$).

Case 1: F is an atom $c = v$. If F^I is $\bot$ then the claim is trivial. If F^I is $c = v$, then it follows that $I(c) = v$. Consider two subcases:

– If $K \models F^I$ then by definition of J, we get $c^J = c^I = v$ so $J \models F^I$.
– If $K \not\models F^I$, then by definition of J, we get $c^J = \alpha_c(c^I) \neq v$ so $J \not\models F^I$.

In either case the claim holds.

Case 2: When F is $G \odot H$ where $\odot \in \{\wedge, \vee, \rightarrow\}$ the claim follows from I.H. on G and H.

Since we assumed that K satisfies F^I, we conclude that J satisfies F^I. ∎

A.1 Proof of Theorem 3 and Corollary 1

Theorem 3 *Let* $\mathbf{c}$ *be a set predicate and function constants, and let* F *be a* $\mathbf{c}$*-plain formula. Formulas*

$$\forall \mathbf{x}((p(\mathbf{x}) \leftrightarrow b(\mathbf{x}) = \text{TRUE}) \wedge ({\sim}p(\mathbf{x}) \leftrightarrow b(\mathbf{x}) = \text{FALSE})),$$

and BC_b *entail*

$$\text{SM}[F;\ p, {\sim}p,\ \mathbf{c}] \leftrightarrow \text{SM}[F_b^{(p,{\sim}p)};\ b,\ \mathbf{c}]\ .$$

Proof. For any interpretation $\mathcal{I} = \langle I, X\rangle$ of signature $\sigma \supseteq \{b, p, \mathbf{c}\}$ satisfying (14), it is clear that $\mathcal{I} \models F$ iff $\mathcal{I} \models F_b^{p{\sim}p}$ since $F_b^{p{\sim}p}$ is simply the result of replacing all $p(\mathbf{t})$ with $b(\mathbf{t}) = \text{TRUE}$ and all ${\sim}p(\mathbf{t})$ with $b(\mathbf{t}) = \text{FALSE}$. Thus it only remains to be shown that $\mathcal{I} \models \neg\exists\widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^{(p,{\sim}p)})^*(\widehat{b}, \widehat{\mathbf{c}}))$ iff $\mathcal{I} \models \neg\exists\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}} <{\sim}p, p, \mathbf{c}) \wedge F^*(\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}}))$ or equivalently, $\mathcal{I} \models \exists\widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^{(p,{\sim}p)})^*(\widehat{f}, \widehat{\mathbf{c}}))$ iff $\mathcal{I} \models \exists\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}} <{\sim}p, p, \mathbf{c}) \wedge F^*(\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}}))$.

($\Rightarrow$) Assume $\mathcal{I} \models \exists\widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^{(p{\sim}p)})^*(\widehat{b}, \widehat{\mathbf{c}}))$. We wish to show that $\mathcal{I} \models \exists\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}} <{\sim}p, p, \mathbf{c}) \wedge F^*(\widehat{{\sim}p}, \widehat{p}, \widehat{\mathbf{c}}))$

That is, take any function a of the same arity as b and any list of predicates and functions $\mathbf{d}$ of the same length $\mathbf{c}$. Now let $\mathcal{I}' = \langle I \cup J_{(a,\mathbf{d})}^{(b,\mathbf{c})}, X \cup Y_{\mathbf{d}}^{\mathbf{c}}\rangle$ be from an extended signature $\sigma' = \sigma \cup \{a, q, \mathbf{d}\}$ where J is an interpretation of functions from the signature σ and I and J agree on all symbols not occurring in $\{b, \mathbf{c}\}$. $J_{(a,\mathbf{d})}^{(b,\mathbf{c})}$ denotes the interpretation from $\sigma_{(a,\mathbf{d})}^{(b,\mathbf{c})}$ (the signature obtained from σ by replacing b with a and all elements of $\mathbf{c}$ with all elements of $\mathbf{d}$) obtained from the interpretation J by replacing b with a and the functions in $\mathbf{c}$ with the corresponding functions in $\mathbf{d}$. Similarly, $Y_{\mathbf{d}}^{\mathbf{c}}$ is the interpretation from σ' obtained from the interpretation Y by replacing predicates from $\mathbf{c}$ by the corresponding predicates from $\mathbf{d}$. We assume

$$\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c} \wedge (F_b^{(p,{\sim}p)})^*(a, \mathbf{d}))$$

and wish to show that there are predicates ${\sim}q$, q of the same arity as ${\sim}p$, p such that

$$\mathcal{I}' \models ({\sim}q, q, \mathbf{d} <{\sim}p, p, \mathbf{c} \wedge F^*({\sim}q, q, \mathbf{d})).$$

We define the new predicates ${\sim}q$, q in terms of b and a as follows:

$$\begin{aligned} {\sim}q(\mathbf{x}) &\leftrightarrow a(\mathbf{x}) = \text{FALSE} \wedge b(\mathbf{x}) = \text{FALSE} \\ q(\mathbf{x}) &\leftrightarrow a(\mathbf{x}) = \text{TRUE} \wedge b(\mathbf{x}) = \text{TRUE} \end{aligned}$$

We first show if $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$ then $\mathcal{I}' \models (\sim q, q, \mathbf{d} < \sim p, p, \mathbf{c})$:
Observe that from the definition of $\sim q$ and q, it follows that $\mathcal{I}' \models \forall \mathbf{x}(\sim q(\mathbf{x}) \rightarrow b(\mathbf{x}) = \text{FALSE}) \wedge \forall \mathbf{x}(q(\mathbf{x}) \rightarrow b(\mathbf{x}) = \text{TRUE})$ and from (14), this is equivalent to $\mathcal{I}' \models \forall \mathbf{x}(\sim q(\mathbf{x}) \rightarrow \sim p(\mathbf{x})) \wedge \forall \mathbf{x}(q(\mathbf{x}) \rightarrow p(\mathbf{x}))$ or simply $\mathcal{I}' \models \sim q, q \leq \sim p, p$. Thus, since $\mathcal{I}' \models \mathbf{d}^{pred} \leq \mathbf{c}^{pred}$, it follows that $\mathcal{I}' \models \sim q, q, \mathbf{d}^{pred} \leq \sim p, p, \mathbf{c}^{pred}$.
Case 1: $\mathcal{I}' \models \forall \mathbf{x}(b(\mathbf{x}) = a(\mathbf{x}))$.
In this case it then must be that $\mathcal{I}' \models \mathbf{d} \neq \mathbf{c}$. Thus it follows that $\mathcal{I}' \models \sim q, q, \mathbf{d} \neq \sim p, p, \mathbf{c}$. Consequently we conclude that

$$\mathcal{I}' \models (\sim q, q, \mathbf{d}^{pred} \leq \sim p, p, \mathbf{c}^{pred}) \wedge \sim q, q, \mathbf{d} \neq \sim p, p, \mathbf{c}$$

or simply, $\mathcal{I}' \models (\sim q, q, \mathbf{d} < \sim p, p, \mathbf{c})$.

Case 2: $\mathcal{I}' \models \neg \forall \mathbf{x}, y(b(\mathbf{x}) = a(\mathbf{x}))$.
That is, since $\mathcal{I}' \models BC_b$, there is some list of object names $\mathbf{t}$ such that either $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) \neq \text{FALSE}$ or $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) \neq \text{TRUE}$.
Subcase 1: $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) \neq \text{FALSE}$
By (14), $\mathcal{I}' \models \sim p(\mathbf{t})$ and by definition of $\sim q$, $\mathcal{I}' \models \neg \sim q(\mathbf{t})$ so $\mathcal{I}' \models \sim q \neq \sim p$.
Subcase 2: $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) \neq \text{TRUE}$
By (14), $\mathcal{I}' \models p(\mathbf{t})$ and by definition of q, $\mathcal{I}' \models \neg q(\mathbf{t})$ so $\mathcal{I}' \models q \neq p$.
Therefore, no matter which subcase holds, we have $\sim q, q \neq \sim p, p$ and thus $\sim q, q, \mathbf{d} \neq \sim p, p, \mathbf{c}$. Consequently we conclude

$$\mathcal{I}' \models (\sim q, q, \mathbf{d}^{pred} \leq \sim p, p, \mathbf{c}^{pred}) \wedge \sim q, q, \mathbf{d} \neq \sim p, p, \mathbf{c}$$

or simply, $\mathcal{I}' \models (\sim q, q, \mathbf{d} < \sim p, p, \mathbf{c})$.

We now show by induction that $\mathcal{I}' \models F^*(\sim q, q, \mathbf{d})$:

Case 1: F is an atomic formula not containing b.
$F_b^{(p,\sim p)}$ is exactly F thus $(F_b^{(p,\sim p)})^*(a, \mathbf{d})$ is exactly $F^*(\sim q, q, \mathbf{d})$ so certainly the claim holds.

Case 2: F is $\sim p(\mathbf{t})$, where $\mathbf{t}$ contains no itensional function constants.
$F^*(\sim q, q, \mathbf{d})$ is $\sim q(\mathbf{t})$.
$F_b^{(p,\sim p)}$ is $b(\mathbf{t}) = \text{FALSE}$.
$(F_b^{(p,\sim p)})^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) = \text{FALSE}$.
By the definition of $\sim q$, it is clear that $\mathcal{I}' \models F^*(\sim q, q, \mathbf{d})$ so certainly the claim holds.

Case 3: F is $p(\mathbf{t})$, where $\mathbf{t}$ contains no itensional function constants.
$F^*(\sim q, q, \mathbf{d})$ is $q(\mathbf{t})$.
$F_b^{(p,\sim p)}$ is $b(\mathbf{t}) = \text{TRUE}$.
$(F_b^{(p,\sim p)})^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) = \text{TRUE}$.
By the definition of q, it is clear that $\mathcal{I}' \models F^*(\sim q, q, \mathbf{d})$ so certainly the claim holds.

Case 4: F is $G \odot H$ where $\odot \in \{\wedge, \vee\}$.
By I.H. on G and H.

Case 5: F is $G \to H$.
By I.H. on G and H.

Case 6: F is $Q\mathbf{x}G(\mathbf{x})$ where $Q \in \{\forall, \exists\}$.
By I.H. on G.

($\Leftarrow$) Assume $\mathcal{I} \models \exists\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}} <\sim p, p, \mathbf{c}) \wedge F^*(\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}))$. We wish to show that $\mathcal{I} \models \exists\widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^{(p,\sim p)})^*(\widehat{b}, \widehat{\mathbf{c}}))$

That is, take any predicates $\sim q, q$ of the same arity as $\sim p, p$ and any list of predicates and functions $\mathbf{d}$ of the same length as $\mathbf{c}$ and let $\mathcal{I}' = \langle I \cup J_{(a,\mathbf{d})}^{(b,\mathbf{c})}, X \cup Y_{\mathbf{d}}^{\mathbf{c}}\rangle$ is defined as before. We assume

$$\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c} \wedge F^*(\sim q, q, \mathbf{d}))$$

and wish to show that there is a function a of the same arity as b such that

$$\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c} \wedge (F_b^{(p,\sim p)})^*(a, \mathbf{d})).$$

We define the new function a in terms of $\sim p$, p, $\sim q$, and q as follows:

$$\mathcal{I}' \models a(\mathbf{x}) = \text{TRUE iff } \mathcal{I}' \models ((p(\mathbf{x}) \wedge q(\mathbf{x})) \vee (\sim p(\mathbf{x}) \wedge \neg \sim q(\mathbf{x})))$$
$$\mathcal{I}' \models a(\mathbf{x}) = \text{FALSE iff } \mathcal{I}' \models\leftrightarrow ((\sim p(\mathbf{x}) \wedge \sim q(\mathbf{x})) \vee (p(\mathbf{x}) \wedge \neg q(\mathbf{x})))$$

Note that since $\mathcal{I}' \models (14)$, $\mathcal{I}' \models BC_b$ and $\mathcal{I}' \models\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c}$ this is a well-defined function. This is because $\mathcal{I}' \models (14)$ and $\mathcal{I}' \models BC_b$ guarantee that $\mathcal{I}'$ is complete on p. In addition to this, $\mathcal{I}' \models\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c}$ guarantees that the four cases covered in this definition are the only ones possible; for any given $\mathbf{t}$ exactly one of $p(\mathbf{t})$ and $\sim p(\mathbf{t})$ is true. Wlog, assume $p(\mathbf{t})$ then $\mathcal{I}' \models\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c}$ gives us that $\sim q(\mathbf{t})$ must be false and $q(\mathbf{t})$ may be true or false. The other two cases are symmetric by considering when $\sim p(\mathbf{t})$ is true.

We first show if $\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c})$ then $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$:
Observe that $\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c})$ by definition entails $\mathcal{I}' \models (\sim q, q, \mathbf{d}^{pred} \leq\sim p, p, \mathbf{c}^{pred})$ and further by definition, $\mathcal{I}' \models (\mathbf{d}^{pred} \leq \mathbf{c}^{pred})$ and then since b and a are not predicates, $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred})$.

Case 1: $\mathcal{I}' \models \forall\mathbf{x}(p(\mathbf{x}) \leftrightarrow q(\mathbf{x})) \wedge \forall\mathbf{x}(\sim p(\mathbf{x}) \leftrightarrow\sim q(\mathbf{x}))$.
In this case, $\mathcal{I}' \models (\sim p, p =\sim q, q)$ so for it to be the case that $\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c})$, it must be that $\mathcal{I}' \models \neg(\mathbf{c} = \mathbf{d})$. It then follows that $\mathcal{I}' \models \neg(b, \mathbf{c} = a, \mathbf{d})$. Consequently in this case, $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred}) \wedge \neg(b, \mathbf{c} = a, \mathbf{d})$ or simply $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.

Case 2: $\mathcal{I}' \models \neg(\forall\mathbf{x}y(p(\mathbf{x}) \leftrightarrow q(\mathbf{x})) \wedge \forall\mathbf{x}(\sim p(\mathbf{x}) \leftrightarrow\sim q(\mathbf{x})))$.
Since $\mathcal{I}' \models\sim q, q <\sim p, p$ and $\mathcal{I}' \models (14)$ and since $\mathcal{I}'$ is complete on p, there is some list of object names $\mathbf{t}$ such that either $\mathcal{I}' \models p(\mathbf{t}) \wedge \neg q(\mathbf{t})$ or $\mathcal{I}' \models\sim p(\mathbf{t}) \wedge \neg \sim q(\mathbf{t})$.
Subcase 1: $\mathcal{I}' \models p(\mathbf{t}) \wedge \neg q(\mathbf{t})$.
By (14), $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE}$ and by definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{FALSE}$. Thus, $\mathcal{I}' \models a \neq b$. Consequently, in this case $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred}) \wedge \neg(b, \mathbf{c} = a, \mathbf{d})$ or simply $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.

Subcase 2: $\mathcal{I}' \models \sim p(\mathbf{t}) \wedge \neg \sim q(\mathbf{t})$.
By (14), $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE}$ and by definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{TRUE}$. Thus, $\mathcal{I}' \models a \neq b$. Consequently, in this case $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred}) \wedge \neg(b, \mathbf{c} = a, \mathbf{d})$ or simply $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.

We now show by induction that $\mathcal{I}' \models (F_b^{(p,\sim p)})^*(a, \mathbf{d})$:

Case 1: F is an atomic formula not containing b.
$F_b^{(p,\sim p)}$ is exactly F thus $(F_b^{(p,\sim p)})^*(a, \mathbf{d})$ is exactly $F^*(\sim q, q, \mathbf{d})$ so certainly the claim holds.

Case 2: F is $\sim p(\mathbf{t})$.
$F^*(q, \mathbf{d})$ is $\sim q(\mathbf{t})$.
$F_b^{(p,\sim p)}$ is $b(\mathbf{t}) = \text{FALSE}$.
$(F_b^{(p,\sim p)})^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) = \text{FALSE}$.
By (14), $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE}$. By definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{FALSE}$.

Case 3: F is $p(\mathbf{t})$.
$F^*(q, \mathbf{d})$ is $q(\mathbf{t})$.
$F_b^{(p,\sim p)}$ is $b(\mathbf{t}) = \text{TRUE}$.
$(F_b^{(p,\sim p)})^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) = \text{TRUE}$.
By (14), $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE}$. By definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{TRUE}$.

Case 4: F is $G \odot H$ where $\odot \in \{\wedge, \vee\}$.
By I.H. on G and H.

Case 5: F is $G \to H$.
By I.H. on G and H.

Case 6: F is $Q\mathbf{x}G(\mathbf{x})$ where $Q \in \{\forall, \exists\}$.
By I.H. on G. ∎

Corollary 1 *For any formula F and any interpretation I of the signature of F that is complete on p, (a) I is a model of $\text{SM}[F;\ p, \sim p,\ \mathbf{c}]$ iff $I_b^{(p,\ \sim p)}$ is a model of $\text{SM}[F_b^{(p,\ \sim p)} \wedge BC_b;\ b,\ \mathbf{c}]$. (b) An interpretation J of the signature of $F_b^{(p,\ \sim p)}$ is a model of $\text{SM}[F_b^{(p,\ \sim p)} \wedge BC_b;\ b,\ \mathbf{c}]$ iff $J = I_b^{(p,\ \sim p)}$ for some model I of $\text{SM}[F; p, \sim p,\ \mathbf{c}]$.*

Proof. For two interpretations I of signature σ_1 and J of signature σ_2, by $I \cup J$ we denote the interpretation of signature $\sigma_1 \cup \sigma_2$ and universe $|I| \cup |J|$ that interprets all symbols occurring only in σ_1 in the same way I does and similarly for σ_2 and J. For symbols appearing in both σ_1 and σ_2, I must interpret these the same as J does, in which case $I \cup J$ also interprets the symbol in this way.

(a$\Rightarrow$) Assume $I \models \text{TRUE} \neq \text{FALSE}$ and $I_p^b \models \text{SM}[F; p, \sim p, \mathbf{c}]$. Since $I \models \text{TRUE} \neq \text{FALSE}$, $I \cup I_b^{(p \sim p)} \models \text{TRUE} \neq \text{FALSE}$ since by definition of $I_b^{(p \sim p)}$, I and $I_b^{(p \sim p)}$ share the same universe. By definition of $I_b^{(p \sim p)}$, $I \cup I_b^{(p \sim p)} \models (14)$. Therefore, since I is complete on p and by (14), $I \cup I^{(p\,\sim p)}_{b} \models BC_b$. Thus by Theorem 3, $I \cup I^{(p\,\sim p)}_{b} \models$ SM$[F^{p\,\sim p}_{b} \wedge BC_b;\ b\ \mathbf{c}] \leftrightarrow$ SM$[F; p, {\sim}p, \mathbf{c}]$.

Since we assume $I \models$ SM$[F; p, {\sim}p, \mathbf{c}]$, it is the case that $I \cup I^{(p\,\sim p)}_{b} \models$ SM$[F; p, {\sim}p, \mathbf{c}]$ and thus it must be the case that $I \cup I^{(p\,\sim p)}_{b} \models$ SM$[F^{(p\,\sim p)}_{b};\ b,\ \mathbf{c}]$. Since $I \cup I^{(p\,\sim p)}_{b} \models BC_b$ and BC_b is a constraint, $I \cup I^{(p\,\sim p)}_{b} \models$ SM$[F^{p\,\sim p}_{b} \wedge BC_b;\ b,\ \mathbf{c}]$. Therefore since the signature of I does contain b, we conclude $I^{(p\,\sim p)}_{b} \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b;\ b,\ \mathbf{c}]$.

(a$\Leftarrow$) Assume $I \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b; b, \mathbf{c}] \wedge (\text{TRUE} \neq \text{FALSE})$. Since $I \models \text{TRUE} \neq \text{FALSE}$, $I \cup I^{(p\,\sim p)}_{b} \models \text{TRUE} \neq \text{FALSE}$ since by definition of $I^{(p\,\sim p)}_{b}$, I and $I^{(p\,\sim p)}_{b}$ share the same universe. By definition of $I^{(p\,\sim p)}_{b}$, $I \cup I^{(p\,\sim p)}_{b} \models (14)$. Since we assume $I \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b; b, \mathbf{c}]$, it follows that $I \models BC_b$. Thus by Theorem 3, $I \cup I^{(p\,\sim p)}_{b} \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b; b, \mathbf{c}] \leftrightarrow$ SM$[F; p, {\sim}p, \mathbf{c}]$.

Since we assume $I^{(p\,\sim p)}_{b} \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b; b, \mathbf{c}]$, it is the case that $I \cup I^{b}_{p} \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b; b, \mathbf{c}]$ and thus since BC_b is a constraint, it follows that $I \cup I^{b}_{p} \models$ SM$[F^{(p\,\sim p)}_{b}; b, \mathbf{c}]$. It then follows that $I \cup I^{b}_{p} \models$ SM$[F; p, {\sim}p, \mathbf{c}]$. However since the signature of $I^{(p\,\sim p)}_{b}$ does not contain p, we conclude $I \models$ SM$[F; p, {\sim}p, \mathbf{c}]$.

(b$\Rightarrow$) Assume $J \models \text{TRUE} \neq \text{FALSE}$ and $J \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b;\ b\ \mathbf{c}]$. Let $I = J_{(p\,\sim p)}^{\;b}$ where $J_{(p\,\sim p)}^{\;b}$ denotes the interpretation of the signature of $F^{(p\,\sim p)}_{b} \wedge BC_b$ obtained from J by replacing the boolean function b with the predicate p such that
$I \models p^I(\xi_1, \dots, \xi_k)$ for all tuples such that $b^I(\xi_1, \dots, \xi_k) = \text{TRUE}$ and,
$I \models {\sim}p^I(\xi_1, \dots, \xi_k)$ for all tuples such that $b^I(\xi_1, \dots, \xi_k) = \text{FALSE}$.
Since $J \models BC_b$, this is a well-defined function.
Clearly, $J = I^{(p\,\sim p)}_{b}$ so it only remains to be shown that $I \models$ SM$[F; p, {\sim}p,\ \mathbf{c}]$.

Since I and J have the same universe and $J \models \text{TRUE} \neq \text{FALSE}$, it follows that $I \cup J \models \text{TRUE} \neq \text{FALSE}$. Also by the definition of $J_{(p\,\sim p)}^{\;b}$ $I \cup J \models (14)$. Also, since $J \models BC_b$, it follows that $I \cup J \models BC_b$.Thus by Theorem 3, $I \cup J \models$ SM$[F^{p\,\sim p}_{b};\ b,\ \mathbf{c}] \leftrightarrow$ SM$[F; p, {\sim}p, \mathbf{c}]$.

Since we assume $J \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b;\ b,\ \mathbf{c}]$, it is the case that $I \cup J \models$ SM$[F^{(p\,\sim p)}_{b} \wedge BC_b;\ b,\ \mathbf{c}]$ and since BC_b is a constraint, $I \cup J \models$ SM$[F^{(p\,\sim p)}_{b};\ b,\ \mathbf{c}]$. Thus it must be the case that $I \cup J \models$ SM$[F; p, {\sim}p, \mathbf{c}]$. Now since the signature of J does not contain p, we conclude $I \models$ SM$[F; p, {\sim}p, \mathbf{c}]$.

(b$\Leftarrow$)Take any I such that $J = I^{(p\,\sim p)}_{b}$ and $I \models$ SM$[F; p, {\sim}p, \mathbf{c}]$. Since $J \models \text{TRUE} \neq \text{FALSE}$ and I and J share the same universe, $I \cup J \models \text{TRUE} \neq \text{FALSE}$. By definition of $J = I^{(p\,\sim p)}_{b}$, $I \cup J \models (14)$. Since I is complete on p and $I \cup J \models (14)$, it follows that $I \cup I^{b}_{p} \models BC_b$. Thus by Theorem 3, $I \cup J \models$ SM$[F^{p\,\sim p}_{b};\ b,\ \mathbf{c}] \leftrightarrow$ SM$[F; p, {\sim}p, \mathbf{c}]$

Since we assume $I \models$ SM$[F; p, {\sim}p, \mathbf{c}]$, it is the case that $I \cup J \models$ SM$[F; p, {\sim}p, \mathbf{c}]$ and thus it must be the case that $I \cup J \models$ SM$[F^{p\,\sim p}_{b};\ b,\ \mathbf{c}]$. Since BC_b is a constraint, it then follows that $I \cup J \models$ SM$[F^{p\,\sim p}_{b} \wedge BC_b;\ b,\ \mathbf{c}]$. However since the signature of I does not contain b, we conclude $J \models$ SM$[F^{p\,\sim p}_{b} \wedge BC_b;\ b,\ \mathbf{c}]$. ∎

A.2 Proof of Theorem 1

Theorem 1 *A complete set of literals is an answer set of Π in the sense of [Gelfond* et al.*, 1991] iff it is a stable model of Π^{mv} in the sense of [Bartholomew and Lee, 2012].*

Proof. Let I be the interpretation formed from including all of the literals from X and all the assignments from the multi-valued view of X. Let us denote the set of all predicate symbols from X as $\mathbf{p}$ and their negative counterparts as $\sim\mathbf{p}$ and all of the function symbols from the multi-valued view of X as $\mathbf{b}$. Clearly I satisfies

$$\forall \mathbf{x}((p(\mathbf{x}) \leftrightarrow b(\mathbf{x}) = \text{TRUE}) \wedge (\sim p(\mathbf{x}) \leftrightarrow b(\mathbf{x}) = \text{FALSE})),$$

for each $p \in \mathbf{p}$ and the corresponding $b \in \mathbf{b}$. From this and since X is complete, it follows that $I \models BC_b$ for each $b \in \mathbf{b}$. Thus, we can apply Theorem 3 (multiple times) to conclude that $\text{SM}[\Pi^{FOL};\ \mathbf{p} \sim\mathbf{p}] \leftrightarrow \text{SM}[(\Pi^{mv})^{FOL};\ \mathbf{b}]$.

A.3 Proof of Theorem 2

Proposition 2 *If F is negative on $\mathbf{c}$ then*

$$(\widehat{\mathbf{c}} \leq \mathbf{c}) \rightarrow (F^*(\widehat{\mathbf{c}}) \leftrightarrow F)$$

is logically valid.

Proof. By induction. ∎

Theorem 2 *Let F be a sentence, let $\mathbf{c}$ be a list of predicate and function constants, and let I be a (coherent) interpretation. Let F' be a formula obtained from F by replacing a subformula $\neg H$ with $\neg H'$ such that $I \models \widetilde{\forall}(H \leftrightarrow H')$. Then*

$$I \models \text{SM}[F; \mathbf{c}] \text{ iff } I \models \text{SM}[F'; \mathbf{c}]\ .$$

Proof. By Proposition 2, the following formulas are logically valid:

$$\begin{array}{l}(\mathbf{d} \leq \mathbf{c}) \rightarrow ((\neg H)^*(\mathbf{d}) \leftrightarrow (\neg H)) \\ (\mathbf{d} \leq \mathbf{c}) \rightarrow ((\neg H')^*(\mathbf{d}) \leftrightarrow (\neg H')).\end{array}$$

where $\mathbf{d}$ is a list of new constants corresponding to $\mathbf{c}$. Since $I \models \widetilde{\forall}(H \leftrightarrow H')$, we conclude that

$$I \models (\mathbf{d} \leq \mathbf{c}) \rightarrow ((\neg H)^*(\mathbf{d}) \leftrightarrow (\neg H')^*(\mathbf{d}))$$

and consequently

$$I \models (\mathbf{d} \leq \mathbf{c}) \rightarrow ((F)^*(\mathbf{d}) \leftrightarrow (F')^*(\mathbf{d}))$$

and thus

$$I \models \text{SM}[F; \mathbf{c}] \text{ iff } I \models \text{SM}[F'; \mathbf{c}]\ .$$

∎

A.4 Proof of Theorem 4 and Corollary 2

Theorem 4 *Let* $\mathbf{c}$ *be a set of predicate and function constants, let* b *be a function constant, and let* F *be a* $(b, \mathbf{c})$*-plain formula such that every atomic formula containing* b *has the form* $b(\mathbf{t}) = \text{TRUE}$ *or* $b(\mathbf{t}) = \text{FALSE}$*. Formulas (14) and* BC_b *entail*

$$\text{SM}[F;\ b, \mathbf{c}] \leftrightarrow \text{SM}[F^b_{(p,\sim p)};\ p, \sim p, \mathbf{c}]\ .$$

Proof.

For any interpretation $\mathcal{I} = \langle I, X\rangle$ of signature $\sigma \supseteq \{b, p, \mathbf{c}\}$ satisfying (14) and BC_b, it is clear that $\mathcal{I} \models F$ iff $\mathcal{I} \models F^b_{(p,\sim p)}$ since $F^b_{(p,\sim p)}$ is simply the result of replacing all $b(\mathbf{x}) = \text{TRUE}$ with $p(\mathbf{x})$ and all $b(\mathbf{x}) = \text{FALSE}$ with $\sim p(\mathbf{x})$. Thus it only remains to be shown that $\mathcal{I} \models \neg\exists\widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge F^*(\widehat{b}, \widehat{\mathbf{c}}))$ iff $\mathcal{I} \models \neg\exists\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}} <\sim p, p, \mathbf{c}) \wedge (F^b_{(p,\sim p)})^*(\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}))$ or equivalently, $\mathcal{I} \models \exists\widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge F^*(\widehat{f}, \widehat{\mathbf{c}}))$ iff $\mathcal{I} \models \exists\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}} <\sim p, p, \mathbf{c}) \wedge (F^b_{(p,\sim p)})^*(\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}))$.

($\Rightarrow$) Assume $\mathcal{I} \models \exists\widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge F^*(\widehat{b}, \widehat{\mathbf{c}}))$. We wish to show that $\mathcal{I} \models \exists\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}} <\sim p, p, \mathbf{c}) \wedge (F^b_{(p,\sim p)})^*(\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}))$

That is, take any function a of the same arity as b and any list of predicates and functions $\mathbf{d}$ of the same length $\mathbf{c}$. Now let $\mathcal{I}' = \langle I \cup J^{(b,\mathbf{c})}_{(a,\mathbf{d})}, X \cup Y^{\mathbf{c}}_{\mathbf{d}}\rangle$ be from an extended signature $\sigma' = \sigma \cup \{a, q, \mathbf{d}\}$ where J is an interpretation of functions from the signature σ and I and J agree on all symbols not occurring in $\{b, \mathbf{c}\}$. $J^{(b,\mathbf{c})}_{(a,\mathbf{d})}$ denotes the interpretation from $\sigma^{(b,\mathbf{c})}_{(a,\mathbf{d})}$ (the signature obtained from σ by replacing b with a and all elements of $\mathbf{c}$ with all elements of $\mathbf{d}$) obtained from the interpretation J by replacing b with a and the functions in $\mathbf{c}$ with the corresponding functions in $\mathbf{d}$. Similarly, $Y^{\mathbf{c}}_{\mathbf{d}}$ is the interpretation from σ' obtained from the interpretation Y by replacing predicates from $\mathbf{c}$ by the corresponding predicates from $\mathbf{d}$. We assume

$$\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c} \wedge F^*(a, \mathbf{d}))$$

and wish to show that there are predicates $\sim q$, q of the same arity as $\sim p$, p such that

$$\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c} \wedge (F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d})).$$

We define the new predicates $\sim q$, q in terms of b and a as follows:

$$\begin{aligned} \sim q(\mathbf{x}) &\leftrightarrow a(\mathbf{x}) = \text{FALSE} \wedge b(\mathbf{x}) = \text{FALSE} \\ q(\mathbf{x}) &\leftrightarrow a(\mathbf{x}) = \text{TRUE} \wedge b(\mathbf{x}) = \text{TRUE} \end{aligned}$$

We first show if $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$ then $\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c})$:
Observe that from the definition of $\sim q$ and q, it follows that $\mathcal{I}' \models \forall\mathbf{x}(\sim q(\mathbf{x}) \rightarrow b(\mathbf{x}) = \text{FALSE}) \wedge \forall\mathbf{x}(q(\mathbf{x}) \rightarrow b(\mathbf{x}) = \text{TRUE})$ and from (14), this is equivalent to $\mathcal{I}' \models \forall\mathbf{x}(\sim q(\mathbf{x}) \rightarrow\sim p(\mathbf{x})) \wedge \forall\mathbf{x}(q(\mathbf{x}) \rightarrow p(\mathbf{x}))$ or simply $\mathcal{I}' \models\sim q, q \leq\sim p, p$. Thus, since $\mathcal{I}' \models \mathbf{d}^{pred} \leq \mathbf{c}^{pred}$, it follows that $\mathcal{I}' \models q, \mathbf{d}^{pred} \leq p, \mathbf{c}^{pred}$.
Case 1: $\mathcal{I}' \models \forall\mathbf{x}(b(\mathbf{x}) = a(\mathbf{x}))$.
In this case it then must be that $\mathcal{I}' \models \mathbf{d} \neq \mathbf{c}$. Thus it follows that $\mathcal{I}' \models\sim q, q, \mathbf{d} \neq\sim p, p, \mathbf{c}$. Consequently we conclude that

$$\mathcal{I}' \models (\sim q, q, \mathbf{d}^{pred} \leq\sim p, p, \mathbf{c}^{pred}) \wedge \sim q, q, \mathbf{d} \neq\sim p, p, \mathbf{c}$$

or simply, $\mathcal{I}' \models (\sim q, q, \mathbf{d} < \sim p, p, \mathbf{c})$.

Case 2: $\mathcal{I}' \models \neg\forall \mathbf{x}y(b(\mathbf{x}) = a(\mathbf{x}))$.
That is, since $\mathcal{I}' \models BC_b$, there is some list of object names $\mathbf{t}$ such that either $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) \neq \text{FALSE}$ or $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) \neq \text{TRUE}$.
Subcase 1: $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) \neq \text{FALSE}$
By (14), $\mathcal{I}' \models \sim p(\mathbf{t})$ and by definition of $\sim q$, $\mathcal{I}' \models \neg \sim q(\mathbf{t})$ so $\mathcal{I}' \models \sim q \neq \sim p$.
Subcase 2: $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) \neq \text{TRUE}$
By (14), $\mathcal{I}' \models p(\mathbf{t})$ and by definition of q, $\mathcal{I}' \models \neg q(\mathbf{t})$ so $\mathcal{I}' \models q \neq p$.
Therefore, no matter which subcase holds, we have $\sim q, q \neq \sim p, p$ and thus $\sim q, q, \mathbf{d} \neq \sim p, p, \mathbf{c}$. Consequently we conclude

$$\mathcal{I}' \models (\sim q, q, \mathbf{d}^{pred} \leq \sim p, p, \mathbf{c}^{pred}) \wedge \sim q, q, \mathbf{d} \neq \sim p, p, \mathbf{c}$$

or simply, $\mathcal{I}' \models (\sim q, q, \mathbf{d} < \sim p, p, \mathbf{c})$.

We now show by induction that $\mathcal{I}' \models (F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d})$:

Case 1: F is an atomic formula not containing b.
$F^b_{(p,\sim p)}$ is exactly F thus $F^*(a, \mathbf{d})$ is exactly $(F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d})$ so certainly the claim holds.

Case 2: F is $b(\mathbf{t}) = \text{FALSE}$.
$F^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) = \text{FALSE}$.
$F^b_{(p,\sim p)}$ is $\sim p(\mathbf{t})$.
$(F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d})$ is $\sim q(\mathbf{t})$.
By the definition of $\sim q$, it is clear that $\mathcal{I}' \models (F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d})$ so certainly the claim holds.

Case 3: F is $b(\mathbf{t}) = \text{TRUE}$.
$F^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) = \text{TRUE}$.
$F^b_{(p,\sim p)}$ is $p(\mathbf{t})$.
$(F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d})$ is $q(\mathbf{t})$.
By the definition of q, it is clear that $\mathcal{I}' \models (F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d})$ so certainly the claim holds.

Case 4: F is $G \odot H$ where $\odot \in \{\wedge, \vee\}$.
By I.H. on G and H.

Case 5: F is $G \rightarrow H$.
By I.H. on G and H.

Case 6: F is $Q\mathbf{x}G(\mathbf{x})$ where $Q \in \{\forall, \exists\}$.
By I.H. on G.

($\Leftarrow$) Assume $\mathcal{I} \models \exists \widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}((\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}} < \sim p, p, \mathbf{c}) \wedge (F^b_{(p,\sim p)})^*(\widehat{\sim p}, \widehat{p}, \widehat{\mathbf{c}}))$. We wish to show that $\mathcal{I} \models \exists \widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge F^*(\widehat{b}, \widehat{\mathbf{c}}))$

That is, take any predicates $\sim q$, q of the same arity as $\sim p$, p and any list of predicates and functions $\mathbf{d}$ of the same length as $\mathbf{c}$ and let $\mathcal{I}' = \langle I \cup J^{(b,\mathbf{c})}_{(a,\mathbf{d})}, X \cup Y^{\mathbf{c}}_{\mathbf{d}} \rangle$ is defined as before. We assume

$$\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c} \wedge (F^b_{(p,\sim p)})^*(\sim q, q, \mathbf{d}))$$

and wish to show that there is a function a of the same arity as b such that

$$\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c} \wedge F^*(a, \mathbf{d})).$$

We define the new function a in terms of $\sim p$, p, $\sim q$, and q as follows:

$$\begin{array}{l} a(\mathbf{x}) = \text{TRUE} \leftrightarrow ((p(\mathbf{x}) \wedge q(\mathbf{x})) \vee (\sim p(\mathbf{x}) \wedge \neg \sim q(\mathbf{x}))) \\ a(\mathbf{x}) = \text{FALSE} \leftrightarrow ((\sim p(\mathbf{x}) \wedge \sim q(\mathbf{x})) \vee (p(\mathbf{x}) \wedge \neg q(\mathbf{x}))) \end{array}$$

Note that since $\mathcal{I}' \models (14)$ and $\mathcal{I}' \models \sim q, q, \mathbf{d} <\sim p, p, \mathbf{c}$ this is a well-defined function.

We first show if $\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c})$ then $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$:
Observe that $\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c})$ by definition entails $\mathcal{I}' \models (\sim q, q, \mathbf{d}^{pred} \leq\sim p, p, \mathbf{c}^{pred})$ and further by definition, $\mathcal{I}' \models (\mathbf{d}^{pred} \leq \mathbf{c}^{pred})$ and then since b and a are not predicates, $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred})$.

Case 1: $\mathcal{I}' \models \forall \mathbf{x}(p(\mathbf{x}) \leftrightarrow q(\mathbf{x})) \wedge \forall \mathbf{x}(\sim p(\mathbf{x}) \leftrightarrow \sim q(\mathbf{x}))$.
In this case, $\mathcal{I}' \models (\sim p, p = \sim q, q)$ so for it to be the case that $\mathcal{I}' \models (\sim q, q, \mathbf{d} <\sim p, p, \mathbf{c})$, it must be that $\mathcal{I}' \models \neg(\mathbf{c} = \mathbf{d})$. It then follows that $\mathcal{I}' \models \neg(b, \mathbf{c} = a, \mathbf{d})$. Consequently in this case, $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred}) \wedge \neg(b, \mathbf{c} = a, \mathbf{d})$ or simply $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.

Case 2: $\mathcal{I}' \models \neg(\forall \mathbf{x} y(p(\mathbf{x}) \leftrightarrow q(\mathbf{x})) \wedge \forall \mathbf{x}(\sim p(\mathbf{x}) \leftrightarrow \sim q(\mathbf{x})))$.
Since $\mathcal{I}' \models \sim q, q <\sim p, p$ and $\mathcal{I}' \models (14)$, there is some list of object names $\mathbf{t}$ such that either $\mathcal{I}' \models p(\mathbf{t}) \wedge \neg q(\mathbf{t})$ or $\mathcal{I}' \models \sim p(\mathbf{t}) \wedge \neg \sim q(\mathbf{t})$.
Subcase 1: $\mathcal{I}' \models p(\mathbf{t}) \wedge \neg q(\mathbf{t})$.
By (14) $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE}$ and by definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{FALSE}$. Thus, $\mathcal{I}' \models a \neq b$. Consequently, in this case $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred}) \wedge \neg(b, \mathbf{c} = a, \mathbf{d})$ or simply $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.
Subcase 2: $\mathcal{I}' \models \sim p(\mathbf{t}) \wedge \neg \sim q(\mathbf{t})$.
By (14) $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE}$ and by definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{TRUE}$. Thus, $\mathcal{I}' \models a \neq b$. Consequently, in this case $\mathcal{I}' \models ((a, \mathbf{d})^{pred} \leq (b, \mathbf{c})^{pred}) \wedge \neg(b, \mathbf{c} = a, \mathbf{d})$ or simply $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.

We now show by induction that $\mathcal{I}' \models F^*(a, \mathbf{d})$:

Case 1: F is an atomic formula not containing b.
$F^b_{(p,\sim p)}$ is exactly F thus $F^*(a, \mathbf{d})$ is exactly $(F^b_{(p,\sim p)})^*(\sim q, \mathbf{d})$ so certainly the claim holds.

Case 2: F is $b(\mathbf{t}) = \text{FALSE}$.
$F^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{FALSE} \wedge a(\mathbf{t}) = \text{FALSE}$.

$F^b_{(p,\sim p)}$ is $\sim p(\mathbf{t})$.
$(F^b_{(p,\sim p)})^*(q, \mathbf{d})$ is $\sim q(\mathbf{t})$.
By (14), $\mathcal{I}' \models b(\mathbf{t}) = \text{FALSE}$. By definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{FALSE}$.

Case 3: F is $b(\mathbf{t}) = \text{TRUE}$.
$F^*(a, \mathbf{d})$ is $b(\mathbf{t}) = \text{TRUE} \wedge a(\mathbf{t}) = \text{TRUE}$.
$F^b_{(p,\sim p)}$ is $p(\mathbf{t})$.
$(F^b_{(p,\sim p)})^*(q, \mathbf{d})$ is $q(\mathbf{t})$.
By (14), $\mathcal{I}' \models b(\mathbf{t}) = \text{TRUE}$. By definition of a, $\mathcal{I}' \models a(\mathbf{t}) = \text{TRUE}$.

Case 4: F is $G \odot H$ where $\odot \in \{\wedge, \vee\}$.
By I.H. on G and H.

Case 5: F is $G \rightarrow H$.
By I.H. on G and H.

Case 6: F is $Q\mathbf{x}G(\mathbf{x})$ where $Q \in \{\forall, \exists\}$.
By I.H. on G. ∎

Corollary 2 *Let* $\mathbf{c}$ *be a set of predicate and function constants, let* b *be a function constant, and let* F *be a* $(b, \mathbf{c})$*-plain sentence such that every atomic formula containing* b *has the form* $b(\mathbf{t}) = \text{TRUE}$ *or* $b(\mathbf{t}) = \text{FALSE}$*. (a) A coherent interpretation* I *of the signature of* F *is a model of* $\text{SM}[F \wedge BC_b;\ b, \mathbf{c}]$ *iff* $I^b_{(p,\sim p)}$ *is a model of* $\text{SM}[F^b_{(p,\sim p)};\ p, \sim p, \mathbf{c}]$*. (b) An interpretation* J *of the signature of* $F^b_{(p,\sim p)}$ *is a model of* $\text{SM}[F^b_{(p,\sim p)}; p, \sim p, \mathbf{c}]$ *iff* $J = I^b_{(p,\sim p)}$ *for some model* I *of* $\text{SM}[F \wedge BC_b; b, \mathbf{c}]$.

Proof.

For two interpretations I of signature σ_1 and J of signature σ_2, by $I \cup J$ we denote the interpretation of signature $\sigma_1 \cup \sigma_2$ and universe $|I| \cup |J|$ that interprets all symbols occurring only in σ_1 in the same way I does and similarly for σ_2 and J. For symbols appearing in both σ_1 and σ_2, I must interpret these the same as J does, in which case $I \cup J$ also interprets the symbol in this way.

(a$\Rightarrow$) Assume $I \models \text{SM}[F; b, \mathbf{c}] \wedge (\text{TRUE} \neq \text{FALSE})$. Since $I \models \text{TRUE} \neq \text{FALSE}$, $I \cup I^b_p \models \text{TRUE} \neq \text{FALSE}$ since by definition of I^b_p, I and I^b_p share the same universe. By definition of I^b_p, $I \cup I^b_p \models (14)$. Since we assume $I \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$, it follows that $I \models BF_b$ which further means that $I \models BC_b$ and so $I \cup I^b_p \models BC_b$. Thus by Theorem 4, $I \cup I^b_p \models \text{SM}[F \wedge BF_b; b, \mathbf{c}] \leftrightarrow \text{SM}[(F \wedge BF_b)^b_p; p, \sim p, \mathbf{c}]$.

Since we assume $I \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$, it is the case that $I \cup I^b_p \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$ and thus it must be the case that $I \cup I^b_p \models \text{SM}[(F \wedge BF_b)^b_p; p, \sim p, \mathbf{c}]$. However since the signature of I does not contain p, we conclude $I^b_p \models \text{SM}[(F \wedge BF_b)^b_p; p, \sim p, \mathbf{c}]$.

(a$\Leftarrow$) Assume $I \models \text{TRUE} \neq \text{FALSE}$ and $I^b_p \models \text{SM}[(F \wedge BF_b)^b_p; p, \sim p, \mathbf{c}]$. Since $I \models \text{TRUE} \neq \text{FALSE}$, $I \cup I^b_p \models \text{TRUE} \neq \text{FALSE}$ since by definition of I^b_p, I and I^b_p share the same universe. By definition of I^b_p, $I \cup I^b_p \models (14)$. Therefore, since $I^b_p \models (BF_b)^b_p$, it follows that $I \models BF_b$ and thus, $I \cup I^b_p \models BC_b$. Thus by Theorem 4, $I \cup I^b_p \models \text{SM}[F \wedge BF_b; b, \mathbf{c}] \leftrightarrow \text{SM}[(F \wedge BF_b)^b_p; p, \sim p, \mathbf{c}]$.

Since we assume $I_p^b \models \text{SM}[(F \wedge BF_b)_p^b; p, \sim p, \mathbf{c}]$, it is the case that $I \cup I_p^b \models \text{SM}[(F \wedge BF_b)_p^b; p, \sim p, \mathbf{c}]$ and thus it must be the case that $I \cup I_p^b \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$. Therefore since the signature of I_p^b does contain b, we conclude $I \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$.

(b$\Rightarrow$) Assume $J \models \text{TRUE} \neq \text{FALSE}$ and $J \models \text{SM}[(F \wedge BF_b)_p^b; p\mathbf{c}]$. Let $I = J_b^p$ where J_b^p denotes the interpretation of the signature of F obtained from J by replacing the predicate p with the boolean function b such that
$b^I(\xi_1, \dots, \xi_k) = \text{TRUE}$ for all tuples such that $I \models p^I(\xi_1, \dots, \xi_k)$,
$b^I(\xi_1, \dots, \xi_k) = \text{FALSE}$ for all tuples such that $I \models \sim p^I(\xi_1, \dots, \xi_k)$. Since $J \models (BF_b)_p^b$, this is a well-defined function.
Clearly, $J = I_p^b$ so it only remains to be shown that $I \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$.

Since I and J have the same universe and $J \models \text{TRUE} \neq \text{FALSE}$, it follows that $I \cup J \models \text{TRUE} \neq \text{FALSE}$. Also by the definition of J_b^p $I \cup J \models (14)$. Also, since $J \models (BF_b)_p^b$, it follows that $I \models BF_b$ and thus, $I \cup J \models BC_b$. Thus by Theorem 4, $I \cup J \models \text{SM}[F \wedge BF_b; b, \mathbf{c}] \leftrightarrow \text{SM}[(F \wedge BF_b)_p^b \wedge CC_p; p\mathbf{c}]$

Since we assume $J \models \text{SM}[(F \wedge BF_b)_p^b; p, \sim p, \mathbf{c}]$, it is the case that $I \cup J \models \text{SM}[(F \wedge BF_b)_p^b; p, \sim p, \mathbf{c}]$ and thus it must be the case that $I \cup J \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$. Now since the signature of J does not contain b, we conclude $I \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$.

(b$\Leftarrow$)Take any I such that $J = I_p^b$ and $I \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$. Since $J \models \text{TRUE} \neq \text{FALSE}$ and I and J share the same universe, $I \cup J \models \text{TRUE} \neq \text{FALSE}$. By definition of $J = I_p^b$, $I \cup J \models (14)$. Since we assume $I \models \text{SM}[F \wedge BF_b; b, \mathbf{c}]$, it follows that $I \models BF_b$ which further means that $I \models BC_b$ and so $I \cup I_p^b \models BF_b$. Thus by Theorem 4, $I \cup J \models \text{SM}[F \wedge BF_b; b, \mathbf{c}] \leftrightarrow \text{SM}[(F \wedge BF_b)_p^b; \sim p, p, \mathbf{c}]$

Since we assume $I \models \text{SM}[F; b, \mathbf{c}]$, it is the case that $I \cup J \models \text{SM}[F; b, \mathbf{c}]$ and thus it must be the case that $I \cup J \models \text{SM}[F_{(p,\sim p)}^b \wedge CC_p; p, \sim p, \mathbf{c}]$. However since the signature of I does not contain p, we conclude $J \models \text{SM}[F_{(p,\sim p)}^b \wedge CC_p; p, \sim p, \mathbf{c}]$. ∎

A.5 Proof of Theorem 5 and Corollary 3

Theorem 5 *For any f-plain formula F,*

$$\begin{array}{l}\forall \mathbf{x} y\big((f(\mathbf{x}) = y \leftrightarrow b(\mathbf{x}, y) = \text{TRUE}) \\ \quad \wedge (f(\mathbf{x}) \neq y \leftrightarrow b(\mathbf{x}, y) = \text{FALSE})\big)\end{array} \tag{21}$$

and $\exists xy(x \neq y)$ entail

$$\text{SM}[F; f, \mathbf{c}] \;\leftrightarrow\; \text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]\,.$$

Proof. For any interpretation $\mathcal{I} = \langle I, X \rangle$ of signature $\sigma \supseteq \{f, b, \mathbf{c}\}$ satisfying (21), it is clear that $\mathcal{I} \models F$ iff $\mathcal{I} \models F_b^f$ since F_b^f is simply the result of replacing all $f(\mathbf{x}) = y$ with $b(\mathbf{x}, y) = \text{TRUE}$. Thus it only remains to be shown that $\mathcal{I} \models \neg\exists \widehat{f}, \widehat{\mathbf{c}}((\widehat{f}, \widehat{\mathbf{c}} < f, \mathbf{c}) \wedge F^*(\widehat{f}, \widehat{\mathbf{c}}))$ iff $\mathcal{I} \models \neg\exists \widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^f \wedge UE_b)^*(\widehat{b}\widehat{\mathbf{c}}))$ or equivalently, $\mathcal{I} \models \exists \widehat{f}\widehat{\mathbf{c}}((\widehat{f}, \widehat{\mathbf{c}} < f, \mathbf{c}) \wedge F^*(\widehat{f}, \widehat{\mathbf{c}}))$ iff $\mathcal{I} \models \exists \widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^f)^*(\widehat{b}, \widehat{\mathbf{c}}) \wedge (UE_b)^*(\widehat{b}, \widehat{\mathbf{c}}))$.

($\Rightarrow$) Assume $\mathcal{I} \models \exists \widehat{f}, \widehat{\mathbf{c}}((\widehat{f}, \widehat{\mathbf{c}} < f, \mathbf{c}) \wedge F^*(\widehat{f}, \widehat{\mathbf{c}}))$. We wish to show that $\mathcal{I} \models \exists \widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^f)^*(\widehat{b}, \widehat{\mathbf{c}}) \wedge (UE_b)^*(\widehat{b}, \widehat{\mathbf{c}}))$

That is, take any function g of the same arity as f and any list of predicates and functions $\mathbf{d}$ of the same length $\mathbf{c}$. Now let $\mathcal{I}' = \langle I \cup J^{(b,f,\mathbf{c})}_{(a,g,\mathbf{d})}, X \cup Y^{\mathbf{c}}_{\mathbf{d}} \rangle$ be from an extended signature $\sigma' = \sigma \cup \{g, a, \mathbf{d}\}$ where J is an interpretation of functions from the signature σ and I and J agree on all symbols not occurring in $\{b, f, \mathbf{c}\}$. $J^{(b,f,\mathbf{c})}_{(a,g,\mathbf{d})}$ denotes the interpretation from $\sigma^{(b,f,\mathbf{c})}_{(a,g,\mathbf{d})}$ (the signature obtained from σ by replacing f with g, b with a, and all elements of $\mathbf{c}$ with all elements of $\mathbf{d}$) obtained from the interpretation J by replacing f with g, b with a, and the functions in $\mathbf{c}$ with the corresponding functions in $\mathbf{d}$. Similarly, $Y^{\mathbf{c}}_{\mathbf{d}}$ is the interpretation from σ' obtained from the interpretation Y by replacing predicates from $\mathbf{c}$ by the corresponding predicates from $\mathbf{d}$. We assume

$$\mathcal{I}' \models (g, \mathbf{d} < f, \mathbf{c} \wedge F^*(g, \mathbf{d}))$$

and wish to show that there is a (boolean) function a of the same arity as b such that

$$\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c} \wedge (F_b^f)^*(a, \mathbf{d}) \wedge (UE_b)^*(a, \mathbf{d})).$$

We define the new function a in terms of g as follows:

$$a^{\mathcal{I}'}(\boldsymbol{\xi}, \xi') = \begin{cases} \text{TRUE} & \text{if } \mathcal{I}' \models g(\boldsymbol{\xi}) = \xi' \\ \text{FALSE} & \text{otherwise.} \end{cases}$$

We first show $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$:

Case 1: $\mathcal{I}' \models \forall \mathbf{x} y (f(\mathbf{x}) = y \leftrightarrow g(\mathbf{x}) = y)$.
In this case it then must be the case that $\mathcal{I}' \models \mathbf{d} \neq \mathbf{c}$. Thus it follows that $\mathcal{I}' \models a, \mathbf{d} \neq b, \mathbf{c}$. Consequently we conclude that

$$\mathcal{I}' \models (\mathbf{d}^{pred} \leq \mathbf{c}^{pred}) \wedge a, \mathbf{d} \neq b, \mathbf{c}$$

or simply, $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.

Case 2: $\mathcal{I}' \models \neg \forall \mathbf{x} y (f(\mathbf{x}) = y \leftrightarrow g(\mathbf{x}) = y)$.
In this case it then must be the case that for some list of object names $\mathbf{t}$ and some c that $\mathcal{I}' \models f(\mathbf{t}) = c \wedge g(\mathbf{t}) \neq c$. By the definition of a, this means that $a(\mathbf{t}, c)^{\mathcal{I}'} = \text{FALSE}$ but by (21), $b(\mathbf{t}, c)^{\mathcal{I}'} = \text{TRUE}$. Therefore, $\mathcal{I}' \models a \neq b$ and thus $\mathcal{I}' \models a, \mathbf{d} \neq b, \mathbf{c}$. Consequently we conclude

$$\mathcal{I}' \models (\mathbf{d}^{pred} \leq \mathbf{c}^{pred}) \wedge a, \mathbf{d} \neq b, \mathbf{c}$$

or simply, $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$.

We now show by induction that $\mathcal{I}' \models (F_b^f)^*(a, \mathbf{d})$:

Case 1: F is an atomic formula not containing f.
F_b^f is exactly F thus $F^*(g, \mathbf{d})$ is exactly $(F_b^f)^*(a, \mathbf{d})$ so certainly the claim holds.

Case 2: F is $f(\mathbf{t}) = c$, where $\mathbf{t}$ contains no intensional functions.
$F^*(g, \mathbf{d})$ is $f(\mathbf{t}) = c \wedge g(\mathbf{t}) = c$.

F_b^f is $b(\mathbf{t}, c) = \text{TRUE}$.
$(F_b^f)^*(a, \mathbf{d})$ is $b(\mathbf{t}, c) = \text{TRUE} \wedge a(\mathbf{t}, c) = \text{TRUE}$.
Since $\mathcal{I}' \models F^*(g, \mathbf{d})$, by the definition of a, $\mathcal{I}' \models a(\mathbf{t}, c) = \text{TRUE}$ and from (21) it follows that $\mathcal{I}' \models b(\mathbf{t}, c) = true$.

Case 3: F is $f(\mathbf{t}) = c$, where $\mathbf{t}$ contains at least one intensional function.
$F^*(g, \mathbf{d})$ is $f(\mathbf{t}) = c \wedge g(\mathbf{t}^{f,\mathbf{c}}_{(g,\mathbf{d})}) = c$.
F_b^f is $b(\mathbf{t}, c) = \text{TRUE}$.
$(F_b^f)^*(a, \mathbf{d})$ is $b(\mathbf{t}, c) = \text{TRUE} \wedge a(\mathbf{t}^{(f,\mathbf{c})}_{(g,\mathbf{d})}, c) = \text{TRUE}$.
Since $\mathcal{I}' \models F^*(g, \mathbf{d})$, by the definition of a, $\mathcal{I}' \models a(\mathbf{t}^{(f,\mathbf{c})}_{(g,\mathbf{d})}, c) = \text{TRUE}$ and from (21) it follows that $\mathcal{I}' \models b(\mathbf{t}, c) = true$.

Case 4: F is $G \odot H$ where $\odot \in \{\wedge, \vee\}$.
By I.H. on G and H.

Case 5: F is $G \rightarrow H$.
By I.H. on G and H.

Case 6: F is $Q\mathbf{x}G(\mathbf{x})$ where $Q \in \{\forall, \exists\}$.
By I.H. on G.

We now show that $\mathcal{I}' \models (UE_b)^*(a, \mathbf{d})$:
$(UE_b)^*(a, \mathbf{d})$ is equivalent to the following 2 formulas:

$$\begin{aligned}&\forall \mathbf{x}yz(y \neq z \wedge (b(\mathbf{x}, y) = \text{TRUE} \rightarrow b(\mathbf{x}, z) = \text{FALSE}) \wedge \\ &(b(\mathbf{x}, y) = \text{TRUE} \wedge a(\mathbf{x}, y) = \text{TRUE} \rightarrow \\ &b(\mathbf{x}, z) = \text{FALSE} \wedge a(\mathbf{x}, z) = \text{FALSE}))\end{aligned} \tag{15*}$$

$$\forall \mathbf{x} \exists y (b(\mathbf{x}, y) = \text{TRUE})(16^*)$$

(16^*) follows from (21). The first implication of (15^*) follows from (21) . All that remains to show is that the second implication of (15^*) is satisfied by $\mathcal{I}'$.

Now, take any vector of terms $\mathbf{t_x}$ the same length as $\mathbf{x}$ and any terms t_y and t_z. If $\mathcal{I}' \not\models b(\mathbf{t_x}, t_y) = \text{TRUE} \wedge a(\mathbf{t_x}, t_y) = \text{TRUE}$, then the second implication of (15^*) is satisfied by $\mathcal{I}'$. If instead $\mathcal{I}' \models b(\mathbf{t_x}, t_y) = \text{TRUE} \wedge a(\mathbf{t_x}, t_y) = \text{TRUE}$ then by (21) , $\mathcal{I}' \models b(\mathbf{t_x}, t_z) = \text{FALSE}$ for $t_y \neq t_z$ and by definition of a, $\mathcal{I}' \models a(\mathbf{t_x}, t_z) = \text{FALSE}$ for $t_y \neq t_z$ so in this case too, the second implication of (15^*) is satisfied by $\mathcal{I}'$.

($\Leftarrow$) Assume $\mathcal{I} \models \exists \widehat{b}, \widehat{\mathbf{c}}((\widehat{b}, \widehat{\mathbf{c}} < b, \mathbf{c}) \wedge (F_b^f)^*(\widehat{b}, \widehat{\mathbf{c}}) \wedge (UE_b)^*(\widehat{b}, \widehat{\mathbf{c}}))$. We wish to show that $\mathcal{I} \models \exists \widehat{f}, \widehat{\mathbf{c}}((\widehat{f}, \widehat{\mathbf{c}} < f, \mathbf{c}) \wedge F^*(\widehat{f}, \widehat{\mathbf{c}}))$

That is, take any (boolean) function a of the same arity as b and any list of predicates and functions $\mathbf{d}$ the same length as $\mathbf{c}$ and let $\mathcal{I}' = \langle I \cup J^{(b,f,\mathbf{c})}_{(a,g,\mathbf{d})}, X \cup Y^{\mathbf{c}}_{\mathbf{d}} \rangle$ be defined as before. We assume

$$\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c} \wedge (F_b^f)^*(a, \mathbf{d}) \wedge (UE_b)^*(a, \mathbf{d}))$$

and wish to show that there is a function g of the same arity as f such that

$$\mathcal{I}' \models (g, \mathbf{d} < f, \mathbf{c} \wedge F^*(g, \mathbf{d})).$$

We define the new function g in terms of a as follows:

$$g^{\mathcal{I}'}(\boldsymbol{\xi}) = \begin{cases} f(\boldsymbol{\xi}) & \text{if } \mathcal{I}' \models a(\boldsymbol{\xi}, f(\boldsymbol{\xi})) = \text{TRUE} \\ m(f(\boldsymbol{\xi})) & \text{otherwise} \end{cases}$$

where m is a mapping from the universe to itself such that $\forall x(m(x) \neq x)$. Note that the assumption that there are at least two elements in the universe is essential to this definition.

We first show $\mathcal{I}' \models (g, \mathbf{d} < f, \mathbf{c})$:

Case 1: $\mathcal{I}' \models \forall \mathbf{x} y(b(\mathbf{x}, y) = a(\mathbf{x}, y))$.
In this case, $\mathcal{I}' \models (a = b)$ so for it to be the case that $\mathcal{I}' \models (a, \mathbf{d} < b, \mathbf{c})$, it must be that $\mathcal{I}' \models \neg(\mathbf{c} = \mathbf{d})$. It then follows that $\mathcal{I}' \models \neg(f, \mathbf{c} = g, \mathbf{d})$. Consequently in this case, $\mathcal{I}' \models ((g, \mathbf{d})^{pred} \leq (f, \mathbf{c})^{pred}) \wedge \neg(f, \mathbf{c} = g, \mathbf{d})$ or simply $\mathcal{I}' \models (g, \mathbf{d} < f, \mathbf{c})$.

Case 2: $\mathcal{I}' \models \neg\forall \mathbf{x} y(b(\mathbf{x}, y) = a(\mathbf{x}, y))$.
Thus, for some $\boldsymbol{\xi}$, $\mathcal{I}' \models \neg\forall y(b(\boldsymbol{\xi}, y) = a(\boldsymbol{\xi}, y))$. Further, for some ξ', $b(\boldsymbol{\xi}, \xi') \neq a(\boldsymbol{\xi}, \xi')$. Then from $\mathcal{I}' \models UE_b^*(a, \mathbf{d})$, it follows that for some ξ', $b(\boldsymbol{\xi}, \xi') = \text{TRUE} \wedge a(\boldsymbol{\xi}, \xi') = \text{FALSE}$. This is because $\mathcal{I}' \models UE_b^*(a, \mathbf{d})$ means that there must be some ξ' for which $b(\boldsymbol{\xi}, \xi') = \text{TRUE}$ and since if $b(\boldsymbol{\xi}, \xi') = \text{TRUE} \wedge a(\boldsymbol{\xi}, \xi') = \text{TRUE}$, then, $a = b$, which we assume not to be the case. Thus by definition of g, $\mathcal{I}' \models g(\boldsymbol{\xi}) = m(f(\boldsymbol{\xi}))$. And since $m(f(\boldsymbol{\xi}) \neq f(\boldsymbol{\xi}$, we conclude $\mathcal{I}' \models \neg(g = f)$. It then follows that $\mathcal{I}' \models \neg(f, \mathbf{c} = g, \mathbf{d})$. Consequently in this case, $\mathcal{I}' \models ((g, \mathbf{d})^{pred} \leq (f, \mathbf{c})^{pred}) \wedge \neg(f, \mathbf{c} = g, \mathbf{d})$ or simply $\mathcal{I}' \models (g, \mathbf{d} < f, \mathbf{c})$.

We now show by induction that $\mathcal{I}' \models F^*(g, \mathbf{d})$:

Case 1: F is an atomic formula not containing f.
F_b^f is exactly F thus $F^*(g, \mathbf{d})$ is exactly $(F_b^f)^*(a, \mathbf{d})$ so certainly the claim holds.

Case 2: F is $f(\mathbf{t}) = c$, where $\mathbf{t}$ contains no intensional functions.
$F^*(g, \mathbf{d})$ is $f(\mathbf{t}) = c \wedge g(\mathbf{t}) = c$.
F_b^f is $b(\mathbf{t}, c) = \text{TRUE}$.
$(F_b^f)^*(a, \mathbf{d})$ is $a(\mathbf{t}, c) = \text{TRUE} \wedge b(\mathbf{t}, c) = \text{TRUE}$.
Since $\mathcal{I}' \models (F_b^f)^*(a, \mathbf{d})$, it follows from the definition of g that $g(\mathbf{t}) = f(\mathbf{t})$. From (21), it follows that $\mathcal{I}' \models f(\mathbf{t}) = c$ and so it must be that $\mathcal{I}' \models g(\mathbf{t}) = c$, from which we conclude $\mathcal{I}' \models F^*(g, \mathbf{d})$.

Case 3: F is $f(\mathbf{t}) = c$, where $\mathbf{t}$ contains at least one intensional function.
$F^*(g, \mathbf{d})$ is $f(\mathbf{t}) = c \wedge g(\mathbf{t}_{(g,\mathbf{d})}^{(f,\mathbf{c})}) = c$.
F_b^f is $b(\mathbf{t}, c) = \text{TRUE}$.
$(F_b^f)^*(a, \mathbf{d})$ is $a(\mathbf{t}_{(g,\mathbf{d})}^{(f,\mathbf{c})}, c) = \text{TRUE} \wedge b(\mathbf{t}, c) = \text{TRUE}$.
Since $\mathcal{I}' \models (F_b^f)^*(a, \mathbf{d})$, it follows from the definition of g that $g(\mathbf{t}_{(g,\mathbf{d})}^{(f,\mathbf{c})}) = f(\mathbf{t})$. From (21), it follows that $\mathcal{I}' \models f(\mathbf{t}) = c$ and so it must be that $\mathcal{I}' \models g(\mathbf{t}_{(g,\mathbf{d})}^{(f,\mathbf{c})}) = c$, from which we conclude $\mathcal{I}' \models F^*(g, \mathbf{d})$.

Case 4: F is $G \odot H$ where $\odot \in \{\wedge, \vee\}$.
By I.H. on G and H.

Case 5: F is $G \to H$.
By I.H. on G and H.

Case 6: F is $Q\mathbf{x}G(\mathbf{x})$ where $Q \in \{\forall, \exists\}$.
By I.H. on G. ∎

Corollary 3 *Let F be an f-plain sentence. (a) An interpretation I of the signature of F that satisfies $\exists xy(x \neq y)$ is a model of $\mathrm{SM}[F; f, \mathbf{c}]$ iff I_b^f is a model of $\mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$. (b) An interpretation J of the signature of F_b^f that satisfies $\exists xy(x \neq y)$ is a model of $\mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$ iff $J = I_b^f$ for some model I of $\mathrm{SM}[F; f, \mathbf{c}]$.*

Proof. For two interpretations I of signature σ_1 and J of signature σ_2, by $I \cup J$ we denote the interpretation of signature $\sigma_1 \cup \sigma_2$ and universe $|I| \cup |J|$ that interprets all symbols occurring only in σ_1 in the same way I does and similarly for σ_2 and J. For symbols appearing in both σ_1 and σ_2, I must interpret these the same as J does, in which case $I \cup J$ also interprets the symbol in this way.

(a$\Rightarrow$) Assume $I \models \mathrm{SM}[F; f, \mathbf{c}] \wedge \exists xy(x \neq y)$. Since $I \models \exists xy(x \neq y)$, $I \cup I_b^f \models \exists xy(x \neq y)$ since by definition of I_b^f, I and I_b^f share the same universe. By definition of I_b^f, $I \cup I_b^f \models (21)$. Thus by Theorem 5, $I \cup I_b^f \models \mathrm{SM}[F; f, \mathbf{c}] \leftrightarrow \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$.

Since we assume $I \models \mathrm{SM}[F; f, \mathbf{c}]$, it is the case that $I \cup I_b^f \models \mathrm{SM}[F; f, \mathbf{c}]$ and thus it must be the case that $I \cup I_b^f \models \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$. However since the signature of I does not contain b, we conclude $I_b^f \models \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$.

(a$\Leftarrow$) Assume $I \models \exists xy(x \neq y)$ and $I_b^f \models \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$. Since $I \models \exists xy(x \neq y)$, $I \cup I_b^f \models \exists xy(x \neq y)$ since by definition of I_b^f, I and I_b^f share the same universe. By definition of I_b^f, $I \cup I_b^f \models (21)$. Thus by Theorem 5, $I \cup I_b^f \models \mathrm{SM}[F; f, \mathbf{c}] \leftrightarrow \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$.

Since we assume $I_b^f \models \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$, it is the case that $I \cup I_b^f \models \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$ and thus it must be the case that $I \cup I_b^f \models \mathrm{SM}[F; f, \mathbf{c}]$. Therefore since the signature of I_b^f does contain f, we conclude $I \models \mathrm{SM}[F; f, \mathbf{c}]$.

(b$\Rightarrow$) Assume $J \models \exists xy(x \neq y)$ and $J \models \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$. Let $I = J_f^b$ where J_f^b denotes the interpretation of the signature of F obtained from J by replacing the boolean function b with the function f such that $f^I(\xi_1, \dots, \xi_k) = \xi_{k+1}$ for all tuples such that $b^I(\xi_1, \dots, \xi_k, \xi_{k+1}) = \text{TRUE}$. This is a valid definition of a function since we assume $J \models \mathrm{SM}[F_p^f \wedge UE_b; b, \mathbf{c}]$, from which we obtain $J \models UE_b$. Clearly, $J = I_b^f$ so it only remains to be shown that $I \models \mathrm{SM}[F; f, \mathbf{c}]$.

Since I and J have the same universe and $J \models \exists xy(x \neq y)$, it follows that $I \cup J \models \exists xy(x \neq y)$. Also by the definition of J_f^b $I \cup J \models (21)$. Thus by Theorem 5, $I \cup J \models \mathrm{SM}[F; f, \mathbf{c}] \leftrightarrow \mathrm{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$

Since we assume $J \models \text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$, it is the case that $I \cup J \models \text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$ and thus it must be the case that $I \cup J \models \text{SM}[F; f, \mathbf{c}]$. Now since the signature of J does not contain f, we conclude $I \models \text{SM}[F; f, \mathbf{c}]$.

(b$\Leftarrow$)Take any I such that $J = I_b^f$ and $I \models \text{SM}[F; f, \mathbf{c}]$. Since $J \models \exists xy(x \neq y)$ and I and J share the same universe, $I \cup J \models \exists xy(x \neq y)$. By definition of $J = I_b^f$, $I \cup J \models (21)$. Thus by Theorem 5, $I \cup J \models \text{SM}[F; f, \mathbf{c}] \leftrightarrow \text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$

Since we assume $I \models \text{SM}[F; f, \mathbf{c}]$, it is the case that $I \cup J \models \text{SM}[F; f, \mathbf{c}]$ and thus it must be the case that $I \cup J \models \text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$. However since the signature of I does not contain b, we conclude $J \models \text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$. ∎

A.6 Proof of Theorem 6 and Corollary 4

Corollary 6 *For any $(f, \mathbf{c})$-plain formula F, formulas*

$$\forall \mathbf{x}y(f(\mathbf{x}) = y \leftrightarrow p(\mathbf{x}, y)), \tag{22}$$

$$\forall \mathbf{x}y(f(\mathbf{x}) \neq y \leftrightarrow \sim p(\mathbf{x}, y)), \tag{23}$$

$$\exists xy(x \neq y)$$

entail

$$\text{SM}[F;\ f, \mathbf{c}] \leftrightarrow \text{SM}[F_p^f \wedge UE_p;\ p, \sim p, \mathbf{c}]\ .$$

Proof. Let b be a function such that the following hold:

$$\forall \mathbf{x}y(f(\mathbf{x}) = y \leftrightarrow b(\mathbf{x}, y) = \text{TRUE})$$

$$\forall \mathbf{x}y(f(\mathbf{x}) \neq y \leftrightarrow b(\mathbf{x}, y) = \text{FALSE})$$

$$\forall \mathbf{x}y(b(\mathbf{x}, y) = \text{TRUE} \leftrightarrow p(\mathbf{x}, y))$$

$$\forall \mathbf{x}y(b(\mathbf{x}, y) = \text{FALSE} \leftrightarrow \sim p(\mathbf{x}, y))$$

Note that such a definition is only valid for interpretations satisfying (22) and (22), which we assume to be the case. Now, by Theorem 5, $\text{SM}[F; f, \mathbf{c}] \leftrightarrow \text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}]$. By Theorem 5, $\text{SM}[F_b^f \wedge UE_b; b, \mathbf{c}] \leftrightarrow \text{SM}[(F_b^f \wedge UE_b)_p^b; \sim p, p, \mathbf{c}]$. Thus, it follows that $\text{SM}[F; f, \mathbf{c}] \leftrightarrow \text{SM}[(F_b^f \wedge UE_b)_p^b; \sim p, p, \mathbf{c}]$.

It only remains to show that $\text{SM}[(F_b^f \wedge UE_b)_p^b; \sim p, p, \mathbf{c}]$ is precisely $\text{SM}[F_p^f \wedge UE_{\sim p}; \sim p, p, \mathbf{c}]$. $(F_b^f)_p^b$ first replaces f in the signature with b, replaces all $f(\mathbf{t}) = c$ with $b(\mathbf{t}, c) = \text{TRUE}$, The composed translation then replaces b with p and $\sim p$ in the signature, replaces all $b(\mathbf{t}, c) = \text{TRUE}$ with $p(\mathbf{t}, c)$, and replaces all $b(\mathbf{t}, c) = \text{FALSE}$ with $\sim p(\mathbf{t}, c)$ (however F_b^f does not contain $b(\mathbf{t}, c) = \text{FALSE}$; only UE_b contains $b(\mathbf{t}, c) = \text{FALSE}$). This part is equivalent to F_p^f. It is easy to see that $(UE_b)_p^b$ is $UE_{\sim p}$.
∎

Corollary 4 *Let F be an $(f, \mathbf{c})$-plain sentence. (a) An interpretation I of the signature of F that satisfies $\exists xy(x \neq y)$ is a model of $\text{SM}[F;\ f, \mathbf{c}]$ iff $I_{(p,\sim p)}^f$ is a model of*

SM$[F_p{}^f \wedge UE_p;\ p, {\sim}p, \mathbf{c}]$. *(b) An interpretation J of the signature of F_p^f that satisfies $\exists xy(x \neq y)$ is a model of* SM$[F_p^f \wedge UE_p;\ p, {\sim}p, \mathbf{c}]$ *iff $J = I^f_{(p,{\sim}p)}$ for some model I of* SM$[F;\ f, \mathbf{c}]$.

Proof. For two interpretations I of signature σ_1 and J of signature σ_2, by $I \cup J$ we denote the interpretation of signature $\sigma_1 \cup \sigma_2$ and universe $|I| \cup |J|$ that interprets all symbols occurring only in σ_1 in the same way I does and similarly for σ_2 and J. For symbols appearing in both σ_1 and σ_2, I must interpret these the same as J does, in which case $I \cup J$ also interprets the symbol in this way.

(a$\Rightarrow$) Assume $I \models$ SM$[F; b, \mathbf{c}] \wedge \exists xy(x \neq y)$. Since $I \models \exists xy(x \neq y)$, $I \cup I^f_{\sim pp} \models \exists xy(x \neq y)$ since by definition of $I^f_{\sim pp}$, I and $I^f_{\sim pp}$ share the same universe. By definition of $I^f_{\sim pp}$, $I \cup I^f_{\sim pp} \models (22) \wedge (23)$. Thus by Theorem 6, $I \cup I^f_{\sim pp} \models$ SM$[F; f, \mathbf{c}] \leftrightarrow$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$.

Since we assume $I \models$ SM$[F; f, \mathbf{c}]$, it is the case that $I \cup I^f_{\sim pp} \models$ SM$[F; f, \mathbf{c}]$ and thus it must be the case that $I \cup I^f_{\sim pp} \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$. However since the signature of I does not contain p or ${\sim}p$, we conclude $I^f_{\sim pp} \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$.

(a$\Leftarrow$) Assume $I \models \exists xy(x \neq y)$ and $I^f_{\sim pp} \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$. Since $I \models \exists xy(x \neq y)$, $I \cup I^f_{\sim pp} \models \exists xy(x \neq y)$ since by definition of $I^f_{\sim pp}$, I and $I^f_{\sim pp}$ share the same universe. By definition of $I^f_{\sim pp}$, $I \cup I^f_{\sim pp} \models (22) \wedge (23)$. Thus by Theorem 6, $I \cup I^f_{\sim pp} \models$ SM$[F; f, \mathbf{c}] \leftrightarrow$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$.

Since we assume $I^f_{\sim pp} \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$, it is the case that $I \cup I^f_{\sim pp} \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$ and thus it must be the case that $I \cup I^f_{\sim pp} \models$ SM$[F; f, \mathbf{c}]$. Therefore since the signature of $I^f_{\sim pp}$ does contain f, we conclude $I \models$ SM$[F; f, \mathbf{c}]$.

(b$\Rightarrow$) Assume $J \models \exists xy(x \neq y)$ and $J \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$. Let $I = J_f^{\sim pp}$ where $J_f^{\sim pp}$ denotes the interpretation of the signature of F obtained from J by replacing the predicates p and ${\sim}p$ with the function f such that
$f^I(\xi_1, \dots, \xi_k) = \xi_{k+1}$ for all tuples such that $J \models p^J(\xi_1, \dots, \xi_k, \xi_{k+1})$. Note that this definition of f is well-defined due to the fact that $J \models UE_{\sim pp}$.
Clearly, $J = I^f_{\sim pp}$ so it only remains to be shown that $I \models$ SM$[F; f, \mathbf{c}]$.

Since I and J have the same universe and $J \models \exists xy(x \neq y)$, it follows that $I \cup J \models \exists xy(x \neq y)$. Also by the definition of $J_f^{\sim pp}$, $I \cup J \models (22) \wedge (23)$. Thus by Theorem 6, $I \cup J \models$ SM$[F; f, \mathbf{c}] \leftrightarrow$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$.

Since we assume $J \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$, it is the case that $I \cup J \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$ and thus it must be the case that $I \cup J \models$ SM$[F; f, \mathbf{c}]$. Now since the signature of J does not contain f, we conclude $I \models$ SM$[F; f, \mathbf{c}]$.

(b$\Leftarrow$)Take any I such that $J = I^f_{\sim pp}$ and $I \models$ SM$[F; f, \mathbf{c}]$. Since $J \models \exists xy(x \neq y)$ and I and J share the same universe, $I \cup J \models \exists xy(x \neq y)$. By definition of $J = I^f_{\sim pp}$, $I \cup J \models (22) \wedge (23)$. Thus by Theorem 6, $I \cup J \models$ SM$[F; f, \mathbf{c}] \leftrightarrow$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$.

Since we assume $I \models$ SM$[F; f, \mathbf{c}]$, it is the case that $I \cup J \models$ SM$[F; f, \mathbf{c}]$ and thus it must be the case that $I \cup J \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$. However since the signature of I does not contain p or ${\sim}p$, we conclude $J \models$ SM$[F_p^f \wedge UE_{\sim p}; {\sim}p, p, \mathbf{c}]$. ∎

A.7 Proof of Theorem 7

Theorem 7 *For any two-valued program Π, an interpretation I is a stable model of Π in the sense of [Lifschitz, 2012] iff I' is a stable model of tv2sm(Π) in the sense of [Bartholomew and Lee, 2012].*

Proof. Given a two-valued program Π and a multi-valued interpretation I, we begin by showing that $I \models \Pi^I$ iff $I' \models T(\Pi)^{I'}$. Consider a rule R of the form (19). There are three cases–either $I \models F$ and $I \models R^I$, $I \models F$ and $I \not\models R^I$ or $I \not\models F$:

- $I \models F$ and $I \models R^I$.
  In this case, the reduct R^I is $L_0 \leftarrow L_1, \ldots, L_n$ and since $I \models R^I$, then either $I \models L_0$ or there is some $L_i \in \{L_1, \ldots, L_n\}$ such that $I \not\models L_i$. If $I \models L_0$, then by definition of I', $I' \models T(L_0)$ and thus the reduct $T(R)^{I'}$ is $body \rightarrow T(L_0)$ which is satisfied by I' no matter what $body$ happens to be. If on the other than $I \not\models L_i$ for some $L_i \in \{L_1, \ldots, L_n\}$, then the reduct $T(R)^{I'}$ is $\bot \rightarrow head$ since by definition of I', $I' \not\models T(L_i)$ and thus the subformula $T(L_1) \wedge \ldots T(L_i) \wedge \ldots, T(L_n)$ is replaced by $\bot$ and then in this case too, the reduct $T(R)^{I'}$ is satisfied by I'.
- $I \models F$ and $I \not\models R^I$
  In this case, the reduct R^I is $L_0 \leftarrow L_1, \ldots, L_n$ and since $I \not\models R^I$, it must be the case that $I \models L_1, \ldots, L_n$ and $I \not\models L_0$. On the other hand, by definition of I', this means that $I' \models T(L_1) \wedge \cdots \wedge T(L_n)$. Since $I \models F$, $I' \models F$ and thus $I' \models \neg\neg F$. However $I' \not\models T(L_0)$ so the entire rule is a maximal subformula not satisfied by I' so the entire rule becomes $\bot$ in the reduct and thus $I' \not\models T(R)^{I'}$.
- $I \not\models F$
  In this case, the reduct R^I omits this rule entirely so certainly $I \models R^I$. On the other hand, by definition of I', $I' \not\models T(F)$ and thus $I' \not\models \neg\neg T(F)$ and further $I' \not\models \neg\neg T(F) \wedge T(L_1) \wedge \cdots \wedge T(L_n)$. Thus, the reduct $T(R)^{I'}$ is $\bot \rightarrow head$ since the entire body is not satisfied by I' and so $I' \models T(R)^{I'}$ no matter what $head$ happens to be.

■