Stable Models of Formulas with Generalized Quantifiers

Joohyung Lee and Yunsong Meng
School of Computing, Informatics, and Decision Systems Engineering
Arizona State University, Tempe, AZ, USA
(*e-mail:* {joolee, Yunsong.Meng}@asu.edu)

Abstract

Applications of answer set programming motivated various extensions to the stable model semantics, for instance, to incorporate aggregates, to facilitate interface with external information source, such as ontology descriptions, and to integrate with other computing paradigms, such as constraint solving. This paper provides a unifying and reductive view on some of these extensions by viewing them as special cases of formulas with generalized quantifiers under the stable model semantics, an extension of the first-order stable model semantics by Ferraris, Lee and Lifschitz. The general semantics provides a systematic approach to study the individual extensions of the stable model semantics, and even allows them to be combined in a single language. We show that important theorems in answer set programming, such as the splitting theorem, the theorem on completion, and the theorem on strong equivalence, can be naturally extended to formulas with generalized quantifiers, which in turn can be applied to the individual extensions of the stable model semantics.

1 Introduction

Applications of answer set programming motivated various recent extensions to the stable model semantics, for instance, to incorporate aggregates (e.g., (Faber et al. 2011; Ferraris 2005; Son and Pontelli 2007)) and abstract constraint atoms (Marek and Truszczynski 2004), and to facilitate interface with external information source, such as ontology descriptions (Eiter et al. 2008). While the extensions were driven by different motivations and applications, a common underlying issue is to extend the stable model semantics to incorporate "complex atoms," such as aggregates, abstract constraint atoms and dl-atoms.

HEX programs (Eiter et al. 2005) provide an elegant solution to incorporate such different extensions in a uniform framework via "external atoms." The idea is to define the meaning of external atoms in terms of external functions. For example, aggregate $\text{COUNT}\langle x.p(x,a)\rangle \geq 3$ is modelled by a binary external function $f_{\#count}$ such that given an Herbrand interpretation I, $f_{\#count}(I,a,3) = 1$ iff the cardinality of the set $\{c \mid c \in |I|, I \models p(c,a)\}$ is ≥ 3. Once the notion of satisfaction is extended to cover external atoms, the stable models of HEX programs are defined as minimal models of the "FLP-reduct" (Faber et al. 2011). The adoption of the FLP reduct instead of the traditional Gelfond-Lifschitz reduct was a key idea to incorporate external atoms in HEX programs. HEX programs are well studied (Eiter et al. 2006; Eiter et al. 2008; Eiter et al. 2011), and was implemented in the system DLV-HEX.[1]

However, the underlying semantics of HEX programs, the FLP semantics, diverges from the

[1] http://www.kr.tuwien.ac.at/research/systems/dlvhex/

traditional stable model semantics in some essential ways. For example, consider a program

$$p(a) \leftarrow \textit{not}\ \text{COUNT}\langle x.p(x)\rangle < 1, \tag{1}$$

and another program which rewrites the first program as

$$\begin{array}{l} p(a) \leftarrow \textit{not}\ q \\ q \leftarrow \text{COUNT}\langle x.p(x)\rangle < 1, \end{array} \tag{2}$$

where the second rule defines q in terms of COUNT aggregates. One may expect this transformation to modify the collection of answer sets in a "conservative" way. That is, each answer set of (2) is obtained from an answer set of (1) in accordance with the definition of q.[2] However, this is not the case under the FLP stable model semantics: the former has $\emptyset$ as the only FLP answer set while the latter has both $\{p(a)\}$ and $\{q\}$ as the FLP answer sets. [3]

Related to this issue is the anti-chain property that is ensured by the FLP semantics: no FLP answer set is a proper subset of another FLP answer set. This prevents us from allowing choice rules (Niemelä and Simons 2000), which are a useful construct in the "generate-and-test" organization of ASP programming (Lifschitz 2002).

Also, the extensions of the FLP semantics to allow complex formulas in (Truszczyński 2010; Bartholomew et al. 2011) encounter some unintuitive cases. For example, according to the extensions, $\{p\}$ is the FLP answer set of $p \leftarrow p \vee \neg p$, but this has a circular justification.

On the other hand, these issues do not arise with the stable model semantics from (Ferraris 2005; Lee et al. 2008). According to (Ferraris 2005), which defines the semantics of aggregates by reduction to propositional formulas under the stable model semantics, program (1) has $\{p(a)\}$ and $\emptyset$ as the answer sets, and program (2) has $\{p(a)\}$ and $\{q\}$ as the answer sets. According to (Lee et al. 2008), choice rules are also understood in a reductive way. For instance, $\{q(x)\} \leftarrow p(x)$ is identified with $\forall x(p(x) \rightarrow q(x) \vee \neg q(x))$ under the first-order stable model semantics from (Ferraris et al. 2007; Ferraris et al. 2011). In the same paper (Lee et al. 2008), the reductive approach to defining aggregates in (Ferraris 2005) was extended to first-order formulas, but was limited to counting aggregates. The extensions to cover arbitrary aggregates in the first-order case was done in (Lee and Meng 2009; Ferraris and Lifschitz 2010) by extending the first-order stable model semantics to formulas containing aggregate expressions. But even then the semantics does not account for other complex atoms like dl-atoms and external atoms.

So one wonders: is it possible to combine the versatility of HEX programs and the semantic properties of the first-order stable model semantics? This is the subject of this paper.

It is hinted in (Ferraris and Lifschitz 2010) that aggregates may be viewed in terms of generalized quantifiers—a generalizations of the standard quantifiers, $\forall$ and $\exists$, introduced by Mostowski (1957). We follow up on that suggestion, and present an alternative approach to HEX programs by understanding external atoms in terms of generalized quantifiers. Our semantics avoids the above issues with the FLP semantics, and allows natural extensions to several important theorems about the first-order stable model semantics from (Ferraris et al. 2011), such as the splitting theorem, the theorem on completion and the theorem on strong equivalence, to formulas with generalized quantifiers, which in turn can be applied to the individual extensions, such as programs with aggregates, and nonmonotonic dl-programs. This

[2] Indeed, this is what happens in expressing a rule with nested expressions like $p \leftarrow \textit{not not}\ p$ into $p \leftarrow \textit{not}\ q$, $q \leftarrow \textit{not}\ p$ by defining q as *not p*.

[3] See the related discussion in `http://www.cs.utexas.edu/~vl/tag/aggregates`.

saves efforts in re-proving the theorems for these individual cases. It also allows us to combine the individual extensions in a single language as in the following example.

Example 1
We consider an extension of nonmonotonic dl-programs $(\mathcal{T}, \Pi)$ that allows aggregates. For instance, the ontology description $\mathcal{T}$ specifies that every married man has a spouse who is a woman and similarly for married woman:

$$\begin{aligned} \mathit{Man} \sqcap \mathit{Married} &\sqsubseteq \exists \mathit{Spouse.Woman} \\ \mathit{Woman} \sqcap \mathit{Married} &\sqsubseteq \exists \mathit{Spouse.Man}. \end{aligned}$$

The following program Π counts the number of people who are eligible for an insurance discount:

$$\begin{array}{l} \mathit{discount}(x) \leftarrow \mathit{not\ accident}(x), \\ \quad \#\mathit{dl}[\mathit{Man} \uplus \mathit{mm}, \mathit{Married} \uplus \mathit{mm}, \mathit{Woman} \uplus \mathit{mw}, \mathit{Married} \uplus \mathit{mw}; \exists \mathit{Spouse}.\top](x). \\ \mathit{discount}(x) \leftarrow \mathit{discount}(y), \mathit{family}(y,x), \mathit{not\ accident}(x). \\ \mathit{numOfDiscount}(z) \leftarrow \text{COUNT}\langle x.\mathit{discount}(x)\rangle = z. \end{array}$$

The first rule describes that everybody who has a spouse and has no accident is eligible for a discount. The second rule describes that everybody who has no accident and has a family member with a discount is eligible for a discount. We will see that our method can provide the semantics of this combination.

Interestingly, our approach allows us to discover two new extensions of the stable model semantics, yet another semantics of logic programs with abstract constraints, and yet another semantics of nonmonotonic dl-programs, both of which are again special cases of GQ formulas, and, distinct from the previous definitions, are close to the first-order stable model semantics.

The paper is organized as follows. We first review the syntax and the classical semantics of formulas with generalized quantifiers (GQ-formuals). Next we define stable models of formulas with generalized quantifiers and then show the individual extensions of the stable model semantics, such as logic programs with aggregates, abstract constraint atoms, and nonmonotonic dl-atoms, can be viewed as special cases of GQ formulas. We extend important theorems in answer set programming, such as the splitting theorem, the theorem on completion, and the theorem on strong equivalence, to formulas with generalized quantifiers. [4]

2 Preliminary

2.1 Syntax of Formulas with Generalized Quantifiers

We follow the definition of a formula with generalized quantifiers (GQ-formula) from (Westerståhl 2008), Section 5 (that is to say, with Lindström quantifiers (Lindström 1966) without the isomorphism closure condition).

As in first-order logic, a *signature* σ is a set of symbols consisting of *function constants* and *predicate constants*. Each symbol is assigned a nonnegative integer, called the arity. Function constants with arity 0 are called *object constants*, and predicate constants with arity 0 are called *propositional constants*. A *term* is an *object variable* or $f(t_1, \dots, t_n)$, where f is a function

[4] A longer version with all proofs is available at http://peace.eas.asu.edu/joolee/papers/smgq-iclp-long.pdf.

constant in σ of arity n, and t_i are terms. An *atomic formula* is an expression of the form $p(t_1,\dots,t_n)$ or $t_1 = t_2$, where p is a predicate constant in σ of arity n.

We assume a set $\mathbf{Q}$ of symbols for generalized quantifiers. Each symbol in $\mathbf{Q}$ is associated with a tuple of nonnegative integer $\langle n_1,\dots,n_k\rangle$ ($k \geq 0$, and each n_i is ≥ 0), called the *type*. A *formula (with the set* $\mathbf{Q}$ *of generalized quantifiers)* is defined in a recursive way.

- an atomic formula is a formula;
- if $F_1,\dots,F_k$ are formulas and Q is a generalized quantifier of type $\langle n_1,\dots,n_k\rangle$, then
$$Q[\mathbf{x}_1]\dots[\mathbf{x}_k](F_1(\mathbf{x}_1),\dots,F_k(\mathbf{x}_k)) \tag{3}$$
is a formula, where each $\mathbf{x}_i$ ($1 \leq i \leq k$) is a list of distinct object variables whose length is n_i.

We say that an occurrence of a variable x in a formula F is *bound* if it belongs to a subformula of F that has the form $Q[\mathbf{x}_1]\dots[\mathbf{x}_k](F_1(\mathbf{x}_1),\dots,F_k(\mathbf{x}_k))$, where x is in some $\mathbf{x}_i$. Otherwise it is *free*. We say that x is *free* in F if F contains a free occurrence of x. A *sentence* is a formula with no free variables.

We assume that $\mathbf{Q}$ contains a type $\langle\rangle$ quantifier $Q_\bot$, a type $\langle 0\rangle$ quantifier $Q_\neg$, type $\langle 0,0\rangle$ quantifiers $Q_\wedge, Q_\vee, Q_\rightarrow$, and type $\langle 1\rangle$ quantifiers $Q_\forall, Q_\exists$. Each of them corresponds to the standard logical connectives and quantifiers, $\bot, \neg, \wedge, \vee, \rightarrow, \forall, \exists$. These generalized quantifiers will often be written in the familiar form. For example, we write $F\wedge G$ in place of $Q_\wedge[][](F,G)$, and write $\forall x F(x)$ in place of $Q_\forall[x](F(x))$.

2.2 Semantics of Formulas with Generalized Quantifiers

An interpretation I of a signature σ consists of a nonempty set U, called the *universe* of I, and a mapping c^I for each constant c in σ. For each function constant f of σ whose arity is n, f^I is an element of U if n is 0, and is a function from from U^n to U otherwise. For each predicate constant p of σ whose arity is n, p^I is an element of $\{\mathbf{t},\mathbf{f}\}$ if n is 0, and is a function from U^n to $\{\mathbf{t},\mathbf{f}\}$ otherwise. For each generalized quantifier Q of type $\langle n_1,\dots,n_k\rangle$, Q^U is a function from $\mathcal{P}(U^{n_1})\times\dots\times\mathcal{P}(U^{n_k})$ to $\{\mathbf{t},\mathbf{f}\}$, where $\mathcal{P}(U^{n_i})$ denotes the power set of U^{n_i}.

Example 2
Besides the standard connective and quantifiers, the following are other examples of generalized quantifiers.

- type $\langle 1\rangle$ quantifier $Q_{\leq 2}$ such that $Q^U_{\leq 2}(R) = \mathbf{t}$ iff the cardinality of R is ≤ 2; [5]
- type $\langle 1\rangle$ quantifier $Q_{majority}$ such that the cardinality of R is greater than the cardinality of $U\setminus R$;
- type $\langle 2,1,1\rangle$ reachability quantifier Q_{reach} such that $Q^U_{reach}(R_1,R_2,R_3) = \mathbf{t}$ iff there are some $u,v\in U$ such that $R_2=\{u\}$, $R_3=\{v\}$ and (u,v) belongs to the transitive closure of R_1.

Consider an interpretation I of a first-order signature σ. By σ^I we mean the signature obtained from σ by adding new object constants ξ^*, called *names*, for every element ξ in the universe of I. We identify an interpretation I of σ with its extension to σ^I defined by $I(\xi^*)=\xi$. For any term t of σ^I that does not contain variables, we define recursively the element t^I of

[5] It is clear from the type that R is any subset of U. We will skip such explanation.

the universe that is assigned to t by I. If t is an object constant then t^I is an element of U. For other terms, t^I is defined by the equation $f(t_1,\dots,t_n)^I = f^I(t_1^I,\dots,t_n^I)$ for all function constants f of arity $n > 0$.

Given a sentence F of σ^I, F^I is defined recursively as follows:

- $p(t_1,\dots,t_n)^I = p^I(t_1^I,\dots,t_n^I)$,
- $(t_1 = t_2)^I = (t_1^I = t_2^I)$,
- For a generalized quantifier Q of type $\langle n_1,\dots,n_k\rangle$,

  $$(Q[\mathbf{x}_1]\dots[\mathbf{x}_k](F_1(\mathbf{x}_1),\dots,F_k(\mathbf{x}_k)))^I = Q^U((\mathbf{x}_1.F_1(\mathbf{x}_1))^I,\dots,(\mathbf{x}_k.F_k(\mathbf{x}_k))^I),$$

  where $(\mathbf{x}_i.F_i(\mathbf{x}_i))^I = \{\boldsymbol{\xi} \in U^{n_i} \mid (F_i(\boldsymbol{\xi}^*))^I = \mathbf{t}\}$.

We assume that, for the standard logical connectives and quantifiers Q, functions Q^U have the standard meaning:

- $Q_\forall^U(R) = \mathbf{t}$ iff $R = U$;
- $Q_\exists^U(R) = \mathbf{t}$ iff $R \cap U \neq \emptyset$;
- $Q_\wedge^U(R_1, R_2) = \mathbf{t}$ iff $R_1 = R_2 = \{\epsilon\}$;[6]
- $Q_\vee^U(R_1, R_2) = \mathbf{t}$ iff at least one of them is $\{\epsilon\}$;
- $Q_\to^U(R_1, R_2) = \mathbf{t}$ iff R_1 is $\emptyset$ or R_2 is $\{\epsilon\}$;
- $Q_\neg^U(R) = \mathbf{t}$ iff $R = \emptyset$.
- $Q_\bot^U() = \mathbf{f}$.

We say that an interpretation I satisfies a sentence F, or is a model of F, and write $I \models F$, if $F^I = \mathbf{t}$. A sentence F is *logically valid* if every interpretation satisfies F.

Example 3
Let I_1 be an interpretation whose universe is $\{1,2,3,4\}$ and let p be a unary predicate constant such that $p(\xi^*)^{I_1} = \mathbf{t}$ iff $\xi \in \{1,2,3\}$. We check that I_1 satisfies the formula

$$F = \neg Q_{\le 2}[x]\, p(x) \to Q_{majority}[y]\, p(y)$$

("if p does not contain at most two elements in the universe, then p contains a majority"). Let I_2 be another interpretation with the same universe such that $p(\xi^*)^{I_2} = \mathbf{t}$ iff $\xi \in \{1\}$. It is clear that I_2 also satisfies F.

We say that a generalized quantifier (3) is *monotone in the i-th argument position* if the following holds for any interpretation I: if $Q^U(R_1,\dots,R_k) = \mathbf{t}$ and $R_i \subseteq R_i' \subseteq U^{n_i}$, then $Q^U(R_1,\dots,R_{i-1},R_i',R_{i+1},\dots,R_k) = \mathbf{t}$. Similarly, we say that Q is *anti-monotone in the i-th argument position* if the following holds for any interpretation I: if $Q^U(R_1,\dots,R_k) = \mathbf{t}$ and $R_i' \subseteq R_i \subseteq U^{n_i}$, then $Q^U(R_1,\dots,R_{i-1},R_i',R_{i+1},\dots,R_k) = \mathbf{t}$. We call an argument position of Q *monotone (anti-monotone)* if Q is monotone (anti-monotone) in that argument position.

Let M be a subset of $\{1,\dots,k\}$. We say that Q is *monotone in M* if Q is monotone in the i-th argument position for all i in M. It is easy to check that both $Q_\wedge$ and $Q_\vee$ are monotone in $\{1,2\}$. $Q_\to$ is anti-monotone in $\{1\}$ and monotone in $\{2\}$; $Q_\neg$ is anti-monotone in $\{1\}$. In Example 2, $Q_{\le 2}$ is anti-monotone in $\{1\}$ and $Q_{majority}$ is monotone in $\{1\}$. We will see later that (anti-)monotonicity play an important role in the properties of stable models for formulas with generalized quantifiers.

[6] ϵ denotes the empty tuple. For any interpretation I, $U^0 = \{\epsilon\}$. For I to satisfy $Q_\wedge[][](F,G)$, both $(\epsilon.F)^I$ and $(\epsilon.G)^I$ have to be $\{\epsilon\}$, which means that $F^I = G^I = \mathbf{t}$.

3 Stable Models of GQ-Formulas

We now define the stable model operator SM with a list of intensional predicates. Let $\mathbf{p}$ be a list of distinct predicate constants $p_1, \dots, p_n$, and let $\mathbf{u}$ be a list of distinct predicate variables $u_1, \dots, u_n$. By $\mathbf{u} \le \mathbf{p}$ we denote the conjunction of the formulas $\forall \mathbf{x}(u_i(\mathbf{x}) \rightarrow p_i(\mathbf{x}))$ for all $i = 1, \dots, n$, where $\mathbf{x}$ is a list of distinct object variables of the same length as the arity of p_i, and by $\mathbf{u} < \mathbf{p}$ we denote $(\mathbf{u} \le \mathbf{p}) \wedge \neg(\mathbf{p} \le \mathbf{u})$. For instance, if p and q are unary predicate constants then $(u, v) < (p, q)$ is

$$\forall x(u(x) \rightarrow p(x)) \wedge \forall x(v(x) \rightarrow q(x)) \wedge \neg(\forall x(p(x) \rightarrow u(x)) \wedge \forall x(q(x) \rightarrow v(x))).$$

For any first-order formula F and any list of predicates $\mathbf{p} = (p_1, \dots, p_n)$, formula SM$[F; \mathbf{p}]$ is defined as

$$F \wedge \neg \exists \mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge F^*(\mathbf{u})), \tag{4}$$

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 that does not contain members of $\mathbf{p}$;
- $$\begin{gathered}(Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)))^* = \\ Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1^*(\mathbf{x}_1), \dots, F_k^*(\mathbf{x}_k)) \wedge\ Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)).\end{gathered} \tag{5}$$

When F is a sentence, the models of SM$[F; \mathbf{p}]$ are called the $\mathbf{p}$-*stable* models of F: they are the models of F that are "stable" on $\mathbf{p}$. We often simply write SM$[F]$ in place of SM$[F; \mathbf{p}]$ when $\mathbf{p}$ is the list of all predicate constants occurring in F, and call $\mathbf{p}$-stable models simply stable models.

Proposition 1
Let M be a subset of $\{1, \dots, k\}$ and let $Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k))$ be a formula such that no predicate constant from $\mathbf{p}$ occurs in F_j for all $j \in \{1, \dots, k\} \setminus M$.

(a) If Q is monotone in M, then

$$\mathbf{u} \le \mathbf{p} \rightarrow ((Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)))^* \leftrightarrow Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1^*(\mathbf{x}_1), \dots, F_k^*(\mathbf{x}_k)))$$

is logically valid.

(b) If Q is anti-monotone in M, then

$$\mathbf{u} \le \mathbf{p} \rightarrow ((Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)))^* \leftrightarrow Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)))$$

is logically valid.

Proposition 1 allows us to simplify the formula $F^*(\mathbf{u})$ in (4) without affecting the models of (4). In formula (5), if Q is monotone in all argument positions, we can drop the second conjunctive term in view of Proposition 1 (a). Similarly, if Q is anti-monotone in all argument positions, we can drop the first conjunctive term in view of Proposition 1 (b). For instance, recall that each of $Q_\wedge$, $Q_\vee$, $Q_\forall$, $Q_\exists$ is monotone in all its argument positions, and $Q_\neg$ is anti-monotone in $\{1\}$. If F is a standard first-order formula, then (5) can be equivalently rewritten as

- $(\neg F)^* = \neg F$;
- $(F \wedge G)^* = F^* \wedge G^*$; $\;(F \vee G)^* = F^* \vee G^*$;
- $(F \rightarrow G)^* = (F^* \rightarrow G^*) \wedge (F \rightarrow G)$;

- $(\forall x F)^* = \forall x F^*$; $(\exists x F)^* = \exists x F^*$.

This is almost the same as the definition of F^* given in (Ferraris et al. 2011), except for $(\neg F)^*$.[7] The only propositional connective which is neither monotone nor anti-monotone in all argument positions is $Q_\rightarrow$, for which the simplification does not apply.

Example 3 continued *For formula F considered earlier,* SM$[F]$ *is*

$$F \wedge \neg \exists u (u < p \wedge F^*(u)) \,, \tag{6}$$

where $F^*(u)$ is equivalent to the conjunction of F and

$$\neg Q_{\le 2}[x]\; p(x) \rightarrow \; Q_{majority}[y]\; u(y). \tag{7}$$

I_1 considered earlier satisfies (6): it satisfies F and for any "proper subset" u of p, it satisfies the antecedent of (7) but not the consequent. Thus it is a stable model of F. On the other hand, we can check that I_2 does not satisfy (6).

4 Aggregates as GQ Formulas

4.1 Formulas with Aggregates

The following definition of a formula with aggregates is from (Ferraris and Lifschitz 2010), which extends (Lee and Meng 2009). By a *number* we understand an element of some fixed set **Num**. For example, **Num** is $\mathbf{Z} \cup \{+\infty, -\infty\}$, where **Z** is the set of integers. A *multiset* is usually defined as a set of elements along with a function assigning a positive integer, called multiplicity, to each of its elements. An *aggregate function* is a partial function from the class of multisets to **Num**. We assume that the signature σ contains the background signature σ_{bg} that contains symbols for all numbers. We assume that the interpretation of symbols in the background signature is fixed. That is, each number is interpreted as itself. An *expansion* I of I_{bg} to σ is an interpretation of σ such that

- the universe of I is the same as the universe of I_{bg}, and
- I agrees with I_{bg} on the constants in σ_{bg}.

An *aggregate formula* is defined as an extension of a first-order formula by adding the following clause:

- $$\textsc{op}\langle \mathbf{x}_1.F_1, \dots, \mathbf{x}_n.F_n \rangle \succeq b \tag{8}$$

  is *a first-order formula with aggregates*, where

  — $\textsc{op}$ is a symbol for an *aggregate function* (not from σ);
  — $\mathbf{x}_1, \dots, \mathbf{x}_n$ are non-empty lists of distinct object variables;
  — $F_1, \dots, F_n$ are arbitrary *first-order formulas with aggregates* of signature σ;
  — $\succeq$ is a symbol for a comparison operator (may not necessarily be from σ);
  — b is a term of σ.

[7] $\neg F$ is understood as $F \rightarrow \bot$ in (Ferraris et al. 2011), but this difference does not affect stable models.

4.2 Aggregates as Generalized Quantifiers

Due to the space limit, we refer the reader to (Ferraris and Lifschitz 2010) for the stable model semantics of formulas with aggregates. We can explain their semantics by viewing it as a special case of the stable model semantics presented here. Following (Ferraris and Lifschitz 2010), for any set X of n-tuples ($n \geq 1$), let *msp*(X) ("the multiset projection of X") be the multiset consisting of all ξ_1 such that $(\xi_1, \dots, \xi_n) \in X$ for at least one $(n-1)$-tuple $(\xi_2, \dots, \xi_n)$, with the multiplicity equal to the number of such $(n-1)$-tuples (and to $+\infty$ if there are infinitely many of them). For example, *msp*$(\{(a,a),(a,b),(b,a)\}) = \{\!\{a,a,b\}\!\}$.

We identify expression (8) with the GQ formula

$$Q_{(\text{OP},\succeq)}[\mathbf{x}_1] \dots [\mathbf{x}_n][y](F_1(\mathbf{x}_1), \dots, F_n(\mathbf{x}_n), y = b) \ , \tag{9}$$

where, for any interpretation I, $Q^U_{(\text{OP},\succeq)}$ is a function that maps $\mathcal{P}(U^{|\mathbf{x_1}|}) \times \dots \times \mathcal{P}(U^{|\mathbf{x_n}|}) \times \mathcal{P}(U)$ to $\{\mathbf{t}, \mathbf{f}\}$ such that $Q^U_{(\text{OP},\succeq)}(R_1, \dots, R_n, R_{n+1}) = \mathbf{t}$ iff

- $\text{OP}(\alpha)$ is defined, where α is the join of the multisets $msp(R_1), \dots, msp(R_n)$,
- $R_{n+1} = \{b^I\}$, where $b^I \in \mathbf{Num}$, and
- $\text{OP}(\alpha) \succeq b^I$;

Example 4
$\{$*discount*$($*alice*$),$ *discount*$($*carol*$),$ *numOfDiscounts*$(2)\}$ is a stable model of the formula

$$\begin{array}{l} \mathit{discount}(\mathit{alice}) \land \mathit{discount}(\mathit{carol}) \land \\ \forall z(\text{COUNT}\langle x.\mathit{discount}(x)\rangle = z \rightarrow \mathit{numOfDiscounts}(z)). \end{array}$$

The following proposition states that this definition is equivalent to the definition from (Ferraris and Lifschitz 2010).

Proposition 2
Let F be a first-order sentence with aggregates whose signature is σ, and let $\mathbf{p}$ be a list of predicate constants. For any expansion I of σ_{bg} to σ, I is a $\mathbf{p}$-stable model of F in the sense of (Ferraris and Lifschitz 2010) iff I is a $\mathbf{p}$-stable model of F in our sense.

5 Abstract Constraint Atoms as GQ Formulas

Marek and Truszczynski (2004) viewed propositional aggregates as a special case of abstract constraint atoms. Son et al. (2007) generalized this semantics to account for arbitrary abstract constraint atoms. In this section we present an alternative semantics of programs with abstract constraint atoms by reduction to formulas with generalized quantifiers.

Let σ be a propositional signature, D be a finite list of atoms of σ and C be a subset of the power set $\mathcal{P}(D)$.[8] An *abstract constraint atom* (or c-atom) (Son et al. 2007) is of the form $\langle D, C\rangle$. We say that an interpretation of σ satisfies a c-atom $\langle D, C\rangle$ if $I \cap D \in C$.

We view c-atoms as a special case of generalized quantifiers containing no variables, and this provides an alternative semantics of c-atoms that is different from (Son et al. 2007). An abstract constraint $\langle D, C\rangle$, where D is $(p_1, \dots, p_n)$, can be viewed as a generalized quantified formula

$$Q_C[\,], \dots, [\,]\ D \ , \tag{10}$$

[8] We will often identify a list with a set if there is no confusion.

where, for any interpretation I of σ, Q_C^U is a function that maps $\mathcal{P}(\{\epsilon\}) \times \cdots \times \mathcal{P}(\{\epsilon\})$ to $\{\mathbf{t}, \mathbf{f}\}$ such that $Q_C^U(R_1, \ldots, R_n) = \mathbf{t}$ iff $\{p_i \mid 1 \leq i \leq n,\ R_i = \{\epsilon\}\} \in C$.

Example 5
The following example is from (Liu et al. 2010). Let F be the formula

$$a \wedge b \wedge \langle (a, b, c), \{\{a\}, \{a, c\}, \{a, b, c\}\rangle \rightarrow c.$$

For new atoms d,e,f, formula $F^*(d, e, f)$ is

$$\begin{array}{l} d \wedge e \wedge \langle (a, b, c), \{\{a\}, \{a, c\}, \{a, b, c\}\}\rangle \rightarrow c) \wedge \\ \langle (a, b, c), \{\{a\}, \{a, c\}, \{a, b, c\}\}\rangle \wedge \\ \qquad\qquad \langle (d, e, f), \{\{d\}, \{d, f\}, \{d, e, f\}\}\rangle \rightarrow f). \end{array}$$

Any subset X of $\{a, b, c\}$ is an answer set of F iff X satisfies F and for any proper subset Y of X, $X \cup Y^{abc}_{def}$ does not satisfy $F^*(d, e, f)$. (Here Y^{abc}_{def} is the set obtained from Y by replacing a, b, c with d, e, f.)

We can check that $\{a, b\}$ is the only answer set of F. Indeed, $\{a, b\}$ satisfies F and each of $\{a, b\}$, $\{a, b, d\}$, $\{a, b, e\}$ does not satisfy $F^*(d, e, f)$.

Lemma 1
For any c-atom $\langle D, C\rangle$ of σ, let I be an interpretation of σ. I satisfies $\langle D, C\rangle$ in the sense of (Son et al. 2007) iff $I \models$ (10).

Given a c-atom (10), we define its *propositional formula representation* as

$$\bigwedge_{\overline{C} \in \mathcal{P}(D) \setminus C} \Big(\bigwedge_{p \in \overline{C}} p \rightarrow \bigvee_{p \in D \setminus \overline{C}} p \Big). \tag{11}$$

A *propositional formula with c-atoms* extends the standard syntax of propositional formula by treating c-atoms as a base case in addition to standard atoms. For any propositional formula F with c-atoms, by *Fer*(F), we denote the usual propositional formula obtained from F by replacing every c-atom (10) with (11).

The following proposition tells us that c-atoms in a formula can be rewritten as propositional formulas under the stable model semantics from (Ferraris 2005).

Proposition 3
For any propositional formula F with c-atoms and any propositional interpretation X, X is an answer set of F iff X is an answer set of *Fer*(F).

Example 5 continued *For the formula F above, Fer(F) is*

$$a \wedge b \wedge (((a \vee b \vee c) \wedge (b \rightarrow a \vee c) \wedge (c \rightarrow a \vee b) \wedge (a \wedge b \rightarrow c) \wedge (b \wedge c \rightarrow a)) \rightarrow c).$$

We check that $\{a, b\}$ is the only answer set of Fer(F) in accordance with Proposition 3.

Note that our semantics of logic programs with abstract constraint atoms is not equivalent to the one from (Son et al. 2007). Lee and Meng (2009) present a propositional formula representation of abstract constraint atoms under the semantics from (Son et al. 2007), which is classically equivalent, but not strongly equivalent to (11).

6 Nonmonotonic dl-Programs as GQ Formulas

6.1 Review of Nonmonotonic dl-Programs

Let C be a set of object constants, and let $P_{\mathcal{T}}$ and P_Π be disjoint sets of predicate constants. A nonmonotonic $dl\text{-}program$ (Eiter et al. 2008) is a pair $(\mathcal{T}, \Pi)$, where $\mathcal{T}$ is a theory in description logic (DL) of signature $\langle C, P_{\mathcal{T}}\rangle$ and Π is a *generalized* normal logic program of signature $\langle C, P_\Pi\rangle$ such that $P_{\mathcal{T}} \cap P_\Pi = \emptyset$. We assume that Π contains no variables by applying grounding w.r.t. C. A generalized normal logic program is a set of nondisjunctive rules that can contain queries to $\mathcal{T}$ in the form of "dl-atoms." A *dl-atom* is of the form

$$DL[S_1 op_1 p_1, \dots, S_k op_k p_k;\ Query](\mathbf{t}) \quad (k \geq 0), \tag{12}$$

where $S_i \in P_{\mathcal{T}}$, $p_i \in P_\Pi$, and $op_i \in \{\uplus, \cup\!\!\!\!-, \cap\!\!\!\!-\}$; $Query(\mathbf{t})$ is a *dl-query* as defined in (Eiter et al. 2008). A *dl-rule* is of the form

$$a \leftarrow b_1, \dots, b_m, \textit{not } b_{m+1}, \dots, \textit{not } b_n\ , \tag{13}$$

where a is an atom and each b_i is either an atom or a dl-atom. We identify rule (13) with

$$a \leftarrow B, N\ , \tag{14}$$

where B is $b_1, \dots, b_m$ and N is $\textit{not } b_{m+1}, \dots, \textit{not } b_n$. An Herbrand interpretation I *satisfies* a ground atom A *relative to* $\mathcal{T}$ if I satisfies A. An Herbrand interpretation I *satisfies* a ground dl-atom (12) *relative to* $\mathcal{T}$ if $\mathcal{T} \cup \bigcup_{i=1}^{k} A_i(I)$ entails $Query(\mathbf{t})$, where $A_i(I)$ is

- $\{S_i(\mathbf{e}) \mid\ p_i(\mathbf{e}) \in I\}$ if op_i is $\uplus$,
- $\{\neg S_i(\mathbf{e}) \mid\ p_i(\mathbf{e}) \in I\}$ if op_i is $\cup\!\!\!\!-$,
- $\{\neg S_i(\mathbf{e}) \mid\ p_i(\mathbf{e}) \notin I\}$ if op_i is $\cap\!\!\!\!-$.

A ground dl-atom A is *monotonic* relative to $\mathcal{T}$ if, for any two Herbrand interpretations I and I' such that $I \subseteq I'$, $I \models_{\mathcal{T}} A$ implies $I' \models_{\mathcal{T}} A$. Similarly, a ground dl-atom A is *anti-monotonic* relative to $\mathcal{T}$ if, for any two Herbrand interpretations I and I' such that $I \subseteq I'$, $I' \models_{\mathcal{T}} A$ implies $I \models_{\mathcal{T}} A$.

Given a dl-program $(\mathcal{T}, \Pi)$ and an Herbrand interpretation I of $\langle C, P_\Pi\rangle$, the *weak dl-transform* of Π relative to $\mathcal{T}$, denoted by $w\Pi^I_{\mathcal{T}}$, is the set of rules

$$a \leftarrow B' \tag{15}$$

where $a \leftarrow B, N$ is in Π, $I \models_{\mathcal{T}} B \wedge N$, and B' is obtained from B by removing all dl-atoms in it. Similarly, the *strong dl-transform* of Π relative to $\mathcal{T}$, denoted by $s\Pi^I_{\mathcal{T}}$, is the set of rules (15), where $a \leftarrow B, N$ is in Π, $I \models_{\mathcal{T}} B \wedge N$ and B' is obtained from B by removing all nonmonotonic dl-atoms in it. The only difference between these two definitions is whether monotonic dl-atoms in the positive body remain in the reduct or not.

An Herbrand interpretation I is a *weak (strong, respectively) answer set* of $(\mathcal{T}, \Pi)$ if I is minimal among the sets of atoms that satisfy $w\Pi^I_{\mathcal{T}}(s\Pi^I_{\mathcal{T}}, respectively)$.

6.2 Nonmonotonic dl-program as Formulas with Generalized Quantifiers

Here we understand dl-programs as a special case of GQ formulas. Consider a dl-program $(\mathcal{T}, \Pi)$ such that Π is ground. Under the strong answer set semantics, we identify every dl-atom (12) in Π with

$$Q_{(12)}[\mathbf{x}_1] \dots [\mathbf{x}_k](p_1(\mathbf{x}_1), \dots, p_k(\mathbf{x}_k)) \tag{16}$$

if it is monotonic relative to $(\mathcal{T}, \Pi)$, and

$$\neg\neg Q_{(12)}[\mathbf{x}_1]\dots[\mathbf{x}_k](p_1(\mathbf{x}_1),\dots,p_k(\mathbf{x}_k)) \tag{17}$$

otherwise.

Given an interpretation I, $Q^U_{(12)}$ is a function that maps $\mathcal{P}(U^{|\mathbf{x}_1|}) \times \cdots \times \mathcal{P}(U^{|\mathbf{x}_k|})$ to $\{\mathbf{t}, \mathbf{f}\}$ such that, $Q^U_{(12)}(R_1,\dots,R_k) = \mathbf{t}$ iff $\mathcal{T} \cup \bigcup_{i=1}^k A_i(R_i)$ entails $Query(\mathbf{t})$, where $A_i(R_i)$ is

- $\{S_i(\boldsymbol{\xi}_i) \mid \boldsymbol{\xi}_i \in R_i\}$ if op_i is $\uplus$,
- $\{\neg S_i(\boldsymbol{\xi}_i) \mid \boldsymbol{\xi}_i \in R_i\}$ if op_i is $\cup\!\!\!-$,
- $\{\neg S_i(\boldsymbol{\xi}_i) \mid \boldsymbol{\xi}_i \in U^{|\mathbf{x}_i|} \setminus R_i\}$ if op_i is $\cap\!\!\!-$.

We say that I is a *strong answer set* of $(\mathcal{T}, \Pi)$ if I satisfies SM$[\Pi; P_\Pi]$.

Similarly, a weak answer set of $(\mathcal{T}, \Pi)$ is defined by identifying every dl-atom (12) in Π with (17) regardless of A being monotonic or not.

The following proposition tells us that the definitions of a strong answer set and a weak answer set given here are equivalent to the definitions from (Eiter et al. 2008).

Proposition 4
For any dl-program $(\mathcal{T}, \Pi)$, an Herbrand interpretation is a strong (weak, respectively) answer set of $(\mathcal{T}, \Pi)$ in the sense of (Eiter et al. 2008) iff it is a strong (weak, respectively) answer set of $(\mathcal{T}, \Pi)$ in our sense.

6.3 Another Semantics of Nonmonotonic dl-programs

Shen (2011) notes that both strong and weak answer set semantics suffer from circular justifications.

Example 6 (*(Shen 2011)*)
Consider $(\mathcal{T}, \Pi)$, where $\mathcal{T} = \emptyset$ and Π is the program

$$p(a) \leftarrow \#dl[c \uplus p, b \cap\!\!\!-\, q;\; c \sqcap \neg b](a). \tag{18}$$

The dl-program has two strong (weak, respectively) answer sets: $\emptyset$ and $\{p(a)\}$. According to (Shen 2011), the second answer set is circularly justified:

$$p(a) \Leftarrow \#dl[c \uplus p, b \cap\!\!\!-\, q; c \sqcap \neg b](a) \Leftarrow p(a) \wedge \neg q(a).$$

Indeed, $s\Pi^{\{p(a)\}}_{\mathcal{T}}$ ($w\Pi^{\{p(a)\}}_{\mathcal{T}}$, respectively) is $p(a) \leftarrow$, and $\{p(a)\}$ is its minimal model.

The example suggests that the issue is related to the fact that both strong and weak answer set semantics do not distinguish between different kinds of nonmonotonic dl-atoms: anti-monotonic and non-anti-monotonic ones. The former does not contribute to loops, but the latter does, so that they should participate in enforcing minimality of answer sets (See the later section on loops). This suggests the following alternative definition of the semantics of dl-programs. Instead of removing every nonmonotonic dl-atoms in forming the reduct under strong answer set semantics, we remove only anti-monotonic dl-atoms from the bodies, but leave non-anti-monotonic dl-atoms. In other words, the *dl-transform* of Π relative to $\mathcal{T}$ and an Herbrand interpretation I of $\langle C, P_\Pi\rangle$, denoted by $\Pi^I_{\mathcal{T}}$, is the set of rules (15), where $a \leftarrow B, N$ is in Π, $I \models_{\mathcal{T}} B \wedge N$ and B' is obtained from B by removing all anti-monotonic dl-atoms in it. We say that an Herbrand interpretation I is an *answer set* of $(\mathcal{T}, \Pi)$ if I is minimal among the sets of atoms that satisfy $\Pi^I_{\mathcal{T}}$.

Example 6 continued $\{p(a)\}$ *is not an answer set of* $(\mathcal{T}, \Pi)$ *according to the new definition.* $\Pi_{\mathcal{T}}^{\{p(a)\}}$ *is (18) itself, and* $\emptyset$*, a proper subset of* $\{p(a)\}$ *satisfies it.*

This new definition can be also characterized in terms of generalized quantifiers. In fact, the characterization is simpler than those for the other two semantics. We simply identify (12) with (16) regardless of the (anti-)monotonicity of the dl-atom.

Proposition 5
For any dl-program $(\mathcal{T}, \Pi)$, and any Herbrand interpretation X of $\langle C, P_\Pi \rangle$, X is an answer set of $(\mathcal{T}, \Pi)$ as defined here iff X satisfies SM$[\Pi; \mathbf{p}]$ when we identify every dl-atom (12) in Π with (16).

The new definition is closely related to another variant of FLP-reduct based semantics of nonmonotonic dl-programs from (Fink and Pearce 2010). The following proposition states that the relationship between the two semantics.

Proposition 6
For any dl-program $(\mathcal{T}, \Pi)$, and any Herbrand interpretation X of $\langle C, P_\Pi \rangle$, if every occurrence of non-monotonic dl-atoms is in the positive body of a rule, then X is an answer set of $(\mathcal{T}, \Pi)$ in the sense of (Fink and Pearce 2010) iff X is an answer set of $(\mathcal{T}, \Pi)$ in our sense.

The following example shows why the condition in the statement is essential.

Example 7
Consider the single-rule dl-program

$$p(a) \leftarrow \mathit{not}\ \#\mathit{dl}[C \Cap p; \neg C](a).$$

While $\emptyset$ and $\{p(a)\}$ are answer sets according to us, only $\emptyset$ is the answer set according to (Fink and Pearce 2010).

7 Strong Equivalence

Strong equivalence (Lifschitz et al. 2001) is an important notion that allows us to substitute one subformula for another subformula without affecting the stable models. The theorem on strong equivalence can be extended to GQ formulas as follows.

About GQ formulas F and G we say that F is *strongly equivalent* to G if, for any formula H, any occurrence of F in H, and any list $\mathbf{p}$ of distinct predicate and function constants, SM$[H; \mathbf{p}]$ is equivalent to SM$[H'; \mathbf{p}]$, where H' is obtained from H by replacing the occurrence of F by G. In this definition, H is allowed to contain object, function and predicate constants that do not occur in F, G; Theorem 1 below shows, however, that this is not essential.

Theorem 1
Let F and G be GQ formulas, let $\mathbf{p}$ be the list of all predicate constants occurring in F or G and let $\mathbf{u}$ be a list of distinct predicate variables corresponding to $\mathbf{p}$. Formulas F and G are strongly equivalent to each other iff the formula

$$(\mathbf{u} \le \mathbf{p}) \rightarrow (F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u}))$$

is logically valid.

Example 8
The program (2) in the introduction can be identified with the formula F

$$(\neg\text{COUNT}\langle x.p(x)\rangle < 1 \rightarrow p(a)) \wedge\ (\text{COUNT}\langle x.p(x)\rangle < 1 \rightarrow q),$$

and is strongly equivalent to the following formula G:

$$(\neg q \rightarrow p(a)) \wedge\ (\text{COUNT}\langle x.p(x)\rangle < 1 \rightarrow q).$$

One can check that $F^*(u,v)$ and $G^*(u,v)$ are equivalent to each other.

8 Splitting Theorem

We extend the splitting theorem from (Ferraris et al. 2009) to GQ formulas.

Let F be a GQ formula. We say that an occurrence of p in F is *mixed* if there is some generalized quantifier Q that contains the occurrence in its argument position which is neither monotone nor anti-monotone. Let l be the number of generalized quantifiers Q in F such that the occurrence of p belongs to an anti-monotone argument position of Q. If the occurrence is not mixed then we call it *positive* in F if l is even, and *negative* otherwise. The occurrence is *strictly positive* in F if $l = 0$. We call an occurrence of predicate constant *semi-positive* if it is positive or mixed. Similarly, it is *semi-negative* if it is negative or mixed.

We say that F is *negative on* $\mathbf{p}$ if there is no strictly positive occurrence of a predicate constant from $\mathbf{p}$ in F. An occurrence of a predicate constant or a subformula of F is $\mathbf{p}$*-negated* in F if it is contained in a subformula of F that is negative on $\mathbf{p}$.

The dependency graph of F relative to a list $\mathbf{p}$ of intensional predicates (denoted by $\text{DG}_{\mathbf{p}}[F]$) is a directed graph such that

- the vertices are the members of $\mathbf{p}$, and
- there is an edge from p to q if there is a strictly positive occurrence of a subformula $G = Q[\mathbf{x}_1]\dots[\mathbf{x}_k](F_1,\dots,F_k)$ such that
  - p has a strictly positive occurrence in G, and
  - q has a semi-positive, non-$\mathbf{p}$-negated occurrence in a non-monotone argument position of Q.

A *loop* of F (relative to a list $\mathbf{p}$ of intensional predicates) is a nonempty subset $\mathbf{l}$ of $\mathbf{p}$ such that the subgraph of $\text{DG}_{\mathbf{p}}[F]$ induced by $\mathbf{l}$ is strongly connected. It is clear that the strongly connected components of $\text{DG}_{\mathbf{p}}[F]$ can be characterized as the maximal loops of F.

Example 1 continued *Figure 1 shows the dependency graph of F relative to* $\{$*discount*, *family*, *mm*, *mw*, *accident*, *numOfDisc*

Theorem 2
Let F be a GQ sentence, and let $\mathbf{p}$ be a tuple of distinct predicate constants. If $\mathbf{l}^1,\dots,\mathbf{l}^n$ are all the loops of F relative to $\mathbf{p}$ then

$$\text{SM}[F;\mathbf{p}] \quad \text{is equivalent to} \quad \text{SM}[F;\mathbf{l}^1] \wedge \dots \wedge \text{SM}[F;\mathbf{l}^n].$$

The following theorem extends the splitting theorem from (Ferraris et al. 2009) to GQ sentences.

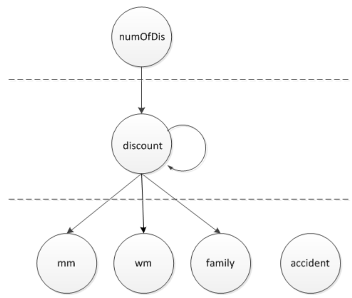

Fig. 1. Dependency graph of the Formula in Example 1

Theorem 3
Let F, G be GQ sentences, and let $\mathbf{p}$, $\mathbf{q}$ be disjoint tuples of distinct predicate constants. If

- each strongly connected component of $\mathrm{DG}_{\mathbf{pq}}[F \wedge G]$ is a subset of $\mathbf{p}$ or a subset of $\mathbf{q}$,
- F is negative on $\mathbf{q}$, and
- G is negative on $\mathbf{p}$

then

$$\mathrm{SM}[F \wedge G; \mathbf{pq}] \quad \text{is equivalent to} \quad \mathrm{SM}[F; \mathbf{p}] \wedge \mathrm{SM}[G; \mathbf{q}].$$

Example 1 continued $\mathrm{SM}[F;\ \mathit{discount}, \mathit{numOfDiscount}]$ is equivalent to

$$\mathrm{SM}[G_1; \mathit{discount}] \wedge \mathrm{SM}[G_2; \mathit{numOfDiscount}],$$

where G_1 is the conjunction of the first two implications in F and G_2 is the last implication.

9 Completion

A GQ formula F is in *Clark normal form* if it is a conjunction of sentences of the form

$$\forall \mathbf{x}(G \rightarrow p(\mathbf{x})), \tag{19}$$

one for each intensional predicate p, where $\mathbf{x}$ is a list of distinct object variables, and G has no free variables other than $\mathbf{x}$. The *completion* (relative to $\mathbf{p}$) of a GQ formula F in Clark normal form, denoted by $\mathrm{COMP}[F]$, is obtained by replacing each conjunctive term (19) with

$$\forall \mathbf{x}(p(\mathbf{x}) \leftrightarrow G).$$

We say that a GQ formula is *tight* on $\mathbf{p}$ if its dependency graph relative to $\mathbf{p}$ is acyclic.

Theorem 4
For any GQ formula F in Clark normal form that is tight on $\mathbf{p}$, $\mathrm{SM}[F; \mathbf{p}]$ is equivalent to the completion of F relative to $\mathbf{p}$.

Example 1 continued *Let F' be the formula obtained from F by dropping the second implication. The Clark normal form of F' is tight on $\{\mathit{discount}, \mathit{numOfDiscount}\}$. So $\mathrm{SM}[F_3; \mathit{discount}, \mathit{numOfDiscount}]$ equivalent to*

$$\begin{array}{l}\forall x(\mathit{discount}(x) \leftrightarrow \neg \mathit{accident}(x) \wedge \\ \quad \#\mathit{dl}[\mathit{Man} \uplus \mathit{mm}, \mathit{Married} \uplus \mathit{mm}, \mathit{Woman} \uplus \mathit{mw}, \mathit{Married} \uplus \mathit{mw}; \exists \mathit{Spouse}.\top](x)) \\ \wedge\ \forall y(\mathit{numOfDiscount}(y) \leftrightarrow \mathrm{COUNT}\langle x.\mathit{discount}(x)\rangle = y).\end{array}$$

10 Related Work

10.1 HEX Programs

We refer the reader to (Eiter et al. 2005) for the semantics of HEX programs. It is not difficult to see that an external atom in a HEX program can be represented in terms of a generalized quantifier. (Eiter et al. 2005) show how dl-atoms can be simulated by external atoms $\#dl[](x)$. The treatment is similar to ours in terms of generalized quantifiers. For another example, rule

$$reached(x) \leftarrow \#reach[edge, a](x)$$

defines all the vertices that are reachable from the vertex a in the graph with $edge$. The external atom $\#reach[edge, a](x)$ can be represented by a generalized quantified formula

$$Q_{reach}[x_1x_2][x_3][x_4](edge(x_1, x_2), x_3 = a, x_4 = x),$$

where Q_{reach} is as defined in Example 2.

On the other hand, unlike HEX programs that resorts to external functions that do not even occur in the program, generalized quantifiers are part of the language.

10.2 Logic Programs with GQ by Eiter et al.

In fact, the incorporation of generalized quantifiers in logic programming was considered earlier in (Eiter et al. 1997a; Eiter et al. 1997b), but the treatment there was to simply view them like negative literals. This approach does not allow recursion through generalized quantified formulas, and often yields an unintuitive result even when we limit attention to standard quantifiers. For instance, according to (Eiter et al. 1997a), program

$$p(a) \leftarrow \forall x\, p(x) \tag{20}$$

has two answer sets, $\emptyset$ and $\{p(a)\}$. The latter is "unfounded." This is not the case in the first-order stable model semantics (Ferraris et al. 2011; Lin and Zhou 2011), which allows the standard quantifiers, but no other generalized quantifiers. According to our semantics, which properly extends the semantics from (Ferraris et al. 2011) does not have the unintuitive answer set $\{p(a)\}$.

11 Conclusion

We presented the stable model semantics for formulas containing generalized quantifiers, and showed that several recent extensions of the stable model semantics can be viewed as special cases of this formalism. We expect that the generality of the formalism is useful in providing a principled way to study and compare the different extensions of the stable model semantics. Indeed, it led us to define yet another semantics of logic programs with abstract constraints, and yet another semantics of nonmonotonic dl-programs, both of which are in the spirit of the first-order stable model semantics.

Bartholomew et al. (2011) provide a first-order extension of the FLP semantics that is applied to formulas with aggregates. Similar to the approach here, FLP semantics of GQ-formulas can be defined, which can serve as a first-order extension of HEX programs. Defining this and studying its relationship to the stable model semantic presented here is a future work.

References

BARTHOLOMEW, M., LEE, J., AND MENG, Y. 2011. First-order extension of the flp stable model semantics via modified circumscription. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. 724–730.

EITER, T., FINK, M., IANNI, G., KRENNWALLNER, T., AND SCHÜLLER, P. 2011. Pushing efficient evaluation of hex programs by modular decomposition. In *Proceedings of the 11th international conference on Logic programming and nonmonotonic reasoning*. LPNMR'11. Springer-Verlag, Berlin, Heidelberg, 93–106.

EITER, T., GOTTLOB, G., AND VEITH, H. 1997a. Generalized quantifiers in logic programs. In *In Proceedings of the ESSLLI Workshop on Generalized Quantifiers, Aix-en-Provence*. Springer, 72–98.

EITER, T., GOTTLOB, G., AND VEITH, H. 1997b. Modular logic programming and generalized quantifiers. In *LPNMR'97*. 290–309.

EITER, T., IANNI, G., LUKASIEWICZ, T., SCHINDLAUER, R., AND TOMPITS, H. 2008. Combining answer set programming with description logics for the semantic web. *Artificial Intelligence 172,* 12-13, 1495–1539.

EITER, T., IANNI, G., SCHINDLAUER, R., AND TOMPITS, H. 2005. A uniform integration of higher-order reasoning and external evaluations in answer-set programming. In *IJCAI*. 90–96.

EITER, T., IANNI, G., SCHINDLAUER, R., AND TOMPITS, H. 2006. Effective integration of declarative rules with external evaluations for semantic-web reasoning. In *ESWC*. 273–287.

FABER, W., PFEIFER, G., AND LEONE, N. 2011. Semantics and complexity of recursive aggregates in answer set programming. *Artificial Intelligence 175,* 1, 278–298.

FERRARIS, P. 2005. Answer sets for propositional theories. In *Proceedings of International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*. 119–131.

FERRARIS, P., LEE, J., AND LIFSCHITZ, V. 2006. A generalization of the Lin-Zhao theorem. *Annals of Mathematics and Artificial Intelligence 47*, 79–101.

FERRARIS, P., LEE, J., AND LIFSCHITZ, V. 2007. A new perspective on stable models. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. 372–379.

FERRARIS, P., LEE, J., AND LIFSCHITZ, V. 2011. Stable models and circumscription. *Artificial Intelligence 175*, 236–263.

FERRARIS, P., LEE, J., LIFSCHITZ, V., AND PALLA, R. 2009. Symmetric splitting in the general theory of stable models. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*. 797–803.

FERRARIS, P. AND LIFSCHITZ, V. 2010. On the stable model semantics of firsr-order formulas with aggregates. In *NMR*.

FINK, M. AND PEARCE, D. 2010. A logical semantics for description logic programs. In *Proceedings of European Conference on Logics in Artificial Intelligence (JELIA)*. 156–168.

LEE, J., LIFSCHITZ, V., AND PALLA, R. 2008. A reductive semantics for counting and choice in answer set programming. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 472–479.

LEE, J. AND MENG, Y. 2009. On reductive semantics of aggregates in answer set programming. In *Proceedings of International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*. 182–195.

LIFSCHITZ, V. 2002. Answer set programming and plan generation. *Artificial Intelligence 138*, 39–54.

LIFSCHITZ, V., PEARCE, D., AND VALVERDE, A. 2001. Strongly equivalent logic programs. *ACM Transactions on Computational Logic 2*, 526–541.

LIN, F. AND ZHOU, Y. 2011. From answer set logic programming to circumscription via logic of GK. *Artificial Intelligence 175*, 264–277.

LINDSTRÖM, P. 1966. First-order predicate logic with generalized quantifiers. *Theoria 32*, 186–195.

LIU, L., PONTELLI, E., SON, T. C., AND TRUSZCZYNSKI, M. 2010. Logic programs with abstract constraint atoms: The role of computations. *Artificial Intelligence 174,* 34, 295 – 315.

MAREK, V. W. AND TRUSZCZYNSKI, M. 2004. Logic programs with abstract constraint atoms. In *AAAI*. 86–91.

MOSTOWSKI, A. 1957. On a Generalization of Quantifiers. *Fundamenta Mathematicae 44*, 12–35.

NIEMELÄ, I. AND SIMONS, P. 2000. Extending the Smodels system with cardinality and weight constraints. In *Logic-Based Artificial Intelligence*, J. Minker, Ed. Kluwer, 491–521.

SHEN, Y.-D. 2011. Well-supported semantics for description logic programs. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*. 1081–1086.

SON, T. C. AND PONTELLI, E. 2007. A constructive semantic characterization of aggregates in answer set programming. *TPLP 7,* 3, 355–375.

SON, T. C., PONTELLI, E., AND TU, P. H. 2007. Answer sets for logic programs with arbitrary abstract constraint atoms. *J. Artif. Intell. Res. (JAIR) 29*, 353–389.

TRUSZCZYŃSKI, M. 2010. Reducts of propositional theories, satisfiability relations, and generalizations of semantics of logic programs. *Artificial Intelligence 174,* 16-17, 1285–1306.

WESTERSTÅHL, D. 2008. Generalized quantifiers. In *The Stanford Encyclopedia of Philosophy (Winter 2008 Edition)*. URL = <http://plato.stanford.edu/archives/win2008/entries/generalized-quantifiers/>.

Appendix A Proofs

A.1 Useful Lemmas

Lemma 2
Let F be a GQ-formula. Formula

$$(\mathbf{u} \leq \mathbf{p}) \wedge F^*(\mathbf{u}) \rightarrow F$$

is logically valid.

Proof. By induction on F. ▮

Lemma 3
Let F be a GQ-formula. Formula

$$\mathbf{q} = \mathbf{p} \rightarrow (F^*(\mathbf{q}) \leftrightarrow F^*(\mathbf{p}))$$

is logically valid.

Proof. By induction on F. ▮

To facilitate the proofs, we introduce the following notion. Let Q be a generalized quantifier and let I be an interpretation. We say that Q^U is *monotone in the i-th argument position* if the following holds: if $Q^U(R_1, \dots, R_k) =$ **t** and $R_i \subseteq R'_i \subseteq |I|^{|\mathbf{x}^i|}$, then $Q^U(R_1, \dots, R_{i-1}, R'_i, R_{i+1}, \dots, R_k) =$ **t**. Similarly, we say that Q^U is *anti-monotone in the i-th argument position* if the following holds: if $Q^U(R_1, \dots, R_k) =$ **t** and $R'_i \subseteq R_i \subseteq |I|^{|\mathbf{x}^i|}$, then $Q^U(R_1, \dots, R_{i-1}, R'_i, R_{i+1}, \dots, R_k) =$ **t**. Clearly, Q is monotone (anti-monotone) in the i-th argument position iff Q^U is monotone (anti-monotone) in the i-th argument position for any interpretation I. Similarly, we define that Q^U is monotone (anti-monotone) in some set of argument positions.

Lemma 4
Consider GQ sentences

$$F = Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)),$$

$$G = Q[\mathbf{x}_1] \dots [\mathbf{x}_k](G_1(\mathbf{x}_1), \dots, G_k(\mathbf{x}_k)),$$

any subset M of $\{1, \dots, k\}$, and any interpretation I.

(a) If Q^U is monotone in M, then

$$I \models \Big(\bigwedge_{i \in M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \rightarrow G_i(\mathbf{x}_i)) \wedge \bigwedge_{i \in \{1,\dots,k\} \setminus M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \leftrightarrow G_i(\mathbf{x}_i)) \Big) \rightarrow (F \rightarrow G).$$

(b) If Q^U is anti-monotone in M, then

$$I \models \Big(\bigwedge_{i \in M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \rightarrow G_i(\mathbf{x}_i)) \wedge \bigwedge_{i \in \{1,\dots,k\} \setminus M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \leftrightarrow G_i(\mathbf{x}_i)) \Big) \rightarrow (G \rightarrow F).$$

Proof.
(a): Assume

$$I \models \bigwedge_{i\in M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \rightarrow G_i(\mathbf{x}_i)) \land \\ \bigwedge_{i\in \{1,\dots,k\}\setminus M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \leftrightarrow G_i(\mathbf{x}_i))$$

and $I \models F$. Consider $(\mathbf{x}_i.F_i)^I = \{\boldsymbol{\xi} \mid I \models F_i(\boldsymbol{\xi}^*)\}$ and $(\mathbf{x}_i.G_i)^I = \{\boldsymbol{\xi} \mid I \models G_i(\boldsymbol{\xi}^*)\}$ for each i in $\{1,\dots,k\}$.

- If $i \in M$, it follows from $I \models \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \rightarrow G_i(\mathbf{x}_i))$ that $(\mathbf{x}_i.F_i)^I \subseteq (\mathbf{x}_i.G_i)^I$.
- If $i \in \{1,\dots,k\} \setminus M$, it follows from $I \models \forall \mathbf{x}_i (F_i(\mathbf{x}_i) \leftrightarrow G_i(\mathbf{x}_i))$ that $(\mathbf{x}_i.F_i)^I = (\mathbf{x}_i.G_i)^I$.

From $I \models F$, by definition, $Q^U((\mathbf{x}_1.F_1)^I,\dots,(\mathbf{x}_k.F_k)^I) = \mathbf{t}$. Since Q^U is monotone in M, it follows that $Q^U((\mathbf{x}_1.G_1)^I,\dots,(\mathbf{x}_k.G_k)^I) = \mathbf{t}$. Thus $I \models G$.

(b): Similar to (a). ∎

The following lemma follows immediately from Lemma 4.

Lemma 5
Let M be a subset of $\{1,\dots,k\}$ and Q a generalized quantifier. Consider formulas

$$F(\mathbf{x}) = Q[\mathbf{x}_1],\dots,[\mathbf{x}_k](F_1(\mathbf{x}_1,\mathbf{x}),\dots,F_k(\mathbf{x}_k,\mathbf{x})),$$

$$G(\mathbf{x}) = Q[\mathbf{x}_1],\dots,[\mathbf{x}_k](G_1(\mathbf{x}_1,\mathbf{x}),\dots,G_k(\mathbf{x}_k,\mathbf{x})),$$

where $\mathbf{x}$ is a list of all free variables in F and G.

(a) If Q is monotone in M, then

$$\Big(\bigwedge_{i\in M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i,\mathbf{x}) \rightarrow G_i(\mathbf{x}_i,\mathbf{x})) \land \\ \bigwedge_{i\in \{1,\dots,k\}\setminus M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i,\mathbf{x}) \leftrightarrow G_i(\mathbf{x}_i,\mathbf{x}))\Big) \\ \rightarrow (F(\mathbf{x}) \rightarrow G(\mathbf{x}))$$

is logically valid.

(b) If Q is anti-monotone in M, then

$$\Big(\bigwedge_{i\in M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i,\mathbf{x}) \rightarrow G_i(\mathbf{x}_i,\mathbf{x})) \land \\ \bigwedge_{i\in \{1,\dots,k\}\setminus M} \forall \mathbf{x}_i (F_i(\mathbf{x}_i,\mathbf{x}) \leftrightarrow G_i(\mathbf{x}_i,\mathbf{x}))\Big) \\ \rightarrow (G(\mathbf{x}) \rightarrow F(\mathbf{x}))$$

is logically valid.

Lemma 6
If F is negative on $\mathbf{p}$ then

$$(\mathbf{u} \leq \mathbf{p}) \rightarrow (F^*(\mathbf{u}) \leftrightarrow F)$$

is logically valid.

Proof. By induction on F.
Case 1: F is an atomic formula. If F is of the form $p_i(\mathbf{t})$ then $p_i \notin \mathbf{p}$ since F is negative on $\mathbf{p}$. Consequently, $F^*(\mathbf{u})$ is the same as F. Otherwise, $F^*(\mathbf{u})$ is the same as F by definition.

Case 2: F is of the form (3). In view of Lemma 2, it is sufficient to show that

$$(\mathbf{u} \leq \mathbf{p}) \rightarrow (F \rightarrow F^*(\mathbf{u})) \tag{A1}$$

is logically valid. Let *Anti* be the set of all anti-monotone argument positions of Q.

- Consider any F_i, where $i \in \{1, \ldots, k\} \setminus \textit{Anti}$. Since F is negative on $\mathbf{p}$, it follows that F_i is negative on $\mathbf{p}$. By I.H.,
$$(\mathbf{u} \leq \mathbf{p}) \rightarrow (F_i^*(\mathbf{u}) \leftrightarrow F_i)$$
is logically valid.
- Consider any F_i, where $i \in \textit{Anti}$. By Lemma 2,
$$(\mathbf{u} \leq \mathbf{p}) \rightarrow (F_j^*(\mathbf{u}) \rightarrow F_j)$$
is logically valid.

From the two bullets, by Lemma 5 (b), we conclude (A1). ∎

A.2 Proof of Proposition 1

An interpretation I of a signature σ can be represented as a pair $\langle J, X\rangle$, where J is the restriction of I to the function constants in σ, and X is the set of the atoms, formed using predicate constants from σ and the names of elements of $|I|$, which are satisfied by I. When I is an Herbrand interpretation, we often omit J and represent I by X.

By σ^+ we denote the signature obtained from σ by adding new predicate constants $\mathbf{q}$, one per each member of $\mathbf{p}$. About an atomic formula formed using a predicate constant from σ^+ and names of elements of $|I|$ we say that it is a $\mathbf{p}$-*atom* if its predicate constant belongs to $\mathbf{p}$, and that it is a $\mathbf{q}$-*atom* otherwise. For any set X of $\mathbf{p}$-atoms we denote by $X^{\mathbf{p}}_{\mathbf{q}}$ the set of the $\mathbf{q}$-atoms that are obtained from the elements of X by replacing their predicate constants by the corresponding predicate constants from $\mathbf{q}$.

Lemma 7
Let M be a subset of $\{1, \ldots, k\}$ and let $Q[\mathbf{x}_1]\ldots[\mathbf{x}_k](F_1(\mathbf{x}_1), \ldots, F_k(\mathbf{x}_k))$ be a sentence such that F_j is negative on $\mathbf{p}$ for all j in $\{1, \ldots, k\}\setminus M$. Consider any interpretation $I = \langle J, X\rangle$ and any subset Y of X.

(a) If Q^U is monotone in M, then
$$\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle \models (Q[\mathbf{x}_1]\ldots[\mathbf{x}_k](F_1(\mathbf{x}_1), \ldots, F_k(\mathbf{x}_k)))^*(\mathbf{q}) \\ \leftrightarrow Q[\mathbf{x}_1]\ldots[\mathbf{x}_k](F_1^*(\mathbf{x}_1), \ldots, F_k^*(\mathbf{x}_k)).$$

(b) If Q^U is anti-monotone in M, then
$$\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle \models (Q[\mathbf{x}_1]\ldots[\mathbf{x}_k](F_1(\mathbf{x}_1), \ldots, F_k(\mathbf{x}_k)))^*(\mathbf{q}) \\ \leftrightarrow Q[\mathbf{x}_1]\ldots[\mathbf{x}_k](F_1(\mathbf{x}_1), \ldots, F_k(\mathbf{x}_k)).$$

Proof. (a) It is sufficient to show that

$$\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle \models Q[\mathbf{x}_1]\ldots[\mathbf{x}_k](F_1^*(\mathbf{x}_1), \ldots, F_k^*(\mathbf{x}_k)) \\ \rightarrow Q[\mathbf{x}_1]\ldots[\mathbf{x}_k](F_1(\mathbf{x}_1), \ldots, F_k(\mathbf{x}_k)). \tag{A2}$$

- For every $i \in \{1, \ldots, k\} \setminus M$, F_i is negative on $\mathbf{p}$. By Lemma 6, $\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle \models F_i^*(\mathbf{q}) \leftrightarrow F_i$.

- For every $i \in M$, by Lemma 2, $\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}} \rangle \models F_i^*(\mathbf{q}) \rightarrow F_i$.

From the above two facts, by Lemma 4(a), we conclude (A2).

(b) It is sufficient to show that

$$\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}} \rangle \models Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)) \\ \rightarrow Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1^*(\mathbf{x}_1), \dots, F_k^*(\mathbf{x}_k)). \tag{A3}$$

- For every $i \in \{1, \dots, k\} \setminus M$, F_i is negative on $\mathbf{p}$. By Lemma 6, $\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}} \rangle \models F_i^*(\mathbf{q}) \leftrightarrow F_i$.
- For every $i \in M$, by Lemma 2, $\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}} \rangle \models F_i^*(\mathbf{q}) \rightarrow F_i$.

From the above two facts, by Lemma 4(b), we conclude (A3). ∎

We now prove a slightly more general version of Proposition 1.

Proposition 1′ *Let M be a subset of $\{1, \dots, k\}$ and let $Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k))$ be a formula such F_j is negative on $\mathbf{p}$ for all $j \in \{1, \dots, k\} \setminus M$.*

(a) If Q is monotone in M, then

$$\mathbf{u} \le \mathbf{p} \rightarrow ((Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)))^* \\ \leftrightarrow Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1^*(\mathbf{x}_1), \dots, F_k^*(\mathbf{x}_k)))$$

is logically valid.

(b) If Q is anti-monotone in M, then

$$\mathbf{u} \le \mathbf{p} \rightarrow ((Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)))^* \\ \leftrightarrow Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1(\mathbf{x}_1), \dots, F_k(\mathbf{x}_k)))$$

is logically valid.

Proof. Clear from Lemma 7. ∎

A.3 Proof of Proposition 2

Lemma 8

let E be an aggregate expression (8) that contains no free variables, let E^{GQ} be the GQ-representation of E, and let I be an interpretation. $I \models E$ iff $I \models E^{GQ}$.

Proof. Let α be the join of the multisets $msp(\mathbf{x}_1.F_1)^I, \dots, msp(\mathbf{x}_n.F_n)^I$. By definition, $I \models E$ iff (i) $\textsc{op}(\alpha)$ is defined, (ii) $b^I \in \mathbf{Num}$, and (iii) $\textsc{op}(\alpha) \succeq b^I$.

The three conditions hold iff $Q^I_{(\textsc{op},\succeq)}((\mathbf{x}_1.F_1)^I, \dots, (\mathbf{x}_n.F_n)^I, \{b^I\}) = \mathbf{t}$, which is the same as saying that $I \models E^{GQ}$. ∎

Proposition 2 *Let F be an aggregate sentence of σ, let F^{GQ} be the GQ-representation of F, and let $\mathbf{p}$ be a list of predicate constants. For any expansion I of σ_{bg} to σ, $I \models \mathrm{SM}[F; \mathbf{p}]$ (according to Ferraris and Lifschitz) iff $I \models \mathrm{SM}[F^{GQ}; \mathbf{p}]$.*

Proof. In view of Theorem 1, we will prove that for any aggregate formula F, F^* is equivalent to $(F^{GQ})^*$. It is sufficient to prove that, for any aggregate expression

$$\textsc{op}\langle \mathbf{x}_1.F_1(\mathbf{x}_1, \mathbf{p}), \dots, \mathbf{x}_n.F_n(\mathbf{x}_n, \mathbf{p}) \rangle \succeq b$$

that contains no free variables, any interpretation $I = \langle J, X\rangle$, and any subset Y of X, $\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle$ satisfies

$$\begin{array}{r}\text{OP}\langle \mathbf{x}_1.F_1(\mathbf{x}_1,\mathbf{p}),\dots,\mathbf{x}_n.F_n(\mathbf{x}_n,\mathbf{p})\rangle \succeq b\\ \wedge\ \text{OP}\langle \mathbf{x}_1.F^*_1(\mathbf{x}_1,\mathbf{q}),\dots,\mathbf{x}_n.F^*_n(\mathbf{x}_n,\mathbf{q})\rangle \succeq b\end{array} \tag{A4}$$

iff $\langle J, X \cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle$ satisfies

$$\begin{array}{r}Q_{(\text{OP},\succeq)}\ [\mathbf{x}_1]\dots[\mathbf{x}_n][y](F_1(\mathbf{x}_1,\mathbf{p}),\dots,F_n(\mathbf{x}_n,\mathbf{p}),y=b)\wedge\\ Q_{(\text{OP},\succeq)}\ [\mathbf{x}_1]\dots[\mathbf{x}_n][y](F^*_1(\mathbf{x}_1,\mathbf{q}),\dots,F^*_n(\mathbf{x}_n,\mathbf{q}),y=b).\end{array} \tag{A5}$$

By Lemma 8, $\langle J, X\cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle$ satisfies the first (second) conjunctive term of (A4) iff $\langle J, X\cup Y^{\mathbf{p}}_{\mathbf{q}}\rangle$ satisfies the first (second) conjunctive term of (A5). ∎

A.4 Proof of Proposition 3

Lemma 9
For any c-atom (D, C) of σ, let I be an interpretation of σ. I satisfies (D, C) iff $I \models (10)$.

Proof. $I \models (D, C)$ iff $I \cap D \in C$ iff $R \in C$ where $R = I \cap D$. Consider $R_1, \dots, R_n$ such that $R_i = \{\epsilon\}$ if $p_i \in R$ and $R_i = \emptyset$ otherwise. $R = I \cap D$ iff $Q^I_C(R_1,\dots,R_n) = \mathbf{t}$. ∎

Lemma 10
For any c-atom (D, C),

$$Q_C[\,]\dots[\,]D \tag{A6}$$

is equivalent to

$$\bigwedge_{\overline{C}\in\mathcal{P}(D)\setminus C}\ (\bigwedge_{p\in\overline{C}} p \rightarrow \bigvee_{p\in D\setminus\overline{C}} p). \tag{A7}$$

Proof. Consider any subset X of $\{p_1,\dots,p_n\}$. It is sufficient to prove that $X^{\mathbf{p}}_{\mathbf{q}} \models (A6)$ iff $X^{\mathbf{p}}_{\mathbf{q}} \models (A7)$.

From left to right: Assume $X^{\mathbf{p}}_{\mathbf{q}} \models (A6)$. It is follows from Lemma 9 that $X \models (D, C)$. As a result, $X \notin \mathcal{P}(D) \setminus C$. Consider any $\overline{C} \in \mathcal{P}(D) \setminus C$ such that $X^{\mathbf{p}}_{\mathbf{q}} \models \bigwedge_{p\in\overline{C}} q$. It is clear that $\overline{C} \subseteq X$ and $\overline{C} \neq X$ (since $X \notin \mathcal{P}(D) \setminus C$). Consequently,

$$X^{\mathbf{p}}_{\mathbf{q}} \models \bigvee_{p\in D\setminus\overline{C}} q.$$

From right to left: Assume $X^{\mathbf{p}}_{\mathbf{q}} \models (A7)$. Clearly, $X \notin \mathcal{P}(D) \setminus C$. So $X \models (D, C)$ and, by Lemma 9, $X^{\mathbf{p}}_{\mathbf{q}} \models (A6)$. ∎

Lemma 11
For any c-atoms (D, C), (10) is strongly equivalent to (11).

Proof. In view of Theorem 1, it is sufficient to prove that for any list $(q_1,\dots,q_n)$ of new atoms such that $(q_1,\dots,q_n) \le (p_1,\dots,p_n)$,

$$Q_C[\,]\dots[\,](p_1,\dots,p_n) \wedge Q_C[\,]\dots[\,](q_1,\dots,q_n) \tag{A8}$$

is equivalent to

$$\begin{array}{l}\bigwedge_{\overline{C}\in\mathcal{P}(D)\setminus C}\ (\bigwedge_{p\in\overline{C}} p \rightarrow \bigvee_{p\in D\setminus\overline{C}} p)\wedge\\ \qquad\bigwedge_{\overline{C}\in\mathcal{P}(D)\setminus C}\ (\bigwedge_{p\in\overline{C}} q \rightarrow \bigvee_{p\in D\setminus\overline{C}} q).\end{array} \tag{A9}$$

Consider any subset X of $\{p_1, \dots, p_n\}$ and any subset Y of X, we will show that $X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models (A8)$ iff $X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models (A9)$. It follows from Lemma 10 that X satisfies the first conjunctive term of (A8) iff X satisfies the first conjunctive term of (A9). Similarly, $Y^{\mathbf{p}}_{\mathbf{q}}$ satisfies the second conjunctive term of (A8) iff $Y^{\mathbf{p}}_{\mathbf{q}}$ satisfies the second conjunctive term of (A9). ∎

Proposition 3 *For any propositional formula F with c-atoms and any propositional interpretation X, X is an answer set of F iff X is an answer set of $Fer(F)$.*

Proof. Clear from Lemma 11. ∎

A.5 Proof of Proposition 4

Lemma 12
For any dl-program $(\mathcal{T}, \Pi)$, any dl-atom (12) in Π that contains no free variables and any Herbrand interpretation I of $\langle C, P_\Pi \rangle$, $I \models_\mathcal{T}$ (12) iff $I \models$ (16) iff $I \models$ (17).

Proof. It is sufficient to consider a GQ-formula of the form (16) since (17) is equivalent to (16).

By definition, $I \models_\mathcal{T}$ (12) iff $\mathcal{T} \cup \bigcup_{i=1}^{k} A_i(I)$ entails $Query(\mathbf{t})$. Note that $p_i(\mathbf{e}) \in I$ iff $\mathbf{e} \in \{\mathbf{c} \mid I \models p_i(\mathbf{c})\}$ and $p_i(\mathbf{e}) \notin I$ iff $\mathbf{e} \in |I|^{|\mathbf{e}|} \setminus \{\mathbf{c} \mid I \models p_i(\mathbf{c})\}$. Consequently, $A_i(I)$ is the same as $A_i(R_i)$, which is defined as

- $\{S_i(\mathbf{e}) \mid \mathbf{e} \in R_i\}$ if op_i is $\oplus$,
- $\{\neg S_i(\mathbf{e}) \mid \mathbf{e} \in R_i\}$ if op_i is $\odot$,
- $\{\neg S_i(\mathbf{e}) \mid \mathbf{e} \in |I|^{|\mathbf{e}|} \setminus R_i\}$ if op_i is $\ominus$,

where $R_i = \{\mathbf{c} \mid I \models p_i(\mathbf{c})\}$. Clearly, $\mathcal{T} \cup \bigcup_{i=1}^{k} A_i(I)$ entails $Query(\mathbf{t})$ iff $\mathcal{T} \cup \bigcup_{i=1}^{k} A_i(R_i)$ entails $Query(\mathbf{t})$ iff $I \models$ (16). ∎

Given a dl-program $(\mathcal{T}, \Pi)$, we denote $s(\Pi)^X_\mathcal{T}$ as the strong reduct of Π relative to $\mathcal{T}$. For a set X of dl-atoms, we denote X^{sGQ} as the set of atoms obtained from X by identifying each dl-atom as (16) if it is monotonic and (17) otherwise. Similarly, X^{wGQ} is the set of atoms obtained from X by identifying each dl-atom as (17).

Lemma 13
For any dl-program $(\mathcal{T}, \Pi)$, any Herbrand interpretations X, Y of $\langle C, P_\Pi \rangle$ such that $Y \subseteq X$, and any rule $p(\mathbf{t}) \leftarrow B, N$ in Π,

$$Y \models_\mathcal{T} s(p(\mathbf{t}) \leftarrow B, N)^X_\mathcal{T}$$

iff

$$X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models_\mathcal{T} (B^{sGQ} \wedge N^{sGQ})^*(\mathbf{q}) \rightarrow q(\mathbf{t}). \tag{A10}$$

Proof. By Lemma 6, $(N^{sGQ})^*(\mathbf{q})$ is equivalent to N^{sGQ}. We partition B into two sets: the set B_1 of all monotonic dl-atoms and the set B_2 of all non-monotonic dl-atoms.

It is clear from (17) that B_2^{sGQ} is negative on $\mathbf{p}$. By Lemma 6 again, $(B_2^{sGQ})^*(\mathbf{q})$ is equivalent to B_2^{sGQ}. Thus (A10) is equivalent to saying that

$$X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models_\mathcal{T} (B_1^{sGQ})^*(\mathbf{q}) \wedge B_2^{sGQ} \wedge N^{sGQ} \rightarrow q(\mathbf{t}). \tag{A11}$$

Consider two cases.

Case 1: $X \models_{\mathcal{T}} B_2 \wedge N$. Then $s(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ is $p(\mathbf{t}) \leftarrow B_1$. By Lemma 12, $Y \models_{\mathcal{T}} B_1 \rightarrow p(\mathbf{t})$ iff

$$Y^{\mathbf{p}}_{\mathbf{q}} \models_{\mathcal{T}} (B_1^{sGQ})(\mathbf{q}) \rightarrow q(\mathbf{t}). \tag{A12}$$

From $Y \subseteq X$ and that all dl-atoms in B_1 are monotonic, it follows that $Y^{\mathbf{p}}_{\mathbf{q}} \models_{\mathcal{T}} (B_1^{sGQ})(\mathbf{q})$ implies $X \models_{\mathcal{T}} B_1^{sGQ}$. So (A12) is equivalent to

$$X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models_{\mathcal{T}} (B_1^{sGQ})(\mathbf{q}) \wedge B_1^{sGQ} \rightarrow q(\mathbf{t}),$$

which is also equivalent to (A11) under the assumption that $X \models_{\mathcal{T}} B_2 \wedge N$.

Case 2: $X \not\models_{\mathcal{T}} B_2 \wedge N$. Then $s(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ is equivalent to $\top$. On the other hand, by Lemma 12, $X \not\models_{\mathcal{T}} B_2^{sGQ} \wedge N^{sGQ}$. So we get (A11). ∎

Lemma 14

For any dl-program $(\mathcal{T}, \Pi)$ and any Herbrand interpretation X of $\langle C, P_\Pi \rangle$, $X \models \Pi^{sGQ}$ iff $X \models_{\mathcal{T}} s\Pi^X_{\mathcal{T}}$.

Proof. Immediate from the definition of $s\Pi^X_{\mathcal{T}}$, $X \models_{\mathcal{T}} s\Pi^X_{\mathcal{T}}$ iff $X \models_{\mathcal{T}} \Pi$. By Lemma 12, $X \models_{\mathcal{T}} \Pi$ iff $X \models \Pi^{sGQ}$. ∎

Proposition 4 *For any dl-program $(\mathcal{T}, \Pi)$, the weak (strong, respectively) answer sets of $(\mathcal{T}, \Pi)$ are precisely the Herbrand interpretations of $\langle C, P_\Pi \rangle$ that satisfy* SM$[\mathcal{P}^{wGQ}; P_\Pi]$ *(*SM$[\mathcal{P}^{sGQ}; P_\Pi]$*, respectively) relative to $\mathcal{T}$.*

Proof. We only prove the case for strong answer sets. The proof for weak answer sets is similar.

Let X be an Herbrand interpretation of $\langle C, P_\Pi \rangle$. X is a strong answer set of $(\mathcal{T}, \Pi)$ iff

(i) $X \models_{\mathcal{T}} s\Pi^X_{\mathcal{T}}$, and
(ii) no proper subset Y of X satisfies $s\Pi^X_{\mathcal{T}}$ relative to $\mathcal{T}$.

On the other hand, $X \models$ SM$[P^{sGQ}; P_\Pi]$ iff

(i′) $X \models \Pi^{sGQ}$, and
(ii′) X does not satisfy $\exists \mathbf{u}(\mathbf{u} < P_\Pi \wedge (\Pi^{sGQ})^*(\mathbf{u}))$.

By Lemma 14, (i) is equivalent to (i′). Assume (i′). Condition (ii) can be reformulated as: no proper subset Y of X satisfies $s(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ relative to $\mathcal{T}$ for every rule $p(\mathbf{t}) \leftarrow B, N \in \Pi$. Under the assumption (i′), condition (ii′) can be reformulated as: there is no proper subset Y of X such that $X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models_{\mathcal{T}} (B^{sGQ} \wedge N^{sGQ})^*(\mathbf{q}) \rightarrow q(\mathbf{t})$ for every rule $p(\mathbf{t}) \leftarrow B, N$ in Π. By Lemma 13, (ii) is equivalent to (ii′). ∎

A.6 Proof of Proposition 5

Lemma 15

For any dl-program $(\mathcal{T}, \Pi)$, any dl-atom A of the form (12) in Π that contains no free variables, A is monotonic (anti-monotonic) relative to $\mathcal{T}$ iff Q^U_A is motonone (anti-monotone) in $\{1, \ldots, k\}$ for all Herbrand interpretations I of $\langle C, P_\Pi \rangle$.

Proof. We will show the case of monotonic dl-atoms. The case of anti-monotonic dl-atoms is similar.

From left to right: Assume that A is monotonic relative to $\mathcal{T}$. We further assume $Q_A^U(R_1, \dots, R_k) = \mathbf{t}$, where $R_j \subseteq |I|^{|\mathbf{x}_j|}$ for $1 \le j \le k$. Consider any $i \in \{1, \dots, k\}$ and any $R_i' \subseteq |I|^{|\mathbf{x}_i|}$ such that $R_i \subseteq R_i'$, we will show that $Q_A^U(R_1, \dots, R_{i-1}, R_i', R_{i+1}, \dots, R_k) = \mathbf{t}$.

Let I' be the Herbrand interpretation

$$I \cup \{p_i(\mathbf{d}) \mid \mathbf{d} \in R_i' \setminus R_i\},$$

whose signature is the same as I. It is clear that $I \subseteq I'$. Also, by Lemma 12, $I \models_{\mathcal{T}} A$ follows from $Q_A^U(R_1, \dots, R_i, \dots, R_k) = \mathbf{t}$. Since A is monotonic relative to $\mathcal{T}$, $I' \models_{\mathcal{T}} A$ and by Lemma 12, $Q_A^U(R_1, \dots, R_{i-1}, R_i', R_{i+1}, \dots, R_k) = \mathbf{t}$. Since I and I' have the same universe, $Q_A^U(R_1, \dots, R_{i-1}, R_i', R_{i+1}, \dots, R_k) = \mathbf{t}$ follows.

From right to left: Assume that Q_A^U is monotone in $\{1, \dots, k\}$ for all Herbrand interpretations I of $\langle C, P_\Pi \rangle$. Consider any Herbrand interpretations J, J' of $\langle C, P_\Pi \rangle$ such that $J \subseteq J'$ and assume that $J \models_{\mathcal{T}} A$. We will show that $J' \models_{\mathcal{T}} A$.

Let $R_i = \{\mathbf{d} \in |J|^{|\mathbf{x}_i|} \mid (p_i(\mathbf{d}))^J = \mathbf{t}\}$ and $R_i' = \{\mathbf{d} \in |J'|^{|\mathbf{x}_i|} \mid (p_i(\mathbf{d}))^{J'} = \mathbf{t}\}$ for each $1 \le i \le k$. From $J \models_{\mathcal{T}} A$, by Lemma 12, $Q_A^U(R_1, \dots, R_k) = \mathbf{t}$. Since $J \subseteq J'$, it follows that $R_i \subseteq R_i'$ for each $1 \le i \le k$. From the fact that Q_A^U is monotone in $\{1, \dots, k\}$, $Q_A^U(R_1', \dots, R_k') = \mathbf{t}$ follows. By Lemma 12, $J' \models_{\mathcal{T}} A$. ∎

Lemma 16

For any dl-program $(\mathcal{T}, \Pi)$, any Herbrand interpretations X, Y of $\langle C, P_\Pi \rangle$ such that $Y \subseteq X$, and any rule $p(\mathbf{t}) \leftarrow B, N$ in Π,

$$Y \models_{\mathcal{T}} (p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$$

iff

$$X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models (B^{GQ})^*(\mathbf{q}) \wedge (N^{GQ})^*(\mathbf{q}) \rightarrow q(\mathbf{t}). \tag{A13}$$

Proof. We partition B into two sets: the set B_2 of all anti-monotonic dl-atoms and the set B_1 of all remaining dl-atoms. In view of Lemma 15, B_2^{GQ} is a conjunction of GQ-formulas (16) such that Q^U is anti-monotone in all argument positions. By Lemma 6 and Proposition 1 (b), (A13) is equivalent to

$$X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models (B_1^{GQ})^*(\mathbf{q}) \wedge B_2^{GQ} \wedge N^{GQ} \rightarrow q(\mathbf{t}),$$

which is the same as

$$X \cup Y^{\mathbf{p}}_{\mathbf{q}} \models B_1^{GQ}(\mathbf{q}) \wedge B_1^{GQ} \wedge B_2^{GQ} \wedge N^{GQ} \rightarrow q(\mathbf{t}). \tag{A14}$$

Consider two cases.

Case 1: $X \models_{\mathcal{T}} B_2 \wedge N$. $(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ is $p(\mathbf{t}) \leftarrow B_1$. On the other hand, by Lemma 12, $X \models B_2^{GQ} \wedge N^{GQ}$. Since $Y \subseteq X$, $Y^{\mathbf{p}}_{\mathbf{q}} \models B_1^{GQ}(\mathbf{q})$ implies $X \models B_1^{GQ}$. Thus (A14) is equivalent to saying that $Y^{\mathbf{p}}_{\mathbf{q}} \models B_1^{GQ}(\mathbf{q}) \rightarrow q(\mathbf{t})$, which in turn is equivalent to saying that $Y \models B_1^{GQ} \rightarrow p(\mathbf{t})$. By Lemma 12 again, $Y \models B_1^{GQ} \rightarrow p(\mathbf{t})$ iff $Y \models_{\mathcal{T}} B_1 \rightarrow p(\mathbf{t})$.

Case 2: $X \not\models_{\mathcal{T}} B_2 \wedge N$. $(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ is equivalent to $\top$. By Lemma 12, $X \not\models B_2^{GQ} \wedge N^{GQ}$. Thus (A14) follows. ∎

Lemma 17
For any dl-program $(\mathcal{T},\Pi)$, $X \models \Pi^{GQ}$ iff $X \models_{\mathcal{T}} \Pi^X_{\mathcal{T}}$.

Proof. Immediate from the definition of $\Pi^X_{\mathcal{T}}$ that $X \models_{\mathcal{T}} \Pi^X_{\mathcal{T}}$ iff $X \models_{\mathcal{T}} \Pi$. It follows from Lemma 12 that $X \models_{\mathcal{T}} \Pi$ iff $X \models \Pi^{GQ}$. ∎

Proposition 5 *For any dl-program* $(\mathcal{T},\Pi)$*, and any Herbrand interpretation* X *of* $\langle C, P_\Pi\rangle$*,* X *is an answer set of* $(\mathcal{T},\Pi)$ *iff* X *satisfies* SM$[\Pi^{GQ}; P_\Pi]$ *relative to* $\mathcal{T}$*.*

Proof. X is an answer set of $(\mathcal{T},\Pi)$ iff

(i) $X \models_{\mathcal{T}} \Pi^X_{\mathcal{T}}$, and
(ii) no proper subset Y of X satisfies $\Pi^X_{\mathcal{T}}$ relative to $\mathcal{T}$.

On the other hand, $X \models$ SM$[\Pi^{GQ}; P_\Pi]$ iff

(i′) $X \models \Pi^{GQ}$, and
(ii′) X does not satisfy $\exists \mathbf{u}(\mathbf{u} < P_\Pi \wedge (\Pi^{GQ})^*(\mathbf{u}))$.

By Lemma 17, (i) is equivalent to (i′). Assume (i′). Condition (ii) can be reformulated as: no proper subset Y of X satisfies $(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ relative to $\mathcal{T}$ for every rule $p(\mathbf{t}) \leftarrow B, N$ in Π. Under the assumption (i′), condition (ii′) can be reformulated as: there is no proper subset Y of X such that, for every rule $p(\mathbf{t}) \leftarrow B, N$ in Π, $X \cup Y^{\mathbf{p}}_{\mathbf{q}}$ satisfies $(B^{GQ})^*(\mathbf{q}) \wedge (N^{GQ})^*(\mathbf{q}) \rightarrow q(\mathbf{t})$. By Lemma 16, it follows that (ii) is equivalent to (ii′). ∎

A.7 Proof of Proposition 6

Let $(\mathcal{T},\Pi)$ be a dl-program and X an Herbrand interpretation of $\langle C, P_\Pi\rangle$. By $f\Pi^X_{\mathcal{T}}$, we denote the FLP reduct of Π as defined by viewing a dl-program as a HEX program.

Lemma 18
For any dl-program $(\mathcal{T},\Pi)$ such that every occurrence of non-monotonic dl-atoms is in the positive body of a rule, any Herbrand interpretations X, Y of $\langle C, P_\Pi\rangle$ such that $Y \subseteq X$ and any rule $p(\mathbf{t}) \leftarrow B, N$ in Π,

$$Y \models_{\mathcal{T}} f(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$$

iff

$$Y \models_{\mathcal{T}} (p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$$

Proof. We partition B into two sets: the set B_1 of all anti-monotonic dl-atoms and the set B_2 of rest of all dl-atoms.

Consider two cases.
Case 1: $X \models_{\mathcal{T}} B \wedge N$. $f(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ is $p(\mathbf{t}) \leftarrow B, N$. On the other hand, $(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ is $p(\mathbf{t}) \leftarrow B_2$. It is sufficient to show that $Y \models_{\mathcal{T}} B_1 \wedge B_2 \wedge N$ iff $Y \models_{\mathcal{T}} B_2$. From $X \models B \wedge N$, it follows that $X \models B_1$ and $X \models N$. Since B_1 contains only anti-monotonic dl-atoms, $Y \models B_1$ follows from $X \models B_1$. Since N contains only negation of monotonic dl-atoms, $Y \models N$ follows from $X \models N$.

Case 2: $X \not\models_{\mathcal{T}} B \wedge N$. both $(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ and $f(p(\mathbf{t}) \leftarrow B, N)^X_{\mathcal{T}}$ are equivalent to $\top$. ∎

Lemma 19
For any dl-program $(\mathcal{T},\Pi)$ such that Π is a ground program and every occurrence of non-monotonic dl-atoms is in the positive body of a rule, $X \models \Pi^X_{\mathcal{T}}$ iff $X \models_{\mathcal{T}} f\Pi^X_{\mathcal{T}}$.

Proof. Immediate from the definition of $f\Pi^X_{\mathcal{T}}$ that $X \models_{\mathcal{T}} f\Pi^X_{\mathcal{T}}$ iff $X \models_{\mathcal{T}} \Pi$. It follows from Lemma 12 that $X \models_{\mathcal{T}} \Pi$ iff $X \models \Pi^{GQ}$. ∎

Proposition 6 *For any dl-program $(\mathcal{T},\Pi)$, and any Herbrand interpretation X of $\langle C, P_\Pi\rangle$, if every occurrence of non-monotonic dl-atoms is in the positive body of a rule, then X is an answer set of $(\mathcal{T},\Pi)$ in the sense of (Fink and Pearce 2010) iff X is an answer set of $(\mathcal{T},\Pi)$ in our sense.*

Proof. X is an answer set of $(\mathcal{T},\Pi)$ according to Fink and Pearce iff

(i) $X \models_{\mathcal{T}} f\Pi^X_{\mathcal{T}}$, and
(ii) no proper subset Y of X satisfies $f\Pi^X_{\mathcal{T}}$ relative to $\mathcal{T}$.

On the other hand, X satisfies FLP$[\Pi^{GQ}; P_\Pi]$ iff

(i′) $X \models_{\mathcal{T}} \Pi^X_{\mathcal{T}}$, and
(ii′) no proper subset Y of X satisfies $\Pi^X_{\mathcal{T}}$ relative to $\mathcal{T}$.

By Lemma 19, (i) is equivalent to (i′). By Lemma 18, (ii) is equivalent to (ii′). ∎

A.8 Proof of Theorem 1

Lemma 20
Let F be an arbitrary formula with generalized quantifiers, let G be a subformula of F, and let F' be the formula obtained from F by replacing G with a formula G'. Then formula $(G \leftrightarrow G') \rightarrow (F \leftrightarrow F')$ is logically valid.

Proof. By induction on F.

Case 1: F is an atomic formula. G is the same as F and G' is the same as F'. So it is clear that $(G \leftrightarrow G') \rightarrow (F \leftrightarrow F')$ is logically valid.

Case 2: F is $Q[\mathbf{x}_1]\dots[\mathbf{x}_k](F_1(\mathbf{x}_1),\dots,F_k(\mathbf{x}_k))$. Assume that G is a subformula of some F_i. By I.H.,

$$(G \leftrightarrow G') \rightarrow (F_i \leftrightarrow F'_i)$$

is logically valid, where F'_i is defined similarly. Consequently, $(G \leftrightarrow G') \rightarrow (F \leftrightarrow F')$ is logically valid. ∎

Lemma 21
Let H be an arbitrary formula with generalized quantifiers, let F be a subformula of H, and let H' be the formula obtained from H by replacing F with a formula G. Then formula

$$(F \leftrightarrow G) \wedge (F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u})) \rightarrow (H^*(\mathbf{u}) \leftrightarrow (H')^*(\mathbf{u}))$$

is logically valid.

Proof. By induction on H. ∎

Lemma 22
If the formula

$$(\mathbf{u} \leq \mathbf{p}) \rightarrow (F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u})) \tag{A15}$$

is logically valid, then F is strongly equivalent to G.

Proof. We will show that

$$H \wedge \neg\exists\mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge H^*(\mathbf{u})) \tag{A16}$$

is equivalent to

$$H' \wedge \neg\exists\mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge (H')^*(\mathbf{u})). \tag{A17}$$

Since (A15) is logically valid, $F^*(\mathbf{p})$ is equivalent to $G^*(\mathbf{p})$. By Lemma 3, it follows that F is equivalent to G. Consequently, by Lemma 20, H is equivalent to H'. By Lemma 21, it follows that the second conjunctive terms of (A16) and (A17) are equivalent to each other. ∎

Lemma 23
For two formula F and G with generalized quantifiers, if F is strongly equivalent to G, then

$$(\mathbf{u} \leq \mathbf{p}) \rightarrow (F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u})) \tag{A18}$$

is logically valid.

Proof. Let E stand for $F \leftrightarrow G$, and let E' be $F \leftrightarrow F$. Since F is strongly equivalent to G, it follows that SM$[E \leftrightarrow C]$ is equivalent to SM$[E' \leftrightarrow C]$, where C is the conjunction of choice formulas $\forall\mathbf{x}(p(\mathbf{x}) \vee \neg p(\mathbf{x}))$ for all $p \in \mathbf{p}$. We can simplify SM$[E \leftrightarrow C]$ as

$$\begin{array}{ll} & \text{SM}[E \leftrightarrow C] \\ \Leftrightarrow & (E \leftrightarrow C) \wedge \neg\exists\mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge E \wedge (E^*(\mathbf{u}) \leftrightarrow (\mathbf{p} \leq \mathbf{u}))) \\ \Leftrightarrow & E \wedge \neg\exists\mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge (E^*(\mathbf{u}) \leftrightarrow (\mathbf{p} \leq \mathbf{u}))) \\ \Leftrightarrow & E \wedge \neg\exists\mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge \neg E^*(\mathbf{u})) \\ \Leftrightarrow & E \wedge \neg\exists\mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge \neg(F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u}))) \\ = & (F \leftrightarrow G) \wedge \forall\mathbf{u}((\mathbf{u} < \mathbf{p}) \rightarrow (F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u}))) \\ \Leftrightarrow & \forall\mathbf{u}(\mathbf{u} \leq \mathbf{p} \rightarrow (F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u}))). \end{array}$$

On the other hand, SM$[E' \leftrightarrow C]$ is equivalent to

$$\forall\mathbf{u}(\mathbf{u} \leq \mathbf{p} \rightarrow (F^*(\mathbf{u}) \leftrightarrow F^*(\mathbf{u}))),$$

which is logically valid. Consequently, (A18) is logically valid. ∎

Theorem 1 *Let F and G be GQ formulas, let $\mathbf{p}$ be the list of all predicate constants occurring in F or G and let $\mathbf{u}$ be a list of distinct predicate variables corresponding to $\mathbf{p}$. Formulas F and G are strongly equivalent to each other iff the formula*

$$(\mathbf{u} \leq \mathbf{p}) \rightarrow (F^*(\mathbf{u}) \leftrightarrow G^*(\mathbf{u}))$$

is logically valid.

Proof. The "if" part of the theorem follows from Lemma 22 while the other direction follows from Lemma 23. ∎

A.9 Proofs of Theorems 2 and 3

Lemma 24

If every occurrence of every predicate constant from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F, then

$$(\mathbf{u}_1,\mathbf{u}_2) \leq (\mathbf{p}_1,\mathbf{p}_2) \rightarrow (F^*(\mathbf{u}_1,\mathbf{u}_2) \leftrightarrow F^*(\mathbf{u}_1,\mathbf{p}_2)) \tag{A19}$$

is logically valid.

Proof. By induction on F.

Case 1: F is an atomic formula.

- If F is of the form $p(\mathbf{t})$ then $p \notin \mathbf{p}_2$ since every occurrence of every predicate constant from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F. Clearly, $F^*(\mathbf{u}_1,\mathbf{u}_2)$ is the same as $F^*(\mathbf{u}_1,\mathbf{p}_2)$.
- Otherwise, it is clear that both $F^*(\mathbf{u}_1,\mathbf{u}_2)$ and $F^*(\mathbf{u}_1,\mathbf{p}_2)$ are the same as F.

Case 2: F is of the form (3).

- If F is negative on $\mathbf{p}$, by Lemma 6, both $F^*(\mathbf{u}_1,\mathbf{u}_2)$ and $F^*(\mathbf{u}_1,\mathbf{p}_2)$ are equivalent to F.
- Otherwise, F is not negative on $\mathbf{p}$. Consider any F_i where $i \in \{1,\dots,k\}$. Note that every occurrence of every predicate constant from $\mathbf{p}_2$ in F is contained in a subformula of F that is negative on $\mathbf{p}$. Since F is not negative on $\mathbf{p}$, such subformula can not be F. It follows that every occurrence of every predicate constant from $\mathbf{p}_2$ in F_i is $\mathbf{p}$-negated in F_i. By I.H.,

$$(\mathbf{u}_1,\mathbf{u}_2) \leq (\mathbf{p}_1,\mathbf{p}_2) \rightarrow (F_i^*(\mathbf{u}_1,\mathbf{u}_2) \leftrightarrow F_i^*(\mathbf{u}_1,\mathbf{p}_2))$$

  is logically valid. Consequently, (A19) is logically valid.

∎

Lemma 25

Let $\mathbf{p}$ be the list of all intensional predicates and let $\mathbf{p}_1$, $\mathbf{p}_2$ be a partition of $\mathbf{p}$, and let $\mathbf{u}_1$, $\mathbf{u}_2$ be disjoint lists of distinct predicate variables of the same length as $\mathbf{p}_1$, $\mathbf{p}_2$ respectively.

(a) If every semi-positive occurrence of every predicate constant from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F, then

$$((\mathbf{u}_1,\mathbf{u}_2) \leq (\mathbf{p}_1,\mathbf{p}_2)) \wedge F^*(\mathbf{u}_1,\mathbf{p}_2) \rightarrow F^*(\mathbf{u}_1,\mathbf{u}_2)$$

is logically valid.

(b) If every semi-negative occurrence of every predicate constant from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F, then

$$((\mathbf{u}_1,\mathbf{u}_2) \leq (\mathbf{p}_1,\mathbf{p}_2)) \wedge F^*(\mathbf{u}_1,\mathbf{u}_2) \rightarrow F^*(\mathbf{u}_1,\mathbf{p}_2)$$

is logically valid.

Proof. Both parts are proven simultaneously by induction on F.

Case 1: F is an atomic formula $p_i(\mathbf{t})$.

(a) Since every semi-positive occurrence of every predicate constant from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F, predicate constant p_i is not in $\mathbf{p}_2$, so $F^*(\mathbf{u}_1,\mathbf{p}_2)$ is the same as $F^*(\mathbf{u}_1,\mathbf{u}_2)$.

(b) Clear from $(\mathbf{u}_1,\mathbf{u}_2) \leq (\mathbf{p}_1,\mathbf{p}_2)$.

Case 2: F is $t_1 = t_2$ or $\bot$. Clear since both $F^*(\mathbf{u}_1, \mathbf{p}_2)$ and $F^*(\mathbf{u}_1, \mathbf{u}_2)$ are the same as F.

Case 3: F is of the form (3). Without loss of generality, we partition the set of all argument positions of Q into three sets: the set of monotone argument positions *Mon*, the set of anti-monotone argument positions *Anti* and the rest of argument positions *Mixed*.

(a) If F is negative on $\mathbf{p}$, by Lemma 6, both $F^*(\mathbf{u}_1, \mathbf{u}_2)$ and $F^*(\mathbf{u}_1, \mathbf{p}_2)$ are equivalent to F. Otherwise, assume $(\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)$.

- Consider any F_i, where $i \in$ *Mon*. Note that every semi-positive occurrence of predicates from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F. Since F is not negative on $\mathbf{p}$, such subformula can not be F. It follows that every semi-positive occurrence of predicates from $\mathbf{p}_2$ in F_i is $\mathbf{p}$-negated in F_i. By I.H. (a),
$$((\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)) \wedge F_i^*(\mathbf{u}_1, \mathbf{p}_2) \rightarrow F_i^*(\mathbf{u}_1, \mathbf{u}_2) \tag{A20}$$
is logically valid.
- Consider any F_i, where $i \in$ *Mixed*. Note that every semi-positive occurrence of predicates from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F. Since F is not negative on $\mathbf{p}$, such subformula can not be F. It follows that every occurrence of predicates from $\mathbf{p}_2$ in F_i is $\mathbf{p}$-negated in F_i. By Lemma 24,
$$((\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)) \rightarrow (F_i^*(\mathbf{u}_1, \mathbf{p}_2) \leftrightarrow F_i^*(\mathbf{u}_1, \mathbf{u}_2)) \tag{A21}$$
is logically valid.
- Consider any F_i, where $i \in$ *Anti*. Note that every semi-positive occurrence of predicates from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F. Since F is not negative on $\mathbf{p}$, such subformula can not be F. It follows that every semi-negative occurrence of predicates from $\mathbf{p}_2$ in F_i is $\mathbf{p}$-negated in F_i. By I.H. (b),
$$((\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)) \wedge F_i^*(\mathbf{u}_1, \mathbf{u}_2) \rightarrow F_i^*(\mathbf{u}_1, \mathbf{p}_2) \tag{A22}$$
is logically valid.

Since Q is monotone in *Mon* and anti-monotone in *Anti*, by Lemma 5 (a) and Lemma 5 (b),
$$((\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)) \wedge F^*(\mathbf{u}_1, \mathbf{p}_2) \rightarrow F^*(\mathbf{u}_1, \mathbf{u}_2)$$
follows from (A20), (A21) and (A22).

(b) If F is negative on $\mathbf{p}$, by Lemma 6, both $F^*(\mathbf{u}_1, \mathbf{u}_2)$ and $F^*(\mathbf{u}_1, \mathbf{p}_2)$ are equivalent to F. Otherwise, assume $(\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)$.

- Consider any F_i, where $i \in$ *Mon*. Note that every semi-negative occurrence of predicates $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F_i. Since F is not negative on $\mathbf{p}$, such subformula can not be F. It follows that every semi-negative occurrence of predicates from $\mathbf{p}_2$ in F_i is $\mathbf{p}$-negated in F_i. By I.H. (b),
$$((\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)) \wedge F_i^*(\mathbf{u}_1, \mathbf{u}_2) \rightarrow F_i^*(\mathbf{u}_1, \mathbf{p}_2) \tag{A23}$$
is logically valid.
- Consider any F_i, where $i \in$ *Mixed*. Note that every semi-negative occurrence of predicates from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F. Since F is not negative on $\mathbf{p}$, such subformula can not be F. It follows that every occurrence of predicates from $\mathbf{p}_2$ in F_i is $\mathbf{p}$-negated in F_i. By Lemma 24,
$$((\mathbf{u}_1, \mathbf{u}_2) \leq (\mathbf{p}_1, \mathbf{p}_2)) \rightarrow (F_i^*(\mathbf{u}_1, \mathbf{p}_2) \leftrightarrow F_i^*(\mathbf{u}_1, \mathbf{u}_2)) \tag{A24}$$

is logically valid.

- Consider any F_i where $i \in$ *Anti*. Note that every semi-negative occurrence of predicates from $\mathbf{p}_2$ in F is $\mathbf{p}$-negated in F. Since F is not negative on $\mathbf{p}$, such subformula can not be F. It follows that every semi-positive occurrence of predicates from $\mathbf{p}_2$ in F_i is $\mathbf{p}$-negated in F_i. By I.H. (a),

$$((\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)) \wedge F_i^*(\mathbf{u}_1, \mathbf{p}_2) \rightarrow F_i^*(\mathbf{u}_1, \mathbf{u}_2) \quad \text{(A25)}$$

is logically valid.

Since Q is monotone in *Mon* and anti-monotone in *Anti*, by Lemma 5(a) and Lemma 5(b),

$$((\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)) \wedge F^*(\mathbf{u}_1, \mathbf{u}_2) \rightarrow F^*(\mathbf{u}_1, \mathbf{p}_2)$$

follows from (A23), (A24) and (A25). ∎

Lemma 26
Let $\mathbf{p}_1$, $\mathbf{p}_2$ be disjoint lists of distinct predicate constants such that $\mathrm{DG}_{\mathbf{p}_1\mathbf{p}_2}[F]$ has no edges from predicate constants in $\mathbf{p}_1$ to predicate constants in $\mathbf{p}_2$, and let $\mathbf{u}_1$, $\mathbf{u}_2$ be disjoint lists of distinct predicate variables of the same length as $\mathbf{p}_1$, $\mathbf{p}_2$ respectively. Formula

$$((\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)) \wedge F^*(\mathbf{u}_1, \mathbf{u}_2) \rightarrow F^*(\mathbf{u}_1, \mathbf{p}_2)$$

is logically valid.

Proof. By induction on F.
Case 1: F is an atomic formula.

- If F is of the form $p(\mathbf{t})$, where $p \in \mathbf{p}_1$, then both $F^*(\mathbf{u}_1, \mathbf{u}_2)$ and $F^*(\mathbf{u}_1, \mathbf{p}_2)$ are $u(\mathbf{t})$.
- If F is of the form $p(\mathbf{t})$ where $p \in \mathbf{p}_2$, clear from Lemma 2 and the assumption $\mathbf{u}_2 \le \mathbf{p}_2$.
- Otherwise, $F^*(\mathbf{u}_1, \mathbf{u}_2)$ and $F^*(\mathbf{u}_1, \mathbf{p}_2)$ are the same as F.

Case 2: F is of the form (3). Without loss of generality, we partition the set of all argument positions of Q into three sets: the set of monotone argument positions *Mon*, the set of anti-monotone argument positions *Anti*, and the rest of argument positions *Mixed*.

SubCase 2.1: F_i is negative on $\mathbf{p}_1$ for each $i \in$ *Mon* $\cup$ *Mixed*. Then F is negative on $\mathbf{p}_1$. Assuming

$$((\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)) \wedge F^*(\mathbf{u}_1, \mathbf{u}_2),$$

by Lemma 2, we get F, or equivalently $F^*(\mathbf{p}_1, \mathbf{p}_2)$, and by Lemma 6, we get $F^*(\mathbf{u}_1, \mathbf{p}_2)$.
SubCase 2.2: F_i is not negative on $\mathbf{p}_1$ for some $i \in$ *Mon* $\cup$ *Mixed*.

— Consider any F_j where $j \in$ *Anti* $\cup$ *Mixed*. Since $\mathrm{DG}_{\mathbf{p}_1\mathbf{p}_2}[F]$ has no edges from predicates in $\mathbf{p}_1$ to predicates in $\mathbf{p}_2$, every semi-positive occurrence of predicates from $\mathbf{p}_2$ in F_j is $\mathbf{p}$-negated in F_j. By Lemma 25 (a),

$$((\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)) \wedge F_j^*(\mathbf{u}_1, \mathbf{p}_2) \rightarrow F_j^*(\mathbf{u}_1, \mathbf{u}_2) \quad \text{(A26)}$$

is logically valid.

— Consider any F_j where $j \in$ *Mon* $\cup$ *Mixed*. Since the occurrence of F_j is strictly positive in F, $\mathrm{DG}_{\mathbf{p}_1\mathbf{p}_2}[F_j]$ is a subgraph of $\mathrm{DG}_{\mathbf{p}_1\mathbf{p}_2}[F]$. It follows that $\mathrm{DG}_{\mathbf{p}_1\mathbf{p}_2}[F_j]$ has no edges from predicate constants in $\mathbf{p}_1$ to predicate constants in $\mathbf{p}_2$. By I.H.,

$$((\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)) \wedge F_j^*(\mathbf{u}_1, \mathbf{u}_2) \rightarrow F_j^*(\mathbf{u}_1, \mathbf{p}_2) \quad \text{(A27)}$$

is logically valid.

From (A26) and (A27), it follows that

$$((\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)) \rightarrow (F_i^*(\mathbf{u}_1, \mathbf{u}_2) \leftrightarrow F_i^*(\mathbf{u}_1, \mathbf{p}_2)) \tag{A28}$$

is logically valid for every $i \in \mathit{Mixed}$.

Assume $(\mathbf{u}_1, \mathbf{u}_2) \le (\mathbf{p}_1, \mathbf{p}_2)$, and

$$Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1^*(\mathbf{u}_1, \mathbf{u}_2), \dots, F_k^*(\mathbf{u}_1, \mathbf{u}_2)). \tag{A29}$$

Let F' be the formula obtained from (A29) by replacing $F_i^*(\mathbf{u}_1, \mathbf{u}_2)$ with $F_i^*(\mathbf{u}_1, \mathbf{p}_2)$ for every $i \in \mathit{Mon} \cup \mathit{Mixed}$. Since Q is monotone in *Mon*, by Lemma 5 (a), formula F' follows from (A27) and (A28). Since Q is anti-monotone in *Anti*, by Lemma 5 (b),

$$Q[\mathbf{x}_1] \dots [\mathbf{x}_k](F_1^*(\mathbf{u}_1, \mathbf{p}_2), \dots, F_k^*(\mathbf{u}_1, \mathbf{p}_2))$$

follows from F' and (A26).

∎

Lemma 27

For any GQ formula F and any nonempty set Y of intensional predicates, there exists a subset Z of Y such that

(a) Z is a loop of F, and
(b) the predicate dependency graph of F has no edges from predicate constants in Z to predicate constants in $Y \setminus Z$.

The proof is essentially the same as the proof of Lemma 4 in (Ferraris et al. 2006).

Theorem 2 *Let F be a GQ sentence, and let $\mathbf{p}$ be a tuple of distinct predicate constants. If $\mathbf{l}^1, \dots, \mathbf{l}^n$ are all the loops of F relative to $\mathbf{p}$ then*

$$\text{SM}[F; \mathbf{p}] \quad \textit{is equivalent to} \quad \text{SM}[F; \mathbf{l}^1] \wedge \dots \wedge \text{SM}[F; \mathbf{l}^n].$$

Proof. It is sufficient to prove the logical validity of the formula

$$\begin{aligned} &\exists \mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge F^*(\mathbf{u})) \\ &\quad \leftrightarrow \exists \mathbf{u}^1((\mathbf{u}^1 < \mathbf{l}^1) \wedge F^*(\widetilde{\mathbf{u}^1})) \\ &\qquad \vee \dots \vee \exists \mathbf{u}^n((\mathbf{u}^n < \mathbf{l}^n) \wedge F^*(\widetilde{\mathbf{u}^n})), \end{aligned}$$

where each $\mathbf{u}^i$ is the part of $\mathbf{u}$ that corresponds to the part $\mathbf{l}^i$ of $\mathbf{p}$, and $\widetilde{\mathbf{u}^i}$ is the list of symbols obtained from $\mathbf{p}$ by replacing every intensional predicate p that belongs to $\mathbf{l}^i$ with the corresponding predicate variable u.

From right to left: Clear.

From left to right: Assume $\exists \mathbf{u}((\mathbf{u} < \mathbf{p}) \wedge F^*(\mathbf{u}))$ and take $\mathbf{u}$ such that $(\mathbf{u} < \mathbf{p}) \wedge F^*(\mathbf{u})$. Consider several cases, each corresponding to a nonempty subset Y of $\mathbf{p}$. The assumption characterizing each case is that $u < p$ for each member p of $\mathbf{p}$ that belongs to Y, and that $u = p$ for each p that does not belong to Y. By Lemma 27, there is a loop $\mathbf{l}^i$ of F that is contained in Y such that the dependency graph $\text{DG}_{\mathbf{p}}[F]$ has no edges from predicate constants in $\mathbf{l}^i$ to predicate constants in $Y \setminus \mathbf{l}^i$. Since $\mathbf{l}^i$ is contained in Y, from the fact that $u < p$ for each p in Y we can conclude that

$$\mathbf{u}^i < \mathbf{l}^i. \tag{A30}$$

Let $\mathbf{u}'$ be the list of symbols obtained from $\mathbf{p}$ by replacing every member p that belongs to Y with the corresponding variable u. Under the assumption characterizing each case, $\mathbf{u} = \mathbf{u}'$, so that $F^*(\mathbf{u}) \leftrightarrow F^*(\mathbf{u}')$. Consequently, we can derive $F^*(\mathbf{u}')$. It follows from Lemma 26 that the formula

$$(\mathbf{u}' \leq \mathbf{p}) \wedge F^*(\mathbf{u}') \rightarrow F^*(\widetilde{\mathbf{u}^i})$$

is logically valid, so that we further conclude that $F^*(\widetilde{\mathbf{u}^i})$. In view of (A30), it follows that $\exists \mathbf{u}^i((\mathbf{u}^i < \mathbf{l}^i) \wedge F^*(\widetilde{\mathbf{u}^i}))$. ∎

Theorem 3 *Let F, G be GQ sentences, and let $\mathbf{p}$, $\mathbf{q}$ be disjoint tuples of distinct predicate constants. If*

- *each strongly connected component of* $\mathrm{DG}_{\mathbf{pq}}[F \wedge G]$ *is a subset of* $\mathbf{p}$ *or a subset of* $\mathbf{q}$,
- *F is negative on* $\mathbf{q}$, *and*
- *G is negative on* $\mathbf{p}$

then

$$\mathrm{SM}[F \wedge G; \mathbf{pq}] \quad \textit{is equivalent to} \quad \mathrm{SM}[F; \mathbf{p}] \wedge \mathrm{SM}[G; \mathbf{q}].$$

Proof. Same as the proof in (Ferraris et al. 2009). ∎

A.10 Proof of Theorem 4

Theorem 4 *For any GQ formula F in Clark normal form that is tight on* $\mathbf{p}$, $\mathrm{SM}[F; \mathbf{p}]$ *is equivalent to the completion of F relative to* $\mathbf{p}$.

Proof. Since F is tight on $\mathbf{p}$, the loops of F relative to $\mathbf{p}$ are singletons only. By Theorem 3, $\mathrm{SM}[F; \mathbf{p}]$ is equivalent to the conjunction of $\mathrm{SM}[\forall \mathbf{x}_i(G_i(\mathbf{x}_i) \rightarrow p_i(\mathbf{x}_i)); p_i]$ for each $p_i \in \mathbf{p}$, which, under the assumption F, is equivalent to

$$\forall u_i(u_i < p_i \rightarrow \exists \mathbf{x}_i(G_i^*(\mathbf{x}_i) \wedge \neg u_i(\mathbf{x}_i))). \tag{A31}$$

Since F is tight on $\mathbf{p}$, it follows that $G_i(\mathbf{x}_i)$ is negative on p_i. By Lemma 6, $G_i^*(\mathbf{x}_i)$ is equivalent to $G_i(\mathbf{x}_i)$. Consequently, (A31) is equivalent to

$$\forall u_i(u_i < p_i \rightarrow \exists \mathbf{x}_i(G_i(\mathbf{x}_i) \wedge \neg u_i(\mathbf{x}_i))). \tag{A32}$$

It is sufficient to prove that, under the assumption

$$\forall \mathbf{x}_i(G_i(\mathbf{x}_i) \rightarrow p_i(\mathbf{x}_i)), \tag{A33}$$

formula (A32) is equivalent to $\forall \mathbf{x}_i(p_i(\mathbf{x}_i) \rightarrow G_i(\mathbf{x}_i))$.

From left to right: Assume (A32) and, for the sake of contradiction, assume that there exists $\mathbf{x}$ such that

$$p_i(\mathbf{x}) \wedge \neg G_i(\mathbf{x}). \tag{A34}$$

Take u_i such that

$$\forall \mathbf{x}_i(u_i(\mathbf{x}_i) \leftrightarrow G_i(\mathbf{x}_i)). \tag{A35}$$

From (A33), (A34), and (A35), we conclude $u_i < p_i$. From (A32), $\exists \mathbf{x}_i(G_i(\mathbf{x}_i) \wedge \neg u_i(\mathbf{x}_i)))$ follows, which contradicts with (A35).

From right to left: Assume $\forall \mathbf{x}_i(G_i(\mathbf{x}_i) \leftrightarrow p_i(\mathbf{x}_i))$. We further assume that $u_i < p_i$ for some

u_i. From $u_i < p_i$, $\exists \mathbf{x}_i(p_i(\mathbf{x}_i) \land \neg u_i(\mathbf{x}_i))$ follows. Consequently, $\exists \mathbf{x}_i(G_i(\mathbf{x}_i) \land \neg u_i(\mathbf{x}_i))$ follows. $\blacksquare$