TLP **16** (1): 111–138, 2016. © Cambridge University Press 2015
doi:10.1017/S1471068415000101 First published online 13 August 2015

*Characterization of logic program revision as an extension of propositional revision**

NICOLAS SCHWIND

Transdisciplinary Research Integration Center, 2-1-2 Hitotsubashi, Chiyoda-ku, 101-8430 Tokyo, Japan
(*e-mail:* schwind@nii.ac.jp)

KATSUMI INOUE

National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku 101-8430 Tokyo, Japan
(*e-mail:* inoue@nii.ac.jp)

submitted 8 June 2014; revised 12 May 2015; accepted 22 June 2015

Abstract

We address the problem of belief revision of logic programs (LPs), i.e., how to incorporate to a LP P a new LP Q. Based on the structure of SE interpretations, Delgrande *et al.* (2008. *Proc. of the 11th International Conference on Principles of Knowledge Representation and Reasoning (KR'08)*, 411–421; 2013b. *Proc. of the 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'13)*, 264–276) adapted the well-known AGM framework (Alchourrón *et al.* 1985. *Journal of Symbolic Logic 50,* 2, 510–530) to LP revision. They identified the rational behavior of LP revision and introduced some specific operators. In this paper, a constructive characterization of all rational LP revision operators is given in terms of orderings over propositional interpretations with some further conditions specific to SE interpretations. It provides an intuitive, complete procedure for the construction of all rational LP revision operators and makes easier the comprehension of their semantic and computational properties. We give a particular consideration to LPs of very general form, i.e., the *generalized* logic programs (GLPs). We show that every rational GLP revision operator is derived from a propositional revision operator satisfying the original AGM postulates. Interestingly, the further conditions specific to GLP revision are independent from the propositional revision operator on which a GLP revision operator is based. Taking advantage of our characterization result, we embed the GLP revision operators into structures of Boolean lattices, that allow us to bring to light some potential weaknesses in the adapted AGM postulates. To illustrate our claim, we introduce and characterize axiomatically two specific classes of (rational) GLP revision operators which arguably have a drastic behavior. We additionally consider two more restricted forms of LPs, i.e., the *disjunctive* logic programs (DLPs) and the *normal* logic programs (NLPs) and adapt our characterization result to *disjunctive* logic program and *normal* logic program revision operators.

KEYWORDS: belief revision, logic programming, characterization theorems

* This is a revised and full version (including proofs of propositions given in the online appendix of the paper) of Schwind and Inoue (2013).

1 Introduction

Logic programs (LPs) under the answer set semantics are well-suited for modeling problems which involve common sense reasoning (e.g., biological networks, diagnosis, planning, etc.) Due to the dynamic nature of our environment, beliefs represented through an LP P are subject to change, i.e., because one wants to incorporate to it a new LP Q. Since there is no unique, consensual procedure to revise a set of beliefs, Alchourrón *et al.* (1985) introduced a set of desirable principles w.r.t. belief change called *AGM postulates*. Katsuno and Mendelzon (1992) adapted these principles to the case of propositional logic, distinguished two kind of change operations, i.e., *revision* and *update* (Katsuno and Mendelzon 1991), and characterized axiomatically each one of these change operations by a set of so-called *KM postulates*. Revision consists in incorporating a new information into a database that represents a static world, i.e., new and old beliefs describe the same situation but new ones are more reliable. In the case of update, the underlying world evolves w.r.t the occurrence of some events i.e., new and old beliefs describe two different states of the world.

Our interests focus here on the problem of revision of LPs. Most of works dealing with belief change in logic programming are concerned with rule-based update (Zhang and Foo 1997; Alferes *et al.* 2000; Eiter *et al.* 2002; Sakama and Inoue 2003; Zhang 2006; Delgrande *et al.* 2007), and they do not lie into the AGM framework, particularly due to their syntactic essence.

Indeed, given the non-monotonic nature of LPs the AGM/KM postulates can not be directly applied to LPs (Eiter *et al.* 2002). However, the notion of *SE models* introduced by Turner (2003) provided a monotonic semantical characterization of LPs, which is more expressive than the answer set semantics. Initially, SE models were used to characterize the strong equivalence between LPs (Lifschitz *et al.* 2001): precisely, two LPs have the same set of SE models if and only if they are strongly equivalent, that is to say, they admit the same answer sets, and will still do even after adding any arbitrary set of rules to them.

Based on these structures, Delgrande *et al.* (2008, 2013b) adapted the AGM/KM postulates in the context of logic programming. They focused on the revision of LPs, i.e., they proposed several revision operators and investigated their properties w.r.t. the adapted postulates. Slota and Leite (2010, 2014) exploited the same idea for *update* of LPs by adapting the KM postulates in a similar way. These semantical-based belief change operations (revision and update) changed the focus from the dynamic evolution of a syntactic, rule-based representation of beliefs previously proposed in the literature to the evolution of its semantic content; these works covered a serious drawback in the field of belief revision in logic programming. In the context of update, Slota and Leite also proposed a *constructive representation* of such update operators. Such a result provides a sound and complete model-theoretic construction of the rational LP update operators, i.e., a "generic recipe" to construct all operators that fully satisfy the adaptation of the AGM/KM postulates to LPs. It is indeed crucial when defining a logical operator in an axiomatic way to give an intuitive constructive characterization of it in order to aid the analysis of its semantic and computational properties.

In this paper, we give a particular consideration to the revision of *generalized* logic programs (GLPs) (Inoue and Sakama 1998) which are of very general form. Revising a GLP P by an other GLP Q should result in a new GLP that satisfy the adapted set of AGM postulates. We provide a characterization of the set of all GLP revision operators by associating each GLP with a certain structure, called *GLP parted assignment*, which consists of a pair of assignments that are independent from each other. Interestingly, the first one, called here *LP faithful assignment*, is similar to the structure of faithful assignment defined in Katsuno and Mendelzon (1992) and used to characterize the (rational) KM revision operators in the propositional setting; the second one, called here *well-defined assignment*, can be defined independently from the first one. As a consequence, the benefit of our approach is that:

(i) every rational LP revision operator $\star$ can be derived from a propositional revision operator $\circ$ satisfying the KM postulates, with some additional conditions that are independent from $\circ$;
(ii) there is a one-to-one correspondence between the set of rational LP revision operators and the set of all pairs of such assignments.

Our characterization makes the refined analysis of LP revision operators easier. Indeed, we can embed the GLP revision operators into structures of Boolean lattices, that allows us to bring out some potential weaknesses in the original postulates and pave the way for the discrimination of some rational GLP revision operators.

The next section introduces some preliminaries about belief revision in propositional logic. We provide in Section 3 some necessary background on GLPs, and we also introduce the notion on LP revision, an axiomatic characterization of GLP revision operators, and some preliminary results. Section 4 provides our main result, i.e., a constructive characterization of the axiomatic description of the GLP revision operators. We formally compare our characterization result with another recent one proposed in Delgrande *et al.* (2013a); the benefit of our approach is that our construction is one-to-one, as opposite to Delgrande *et al.*'s one. In Section 5, we partition the class of GLP revision operators into subclasses of Boolean lattices, then we introduce and axiomatically characterize two specific classes of (rational) GLP revision operators, i.e., the *skeptical* and *brave* GLP revision operators, and lastly we provide some complexity results which are direct consequences of existing results in the propositional case. In Section 6, we consider the revision of more restricted forms of LPs, i.e., the disjunctive logic programs (DLPs) and normal logic programs (NLPs). We adapt our characterization result to DLP revision operators and NLP revision operators. Though DLP revision operators and NLP revision operators can also be viewed as extensions of propositional revision operators, in contrast with GLP revision operators their construction does not provide us with two independent structures. We conclude in Section 7.

This version of the paper is a revised and extended version of a published LPNMR'13 paper (Schwind and Inoue 2013). The main extensions include a comparison of our main characterization result with the one proposed in Delgrande *et al.* (2013a), some complexity results, characterization results for DLP and NLP revision operators and the proofs of propositions given in the online appendix of the paper.

2 Belief revision in propositional logic

2.1 Formal preliminaries

We consider a propositional language $\mathscr{L}$ defined from a finite set of propositional variables (also called *atoms*) $\mathscr{A}$ and the usual connectives. $\bot$ (resp. $\top$) is the Boolean constant always false (resp. true). A (classical) *interpretation* over $\mathscr{A}$ is a total function from $\mathscr{A}$ to $\{0,1\}$. To avoid heavy expressions, an interpretation I is also viewed as the subset of atoms from $\mathscr{A}$ that are true in I. For instance, if $\mathscr{A} = \{p,q\}$, then the interpretation over $\mathscr{A}$ such that $I(p) = 1$ and $I(q) = 0$ is also represented as the set $\{p\}$. For the sake of simplicity, set-notations will be dropped within interpretations (except for the case where the interpretation is the empty set), e.g., the interpretation $\{p,q\}$ will be simply denoted pq. The set of all interpretations is denoted Ω. An interpretation I is a *model* of a formula $\phi \in \mathscr{L}$, denoted $I \models \phi$, if it makes it true in the usual truth functional way. A *consistent* formula is a formula that admits a model. The set $mod(\phi)$ denotes the set of models of the formula ϕ, i.e., $mod(\phi) = \{I \in \Omega \mid I \models \phi\}$. Two formulae ϕ, ψ are said to be *equivalent*, denoted by $\phi \equiv \psi$ if and only if $mod(\phi) = mod(\psi)$.

2.2 Propositional revision operators

We now introduce some background on propositional belief revision. We start by introducing a revision operator as a simple function, that considers two formulae (the original formula and the new one) and that returns the revised formula:

Definition 1 (*Propositional revision operator, equivalence between operators*)
A (propositional) revision operator $\circ$ is a mapping associating two formulae ϕ, ψ with a new formula, denoted $\phi \circ \psi$. Two revision operators $\circ, \circ'$ are said to be *equivalent* (denoted $\circ \equiv \circ'$) when for all formulae ϕ, ψ, $\phi \circ \psi \equiv \phi \circ' \psi$.

The AGM framework (Alchourrón *et al.* 1985) describes the standard principles for belief revision (e.g., consistency preservation and minimality of change), which capture changes occurring in a static domain. Katsuno and Mendelzon (1991) equivalently rephrased the AGM postulates as follows:

Definition 2 (*KM revision operator*)
A KM revision operator $\circ$ is a propositional revision operator that satisfies the following postulates, for all formulae $\phi, \phi_1, \phi_2, \psi, \psi_1, \psi_2$:

(R1) $\phi \circ \psi \models \psi$.
(R2) If $\phi \wedge \psi$ is consistent, then $\phi \circ \psi \equiv \phi \wedge \psi$.
(R3) If ψ is consistent, then $\phi \circ \psi$ is consistent.
(R4) If $\phi_1 \equiv \phi_2$ and $\psi_1 \equiv \psi_2$, then $\phi_1 \circ \psi_1 \equiv \phi_2 \circ \psi_2$.

(R5) $(\phi \circ \psi_1) \wedge \psi_2 \models \phi \circ (\psi_1 \wedge \psi_2)$.
(R6) If $(\phi \circ \psi_1) \wedge \psi_2$ is consistent, then $\phi \circ (\psi_1 \wedge \psi_2) \models (\phi \circ \psi_1) \wedge \psi_2$.

These so-called *KM postulates* capture the desired behavior of a revision operator, e.g., in terms of consistency preservation and minimality of change. We now draw the reader's attention to the following important detail. The KM postulates also tell us that the outcome of a revision operator relies on an arbitrary syntactic distinction: one can see that a revision operator $\circ$ is a KM revision operator (i.e., it satisfies postulates (R1–R6)) if and only if any revision operator equivalent to $\circ$ is also a KM revision operator. In this paper, since we are only interested in whether an operator satisfies a set of rationality postulates or not, only the semantic contents of the revised base play a role, that is, relevance is considered only within the *models* of a revised base rather than on its explicit representation. This is why from now on, abusing terms we identify a revision operator modulo equivalence, that is, we actually refer to *any* revision operator equivalent to it. It becomes then harmless to define the resulting revised base in a modelwise fashion, as a set of models implicitly interpreted disjunctively. As a consequence, given two propositional revision operators $\circ, \circ'$, one can switch between the notations $\circ \equiv \circ'$ and $\circ = \circ'$ since there is no longer danger of confusion.

KM revision operators can be represented in terms of total preorders over interpretations. Indeed, each KM revision operator is associated with some faithful assignment (Katsuno and Mendelzon 1991). For each pre-order $\leqslant$, $\simeq$ denotes the corresponding indifference relation, and $<$ denotes the corresponding strict ordering; given a binary relation $\leqslant$ over a set E and any set $F \subseteq E$, the set $\min(F, \leqslant)$ denotes the subset of "minimal" elements from F w.r.t. $\leqslant$, i.e., $\min(F, \leqslant) = \{a \in F \mid \forall b \in F, b \leqslant a \Rightarrow a \leqslant b\}$.

Definition 3 (*Faithful assignment*)
A *faithful assignment* is a mapping which associates with every formula ϕ a pre-order $\leqslant_\phi$ over interpretations such that for all interpretations I, J, and all formulae ϕ, ϕ_1, ϕ_2, the following conditions hold:

(a) If $I \models \phi$ and $J \models \phi$, then $I \simeq_\phi J$.
(b) If $I \models \phi$ and $J \not\models \phi$, then $I <_\phi J$.
(c) If $\phi_1 \equiv \phi_2$, then $\leqslant_{\phi_1} = \leqslant_{\phi_2}$.

Theorem 1 (*Katsuno and Mendelzon 1992*)
A revision operator $\circ$ is a KM revision operator if and only if there exists a faithful assignment associating every formula ϕ with a total preorder $\leqslant_\phi$ such that for all formulae ϕ, ψ, $mod(\phi \circ \psi) = \min(mod(\psi), \leqslant_\phi)$.

Example 1
Consider the propositional language defined from the set of atoms $\mathscr{A} = \{p, q\}$. Let $\phi = p \Leftrightarrow \neg q$. Consider the total preorder $\leqslant_\phi$ defined as $p \simeq_\phi q <_\phi pq <_\phi \emptyset$. It can be easily checked that the conditions of a faithful assignment are satisfied by $\leqslant_\phi$. Then denote by $\circ$ the corresponding KM revision operator. Now, let $\psi_1 = \neg p \wedge q$ and $\psi_2 = p \Leftrightarrow q$. Figure 1 illustrates the total preorder $\leqslant_\phi$ and graphically identifies the models of ψ_1 and ψ_2. We get following:

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

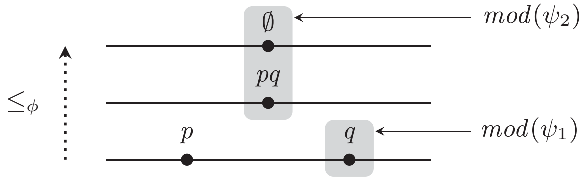

Fig. 1. The total preorder $\leqslant_\phi$ over interpretations associated with some faithful assignment.

- $mod(\phi \circ \psi_1) = \min(mod(\psi_1), \leqslant_\phi) = mod(\psi_1)$. Hence, $\phi \circ \psi_1 \equiv \psi_1$;
- $mod(\phi \circ \psi_2) = \min(mod(\psi_2), \leqslant_\phi) = \{pq\}$. Hence, $\phi \circ \psi_2 \equiv p \wedge q$.

In fact, an implicit consequence of Theorem 1 is that every KM revision operator is represented by a unique faithful assignment, and conversely, every faithful assignment represents a unique KM revision operator (modulo equivalence):

Proposition 1
There is a one-to-one correspondence between the KM revision operators and the set of all faithful assignments.

KM revision operators include the class of distance-based revision operators (see, for instance, (Dalal 1988)), i.e., those operators characterized by a distance between interpretations:

Definition 4 (*Distance-based revision operators*)
Let d be a distance between interpretations[1], extended to a distance between every interpretation I and every formula ϕ by

$$d(I,\phi) = \begin{cases} min\{d(I,J) \mid J \models \phi\} & \text{if } \phi \text{ is consistent,} \\ 0 \text{ otherwise.} \end{cases}$$

The revision operator based on the distance d is the operator $\circ^d$ satisfying for all formulae ϕ, ψ, $mod(\phi \circ^d \psi) = min(mod(\psi), \leqslant^d_\phi)$ where the preorder $\leqslant^d_\phi$ induced by ϕ is defined for all interpretations I, J by $I \leqslant^d_\phi J$ if and only if $d(I,\phi) \leqslant d(J,\phi)$.

The following result is a direct consequence of Theorem 1:

Corollary 1
Every distance-based revision operator is a KM revision operator, i.e., it satisfies the postulates (R1–R6).

The result of revising old beliefs (a propositional formula ϕ) by new beliefs (a propositional formula ψ) is any propositional formula whose models are models of ψ having a distance to a model of ϕ which is minimal among all models of ψ.

It is clear from Definition 4 that a distance fully characterizes the induced revision operator, that is, different choices for the distance induce different revision operators. Usual distances are d_D, the drastic distance ($d_D(I,J) = 1$ if and only if $I \neq J$), and d_H the Hamming distance ($d_H(I,J) = n$ if I and J differ on n variables). One can

[1] Actually, a pseudo-distance is enough, i.e., triangular inequality is not mandatory.

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

remark that when the drastic distance d_D is used, the induced faithful assignment associates with every formula ϕ a two-level preorder $\leqslant_\phi$; indeed, it can be easily verified that the revision operator based on the drastic distance d_D is equivalent to the so-called *drastic revision operator*, which is defined syntactically as follows:

Definition 5 (*Drastic revision operator*)
The *drastic revision operator*, denoted $\circ_D$, is the revision operator defined for all formulae ϕ, ψ as

$$\phi \circ_D \psi = \begin{cases} \phi \wedge \psi & \text{if } \phi \wedge \psi \text{ is consistent,} \\ \psi & \text{otherwise.} \end{cases}$$

This operator was first introduced in Alchourrón *et al.* (1985) under the name of *full meet revision function*. Though "fully rational" in the sense that it satisfies all the KM rationality postulates (i.e., all AGM postulates in Alchourrón *et al.* (1985)), it is often considered as unreasonable because it throws away all the old beliefs if the new formula is inconsistent with them.

Likewise, the revision operator based on Hamming distance d_H is equivalent to the well-known Dalal revision operator (Dalal 1988). In fact, in Dalal (1988) the Dalal revision is also defined in a modelwise fashion, i.e., there is no syntactic definition of it (as opposite to the drastic revision operator, cf. Definition 5):

Definition 6 (*Dalal revision operator*)
A *Dalal revision operator*, denoted $\circ_{Dal}$, is any revision operator based on the Hamming distance.

From now on, the revision operator based on the Hamming distance (i.e., the revision operator $\circ^{d_H}$) will simply be referred as the Dalal revision operator, and thus will be denoted $\circ_{Dal}$.

Example 2
Let $\mathscr{A} = \{p, q, r\}$, $\phi = p \wedge q \wedge \neg r$, and $\psi = r$. We have

- $\phi \circ_D \psi = r$.
- $\phi \circ_{Dal} \psi \equiv p \wedge q \wedge r$.

It is clear from Example 2 that the Dalal revision operator has a more parsimonious behavior than the drastic revision operator, because it integrates the new information while keeping as much previous beliefs as possible.

Before concluding this section, let us remark that distance-based revision operators as defined above do not fully characterize KM revision operators: this comes from the fact that given two formulae ϕ, ϕ' such that $\phi \not\equiv \phi'$, one can associate within the same faithful assignment two preorders $\leqslant_\phi, \leqslant_{\phi'}$ in an independent way; given that observation, one can easily build $\leqslant_\phi, \leqslant_{\phi'}$ using two different distances, whereas Definition 4 requires that the *same* distance is used to define the total preorder $\leqslant_\phi$ associated with *any* formula. However, as far as we know there does not exist in the literature any "fully rational" (with respect to postulates (R1–R6)) revision operator of interest that is not distance-based.

3 Belief revision in logic programming

3.1 Preliminaries on logic programming

We define the syntax and semantics of GPLs. We use the same notations as in Delgrande *et al.* (2008). A GLP is a finite set of rules of the form

$$a_1;\ldots;a_k;\sim b_1;\ldots;\sim b_l \leftarrow c_1,\ldots,c_m,\sim d_1,\ldots,\sim d_n,$$

where $k,l,m,n \geqslant 0$.

Each a_i, b_i, c_i, d_i is either one of the constant symbols $\bot$, $\top$, or an atom from $\mathscr{A}$; $\sim$ is the negation by failure; ";" is the disjunctive connective, "," is the conjunctive connective of atoms. The right-hand and left-hand sides of r are respectively called the head and body of r. For each rule r, we define $H(r)^+ = \{a_1,\ldots,a_k\}$, $H(r)^- = \{b_1,\ldots,b_l\}$, $B(r)^+ = \{c_1,\ldots,c_m\}$, and $B(r)^- = \{d_1,\ldots,d_n\}$. For the sake of simplicity, a rule r is also expressed as follows:

$$H(r)^+;\sim H(r)^- \leftarrow B(r)^+,\sim B(r)^-\cdot$$

A LP is interpreted through its preferred models based on the answer set semantics. A *(classical) model* X of a GLP P (written $X \models P$) is an interpretation from Ω that satisfies all rules from P according to the classical definition of truth in propositional logic. $mod(P)$ will denote the set of all models of a GLP P. An *answer set* X of a GLP P is a minimal (w.r.t. set inclusion) set of atoms from $\mathscr{A}$ that is a model of the program P^X, where P^X is called the *reduct* of P relative to X and is defined as $P^X = \{H(r)^+ \leftarrow B(r)^+ \mid r \in P, H(r)^- \subseteq X, B(r)^- \cap X = \emptyset\}$. The classical notion of equivalence between programs corresponds to the correspondence of their answer sets. Recall that we denote an interpretation by dropping set-notations except for the case of the interpretation corresponding to the empty set; for instance, the set of interpretations $\{\emptyset,\{p\},\{pq\}\}$ will be simply denoted $\{\emptyset, p, pq\}$.

Example 3

Consider the LP $P = \left\{ \begin{smallmatrix} p \leftarrow \sim q, \\ \bot \leftarrow p,q \end{smallmatrix} \right\}$. To determine $AS(P)$, the set of answer sets of P, we need to check for each interpretation X whether X is a minimal (w.r.t. set inclusion) model of P^X, the reduct of P relative to X:

- $P^{\emptyset} = \left\{ \begin{smallmatrix} p \leftarrow \top, \\ \bot \leftarrow p,q \end{smallmatrix} \right\}$, and $mod(P^{\emptyset}) = \{p\}$. Since $\emptyset$ is not a model of $P^{\emptyset}$, we get that $\emptyset \notin AS(P)$.
- $P^p = P^{\emptyset}$, so $mod(P^p) = \{p\}$. Since p is a minimal (w.r.t. set inclusion) model of P^p, we get that $p \in AS(P)$.
- $P^q = \{\bot \leftarrow p,q\}$, so $mod(P^q) = \{\emptyset, p, q\}$. Hence, q is a model of P^q but is not minimal w.r.t. set inclusion, since $\emptyset \in mod(P^q)$. Thus, $q \notin AS(P)$.
- Lastly, $P^{pq} = P^q$, so $mod(P^{pq}) = \{\emptyset, p, q\}$. Hence, pq is a not a model of P^{pq}, so we get that $pq \notin AS(P)$.

Therefore, $AS(P) = \{p\}$.

SE interpretations are semantic structures characterizing *strong equivalence* between LPs (Turner 2003), they provide a monotonic semantic foundation of LPs under answer set semantics. An SE interpretation over $\mathscr{A}$ is a pair (X,Y) of interpretations over $\mathscr{A}$ such that $X \subseteq Y$. An *SE model* (X, Y) of a LP P is an SE interpretation over $\mathscr{A}$ that satisfies $Y \models P$ and $X \models P^Y$, where P^Y is the reduct of P relative to Y. The set SE denotes the set of all SE interpretations over $\mathscr{A}$; given a LP P, the set $SE(P)$ denotes the set of SE models of P.

Example 4
Consider again the LP P defined in Example 3. We have $mod(P) = \{p, q\}$. Hence

$$
\begin{aligned}
SE(P) \quad &= \{(X, p) \in SE \mid X \in mod(P^p)\} \cup \{(X, q) \in SE \mid X \in mod(P^q)\} \\
&= \{(X, p) \in SE \mid X \in \{p\}\} \cup \{(X, q) \in SE \mid X \in \{\emptyset, p, q\}\} \\
&= \{(p, p), (\emptyset, q), (q, q)\}.
\end{aligned}
$$

Through their SE models, LPs are semantically described in a stronger way than through their answer sets, as shown in the following example.

Example 5
Let $P_1 = \{p \leftarrow\sim q\}$ and $P_2 = \left\{ \begin{smallmatrix} p \leftarrow\sim q, \\ p; q \leftarrow \top \end{smallmatrix} \right\}$, and consider again the LP P defined in Example 3. Then, we get that

$$AS(P) = AS(P_1) = AS(P_2) = \{p\},$$

that is, P, P_1, and P_2 admit the same answer sets. However, their SE models differ:

$$
\begin{aligned}
&SE(P) = \{(p, p), (\emptyset, q), (q, q)\} && \text{(cf. Example 3)}, \\
&SE(P_1) = \{(p, p), (\emptyset, q), (q, q), (\emptyset, pq), (p, pq), (q, pq), (pq, pq)\}, \\
&SE(P_2) = \{(p, p), (p, pq), (q, pq), (pq, pq)\}.
\end{aligned}
$$

A program P is *consistent* if $SE(P) \neq \emptyset$. Two programs P and Q are said to be *strongly equivalent*, denoted $P \equiv_s Q$, whenever $SE(P) = SE(Q)$. We also write $P \subseteq_s Q$ if $SE(P) \subseteq SE(Q)$. Two programs are equivalent if they are strongly equivalent, but the other direction does not hold in general (cf. Example 5). Note that Y is an answer set of P if and only if $(Y, Y) \in SE(P)$ and no $(X, Y) \in SE(P)$ with $X \subsetneq Y$ exists. We also have $(Y, Y) \in SE(P)$ if and only if $Y \in mod(P)$. A set of SE interpretations S is *well-defined* if for every interpretation X, Y with $X \subseteq Y$, if $(X, Y) \in S$, then $(Y, Y) \in S$. Every GLP has a well-defined set of SE models. Moreover, from every well-defined set S of SE models, one can build a GLP P such that $SE(P) = S$ (Eiter *et al.* 2005; Cabalar and Ferraris 2007).

We close this section by introducing two further notations. For every GLP P, α_P^2 is any propositional formula satisfying $mod(\alpha_P^2) = mod(P)$, and α_P^1 is any propositional formula satisfying $mod(\alpha_P^1) = \{X \in \Omega \mid (X, Y) \in SE(P)\}$.

3.2 Logic program revision operators

We now consider belief revision in the context of LPs. Given two programs P, Q the goal is to define a program $P \star Q$ which is the revision of P by Q. Delgrande *et al.* (2008, 2013b) proposed an adaptation of the KM postulates (cf. Definition 2) in the context of logic programming; this can be done using the monotonic characterization of LPs through their SE models. First, they considered the operation of *expansion* of two LPs:

Definition 7 (*Expansion operator (Delgrande et al. 2008)*)
Given two programs P, Q, the *expansion* of P by Q, denoted $P + Q$ is any program R such that $SE(R) = SE(P) \cap SE(Q)$.

Though the expansion of LPs trivializes the result whenever the two input LPs admit no common SE models, this operation is of interest in its own right. For instance, it can be observed that the intersection of two well-defined sets of SE interpretations leads to a well-defined set of SE interpretations, and thus the expansion of two GLPs is always defined as a GLP.

Example 6
Consider again the program P from Example 3, and recall that $SE(P) = \{(p, p), (\emptyset, q), (q, q)\}$. Let Q be the GLP $Q = \{q \leftarrow \top\}$, we have $SE(Q) = \{(q, q), (q, pq), (pq, pq)\}$. Furthermore, the GLP $R = \left\{ \begin{smallmatrix} q \leftarrow \top, \\ \bot \leftarrow p \end{smallmatrix} \right\}$ is such that $SE(R) = \{(q, q)\} = SE(P) \cap SE(Q)$. Therefore

$$P + Q = \left\{ \begin{array}{l} p \leftarrow \sim q, \\ \bot \leftarrow p, q \end{array} \right\} + \{q \leftarrow \top\} \equiv_s \left\{ \begin{array}{l} q \leftarrow \top, \\ \bot \leftarrow p \end{array} \right\}.$$

We refer the reader to Delgrande *et al.* (2013b), Section 3.1 for further examples of the use of the expansion operator.

Expansion of programs corresponds to the model-theoretical definition of expansion expressed through the KM postulates R2, R5, and R6. Delgrande *et al.* rephrased the full set of KM postulates (R1–R6) in the context of GLPs. Beforehand, we define a LP revision operator as a simple function, that considers two GLPs (the original one and the new one) and returns a revised GLP:

Definition 8 (*LP revision operator, equivalence between LP revision operators*)
A *LP revision operator* $\star$ is a mapping associating two GLPs P, Q with a new GLP, denoted $P \star Q$. Two LP revision operators $\star, \star'$ are said to be *equivalent* (denoted $\star \equiv \star'$) when for all GLPs P, Q, $P \star Q \equiv_s P \star' Q$.

Definition 9 (*GLP revision operator (Delgrande et al. 2008)*)
A *GLP revision operator* $\star$ is an LP revision operator that satisfies the following postulates, for all GLPs $P, P_1, P_2, Q, Q_1, Q_2, R$:

(RA1) $P \star Q \subseteq_s Q$.
(RA2) If $P + Q$ is consistent, then $P \star Q \equiv_s P + Q$.
(RA3) If Q is consistent, then $P \star Q$ is consistent.
(RA4) If $P_1 \equiv_s P_2$ and $Q_1 \equiv_s Q_2$, then $P_1 \star Q_1 \equiv_s P_2 \star Q_2$.
(RA5) $(P \star Q) + R \subseteq_s P \star (Q + R)$.
(RA6) If $(P \star Q) + R$ is consistent, then $P \star (Q + R) \subseteq_s (P \star Q) + R$.

As to the case of (propositional) KM revision operators, an LP revision operator $\star$ is a GLP revision operator if and only if any LP revision operator equivalent to $\star$ is also a GLP revision operator. This is why in the rest of the paper, as we identify a propositional revision operator modulo equivalence, we also identify an LP revision operator modulo equivalence. This allows us to define a revised program in a modelwise fashion, i.e., as its set of SE models, and given two LP revision operators $\star, \star'$, the notations $\star \equiv \star'$ and $\star = \star'$ are confounded with no harm.

Delgrande *et al.* (2008) proposed a revision operator inspired from Satoh's propositional revision operator (Satoh 1988). This operator, based on the set containment of SE interpretations, satisfies postulates (RA1–RA5). Though it seems to have a good behavior on some instances, this operator does not satisfy (RA6), so that it does not fully respect the principle of minimality of change (see Katsuno and Mendelzon (1989), Section 3.1 for details on this postulate). However, the whole set of postulates is consistent, as they later introduce the so-called *cardinality-based revision operator* (Delgrande *et al.* 2013b) that reduces to the Dalal revision scheme over propositional models and that satisfies all the postulates (RA1–RA6). The following definition is a concise, equivalent reformulation of the original one introduced in Delgrande *et al.* (2013b), Definition 3.10:

Definition 10 (*Cardinality-based revision operator*)
Given a GLP P and an interpretation Y, let $form_Y$ be any propositional formula satisfying $mod(form_Y) = \{Y\}$, let $\alpha_{(P,Y)}$ be any propositional formula satisfying $mod(\alpha_{(P,Y)}) = \{X \in \Omega \mid (X, Y') \in SE(P), Y' \models form_Y \circ_{Dal} \alpha_P^2\}$, and let α_Y be any propositional formula satisfying $mod(\alpha_Y) = \{X \in \Omega \mid X \subseteq Y\}$. The cardinality-based revision operator, denoted $\star_c$, is defined for all GLPs P, Q by any program $P \star_c Q$ satisfying

$$SE(P \star_c Q) = \{(X, Y) \in SE(Q) \mid Y \models \alpha_P^2 \circ_{Dal} \alpha_Q^2 \text{ and if } X \subsetneq Y \text{ then } X \models \alpha_{(P,Y)} \circ_{Dal} \alpha_Y\}.$$

Theorem 2 (*Delgrande et al. 2013b*)
$\star_c$ is a GLP revision operator.

In addition, we introduce below a simple, syntactically defined LP revision operator which also satisfies the whole set of postulates (RA1–RA6):

Definition 11 (*Drastic LP revision operator*)
The *drastic LP revision operator* $\star_D$ is defined for all GLPs P, Q as

$$P \star_D Q = \begin{cases} P + Q & \text{if } P + Q \text{ is consistent,} \\ Q & \text{otherwise.} \end{cases}$$

Proposition 2
$\star_D$ is a GLP revision operator.

Note that the drastic LP revision operator is the counterpart of the propositional drastic revision operator (cf. Definition 5) for LPs: the old program is thrown away if the new program is inconsistent with it. The cardinality-based revision operator has a more parsimonious behavior. However, Theorem 2 and Proposition 2 show that these operators are both fully satisfactory in terms of revision principles; this raises the problem on how to discard some rational operators from others. Moreover, it is not clear whether there even exist other GLP revision operators than the cardinality-based and the drastic LP revision operators. In the next section, we fill the gap and we give a constructive, full characterization of the class of GLP revision operators, that provides us a clear and complete picture of it.

4 Characterization of GLP revision operators

4.1 Characterization result

We now provide the main result of our paper, i.e., a characterization theorem for GLP revision operators. That is, we show that each GLP revision operator (i.e., each LP revision operator satisfying the postulates (RA1–RA6)) can be characterized in terms of preorders over the set of all classical interpretations, with some further conditions specific to SE interpretations.

Definition 12 (*LP faithful assignment*)
An *LP faithful assignment* is a mapping which associates with every GLP P a total preorder $\leqslant_P$ over interpretations such that for all GLPs P, Q and all interpretations Y, Y', the following conditions hold:

(1) If $Y \models P$ and $Y' \models P$, then $Y \simeq_P Y'$.
(2) If $Y \models P$ and $Y' \not\models P$, then $Y <_P Y'$.
(3) If $mod(P) = mod(Q)$, then $\leqslant_P = \leqslant_Q$.

Please note the similarities between an LP faithful assignment and a faithful assignment (cf. Definition 3). That is as follows:

Remark 1
Let Φ_1 be an assignment that associates with every GLP P a total preorder $\leqslant_P$ over interpretations, and Φ_2 be and assignment that associates with every formula ϕ a total preorder $\leqslant_\phi$ over interpretations. If for every GLP P, we have $\Phi_1(P) = \Phi_2(\alpha_P^2)$, then Φ_1 is an LP faithful assignment if and only if Φ_2 is a faithful assignment.

Definition 13 (*Well-defined assignment*)
A *well-defined assignment* is a mapping which associates with every GLP P and every interpretation Y a set of interpretations, denoted by $P(Y)$, such that for all GLPs P, Q and all interpretations X, Y, the following conditions hold:

(a) $Y \in P(Y)$.
(b) If $X \in P(Y)$, then $X \subseteq Y$.
(c) If $(X, Y) \in SE(P)$, then $X \in P(Y)$.
(d) If $(X, Y) \notin SE(P)$ and $Y \models P$, then $X \notin P(Y)$.
(e) If $P \equiv_s Q$, then $P(Y) = Q(Y)$.

Definition 14 (*GLP parted assignment*)
A *GLP parted assignment* is a pair (Φ, Ψ), where Φ is an LP faithful assignment and Ψ is a well-defined assignment.

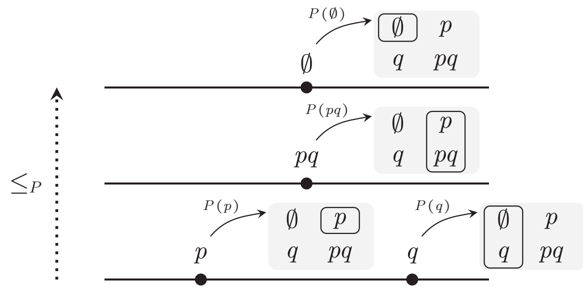

Fig. 2. The total preorder $\leqslant_P$ over SE interpretations, and the sets $P(Y)$ enclosed in boxes for all $Y \in \Omega$, associated with some GLP parted assignment.

We are ready to bring to light our main result:

Proposition 3
An LP operator $\star$ is a GLP revision operator if and only if there exists a GLP parted assignment (Φ, Ψ), where Φ associates with every GLP P a total preorder $\leqslant_P$, Ψ associates with every GLP P and every interpretation Y a set of interpretations $P(Y)$, and such that for all GLPs P, Q,

$$SE(P \star Q) = \{(X, Y) \mid (X, Y) \in SE(Q), Y \in \min(mod(Q), \leqslant_P), X \in P(Y)\}.$$

Note that there is no relationship between the LP faithful assignment Φ and the well-defined assignment Ψ forming a GLP parted assignment, that is, each one of these two mappings can be defined in a completely independent way.

Example 7
Let us consider again the GLP $P = \left\{ \begin{smallmatrix} p \leftarrow \sim q, \\ \bot \leftarrow p, q \end{smallmatrix} \right\}$ from Example 3, and recall that $SE(P) = \{(p, p), (\emptyset, q), (q, q)\}$. Note that the (classical) models of P (i.e., $mod(P) = \{p, q\}$) correspond to the models of the propositional formula ϕ given in Example 1 (i.e., $mod(\phi) = \{p, q\}$). Hence, due to Remark 1 the total preorder $\leqslant_P = \leqslant_\phi$, i.e., defined as $p \simeq_P q <_P pq <_P \emptyset$ satisfies the conditions of an LP faithful assignment (denoted Φ). Furthermore, let us consider the mapping Ψ associating with P and every interpretation Y the following sets of interpretations: $P(\emptyset) = \{\emptyset\}$, $P(p) = \{p\}$, $P(q) = \{\emptyset, q\}$, and $P(pq) = \{p, pq\}$. One can also check that Ψ satisfies the conditions (a–e) from Definition 13, so Ψ is a well-defined assignment. Hence, (Φ, Ψ) is a GLP parted assignment. Figure 2 gives a graphical representation of the total preorder $\leqslant_P$ and the sets $P(Y)$ for each $Y \in \Omega$. In the figure, all interpretations are ordered w.r.t. $\leqslant_P$ (similarly to Figure 1), and for each such interpretation Y, the set of circle interpretations next to Y corresponds to the set $P(Y)$.

Now, let us denote $\star$ the GLP revision operator corresponding to this GLP parted assignment, and let Q_1 and Q_2 be two GLPs defined as $Q_1 = \{q \leftarrow \sim p\}$ and $Q_2 = \left\{ \begin{smallmatrix} \bot \leftarrow p, \sim q, & \bot \leftarrow q, \sim p, \\ p; \sim p \leftarrow \top, & q; \sim q \leftarrow \top. \end{smallmatrix} \right\}$. We get that:

- $SE(Q_1) = \{(\emptyset, p), (p, p), (q, q), (\emptyset, pq), (p, pq), (q, pq), (pq, pq)\}$; then according to Proposition 3, we get that $SE(P \star Q_1) = \{(p, p), (q, q)\}$. Furthermore, the GLP

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

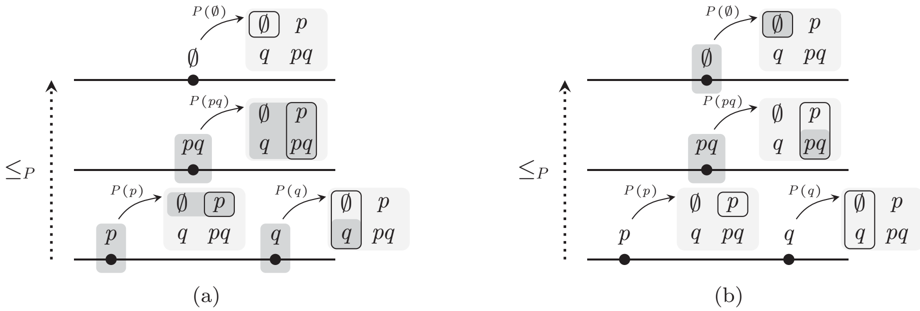

Fig. 3. The SE models of Q_1 and Q_2 highlighted within $\leqslant_P$ and sets $P(Y)$ for each interpretation Y. (a) The SE models of Q_1 are highlighted. We have $SE(Q_1) = \{(\emptyset, p), (p, p), (q, q), (\emptyset, pq), (p, pq), (q, pq), (pq, pq)\}$ and $SE(P \star Q_1) = \{(p, p), (q, q)\}$. (b) The SE models of Q_2 are highlighted. We have $SE(Q_2) = \{(\emptyset, \emptyset), (pq, pq)\}$ and $SE(P \star Q_2) = \{(pq, pq)\}$.

$R_1 = \left\{ \begin{smallmatrix} p \leftarrow\sim q, \\ q \leftarrow\sim p, \\ \bot \leftarrow p, q \end{smallmatrix} \right\}$ is such that $SE(R_1) = \{(p, p), (q, q)\} = SE(P \star Q_1)$. Therefore

$$P \star Q_1 = \left\{ \begin{array}{c} p \leftarrow\sim q, \\ \bot \leftarrow p, q \end{array} \right\} \star \{q \leftarrow\sim p\} \equiv_s \left\{ \begin{array}{c} p \leftarrow\sim q, \\ q \leftarrow\sim p, \\ \bot \leftarrow p, q \end{array} \right\}.$$

- $SE(Q_2) = \{(\emptyset, \emptyset), (pq, pq)\}$; then according to Proposition 3, we get that $SE(P \star Q_2) = \{(pq, pq)\}$. Furthermore, the GLP $R_2 = \left\{ \begin{smallmatrix} p \leftarrow \top, \\ q \leftarrow \top \end{smallmatrix} \right\}$ is such that $SE(R_2) = \{(pq, pq)\} = SE(P \star Q_2)$. Therefore

$$P \star Q_2 = \left\{ \begin{array}{c} p \leftarrow\sim q, \\ \bot \leftarrow p, q \end{array} \right\} \star \left\{ \begin{array}{cc} \bot \leftarrow p, \sim q, & \bot \leftarrow q, \sim p, \\ p; \sim p \leftarrow \top, & q; \sim q \leftarrow \top. \end{array} \right\} \equiv_s \left\{ \begin{array}{c} p \leftarrow \top, \\ q \leftarrow \top \end{array} \right\}.$$

The SE models of Q_1 and Q_2 are respectively illustrated in Figures 3(a) and (b).

Due to the similarities between an LP faithful assignment (cf. Definition 12) and a faithful assignment (cf. Definition 3), an interesting consequence from Theorem 1 and Proposition 3 is that every GLP revision operator can be viewed as an extension of a (propositional) KM revision operator:

Definition 15 (*Propositional-based LP revision operator*)
Let $\circ$ be a propositional revision operator and f be a mapping from Ω to 2^{Ω} such that for every interpretation Y, $Y \in f(Y)$ and if $X \in f(Y)$, then $X \subseteq Y$. The *propositional-based LP revision operator* w.r.t. $\circ$ and f, denoted $\star^{\circ,f}$, is defined for all GLPs P, Q by

$$SE(P \star^{\circ,f} Q) = \begin{cases} SE(P + Q) & \text{if } P + Q \text{ is consistent,} \\ \{(X, Y) \in SE(Q) \mid Y \models \alpha_P^2 \circ \alpha_Q^2, X \in f(Y)\} & \text{otherwise.} \end{cases}$$

$\star^{\circ,f}$ is said to be a *propositional-based GLP revision operator* if $\circ$ is a KM revision operator (i.e., satisfying postulates (R1–R6)).

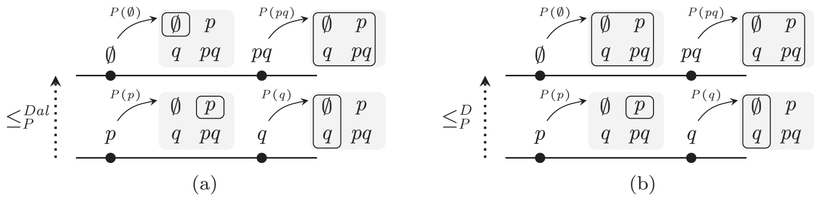

Fig. 4. The GLP parted assignments corresponding to the cardinality-based and drastic GLP revision operators, focusing on the GLP P. (a) The GLP parted assignment (for P) corresponding to the cardinality-based revision operator $\star_c$. (b) The GLP parted assignment (for P) corresponding to the drastic GLP revision operator $\star_D$.

Proposition 4
An LP revision operator is a GLP revision operator if and only if it is a propositional-based GLP revision operator.

In the previous section, we noticed that there is a one-to-one correspondence between the KM revision operators (modulo equivalence) and the set of all faithful assignments (cf. Proposition 1). Interestingly, we get a similar result in the case of GLP revision operators with respect to propositional-based GLP revision operators (cf. Corollary 2 below). Let us introduce an intermediate result:

Proposition 5
For all propositional-based GLP revision operators $\star^{\circ_1,f_1}, \star^{\circ_2,f_2}$, we have $\star^{\circ_1,f_1} = \star^{\circ_2,f_2}$ if and only if $\circ_1 = \circ_2$ and $f_1 = f_2$.

This proposition tells us that if $\circ_1 \neq \circ_2$ or $f_1 \neq f_2$, then for some pair of GLPs P, Q we will get $P \star^{\circ_1,f_1} Q \not\equiv P \star^{\circ_2,f_2} Q$, that is to say, different choices of parameters for a propositional-based LP revision operator lead to different propositional-based LP revision operators. As a direct consequence of Propositions 4 and 5, we get that:

Corollary 2
There is a one-to-one correspondence between the set of GLP revision operators and the set of propositional-based GLP revision operators.

Note that the cardinality-based revision operator $\star_c$ (cf. Definition 10) corresponds to the propositional-based GLP revision operator $\star^{\circ_{Dal},f_1}$, where $\circ_{Dal}$ is the Dalal revision operator (cf. Definition 6) and f_1 is defined for every interpretation Y as $f_1(Y) = \{X \in \Omega \mid X \subseteq Y \text{ and if } X \subsetneq Y, \text{ then } X \models \alpha_{(P,Y)} \circ_{Dal} \alpha_Y\}$, where α_Y is any propositional formula such that $mod(\alpha_Y) = \{X \in \Omega \mid X \subseteq Y\}$, $\alpha_{(P,Y)}$ is any propositional formula satisfying $mod(\alpha_{(P,Y)}) = \{X \in \Omega \mid (X, Y') \in SE(P), Y' \models form_Y \circ_{Dal} \alpha_P^2\}$, and $form_Y$ is any propositional formula satisfying $mod(form_Y) = \{Y\}$. In addition, the drastic GLP revision operator (cf. Definition 11) corresponds to the propositional-based GLP revision operator $\star^{\circ_D,f_2}$, where $\circ_D$ is the drastic revision operator (cf. Definition 5) and f_2 is defined for every interpretation Y as $f_2(Y) = 2^Y$. Figures 4(a) and (b) provide the graphical representation of these two operators in terms of parted assignments similarly to Figure 2, focusing on the GLP P from Example 3.

Remark that in the case where P and Q have no common SE models, then a (propositional-based) GLP revision operator $\star^{\circ,f}$ "rejects" as candidates for the

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

SE models of the revised program $P \star^{\circ,f} Q$ those SE interpretations whose second component is not a classical model of $\alpha_P^2 \circ \alpha_Q^2$; that is to say, as an upstream selection step the potential resulting SE models are chosen with respect to their second component by the underlying propositional revision operator $\circ$. Then, one can see from Definition 15 that the function f is used as a second filtering step that is made with respect to the first component of those preselected SE interpretations, and that this final selection becomes independent of the underlying input program P. Then it becomes questionable whether the postulates (RA1–RA6) sufficiently describe the rational behavior of LP revision operators. Indeed, we will show in the next section that this "freedom" on the definition of the function f raises some issues for some specific subclasses of fully rational LP revision operators.

4.2 Comparison with other existing works

As we already briefly mentioned in the introduction, Delgrande *et al.* (2013a) also recently proposed a constructive characterization of belief revision operators for LPs that satisfy the whole set of postulates (RA1–RA6). They considered various forms of LPs, i.e., generalized, disjunctive, normal, positive, and Horn, so we shall now compare our characterization with the one given in Delgrande *et al.* (2013a) for the case of GLPs:

Definition 16 (*GLP compliant faithful assignment (Delgrande et al. 2013a)*)
A *GLP compliant faithful assignment* is a mapping which associates every GLP P with a total preorder $\leqslant_P^*$ over SE interpretations such that for all GLPs P, Q, and all SE interpretations $(X, Y), (X', Y')$, the following conditions hold:

(1) If $(X, Y) \in SE(P)$ and $(X', Y') \in SE(P)$, then $(X, Y) \simeq_P^* (X', Y')$.
(2) If $(X, Y) \in SE(P)$ and $(X', Y') \notin SE(P)$, then $(X, Y) <_P^* (X', Y')$.
(3) If $P \equiv_s Q$, then $\leqslant_P = \leqslant_Q$.
(4) $(Y, Y) \leqslant_P^* (X, Y)$.

The following theorem is expressed as a combination of Theorems 4 and 5 from Delgrande *et al.* (2013a) applied to GLPs:

Theorem 3 (*Delgrande et al. 2013a*)
An LP revision operator $\star$ is a GLP revision operator (i.e., it satisfies postulates (RA1 - RA6)) if and only if there exists a GLP compliant faithful assignment associating every GLP P with a total preorder $\leqslant_P^*$ such that for all GLPs P, Q, $SE(P \star Q) = \min(SE(Q), \leqslant_P^*)$.[2]

Since both our GLP parted assignments and Delgrande *et al.*'s GLP compliant faithful assignments characterize the class of GLP revision operators, there must exist a relationship between the two structures. We denote by GLP_{part} the

[2] In Delgrande *et al.* (2013a), an additional postulate is considered in the characterization theorems, namely (Acyc). However, it is harmless to omit this postulate here since (Acyc) is a logical consequence of the postulates (RA1 - RA6) in the case of generalized logic programs (cf. (Delgrande *et al.* 2013a), Theorem 2).

set of all GLP parted assignments and GLP_{faith} the set of all GLP compliant faithful assignments. We now formally establish a correspondence between the two sets.

Definition 17

Let $\sigma_{part \to faith}$ be a binary relation on $\mathrm{GLP}_{part} \times \mathrm{GLP}_{faith}$ defined as follows. For every $(\Phi, \Psi) \in \mathrm{GLP}_{part}$ (where Φ associates every GLP P with a total preorder $\leqslant_P$, and Ψ associates every GLP P and every interpretation Y with a set of interpretations $P(Y)$), and for every $\Gamma \in \mathrm{GLP}_{faith}$ (where Γ associates every GLP P with a total preorder $\leqslant^*_P$), we have $((\Phi, \Psi), \Gamma) \in \sigma_{part \to faith}$ if and only if for every GLP P, for all interpretations X, Y, Y', $X \subseteq Y$, the following conditions are satisfied:

(i) $(Y, Y) \leqslant^*_P (Y', Y')$ if and only if $Y \leqslant_P Y'$, and
(ii) $(X, Y) \leqslant^*_P (Y, Y)$ if and only if $X \in P(Y)$.

We show now that a pair of assignments from $\mathrm{GLP}_{part} \times \mathrm{GLP}_{faith}$ satisfies the relation $\sigma_{part \to faith}$ if and only if represent both assignments represent the same GLP revision operator:

Proposition 6

For every $(\Phi, \Psi) \in \mathrm{GLP}_{part}$ and every $\Gamma \in \mathrm{GLP}_{faith}$, $((\Phi, \Psi), \Gamma) \in \sigma_{part \to faith}$ if and only if for all GLPs P, Q, $\min(SE(Q), \leqslant^*_P) = \{(X, Y) \mid (X, Y) \in SE(Q), Y \in \min(mod(Q), \leqslant_P), X \in P(Y)\}$.

Whereas our GLP parted assignments are formed of two structures which are independent from each other (an LP faithful assignment used to order the second components of SE interpretations, and a well-defined assignment selecting the first component of SE interpretations), Delgrande *et al.*'s GLP compliant faithful assignments consist of a single structure, i.e., a set of total preoders over SE interpretations. Though it may look simpler to represent a GLP revision operator through a single assignment, it turns out that the induced characterization (cf. Theorem 3) is not a one-to-one correspondence; more precisely, $\sigma_{part \to faith}$ is not a function and as a consequence, a given GLP revision operator can be represented by different GLP compliant faithful assignments. Roughly speaking, this is due to the fact that totality required by preorders $\leqslant^*_P$ is actually not needed. Many comparisons between pairs of SE interpretations within a total preorder $\leqslant^*_P$ are irrelevant to the GLP revision operator they correspond to. This is illustrated in the following example:

Example 8

Consider again the GLP P from Example 3 and the GLP parted assignment (Φ, Ψ) focusing on P depicted in Figure 2. Then Figure 5 depicts three total preorders $\leqslant^1_P$, $\leqslant^2_P$, and $\leqslant^3_P$ induced from three different GLP compliant faithful assignments Γ^1, Γ^2, and Γ^3 which both correspond to the GLP parted assignment (Φ, Ψ), i.e., $((\Phi, \Psi), \Gamma^1), ((\Phi, \Psi), \Gamma^2), ((\Phi, \Psi), \Gamma^3) \in \sigma_{part \to faith}$. It can be easily checked that for any GLP Q, $\min(SE(Q), \leqslant^1_P) = \min(SE(Q), \leqslant^2_P) = \min(SE(Q), \leqslant^3_P)$. The SE interpretations enclosed in dashed boxes correspond to those $(X, Y) \in \{(\emptyset, p), (\emptyset, pq), (q, pq)\}$

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

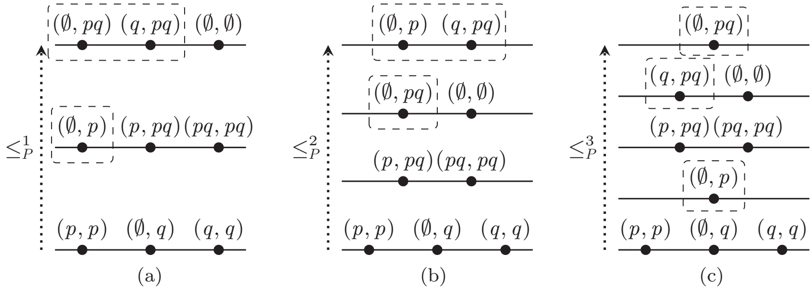

Fig. 5. Three total preorders corresponding to three different GLP compliant faithful assignments which correspond to the same GLP parted assignment. (a) The total preorder $\leqslant_P^1$ associated with Γ^1. (b) The total preorder $\leqslant_P^2$ associated with Γ^2. (c) The total preorder $\leqslant_P^3$ as-sociated with Γ^3.

whose comparison with other SE interpretations is irrelevant to the represented GLP revision operator, as far as one has $(Y, Y) <_P^i (X, Y)$ for $i \in \{1, 2, 3\}$.

In fact, one can see that as soon as the language contains at least two propositional variables, e.g., $\{p, q\} \subseteq \mathscr{A}$ with $p \neq q$, then the GLP P satisfying $(p, p), (q, q) \in SE(P)$, and $(\emptyset, p), (\emptyset, q) \notin SE(P)$ can be associated through a GLP compliant faithful assignment with at least three different total preorders; an arbitrary relative ordering between the SE interpretations $(\emptyset, p)$ and $(\emptyset, q)$ will have no effect on the corresponding GLP revision operator.

Removing the property of totality from preorders involved in a GLP compliant faithful could be an alternative towards establishing another one-to-one correspondence with GLP revision operators. However, our GLP parted assignments make clear the different roles played by the first and second components of SE interpretations in terms of GLP revision. On the one hand the second components are totally ordered, on the other hand the first components are arbitrarily selected as possible candidates for SE interpretations. This allows us to make precise the link with propositional faithful assignments and propositional revision operators, which would not be clear with a slight adjustment of GLP compliant faithful assignments. The next section shows how our propositional-based GLP revision operator facilitate the comprehension and analysis of GLP revision.

5 GLP revision operators embedded into Boolean lattices

For every propositional revision operator $\circ$, let $GLP(\circ)$ denote the set of all propositional-based LP revision operators w.r.t. $\circ$. One can remark that from Proposition 5, the set $\{GLP(\circ) \mid \circ \text{ is a KM revision operator}\}$ forms a partition of the class of all GLP revision operators. Let us now take a closer look to the set of GLP revision operators $GLP(\circ)$ when we are given any specific KM revision operator $\circ$:

Definition 18
Let $\circ$ be a propositional revision operator. We define the binary relation $\preceq_\circ$ over $GLP(\circ)$ as follows: for all propositional-based LP revision operators $\star^{\circ,f_1}, \star^{\circ,f_2}$, $\star^{\circ,f_1} \preceq_\circ \star^{\circ,f_2}$ if and only for every interpretation Y, we have $f_2(Y) \subseteq f_1(Y)$.

One can see that for each revision operator $\circ$, the set $(GLP(\circ), \preceq_\circ)$ forms a structure that is isomorphic to a Boolean lattice[3], and the careful reader will notice that $(GLP(\circ), \preceq_\circ)$ precisely corresponds to the product of the Boolean lattices $\{(\mathbb{B}_Y, \subseteq) \mid Y \in \Omega\}$, where $\mathbb{B}_Y = \{Z \cup \{Y\} \mid Z \in 2^{2^Y \setminus Y}\}$. The following result shows that this lattice structure can be used to analyse the relative semantic behavior of GLP revision operators from $(GLP(\circ), \preceq_\circ)$.

Proposition 7
Let $\circ$ be a KM revision operator. It holds that for all GLP revision operators $\star_1, \star_2 \in GLP(\circ)$, $\star_1 \preceq_\circ \star_2$ if and only if for all GLPs P, Q, we have $AS(P \star_1 Q) \subseteq AS(P \star_2 Q)$.

This result paves the way for the choice of a specific GLP revision operator depending on the desired "amount of information" provided by the revised GLP in terms of number of its answer sets. Precisely, any GLP revision operator $\star^{\circ,f}$ can be specified from an answer set point of view by the following roadmap. Since in the case where $P + Q$ is consistent, we always have $P \star^{\circ,f} Q = P + Q$, the intuition underlying this procedure only applies when the programs considered for the revision have no common SE model. First, one chooses a KM revision operator $\circ$ whose role is to filter the undesired answer sets of the resulting revised program: only the models Y of the formula resulting from the revision of P by Q in the propositional sense should be selected as "potential answer set candidates". Then, the function f plays a role in filtering those preselected candidates Y, so that f can be defined according to the following intuition: the more interpretations $X \subsetneq Y$ are included in $f(Y)$, the less likely the interpretation Y will actually be an answer set of the resulting revised program. More precisely, the presence of a given interpretation $X \subsetneq Y$ in $f(Y)$ is enough to discard Y as being an answer set of the resulting revised program when (X, Y) is an SE model of Q.

This brings in light that, depending on the "position" of the GLP revision operator $\star^{\circ,f}$ in the lattice $(GLP(\circ), \preceq_\circ)$, when revising P by Q one may expect divergent results for $AS(P \star^{\circ,f} Q)$. We illustrate this claim by considering two specific classes of GLP revision operators that correspond respectively to the suprema and infima of lattices $(GLP(\circ), \preceq_\circ)$ for all KM revision operators $\circ$. The first "extreme" operators are defined as follows:

Definition 19 (*Skeptical GLP revision operators*)
The *skeptical GLP revision operators*, denoted $\star_S^\circ$ are the propositional-based GLP revision operators $\star^{\circ,f}$ where f is defined for every interpretation Y by $f(Y) = 2^Y$.

[3] A Boolean lattice is a partially ordered set $(E, \leqslant_E)$ which is isomorphic to the set of subsets of some set F together with the usual set-inclusion operation, i.e., $(2^F, \subseteq)$.

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

Note that skeptical GLP revision operators include the drastic GLP revision operator $\star_D$ (cf. Definition 11), i.e., $\star_D = \star_S^{\circ_D}$ where $\circ_D$ is the (propositional) drastic revision operator. For each propositional revision operator $\circ$, we clearly have $\star_S^{\circ} = inf(GLP(\circ), \preceq_\circ)$. We provide now an axiomatic characterization of the skeptical GLP revision operators in order to get a clearer view of their general behavior:

Proposition 8
The skeptical GLP revision operators are the only GLP revision operators $\star$ such that for all GLPs P, Q, whenever $P + Q$ is inconsistent, we have $AS(P \star Q) \subseteq AS(Q)$.

Remark that the drastic GLP revision operator (cf. Definition 11), i.e., the skeptical GLP revision operator based on the propositional drastic revision operator $\star_S^{\circ_D}$, is a specific case from the result given in Proposition 8 where $AS(P \star_S^{\circ_D} Q) = AS(Q)$ whenever $P + Q$ is inconsistent.

We now introduce another class of GLP revision operators which correspond to the other "extreme cases" w.r.t lattices $(GLP(\circ), \preceq_\circ)$:

Definition 20 (*Brave GLP revision operators*)
The *brave GLP revision operators*, denoted $\star_B^{\circ}$ are the propositional-based GLP revision operators $\star^{\circ,f}$ where f is defined for every interpretation Y by $f(Y) = \{Y\}$.

We get now that for each propositional revision operator $\circ$, $\star_B^{\circ} = sup(GLP(\circ), \preceq_\circ)$. The brave operators are axiomatically characterized as follows:

Proposition 9
The brave GLP revision operators are the only GLP revision operators $\star^{\circ,f}$ such that for all GLPs P, Q, whenever $P + Q$ is inconsistent, we have $AS(P \star^{\circ,f} Q) = mod(\alpha_P^2 \circ \alpha_Q^2)$.

Let us remark as a specific case that the brave GLP revision operator based on the propositional drastic revision operator, i.e., the operator $\star_B^{\circ_D}$, satisfies $AS(P \star_B^{\circ_D} Q) = mod(Q)$ whenever $P + Q$ is inconsistent.

The following representative example illustrates how much the behavior of skeptical and brave GLP revision operators diverge:

Example 9
Consider $\circ_D$, i.e., the propositional drastic revision operator. Let $P = \left\{ \begin{array}{c} p \leftarrow \top, \\ q \leftarrow \top, \\ \bot \leftarrow r \end{array} \right\}$ and $Q = \{\bot \leftarrow p, q, \sim r\}$. We have $AS(P) = \{p, q\}$, $AS(Q) = \{\emptyset\}$, and

$$\begin{cases} AS(P \star_S^{\circ_D} Q) = \{\emptyset\}, \\ AS(P \star_B^{\circ_D} Q) = \{\emptyset, p, q, r, pr, qr, pqr\}. \end{cases}$$

Though they are rational LP revision operators w.r.t. the postulates (RA1–RA6), skeptical and brave operators have a rather trivial, thus undesirable behavior. Consider first the case of skeptical operators and assume that the proposition p is believed to be false, then learned to be true. That is, $\{\bot \leftarrow p\} \subseteq P$ and $Q = \{p \leftarrow \top\}$. Then one obtains that $AS(P \star_S^{\circ} Q) \subseteq AS(Q)$, that is, $AS(P \star_S^{\circ} Q) \subseteq \{p\}$, i.e., for any such program P, on learning that p is true the revision states that *only* p may be true, which holds independently from the choice of the KM revision operator $\circ$. On the other hand, brave operators only focus on classical models of LPs P, Q to compute $P \star_B^{\circ} Q$ (whenever $P + Q$ is inconsistent), thus they do not take into consideration the inherent, non-monotonic behavior of LPs. As a consequence, programs $P \star_B^{\circ} Q$ will often admit many answer sets that are actually irrelevant to the input programs P and Q.

Stated otherwise, skeptical and brave GLP revision operators are dual sides of a "drastic" behavior for the revision. These operators are representative examples that provide some "bounds" of the complete picture of GLP revision operators $GLP(\circ)$, for each KM revision operator $\circ$. Discarding such drastic behaviors may call for additional postulates in order to capture more parsimonious revision procedures in logic programming, as for instance the cardinality-based revision operator (cf. Definition 10) which is neither brave nor skeptical. Then it seems necessary to refine the existing properties that every rational revision operator should satisfy so that the answer sets of the revised program $P \star^{\circ,f} Q$ fall "between" these two extremes (i.e., between $AS(Q)$ and $mod(P \circ Q)$ in the sense of set inclusion).

Another benefit from our characterization result is that one can easily derive computational results by exploiting existing ones from propositional revision. We assume that the reader is familiar with the basic concepts of computational complexity, in particular with the classes **P**, **NP**, and co**NP** (see Papadimitriou (1994) for more details). Higher complexity classes are defined using oracles. In particular, P^{C} corresponds to the class of decision problems that are solved in polynomial time by deterministic Turing machines using an oracle for C in polynomial time. For instance, $\Theta_2^p = \mathbf{P}^{\mathbf{NP}[\mathcal{O}(log\ n)]}$ is the class of problems that can be solved in polynomial time by a deterministic Turing machine using a number of calls to an **NP** oracle bounded by a logarithmic function of the size of the input representation of the problem.

We focus here on the the *model-checking problem* (Liberatore and Schaerf 2001) for LP revision operators. In the propositional case, the model-checking problem consists in deciding whether a (propositional) interpretation is supported by a revised formula:

Problem 1 (MC($\circ$))

- **Input:** *A propositional revision operator $\circ$, two formulae ϕ, ψ, and an interpretation I,*
- **Question:** *Does $I \models \phi \circ \psi$ hold?*

The model-checking problem for the drastic revision operator (cf. Definition 5) is coNP-complete, while it is Θ_2^p-complete for the Dalal revision operator (cf. Definition 6):

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

Proposition 10
$\mathsf{MC}(\circ_D)$ is coNP-complete.

Theorem 4 (*Liberatore and Schaerf 2001*)
$\mathsf{MC}(\circ_{Dal})$ is Θ_2^{p}-complete.

Similarly, one can consider the model-checking problem for LP revision operators which consists in deciding whether an SE interpretation is an SE model of a revised program:

Problem 2 ($\mathsf{MC}_{SE}(\star)$)

- **Input:** *An LP revision operator* $\star$*, two GLPs* P, Q *and an SE interpretation* (X, Y)*,*
- **Question:** *Does* (X, Y) *belong to* $SE(P \star Q)$*?*

Remark that given an SE interpretation (X, Y) and a LP P, checking whether (X, Y) is an SE model of P is in **P**: computing the program P^Y, i.e., the reduct of P relative to Y, is performed in polynomial time; then it is enough to check whether $Y \models P$ and $X \models P^Y$ which is performed in polynomial time. Interestingly, when f is computed in polynomial time the model-checking problem for propositional-based LP revision operators $\star^{\circ,f}$ is not harder than the counterpart problem for the propositional revision operator $\circ$. Obviously enough, this applies for both skeptical and brave GLP revision operators, so Proposition 10 and Theorem 4 provide us with the following complexity results:

Corollary 3

- $\mathsf{MC}_{SE}(\star_S^{\circ_D})$ and $\mathsf{MC}_{SE}(\star_B^{\circ_D})$ are coNP-complete.
- $\mathsf{MC}_{SE}(\star_S^{\circ_{Dal}})$ and $\mathsf{MC}_{SE}(\star_B^{\circ_{Dal}})$ are Θ_2^{p}-complete.

6 The case of disjunctive and normal logic programs

In this section, we take a look at more restrictive forms of programs, i.e., the DLPs and the NLPs. A DLP is a GLP where rules are of the form

$$a_1; \dots ; a_k \leftarrow b_1, \dots, b_l, \sim c_1, \dots, \sim c_m,$$

where $k, l, m \geqslant 0$. A NLP is a DLP where $k = 1$.

Let us recall that every GLP has a *well-defined* set S of SE models, which requires that $(Y, Y) \in S$ for every $(X, Y) \in S$, and that conversely, for every well-defined set S of SE interpretations one can build a GLP P such that $SE(P) = S$. Since NLPs and DLPs are syntactically more restrictive than GLPs, these programs are characterized by sets of SE models satisfying stronger conditions. A set of SE interpretations S is said to be:

- *complete* if it is well-defined and for all interpretations X, Y, Z, if $Y \subseteq Z$ and $(X, Y), (Z, Z) \in S$, then also $(X, Z) \in S$;
- *closed under here-intersection* if it is complete and for all interpretations X, Y, Z, if $(X, Z), (Y, Z) \in S$, then also $(X \cap Y, Z) \in S$.

Each DLP (respectively, NLP) has a complete (respectively, closed under here-intersection) set of SE models. Conversely, if a set of SE interpretations S is complete (respectively, closed under here-intersection) then one can build a DLP (respectively, NLP) P such that $SE(P) = S$ (Eiter *et al.* 2005; Cabalar and Ferraris 2007). For instance, one can easily check that:

- the LP $P = \left\{ \begin{smallmatrix} p \leftarrow \sim q, \\ \bot \leftarrow p,q \end{smallmatrix} \right\}$ from Example 3 is a NLP and $SE(P)$ is well-defined, complete, and closed under here-intersection;
- the LP $P_2 = \left\{ \begin{smallmatrix} p \leftarrow \sim q, \\ p;q \leftarrow \top \end{smallmatrix} \right\}$ from Example 5 is a DLP and $SE(P_2)$ is well-defined, and complete, but not closed under here-intersection;
- the LP $Q_2 = \left\{ \begin{smallmatrix} \bot \leftarrow p, \sim q, & \bot \leftarrow q, \sim p, \\ p; \sim p \leftarrow \top, & q; \sim q \leftarrow \top. \end{smallmatrix} \right\}$ from Example 7 is a GLP and $SE(Q)$ is well-defined but not complete.

When revising a LP by another one, one expects the resulting revised program to be expressed in the same language as the input programs.

Definition 21 (*DLP/NLP revision operator*)
A *DLP revision operator* (respectively, a *NLP revision operator*) $\star$ is an LP revision operator associating two DLPs (respectively, two NLPs) P, Q with a new DLP (respectively, a new NLP) $P \star Q$, and which satisfies postulates (RA1–RA6).

We first remark that both sets of DLP revision operators and NLP revision operators are not empty. Indeed, one can observe that the intersection of two complete sets of SE interpretations is also complete, thus the expansion of two DLPs leads to a DLP. This also applies for NLPs. As a direct consequence, the drastic LP revision operator (cf. Definition 11) is both a DLP revision operator and a NLP revision operator. In fact, we have the more general result:

Proposition 11
The skeptical GLP revision operators are both DLP revision operators and NLP revision operators.

However, the above result does not apply for all GLP revision operators. That is to say, there exist some GLP revision operators which associate two NLPs with a GLP which is not a DLP. Hence, our sound and complete construction of GLP revision operators does not hold anymore for DLP and NLP revision operators. For instance, brave GLP revision operators are neither DLP revision operators nor NLP revision operators, as shown in the following example:

Example 10
Let $P = \left\{ \begin{smallmatrix} \bot \leftarrow \sim p, \sim q, \\ \bot \leftarrow q, \sim p, \\ \bot \leftarrow p,q \end{smallmatrix} \right\}$ and $Q = \{q \leftarrow \top\}$ be two NLPs. We have that $SE(P) = \{(\emptyset, p), (p, p)\}$ and $SE(Q) = \{(q, q), (q, pq), (pq, pq)\}$. Consider the brave GLP revision operator $\star_B^{\circ_D}$ based on the propositional drastic revision operator. Then one can verify that $SE(P \star_B^{\circ_D} Q) = \{(q, q), (pq, pq)\}$ is not a complete set of SE interpretations, thus $P \star_B^{\circ_D} Q$ cannot be represented as a DLP.

As a consequence, our characterization result from Proposition 3 does not hold anymore for DLP/NLP revision operators. Nevertheless, we provide below a representation of both DLP and NLP revision operators in terms of two structures where the first one is an LP faithful assignment adapted to DLPs/NLPs and the second one is a well-defined assignment "strengthened" by some further conditions.

Definition 22 (*DLP/NLP faithful assignment*)
A *DLP faithful assignment* (respectively, a *NLP faithful assignment*) is a mapping which associates every DLP (respectively, every NLP) with a total preorder over interpretations such that conditions (1–3) of an LP faithful assignment are satisfied.

Definition 23 (*Complete assignment*)
Let Φ be a DLP faithful assignment which associates every DLP P with a total preorder $\leqslant_P$. A *Φ-based complete assignment* is a mapping which associates with every DLP P and every interpretation Y a set of interpretations denoted by $P_\Phi(Y)$, such that conditions (a–e) of a well-defined assignment are satisfied as well as the following further condition, for all interpretations X, Y, Z:

(f) If $X \in P_\Phi(Y)$, $Y \simeq_P Z$, and $Y \subseteq Z$, then $X \in P_\Phi(Z)$.

A pair (Φ, Ψ_Φ), where Φ is a DLP faithful assignment and Ψ_Φ is a Φ-based complete assignment, is called a *DLP parted assignment*.

Definition 24 (*Normal assignment*)
Let Φ be a NLP faithful assignment. A *Φ-based normal assignment* is a mapping which associates with every NLP P and every interpretation Y a set of interpretations denoted by $P_\Phi(Y)$, such that conditions (a–f) of a complete assignment are satisfied as well as the following further condition, for all interpretations X, Y, Z:

(g) If $X, Y \in P_\Phi(Z)$, then $X \cap Y \in P_\Phi(Z)$.

A pair (Φ, Ψ_Φ), where Φ is a NLP faithful assignment and Ψ_Φ is a Φ-based normal assignment, is called a *NLP parted assignment*.

We are ready to provide our characterization results for DLP revision operators and NLP revision operators:

Proposition 12
An LP operator $\star$ is a DLP (resp. NLP) revision operator if and only if there exists a DLP (resp. NLP) parted assignment (Φ, Ψ_Φ), where Φ associates with every DLP (resp. NLP) P a total preorder $\leqslant_P$, Ψ_Φ is a Φ-based complete (resp. normal) assignment which associates with every DLP (resp. NLP) P and every interpretation Y a set of interpretations $P_\Phi(Y)$, and such that for all DLPs (resp. NLPs) P, Q,

$$SE(P \star Q) = \{(X, Y) \mid (X, Y) \in SE(Q), Y \in \min(mod(Q), \leqslant_P), X \in P_\Phi(Y)\}.$$

As to the case of our characterization of GLP revision operators, Proposition 12 provides us with sound and complete constructions of DLP and NLP revision operators in terms of total preorders over propositional interpretations and some further conditions specific to SE interpretations. Furthermore, because both constructions

https://doi.org/10.1017/S1471068415000101 Published online by Cambridge University Press

are similar to the one of GLP revision operators, without stating it formally one can straightforwardly establish a one-to-one correspondence between DLP/NLP revision operators and propositional-based LP revision operators (cf. Definition 15) satisfying some further conditions on the function f very similar to conditions (f) and (g). Indeed, one can see from Definition 23 and 24 that the two structures involved in DLP/NLP parted assignments are not independent anymore, since by condition (f) the Φ-based complete and normal assignments should both be aligned with the corresponding faithful assignment. As a consequence, these structures are more complex than those of GLP parted assignments and similar embeddings of DLP/NLP revision operators into Boolean lattices are no more applicable. A deeper investigation of the type of ruling structures for Φ-based complete and normal assignments is out of the scope of this paper, but constitutes an interesting direction to explore in a future work.

7 Conclusion

In this paper, we pursued some previous work on revision of LPs, where the adopted approach is based on a monotonic characterization of LPs using SE interpretations. We gave a particular attention to the revision of GLPs and characterized the class of rational GLP revision operators in terms of total orderings among classical interpretations with some further conditions specific to SE interpretations. The constructive characterization we provided facilitates the comprehension of the semantic properties of GLP revision operators by drawing a clear, complete picture of them. Interestingly, we showed that a GLP revision operator can be viewed as an extension of a rational propositional revision operator: each propositional revision operator corresponds to a specific subclass of GLP revision operators, and a GLP revision operator from a particular subclass can be specified independently of the propositional revision operator under consideration. Moreover, we showed that each one of these subclasses can be embedded into a Boolean lattice whose infimum and supremum, the so-called *skeptical* and *brave* GLP revision operators, have some relatively drastic behavior. In addition, we adjusted our representation structures and provided sound and complete constructions for two more specific classes of LPs, i.e., the disjunctive and NLPs.

Our results make easier the improvement of the current AGM framework in the context of logic programming. Indeed, though the subclasses of skeptical and brave revision operators are fully satisfactory w.r.t. the AGM revision principles, their behavior is shown to be rather trivial. This may call for additional postulates which would aim to capture more parsimonious, “balanced” classes of revision operators.

As to the case of *update* of LPs Slota and Leite (2014) argued that semantic rule updates based on SE models seem to be inappropriate. Indeed they showed that in presence of the irrelevance-of-syntax postulate (whose counterpart in the context of revision is (RA4)), semantic rule update operators based on SE models violate some reasonable properties for rule updates, i.e., *dynamic support* and *fact update* (see Slota and Leite (2014) for more details). The property of dynamic support can be expressed informally as follows: an rule update operator $\oplus$ satisfies dynamic support if every atom true in an answer set from any updated program $P \oplus Q$ should be supported by a rule in $P \cup Q$, i.e., it should have some "justification" in either the original program or the new one. The property of fact update requires some notion of atom inertia when updating a consistent set of facts (i.e., a set of rules of the type $p \leftarrow \top$ where p is an atom) by a consistent set of facts. Both of these properties require rule update operators to have a reasonable "syntactic" behavior, away from the purely semantic approach represented by the adapted AGM postulates. In Slota and Leite (2012) the same authors successfully reconciliate semantic-based and syntax-based approaches to updating LPs: they considered different characterizations of LPs in terms of RE models (standing for *robust equivalence models*) that proved to be a more suitable semantic foundation for rule updates than SE models. A straightforward direction of research is to investigate whether these richer characterizations of LPs suit to revision operators.

Additionally, we will investigate the case of LP *merging* operators (merging can be viewed as a multi-source generalization of belief revision, see for instance (Konieczny and Pino Pérez 2002)). Indeed, it is not even known whether there exists a fully rational merging operator, i.e., that satisfies the whole set of postulates proposed by Delgrande *et al.* (2009, 2013b) for LP merging operators based on SE models.

Supplementary material

To view supplementary material for this article, please visit http://dx.doi.org/10.1017/S1471068415000101.

References

Alchourrón, C. E., Gärdenfors, P. and Makinson, D. 1985. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic 50*, 2, 510–530.

Alferes, J. J., Leite, J. A., Pereira, L. M., Przymusinska, H. and Przymusinski, T. C. 2000. Dynamic updates of non-monotonic knowledge bases. *Journal of Logic Programming 45*, 1–3, 43–70.

Cabalar, P. and Ferraris, P. 2007. Propositional theories are strongly equivalent to logic programs. *Theory and Practice of Logic Programming 7*, 6, 745–759.

Dalal, M. 1988. Investigations into a theory of knowledge base revision: Preliminary report. In *Proc. of the 7th National Conference on Artificial Intelligence (AAAI'88)*, AAAI Press/The MIT Press, Saint Paul, MN, USA, 475–479.

Delgrande, J. P., Peppas, P. and Woltran, S. 2013a. AGM-style belief revision of logic programs under answer set semantics. In *Proc. of the 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'13)*, Lecture Notes in Computer Science, vol. 8148. Springer, Corunna, Spain, 264–276.

Delgrande, J. P., Schaub, T. and Tompits, H. 2007. A preference-based framework for updating logic programs. In *Proceedings of the 9th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'07)*, Lecture Notes in Computer Science, vol. 4483. Springer, Tempe, AZ, USA, 71–83.

Delgrande, J. P., Schaub, T., Tompits, H. and Woltran, S. 2008. Belief revision of logic programs under answer set semantics. In *Proceedings of the 11th International Conference* *on Principles of Knowledge Representation and Reasoning (KR'08)*, AAAI Press, Sydney, Australia, 411–421.

Delgrande, J. P., Schaub, T., Tompits, H. and Woltran, S. 2009. Merging logic programs under answer set semantics. In *Proceedings of the 25th International Conference on Logic Programming (ICLP'09)*, Lecture Notes in Computer Science, vol. 5649. Springer, Pasadena, CA, USA, 160–174.

Delgrande, J. P., Schaub, T., Tompits, H. and Woltran, S. 2013b. A model-theoretic approach to belief change in answer set programming. *ACM Transactions on Computational Logic 14,* 2, 1–46.

Eiter, T., Fink, M., Sabbatini, G. and Tompits, H. 2002. On properties of update sequences based on causal rejection. *Theory and Practice of Logic Programming 2,* 6, 711–767.

Eiter, T., Tompits, H. and Woltran, S. 2005. On solution correspondences in answer set programming. In *Proc. of the 19th International Joint Conference on Artificial Intelligence (IJCAI'05)*, Professional Book Center, Edinburgh, Scotland, UK, 97–102.

Inoue, K. and Sakama, C. 1998. Negation as failure in the head. *Journal of Logic Programming 35,* 1, 39–78.

Katsuno, H. and Mendelzon, A. O. 1989. A unified view of propositional knowledge base updates. In *Proc. of the 11th International Joint Conference on Artificial Intelligence (IJCAI'89)*, Morgan Kaufmann, Detroit, MI, USA, 1413–1419.

Katsuno, H. and Mendelzon, A. O. 1991. On the difference between updating a knowledge base and revising it. In *Proc. of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*, Morgan Kaufmann, Cambridge, MA, USA, 387–394.

Katsuno, H. and Mendelzon, A. O. 1992. Propositional knowledge base revision and minimal change. *Artificial Intelligence 52,* 3, 263–294.

Konieczny, S. and Pino Pérez, R. 2002. Merging information under constraints: A logical framework. *Journal of Logic and Computation 12,* 5, 773–808.

Liberatore, P. and Schaerf, M. 2001. Belief revision and update: Complexity of model checking. *Journal of Computer and System Sciences 62,* 1, 43–72.

Lifschitz, V., Pearce, D. and Valverde, A. 2001. Strongly equivalent logic programs. *ACM Transactions on Computational Logic 2,* 4, 526–541.

Papadimitriou, C. M. 1994. *Computational Complexity*. Addison-Wesley, Reading, Massachusetts.

Sakama, C. and Inoue, K. 2003. An abductive framework for computing knowledge base updates. *Theory and Practice of Logic Programming 3,* 6, 671–713.

Satoh, K. 1988. Nonmonotonic reasoning by minimal belief revision. In *Proceedings of FGCS'88*, OHMSHA Ltd. Tokyo and Springer-Verlag, Tokyo, Japan, 455–462.

Schwind, N. and Inoue, K. 2013. Characterization theorems for revision of logic programs. In *Proceedings of the 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'13)*, Lecture Notes in Computer Science, vol. 8148. Springer, Corunna, Spain, 485–498.

Slota, M. and Leite, J. 2010. On semantic update operators for answer-set programs. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI'10)*, vol. 215. IOS Press, Lisbon, Portugal, 957–962.

Slota, M. and Leite, J. 2012. Robust equivalence models for semantic updates of answer-set programs. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR'12)*, AAAI Press, Rome, Italy, 156–168.

SLOTA, M. AND LEITE, J. 2014. The rise and fall of semantic rule updates based on SE-models. *Theory and Practice of Logic Programming 14*, 6, 869–907.

TURNER, H. 2003. Strong equivalence made easy: Nested expressions and weight constraints. *Theory and Practice of Logic Programming 3,* 4–5, 609–622.

ZHANG, Y. 2006. Logic program-based updates. *ACM Transactions on Computational Logic 7,* 3, 421–472.

ZHANG, Y. AND FOO, N. Y. 1997. Towards generalized rule-based updates. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97)*, Morgan Kaufmann, Nagoya, Japan, 82–88.