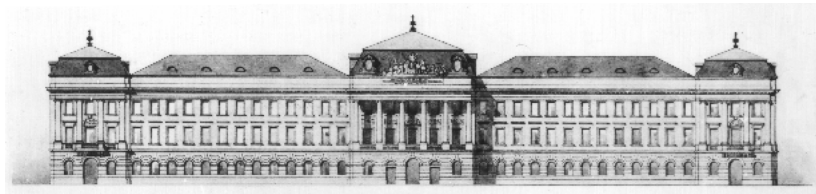
dbai

TECHNICAL REPORT



INSTITUT FÜR INFORMATIONSSYSTEME

ABTEILUNG DATENBANKEN UND ARTIFICIAL INTELLIGENCE

Abstract Preference Frameworks — a Unifying Perspective on Separability and Strong Equivalence

DBAI-TR-2013-81

Wolfgang Faber **Mirosław Truszczyński**
Stefan Woltran

Institut für Informationssysteme
Abteilung Datenbanken und
Artificial Intelligence
Technische Universität Wien
Favoritenstr. 9
A-1040 Vienna, Austria
Tel: +43-1-58801-18403
Fax: +43-1-58801-18493
sekret@dbai.tuwien.ac.at
www.dbai.tuwien.ac.at

DBAI TECHNICAL REPORT
2013

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

DBAI TECHNICAL REPORT

DBAI TECHNICAL REPORT DBAI-TR-2013-81, 2013

Abstract Preference Frameworks

Wolfgang Faber [1] **Mirosław Truszczyński** [2]
Stefan Woltran [3]

Abstract. To study mechanisms common across a variety of preference formalisms, we introduce a novel *abstract preference framework*. We use that framework to study strong equivalence in preference formalisms, a version of equivalence that guarantees semantic-preserving replacements of parts of preference theories. To this end we identify abstract postulates in the language of preference frameworks, capturing natural semantic properties of preferences, and show that they lead to characterizations, applicable in many practical settings. In a similar way, we study the *separability* of constraints and preferences. Preference languages have to capture *constraints* on the domain of interest that give rise to *intended outcomes*, and *preferences* that describe what is desirable. In many preference formalisms these two objectives are clearly separated, in some they are not. We identify abstract postulates that guarantee separability of a preference formalism and lead to its "separated" variant.

[1] University of Calabria, Email: faber@mat.unical.it
[2] University of Kentucky, Email: mirek@cs.uky.edu
[3] Technische Universität Wien, E-mail: woltran@dbai.tuwien.ac.at

Acknowledgements: The first author was supported by Regione Calabria and EU under POR Calabria FESR 2007-2013 within the PIA project of DLVSYSTEM s.r.l.. The second author was supported by the NSF grant IIS-0913459.

1 Introduction

The literature on preferences and preference formalisms is vast; the collection of articles edited by Goldsmith and Junker [10] and the monograph by Kaci [12] provide good overviews and are excellent sources of references. The main objectives of the area are to design expressive yet intuitive languages to model preferences, and to characterize the notions of optimality they define. Recently, researchers identified another fundamental problem related to preference languages, that of characterizing various notions of *equivalence* of preference theories. A particularly important one is *strong equivalence* [9]. Strong equivalence guarantees semantic-preserving replacements of parts of preference theories, the so-called *replacement property*, and is fundamental for understanding preference rewriting and modularity.

Preference formalisms are inherently nonmonotonic, that is, additional preferences can add to the set of optimal outcomes and not only remove from them. Consequently, they figured prominently in the studies of nonmonotonic logics [6]. Faber et al. [9] observed that as in other nonmonotonic formalisms, the standard notion of equivalence, requiring that two theories have the same preferred outcomes, is too weak to guarantee the replacement property. Building on earlier work on strong equivalence of logic programs with the answer-set semantics [13], Faber et al. [9] introduced and studied strong equivalence of preference theories in the language of answer-set optimization (ASO) problems.

Our goal is to identify general principles behind strong equivalence in order to extend the results of Faber et al. [9] to other preference formalisms. The main challenge is the vast diversity of preference formalisms. To overcome it, we propose the notion of an *abstract preference framework* and study strong equivalence in that language.

Preference languages have to capture two phenomena: (i) physical and logical *constraints* on the domain of interest that *must* be obeyed — they give rise to *intended* or *feasible outcomes*; and (ii) *preferences* that describe what is desirable (but not absolutely necessary) — they are employed to select out of the intended outcomes the *most preferred* or *desirable* ones. The distinction between the two is often explicit in the language, such that different constructs specify constraints and preferences, and preference theories are *pairs* consisting of a theory in a "pure" constraint formalism and another theory in a "pure" preference one. Following the terminology proposed by Faber et al. [9], we refer to these two components as *generators* and *selectors*, respectively. Examples of such preference systems include logic programs with optimization statements [5, 15], ASO problems [9], and formalisms obtained by extending common preference languages such as CP-nets with a constraint language [1].

However, in other languages such a distinction is less clear. Theories are built of statements that combine constraints with preferences in ways that make it hard to separate their effects. Examples of such languages include prioritized versions of logic programming and default logic [3, 6] and logic programs with ordered disjunctions (LPODs) [4]. In such formalisms, one can define the concepts of a generator and a selector. The problem is how to represent an arbitrary theory as a pair consisting of a generator and a selector, and whether such a "separation" is even possible. The preference framework that we introduce here provides a setting that encompasses *both* types of preference languages and allows us to study the question of "separability" in abstract terms.

The concepts of generator and selector theories suggest several forms of strong equivalence. We say that preference theories x, y are *strongly equivalent* (*strongly generator equivalent*, *strongly selector equivalent*) if for every preference (generator, selector) theory z, the extensions of x and of y with z have the same semantics, that is, the same preferred (or optimal) outcomes. The main objective when studying strong equivalence is to find characterizations which do not refer to z and can be stated entirely in terms of theories x and y being compared. Such characterizations are known for nonmonotonic logics [13, 16] and for some specific preference formalisms [7, 8, 9]. By imposing abstract postulates on the semantics of preference frameworks, we obtain the result already anticipated by Faber et al. [9], namely that for many preference formalisms the characterization of strong equivalence is fully determined by characterizations of the simpler notions of strong generator and strong selector equivalence. Under some additional postulates on preferences we also obtain a natural abstract characterization of strong selector equivalence.

The main contributions of our paper are as follows:

- We introduce the notion of an abstract preference framework. Abstract preference frameworks make weak syntactic assumptions and their semantics is specified in terms of intended outcomes and preference preorders.

- We introduce abstract *generator* and *selector* frameworks for describing constraints and preferences independently of each other. We use these notions to define *separated* abstract preference frameworks which, speaking informally, are the products of generator and selector frameworks.

- For general abstract preference frameworks we study the problem of their *separability*, that is representability by means of a separated framework, and provide sufficient conditions to guarantee that property.

- We characterize strong equivalence in separated preference frameworks in terms of characterizations of strong equivalence in generator and selector frameworks. We then lift the results to the separable case. The results apply to *all* preference formalisms which are (separable) instantiations of our framework.

- We illustrate the notion of separability by showing that logic programs with ordered disjunction are separable and, therefore, in principle, admit a simpler, separated representation.

2 Abstract Preference Frameworks

We represent preference theories by elements from some set L. We make no assumptions on the syntax of preference theories. However, we allow two preference theories to be combined into a single one. We denote that theory "conjunction" (or "union") operation by the symbol $\wedge$. We impose on $\wedge$ the properties of commutativity (the order in which the theories to be combined are listed should not matter), associativity (if more than two theories are to be combined, the order in which $\wedge$ is applied should not matter), and idempotence (combining a theory with itself should not change the meaning of the theory). We also assume the existence of a preference theory, denoted by $\top$, that when conjoined with any other preference theory, does not change the meaning of the latter (that is, does not impose any constraints and does not distinguish between any two outcomes). This is modeled by assuming that $\top$ is the unit element of $\wedge$. In preference formalisms considered in the literature, preference theories are typically represented as sets of "elementary" preference formulas. In such cases, L is the powerset of the set of preference formulas, the union operator plays the role of $\wedge$ and the empty theory that of $\top$. All such cases fall under the scope of our abstract representation. Examples include the penalty and possibilistic logics [12], answer set optimization [9], and general CP-nets [17, 11].

An algebraic structure $\mathcal{L} = \langle L, \wedge, \top \rangle$ with the properties we enumerated is a *bounded meet semilattice*. Bounded meet semilattices, for simplicity referred to from now on just as semilattices (as we do not consider any other semilattices here), arguably represent the weakest abstract desiderata on the space of preference theories. Semilattices can equivalently be thought of as partially ordered sets that have a greatest element and in which every finite set has a greatest lower bound: the relation $\preceq$ defined by $x \preceq y$ if $x \wedge y = x$ is such a partial order.

Since we do not adopt any syntactic assumptions on preference theories, we also do not make any assumptions about the nature of outcomes. We simply assume that they are elements of some set $\mathcal{I}$ of all possible outcomes. To reflect the fact that preference theories typically encompass both constraints and preferences, we specify the semantics of preference theories by means of two functions. The first one, ι, assigns to each preference theory x its set of feasible outcomes $\iota(x)$. It models the constraints contained in x. However, the constraints show up only implicitly, we see them through the effect they have on the outcomes — $\iota(x)$ consists precisely of those outcomes that satisfy all the constraints. The second function, $\geq$, assigns to x the preorder $\geq_x$ implied by the preferences in x. That preorder specifies the concept of desirability represented by x. If outcomes $\alpha, \beta \in \mathcal{I}$ satisfy $\alpha \geq_x \beta$, then α is *at least as desirable as* β. The preferences in x are, again, implicit, as is the mechanism by which they are combined to yield the preorder $\geq_x$.

These considerations lead us to the following definition of an abstract preference framework.

Definition 1. *An* abstract preference framework *(or just a preference framework) is a quadruple $\sigma = \langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$, where $\mathcal{L} = \langle L, \wedge, \top \rangle$ is a semilattice of preference theories; $\mathcal{I}$ is the space of* outcomes*; ι is a function that assigns to each $x \in L$ a set $\iota(x) \subseteq \mathcal{I}$ of* feasible *(or* intended*) outcomes for x; and $\geq$ is a function that assigns to each $x \in L$ a preorder $\geq_x$ on $\mathcal{I}$ (a binary relation that is reflexive and transitive).*

The preorder $\geq_x$ gives rise to its *strict* version $>_x$. For every outcomes $\alpha, \beta \in \mathcal{I}$, we define $\alpha >_x \beta$ if $\alpha \geq_x \beta$ and $\beta \not\geq_x \alpha$, and read $\alpha >_x \beta$ as "α *is strictly more desirable than* β in a preference theory x." For $\succ \in \{>, \geq\}$ and $S \subseteq \mathcal{I}$, we also use $\succ^S$ for the restriction of $\succ$ to S.

Let $\sigma = \langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$ be a preference framework. For every $x \in L$, we define the set of *optimal* outcomes for x:

$$\pi(x) = \{\alpha \in \iota(x) \mid \text{for every } \beta \in \iota(x),\ \beta \not>_x \alpha\}.$$

The function π assigning optimal outcomes to preference theories determines the semantics of σ.

In some situations, we consider optimality wrt a preorder $\geq_x$ but in the context of a set S of outcomes other than $\iota(x)$. In such cases we use the notation

$$\pi_S(x) = \{\alpha \in S \mid \text{for every } \beta \in S,\ \beta \not>_x \alpha\}.$$

Generators and Selectors. Preference theories typically combine constraint and preference components in a non-trivial way. Some preference theories can however be regarded as concerned exclusively with just one of these two aspects. We thus define for a preference framework $\langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$ its *generator theories* L^g, and its its *selector theories* L^π as

1. $L^g = \{x \in L \mid \text{for every } y \in L,\ \geq_{y \wedge x} = \geq_y\}$,
2. $L^\pi = \{x \in L \mid \text{for every } y \in L,\ \iota(y \wedge x) = \iota(y)\}$.

Elements in L^g (L^π, respectively) can be regarded as concerned purely with constraints (preferences, respectively) as they do not affect preferences (constraints, respectively) when conjoined with any other preference theory.

In any preference framework $\langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$, since $\top$ is the unit element of $\wedge$, $\top \in L^g$ and $\top \in L^\pi$. Moreover, for every $y, z \in L^g$,

$$\geq_{x \wedge (y \wedge z)} = \geq_{(x \wedge y) \wedge z} = \geq_{x \wedge y} = \geq_x$$

and for every $y, z \in L^\pi$,

$$\iota(x \wedge (y \wedge z)) = \iota((x \wedge y) \wedge z) = \iota(x \wedge y) = \iota(x).$$

Thus, both L^g and L^π are closed under $\wedge$. It follows that $\langle L^g, \wedge, \top \rangle$ and $\langle L^\pi, \wedge, \top \rangle$ are semilattices and, in fact, sub-semilattices of $\langle L, \wedge, \top \rangle$. They can be viewed as "one-dimensional" preference frameworks concerned with generating intended outcomes, and with selecting among all outcomes those optimal wrt preferences. One of the key questions, which we discuss later in the paper, is whether a preference framework is "separable," that is, can be reconstructed from its one-dimensional "generator" and "selector" frameworks (or of some sub-frameworks thereof).

(Non)Monotonicity. A semantics s on preference theories, that is, any function from L to $\mathcal{I}$, for instance, ι or π, is *monotone* if for every $x, z \in L$, $s(x \wedge z) \subseteq s(x)$ (equivalently, if for every $x, y \in L$ such that $x \preceq y$, $s(x) \subseteq s(y)$). In general, (the semantics of) preference formalisms are not monotone as new constraints or preferences (or both) may "promote" non-optimal outcomes to become optimal. New constraints may render outcomes that were more desirable no longer feasible and hence, eliminate them from the optimization process; new preferences may put additional importance on some non-optimal outcomes, pushing them up on the scale of desirability.

If only the preferences behave nonmonotonically, our framework is already sufficient. However, in general, we require one further concept determined by ι. An outcome $\alpha \in \mathcal{I}$ is *potentially intended* for a preference theory $x \in L$ if it is an intended outcome of $x \wedge z$, for some $z \in L$ (becomes intended under some extension of x). We denote the set of all potentially intended outcomes of x by $\mu(x)$:

$$\mu(x) = \bigcup_{y \in L} \iota(x \wedge y).$$

Clearly, for every $x \in L$, $\iota(x) \subseteq \mu(x)$ (since $x \wedge \top = x$). One can also check that if ι is monotone then $\iota(x) = \mu(x)$ (and then does not need not to be explicitly introduced).

Notions of Equivalence. The fact that preference theories are not monotonic has an important consequence when considering two preference theories as equivalent. The first concept that comes to mind is based on the comparison of optimal outcomes: two preference theories x and y are *equivalent*, written $x \equiv y$, if they have the *same* optimal outcomes, that is, if $\pi(x) = \pi(y)$. However, extending x and y with new information z may "promote" to the level of optimality outcomes that (without z in the picture) were below optimal ones, a manifestation of nonmonotonicity of preference formalisms discussed above. Hence, $\pi(x) = \pi(y)$ does not necessarily imply that $\pi(x \wedge z) = \pi(y \wedge z)$. In other words, the equivalence of x and y is too weak to guarantee mutual replaceability of x and y wrt common new information.

Consequently, an alternative concept of equivalence, based directly on the idea of replaceability is of more interest. Preference theories x and y are *strongly equivalent*, written $x \equiv_s y$, if for every preference theory z, $x \wedge z \equiv y \wedge z$, that is, $\pi(x \wedge z) = \pi(y \wedge z)$. We already mentioned the concept of separability of a preference framework into its generator and selector components. These give rise for two further definitions: preference theories x and y are *strongly g-equivalent*, in symbols $x \equiv_s^g y$, if for each $z \in L^g$, $\pi(x \wedge z) = \pi(y \wedge z)$, and *strongly π-equivalent*, in symbols $x \equiv_s^\pi y$, if for each $z \in L^\pi$, $\pi(x \wedge z) = \pi(y \wedge z)$. In this paper we study characterizations of strong equivalence in the abstract setting of preference frameworks, and show that strong equivalence of preference theories can be understood in terms of the "one-dimensional" versions of strong equivalence just introduced.

3 Separated Preference Frameworks

We start with the case of preference frameworks in which the separation between constraints and preferences is explicit. Most of the current preference formalisms are or can be extended to be in that form. We start by defining "one-dimensional" preference frameworks which are concerned only with constraints, resp. with preferences.

A *generator framework* is a triple $\sigma_g = \langle \mathcal{G}, \mathcal{I}, \iota \rangle$, where $\mathcal{G} = \langle G, \wedge, \top \rangle$ is a semilattice, $\mathcal{I}$ is a set of outcomes and ι is a function assigning to each $g \in G$ a set $\iota(g) \subseteq \mathcal{I}$ of *intended* outcomes for g. Since the concept of a potentially intended outcome in preference frameworks depends only on the function ι, it can also be defined for generator frameworks where, as before we write $\mu(x)$ for the set of all potentially intended outcomes of $x \in G$. For $x, y \in G$, we write (with some abuse of notation) $x \equiv_s y$ to denote that for every $z \in G$, $\iota(x \wedge z) = \iota(y \wedge z)$ and refer to this relation as strong equivalence in σ_g. Strong equivalence implies equivalence wrt potentially intended outcomes.

Proposition 1. *Let $\mathcal{G} = \langle G, \wedge, \top \rangle$ be a generator framework. For every $x, y \in G$, if $x \equiv_s y$ then $\mu(x) = \mu(y)$.*

Proof: For every $z \in G$ we have $\iota(x \wedge z) = \iota(y \wedge z)$. Thus, the claim follows directly from the definition of the function μ. $\square$

A *selector framework* is a triple $\sigma_\pi = \langle \mathcal{P}, \mathcal{I}, \geq \rangle$, where $\mathcal{P} = \langle P, \wedge, \top \rangle$ is a semilattice, $\mathcal{I}$ is a set of outcomes and $\geq$ is a function assigning to each $p \in P$ a preorder $\geq_p$ on $\mathcal{I}$. For $p, q \in P$, and $S \subseteq \mathcal{I}$ we write $p \equiv_{s,S} q$ if for every $r \in P$, $\pi_S(p \wedge r) = \pi_S(q \wedge r)$ and refer to this relation as strong equivalence in σ_π relative to S.

From a generator framework $\sigma_g = \langle \mathcal{G}, \mathcal{I}, \iota \rangle$ and a selector framework $\sigma_\pi = \langle \mathcal{P}, \mathcal{I}, \geq \rangle$, we can build a preference framework $\sigma_g \times \sigma_\pi = \langle \mathcal{L}, \mathcal{I}, \iota', \geq' \rangle$ by setting:

1. $\mathcal{L} = \mathcal{G} \times \mathcal{P}$, that is, $\mathcal{L}$ is the product of $\mathcal{G}$ and $\mathcal{P}$ ($L = G \times P$; for every $(x, p), (y, q) \in L$, $(x, p) \wedge (y, q) = (x \wedge y, p \wedge q)$; and the top element of $\mathcal{L}$ is $(\top, \top)$)
2. for every $(x, p) \in L$, $\iota'((x, p)) = \iota(x)$ and $\geq'_{(x,p)} = \geq_p$.

We call preference frameworks of that type *separated*. To simplify the notation we use the same symbols for the corresponding concepts coming from different semilattices. Thus, we write ι for ι' and $\geq$ for $\geq'$. We also often write $\iota(x, p)$ for $\iota((x, p))$ and similarly $\mu(x, p)$ for $\mu((x, p))$. Finally, we note that for every $(x, p) \in L$,

$$\begin{aligned} \mu(x, p) &= \bigcup_{(y,q) \in L} \iota((x, p) \wedge (y, q)) = \bigcup_{(y,q) \in L} \iota(x \wedge y, p \wedge q) \\ &= \bigcup_{y \in G} \iota(x \wedge y) = \mu(x). \end{aligned}$$

(Here μ on the left-hand side denotes the "μ"-function for $\mathcal{L}$ and the one on the right-hand side the "μ"-function for $\mathcal{G}$.)

To obtain useful characterizations of strong equivalence in separated frameworks, we impose some additional assumptions on generator and selector frameworks. These assumptions are natural and hold for many specific formalisms.

The filter property. Let $\sigma_g = \langle \mathcal{G}, \mathcal{I}, \iota \rangle$ be a generator framework. An element $y \in G$ is a *filter* if for every $x \in G$, $\iota(x \wedge y) = \iota(x) \cap \mu(y)$. A preference framework σ_g satisfies the *filter* property if for every outcome $\alpha \in \mathcal{I}$, there is a filter y such that $\mu(y) = \{\alpha\}$.

The generator promotion (GP) property. A generator framework $\sigma_g = \langle \mathcal{G}, \mathcal{I}, \iota \rangle$ satisfies the *generator promotion* property if for every $x, y \in G$ and every $\alpha, \beta \in \mu(x) \cap \mu(y)$, there is $z \in G$ such that $\iota(x \wedge z) = \iota(y \wedge z) = \{\alpha, \beta\}$.

All standard constraint satisfaction formalisms, including propositional logic and answer set programming (with choice rules), satisfy both properties.

The first result provides conditions characterizing the property $(x, p) \equiv_s^g (y, q)$. We note that the statement claiming that $\mu(x) = \mu(y)$ is justified (cf. Proposition 1).

Theorem 2. *Let $\sigma_g \times \sigma_\pi$ be a separated preference framework obtained from $\sigma_g = \langle \mathcal{G}, \mathcal{I}, \iota \rangle$ and $\sigma_\pi = \langle \mathcal{P}, \mathcal{I}, \geq \rangle$, and let σ_g satisfy the filter and the GP properties. For every $x, y \in G$ and $p, q \in P$, $(x, p) \equiv_s^g (y, q)$ if and only if $x \equiv_s y$ (in σ_g) and $>_p^\mu = >_q^\mu$, where μ is the common value of $\mu(x)$ and $\mu(y)$.*

Proof: ($\Leftarrow$) Let us consider any $z \in G$. Since $x \equiv_s y$, $\iota(x \wedge z) = \iota(y \wedge z)$, and, by the definition,

$$\iota(x \wedge z, p) = \iota(x \wedge z) = \iota(y \wedge z) = \iota(y \wedge z, q).$$

Denoting the common value of these four sets by ι we have $\iota \subseteq \mu(x) = \mu$. Since $>_p^\mu = >_q^\mu$, it follows that $>^\iota_{(x \wedge z, p)} = >^\iota_{(y \wedge z, q)}$. Thus, $\pi(x \wedge z, p) = \pi(y \wedge z, q)$.

Next, let $(z, s) \in L^g$. Then,

$$\geq_{p \wedge s} = \geq_{(x,p) \wedge (z,s)} = \geq_{(x,p)} = \geq_p \ .$$

In a similar way, we show $>_{q \wedge s} = >_q$. Since $\pi(x \wedge z, p) = \pi(y \wedge z, q)$, $\pi(x \wedge z, p \wedge s) = \pi(y \wedge z, q \wedge s)$. Since (z, s) was an arbitrary element of L^g, $(x, p) \equiv_s^g (y, q)$ follows.

($\Rightarrow$) Since $(x, p) \equiv_s^g (y, q)$, and since for every $z \in G$, $(z, \top) \in L^g$, it follows that for every $z \in G$, $\iota(x \wedge z) = \iota(y \wedge z)$. Indeed, if $\alpha \in \iota(x \wedge z) \setminus \iota(y \wedge z)$, then by the filter property, there is a filter t such that $\mu(t) = \{\alpha\}$. It follows that $\iota(x \wedge z \wedge t) = \{\alpha\}$ and $\iota(y \wedge z \wedge t) = \emptyset$. Consequently, $\iota(x \wedge z \wedge t, p) = \{\alpha\}$ and $\iota(y \wedge z \wedge t, q) = \emptyset$. Thus, trivially, $\pi(x \wedge z \wedge t, p) = \{\alpha\}$ and $\pi(y \wedge z \wedge t, q) = \emptyset$. Since $z \wedge t \in G$, this is a contradiction with $(x, p) \equiv_s^g (y, q)$. It follows that $\iota(x \wedge z) \subseteq \iota(y \wedge z)$. The converse inclusion follows by the symmetry argument and implies $\iota(x \wedge z) = \iota(y \wedge z)$. As z is an arbitrary element form G, $x \equiv_s y$ (in σ_g).

By the property just proved and Proposition 1, we have $\mu(x) = \mu(y)$. Let μ be the common value of these two sets. If $>_p^\mu \neq >_q^\mu$, then there are $\alpha, \beta \in \mu$ such that (i) $\alpha >_p \beta$ and $\alpha \not>_q \beta$, or (ii) $\alpha \not>_p \beta$ and $\alpha >_q \beta$. Wlog we assume the first alternative. By the GP property, there is $z \in G$ such that $\iota(x \wedge z) = \iota(y \wedge z) = \{\alpha, \beta\}$. Thus, $\iota(x \wedge z, p) = \iota(y \wedge z, q) = \{\alpha, \beta\}$. Since $>_p^\mu = >^\mu_{(x \wedge z, p)}$ and $>_q^\mu = >^\mu_{(y \wedge z, q)}$, it follows that $\alpha >_{(x \wedge z, p)} \beta$ and $\alpha \not>_{(y \wedge z, q)} \beta$. Consequently, $\beta \notin \pi(x \wedge z, p)$ and $\beta \in \pi(y \wedge z, q)$, a contradiction. □

This result shows that strong equivalence wrt changing generators in a separated framework can be characterized by the strong equivalence in the generator framework, and a natural condition on the preference orders. An interesting aspect here is that $G \times \{\top\}$ is (in general) a proper subset of L^g. Nevertheless, $\equiv_s^g$, which is defined relative to elements in L^g can be characterized in terms of $\equiv_s$ in σ_g, which is defined relative to elements in G.

Next, we consider strong equivalence wrt changing selectors. As before, we introduce an additional assumption.

The selector promotion (SP) property. For every $\alpha, \beta \in \mathcal{I}$ and for every $p, q \in P$, there is $t \in P$ such that (i) $\alpha >_s \beta$ if and only if $\alpha >_{s \wedge t} \beta$, where $s \in \{p, q\}$; (ii) for every $\gamma \in \mathcal{I} \setminus \{\alpha, \beta\}$, $\gamma \not>_{s \wedge t} \alpha, \beta$, where $s \in \{p, q\}$.

We note that the SP property is not particularly restrictive. It holds for many preference formalisms, in particular, for the selectors used by ASO problems [9], as well as in preference formalisms based on some forms of utility such as the penalty and possibilistic logics.

Theorem 3. *Let $\sigma_g \times \sigma_\pi$ be a separated preference framework obtained from $\sigma_g = \langle \mathcal{G}, \mathcal{I}, \iota \rangle$ and $\sigma_\pi = \langle \mathcal{P}, \mathcal{I}, \geq \rangle$, and let σ_π satisfy the SP property. For every $x, y \in G$ and $p, q \in P$, $(x, p) \equiv_s^\pi (y, q)$ if and only if $\iota(x) = \iota(y)$ and $p \equiv_{s,\iota} q$ (in σ_π) with ι the common value of $\iota(x)$ and $\iota(y)$.*

Proof: ($\Leftarrow$) Let $s \in P$. First, we note that

$$\iota(x, p \wedge s) = \iota(x) = \iota = \iota(y) = \iota(y, q \wedge s).$$

Second, we obeserve that $\pi_\iota(p \wedge s) = \pi_\iota(q \wedge s)$. Since

$$\geq_{(x,p\wedge s)} = \geq_{p\wedge s} \text{ and } \geq_{(y,q\wedge s)} = \geq_{q\wedge s},$$

the equality $\pi(x, p \wedge s) = \pi(y, q \wedge s)$ follows.

Now, let $(z, s) \in L^\pi$. It follows that

$$\iota(x) = \iota(x, p) = \iota((x, p) \wedge (z, s)) = \iota(x \wedge z, p \wedge s) = \iota(x \wedge z).$$

Thus, $\iota(x) = \iota(x \wedge z)$ and, similarly, $\iota(y) = \iota(y \wedge z)$. Since $\pi(x, p\wedge s) = \pi(y, q\wedge s)$, $\pi(x\wedge z, p\wedge s) = \pi(y\wedge z, q\wedge s)$ and, consequently, $\pi((x, p)\wedge(z, s)) = \pi((y, q)\wedge(z, s))$. Since (z, s) was an arbitrary element of L^π, $(x, p) \equiv^\pi_s (y, q)$ follows.

($\Rightarrow$) Let $(x, p) \equiv^\pi_s (y, q)$. Since for every $s \in P$, $(\top, s) \in L^\pi$, for every $s \in P$ we also have $\pi(x, p \wedge s) = \pi(y, q \wedge s)$. Let us assume that for some $\alpha \in \mathcal{I}$, $\alpha \in \iota(x) \setminus \iota(y)$. By the SP property (applied to α and to $\beta = \alpha$), there is $s \in P$ such that $\alpha \in \pi_{\mathcal{I}}(p \wedge s)$. Thus, $\alpha \in \pi(x, p \wedge s)$. Next, we note that $\alpha \notin \iota(y)$ and so, $\alpha \notin \iota(y, q\wedge s)$. It follows that $\alpha \notin \pi(y, q\wedge s)$. This is a contradiction. Thus, $\iota(x) \subseteq \iota(y)$. The converse inclusion holds by symmetry and so we have $\iota(x) = \iota(y)$.

Let ι be the common value of $\iota(x)$ and $\iota(y)$. Let $s \in P$ and let us assume that there is $\alpha \in \iota$ such that $\alpha \in \pi_\iota(p \wedge s) \setminus \pi_\iota(q \wedge s)$. Since

$$\geq_{(x,p\wedge s)} = \geq_{p\wedge s} \text{ and } \geq_{(y,q\wedge s)} = \geq_{q\wedge s},$$

it follows that $\alpha \in \pi(x, p \wedge s) \setminus \pi(y, q \wedge s)$, a contradiction. Thus, $\pi_\iota(p \wedge s) \subseteq \pi_\iota(q \wedge s)$. The converse inclusion holds by symmetry and so $\pi_\iota(p \wedge s) = \pi_\iota(q \wedge s)$. Thus, $p \equiv_{s,\iota} q$ holds. $\square$

This result shows that strong equivalence wrt changing selectors in a separated framework, can be characterized by the strong equivalence in the selector framework relative to a certain set of outcomes (the set of intended outcomes for x and y). As before, it is important to stress a non-trivial aspect of the characterization that comes from the fact that $\{\top\} \times P$ is in general a proper subset of L^π.

Finally, we present results concerning the "combined" case of strong equivalence in a separated framework.

Theorem 4. *Let $\sigma_g \times \sigma_\pi$ be a separated preference framework obtained from $\sigma_g = \langle \mathcal{G}, \mathcal{I}, \iota \rangle$ and $\sigma_\pi = \langle \mathcal{P}, \mathcal{I}, \geq \rangle$, and let σ_g satisfy the filter and the GP properties and σ_π satisfy the SP property. For every $x, y \in G$ and $p, q \in P$, $(x, p) \equiv_s (y, q)$ if and only if $x \equiv_s y$ (in σ_g), and $p \equiv_{s,\mu} q$ (in σ_π), where μ is the common value of $\mu(x)$ and $\mu(y)$.*

Proof: ($\Leftarrow$) Let $z \in G$ and $s \in P$. We have to show that $\pi((x, p) \wedge (z, s)) = \pi((y, q) \wedge (z, s))$. First, we note that

$$\iota((x, p) \wedge (z, s)) = \iota(x \wedge z) = \iota(y \wedge z) = \iota((y, q) \wedge (z, s)).$$

The "middle" equality follows from the assumption $x \equiv_s y$. Next, let us assume that for some $\alpha \in \iota$ (we write ι for the common value of $\iota(x \wedge z)$ and $\iota(y \wedge z)$), $\alpha \in \pi((x,p) \wedge (z,s)) \setminus \pi((y,q) \wedge (z,s))$. It follows that for some $\beta \in \iota$, $\beta >_{(y,q)\wedge(z,s)} \alpha$ and $\beta \not>_{(x,p)\wedge(z,s)} \alpha$. Thus, $\beta >_{q \wedge s} \alpha$ and $\beta \not>_{p \wedge s} \alpha$.

Clearly, $\alpha, \beta \in \mu$. Let $t \in P$ be the element guaranteed by the SP property for α, β, $p \wedge s$ and $q \wedge s$. Then, $\beta >_{q \wedge s \wedge t} \alpha$, $\beta \not>_{p \wedge s \wedge t} \alpha$, and for every $\gamma \neq \alpha, \beta$, $\gamma \not>_{p \wedge s \wedge t} \alpha$. Thus, $\alpha \notin \pi_\mu(q \wedge s \wedge t)$ and $\alpha \in \pi_\mu(p \wedge s \wedge t)$. Since $s \wedge t \in P$, this is a contradiction. It follows that $\pi((x,p) \wedge (z,s)) \subseteq \pi((y,q) \wedge (z,s))$. The converse inclusion follows by the symmetry argument. Thus, $\pi((x,p) \wedge (z,s)) = \pi((y,q) \wedge (z,s))$.

($\Rightarrow$) Since $(x,p) \equiv_s (y,q)$, $(x,p) \equiv_s^g (y,q)$ follows. Thus, by Theorem 2, $x \equiv_s y$. It follows now by Proposition 1 that $\mu(x) = \mu(y)$. We set $\mu = \mu(x) = \mu(y)$. To complete the proof, we need to show that $p \equiv_{s,\mu} q$.

To this end, let $s \in P$. Let us assume that for some $\alpha \in \mu$, $\alpha \in \pi_\mu(p \wedge s) \setminus \pi_\mu(q \wedge s)$. Since $\alpha \notin \pi_\mu(q \wedge s)$, there is $\beta \in \mu$ such that $\beta >_{q \wedge s} \alpha$. By the GP property, there is z such that $\alpha, \beta \in \iota(x \wedge z)$. Since $x \equiv_s y$, $\iota(x \wedge z) = \iota(y \wedge z)$. Thus, $\alpha, \beta \in \iota(y \wedge z)$, too. Since $\iota(x \wedge z) \subseteq \mu(x) = \mu$, $\alpha \in \pi(x \wedge z, p \wedge s) = \pi((x,p) \wedge (z,s))$. Moreover, $\beta >_{q \wedge s} \alpha$ implies $\beta >_{(y \wedge z, q \wedge s)} \alpha$. Since $\alpha, \beta \in \iota(y \wedge z)$, $\alpha \notin \pi((y,q) \wedge (z,s))$. Thus, $\pi((x,p) \wedge (z,s)) \neq \pi((y,q) \wedge (z,s))$, a contradiction. It follows that $\pi_\mu(p \wedge s) \subseteq \pi_\mu(q \wedge s)$. The converse inclusion follows by the symmetry argument and so, $\pi_\mu(p \wedge s) = \pi_\mu(q \wedge s)$. Since $s \in P$ is arbitrary, $p \equiv_{s,\mu} q$ follows. □

This result shows that strong equivalence in a separated framework can be characterized by the strong equivalence of the generator components and the strong equivalence of the selector components relative to the set of potentially intended outcomes for x and y. Potentially intended outcomes are relevant here (and not intended outcomes, as in the case of Theorem 3) because now generators can vary too and that may make *any* potentially intended outcome intended in the extended theory.

4 Separable Preference Frameworks

A preference framework $\sigma = \langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$ gives rise to sub-semilattices $\mathcal{L}^g = \langle L^g, \wedge, \top \rangle$ and $\mathcal{L}^\pi = \langle L^\pi, \wedge, \top \rangle$, and to two related preference frameworks:

1. $\sigma^* = \langle \mathcal{L}^*, \mathcal{I}, \iota, \geq \rangle$, where $L^* = \{x_g \wedge x_\pi \colon x_g \in L^g,\ x_\pi \in L^\pi\}$, and ι and $\geq$ are the restrictions to L^* of the corresponding functions in σ; it is easy to verify that $\mathcal{L}^*$ is a (bounded meet) semilattice and a sub-semilattice of $\mathcal{L}$

2. $\sigma^\times$, the separated framework $\sigma_g \times \sigma_\pi$ obtained from the generator framework $\sigma_g = \langle \mathcal{L}^g, \mathcal{I}, \iota \rangle$ and the selector framework $\sigma_\pi = \langle \mathcal{L}^\pi, \mathcal{I}, \geq \rangle$.

The strong equivalence concepts in $\sigma^\times$ can be expressed in terms of strong equivalence in σ_g and σ_π as discussed in the previous section. We will show that these characterizations extend to strong equivalence notions in σ^* and σ. The first result relates strong equivalence in preference frameworks $\sigma^\times$ and σ^*. Regarding items (2) and (3) in Theorems 5 and 6 below, we note that for elements in L^*, the relations $\equiv_s^g$ and $\equiv_s^\pi$ in σ and σ^* coincide, as both are defined relative to the same sets of elements, L^g and L^π, respectively.

Theorem 5. *For every $x_g, y_g \in L^g$ and $x_\pi, y_\pi \in L^\pi$:*

1. *$x_g \wedge x_\pi \equiv_s y_g \wedge y_\pi$ (in σ^*) if and only if $(x_g, x_\pi) \equiv_s (y_g, y_\pi)$*
2. *$x_g \wedge x_\pi \equiv_s^g y_g \wedge y_\pi$ if and only if $(x_g, x_\pi) \equiv_s^g (y_g, y_\pi)$*
3. *$x_g \wedge x_\pi \equiv_s^\pi y_g \wedge y_\pi$ if and only if $(x_g, x_\pi) \equiv_s^\pi (y_g, y_\pi)$.*

Proof: We prove (1) only. The other parts can be proved by providing similar sequences of identities.

Let z be an arbitrary element of L^*. Then, $z = z_g \wedge z_\pi$ for some $z_g \in L^g$ and $z_\pi \in L^\pi$. We have to show that

$$\pi(x_g \wedge x_\pi \wedge z_g \wedge z_\pi) = \pi(y_g \wedge y_\pi \wedge z_g \wedge z_\pi)$$

if and only if

$$\pi((x_g, x_\pi) \wedge (z_g, z_\pi)) = \pi((y_g, y_\pi) \wedge (z_g, z_\pi)).$$

We have:

1. $\iota(x_g \wedge x_\pi \wedge z_g \wedge z_\pi) = \iota(x_g \wedge z_g)$
2. $\iota(y_g \wedge y_\pi \wedge z_g \wedge z_\pi) = \iota(y_g \wedge z_g)$
3. $\iota((x_g, x_\pi) \wedge (z_g, z_\pi)) = \iota(x_g \wedge z_g)$
4. $\iota((y_g, x_\pi) \wedge (z_g, z_\pi)) = \iota(y_g \wedge z_g)$
5. $\geq_{x_g \wedge x_\pi \wedge z_g \wedge z_\pi} = \geq_{x_\pi \wedge z_\pi}$
6. $\geq_{y_g \wedge y_\pi \wedge z_g \wedge z_\pi} = \geq_{y_\pi \wedge z_\pi}$
7. $\geq_{(x_g, x_\pi) \wedge (z_g, z_\pi)} = \geq_{x_\pi \wedge z_\pi}$
8. $\geq_{(y_g, x_\pi) \wedge (z_g, z_\pi)} = \geq_{y_\pi \wedge z_\pi}$.

It follows that $\pi(x_g \wedge x_\pi \wedge z_g \wedge z_\pi) = \pi((x_g, x_\pi) \wedge (z_g, z_\pi))$ and $\pi(y_g \wedge y_\pi \wedge z_g \wedge z_\pi) = \pi((y_g, y_\pi) \wedge (z_g, z_\pi))$, which implies the equivalence. $\square$

Theorem 5 implies that to relate strong equivalence in $\sigma^\times$ and σ, it suffices to relate strong equivalence in σ^* and σ. To this end, we need additional concepts and assumptions.

A preference framework $\sigma = \langle L, \mathcal{I}, \iota, \geq \rangle$ is *g-complete*, if there is a homomorphism $(\cdot)^g$ from L to L^g (that is, for every $x \in L$, $x \mapsto x^g \in L^g$) such that for every $x \in L$, $\iota(x) = \iota(x^g)$. A preference framework $\langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$ is *π-complete*, if there is a homomorphism $(\cdot)^\pi$ from L to L^π (that is, for every $x \in L$, $x \mapsto x^\pi \in L^\pi$) such that for every $x \in L$, $\geq_x = \geq_{x^\pi}$. A preference framework $\langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$ is *separable* if it is both g- and π-complete and if:

1. the homomorphism $(\cdot)^g$ is the identity on L^g

2. the homomorphism $(\cdot)^\pi$ is the identity on L^π

3. for every $x \in L$, $(x^g)^\pi \in L^g$ and $(x^\pi)^g \in L^\pi$.

We now define a function $(\cdot)^* : L \to L$ (that is, for every $x \in L$, $x \mapsto x^* \in L$) by $x^* = x^g \wedge x^\pi$.

Theorem 6. *Let $\sigma = \langle \mathcal{L}, \mathcal{I}, \iota, \geq \rangle$ be a separable preference framework. For every $x, y \in L$:*

1. $x \equiv_s y$ (in σ) if and only if $x^* \equiv_s y^*$ (in σ^*)

2. $x \equiv_s^g y$ (in σ) if and only if $x^* \equiv_s^g y^*$

3. $x \equiv_s^\pi y$ (in σ) if and only if $x^* \equiv_s^\pi y^*$.

Proof: We observe that for every $x \in L$, $\pi(x) = \pi(x^*)$. Indeed,

$$\iota(x) = \iota(x^g) = \iota(x^g \wedge x^\pi) = \iota(x^*). \tag{1}$$

It follows that when computing $\pi(x)$ and $\pi(x^*)$, the same space of outcomes is considered. Moreover,

$$\geq_x = \geq_{x^\pi} = \geq_{x^g \wedge x^\pi} = \geq_{x^*} \ . \tag{2}$$

Thus, the same preorder is used in each case, too and so, the same optimal elements are specified.

Next, we observe that for every $z \in L^*$, $z^* = z \wedge w$, where $w \in L^g \cap L^\pi$. Indeed, there are $u \in L^g$ and $v \in L^\pi$ such that $z = u \wedge v$. Thus,

$$z^* = (u \wedge v)^g \wedge (u \wedge v)^\pi = u^g \wedge u^\pi \wedge v^g \wedge v^\pi.$$

Since $(\cdot)^g$ and $(\cdot)^\pi$ are identities on L^g and L^π, respectively, $z^* = u \wedge v \wedge u^\pi \wedge v^g$. Setting $w = u^\pi \wedge v^g$, we get $z^* = z \wedge w$. Moreover, $u^\pi \in L^g \cap L^\pi$ and $v^g \in L^g \cap L^\pi$. Thus, $w \in L^g \wedge L^\pi$.

Finally, we note that for every $x \in L$ and $w \in L^g \cap L^\pi$, $\pi(x \wedge w) = \pi(x)$.

(1) Let us assume that $x \equiv_s y$ (in σ) and let $z \in L^*$. Then, $\pi(x \wedge z) = \pi(y \wedge z)$ and $z = z^* \wedge w$, where $w \in L^g \cap L^\pi$. Consequently,

$$\begin{aligned}\pi(x^* \wedge z) &= \pi(x^* \wedge z \wedge w) = \pi(x^* \wedge z^*) = \pi((x \wedge z)^*) \\ &= \pi(x \wedge z) = \pi(y \wedge z) = \pi((y \wedge z)^*) \\ &= \pi(y^* \wedge z^*) = \pi(y^* \wedge z \wedge w) = \pi(y^* \wedge z)\end{aligned}$$

and $x^* \equiv_s y^*$ (in σ^*) follows.

To prove the converse implication, let $z \in L$. Then $z^* \in L^*$ and, since $x^* \equiv_s y^*$ (in σ^*), $\pi(x^* \wedge z^*) = \pi(y^* \wedge z^*)$. Thus, we have

$$\begin{aligned}\pi(x \wedge z) &= \pi((x \wedge z)^*) = \pi(x^* \wedge z^*) = \pi(y^* \wedge z^*) \\ &= \pi((y \wedge z)^*) = \pi(y \wedge z)\end{aligned}$$

and $x \equiv_s y$ (in σ) follows.
(2) Let $z \in L^g$. Then $z^* = z \wedge w$, for some $w \in L^g \cap L^\pi$ (since $L^g \subseteq L^*$). Thus,

$$\begin{aligned}\pi(x \wedge z) &= \pi(x \wedge z \wedge w) = \pi((x \wedge z)^*) = \pi(x^* \wedge z^*)\\ &= \pi(x^* \wedge z \wedge w) = \pi(x^* \wedge z).\end{aligned}$$

Similarly, we show that $\pi(y \wedge z) = \pi(y^* \wedge z)$. Consequently, the claim follows.
(3) Let $z \in L^\pi$. Then $z^* = z \wedge w$, for some $w \in L^g \cap L^\pi$ (since $L^\pi \subseteq L^*$). Thus,

$$\begin{aligned}\pi(x \wedge z) &= \pi((x \wedge z)^*) = \pi(x^* \wedge z^*)\\ &= \pi(x^* \wedge z \wedge w) = \pi(x^* \wedge z).\end{aligned}$$

Similarly, $\pi(y \wedge z) = \pi(y^* \wedge z)$. Consequently, the claim follows. □

Putting Theorems 5 and 6 together with earlier characterizations of strong equivalence in *separated* frameworks, we obtain the following result of strong equivalence in *separable* frameworks.

Theorem 7. *Let σ be a separable preference framework, such that the generator framework σ_g (of $\sigma^\times$) satisfies the filter and the GP properties and the selector framework σ_π (of $\sigma^\times$) satisfies the SP property. Then*

1. *$x \equiv_s^g y$ if and only if $x^g \equiv_s y^g$ (in σ_g) and $>^\mu_{x^\pi} = >^\mu_{y^\pi}$, where μ is the common value of $\mu(x)$ and $\mu(y)$*

2. *$x \equiv_s^\pi y$ if and only if $\iota(x) = \iota(y)$ and $x^\pi \equiv_{s,\iota} y^\pi$ (in σ_π), where ι is the common value of $\iota(x)$ and $\iota(y)$*

3. *$x \equiv_s y$ if and only if $x^g \equiv_s y^g$ (in σ_g) and $x^\pi \equiv_{s,\mu} y^\pi$ (in σ_π), where μ is the common value of $\mu(x)$ and $\mu(y)$.*

5 Characterizing Strong Equivalence in Selector Frameworks

Strong equivalence in formalisms for specifying constraints is well understood. In most cases, such formalisms satisfy a strong monotonicity property that for every theories x and y, $\iota(x \wedge y) = \iota(x) \cap \iota(y)$. It is the case for propositional logic and for all standard constraint satisfaction languages. That property implies that for every theories x and y, if $\iota(x) = \iota(y)$ then for every theory z, $\iota(x \wedge z) = \iota(y \wedge z)$. That, in turn, implies that strong and classical equivalence coincide. The situation is different in the formalisms in which the strong monotonicity property above does not hold. Answer-set programming is a prominent example of that class of constraint systems. Strong equivalence in answer set programming has been thoroughly investigated and characterizations of its basic form and numerous variants are well known [13, 16]. These results can be used to replace the condition "$x^g \equiv_s y^g$ (in σ_g)" in Theorem 7 by more specific ones (depending on a constraint formalism used).

We will now show that under some additional postulates on selector frameworks we can provide an elegant characterization of strong equivalence in selector frameworks, too.

The uniformity property. A selector framework $\sigma_\pi = \langle \mathcal{P}, \mathcal{I}, \geq \rangle$ satisfies the *uniformity* property if for every $p, q, s \in \mathcal{P}$, $\geq_p = \geq_q$ implies $\geq_{p \wedge s} = \geq_{q \wedge s}$.

The conservative promotion (CP) property. For all outcomes $\alpha, \beta \in \mathcal{I}$ and all preferences $p, q \in \mathcal{P}$ such that $\alpha \geq_p \beta$ and $\alpha \not\geq_q \beta$ there is a preference $u \in \mathcal{P}$ such that $\alpha >_{p \wedge u} \beta$ and $\alpha \not>_{q \wedge u} \beta$.

Both postulates hold for standard preference formalisms (for instance, the penalty and the possibilistic logics), and ASO problems [9].

Theorem 8. *Let $\sigma_\pi = \langle \mathcal{P}, \mathcal{I}, \geq \rangle$ be a selector framework satisfying the uniformity, CP and SP properties. For every $X \subseteq \mathcal{I}$ and for every $p, q \in P$, $p \equiv_{s,X} q$ if and only if $\geq_p^X = \geq_q^X$.*

Proof: ($\Leftarrow$) By the uniformity property, for every $s \in P$ we have $\geq_{p \wedge s}^X = \geq_{q \wedge s}^X$. Thus, $\pi_X(p \wedge s) = \pi_X(q \wedge s)$ and $p \equiv_{s,X} q$ follows.

($\Rightarrow$) Let us assume that there are $\alpha, \beta \in X$ such that $\alpha \geq_p \beta$ and $\alpha \not\geq_q \beta$. By the CP property there is $u \in P$ such that $\alpha >_{p \wedge u} \beta$ and $\alpha \not>_{q \wedge u} \beta$. Applying the SP property to $\alpha, \beta, p \wedge u$ and $q \wedge u$, we let $t \in P$ be such that (i) $\alpha >_{p \wedge u \wedge t} \beta$, $\alpha \not>_{q \wedge u \wedge t} \beta$, and (ii) for every $\gamma \in X \setminus \{\alpha, \beta\}$, $\gamma \not>_{s \wedge u \wedge t} \alpha, \beta$, where $s \in \{p, q\}$. It follows that $\beta \notin \pi_X(p \wedge u \wedge t)$ and $\beta \in \pi_X(q \wedge p \wedge q)$, a contradiction.

Thus, if $\alpha \geq_p \beta$ then $\alpha \geq_q \beta$. The converse implication follows by the symmetry argument. Thus, $\geq_p^X = \geq_q^X$. □

This theorem can be used to replace the conditions "$p \equiv_{s,X} q$" and "$x^\pi \equiv_{s,X} y^\pi$" in the earlier results with $\geq_p^X = \geq_q^X$ and $\geq_{x^\pi}^X = \geq_{y^\pi}^X$, respectively (for the appropriate value of X).

6 Discussion

Our results on separated frameworks are broadly applicable, as they rely only on the weakest assumptions on the structure of generator and selector theories. For instance, our results on separated frameworks apply to ASO problems and yield characterizations for the class of "non-ranked" problems obtained by Faber et al. ([9]; Corollary 5 and Corollary 14).

Our results on separable frameworks are also of interest. While preference frameworks that are not explicitly separated are not common, some formalisms of that kind were indeed proposed and studied. A prominent example is the formalism of *logic programs with ordered disjunction* or LPODs [4] stemming from qualitative choice logic [2]. LPODs can be cast as a preference framework and, moreover, this framework can be shown to be separable. We will now state these results formally for a slight generalization of LPODs that allows for the use of so-called *choice* rules [14], which are widely used in answer set programming.

An LPOD [4] is a finite set of rules. These rules can be *ordered disjunction rules* (*od rules*, for short) of the form

$$p_1 \times \cdots \times p_k \leftarrow p_{k+1}, \ldots, p_m, not\ p_{m+1}, \ldots, not\ p_n \tag{3}$$

or *choice rules* of the form

$$\{p_1, \ldots, p_k\} \leftarrow p_{k+1}, \ldots, p_m, not\ p_{m+1}, \ldots, not\ p_n \tag{4}$$

where (in both cases) $1 \leq k \leq m \leq n$, and each p_i $(1 \leq i \leq n)$ is an *atom* from a fixed, suitably large universe $\mathcal{U}$. We denote the positive body atoms of a rule r of form (3) or (4) by $body^+(r) = \{p_{k+1}, \ldots, p_m\}$, the negative body atoms as $body^-(r) = \{p_{m+1}, \ldots, p_n\}$, body atoms $body(r) = body^+(r) \cup body^-(r)$, head atoms $head(r) = \{p_1, \ldots, p_k\}$, and for od rules the i-th head atom as $head_i(r) = p_i$ and $head_{<i}(r) = \{p_1, \ldots, p_{i-1}\}$ $(1 \leq i \leq k)$. We will occasionally write rule bodies as $body^+(r), not\ body^-(r)$.

An interpretation $I \subseteq \mathcal{U}$ satisfies an od rule r ($I \models r$) if $I \cap head(r) \neq \emptyset$ whenever $body^+(r) \subseteq I$, and $I \cap body^-(r) = \emptyset$. A choice rule is satisfied by all interpretations. A model of a program P is an interpretation M that satisfies all of its rules.

The *reduct* r^I, as defined in [8], of an od rule r with respect to an interpretation I is $\{head_i(r) \leftarrow body^+(r)\}$ if $I \cap (body^-(r) \cup head_{<i}(r)) = \emptyset$ and $head_i(r) \in I$ $(i \leq k)$; $r^I = \{head_k(r) \leftarrow body^+(r)\}$ if $I \cap (body^-(r) \cup head(r)) = \emptyset$ (where $head_k(r)$ is the last head atom); and $r^I = \emptyset$ otherwise. The *reduct* r^I [14] of a choice rule r of the form (4) with respect to an interpretation I is $\{p \leftarrow body^+(r) \mid p \in head(r) \cap I\}$ if $I \cap body^-(r) = \emptyset$; and $r^I = \emptyset$ otherwise.

The reduct P^I of an LPOD P is $\bigcup_{r \in P} r^I$. An interpretation I is an *answer set* if I is the subset minimal model of P^I. The set of all answer sets of P is denoted by $AS(P)$.

An od rule r contributes to degree j in interpretation I ($v_I(r) = j$) if $body^+(r) \subseteq I$, $body^-(r) \cap I = \emptyset$, $head_j(r) \in I$, and $head_{<j}(r) \cap I = \emptyset$ $(j \leq k)$. If $body^+(r) \not\subseteq I$ or $body^-(r) \cap I \neq \emptyset$, the rule is irrelevant and we set $v_I(r) = 1$. If $I \not\models r$ ($body^+(r) \subseteq I$, $body^-(r) \cap I = \emptyset$, and $head(r) \cap I = \emptyset$), it is also irrelevant and $v_I(r) = 1$. Choice rules contribute to degree 1 in all interpretations. We denote by $P_I[j]$ the set of rules in an LPOD P that contribute to degree j in interpretation I.

Given two interpretations I, J of an LPOD P, $I >^c_P J$ if there is a k such that $|P_I[k]| > |P_J[k]|$, and for all $j < k$, $|P_I[j]| = |P_J[j]|$; $I \geq^c_P J$ if $I >^c_P J$ or $|P_I[k]| = |P_J[k]|$ for all $1 \leq k$. An interpretation I is a *(card)-preferred answer set* of an LPOD P if $I \in AS(P)$ and there is no $J \in AS(P)$ such that $J >^c_P I$.

Since od rules participate both in “generating” answer sets and in specifying the preference, LPODs do not have an explicit separated nature. However, LPODs are separable. To show that, we first note that LPODs give rise to a semilattice $\mathcal{L}_{od} = \langle L_{od}, \cup, \emptyset \rangle$, where L_{od} is the set of all LPODs over $\mathcal{U}$, $\cup$ plays the role of $\wedge$ and $\emptyset$ the role of $\top$, and $\sigma_{od} = \langle \mathcal{L}_{od}, 2^{\mathcal{U}}, AS, \geq^c \rangle$ forms the preference framework of LPODs under cardinality-based preferences. The notion of preferred outcomes applied to σ_{od} matches exactly the definition of (card)-preferred answer sets.

We next provide characterizations of L^g_{od} and L^π_{od}, respectively. We start with necessary conditions. For L^g_{od} the first (simple) observation is that for such LPODs cardinalities of all degrees must be equal for all interpretations.

Lemma 1. *If $P \in L^g_{od}$ then for all interpretations I, J and all $i \geq 1$, $|P_I[i]| = |P_J[i]|$.*

Proof. Assume that there is an $i \geq 0$ such that $|P_I[i]| < |P_J[i]|$ holds for two interpretations I, J (and for all $k < i$, $|P_I[i]| = |P_J[i]|$). Then for $Q = \emptyset \in L$ we have $I \geq^c_Q J$ but $I \not\geq^c_{Q \cup P} J$, hence $P \notin L^g_{od}$. $\square$

Next we show that rules in programs in L^g_{od} must contribute to degree 1 in all interpretations.

Lemma 2. *If $P \in L^g_{od}$ then each rule $r \in P$ contributes to degree 1 in all interpretations.*

Proof. Consider the maximal-truth interpretation $\mathcal{U}$, in which all atoms are true. Any rule contributes to degree 1 in $\mathcal{U}$: Choice rules contribute to degree 1 in all interpretations. For an od rule, if its negative body is non-empty ($n > m$ in (3)) then it contributes to degree 1 in $\mathcal{U}$ because $\{p_{m+1}, \ldots, p_n\} \cap \mathcal{U} \neq \emptyset$; otherwise $\{p_{k+1}, \ldots, p_m\} \subseteq \mathcal{U}$ and $p_1 \in \mathcal{U}$, so it also contributes to degree 1 in $\mathcal{U}$. Therefore $P_{\mathcal{U}}[1] = P$, and $P_{\mathcal{U}}[i] = 0$ for all $i > 0$, and together with Lemma 1 the result follows. □

The next lemma states that the condition of Lemma 2 is also a sufficient condition for programs in L^g_{od}.

Lemma 3. *If each rule in an LPOD P contributes to degree 1 in all interpretations, then $P \in L^g_{od}$.*

Proof. Given such an LPOD P and an arbitrary LPOD Q, we have that $|(P \cup Q)_I[1]| = |Q_I[1]| + |(P \setminus Q)_I[1]| = |Q_I[1]| + |(P \setminus Q)|$ and $|(Q \cup P)_I[i]| = |Q_I[i]|$ for any interpretation I. Consider interpretations I, J such that $I \geq^c_Q J$. Case 1: $|Q_I[1]| > |Q_J[1]|$, then also $|(P \cup Q)_I[1]| = |Q_I[1]| + |(P \setminus Q)| > |Q_J[1]| + |(P \setminus Q)| = |(P \cup Q)_J[1]|$. Case 2: $|Q_I[i]| > |Q_J[i]|$ for some $i > 1$ and $|Q_I[j]| = |Q_J[j]|$ for all $1 \leq j < i$, then $|(P \cup Q)_I[j]| = |Q_I[j]| = |Q_J[j]| = |(P \cup Q)_J[j]|$ for $1 < j < i$, $|(P \cup Q)_I[1]| = |Q_I[1]| + |(P \setminus Q)| = |Q_J[1]| + |(P \setminus Q)| = |(P \cup Q)_J[1]|$, and $|(P \cup Q)_I[j]| = |Q_I[j]| > |Q_J[j]| = |(P \cup Q)_J[j]|$. Case 3: $|Q_I[i]| = |Q_J[i]|$ for all $i \geq 1$, then $|(P \cup Q)_I[1]| = |Q_I[1]| + |(P \setminus Q)| = |Q_J[1]| + |(P \setminus Q)| = |(P \cup Q)_J[1]|$ and $|(P \cup Q)_I[i]| = |Q_I[i]| = |Q_J[i]| = |(P \cup Q)_J[i]|$ for $i > 1$. So in all cases $I \geq^c_{P \cup Q} J$. The other direction is analogous. □

Corollary 1. *For an LPOD P, $P \in L^g_{od}$ iff each rule in P contributes to degree 1 in all interpretations.*

Definition 2. *An LPOD rule is* purely generating *iff it is (i) a choice rule, (ii) an od rule such that $body^+(r) \cap body^-(r) \neq \emptyset$, (iii) an od rule such that $|head(r)| = 1$, or (iv) an od rule such that $head_1(r) \in body^+(r)$.*

Lemma 4. *For an LPOD P, $P \in L^g_{od}$ iff each rule in P is purely generating.*

Proof. By Corollary 1, it is sufficient to show that an LPOD rule r contributes to degree 1 in all interpretations iff it is purely generating.

Choice rules contribute to degree 1 in all interpretations by definition. If r is of type (ii) then $body^+(r) \not\subseteq I$ or $body^-(r) \cap I \neq \emptyset$ for all interpretations I, hence $v_I(r) = 1$. For rules of type (iii) and an interpretation I, either the body of r is false ($body^+(r) \not\subseteq I$ or $body^-(r) \cap I \neq \emptyset$), or the body of r is true ($body^+(r) \subseteq I$, $body^-(r) \cap I = \emptyset$) and either $head_1(r) \in I$ or $head(r) \cap I = \emptyset$, in each case $v_I(r) = 1$. For rules of type (iv) and an interpretation I, either the body of r is false; or it is true ($body^+(r) \subseteq I$, $body^-(r) \cap I = \emptyset$) and $head_1(r) \in I$, so in both cases $v_I(r) = 1$.

Assume now that r is an LPOD rule not of types (i)-(iv). So it is an od rule for which $body^+(r) \cap body^-(r) = \emptyset$, $|head(r)| > 1$, and $head_1(r) \notin body^+(r)$. We can create an interpretation I such that $body^+(r) \subseteq I$, $body^-(r) \cap I = \emptyset$, $head_1(r) \notin I$, and $head_i(r) \in I$ where $i > 1$ is the smallest such that $head_i(r) \neq head_1(r)$ (guaranteed to exist because $|head(r)| > 1$). It is easy to see that $v_I(r) = i > 1$. □

We now characterize L^{π}_{od}.

Definition 3. *An LPOD rule is* purely selecting *iff it is (a) a choice or od rule such that* $body^+(r) \cap body^-(r) \neq \emptyset$*, (b) a choice rule such that* $head(r) \subseteq body(r)$*, or (c) an od rule such that* $head_i(r) \in body^+(r)$ *and* $head_{<i}(r) \subseteq body^-(r)$*.*

Lemma 5. *For an LPOD* P*,* $P \in L^{\pi}_{od}$ *iff each rule* r *in* P *is purely selecting.*

Proof. Assume first that an LPOD P consists only of purely selecting rules, and consider some LPOD Q. To prove $AS(Q) = AS(Q \cup P)$, we show that $I \models P^I$ holds for all interpretations I (it is sufficient, since then an interpretation I is a subset minimal model of Q^I iff it also a subset minimal for $(Q \cup P)^I = Q^I \cup P^I$).

We now show that $I \models P^I$, for any interpretation I. For any rule $r \in P$ of type (a) we observe that each rule r' in r^I is such that $body^+(r') \not\subseteq I$ because $body^+(r') = body^+(r)$, $body^+(r) \cap body^-(r) \neq \emptyset$ and $body^-(r) \cap I = \emptyset$. For any rule $r \in P$ of type (b) we observe that each rule r' in r^I is such that $head_1(r') \in body^+(r')$, because for any $p \in head(r) \cap body^-(r)$ there is no rule r'' in r^I such that $p \in head(r'')$. For any rule $r \in P$ of type (c) we observe that each rule r' in r^I is one of the following: (i) if $head_i(r) \in I$ then $head_i(r) \in body^+(r)$, or (ii) if $head_i(r) \notin I$ then $body^+(r) \not\subseteq I$.

Assume now that there is a rule in P that is not of type (a)-(c). First assume that it is an od rule such that $head(r) \cap body^+(r) = \emptyset$ and $body^+(r) \cap body^-(r) = \emptyset$. Construct an interpretation $I = body^+(r)$, implying $body^-(r) \cap I = \emptyset$ and $head(r) \cap I = \emptyset$, and consider LPOD $Q = \{a \leftarrow \mid a \in I\}$ for which $AS(Q) = \{I\}$. Now $r^I = \{head_u(r) \leftarrow body^+(r)\}$ for some u, but since by construction $body^+(r) \subseteq I$ and $head_u(r) \notin I$, I is not a model of r^I and thus neither of P^I or $(Q \cup P)^I$, and therefore $I \notin AS(Q \cup P)$ and $AS(Q) \neq AS(Q \cup P)$. In the following we thus may assume that P does not contain any od rule such that $head(r) \cap body^+(r) = \emptyset$ and $body^+(r) \cap body^-(r) = \emptyset$.

Consider now that P contains a choice rule r such that $head(r) \not\subseteq body(r)$ and $body^+(r) \cap body^-(r) = \emptyset$. Construct an interpretation I such that $body^+(r) = I$, hence $body^-(r) \cap I = \emptyset$ and $p \notin I$ for all $p \in head(r) \setminus body(r)$. Consider LPOD $Q = \{a \leftarrow \mid a \in I\}$, for which $AS(Q) = \{I\}$. Consider $I' = I \cup \{p\}$ for some $p \in head(r) \setminus body(r)$. $r^{I'}$ contains one rule r^* of the form $p \leftarrow body^+(r)$ and all other rules r' in $r^{I'}$ are such that $head(r') \subseteq body(r')$, because apart from p the only true atoms in I' are in $body^+(r) = body(r')$. Observe that all rules in $r^{I'}$ are satisfied by I' and r^* is not satisfied by I. For all other choice rules s in P, we trivially obtain that I' satisfies $s^{I'}$. By our earlier assumption, all other od rules s in P are such that $head(s) \cap body^+(s) \neq \emptyset$ or $body^+(s) \cap body^-(s) \neq \emptyset$. For such an od rule s, we have one of the following: (i) $body^-(s) \cap I' \neq \emptyset$, then $s^{I'} = \emptyset$; (ii) $body^-(s) \cap I' = \emptyset$ and $head(s) \cap body^+(s) \neq \emptyset$, then $s^{I'} = \{head_j(s) \leftarrow body^+(s)\}$ and either $body^+(s) \not\subseteq I'$ or both $body^+(s) \subseteq I'$ and $head_j(s) \in I'$; (iii) $body^-(s) \cap I' = \emptyset$ and $body^+(s) \cap body^-(s) \neq \emptyset$, then $s^{I'} = \{head_j(s) \leftarrow body^+(s)\}$ and $body^+(s) \not\subseteq I'$. I' satisfies $s^{I'}$ in all cases (i)-(iii). We can thus observe that I' satisfies $P^{I'}$, and hence also $(Q \cup P)^{I'}$ and it is also a minimal model of $(Q \cup P)^{I'}$ because of r^*. Therefore $I' \in AS(Q \cup P)$ and $AS(Q) \neq AS(Q \cup P)$.

Consider now that P contains an od rule r such that $head_i(r) \in body^+(r)$ and $head_{<i}(r) \not\subseteq body^-(r)$. Let i be the smallest integer that has this property, then we can be sure that $(head_{<i}(r) \setminus$

$body^-(r)) \cap body^+(r) = \emptyset$ since otherwise either i is not minimal or r is of type (c). Similar to the previous case, construct an interpretation I such that $body^+(r) = I$, hence $body^-(r) \cap I = \emptyset$ and $p \notin I$ for all $p \in head_{<i}(r) \setminus body^-(r)$. Consider LPOD $Q = \{a \leftarrow \mid a \in I\}$, for which $AS(Q) = \{I\}$. Consider $I' = I \cup \{p\}$ for the $p \in head_{<i}(r) \setminus body^-(r)$ such that $p = head_j(r)$ and $head_{<j}(r) \nsubseteq body^-(r)$ (i.e., p is the leftmost head atom that is not in $body^-(r)$). We get $r^{I'} = \{p \leftarrow body^+(r)\}$ and clearly I' satisfies $r^{I'}$ while I does not. Now consider the other rules in P: for any choice rule s, I' trivially satisfies $s^{I'}$. For any other od rule s by our earlier assumption $head(s) \cap body^+(s) \neq \emptyset$ or $body^+(s) \cap body^-(s) \neq \emptyset$. We proceed as above: for such an od rule s, we have one of the following: (i) $body^-(s) \cap I' \neq \emptyset$, then $s^{I'} = \emptyset$; (ii) $body^-(s) \cap I' = \emptyset$ and $head(s) \cap body^+(s) \neq \emptyset$, then $s^{I'} = \{head_j(s) \leftarrow body^+(s)\}$ and either $body^+(s) \nsubseteq I'$ or both $body^+(s) \subseteq I'$ and $head_j(s) \in I'$; (iii) $body^-(s) \cap I' = \emptyset$ and $body^+(s) \cap body^-(s) \neq \emptyset$, then $s^{I'} = \{head_j(s) \leftarrow body^+(s)\}$ and $body^+(s) \nsubseteq I'$. I' satisfies $s^{I'}$ in all cases (i)-(iii). We can thus observe that I' satisfies $P^{I'}$, and hence also $(Q \cup P)^{I'}$ and it is also a minimal model of $(Q \cup P)^{I'}$ because of $r^{I'}$. Therefore $I' \in AS(Q \cup P)$ and $AS(Q) \neq AS(Q \cup P)$. □

We next show that σ_{od} is *g-complete* by providing a homomorphism $(\cdot)^g$ from L_{od} to L^g_{od}.

Definition 4. *Let r be an LPOD rule. When r is purely generating, we set $r^g = \{r\}$. Otherwise, $r^g = \{r^g_1, \ldots, r^g_j\}$ where j is the smallest integer such that $head_j(r) \in body^+(r)$ or k (the head size of r) if $head(r) \cap body^+(r) = \emptyset$; for $i < k$, r^g_i is $\{head_i(r)\} \leftarrow body^+(r), not\ (body^-(r) \cup head_{<i}(r))$ and r^g_k is $head_k(r) \leftarrow body^+(r), not\ (body^-(r) \cup head_{<k}(r))$. For an LPOD P, P^g is then $\bigcup_{r \in P} r^g$.*

By virtue of Lemma 4, P^g is guaranteed to be in L^g_{od} as all rules in P^g are purely generating. Moreover, we can show that the mapping preserves the intended outcomes, i.e. the answer sets.

Lemma 6. *For every LPOD P, $AS(P) = AS(P^g)$.*

Proof. We show that for any LPOD rule r and any interpretation I, for each interpretation $J \subseteq I$ it holds that $J \models r^I$ iff $J \models (r^g)^I$, from which the result follows. If r is purely generating, the claim follows immediately. Otherwise, let r be an od rule. We distinguish several cases:

1. $body^-(r) \cap I = \emptyset$ and $head_h(r) \in I$ for some $h \leq k$, $head_{<h}(r) \cap I = \emptyset$, and $head_{<h}(r) \cap body^+(r) = \emptyset$. In this case we have $r^I = \{head_h(r) \leftarrow body^+(r)\} = (r^g)^I$ since, for $r^g = \{r^g_1, \ldots, r^g_i\}$ as defined above, $h \leq i$. So $(r^g_h)^I = \{head_h(r) \leftarrow body^+(r)\}$ while $(r^g_j)^I = \emptyset$ for $1 \leq j < h$ since head atom $head_j(r^g_j) \notin I$, and $(r^g_j)^I = \emptyset$ for $h < j \leq i$ since $head_h(r) \in I$ and $head_h(r) \in body^-(r^g_j)$. Since $r^I = (r^g)^I$, $J \models r^I$ iff $J \models (r^g)^I$ holds for all interpretations J.

2. $body^-(r) \cap I = \emptyset$ and $head_h(r) \in I$ for some $h \leq k$, $head_{<h}(r) \cap I = \emptyset$, and $head_{<h}(r) \cap body^+(r) \neq \emptyset$. In this case we have $r^I = \{head_h(r) \leftarrow body^+(r)\}$ and $(r^g)^I = \emptyset$. For $r^g = \{r^g_1, \ldots, r^g_i\}$ as defined above, here $h > i$. Clearly $(r^g_j)^I = \emptyset$ for $1 \leq j \leq i$ since head atom $head_j(r^g_j) \notin I$. However, we note that since $head_{<h}(r) \cap I = \emptyset$ and $head_{<h}(r) \cap body^+(r) \neq \emptyset$, it follows that $b \notin I$ for some $b \in body^+(r)$, and thus $J \models r^I$ for all $J \subseteq I$ (as $b \notin J$ either). Since $J \models (r^g)^I$ trivially, $J \models r^I$ iff $J \models (r^g)^I$ holds for all $J \subseteq I$.

3. $body^-(r) \cap I = \emptyset$, $head(r) \cap I = \emptyset$, and $head_{<k}(r) \cap body^+(r) = \emptyset$ where k is the head size of r. In this case, $r^I = \{head_k(r) \leftarrow body^+(r)\} = (r^g)^I$, since, for $r^g = \{r^g_1, \ldots, r^g_j\}$ as defined above, $j = k$ (from $head_{<k}(r) \cap body^+(r) = \emptyset$) and $(r^g_k)^I = \{head_k(r) \leftarrow body^+(r)\}$ and $(r^g_i)^I = \emptyset$ for $1 \leq i < k$ because $head_i(r) \notin I$ in the respective rule heads.

4. $body^-(r) \cap I = \emptyset$, $head(r) \cap I = \emptyset$, and $head_{<k}(r) \cap body^+(r) \neq \emptyset$. In this case, $r^I = \{head_k(r) \leftarrow body^+(r)\}$ where k is the head size of r, and $(r^g)^I = \emptyset$ since, for $r^g = \{r^g_1, \ldots, r^g_j\}$ as defined above, $j \leq k$ (from $head_{<k}(r) \cap body^+(r) = \emptyset$) and $(r^g_i)^I = \emptyset$ for $1 \leq i \leq j$ because $head_i(r) \notin I$ in the respective rule heads. However, we note that since $head(r) \cap I = \emptyset$ and $head_{<k}(r) \cap body^+(r) \neq \emptyset$, it follows that $b \notin I$ for some $b \in body^+(r)$, and thus $J \models r^I$ for all $J \subseteq I$ (as $b \notin J$ either). Since $J \models (r^g)^I$ trivially, $J \models r^I$ iff $J \models (r^g)^I$ holds for all $J \subseteq I$.

5. $body^-(r) \cap I \neq \emptyset$. In this case, $r^I = \emptyset = (r^g)^I$ since in each rule $r' \in r^g$, $body^-(r') \supseteq body^-(r)$ and hence $body^-(r') \cap I \neq \emptyset$. Clearly $J \models r^I$ iff $J \models (r^g)^I$ holds for all $J \subseteq I$.

□

We next show that σ_{od} is π-*complete* by providing a homomorphism $(\cdot)^\pi$ from L_{od} to L^π_{od}.

Definition 5. *Let* r *be an LPOD rule. When* r *is purely selecting, we set* $r^\pi = \{r\}$. *Otherwise, if* r *is an od rule we define* $r^\pi = \{r^\pi_1, \ldots, r^\pi_k\}$ *where* r^π_i *is* $head_1(r) \times \cdots \times head_i(r) \leftarrow (head_i(r) \cup body^+(r)), not\ (body^-(r) \cup head_{<i}(r))$ *for* $i \leq k$, *where* k *is the head size of* r. *For choice rules* r *that are not purely selecting, let* $r^\pi = \emptyset$. *For an LPOD* P, P^π *is then* $\bigcup_{r \in P} r^\pi$.

By virtue of Lemma 5 P^π is guaranteed to be in L^π_{od}, as each rule in P^π is purely selecting. Moreover, we can show that the mapping preserves the preference relation.

Lemma 7. *For every LPOD* P, $\geq^c_P = \geq^c_{P^\pi}$.

Proof. Let P^c be the set of choice rules in P and $c_P = |P^c|$. Furthermore, let P^s be the set of purely selecting rules in P. For an od rule of form (3), let c^h_r denote k. We show that for any interpretation I, $|P_I[1]| = |P^\pi_I[1]| - c_P + \sum_{r \in P \setminus (P^c \cup P^s)} (c^h_r - 1)$ and $|P_I[i]| = |P^\pi_I[i]|$ for $i > 1$, from which the stated result follows.

For each od rule r of form (3) in P that is not purely selecting, we distinguish the following cases:

- $body^+(r) \not\subseteq I$ or $body^-(r) \cap I \neq \emptyset$: In this case, r contributes to degree 1 in I. Also all c^h_r rules in r^π contribute to degree 1.

- $body^+(r) \subseteq I$ and $body^-(r) \cap I = \emptyset$ and $head_1(r) \in I$: Also in this case r contributes to degree 1 in I. For r^π, we observe that r^π_1 contributes to degree 1 since $(\{head_1(r)\} \cup body^+(r)) \subseteq I$ and $body^-(r) \cap I = \emptyset$, while $r^\pi_2, \ldots, r^\pi_k$ ($c^h_r - 1$ rules) also contribute to degree 1 because for each r^π_i ($2 \leq i \leq k$), $body^-(r^\pi_i) \cap I \neq \emptyset$ because $head_1(r) \in (body^-(r) \cup head_{<i}(r)) \cap I$.

- $body^+(r) \subseteq I$ and $body^-(r) \cap I = \emptyset$, $head_i(r) \in I$, and $head_{<i}(r) \cap I = \emptyset$ for $i > 1$: Here, r contributes to degree i. Concerning r^π, we observe that r_i^π contributes to degree i, while all other $c_r^h - 1$ rules in r^π contribute to degree 1, since for r_j^π with $j < i$ we have $body^+(r_j^\pi) = (head_j(r) \cup body^+(r)) \not\subseteq I$ and for r_j^π with $j > i$ we have $body^-(r_j^\pi) \cap I \neq \emptyset$ because $head_i(r) \in (body^-(r) \cup head_{<j}(r)) \cap I$.

- $body^+(r) \subseteq I$ and $body^-(r) \cap I = \emptyset$ and $head(r) \cap I = \emptyset$: In this case r is not satisfied and therefore contributes to degree 1. All c_r^h rules in r^π also contribute to degree 1 because one $head_i(r) \notin I$ ($1 \leq i \leq k$) is contained in each positive body.

We obtain thus that whenever r contributes to degree $i \geq 1$ in I, 1 rule in r^π contributes to degree i in I and $c_r^h - 1$ rules in r^π contribute to degree 1 in I. For purely selecting od rules r, $r^\pi = \{r\}$ and hence these rules for each degree contribute exactly the same number of rules in P and P^π. Finally we note that choice rules in P all contribute to degree 1 in any interpretation and no choice rule is present in P^π, and hence conclude that $|P_I[1]| = |P_I^\pi[1]| - c_P + \sum_{r \in P \setminus (P^c \cup P^s)} (c_r^h - 1)$ and $|P_I[i]| = |P_I^\pi[i]|$ for $i > 1$. $\square$

Theorem 9. *σ_{od} is separable.*

Proof. We have already shown that σ_{od} is g-complete and π-complete by providing homomorphisms $(\cdot)^g$ and $(\cdot)^\pi$, respectively. Using Lemmas 4 and 5, it is easy to see that $(\cdot)^g$ and $(\cdot)^\pi$ are the identity functions on L_{od}^g and L_{od}^π, respectively.

What is left to show is that for any LPOD P, $(P^g)^\pi \in L_{od}^g$ and $(P^\pi)^g \in L_{od}^\pi$. Consider first P^g, we know $P^g \in L_{od}^g$ and by Lemma 4 this means that it contains only purely generating rules. Any $r \in P^g$ is therefore either (i) a choice rule, (ii) an od rule such that $body^+(r) \cap body^-(r) \neq \emptyset$, (iii) an od rule such that $|head(r)| = 1$, or (iv) an od rule such that $head_1(r) \in body^+(r)$. In case (i), r is either purely selecting, then $r^\pi = \{r\} \in L_{od}^g$, or otherwise $r^\pi = \emptyset \in L_{od}^g$. For (ii), r is also purely selecting, hence $r^\pi = \{r\} \in L_{od}^g$. For (iii), observe that for all $r' \in r^\pi$, $head(r') \subseteq head(r)$ and therefore $|head(r')| = 1$ implying that $r^\pi \in L_{od}^g$. For (iv), observe that for all $r' \in r^\pi$, $head_1(r') = head_1(r)$ and $body^+(r') \supseteq body^+(r)$, thus $head_1(r') \in body^+(r')$, implying $r^\pi \in L_{od}^g$. We thus obtain $(P^g)^\pi \in L_{od}^g$.

Consider now P^π, we know $P^\pi \in L_{od}^\pi$ and by Lemma 5 this means that it contains only purely selecting rules. Any $r \in P^\pi$ is therefore either (a) a choice or od rule such that $body^+(r) \cap body^-(r) \neq \emptyset$, (b) a choice rule such that $head(r) \subseteq body(r)$, or (c) an od rule such that $head_i(r) \in body^+(r)$ and $head_{<i}(r) \subseteq body^-(r)$. In case (a) and (b), r is purely generating, hence $r^g = \{r\} \in L_{od}^\pi$. For case (c), r^g consists of choice rules $r_j^g = \{head_j(r)\} \leftarrow body^+(r), not\ body^-(r)$ for $1 \leq j < i$, which are purely selecting since $head(r_j^g) \subseteq body^-(r_j^g)$, and either another choice rule $r_i^g = \{head_i(r)\} \leftarrow body^+(r), not\ body^-(r)$ (purely selecting as $head(r_i^g) \subseteq body^+(r_i^g)$) or (if i is equal to the head size) an od rule $r_i^g = head_i(r) \leftarrow body^+(r), not\ body^-(r)$, which is also purely selecting as $head_1(r_i^g) \in body^+(r_i^g)$ (and $head_{<1}(r_i^g) = \emptyset \subseteq body^-(r_i^g)$). $r^g \in L_{od}^\pi$ follows. $\square$

The last two results suggest that the formalism of LPODs can be presented in a simplified form as a separated one based on combinations of a generator from L_{od}^g and a selector from L_{od}^π. Combined with the earlier results, they also provide characterizations of strong equivalence for LPODs.

7 Conclusions

We introduced abstract preference frameworks as a unifying language to study fundamental aspects of preferences that cut across many specific formalisms. We showed the effectiveness of the abstract setting by using it to study the problems of strong equivalence and separability.

Our paper suggests several open problems. First, our conditions for separability are quite restrictive as they require that L^g and L^π be used as generators and selectors, respectively. These sets were defined to contain every element that could possibly be regarded as a generator or a selector, respectively. They may, however, contain redundant elements. Using "non-redundant" sub-semilattices of L^g and L^π may lead to simpler and more natural "separations." Next, many specific preference formalisms explicitly "rank" preferences according to their importance. It is therefore of interest to find an abstract account for ranks in preference frameworks and extend our results to that broader setting. Finally, a comprehensive study of specific preference formalisms from the perspective of abstract preference frameworks (we alluded to some results here and considered in more detail just one, LPODs) is also left for the future.

References

[1] C. Boutilier, R. Brafman, C. Domshlak, H. Hoos, and D. Poole. Preference-based constrained optimization with CP-nets. *Computational Intelligence*, 20:137–157, 2004.

[2] G. Brewka, S. Benferhat, and D. Le Berre. Qualitative choice logic. In D. Fensel, F. Giunchiglia, D. L. McGuinness, and M.-A. Williams, editors, *Proceedings of the 8th International Conference on Principles of Knowledge Representation and Reasoning (KR 2002)*, pages 158–169. Morgan Kaufmann, 2002.

[3] G. Brewka and T. Eiter. Preferred answer sets for extended logic programs. *Artificial Intelligence*, 109(1-2):297–356, 1999.

[4] G. Brewka, I. Niemelä, and T. Syrjänen. Logic programs with ordered disjunctions. *Computational Intelligence*, 20(2):335–357, 2004.

[5] F. Buccafurri, N. Leone, and P. Rullo. Enhancing Disjunctive Datalog by Constraints. *IEEE Transactions on Knowledge and Data Engineering*, 12(5):845–860, 2000.

[6] J. P. Delgrande, T. Schaub, H. Tompits, and K. Wang. A classification and survey of preference handling approaches in nonmonotonic reasoning. *Computational Intelligence*, 20(2):308–334, 2004.

[7] W. Faber and K. Konczak. Strong order equivalence. *Annals of Mathematics and Artificial Intelligence*, 47(1–2):43–78, June 2006.

[8] W. Faber, H. Tompits, and S. Woltran. Notions of strong equivalence for logic programs with ordered disjunction. In G. Brewka and J. Lang, editors, *Proceedings of the 11th International*

Conference on Principles of Knowledge Representation and Reasoning (KR 2008), pages 433–443. AAAI Press, 2008.

[9] W. Faber, M. Truszczyński, and S. Woltran. Strong equivalence of qualitative optimization problems. In G. Brewka, T. Eiter, and S. A. McIlraith, editors, *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR 2012)*, pages 188–198. AAAI Press, 2012.

[10] J. Goldsmith and U. Junker, editors. *Special Issue on Preferences*, volume 29(4) of *AI Magazine*. 2008.

[11] J. Goldsmith, J. Lang, M. Truszczyński, and N. Wilson. The computational complexity of dominance and consistency in CP-nets. *Journal of Artificial Intelligence Research*, 33:403–432, 2008.

[12] S. Kaci. *Working with Preferences*. Springer, 2011.

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

[14] P. Simons. Extending the stable model semantics with more expressive rules. In M. Gelfond, N. Leone, and G. Pfeifer, editors, *Proceedings of the 5th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 1999)*, volume 1730 of *Lecture Notes in AI*, pages 305–316. Springer, 1999.

[15] P. Simons, I. Niemelä, and T. Soininen. Extending and implementing the stable model semantics. *Artificial Intelligence*, 138:181–234, 2002.

[16] H. Turner. Strong equivalence made easy: nested expressions and weight constraints. *Theory and Practice of Logic Programming*, 3:609–622, 2003.

[17] N. Wilson. Extending CP-nets with stronger conditional preference statements. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI 2004)*, pages 735–741. AAAI Press, 2004.