A new logic for jointly representing hard and soft constraints

Jan Maly and Stefan Woltran

Institute of Logic and Computation, TU Wien, Austria

Abstract. Soft constraints play a major role in AI, since they allow to restrict the set of possible worlds (obtained from hard constraints) to a small fraction of preferred or most plausible states. Only a few formalisms fully integrate soft and hard constraints. A prominent example is Qualitative Choice Logic (QCL), where propositional logic is augmented by a dedicated connective and preferred models are discriminated via acceptance degress determined by this connective. In this work, we follow an analogous approach in terms of syntax but propose an alternative semantics. The key idea is to assign to formulas a set of models plus a partial relation on these models. Preferred models are then obtained from this partial relation. We investigate properties of our logic which demonstrate that our semantics shows some favorable behavior compared to QCL. Moreover, we provide a partial complexity analysis of our logic.

1 Introduction

One major problem for handling preferences or soft constraints is often the sheer amount of alternatives. This makes the representation of preferences as an order relation on the set of alternatives impractical in many applications. Languages that represent preferences succinctly help to circumvent this problem, e.g. CP-Nets (introduced in [3]) and its many extensions. However, for many applications, it is even impractical to represent the set of alternatives explicitly, especially if possible combinations of objects are considered, e.g. collections of goods in fair division problems [4] or sets of possible outcomes in decision problems [1].

It is thus a natural, yet challenging, approach to apply preferences without explicitly materializing the set of all alternatives. Indeed, many logical languages in the area of knowledge representation propose an integration of hard and soft constraints, see e.g. [6], but often the two concepts remain conceptually separated. However, integrating the representation of the preference relation in the representation of the set of alternatives offers several advantages. Apart from obvious gains in usability and ease of implementation it could also lead to higher efficiency because the construction of the preference relation can be limited to the actual alternatives. The most prominent approach in this direction is Qualitative Choice Logic (QCL) [5]. QCL extends propositional logic to a language that can represent sets of alternatives and preferences, i.e. hard and soft constraints, at once, by introducing a dedicated connective called ordered disjunction. We would argue, however, that the actual semantics of QCL behaves unintuitively in many situations and is, in the end, not entirely convincing (problems of QCL are discussed in more detail in Section 5). Therefore, we propose a new logic that is inspired by, and has the same syntax as, QCL with a completely new semantics.

Our logic relies on a simple idea for the semantics, where every formula is not only assigned a set of models but also a partial relation on these models. The main aim of the paper is to evaluate our approach along the following axes of desired properties:

1. **Extending propositional logic:** Formulas without specified preference should behave like propositional formulas.
2. **Rules of propositional logic:** As many rules of propositional logic such as associativity and distributivity as possible should hold in the logic.
3. **Expressiveness:** Every partial order on every set of models should be expressible.
4. **Simulate QCL on "basic choice formulas":** Basic choice formulas are a fragment of QCL that behaves very naturally, hence our logic should be equivalent to QCL on this fragment.
5. **Good computational properties:** The complexity of problems like checking whether a model is preferred to another should remain on a complexity level comparable to similar problems in propositional logic.

As we will show, the first, third and fourth requirements are fully satisfied by our logic. Concerning the second property, we show that many important rules of classical logic hold in our logic as well, but some don't. The fifth property is still open as we only have preliminary results on the complexity of our logic.

The rest of the paper is structured roughly around these properties. The next section introduces the logic and discusses its definition; as we will see the first property follows directly from the definition. Section 4 discusses equivalence of formulas in our logic and shows which rules of propositional logic are satisfied by our logic. We then proceed with the expressiveness of our logic addressing the third property from the above list. Section 5 then is concerned with the relation between our logic and QCL. Finally, Section 6 presents some partial results on the complexity of our logic.

Previous and Related Work. In the past, several preference logics were proposed. Perhaps the most influential work in this area is von Wright's paper "The logic of preferences" [12] that could even be considered as reference text on preference logics according to van Benthem [11]. Von Wright and his many successors (see, for example, [8] and [10]) develop logic languages that allow to reason (only) about preferences. The aim of this work is to establish a logic that allows to reason about truth and preferences, or in other words to specify hard and soft constraints at the same time. This concept is quite well studied for first order logic, mostly from a database perspective (see, for example, [9]). For propositional logic, the problem seems to be less well studied. Besides the aforementioned QCL and the closely related Conjunctive Choice Logic [2], nested circumscription [7] provides an alternative idea that provides a handle for minimization of models of subformulas.

2 Introducing Our Logic

Syntax. Syntactically, our logic is propositional logic with the connectives $\wedge$, $\vee$ and $\neg$ extended by an additional connective $\succeq$.

Definition 1 *The set $\mathcal{PF}(V)$ of* formulas of preference logic *over a set of variables V is defined recursively as:*

- $v \in \mathcal{PF}(V)$ *for all* $v \in V$.
- $(A \wedge B) \in \mathcal{PF}(V)$ *if* $A, B \in \mathcal{PF}(V)$.
- $(A \vee B) \in \mathcal{PF}(V)$ *if* $A, B \in \mathcal{PF}(V)$.
- $(\neg A) \in \mathcal{PF}(V)$ *if* $A \in \mathcal{PF}(V)$.
- $(A \succeq B) \in \mathcal{PF}(V)$ *if* $A, B \in \mathcal{PF}(V)$.

For every V, we define $\bot = A \wedge \neg A$ for some fixed $A \in \mathcal{PF}(V)$ and $\top = \neg\bot$.

The intended meaning of $A \succeq B$ is A or B but preferably A. Therefore, from the perspective of truth $A \succeq B$ is equivalent to $A \vee B$.

Definition 2 *Assume $A \in \mathcal{PF}(V)$. Then the* propositional projection *of A, denoted $prop(A)$, is the propositional formula obtained from replacing every occurrence of $\succeq$ in A with $\vee$.*

Semantics. The semantics of our logic relies on two constituents, one related to truth and one related to preferences. The former is given by the classical evaluation of the propositional projection of the formula.

Definition 3 *Let $A \in \mathcal{PF}(V)$. We call a set $M \subseteq V$ a* model *of A and write $M \models A$ if it is a model of $prop(A)$. $\mathcal{M}_A$ denotes the set of all models of A.*

Preferences, on the other hand, are represented by a relation between models. Let $A \in \mathcal{PF}(V)$ and $M, N \in \mathcal{M}_A$. We define a relation R_A on $\mathcal{M}_A$ and write $M \geq_A N$ if M is preferred to N for A. As usual, we write $M >_A N$ for $M \geq_A N$ and not $M \leq_A N$ as well as $M =_A N$ for $M \geq_A N$ and $M \leq_A N$. The definition of R_A is based inductively on the relations of the subformulas of A and depends on the form of A. In the following we introduce and discuss the definition for every type of formula separately. For atomic formulas, there is no reason to prefer any model over any other model.

Definition 4 $R_A = \emptyset$ *for* $A = v$ *with* $v \in V$.

For $A = \neg B$, we observe that, in general, R_B contains no information on the models of $\neg B$. For example, for $B = ((a \wedge c) \succeq (b \wedge c))$, it seems impossible to us, to use any preferences between the models $\{a, c\}, \{a, b, c\}, \{b, c\}$ of B to justify any preferences between the models $\{a\}, \{b\}, \{a, b\}, \{c\}, \emptyset$ of A. To circumvent this problem, we use the rather crude method of erasing all preferences.

Definition 5 $R_A = \emptyset$ *for* $A = \neg B$.

In practice, this is equivalent to using positive and negative literals instead of $\neg$. We chose the presented definition, among other reasons, in order to follow the same intuition as QCL.

One of the major questions for formulas of the form $A = B \circ C$ for $\circ \in \{\wedge, \vee, \succeq\}$ is, how to deal with conflicting preferences in R_B and R_C. In the case $A = B \succeq C$, we can use the additional information that the subformula B is "more important" than C. Hence, the preferences of B should "overwrite" the preferences of C. The preferences of C are only considered for models M and N if both do not satisfy B. Additionally, we, of course, need to add new preferences that codify that B is preferred to C.

Definition 6 *For $A = B \succeq C$, $M \geq_A N$ if:*

- *$M \geq_B N$;*
- *$M, N \notin \mathcal{M}_B$ and $M \geq_C N$;*
- *$M \in \mathcal{M}_B$ and $N \notin \mathcal{M}_B$.*

In the other two cases, where both subformulas are equally important, there are two approaches to dealing with conflicting preferences. The first approach is combining the preferences of both subformulas. Intuitively, this means that if there is a reason to prefer M over N in one of the subformulas of A, then M is preferred to N in A. We believe that this is the correct notion for formulas of the form $A = B \wedge C$.

Definition 7 *For $A = B \wedge C$, $M \geq_A N$ if:*

- *$M \geq_B N$ or $M \geq_C N$;*
- *there exists $N' \in \mathcal{M}_A$, such that $M \geq_A N' \geq_A N$;*

The second approach is to demand that both subformuals agree in their preference and ignoring inconclusive preferences. This approach appears to be a natural fit for formulas of the form $A = B \vee C$. We additionally have to treat the case that M is preferred to N in one of the subformulas, say B, but at least one of the two models does not satisfy C. In this case, we keep the preference of from B unless M does not model C but N does, as this can be seen as a preference of N over M for C.

Definition 8 *For $A = B \vee C$, $M \geq_A N$ if:*

- *Either $M >_B N$ and $M >_C N$ holds or $M =_B N$ and $M =_C N$ holds.*
- *Either $N \notin \mathcal{M}_B$ and $M \geq_C N$ holds or $N \notin \mathcal{M}_C$ and $M \geq_B N$ holds.*

Example 9 *Let $A = (a \succeq b) \wedge (b \succeq a)$ over $V = \{a, b\}$. We have $\mathcal{M}_A = \{\{a, b\}, \{a\}, \{b\}\}$ and $\{a, b\} >_A \{a\}$, $\{a, b\} >_A \{b\}$, and $\{a\} =_A \{b\}$. For $B = (a \succeq b) \vee (b \succeq a)$, we have $\mathcal{M}_B = \mathcal{M}_A$ but R_B is empty.*

It is possible to use the first approach for $\vee$ and the second one for $\wedge$. The resulting "strong or" and "weak and" are expressible in our language as we will see in Section 4. This concludes the definition of R_A.

Definition 10 *We call* $(\mathcal{M}_A, R_A)$ *the* evaluation *of* A. *We say that two formulas* A, B *are equivalent, in symbols* $A \equiv B$, *if* $(\mathcal{M}_A, R_A) = (\mathcal{M}_B, R_B)$.

We observe that the evaluation of $\bot = A \wedge \neg A$ is given by $(\emptyset, \emptyset)$ and for $\top = \neg\bot$ we have as an evalution $(2^V, \emptyset)$. Furthermore, we observe that for any A, that $(A \vee \bot)$ yields the same evaluation as A, while $(A \vee \top)$ yields the same evaluation as $\top$.

Another issue is the explicit transitivity in Definition 7. Otherwise, for example, $A = (c \vee (a \succeq b)) \wedge (a \vee (b \succeq c))$ would lead to a non transitive relation containing $a \geq_A b$ and $b \geq_A c$ but not $a \geq_A c$. We next show that R_A is indeed transitive, even though we only explicitly specified this for conjunction.

Proposition 11 *Let* A *be a formula and* $M, N, O \in \mathcal{M}_A$. *Then* $M \geq_A N$ *and* $N \geq_A O$ *implies* $M \geq_A O$.

Proof. We prove the claim by an induction on the formula complexity of A: The cases $A = a_i$, $A = B \wedge C$ and $A = \neg B$ are clear.

Now assume $A = B \vee C$. First assume $M, N, O \models B \wedge C$. Then we know $M \geq_B N \geq_B O$ and $M \geq_C N \geq_C O$ and hence by induction $M \geq_B O$ and $M \geq_C O$. Obviously either $M >_B O$ or $M =_B O$ holds. Observe that $M >_B O$ holds if and only if $M >_B N$ or $N >_B O$ holds. We assume $M >_B N$. Then $M \geq_A N$ can only be true if $M >_C N$ holds. Hence $M >_C O$ and therefore $M \geq_A O$ holds. On the other hand $M =_B O$ implies $M =_C O$ by a symmetric argument and hence also $M \geq_A O$. Now assume $O \not\models B$. Then $N \geq_C O$ must hold. Furthermore, because N must be a model of C, we know that $M \geq_C N$ must hold. Therefore, by induction, $M \geq_C O$, which implies $M \geq_A O$, because $O \not\models B$. Observe that $N \not\models B$ and $N \geq_A O$ imply $O \not\models B$ and hence $M \geq_A O$ by the argument above. Furthermore $M \not\models B$ and $M \geq_A N$ imply $N \not\models B$, hence again $M \geq_A O$. The remaing cases, $O \not\models C$, $N \not\models C$ and $M \not\models C$ are symmetric.

Finally, assume $A = B \succeq C$. If $M \geq_B N \geq_B O$ or $M, N, O \in \mathcal{M}_C \setminus \mathcal{M}_B$ and $M \geq_C N \geq_C O$ we get $M \geq_A O$ by induction. Observe that the only remaing possible cases are either $M \geq_B N$ and $O \in \mathcal{M}_C \setminus \mathcal{M}_B$ or $M \in \mathcal{M}_B$, $N, O \in \mathcal{M}_C \setminus \mathcal{M}_B$ and $N \geq_B O$. In both cases we know $M \in \mathcal{M}_B$ and $O \in \mathcal{M}_C \setminus \mathcal{M}_B$, hence $M \geq_A O$. □

In general, the relation $\geq_A$ is not reflexive, hence it is not a partial order. It would be possible to change this, by defining $M =_A M$ for all $M \in \mathcal{M}_A$. From a technical standpoint, the resulting logic is nearly equivalent and basically all results in this paper would carry over to this logic. (Obviously, the results on expressiveness would change a bit, as only reflexive relations could be expressed.) However, adding reflexiveness would complicate notation in some places, therefore we omitted it from the definition.

The main aim of the relation R_A is to determine preferred models. We define them next.

Definition 12 *Let* $A \in \mathcal{PF}(V)$. *We say that a model* $M \in \mathcal{M}_A$ *is a* most preferred model *of* A *if there is no* $N \in \mathcal{M}_A$ *such that* $N >_A M$. *We write* *pref*(A) *for the set of most preferred models of* A.

3 Equivalence-Preserving Replacements

Two important concepts of equivalence of formulas are semantic equivalence and replacement equivalence. These two concepts coincide for propositional logic, but typically are distinct concepts in logics which are concerned with preferences. We show that the two concepts are closely related for our logic.

Replacements are denoted via $C[A/B]$, which stands for the formula obtained from C by replacing an occurrence of A by B.

Definition 13 *We say two formulas* $A, B \in \mathcal{PF}(V)$ *are* replacement pref-equivalent *if for all* $C \in \mathcal{PF}(V)$, $\mathit{pref}(C[A/B]) = \mathit{pref}(C)$.

Theorem 14 $A, B \in \mathcal{PF}(V)$ *are replacement equivalent iff* $A \equiv B$.

Proof. $A \equiv B$, i.e. $(\mathcal{M}_A, R_A) = (\mathcal{M}_B, R_B)$, implies replacement pref-equivalence between A and B because the evaluation of a formula only depends on the evaluation of its subformulas and not on their syntax. So assume there are replacement pref-equivalent formulas A and B but $(\mathcal{M}_A, R_A) \neq (\mathcal{M}_B, R_B)$. First assume $\mathcal{M}_A \neq \mathcal{M}_B$. Then, without loss of generality, there is a model $M \in \mathcal{M}_A$ such that $M \notin \mathcal{M}_B$ holds. Consider the formula $D_M = \bigwedge_{v \in M} v \wedge \bigwedge_{v \in V \setminus M} \neg v$. Now let $C = A \wedge D_M$. Then $\mathcal{M}_C = \{M\}$ and $\mathcal{M}_{C(A/B)} = \emptyset$. Hence, $\mathit{pref}(C) = \{\{M\}\}$ and $\mathit{pref}(C[A/B]) = \emptyset$. Now assume $\mathcal{M}_A = \mathcal{M}_B$ and $R_A \neq R_B$. Wlog, there is $(M, N) \in R_A$ and $(M, N) \notin R_B$. For $C = A \wedge (D_N \succeq D_M)$ (with D_M, D_N as used above), we get $\mathit{pref}(C[A/B]) = \{\{N\}\}$ while $\mathit{pref}(C) = \{\{M, N\}\}$. $\square$

In what follows, we explore which rules of equivalence from propositional logic hold in our logic. Before analysing several rules, we provide a different characterization for R_A in order to ease proofs. For sets X and Y we shall write $X\big|_Y$ for $X \cap Y$; however, we use $X\big|_Y$ also as short hand for $X \cap (Y \times Y)$ when clear from the context. Furthermore, for a set of tuples X we write X^r for the reverse set, i.e. $X^r := \{(b, a) \mid (a, b) \in X\}$. We write $X \triangle Y$ for the symmetric difference of X and Y, i.e $X \triangle Y := (X \cup Y) \setminus (X \cap Y)$. Finally, we write $\mathit{trcl}(R)$ for the transitive closure of a relation R.

Proposition 15 *Let* $A, B, C \in \mathcal{PF}(V)$, $v \in V$. *Then, the following holds:*

- *If* $A = v$ *or* $A = \neg B$, *then* $R_A = \emptyset$.
- *If* $A = B \wedge C$, *then* $R_A = \mathit{trcl}\big((R_B \cup R_C)\big|_{\mathcal{M}_A}\big)$.
- *If* $A = B \vee C$, *then* $R_A = ((R_B \cap R_C) \setminus (R_B^r \triangle R_C^r)) \cup R_B\big|_{\mathcal{M}_A \times \mathcal{M}_B \setminus \mathcal{M}_C} \cup R_C\big|_{\mathcal{M}_A \times \mathcal{M}_C \setminus \mathcal{M}_B}$.
- *If* $A = B \succeq C$, *then* $R_A = R_B \cup R_C\big|_{\mathcal{M}_B \setminus \mathcal{M}_A} \cup \big(\mathcal{M}_B \times (\mathcal{M}_C \setminus \mathcal{M}_B)\big)$.

Proof. We prove this by an induction on the formula complexity of a formula A: The cases $A = v$ and $A = \neg B$ are clear.

Assume $A = B \wedge C$. It is clear that $(M, N) \in (R_B \cup R_C)\big|_{\mathcal{M}_A}$ iff $M, N \models B$, $M, N \models C$ and $(M, N) \in R_B$ or $(M, N) \in R_C$ holds. Transitive closure of R_A follows by defintion of the semantics for conjunction cf. Definition 7.

$\top$,$\bot$-rules	$A \wedge \top \equiv A$; $A \wedge \bot \equiv \bot$; $A \vee \bot \equiv A$; $A \vee \top \equiv \top$; $\top \succeq A \equiv \top$; $A \succeq \bot \equiv A$
De Morgan's law	$\neg(A \wedge B) \equiv (\neg A \vee \neg B)$; $\neg(A \vee B) \equiv (\neg A \wedge \neg B)$
Triple Negation	$\neg\neg\neg A \equiv \neg A$
Commutativity	$A \wedge B \equiv B \wedge A$; $A \vee B \equiv B \vee A$
Associativity	$(A \vee B) \vee C \equiv A \vee (B \vee C)$; $(A \succeq B) \succeq C \equiv A \succeq (B \succeq C)$
Absorption	$A \wedge (A \vee B) \equiv A$; $A \wedge (A \succeq B) \equiv A$; $A \succeq (A \wedge B) \equiv A$
Neutrality	$A \wedge A \equiv A$; $A \vee A \equiv A$; $A \succeq A \equiv A$

Table 1. Laws of our preference logic

Now assume $A = B \vee C$. First observe that $(M, N) \in (R_B \cap R_C) \setminus (R^r_A \triangle R^r_B)$ is equivalent to $(M, N) \in R_B$, $(M, N) \in R_C$ and (N, M) either in R_B and R_C or in neither. By induction, this is equivalent to $M =_B N$ and $M =_C N$ in the first case and equivalent to $M >_B N$ and $M >_C N$ in the second. Now observe that $(M, N) \in R_B\big|_{\mathcal{M}_A \times \mathcal{M}_B \setminus \mathcal{M}_C}$ holds if and only if $N \not\models C$ and $M \geq_B N$. Finally, $(M, N) \in R_C\big|_{\mathcal{M}_A \times \mathcal{M}_C \setminus \mathcal{M}_B}$ iff $N \not\models B$ and $M \geq_C N$.

Finally, assume $A = B \succeq C$. By induction $(M, N) \in R_B$ iff $M \geq_B N$. $(M, N) \in R_C\big|_{\mathcal{M}_C \setminus \mathcal{M}_B}$ iff $M, N \not\models B$ and $M \geq_C N$. Finally, $(M, N) \in (\mathcal{M}_B \times (\mathcal{M}_C \setminus \mathcal{M}_B))$ iff $M \models B$ and $N \not\models C$. □

Many classical rules of equivalence hold also for our preference logic.

Theorem 16 *The equivalences in Table 1 hold.*

Proof. For all $A \equiv B$ in table 1, $\mathcal{M}_A = \mathcal{M}_B$ follows easily from the rules of propositional logic. We show three of the $\top, \bot$-rules. First observe $R_{A \wedge \top} = \mathit{trcl}(R_A \cup \emptyset)\big|_{\mathcal{M}_A} = R_A$. Then $R_{A \vee \bot} = R_A\big|_{\mathcal{M}_A \setminus \emptyset} \cup \emptyset\big|_{\emptyset \setminus \mathcal{M}_A} \cup (R_A \cap \emptyset) \setminus (R^r_A \triangle \emptyset) = R_A$. And finally $R_{A \vee \top} = R_A\big|_{\mathcal{M}_A \setminus 2^V} \cup \emptyset\big|_{2^V} \cup (R_A \cap \emptyset) \setminus (R^r_A \triangle \emptyset) = R_\top = \emptyset$. De Morgan's law and triple negation follow immediately from the definition of $\neg$ and the fact that they hold in classical logic. Commutativity is also clear by definition. We show the associativity of $\succeq$:

$$\begin{aligned}
R_{(A \succeq B) \succeq C} = {} & (R_A \cup R_B\big|_{\mathcal{M}_B \setminus \mathcal{M}_A} \cup (\mathcal{M}_A \times (\mathcal{M}_B \setminus \mathcal{M}_A))) \cup \\
& R_C\big|_{\mathcal{M}_C \setminus (\mathcal{M}_A \cup \mathcal{M}_B)} \cup ((\mathcal{M}_A \cup \mathcal{M}_B) \times (\mathcal{M}_C \setminus (\mathcal{M}_A \cup \mathcal{M}_B))) = \\
& R_A \cup R_B\big|_{\mathcal{M}_B \setminus \mathcal{M}_A} \cup R_C\big|_{\mathcal{M}_C \setminus (\mathcal{M}_A \cup \mathcal{M}_B)} \cup (\mathcal{M}_A \times (\mathcal{M}_B \setminus \mathcal{M}_A)) \\
& \cup ((\mathcal{M}_A \cup \mathcal{M}_B) \times (\mathcal{M}_C \setminus (\mathcal{M}_A \cup \mathcal{M}_B))) = \\
& R_A \cup R_B\big|_{\mathcal{M}_B \setminus \mathcal{M}_A} \cup R_C\big|_{\mathcal{M}_C \setminus (\mathcal{M}_A \cup \mathcal{M}_B)} \cup ((\mathcal{M}_B \setminus \mathcal{M}_A) \times (\mathcal{M}_C \setminus (\mathcal{M}_A \cup \mathcal{M}_B))) \cup \\
& (\mathcal{M}_A \times ((\mathcal{M}_B \cup \mathcal{M}_C) \setminus \mathcal{M}_A)) = \\
& R_A \cup (R_B \cup R_C\big|_{\mathcal{M}_C \setminus \mathcal{M}_B} \cup (\mathcal{M}_B \times (\mathcal{M}_C \setminus \mathcal{M}_B)))\big|_{(\mathcal{M}_B \cup \mathcal{M}_C) \setminus \mathcal{M}_A} \\
& \cup (\mathcal{M}_A \times ((\mathcal{M}_B \cup \mathcal{M}_C) \setminus \mathcal{M}_A)) = R_{A \succeq (B \succeq C)}
\end{aligned}$$

Associativity of $\vee$ and the absorption rules can be shown similarly. □

However, a few important rules from classical logic do not hold. Obviously double negation, i.e. $\neg\neg A \equiv A$ is not true. Furthermore $\wedge$ is not associative. For example for $A = \neg b \wedge ((b \succeq a) \wedge (a \succeq b))$ we have $(\{a\}, \{a\}) \in R_A$ but for $B = (\neg b \wedge (b \succeq a)) \wedge (a \succeq b)$ we have $R_B = \emptyset$. For absorption, we have $a \vee (a \wedge b) \not\equiv a$, $a \wedge (b \succeq a) \not\equiv a$ and $(a \wedge b) \succeq a \not\equiv a$. Finally, the distribution law does not hold. For example $A = a \vee ((a \succeq b) \wedge (b \succeq a))$ implies $(b, b) \in R_A$ but $B = (a \vee (a \succeq b)) \wedge (a \vee (b \succeq a))$ implies $(b, b) \notin R_B$.[1]

4 Expressiveness

We want to show that every transitive relation on any set of models can be represented by our logic. As a helpful tool for this proof, we introduce two new connectives, the "strong or" and the "weak and" mentioned earlier. These are not necessary, as they can be expressed through the other connectives, but are useful short hands.

Definition 17 *For formulas A and B we write $A + B$ for $(A \vee (\neg A \wedge B)) \wedge (B \vee (\neg B \wedge A))$ and $A \sqcap B$ for $(A \vee B) \wedge \neg(A \wedge \neg B) \wedge \neg(\neg A \wedge B)$.*

Proposition 18 *$A + B$ is evaluated as $(\mathcal{M}_A \cup \mathcal{M}_B, \mathit{trcl}(R_A \cup R_B))$ and $A \sqcap B$ is evaluated as $(\mathcal{M}_A \cap \mathcal{M}_B, (R_A \cap R_B) \setminus (R^r_A \triangle R^r_B))$.*

Proof. This is shown by an easy computation. In fact, $\mathcal{M}_{A+B} = \mathcal{M}_A \cup \mathcal{M}_B$ and $\mathcal{M}_{A \sqcap B} = \mathcal{M}_A \cap \mathcal{M}_B$ follows from classical logic.
For $A + B$ we have

$$R_{A+B} = \mathit{trcl}((R_{A \vee (\neg A \wedge B)} \cup R_{B \vee (\neg B \wedge A)})\big|_{\mathcal{M}_A \cup \mathcal{M}_B}),$$

where

$$\begin{aligned} R_{A \vee (\neg A \wedge B)} = {} & (R_A \cap R_{\neg A \wedge B}) \setminus (R^r_A \triangle R^r_{\neg A \wedge B}) \cup R_A\big|_{\mathcal{M}_{A \vee (\neg A \wedge B)} \times \mathcal{M}_A \setminus \mathcal{M}_{\neg A \wedge B}} \cup \\ & R_{\neg A \wedge B}\big|_{\mathcal{M}_{A \vee (\neg A \wedge B)} \times \mathcal{M}_{\neg A \wedge B} \setminus \mathcal{M}_A} \end{aligned}$$

Observe that $\mathcal{M}_{A \vee (\neg A \wedge B)}$ is a superset of $\mathcal{M}_A$ and $\mathcal{M}_A \setminus \mathcal{M}_{\neg A \vee B}$ is $\mathcal{M}_A$. Hence $R_A\big|_{\mathcal{M}_{A \vee (\neg A \wedge B)} \times \mathcal{M}_A \setminus \mathcal{M}_{\neg A \wedge B}}$ is (a super set of) $R_A\big|_{\mathcal{M}_A}$, which is just R_A. Furthermore, every tuple in $R_{A \vee (\neg A \wedge B)}$ is either from R_A or R_B because $R_{\neg A} = \emptyset$. Therefore $R_A \subseteq R_{A \vee (\neg A \wedge B)} \subseteq R_A \cup R_B$ holds. By symmetry also, $R_B \subseteq R_{B \vee (\neg B \wedge A)} \subseteq R_A \cup R_B$ holds. Hence,

$$R_A \cup R_B \subseteq R_{A \vee (\neg A \wedge B)} \cup R_{B \vee (\neg B \wedge A)} \subseteq R_A \cup R_B$$

and therefore $R_{A+B} = \mathit{trcl}(R_A \cup R_B)$.

[1] It is also possible to construct a (more complicated) counter example if we assume that the relation is always reflexive

For $A \sqcap B$ we have

$$\begin{aligned} R_{A \sqcap B} &= trcl(R_{A \vee B} \cup R_{\neg(A \wedge \neg B)} \cup R_{\neg(\neg A \wedge B)})\big|_{\mathcal{M}_A \cap \mathcal{M}_B} \\ &= trcl(R_{A \vee B} \cup \emptyset \cup \emptyset)\big|_{\mathcal{M}_A \cap \mathcal{M}_B} \\ &= trcl((R_A \cap R_B) \setminus (R_A^r \triangle R_B^r) \cup R_A\big|_{\mathcal{M}_A \cup \mathcal{M}_B \times \mathcal{M}_A \setminus \mathcal{M}_B} \cup \\ &\quad R_B\big|_{\mathcal{M}_A \cup \mathcal{M}_B \times \mathcal{M}_B \setminus \mathcal{M}_A})\Big|_{\mathcal{M}_A \cap \mathcal{M}_B} = (R_A \cap R_B) \setminus (R_A^r \triangle R_B^r) \end{aligned}$$

This concludes the proof. □

We even prove that every relation can be represented with a special kind of formula, a formula without nested preferences:

Definition 19 *We say a formula A is* a formula without nested preferences *if for every subformula $B \succeq C$ of A, $\succeq$ does neither occur in B nor in C.*

Theorem 20 *Let $\geq$ be a transitive relation on $\mathcal{M} \subseteq \mathcal{P}(V)$, a set of models. Then, there exists a formula without nested preferences over the variables V that is evaluated to $(\mathcal{M}, \geq)$.*

Proof. Say $M_1 \geq M_2$ holds for $M_1, M_2 \in \mathcal{M}$. We already used formulas D_{M_1} and D_{M_2} which have only M_1 resp. M_2 as a model in the proof of theorem 14. Then $D_{M_1} \succeq D_{M_2}$ is evaluated as $(\{M_1, M_2\}, \{(M_1, M_2)\})$ and we say $D_{M_1} \succeq D_{M_2}$ expresses $M_1 \geq M_2$. Now let $D_1, \ldots, D_n$ for $n = |\geq|$ be formulas expressing all preferences in $\geq$. Then $D_1 + \cdots + D_n + D_{M_1} + \cdots + D_{M_m}$ for $m = |M|$ is evaluated as $(\mathcal{M}, \geq)$. □

5 Relation to QCL

We briefly review the main definition for qualitative choice logic (QCL) following [5]. The syntax of QCL is the same as the syntax of our logic defined in Definition 1 (note that in QCL the "preferential" connective is usually represented by symbol $\vec{\times}$ and is called ordered disjunction; we shall use $\succeq$ to allow for easier comparison).

Definition 21 *The* optionality *of a formula A is given as folllows*

- *$opt(A) = 1$ if A is an atom or of form $\neg B$;*
- *$opt(A) = max(opt(B), opt(C))$ if $A = B \circ C$, $\circ \in \{\wedge, \vee\}$;*
- *$opt(A) = opt(B) + opt(C)$ if $A = B \succeq C$.*

Definition 22 *Let A be a formula and $M \in \mathcal{M}_A$. The* degree i of satisfaction *($\models_i$) of A in M is defined as*

- *$M \models_1 A$ if A is an atom or of form $\neg B$;*
- *for $A = B \wedge C$: $M \models_k A$ iff $M \models_m B$, $M \models_n C$ and $k = max(m, n)$*

– *for* $A = B \vee C$*:* $M \models_k A$ *if*
  - $M \models_k B$ *and for no* $j < k$*:* $M \models_j C$*, or*
  - $M \models_k C$ *and for no* $j < k$*:* $M \models_j B$*;*
– *for* $A = B \succeq C$*:* $M \models_k A$ *if* $M \models_k B$ *or (*$M \models_1 \neg B$*,* $M \models_m C$*, and* $k = m + opt(A)$*).*

We can now define preferred models in QCL as follows.

Definition 23 *An interpretation* M *is a* preferred QCL model *of a formula* A *if* $M \in \mathcal{M}_A$*,* $M \models_k A$ *and there is no* N *such that* $N \models_m A$ *with* $m < k$*. We denote the set of QCL-preferred models of a formula* A *by* $pref_{QCL}(A)$*.*

As we will see below the preferred models of QCL and our logic coincide on certain fragments. They are not equivalent, in general. In fact, this is on purpose, since, as we show next, QCL behaves unintuitively in certain cases. This is due to the quite syntactic-driven notion of satisfaction degrees.

Example 24 *In QCL, there exist formulas* A *and* B *such that* $pref_{QCL}(A) \neq pref_{QCL}(A[B/B \succeq B])$*, i.e. in contrast to our logic, QCL does not account for neutrality w.r.t. the preferential connective. In fact, take* $A = ((a \succeq b) \vee (a \succeq c)) \wedge \neg a$ *and* A' *obtained from* A *by replacing the first occurrence of* a *in* A *by* $a \succeq a$*. First observe that* $\mathcal{M}_A = \mathcal{M}_{A'} = \{\{b\}, \{c\}, \{b, c\}\}$*. For each* $M \in \mathcal{M}_A$ *we have* $M \models_2 A$*, and thus each* $M \in \mathcal{M}_A$ *is a preferred QCL model of* A*. However, for* A' *we observe that* $\{b\} \models_3 A'$ *while* $\{b, c\} \models_2 A'$ *and* $\{c\} \models_2 A'$*. Thus only* $\{b, c\}$ *and* $\{c\}$ *are preferred QCL models of* A'*. Hence,* $pref_{QCL}(A) \neq pref_{QCL}(A')$*.*

In order to clarify the relation between our logic and QCL, let us define certain normal forms.

Definition 25 *We say a preference logic formula* A *is in*

– *Conjunctive Form if* A *is build of classical formulas,* $\neg, \wedge$ *and* $\succeq$*.*
– *Disjunctive Form if* A *is build of classical formulas,* $\neg, \vee$ *and* $\succeq$*.*
– *Normal Form if* A *is build of classical formulas and* $\succeq$*.*

Formulas in Normal Form are called basic choice formulas in QCL.

Theorem 26 *Let* M *and* N *be models of a formula* A *in Conjunctive Form, such that* $M \models_m A$ *and* $N \models_n A$ *Then,* $m \geq n$ *implies* $M \geq_A N$*.*

Proof. We prove the result by induction on the formula complexity. If A is a classical formula or $\neg B$ for some formula B the result is clear because every model has satisfaction degree 1.

Assume $A = B \wedge C$ for formulas B, C. If M has smaller satisfaction degree than N then the satisfaction degree of M is smaller than the satisfaction degree of N for either B or C, hence by induction $N \geq_B M$, resp. $N \geq_C M$. Therefore $N \geq_A M$.

Finally, assume $A = B \succeq C$ for formulas B and C. We distinguish three cases: (1) if M and N are both models of B the satisfaction degrees of M and N are the same for A and B, hence, by induction $M \geq_B N$ holds, therefore also $M \geq_A N$. (2) if M and N are both a model of C but not of B, the satisfaction degrees of M and N with respect to A only differs in a constant from the satisfaction degrees with respect to C, hence, by induction, $M \geq_A N$. (3) if M is a model of B and N is not a model of B, we have $M \geq_A N$ by definition. □

For formulas in Disjunctive Form, the relation between satisfaction degrees and the relational structure in our logic is the other way around.

Theorem 27 *Let M and N be models of a formula A in Disjunctive Form, such that $M \models_m A$ and $N \models_n A$ Then, $M >_A N$ implies $m > n$.*

Proof. We prove the result by induction on the formula complexity. If A is a classical formula or $\neg B$ for some formula B, the result is clear because $R_A = \emptyset$.

Assume $A = B \vee C$ for formulas B, C. There are three cases in which M is strictly preferred to N. First, M is strictly preferred to N for both B and C. Then the satisfaction degree of M is smaller than the satisfaction degree of N for both B or C by induction, hence also for A. Second, M and N are both models of B but N is not a model of C and M is preferred to N for B. Then, we know by induction that the satisfaction degree of M is smaller than the satisfaction degree of N for B and the satisfaction degree of N is infinity for C. The third case is symmetric to the second case.

Finally, assume $A = B \succeq C$ for formulas B and C. We distinguish three cases: First, if M and N are both models of B then $M >_A N$ if and only if $M >_B N$ but as the satisfaction degrees of M and N are the same for A and B we know by induction that the satisfaction degree of M is smaller than the one of N. If M and N are both a model of C but not of B, then $M >_A N$ iff $M >_C N$ but as the satisfaction degrees of M and N with respect to A only differs in a constant from the satisfaction degrees with respect to C, we can apply induction again. Finally, if M is a model of B and N is not a model of B, we know that the satisfaction degree of M is smaller than the one of N by definition. □

Corollary 28 *Let M and N be models of a formula A in Normal Form, such that $M \models_m A$ and $N \models_n A$ Then, $M \geq_A N$ iff $m \geq n$.*

The following example further illustrates the difference between QCL and our logic.

Example 29 *First we observe that formulas $(a \succeq b) \vee (b \succeq a)$ and $(a \succeq b) \wedge (b \succeq a)$ are equally evaluated in QCL and our logic. Both deliver all models, $\{a\}$,$\{b\}$ and $\{a, b\}$, as preferred models for both formulas, reflecting the fact that the preferences in the formula are cyclic.*

On the other hand, let $\phi = (a \wedge \neg b) \succeq (a \wedge b) \succeq (\neg a \wedge b)$ and $\psi = (\neg a \wedge b) \succeq (a \wedge b) \succeq (a \wedge \neg b)$. Then the QCL-preferred model of $\phi \wedge \psi$ is $\{a, b\}$, while $\phi \vee \psi$ delivers two QCL-preferred models $\{a\}$ and $\{b\}$. Our logic takes a *different view and combines the implictly obtained relations $\{a\} > \{a,b\} > \{b\}$ and $\{b\} > \{a,b\} > \{a\}$. In fact, we have as preferred models for $\phi \wedge \psi$, $\{a,b\}$, $\{a\}$ and $\{b\}$, reflecting that the relation becomes cyclic. For $\phi \vee \psi$, we obtain the same preferred models, however for a different reason, namely that no relation appears in both subformulas. while QCL again delivers $\{a\}$ and $\{b\}$.*

6 Complexity

The basic computational problems in our logic are model checking, satisfiability and testing if one model is preferred to another. By the definition of truth in our logic, the first two problems have the same complexity in our logic as in propositional logic, i.e. model checking is in P and satisfiability is NP-complete. The complexity of the model preference problem, on the other hand, is still open.

Problem 1 (Model preference) *Let $A \in \mathcal{PF}(V)$ and let M and N be models of A. Does $M \geq_A N$ hold?*

While we do not know the complexity of the model preference problem on arbitrary preference formulas, we know the complexity for the fragments introduced in section 5.

Proposition 30 *The model preference problem is in P for formulas in Disjunctive From.*

Proof. By definition, the relation between M and N does not depend on the relation between any other models for formulas in Disjunctive Form. Hence, we can easily track the relation between the models M and N in a bottom up manner through the syntax tree. □

Observe that every formula in Normal Form is also in Disjunctive Form, hence the model preference problem for formulas in Normal Form is also in P. The model preference problem for formulas in Conjunctive From, on the other hand, is harder, because we have to compute the transitive closure of the relation in the $\wedge$ step. Indeed, this problem is NP-complete. We need a lemma and an additional definition to prove this.

Definition 31 *Let $A \in \mathcal{PF}(V)$ and let (M,N) be a preference introduced by a subformula $D = B \succeq C$ or $D = B \wedge C$ (via transitivity). We say the preference survives until A if there is no subformula E of A containing D such that*

- $E = \neg F$
- $E = F \vee G$ *and* $(M,N) \notin R_E$
- $E = F \wedge G$ *and* $(M,N) \notin (R_F \cap R_G)\big|_{\mathcal{M}_E}$
- $E = F \succeq G$ *and* $(M,N) \notin R_F \cup R_G\big|_{\mathcal{M}_G \setminus \mathcal{M}_F}$

Lemma 32 *Let A be a formula in Conjunctive Form, such that some subformula $D = B \succeq C$ or $D = B \wedge C$ of A introduces preferences (M,N), (M',N) and (M',N'). Then, if (M,N) and (M',N') survive until A then also (M',N) survives until A.*

Proof. Let E be the smallest subformula of A such that (M', N) does not survive until E. Obviously, $E = \neg F$ is not possible. Now assume $E = F \wedge G$ and D is a subformula of F. Then (M', N) does not survive until E if and only if either M' or N is not a model of E. But the first case contradicts the assumption that (M', N') survives and the second case contradicts the assumption that (M, N) survives. Now assume $E = F \succeq G$ and D is a subformula of F. Then (M', N') survives until E if it survives until F which is does by choice of E. Finally, assume $E = G \succeq F$ and D is a subformula of F. Then (M', N) does not survive until E if and only if either M' or N is a model of G. However this leads to a contradiction as in the $\wedge$ case. □

Proposition 33 *The model preference problem is in* NP *for formulas in Conjunctive Form.*

Proof. Let $A \in \mathcal{PF}(V)$ and let M, N be models of A. We enumerate all $\succeq$ and $\wedge$ occurring in A as $\succeq_i$ and $\wedge_i$ and write $A_i \succeq_i B_i$ and $A_i \wedge_i B_i$ for the respective subformulas in A. Then, for every triple $(\succeq_i, \wedge_j, \succeq_k)$ we guess a sequence $\circ_1, N_1, \circ_2, N_2, \ldots, N_n, \circ_n$, for $n = |A|^4$, where for every $l \leq n$, N_l is a model and $\circ_l$ is either $\succeq_i$ for some i or a triple $(\succeq_i, \wedge_j, \succeq_k)$. Furthermore, $\circ_1$ is either $\succeq_i$ or a triple containing $\succeq_i$ as the first element. Analogously, $\circ_n$ is either $\succeq_k$ or a triple containing $\succeq_k$ is the last position.

Now, we label every triple either valid or unvalid in an order such that when we label a triple $(\succeq_i, \wedge_j, \succeq_k)$, we already labeled all triples containing $\wedge_m$ for all $\wedge_m$ contained in $A_j \wedge_j B_j$. First, we verify, that $\succeq_i, \succeq_k$ and all $\succeq_o$ and $\wedge_m$ occurring in the sequence guessed for the triple, are contained in $A_j \wedge_j B_j$. If not, we label the triple unvalid. Otherwise, we check for every pair of models (N_l, N_{l+1}) if it is valid for its operator $\circ_{l+1}$. This is done in the following way:

If $\circ_{l+1} = \succeq_o$, we check if $N_l \models A_o$. $N_{l+1} \not\models A_o$ and $N_{l+1} \models B_o$, i.e. if $N_l \geq_{A_o \succeq_o B_o} N_{l+1}$. If no, the pair is not valid. If yes, we check if the preference survives until $A_j \wedge_j B_j$ and $N_l \geq_{A_j \wedge_j B_j} N_{l+1}$ holds. Observe that this is a ptime check. If no, the pair is unvalid, if yes, the pair is valid.

So now assume $\circ_{l+1} = (\succeq_m, \wedge_n, \succeq_o)$. First, we check if the triple is valid. If no, the pair is unvalid. Otherwise, let $\circ_1^*, N_1^*, \circ_2^*, N_2^*, \ldots, N_{n^*}^*, \circ_{n^*}^*$ be the sequence guessed for $(\succeq_m, \wedge_n, \succeq_o)$. If $\circ_1^* = \succeq_p$, we check if $N_l \geq_p N_1^*$ was introduced by $A_p \succeq_p B_p$ and survives until $A_n \wedge_n B_n$. If $\circ_1^* = (\succeq_i^*, \wedge_j^*, \succeq_k^*)$, we check if the pair (N_l, N_1^*) is valid for that triple. Then, we do the same checks for $\circ_n^*$ and N_{l+1}. If all these checks are positive, the pair is valid and if all the pairs occurring in the sequence are valid, the triple $(\succeq_i, \wedge_j, \succeq_k)$ is valid.

Now to check if $M \geq_A N$, we do the following, we go through the syntax tree of A in a bottom up manner and check if the preference $M \geq_D N$ is introduced by any subformula, and if yes, if it survives until A. For subformulas $A_i \succeq_i B_i$, it is clear how to check, if $M \geq_{A_i \succeq_i B_i} N$ is introduced. For $A_j \wedge_j B_j$, if $M, N \models A_j \wedge_j B_j$ we look for every valid triple $(\succeq_i \wedge_j, \succeq_k)$ containing $\wedge_j$ we check if (M, N_1) is a valid pair for $\circ_1$ and if (N_n, N) is a valid pair for $\circ_n$. If both checks are positive for any valid triple, $M \geq_{A_j \wedge_j B_j} N$ is introduced.

The correctness of this algorithm relies on the following two facts: (1) If there is a valid sequence for a triple, then there is a valid sequence for that triple that contains each triple and each $\succeq_i$ at most once. (2) Let $B = A_i \wedge_i B_i$ be a subformula of A and let M, N be a models of B. Finally, let $\circ_1, N_1, \circ_2, N_2, \ldots, N_n, \circ_n$ be a valid sequence for the triple $(\succeq_j, \wedge_i, \succeq_k)$. If $M \circ_1 N_1$ is valid and there exists a second sequence $\circ_1^*, N_1^*, \circ_2^*, N_2^*, \ldots, N_{n^*}^*, \circ_{n^*}^*$ also valid for the triple $(\succeq_j, \wedge_i, \succeq_k)$ and $N \circ_1^* N_1^*$ is valid, then also $N \circ_1 N_1$ is valid.

The first result guarantees that it suffices to look at sequence of length less than $|A|^4$, the second result guarantees that it suffices to guess one sequence per triple. (1) holds, because if $N_i \circ_i N_{i+1} \circ_{i+1} \ldots N_j \circ_j N_{j+1}$ is valid and $\circ_i = \circ_j$ then $N_i \circ_i N_{j+1}$ is also valid by the definition of validity. (2) follows from Lemma 32. □

This proof does not work for arbitrary formulas, because Lemma 32 does not hold for arbitrary formulas. Assume, for example, that $A \succeq B$ introduces (M, N), (M', N) and (M', N'), that (N, M), $(N', M') \notin R_{A \succeq B}$ holds and that C is a formula such that $M >_C N$ and $M' >_C N'$ holds, but not $M' \geq_C N$.[2] Then (M, N) and (M', N') survive until $(A \succeq B) \vee C$, but (M', N) does not.

Proposition 34 *The model preference problem is* NP*-hard for formulas in Conjunctive Form.*

Proof. Let A be a propositional formula. Now let x be a new variable not occurring in A. Then let $B = ((A \wedge \neg x) \succeq x) \wedge (x \succeq (A \wedge \neg x))$. Obvious, B is a formula in Conjunctive Form. We claim that $\{x\} \geq_B \{x\}$ if and only if A is satisfiable. First assume A is satisfiable and M is a model of A. Then M is also a model of $A \wedge \neg x$. Therefore, M and $\{x\}$ are models of $C = (A \wedge \neg x) \succeq x$ and $M \geq_C x$ holds. Similarly, M and $\{x\}$ are models of $D = x \succeq (A \wedge \neg x)$ and $x \geq_D M$ holds. Hence $x \geq_B M \geq_B x$ and by transitivity $x \geq_B x$. Now assume A is not satisfiable. Then $A \wedge \neg x$ is also not satisfiable. Therefore, we have $\mathcal{M}_C = \{M \in 2^V \mid x \in M\}$ and $R_C = \emptyset$. Similarly, $\mathcal{M}_D = \mathcal{M}_C$ and $R_D = \emptyset$. Hence $R_B = \emptyset$. □

7 Conclusion

We presented a new logic for handling hard and soft constraints at once. The logic has a straightforward semantics that assigns every formula a set of models and a transitive relation on this set. We have shown that replacement pref-equivalence and semantic equivalence coincide for our logic and that many desirable equivalences known from propositional logic hold. Furthermore, we have shown that every transitive relation on every set of models can be expressed using a formula without nested preferences. We proved that our logic coincides with qualitative choice logic [5] on formulas in certain normal form and compares favorably with it on arbitrary formulas. Finally, we have shown that the model preference problem is in P for formulas in Disjunctive Form and NP-complete for formulas in Conjunctive From.

[2] Such a C exists by Theorem 20

In the future, we want to pinpoint the complexity of the model preference problem for arbitrary formulas as well as the complexity of some other, related, problems like finding a preferred model. Furthermore, we would like to compare our logic to other formalisms from the literature like Conjunctive Choice Logic [2] and nested circumscription [7]. We would like to consider other preference operators in our framework. Finally, we want to define an entailment relation and study its properties.

Acknowledgments

This work was funded by the Austrian Science Fund (FWF) under grant number Y698.

References

1. Bossert, W., Pattanaik, P.K., Xu, Y.: Choice under complete uncertainty: axiomatic characterizations of some decision rules. Economic Theory 16(2), 295–312 (2000)
2. Boudjelida, A., Benferhat, S.: Conjunctive choice logic. In: The International Symposium on Artificial Intelligence and Mathematics ISAIM (2016)
3. Boutilier, C., Brafman, R.I., Domshlak, C., Hoos, H.H., Poole, D.: CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. J. Artif. Intell. Res. 21, 135–191 (2004), https://doi.org/10.1613/jair.1234
4. Bouveret, S., Chevaleyre, Y., Maudet, N.: Fair allocation of indivisible goods. In: Brandt, F., Conitzer, V., Endriss, U., Lang, J., Procaccia, A.D. (eds.) Handbook of Computational Social Choice, pp. 284–310. Cambridge University Press (2016)
5. Brewka, G., Benferhat, S., Le Berre, D.: Qualitative choice logic. Artificial Intelligence 157(1-2), 203–237 (2004)
6. Brewka, G., Niemelä, I., Truszczynski, M.: Answer set optimization. In: IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence. pp. 867–872 (2003), http://ijcai.org/Proceedings/03/Papers/125.pdf
7. Cadoli, M., Eiter, T., Gottlob, G.: Complexity of propositional nested circumscription and nested abnormality theories. ACM Trans. Comput. Logic 6(2), 232–272 (Apr 2005), http://doi.acm.org/10.1145/1055686.1055688
8. Domshlak, C., Hüllermeier, E., Kaci, S., Prade, H.: Preferences in AI: An overview. Artificial Intelligence 175(7-8), 1037–1052 (2011)
9. Hadjali, A., Kaci, S., Prade, H.: Database preference queries—a possibilistic logic approach with symbolic priorities. Annals of Mathematics and Artificial Intelligence 63(3-4), 357–383 (2011)
10. Pigozzi, G., Tsoukias, A., Viappiani, P.: Preferences in artificial intelligence. Annals of Mathematics and Artificial Intelligence 77(3-4), 361–401 (2016)
11. Van Benthem, J., Girard, P., Roy, O.: Everything else being equal: A modal logic for ceteris paribus preferences. Journal of Philosophical Logic 38(1), 83–125 (2009)
12. Wright, G.H.: The logic of preference. Edinburgh University Press (1963)