Reasoning with Conditional Interpretations

Giovanni Casini*
CNR-ISTI
Pisa, Italy
giovanni.casini@isti.cnr.it

Umberto Straccia
CNR-ISTI
Pisa, Italy
umberto.straccia@isti.cnr.it

Abstract

We illustrate how key forms of reasoning can be captured within the framework of *Conditional Interpretations*, a semantics for conditional languages that supports the formalization of a wide range of reasoning systems. In particular, we outline how to model fundamental notions of entailment, beginning with relations that satisfy classical closure properties and extending to non-monotonic frameworks such as *Rational Closure*.

CCS Concepts

• **Computing methodologies → Nonmonotonic, default reasoning and belief revision**; • **Theory of computation → Automated reasoning**.

Keywords

Non-monotonic Logics, Conditional Logics

ACM Reference Format:
Giovanni Casini and Umberto Straccia. 2026. Reasoning with Conditional Interpretations. In *The 41st ACM/SIGAPP Symposium on Applied Computing (SAC '26), March 23–27, 2026, Thessaloniki, Greece.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3748522.3779724

1 Introduction

Conditionals - the "if-then" connections between propositions - form the backbone of reasoning and argumentation, and much of formal logic research has focused on analysing this type of relation. This line of work is particularly relevant to *Knowledge Representation and Reasoning* (KRR), as a formal analysis of reasoning is a necessary step toward its implementation in computational systems. The properties a conditional connection satisfies depend on the type of reasoning being modelled. It is well-known that, while classical material implication is suitable for mathematical reasoning, other domains - such as presumptive, normative, causal, probabilistic, fuzzy, or counterfactual reasoning - require different argumentation patterns, and modelling such alternative conditional structures has been a main research topic in KRR.

A common approach for formalizing such reasoning patterns is through *structural properties* [24] (see later on) and semantic characterisations based on possible-worlds approaches. A potential limit of the latter approach is that some classical structural properties, such as *right conjunction* (And) or *right weakening* (RW), despite being undesirable in certain contexts, appear to be tightly linked to possible-world semantics.

This paper focuses on *conditional reasoning*, introduced in [11], that is a flexible semantic framework allowing to model various types of conditionals, applicable to different reasoning domains. Here, the characterisation of reasoning patterns defined by structural properties is not done referring to possible-worlds, but through the adoption of a more general, operational kind of semantics. The key advantage of this approach is its flexibility: (i) for all finite theories of conditionals, for all combination of reasoning patters, a unique characteristic model exists [11][1]; and (ii) while we show how classical non-monotonic entailment relations (e.g., Rational Closure) can be modelled within this framework, it also supports alternatives that are not in line with possible-world frameworks - such as reasoning patterns excluding (And) or (RW) - and aligns more naturally with other paradigms, like inheritance networks.

Contribution: The work in [11] presents the semantic framework and the characterisation of various structural properties through representation theorems. We extend here such a framework by defining a variety of entailment relations, including closure under Horn properties, closure under non-Horn properties, and, notably, a closure that is equivalent to Rational Closure [23]. We present also (brute force) algorithms that operate directly on the characteristic model of a conditional base under a specified set of reasoning rules, and show how entailment of a conditional $A \Rightarrow B$ can be decided by verifying whether a specific condition holds in that model.

The paper is organised as follows: Section 2 recaps the formal framework of conditional interpretations; Section 3 introduces basic forms of entailment based on the closure under Horn properties; Section 4 deals with the closure under non-Horn properties, with a focus on *Rational Monotonicity* [23]; and Section 5 summarises our contribution, addresses related work, and outlines some future research directions.

2 Preliminaries

Syntax and Semantics. Our language contains conditionals of the form $A \Rightarrow B$, but does not allow nesting of conditionals or combining them via propositional operators (A and B are Boolean propositional formulae,). This is in line with popular conditional formalisms, such as the so-called KLM approach [21] and I/O logics [25]. Specifically, let $\mathcal{L}$ be a finitely generated propositional language, with logical connectives $\neg, \vee, \wedge, \rightarrow$ and $\leftrightarrow$, and the propositional symbols $\top, \bot$ having the usual meanings. Capital letters $A, B, \ldots$ denote propositions, while $\mathcal{A}, \mathcal{B}, \ldots$ refer to sets of propositions. The classical propositional consequence relation is denoted by $\models$. The conditional language $\mathcal{L}_\Rightarrow$ built on top of $\mathcal{L}$, is: $\mathcal{L}_\Rightarrow =_{\text{def}} \{A \Rightarrow B \mid A, B \in \mathcal{L}\}$. In the semantics we use a relation $\leq \subseteq$

*Also with CAIR, University of Cape Town, South Africa.

[1]The resulting logic is *paraconsistent*.

ACM ISBN 979-8-4007-2294-3/2026/03
https://doi.org/10.1145/3748522.3779724

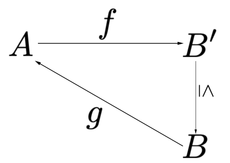

Figure 1: Graphical representation of $\mathcal{I} \Vdash A \Rightarrow B$.

$\mathcal{L} \times \mathcal{L}$ among propositional formulae: Here $A \leq B$ is interpreted as $\models A \rightarrow B$. $\leq$ generates the classical lattice semantics for propositional formulae, with $\vee$ and $\wedge$ represented by the *join* and *meet* operations, respectively. The relations $<$ and $\equiv$ are defined in the usual way from $\leq$: $A < B$ represents $\models A \rightarrow B$ and $\not\models B \rightarrow A$, while $A \equiv B$ represents $\models A \leftrightarrow B$. Clearly, $\leq$ is reflexive and transitive. $\min_{\leq}(\mathcal{A})$ indicates the $\leq$-*minimal* elements in $\mathcal{A}$, that is, $\min_{\leq}(\mathcal{A}) =_{\text{def}} \{B \in \mathcal{A} \mid \nexists C \in \mathcal{A} \text{ s.t. } C < B\}$. Also, let us define $\mathcal{A}^{\uparrow} =_{\text{def}} \{B \mid A \leq B, \text{ for some } A \in \mathcal{A}\}$.

We use a well-known order among sets of formulae, based on $\leq$: namely, the *Smyth* order $\preceq$ over power sets (see, e.g. [30, Section 3]). Formally,

$$\mathcal{A} \preceq \mathcal{B} \text{ iff } \forall B \in \mathcal{B}\ \exists A \in \mathcal{A} \text{ s.t. } A \leq B\ .$$

We also write $\mathcal{A} \cong \mathcal{B}$ iff $\mathcal{A} \preceq \mathcal{B}$ and $\mathcal{B} \preceq \mathcal{A}$.

A *choice* function $h\colon \mathcal{L} \longrightarrow 2^{\mathcal{L}}$ maps a formula to a set of formulae. We say that h is *Smyth-monotone*, or simply *S-monotone*, iff for every $A, B \in \mathcal{L}$, if $A \leq B$ then $h(A) \preceq h(B)$. Furthermore, $A \in \mathcal{L}$ is a *fixed-point* of h iff $A \in h(A)$ (see e.g. [30]).

Eventually, h is $\star$-*closed*, where $\star \in \{\leq, \equiv\}$, iff for all $A, B, C \in \mathcal{L}$, if $A \in h(C)$ and $B \star A$ then $B \in h(C)$; while h is $\star$-*closed*, where $\star \in \{\wedge, \vee\}$, iff for all $A, B, C \in \mathcal{L}$, if $A \in h(C)$ and $B \in h(C)$ then $A \star B \in h(C)$.

The semantics of conditionals relies on the concept of a *conditional interpretation* $\mathcal{I} = (f, g)$, where $f\colon \mathcal{L} \longrightarrow 2^{\mathcal{L}}$ and $g\colon \mathcal{L} \longrightarrow 2^{\mathcal{L}}$ are choice functions: f represents the *relevant effects* of a proposition, and g the *possible conditions* for a proposition to hold. A conditional interpretation $\mathcal{I} = (f, g)$ *satisfies* a conditional $A \Rightarrow B$, denoted $\mathcal{I} \Vdash A \Rightarrow B$, iff

(1) there is $B' \in \mathcal{L}$ s.t. $B' \in f(A)$ and $B' \leq B$; and
(2) $A \in g(B)$.

$A \Rightarrow B$ is *satisfiable* (has a *model*) if there is a conditional interpretation $\mathcal{I}$ such that $\mathcal{I} \Vdash A \Rightarrow B$. A set of conditionals is *satisfiable* iff there is a conditional interpretation satisfying each conditional in it [11, Def. 1].

As shown in Fig. 1, $\mathcal{I} \Vdash A \Rightarrow B$ iff there is a "triangle" $A \xrightarrow{f} B' \leq B \xrightarrow{g} A$. We indicate with $A \triangle B$ that there is a triangle $A \xrightarrow{f} B' \leq B \xrightarrow{g} A$ passing through some $B' \leq B$. This definition implies that an agent accepts a conditional connection between A and B if B is a logical consequence of some *relevant effect* B' of A ($B' \in f(A)$), and A is recognised as a *relevant condition* for B ($A \in g(B)$).

For an interpretation $\mathcal{I}$, $S_{\mathcal{I}}$ indicates the set of conditionals satisfied by $\mathcal{I}$, i.e. $S_{\mathcal{I}} =_{\text{def}} \{A \Rightarrow B \mid \mathcal{I} \Vdash A \Rightarrow B\}$.

We recall from [11] that any set S of conditionals is always satisfiable. Specifically, S is satisfied by its *characteristic model*.

Definition 1 ([11]). *Let S be a set of conditionals. Its* characteristic model *is the conditional interpretation $\mathcal{I}_S = (f_S, g_S)$, where, for every $A \in \mathcal{L}$:*

(1) $\mathcal{A}_A =_{def} \{B \mid B \Rightarrow A \in S\}$ and $C_A =_{def} \{B \mid A \Rightarrow B \in S\}$;
(2) $f_S(A) = \min_{\leq}(C_A)$ and $g_S(A) = \mathcal{A}_A$.

$\mathcal{I}_S$ characterises S, that is, $\mathcal{I}_S$ satisfies *exactly* the set of conditionals in S (i.e. $S_{\mathcal{I}} = S$) [11, Proposition 1]. Furthermore, since any set of conditionals has a characteristic model, the class of conditional interpretations does not impose any closure under any structural property [11, Corollary 2].

Structural Properties. The satisfaction of specific structural properties is achieved by identifying appropriate subclasses of the class of the conditional interpretations, which is done by imposing appropriate constraints on the functions f and g. For example, consider the property of *Monotonicity*.

$$\frac{A \Rightarrow C, \quad \models B \rightarrow A}{B \Rightarrow C} \qquad \text{(Mon)}$$

The satisfaction of the property (Mon) can be obtained by imposing on f and g of any conditional interpretation $\mathcal{I} = (f, g)$ the following constraint **(Mon$_{\mathcal{I}}$)**:

(1) f is S-Monotone;
(2) g is $\leq$-closed.

Proposition 1 ([11]). *A set of conditionals S is closed under (Mon) iff it can be characterised by a conditional model $\mathcal{I} = (f, g)$ that satisfies* **(Mon$_{\mathcal{I}}$)**.

The results presented in [11] are essentially *representational*, like Proposition 1, which demonstrates the suitability of conditional interpretations for modeling various forms of closure. Representation results have been provided for the following set $\mathcal{P}$ of properties:

(LLE)	$\frac{A \Rightarrow C, \quad A \equiv B}{B \Rightarrow C}$	(RLE)	$\frac{A \Rightarrow B, \quad B \equiv C}{A \Rightarrow C}$
(Ref)	$A \Rightarrow A$	(Cut)	$\frac{A \wedge B \Rightarrow C, \quad A \Rightarrow B}{A \Rightarrow C}$
(Mon)	$\frac{A \Rightarrow C, \quad B \leq A}{B \Rightarrow C}$	(And)	$\frac{A \Rightarrow B, \quad A \Rightarrow C}{A \Rightarrow (B \wedge C)}$
(Or)	$\frac{A \Rightarrow C, \quad B \Rightarrow C}{(A \vee B) \Rightarrow C}$	(RW)	$\frac{A \Rightarrow B, \quad B \leq C}{A \Rightarrow C}$
(ExFalso)	$\frac{A \leq \bot}{A \Rightarrow B}$	(CM)	$\frac{A \Rightarrow B, \quad A \Rightarrow C}{(A \wedge B) \Rightarrow C}$

In [11], for each of the properties (P) mentioned above, a corresponding semantic property (P$_{\mathcal{I}}$) has been defined, along with a representation theorem stating that a set of conditionals S is closed under (P) if and only if it can be characterized by a conditional model $\mathcal{I} = (f, g)$ satisfying (P$_{\mathcal{I}}$). For clarity, the specific semantic property corresponding to each structural property will be recalled when need. For more details about their meaning and desirability we refer to [11] and references therein. The following result summarizes the relationship between structural properties and conditional interpretations:

Proposition 2 ([11], Proposition 14). *Let $\mathcal{X} \subseteq \mathcal{P}$ be a set of structural properties in $\mathcal{P}$, and $\mathcal{X}_{\mathcal{I}}$ be the set of the correspondent semantic properties. If a set S of conditionals is closed under the properties in $\mathcal{X}$, then there is a conditional interpretation characterising S and satisfying all the properties in $\mathcal{X}_{\mathcal{I}}$.*

3 Basic Forms of Entailment

We begin with basic entailment relations. A *conditional Knowledge Base* (KB) $\mathcal{K}$ is a finite set of conditionals. The structural properties in $\mathcal{P}$, defined in the previous section, are referred to as *Horn-properties*, because they have the form of a Horn clause: a set of positive premises with a single positive conclusion. The following proposition is straightforward.

Proposition 3. *Let (P) be any Horn-property and S_1 and S_2 be any two sets of conditionals closed under (P). Then $S_1 \cap S_2$ is also closed under (P).*

This intersection property is important for modeling entailment relations. Given a finite set of conditionals S and consider the set $\mathfrak{S}$ of all its possible supersets closed under a set of Horn-properties. By Proposition 3, the intersection $\bigcap \mathfrak{S}$ of all such supersets is itself closed under the same Horn-properties, that is $\bigcap \mathfrak{S} \in \mathfrak{S}$. Thus, we derive the following corollary from Propositions 3.

Corollary 1. *Let $\mathcal{X} \subseteq \mathcal{P}$ be a finite set of Horn-properties and $\mathcal{K}$ a conditional KB. Then there is an unique smallest set $\mathcal{K}'$ s.t. $\mathcal{K} \subseteq \mathcal{K}'$ and $\mathcal{K}'$ is closed under $\mathcal{X}$.*

This property ensures that any form of entailment characterised as closure under Horn-properties is easily definable, as a unique smallest closure is guaranteed to exist. This aligns with the classical Tarskian approach to logical consequence [31], which is defined by closure under three Horn-properties (*Reflexivity*, *Monotonicity*, and *Cut*), and semantically on the intersection of all the formulae satisfied by all the classical models of the premises.

With our representation results established, the next step is defining a reasoning system. Starting from a conditional KB $\mathcal{K}$, we derive new conditionals based on the reasoning patterns (i.e. the structural properties) we want to satisfy. The basic idea is as follows:

(1) we build the characteristic model $\mathcal{I}_{\mathcal{K}} = (f_{\mathcal{K}}, g_{\mathcal{K}})$ of $\mathcal{K}$;
(2) according to the kind of closure we want to enforce, we compute the smallest change in $f_{\mathcal{K}}$ and in $g_{\mathcal{K}}$ so to guarantee such a closure.

Once such a model has been built, the entailment of a conditional $A \Rightarrow B$ amounts to test whether the triangle $A \triangle B$ holds in that model. Now, regarding point 1., $\mathcal{I}_{\mathcal{K}} = (f_{\mathcal{K}}, g_{\mathcal{K}})$ is defined as in Definition 1. Concerning point 2., to 'close' under a Horn-property, we modify f and g accordingly for each structural property.

A notable case involves Left- and Right-Logical Equivalence (LLE and RLE), which are fundamental properties that most logic systems are expected to satisfy. One could argue that a system failing to satisfy them cannot truly be considered a logic system. In this work, we assume systems are closed under (LLE) and (RLE). Consequently, w.l.o.g. we use A as a representative for all B logically equivalent to A. Therefore, if $\mathcal{L}$ is generated by n propositional letters then $|\mathcal{L}| = 2^{2^n}$ and, thus, $\mathcal{L}$ may be assumed finite.
Concerning point 1., Algorithm 1 shows how to build the characteristic model $\mathcal{I}_{\mathcal{K}}$ of $\mathcal{K}$ and its correctness is immediate from a comparison with Definition 1. Concerning point 2., we consider each closure property, one by one.

Algorithm 1: `CharacteristicModel(`$\mathcal{K}$`)`

Input: A conditional KB $\mathcal{K}$
Output: The characteristic model $\mathcal{I}_{\mathcal{K}} = (f_{\mathcal{K}}, g_{\mathcal{K}})$
$\mathcal{A}_{\mathcal{K}} = \{A \mid A \Rightarrow B \in \mathcal{K}\}$
$C_{\mathcal{K}} = \{B \mid A \Rightarrow B \in \mathcal{K}\}$
forall $A \in \mathcal{L}$ **do**
  $f_{\mathcal{K}}(A) = \emptyset$
  $g_{\mathcal{K}}(A) = \emptyset$
forall $A \in \mathcal{L}$ **do**
  if $A \in \mathcal{A}_{\mathcal{K}}$ **then**
    $C_A \leftarrow \{B \mid A \Rightarrow B \in \mathcal{K}\}$
    $f_{\mathcal{K}}(A) = \min_{\leq}(C_A)$
  if $A \in C_{\mathcal{K}}$ **then**
    $\mathcal{A}_A \leftarrow \{B \mid B \Rightarrow A \in \mathcal{K}\}$
    $g_{\mathcal{K}}(A) = \mathcal{A}_A$
return $\mathcal{I}_{\mathcal{K}} = (f_{\mathcal{K}}, g_{\mathcal{K}})$

Reflexivity (Ref). Also (Ref) is a fundamental property, stating that any proposition assumed is also included in the conclusions. While generally desirable, there are notable exceptions. For instance, if the conditional $A \Rightarrow B$ has a *deontic* interpretation, such as "If A, then it ought to be B", reflexivity can lead to undesirable outcomes, such as "If there has been a crime, then there ought to be a crime". Ensuring closure under (Ref) is easy: for every formula A, it suffices to require that A itself is a fixed point of both f and g. That is [11],

(Ref$_{\mathcal{I}}$) for all A:
(1) A is a fixed-point of both f and g.

The closure under (Ref$_{\mathcal{I}}$) can be enforced by Algorithm 2. For Algorithm 2, as with the others in this section, we must demonstrate that it enforces the minimal change required in the set of satisfied conditionals to satisfy the involved Horn-property ((Ref$_{\mathcal{I}}$), in this case). To achieve this, we first formalise the concept of minimal change.

Algorithm 2: `Ref(`$\mathcal{I}$`)`

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (Ref)
foreach $A \in \mathcal{L}$ **do**
  $f(A) = f(A) \cup \{A\}$
  $g(A) = g(A) \cup \{A\}$
return $\mathcal{I} = (f, g)$

Definition 2. *Given two conditional interpretations $\mathcal{I}$ and $\mathcal{I}'$, $\mathcal{I}'$ is an* extension *of $\mathcal{I}$ (written $\mathcal{I} \leq_{ext} \mathcal{I}'$) iff $S_{\mathcal{I}} \subseteq S_{\mathcal{I}'}$. Given a finite set $\mathcal{X} \subseteq \mathcal{P}$ of Horn-properties, $\mathcal{I}'$ is a* minimal $\mathcal{X}$-extension *of $\mathcal{I}$ iff*

- *$\mathcal{I} \leq_{ext} \mathcal{I}'$;*
- *$\mathcal{I}'$ is closed under all the properties in $\mathcal{X}$;*
- *there is no interpretation $\mathcal{I}''$ s.t. $\mathcal{I} \leq_{ext} \mathcal{I}'' <_{ext} \mathcal{I}'$ and $\mathcal{I}''$ is closed under all the properties in $\mathcal{X}$.*

Note that minimality in change refers to the sets of satisfied conditionals, not necessarily to the change applied to the functions f and g. Corollary 1 ensures that for any conditional KB $\mathcal{K}$ and any finite set $\mathcal{X}$ of Horn-properties, there is an unique smallest extension of $\mathcal{K}$ that is closed under $\mathcal{X}$. However, more than one conditional interpretation may satisfy exactly such an extension.

Algorithms 3-9 correspond each to one of the remaining structural properties from Section 2: (Cut), (Mon), (And), (Or), (RW), (Ex-Falso), and (CM), which we address next.

Cut (Cut). This well-known property of classical logic is generally considered desirable. Algorithm 3 returns a minimal (Cut)-extension, by enforcing the property ($\text{Cut}_\mathcal{I}$) defined as [11]:

($\text{Cut}_\mathcal{I}$) for all A, B, C:
(1) If $A \triangle B$, then $f(A) \preceq f(A \wedge B)$;
(2) If $A \in g(B)$ and $A \wedge B \in g(C)$, then $A \in g(C)$.

Algorithm 3: $\texttt{Cut}(\mathcal{I})$

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (Cut)

$f' \leftarrow \emptyset$
$g' \leftarrow \emptyset$
while $g' \neq g$ *or* $f' \neq f$ **do**
  $g' \leftarrow g$
  $f' \leftarrow f$
  foreach $A \in \mathcal{L}$ **do**
    foreach $B \in \mathcal{L}$ **do**
      if $A \in g(B)$ **then**
        foreach $C \in \mathcal{L}$ **do**
          if $A \wedge B \in g(C)$ **then**
            $g(C) \leftarrow g(C) \cup \{A\}$
  foreach $A \in \mathcal{L}$ **do**
    foreach $B \in \mathcal{L}$ **do**
      if $A \triangle B \in \mathcal{I}$ **then**
        $f(A) \leftarrow f(A) \cup f(A \wedge B)$
return $\mathcal{I} = (f, g)$

Monotonicity (Mon). Another well-known property of classical logic, (Mon) models the certainty of conclusions given the premises. It often needs to be relaxed or reformulated in domains involving incomplete or defeasible information. Algorithm 4 returns a minimal (Mon)-extension, enforcing the following property [11]:

($\text{Mon}_\mathcal{I}$)
(1) f is S-Monotone;
(2) g is $\leq$-closed.

Right Conjunction (And). This is a standard property in possible-worlds semantics, but it may be undesirable in certain frameworks, such as probabilistic entailment or some deontic settings. For example, in some cases, multiple preferable but mutually exclusive courses of action may exist. Algorithm 5 returns a minimal (And)-extension, by enforcing the characterising condition ($\text{And}_\mathcal{I}$) defined as [11]:

($\text{And}_\mathcal{I}$) for all A, B:
(1) if $B, C \in \min_\leq(f(A))$, then $B \equiv C$;
(2) $g(A) \cap g(B) \subseteq g(B \wedge C)$.

Left Disjunction (OR).. This models the classical principle of 'reasoning by cases'. Algorithm 6 enforces the closure under ($\text{Or}_\mathcal{I}$) and returns a minimal (Or)-extension, where [11]

($\text{Or}_\mathcal{I}$) for all A, B:
(1) $\min_\leq(f(A)^\uparrow \cap f(B)^\uparrow) \subseteq f(A \vee B)$;

Algorithm 4: $\texttt{Mon}(\mathcal{I})$

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (Mon)

foreach $A \in \mathcal{L}$ **do**
  foreach $B \in \mathcal{L}$ **do**
    if $A \leq B$ **then**
      $f(A) \leftarrow f(A) \cup f(B)$
foreach $A \in \mathcal{L}$ **do**
  foreach $B \in \mathcal{L}$ **do**
    if $A \leq B$ **then**
      foreach $C \in \mathcal{L}$ **do**
        if $B \in g(C)$ **then**
          $g(C) \leftarrow g(C) \cup \{A\}$
return $\mathcal{I} = (f, g)$

(2) g is $\vee$-closed.

Right Weakening (RW). This rule states that from $A \Rightarrow B$ we can conclude $A \Rightarrow C$ for any C s.t. $B \leq C$, and semantically it is characterised by ($\text{RW}_\mathcal{I}$), which is defined as [11]:

($\text{RW}_\mathcal{I}$) for all A, B:
(1) if $A \leq B$ then $g(A) \subseteq g(B)$.

It is a property almost unavoidable in possible-worlds semantics, and generally desirable. However, in some contexts, such as deontic or causal settings, it may be counterintuitive. For example, while we might believe that in the presence of a crime we should call the police, it would make little sense to think that we should either call or not call the police. Algorithm 7 returns a minimal (RW)-extension.

Ex Falso Quodlibet (ExFalso). This classical principle asserts that from absurdity, anything follows. It is characterised by the following semantic property [11]:

($\text{EFQ}_\mathcal{I}$) for all A: if $A \equiv \bot$, then
(1) $\bot \in f(A)$;
(2) $A \in g(B)$, for all B.

While a principle of classical logic, it is rejected in some approaches, notably paraconsistent logic. Algorithm 8 returns a minimal (ExFalso)-extension.

Cautious Monotonicity (CM). This constrained form of monotonicity is generally desirable in most non-monotonic reasoning applications [24]. Algorithm 9 returns a minimal (CM)-extension, by enforcing the closure under ($\text{CM}_\mathcal{I}$), defined as [11]:

($\text{CM}_\mathcal{I}$) for all A, B,
(1) if $A \triangle B$, then $f(A \wedge B) \preceq f(A)$;
(2) if $A \in g(B) \cap g(C)$ then $A \wedge B \in g(C)$.

For each of the above closure properties it is possible to prove that, given any conditional KB $\mathcal{K}$, the correspondent algorithm returns a model of the minimal closure of $\mathcal{K}$ under the associated property.

Proposition 4. *Let* $(P) \in \{$(Ref), (Cut), (Mon), (And), (Or), (RW), (ExFalso), (CM)$\}$. *Given a conditional interpretation* $\mathcal{I}_\mathcal{K}$*, that is the characteristic model of a KB* $\mathcal{K}$*, Algorithm* $\mathsf{P}(\mathcal{I}_\mathcal{K})$ *returns a minimal* (P)*-extension of* $\mathcal{I}_\mathcal{K}$*.*

This section concludes by addressing the combination of closures under multiple Horn properties. Each algorithm takes as input the

Algorithm 5: And($\mathcal{I}$)

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (And)

foreach $A \in \mathcal{L}$ **do**
  $f(A) \leftarrow \bigwedge f(A)$
$g' \leftarrow \emptyset$
while $g' \neq g$ **do**
  $g' \leftarrow g$
  foreach $A \in \mathcal{L}$ **do**
    foreach $B \in \mathcal{L}$ **do**
      $g(A \wedge B) \leftarrow g(A \wedge B) \cup (g(A) \cap g(B))$
return $\mathcal{I} = (f, g)$

Algorithm 6: Or($\mathcal{I}$)

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (Or)

foreach $A \in \mathcal{L}$ **do**
  foreach $\mathcal{A} \in \mathscr{P}(g(A))$ **do**
    $g(A) \leftarrow g(A) \cup \{\bigvee \mathcal{A}\}$
  foreach $B \in g(A)$ **do**
    $f(B) \leftarrow f(B) \cup \{A\}$
return $\mathcal{I} = (f, g)$

Algorithm 7: RW($\mathcal{I}$)

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (RW)

foreach $A \in \mathcal{L}$ **do**
  foreach $B \in \mathcal{L}$ **do**
    if $B \leq A$ **then**
      $g(A) \leftarrow g(A) \cup g(B)$
return $\mathcal{I} = (f, g)$

Algorithm 8: ExFalso($\mathcal{I}$)

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (ExFalso)

$f(\bot) \leftarrow f(\bot) \cup \{\bot\}$
foreach $A \in \mathcal{L}$ **do**
  $g(A) \leftarrow g(A) \cup \{\bot\}$
return $\mathcal{I} = (f, g)$

Algorithm 9: CM($\mathcal{I}$)

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: A minimal extension of $\mathcal{I} = (f, g)$ closed under (CM)

$f' \leftarrow \emptyset$
$g' \leftarrow \emptyset$
while $g' \neq g$ *or* $f' \neq f$ **do**
  $g' \leftarrow g$
  $f' \leftarrow f$
  foreach $A \in \mathcal{L}$ **do**
    foreach $B \in \mathcal{L}$ **do**
      foreach $C \in \mathcal{L}$ **do**
        if $A \in g(B) \cap g(C)$ **then**
          $g(B) \leftarrow g(B) \cup \{A \wedge C\}$
          $g(C) \leftarrow g(C) \cup \{A \wedge B\}$
  foreach $A \in \mathcal{L}$ **do**
    foreach $B \in \mathcal{L}$ **do**
      if $A \triangle B \in \mathcal{I}$ **then**
        $f(A \wedge B) \leftarrow f(A \wedge B) \cup f(A)$
return $\mathcal{I} = (f, g)$

Algorithm 10: Reduction($\mathcal{I}$)

Input: A conditional interpretation $\mathcal{I} = (f, g)$
Output: The characteristic model of $S_{\mathcal{I}}$

foreach $A \in \mathcal{L}$ **do**
  $f(A) \leftarrow \min_{\leq}(f(A))$
  foreach $B \in f(A)$ **do**
    if $A \triangle B \notin \mathcal{I}$ **then**
      $f(A) \leftarrow f(A) \setminus \{B\}$
  foreach $B \in g(A)$ **do**
    if $B \triangle A \notin \mathcal{I}$ **then**
      $g(A) \leftarrow g(A) \setminus \{B\}$
return $\mathcal{I} = (f, g)$

Algorithm 11: PClosure($\mathcal{K}, \mathcal{X}$)

Input: A conditional KB $\mathcal{K}$ and a set of Horn-Properties $\mathcal{X} \subseteq \mathcal{P}$
Output: The minimal extension $\mathcal{I}$ of $\mathcal{I}_{\mathcal{K}}$ closed under $\mathcal{X}$

1: $\mathcal{I}_{\mathcal{K}} = (f_{\mathcal{K}}, g_{\mathcal{K}}) \leftarrow$ CharacteristicModel($\mathcal{K}$)
2: $\mathcal{I} = (f, g) \leftarrow \mathcal{I}_{\mathcal{K}}$
3: $f' \leftarrow \emptyset$
4: $g' \leftarrow \emptyset$
5: **while** $g' \neq g$ *or* $f' \neq f$ **do**
6:   $g' \leftarrow g$
7:   $f' \leftarrow f$
8:   **foreach** $P \in \mathcal{X}$ **do**
9:     $\mathcal{I} = (f, g) \leftarrow P_{alg}(\mathcal{I})$
10:     $\mathcal{I} = (f, g) \leftarrow$ Reduction($\mathcal{I}$)
11: **return** $\mathcal{I} = (f, g)$

characteristic model of a KB $\mathcal{K}$ and Proposition 4 guarantees that its output is a conditional interpretation representing the minimal closure S of $\mathcal{K}$ under a given Horn-property. However, this does not guarantee that it is actually the *characteristic model* $\mathcal{I}_S$ of S, as per Definition 1. Nevertheless, Algorithm 10 reduces any conditional interpretation to another one that satisfies the same conditionals, but additionally satisfies Definition 1 (Proposition 5), which is then used in Algorithm 11 to prove Proposition 6. Algorithm 11 has as input a conditional KB $\mathcal{K}$ and an arbitrary set of Horn-properties $\mathcal{X} \subseteq \mathcal{P}$, and returns a model of $\mathcal{K}$ that is the minimal extension of $\mathcal{I}_{\mathcal{K}}$ w.r.t. the properties in $\mathcal{X}$.[2]

Proposition 5. *Let $\mathcal{I}$ be any conditional interpretation. Algorithm 10 returns a conditional interpretation $\mathcal{I}'$ s.t. $S_{\mathcal{I}'} = S_{\mathcal{I}}$ and $\mathcal{I}'$ is the characteristic model of $S_{\mathcal{I}}$ as per Definition 1.*

Proposition 6. *Let $\mathcal{K}$ be a conditional KB and $\mathcal{X} \subseteq \mathcal{P}$ a set of Horn-properties. Then Algorithm 11 terminates and returns the characteristic model of the smallest set S of conditionals s.t. $\mathcal{K} \subseteq S$ and S is closed under $\mathcal{X}$.*

We can now define the notion of entailment.

Definition 3. *Let $\mathcal{K}$ be a conditional KB, and $\mathcal{X} \subseteq \mathcal{P}$ a set of Horn-properties. Let $\mathfrak{M}_{\mathcal{K}}^{\mathcal{X}}$ be the set of all the conditional models of $\mathcal{K}$ that are closed under the properties $\mathcal{X}$. Then a conditional $A \Rightarrow B$ is entailed by $\mathcal{K}$ w.r.t. $\mathcal{X}$, denoted $\mathcal{K} \vDash_{\mathcal{X}} A \Rightarrow B$, is defined as follows:*

$$\mathcal{K} \vDash_{\mathcal{X}} A \Rightarrow B \text{ iff } \mathcal{I} \Vdash A \Rightarrow B \text{ for all } \mathcal{I} \in \mathfrak{M}_{\mathcal{K}}^{\mathcal{X}}.$$

[2] Please note that at line 9 of the algorithm, P_{alg} refers to the algorithm corresponding to the Horn-property P that is taken under consideration: for instance, if P is (And), then $P_{alg}(\mathcal{I})$ is And($\mathcal{I}$).

The following is an immediate consequence of Proposition 6 and Corollary 1.

Corollary 2. *Let $\mathcal{K}$ be a conditional KB, $\mathcal{X} \subseteq \mathcal{P}$ a set of Horn-properties, $A \Rightarrow B$ a conditional, and $\mathcal{I}^*$ the output of* `PClosure`$(\mathcal{K},\mathcal{X})$. *Then*

$$\mathcal{K} \vDash_{\mathcal{X}} A \Rightarrow B \text{ iff } \mathcal{I}^* \Vdash A \Rightarrow B\,.$$

By Corollary 2, we may decide $\mathcal{K} \vDash_{\mathcal{X}} A \Rightarrow B$ by checking whether $A \triangle B$ holds in the model returned by `PClosure`$(\mathcal{K},\mathcal{X})$.

4 Rational Monotonicity

So far we have focused on structural properties in Horn form. However, there are also significant structural properties that cannot be represented in Horn form. In order to show how to manage non-Horn properties in the present semantics, we consider one of the most notable ones, *Rational Monotonicity* (RM) [23].

Rational Monotonicity:

$$\frac{A \not\Rightarrow \neg B, \quad A \Rightarrow C}{(A \wedge B) \Rightarrow C} \qquad \text{(RM)}$$

(RM) is a form or constrained monotonicity, stronger than (CM), that is regarded as a key property for reasoning systems modeling *presumptive reasoning* [22]. From a semantics perspective, (RM) and other non-Horn properties have not been addressed in [11]. Here, we demonstrate that the class of conditional interpretations satisfying (RM) can be characterised by imposing the following semantic constraint:

(RM$_\mathcal{I}$) Let $\mathcal{I} = (f, g)$ be a conditional interpretation. For any formulae A, B, C, if $A \triangle \neg B \notin \mathcal{I}$, then
 (1) $f(A \wedge B) \preceq f(A)$;
 (2) if $A \in g(C)$ then $A \wedge B \in g(C)$.

The condition (RM$_\mathcal{I}$) gives a semantic characterisation of the closure under (RM).

Proposition 7. *A set of conditionals S is closed under (RM) iff $S = S_\mathcal{I}$ for some $\mathcal{I} = (f, g)$ that satisfies (RM$_\mathcal{I}$).*

Next, we address entailment. A main challenge with non-Horn properties is that that their closure is not preserved under intersection, as it can be easily shown for (RM).

Example 1. *Let $\mathcal{L}$ be the propositional language generated from the propositional letters $\{a, b\}$. The two sets of conditionals $S_1 = \{a \Rightarrow b, (a \wedge b) \Rightarrow b\}$ and $S_2 = \{a \Rightarrow b, a \Rightarrow \neg b\}$ are both closed under (RM), but their intersection $S_1 \cap S_2 = \{a \Rightarrow b\}$ is not.*

Proposition 8. *Closure under (RM) is not preserved under the intersection of sets of conditionals.*

Therefore, the Tarskian characterisation of entailment in Definition 3 cannot be directly applied. Several entailment relations satisfy (RM), usually based on the choice of a specific model (e.g. [10, 22]). This includes the well-known notion of *Rational Closure* (RC) [23], that we take under consideration here. There are multiple semantic characterisation of RC in a possible-worlds framework [2–4, 16, 23, 27], as well as various decision procedures [8, 9, 14, 23], all modelling the same entailment relation.

In what follows, we assume that the final set of conditionals will be closed under all the properties in $\mathcal{P} \setminus \{(\text{Mon})\}$. Since (RLE), (Cut), and (ExFalso) can be derived from the others properties, they are redundant and, thus, we consider the set $\mathcal{P}^* = \{$(LLE), (Ref), (And), (Or), (RW), (CM)$\}$.

The core idea of (RM), particularly in its application to RC, is to reason classically (that is, we apply monotonicity) unless we recognise the situation as atypical. If we believe that usually $A \Rightarrow C$ holds (e.g., typically, horses are fast), and we are not aware that $A \Rightarrow \neg B$ (e.g., we are not aware that a typical horse is not black), that is, B is compatible with typical cases of A, then we must conclude, reasoning monotonically, that typically the situations in which both A and B are true imply C (that is, typically black horses are fast). However, if we are aware that some extra-premise B is not compatible with the typical cases of A (e.g., we believe that typical horses are not crippled), then we are not forced to apply monotonicity reasoning about the most typical situations in which A and B are both true (that is, we are not forced to derive that crippled horses are fast). This constrained application of monotonicity is evident when comparing (RM$_\mathcal{I}$) with (Mon$_\mathcal{I}$): (RM$_\mathcal{I}$) represents a constrained version of (Mon$_\mathcal{I}$), where the S-Monotonicity of f and the $\leq$-closure of g are applied only until a potential conflict arises.

We are going now to present the algorithms to model an entailment relation equivalent to RC within our framework. Conceptually, the procedure, is as follows:

- identify conflicts and, thus, exceptional conditionals;
- assign a rank to the conditionals;
- build a conditional interpretation that is closed under RM; and
- decide entailment of a conditional via triangle verification with that model.

Algorithm 12 identifies potential conflicts: that is, to identify among the conditionals $A \Rightarrow B$ in $\mathcal{K}$ all the potential conflicts, i.e. the cases in which we would be forced to conclude $A \Rightarrow \bot$ if we were reasoning monotonically. Formally, $A \Rightarrow B \in \mathcal{K}$ is *exceptional* w.r.t. $\mathcal{K}$ iff $\mathcal{K} \vDash_{\mathcal{P}^*} \top \Rightarrow \neg A$ (A is not satisfied in all the typical situations).

Algorithm 12: `Exceptional`$(\mathcal{K})$

Input: A conditional KB $\mathcal{K}$
Output: $\mathcal{E} \subseteq \mathcal{K}$ s.t. $\mathcal{E}$ is the set of exceptional conditionals w.r.t. $\mathcal{K}$ and $\mathcal{P}^*$
$\mathcal{I}_\mathcal{K} \leftarrow$ `CharacteristicModel`$(\mathcal{K})$
$\mathcal{I} \leftarrow$ `PClosure`$(\mathcal{K}, \mathcal{P}^*)$
$\mathcal{E} \leftarrow \emptyset$
foreach $A \Rightarrow B \in \mathcal{K}$ **do**
  if $\top \triangle \neg A \in \mathcal{I}$ **then**
    $\mathcal{E} \leftarrow \mathcal{E} \cup \{A \Rightarrow B\}$
return $\mathcal{E}$

Using Algorithm 13, which builds on Algorithm 12, we rank conditionals in $\mathcal{K}$ into different levels of exceptionality.

This ranking enables Algorithm 14 to assign a level of exceptionality to any formula relative to the overall ranking. Finally, Algorithm 15 uses the ranking to construct the characteristic model of the closure of $\mathcal{K}$ under $\mathcal{P}^*$ and (RM) by modifying f and g in such a way to enforce Monotonicity wherever possible, while avoiding potential conflicts. Now, we can prove that $\mathcal{I}^r_\mathcal{K}$, the output of `RClosure`$(\mathcal{K},\mathcal{X})$, satisfies (RM).

Algorithm 13: ComputeLayers($\mathcal{K}$)

Input: A conditional KB $\mathcal{K}$
Output: An exceptionality ranking $r_{\mathcal{K}}$ of the conditionals and an exceptionality ranking $\mathfrak{A}_{\mathcal{K}}$ of the antecedents

$i \leftarrow 0$
$\mathcal{E}_0 \leftarrow \mathcal{K}$
$\mathcal{E}_1 \leftarrow$ Exceptional($\mathcal{E}_0$)
$\mathcal{A}_0 = \{A \mid A \Rightarrow B \in \mathcal{E}_0 \setminus \mathcal{E}_1\}$
while $\mathcal{E}_{i+1} \neq \mathcal{E}_i$ **do**
    $i \leftarrow i + 1$
    $\mathcal{E}_{i+1} \leftarrow$ Exceptional($\mathcal{E}_i$)
    $\mathcal{A}_i = \{A \mid A \Rightarrow B \in \mathcal{E}_i \setminus \mathcal{E}_{i+1}\}$
$\mathcal{E}_\infty \leftarrow \mathcal{E}_i$
$\mathcal{A}_\infty \leftarrow \{A \mid A \Rightarrow B \in \mathcal{E}_\infty\}$
$r_{\mathcal{K}} \leftarrow (\mathcal{E}_0, \ldots, \mathcal{E}_{i-1}, \mathcal{E}_\infty)$
$\mathfrak{A}_{\mathcal{K}} \leftarrow (\mathcal{A}_0, \ldots, \mathcal{A}_{i-1}, \mathcal{A}_\infty)$
return $\langle r_{\mathcal{K}}, \mathfrak{A}_{\mathcal{K}} \rangle$

Algorithm 14: ComputeHeight($A, \mathcal{K}$)

Input: A formula A, a conditional KB $\mathcal{K}$
Output: The height of the formula A

$\langle r_{\mathcal{K}}, \mathfrak{A}_{\mathcal{K}} \rangle \leftarrow$ ComputeLayers($\mathcal{K}$)
$n \leftarrow (|r_{\mathcal{K}}| - 1)$
$i \leftarrow 0$
$h_A \leftarrow 0$
while $i \leq n$ *or* $h' = h_A$ **do**
    $i \leftarrow i + 1$
    $h' \leftarrow h_A$
    $\mathcal{I} \leftarrow$ PClosure($\mathcal{E}_i, \mathcal{P}^*$)
    if $\top \triangle \neg A \in \mathcal{I}$ **then**
        $h_A \leftarrow h_A + 1$
if $h_A = n + 1$ **then**
    $\mathcal{I} \leftarrow$ PClosure($\mathcal{E}_\infty, \mathcal{P}^*$)
    if $\top \triangle \neg A \in \mathcal{I}$ **then**
        $h_A \leftarrow \infty$
return h_A

Algorithm 15: RClosure($\mathcal{K}$)

Input: A conditional KB $\mathcal{K}$
Output: The conditional interpretation $\mathcal{I}^r_{\mathcal{K}} = (f, c)$, characterising the RC of $\mathcal{K}$

1: $\langle r_{\mathcal{K}}, \mathfrak{A}_{\mathcal{K}} \rangle \leftarrow$ ComputeLayers($\mathcal{K}$)
2: $n \leftarrow (|r_{\mathcal{K}}| - 1)$
3: $\mathcal{I}^r_{\mathcal{K}} = (f, g) \leftarrow$ PClosure($\mathcal{K}, \mathcal{P}^*$)
4: **foreach** $A \in \mathcal{L}$ **do**
5:     $h_A \leftarrow$ ComputeHeight($A, \mathcal{K}$)
6:     $j \leftarrow h_A$
7:     **foreach** $B \in \mathcal{A}_j$ ($\mathcal{A}_j \in \mathfrak{A}_{\mathcal{K}}$) **do**
8:         **if** $A \leq B$ **then**
9:             $f(A) \leftarrow f(A) \cup f(B)$
10:             **foreach** $D \in \mathcal{L}$ **do**
11:                 **if** $B \in g(D)$ **then**
12:                     $g(D) \leftarrow g(D) \cup \{A\}$
13: **foreach** $P \in \mathcal{P}^*$ **do**
14:     $\mathcal{I}^r_{\mathcal{K}} \leftarrow \mathsf{P}_{\mathtt{alg}}(\mathcal{I}^r_{\mathcal{K}})$
15: $\mathcal{I}^r_{\mathcal{K}} \leftarrow$ Reduction($\mathcal{I}^r_{\mathcal{K}}$)
16: **return** $\mathcal{I}^r_{\mathcal{K}}$

Proposition 9. *Given any conditional base $\mathcal{K}$,* RClosure($\mathcal{K}$) *returns a model $\mathcal{I}^r_{\mathcal{K}}$ that corresponds to the RC of $\mathcal{K}$.*

A straightforward consequence of Proposition 9 is that the model $\mathcal{I}^r_{\mathcal{K}}$ satisfies (RC), since RC is closed under such a property [23].

Corollary 3. *Given any conditional base $\mathcal{K}$,* RClosure($\mathcal{K}$) *returns a model $\mathcal{I}^r_{\mathcal{K}}$ closed under (RM).*

Referring to the model $\mathcal{I}^r_{\mathcal{K}}$, we define an entailment relation that adapts RC to our semantic framework.

Definition 4. *Let $\mathcal{K}$ be a conditional KB. A conditional $A \Rightarrow B$ is* RC-entailed *by $\mathcal{K}$, denoted $\mathcal{K} \vDash_{\mathsf{rc}} A \Rightarrow B$, iff $\mathcal{I}^r_{\mathcal{K}} \Vdash A \Rightarrow B$, where $\mathcal{I}^r_{\mathcal{K}}$ is the model returned by* RClosure($\mathcal{K}$).

Computational Complexity. We shortly address the computational complexity of our algorithms. It is not difficult to see that the computational complexity of our algorithms depends critically on $|\mathcal{L}|$. Specifically, the following can be shown.

Proposition 10. *For all algorithms 1-15 and entailment checks the worst case running time is polynomial w.r.t. $|\mathcal{L}|$.*

We conclude with an illustrative example.

Example 2. *Let $\mathcal{K} = \{\mathsf{b} \Rightarrow \mathsf{f}, (\mathsf{b} \wedge \mathsf{p}) \Rightarrow \neg\mathsf{f}\}$. It is the usual penguin example:* b *is* bird, p *is* penguin, f *is* flying*; also,* s *is* sparrow*. Let's check whether $(\mathsf{p} \wedge \mathsf{b})\triangle\mathsf{f} \in \mathcal{I}^r_{\mathcal{K}}$. In the characteristic model of $\mathcal{K}$, $\mathcal{I}_{\mathcal{K}} = (f_{\mathcal{K}}, g_{\mathcal{K}})$, $(\mathsf{p} \wedge \mathsf{b}) \notin g_{\mathcal{K}}(\mathsf{f})$, otherwise, by Definition 1, we would have had $(\mathsf{p} \wedge \mathsf{b}) \Rightarrow \mathsf{f} \in \mathcal{K}$. Since $\mathsf{p} \wedge \mathsf{b}$ appears as an antecedent in $\mathcal{K}$, we know its level of exceptionality from* ComputeLayers($\mathcal{K}$)*: $(\mathsf{p} \wedge \mathsf{b})\triangle\neg\mathsf{f}$ implies $\mathsf{b}\triangle\mathsf{p} \rightarrow \neg\mathsf{f}$[3]; By (And) we have $\mathsf{b}\triangle\mathsf{f} \wedge (\mathsf{p} \rightarrow \neg\mathsf{f})$ and, by (RW), $\mathsf{b}\triangle\neg\mathsf{p}$, that implies $\top\triangle\mathsf{b} \rightarrow \neg\mathsf{p}$, that is, by (RLE), $\top\triangle\neg(\mathsf{b} \wedge \mathsf{p})$. In fact, from* ComputeLayers($\mathcal{K}$) *we obtain $\mathcal{E}_1 = \{(\mathsf{b} \wedge \mathsf{p}) \Rightarrow \neg\mathsf{f}\}$ and $\mathcal{E}_2 = \emptyset$. Hence $(\mathsf{p} \wedge \mathsf{b}) \in \mathcal{A}_1$ and (easy to check) being in $\mathcal{A}_1$ implies that $h_{(\mathsf{p}\wedge\mathsf{b})} = 1$. Since the only element of $\mathcal{A}_1$ is $(\mathsf{p} \wedge \mathsf{b})$ itself, there are no changes in the f and in the g involving $(\mathsf{p} \wedge \mathsf{b})$, and consequently $(\mathsf{p} \wedge \mathsf{b})\triangle\mathsf{f} \notin \mathcal{I}^r_{\mathcal{K}}$. That is, we cannot derive that penguins fly. Now we can check whether $(\mathsf{s} \wedge \mathsf{b})\triangle\mathsf{f} \in \mathcal{I}^r_{\mathcal{K}}$. Since $(\mathsf{s}\wedge\mathsf{b}) \Rightarrow \mathsf{f} \notin \mathcal{K}$, $(\mathsf{s}\wedge\mathsf{b}) \notin g_{\mathcal{K}}(\mathsf{f})$. From* Computeheight($(\mathsf{s}\wedge\mathsf{b}), \mathcal{K}$) *we obtain $h_{\mathsf{s}\wedge\mathsf{b}} = 0$, since we cannot obtain $\top \Rightarrow \neg(\mathsf{s} \wedge \mathsf{b})$ from the closure of $\mathcal{K}$ under $\mathcal{P}^*$. Let's check how f and g are modified w.r.t. $(\mathsf{s} \wedge \mathsf{b})$ by Algorithm 15. We have that $\mathsf{b} \in \mathcal{A}_0$ and $(\mathsf{s} \wedge \mathsf{b}) \leq \mathsf{b}$. Since $\mathsf{f} \in f_{\mathcal{K}}(\mathsf{b})$, by line 9 in Algorithm 15 we end up with $\mathsf{f} \in f(\mathsf{s} \wedge \mathsf{b})$ in the final model $\mathcal{I}^r_{\mathcal{K}}$. Analogously, $\mathsf{b} \in g_{\mathcal{K}}(\mathsf{f})$, and, by line 12 in Algorithm 15, in the final model $\mathcal{I}^r_{\mathcal{K}}$ we have $(\mathsf{s} \wedge \mathsf{b}) \in g(\mathsf{f})$. Hence we can conclude that $(\mathsf{s} \wedge \mathsf{b})\triangle\mathsf{f} \in \mathcal{I}^r_{\mathcal{K}}$: since we have no potential conflicts, sparrows inherit the property of flying from typical birds.*

5 Conclusions

Summary. We have demonstrated how key forms of reasoning can be captured within the framework of conditional interpretations [11], an approach that departs significantly from the traditional possible-worlds semantics commonly used in the literature. Instead, reasoning is modelled via the manipulation of choice functions f and g, offering a level of flexibility not typically afforded by possible-worlds frameworks. Notably, this semantics supports the formalization of a wide range of reasoning rules, with the distinctive property that, regardless of the chosen rule set, a characteristic model of its closure always exists.

In particular, we have shown how to model different forms of entailment, beginning with classical monotonic entailment relations and extending to non-Horn properties such as Rational Monotonicity (RM), and have proposed algorithms that operate directly on the characteristic model of a conditional base under a given set of

[3]The closure under $\mathcal{P}^*$ validates the derivation of $A \Rightarrow B \rightarrow C$ from $A \wedge B \Rightarrow C$ [21, Lemma 5.2].

reasoning rules. This enables entailment of a conditional $A \Rightarrow B$ to be determined by simply verifying whether $A \triangle B$ holds in the model - a development that underscores the flexibility of our approach. We believe this framework has the potential to support novel characterizations of entailment relations, including the integration of pattern-based constraints, as found in multiple inheritance networks [10, 18, 29].

Closely related work. There have been a few attempts to formalise non-classical forms of conditional reasoning that do not satisfy properties like (RW), e.g. [5, 7, 28]. The property (RW) is inbuilt in other possible-worlds semantics, such as the very general framework based on *Plausibility Measures* [15]. Our approach diverges from traditional methods by abandoning possible worlds and modeling reasoning through the manipulation of choice functions f and g, which we believe offers greater flexibility. If we consider forms of reasoning that are closed at least under (LLE), (RLE), and (Anti-RW) (a constrained for of (RW)), it is possible to revert to the possible-worlds framework as in [7], though the relationship between that framework and ours remains unexplored. Additionally, in [26] a deontic system, based on I/O logics [25], which initially satisfies (only) (RLE), is presented.

Future work. First of all, we would like to look at various other forms of non-monotonic closure operations, such as *Relevant Closure* [6], *Lexicographic Closure* [22], *Inheritance-based* ones [10, 18, 29], *c-inference* [1, 19], and *System W* [17, 20]. We want to investigate how these frameworks can be addressed in the context of conditional interpretations. Also, we intend to investigate the closure under constrained forms of the classical closure properties (e.g., (AntiRW) w.r.t. (RW) [7]), in the same perspective used for the development of systems satisfying constrained forms of monotonicity. Another topic for future work involves the development of scalable reasoning algorithms. The (brute-force) reasoning algorithm presented here are intended primarily as a theoretical tool. Nonetheless, given the operational nature of our semantics, we are looking for an implementation for some suitable logics. Achieving this will require refining the algorithms, both in terms of their formulation (e.g. removal of redundant computations –as they occur right now) and through the use of efficient data structures. For example, one could represent a characteristic model as an annotated graph with f, g, and $\leq$ edges, enabling direct graph-based construction of the characteristic model. In the light of Proposition 10 another direction of future research is to consider those cases in which both $|\mathcal{L}|$ and testing $\leq$ may be polynomially bounded by the size of a KB. In fact, we believe that our approach holds particular promise, potentially also from a computational standpoint, when applied to non-classical reasoning in low-complexity knowledge compilation based methods, such as graph-based semantic frameworks (RDFS or knowledge graphs [12]) or the ontology language OWL 2 EL [13].

Acknowledgments

This research was supported by the H2020 STARWARS Project (GA No. 101086252), action HORIZON TMA MSCA Staff Exchanges.

References

[1] Christoph Beierle, C. Eichhorn, G. Kern-Isberner, and S. Kutsch. 2018. Properties of skeptical c-inference for conditional knowledge bases and its realization as a constraint satisfaction problem. *Ann. of Math. and Arti. Intell.* 83 (2018), 247–275.

[2] R. Booth and J. B. Paris. 1998. A Note on the Rational Closure of Knowledge Bases with Both Positive and Negative Knowledge. *Journal of Logic, Language and Information* 7, 2 (1998), 165–190.

[3] K. Britz, G. Casini, T. Meyer, Kody Moodley, U. Sattler, and I. Varzinczak. 2020. Principles of KLM-Style Defeasible Description Logics. *ACM Trans. Comput. Logic* 22, 1 (nov 2020), 46 pages.

[4] K. Britz, G. Casini, T. Meyer, and I. Varzinczak. 2019. *A KLM Perspective on Defeasible Reasoning for Description Logics.* Springer, 147–173.

[5] K. Britz, J. Heidema, and I. Varzinczak. 2011. Constrained Consequence. *Logica Universalis* 5, 2 (2011), 327–350.

[6] G. Casini, T. Meyer, K. Moodley, and R. Nortje. 2014. Relevant Closure: A New Form of Defeasible Reasoning for Description Logics. In *Proceedings of JELIA 2014 (LNCS, Vol. 8761)*, Springer, 92–106.

[7] G. Casini, T. Meyer, and I. Varzinczak. 2019. Simple Conditionals with Constrained Right Weakening. In *Proceedings of IJCAI 2019.* 1632–1638.

[8] G. Casini, T. Meyer, and I. Varzinczak. 2023. Situated Conditional Reasoning. *Artif. Intell.* 319, C (2023), 23 pages.

[9] G. Casini and U. Straccia. 2010. Rational Closure for Defeasible Description Logics. In *Proceedings of JELIA 2010 (LNCS, 6341)*, Springer, 77–90.

[10] G. Casini and U. Straccia. 2013. Defeasible Inheritance-Based Description Logics. *J. Artif. Intell. Res. (JAIR)* 48 (2013), 415–473.

[11] G. Casini and U. Straccia. 2022. A General Framework for Modelling Conditional Reasoning - Preliminary Report. In *Proceedings of KR 2022*,

[12] G. Casini and U. Straccia. 2023. Defeasible RDFS via rational closure. *Information Sciences* 643 (2023), 118409.

[13] G. Casini, U. Straccia, and T. Meyer. 2019. A Polynomial Time Subsumption Algorithm for Nominal Safe $\mathcal{ELO}_\bot$ under Rational Closure. *Information Sciences* 501 (2019), 588–620.

[14] M. Freund. 1998. Preferential Reasoning in the perspective of Poole default logic. *Artif. Intell.* 98 (1998), 209–235.

[15] N. Friedman and J.Y. Halpern. 2001. Plausibility Measures and Default Reasoning. *J. ACM* 48, 4 (2001), 648–685.

[16] L. Giordano, V. Gliozzi, N. Olivetti, and G. L. Pozzato. 2015. Semantic characterization of rational closure: From propositional logic to Description Logics. *Artif. Intell. J.* 226 (2015), 1–33.

[17] J. Haldimann and C. Beierle. 2022. Characterizing Multipreference Closure with System W. In *Proceedings of SUM 2022.* Springer, 79–91.

[18] J. F. Horty. 1994. Some direct theories of nonmonotonic inheritance. In *Handbook of logic in artificial intelligence and logic programming: nonmonotonic reasoning and uncertain reasoning.* Vol. 3. Oxford University Press, Inc., 111–187.

[19] G. Kern-Isberner. 2001. *Conditionals in Nonmonotonic Reasoning and Belief Revision.* Springer.

[20] C. Komo and C. Beierle. 2022. Nonmonotonic reasoning from conditional knowledge bases with System W. *Ann. of Math. and Art. Intell.* 90, 1 (2022). 107 – 144.

[21] S. Kraus, D. Lehmann, and M. Magidor. 1990. Nonmonotonic reasoning, preferential models and cumulative logics. *Artif. Intell.* 44 (1990), 167–207.

[22] D. Lehmann. 1995. Another perspective on default reasoning. *Ann. of Math. and Art. Intelligence* 15, 1 (1995), 61–82.

[23] d. Lehmann and M. Magidor. 1992. What does a conditional knowledge base entail? *Artif. Intell.* 55 (1992), 1–60.

[24] D. Makinson. 1994. General Patterns in Nonmonotonic Reasoning. In *Handbook of Logic in Artificial Intelligence and Logic Programming, Vol. III.* Oxford, 35–110.

[25] D. Makinson and L. van der Torre. 2000. Input/output logics. *Journal of Philosophical Logic* 29, 4 (2000), 383–408.

[26] X. Parent and L. van der Torre. 2014. "Sing and Dance!". In *Deontic Logic and Normative Systems*, Springer, 149–165.

[27] J. Pearl. 1990. System Z: a natural ordering of defaults with tractable applications to nonmonotonic reasoning. In *Proceedings of TARK 1990* .

[28] H. Rott. 2019. Difference-Making Conditionals and the Relevant Ramsey Test. *The Review of Symbolic Logic* (2019), 1–32.

[29] E. Sandewall. 2010. Defeasible inheritance with doubt index and its axiomatic characterization. *Artificial Intelligence* 18, 174 (2010), 1431–1459.

[30] U. Straccia, M. Ojeda-Aciego, and C. V. Damásio. 2009. On Fixed-points of Multivalued Functions on Complete Lattices and their Application to Generalized Logic Programs. *SIAM J. Comput.* 8, 5 (2009), 1881–1911.

[31] A. Tarski. 1956. On the concept of logical consequence. In *On some fundamental concepts of metamathematics. Logic, Semantics, Metamathematics. Papers from 1923 to 1938*, J. H. Woodger (Ed.). Clarendon Press.