A classical-logic view of a paraconsistent logic

Middelburg, C.A.

DOI
10.48550/arXiv.2008.07292
Publication date
2020
Document Version
Submitted manuscript
License
Unspecified

Citation for published version (APA):
Middelburg, C. A. (2020). *A classical-logic view of a paraconsistent logic*. (v1 ed.) ArXiv.
https://doi.org/10.48550/arXiv.2008.07292

A Classical-Logic View of a Paraconsistent Logic

C.A. Middelburg

Informatics Institute, Faculty of Science, University of Amsterdam,
Science Park 904, 1098 XH Amsterdam, the Netherlands
`C.A.Middelburg@uva.nl`

Abstract. This paper is concerned with the first-order paraconsistent logic $\mathrm{LPQ}^{\supset,\mathsf{F}}$. A sequent-style natural deduction proof system for this logic is given and, for this proof system, both a model-theoretic justification and a logical justification by means of an embedding into first-order classical logic is presented. For no logic that is essentially the same as $\mathrm{LPQ}^{\supset,\mathsf{F}}$, a natural deduction proof system is currently available in the literature. The presented embedding provides both a classical-logic explanation of this logic and a logical justification of its proof system.

Keywords: paraconsistent logic, classical logic, natural deduction, embedding, three-valued logic.

Mathematics Subject Classification (2010): 03B53, 03B10, 03B50.

1 Introduction

A set of formulas is contradictory if there exists a formula such that both that formula and the negation of that formula can be deduced from it. In classical logic, every formula can be deduced from every contradictory set of formulas. A paraconsistent logic is a logic in which not every formula can be deduced from every contradictory set of formulas.

In [10], Priest proposed the paraconsistent propositional logic LP (Logic of Paradox) and its first-order extension LPQ. The paraconsistent logic considered in this paper, called $\mathrm{LPQ}^{\supset,\mathsf{F}}$, is LPQ enriched with a falsity constant and an implication connective for which the standard deduction theorem holds. A sequent-style natural deduction proof system for $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is presented. In addition to the usual model-theoretic justification of the proof system, a logical justification by means of an embedding into FOCL (First-Order Classical Logic) is given. Classical logic is used meta-logically here: the embedding provides a classical-logic explanation of $\mathrm{LPQ}^{\supset,\mathsf{F}}$.

$\mathrm{LPQ}^{\supset,\mathsf{F}}$ is essentially the same as CLuNs [1], $\mathrm{LFI1}^*$ [4], $\mathrm{QLFI1}_\circ$ [5], $\mathrm{J}_3^*{=}$ [6], and LP° [9]. The proof systems for these logics available in the literature are Hilbert systems for the first four logics and a Gentzen-style sequent system for the last one. To fill this gap, a natural deduction proof system is given for $\mathrm{LPQ}^{\supset,\mathsf{F}}$ in this paper. An important reason to present a justification of this proof system by means of an embedding into classical logic is to draw attention to the viewpoint that, although it may be convenient to use a paraconsistent logic like

$\mathrm{LPQ}^{\supset,\mathsf{F}}$ if contradictory formulas have to be dealt with, classical logic is the ultima ratio of formal reasoning.

The only difference between CLuNs and $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is that the former has a bi-implication connective and the latter does not have that connective. However, bi-implication is definable in $\mathrm{LPQ}^{\supset,\mathsf{F}}$. $\mathrm{LFI1}^*$, $\mathrm{QLFI1}_\circ$, $\mathrm{J}_3^*{=}$, and LP° do not have the falsity constant of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ and $\mathrm{J}_3^*{=}$ and LP° also do not have the implication connective of $\mathrm{LPQ}^{\supset,\mathsf{F}}$. Instead, each of $\mathrm{LFI1}^*$, $\mathrm{QLFI1}_\circ$, $\mathrm{J}_3^*{=}$, and LP° has a connective that is foreign to classical logic. However, the constants and connectives of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ are definable in terms of those of each of these logics and vice versa. That is why it is said that $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is essentially the same as these logics. I prefer $\mathrm{LPQ}^{\supset,\mathsf{F}}$ because it does not have a connective that is foreign to classical logic.

The structure of this paper is as follows. First, the language of the paraconsistent logic $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is defined (Section 2). Next, a sequent-style natural deduction proof system for $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is given (Section 3). After that, a model-theoretic justification of this proof system is given (Section 4). Then, a justification of this proof system by means of an embedding into FOCL is given (Section 5). Following this, selected points related to the preceding sections are discussed (Section 6). Finally, some concluding remarks are made (Section 7).

2 The Language of $\mathrm{LPQ}^{\supset,\mathsf{F}}$

In this section the language of the paraconsistent logic $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is described. First, the assumptions which are made about function and predicate symbols are given and the notion of a signature is introduced. Next, the terms and formulas of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ are defined for a fixed but arbitrary signature. Thereafter, notational conventions and abbreviations are presented and some remarks about free variables and substitution are made. In coming sections, the proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ and the interpretation of the terms and formulas of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ are defined for a fixed but arbitrary signature.

2.1 Signatures

It is assumed that the following has been given: (a) a countably infinite set $\mathcal{V}$ of *variable symbols*, (b) a countably infinite set $\mathcal{C}$ of *constant symbols*, (c) for each $n \in \mathbb{N}_1$, a countably infinite set $\mathcal{F}_n$ of *function symbols* of *arity* n, and, (d) for each $n \in \mathbb{N}_1$, a countably infinite set $\mathcal{P}_n$ of *predicate symbols* of *arity* n. It is also assumed that all these sets and $\{=\}$ are mutually disjoint.

We write $\mathcal{S}ym$ for $\mathcal{V} \cup \mathcal{C} \cup \bigcup\{\mathcal{F}_n \mid n \in \mathbb{N}_1\} \cup \bigcup\{\mathcal{P}_n \mid n \in \mathbb{N}_1\}$. The notation $w \equiv w'$, where $w, w' \in \mathcal{S}ym$, is used to indicate that w and w' are identical.

A *signature* Σ is a subset of $\mathcal{C} \cup \bigcup\{\mathcal{F}_n \mid n \in \mathbb{N}_1\} \cup \bigcup\{\mathcal{P}_n \mid n \in \mathbb{N}_1\}$.

We write $\mathcal{S}ig$ for the set of all signatures. We write $\mathrm{C}(\Sigma)$, $\mathrm{F}_n(\Sigma)$, and $\mathrm{P}_n(\Sigma)$, where $\Sigma \in \mathcal{S}ig$ and $n \in \mathbb{N}_1$, for $\Sigma \cap \mathcal{C}$, $\Sigma \cap \mathcal{F}_n$, and $\Sigma \cap \mathcal{P}_n$, respectively.

The language of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ will be defined for a fixed but arbitrary signature Σ. This language will be called the language of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ over Σ or shortly the language of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$. The corresponding proof system and interpretation will be called the proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ and the interpretation of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$.

2.2 Terms and formulas

The language of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ contains terms and formulas. They are constructed according to the formation rules given below.

The set of all terms of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$, written $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, is inductively defined by the following formation rules:

- if $x \in \mathcal{V}$, then $x \in \mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$;
- if $c \in \mathrm{C}(\Sigma)$, then $c \in \mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$;
- if $f \in \mathrm{F}_n(\Sigma)$ and $t_1, \ldots, t_n \in \mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, then $f(t_1, \ldots, t_n) \in \mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$.

The set of all formulas of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$, written $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, is inductively defined by the following formation rules:

- $\mathsf{F} \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$;
- if $t_1, t_2 \in \mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, then $t_1 = t_2 \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$;
- if $P \in \mathrm{P}_n(\Sigma)$ and $t_1, \ldots, t_n \in \mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, then $P(t_1, \ldots, t_n) \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$;
- if $A \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, then $\neg A \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$;
- if $A_1, A_2 \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, then $A_1 \land A_2$, $A_1 \lor A_2$, $A_1 \supset A_2 \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$;
- if $x \in \mathcal{V}$ and $A \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, then $\forall x \bullet A$, $\exists x \bullet A \in \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$.

For the connectives $\neg$, $\land$, $\lor$, and $\supset$ and the quantifiers $\forall$ and $\exists$, the classical truth-conditions and falsehood-conditions are retained. Except for implications, a formula is classified as both-true-and-false exactly when it cannot be classified as true or false by these conditions.

2.3 Notational conventions and abbreviations

In the sequel, some notational conventions and abbreviations will be used.

The following will sometimes be used without mentioning (with or without subscripts): x as a syntactic variable ranging over all variable symbols from $\mathcal{V}$, t as a syntactic variable ranging over all terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, A as a syntactic variable ranging over all formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, and Γ as a syntactic variable ranging over all finite sets of formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$.

The string representation of terms and formulas suggested by the formation rules given above can lead to syntactic ambiguities. Parentheses are used to avoid such ambiguities. The need to use parentheses is reduced by ranking the precedence of the logical connectives $\neg$, $\land$, $\lor$, $\supset$. The enumeration presents this order from the highest precedence to the lowest precedence. Moreover, the scope of the quantifiers extends as far as possible to the right and $\forall x_1 \bullet \cdots \forall x_n \bullet A$ is usually written as $\forall x_1, \ldots, x_n \bullet A$.

Non-equality, truth, and bi-implication are defined as abbreviations: $t_1 \neq t_2$ stands for $\neg(t_1 = t_2)$, T stands for $\neg\mathsf{F}$, $A_1 \equiv A_2$ stands for $(A_1 \supset A_2) \land (A_2 \supset A_1)$.

2.4 Free variables and substitution

Free variables of a term or formula and substitution for variables in a term or formula are defined in the usual way.

We write *free*(e), where e is a term from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ or a formula from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, for the set of *free variables* of e. We write *free*(Γ), where Γ is a finite set of formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, for $\bigcup\{\mathit{free}(A) \mid A \in \Gamma\}$.

Let x be a variable symbol from $\mathcal{V}$, t be a term from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, and e be a term from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ or a formula from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$. Then $[x := t]e$ is the result of replacing the free occurrences of the variable symbol x in e by the term t, avoiding — by means of renaming of bound variables — free variables becoming bound in t.

3 Proof System of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$

The proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ is formulated as a sequent-style natural deduction proof system. This means that the inference rules have sequents as premises and conclusions. First, the notion of a sequent is introduced. Next, the inference rules of the proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ are presented. Then, the notion of a derivation of a sequent from a set of sequents and the notion of a proof of a sequent are introduced. An extension of the proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ which can serve as a proof system for $\mathrm{FOCL}(\Sigma)$ is also described.

3.1 Sequents

In $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$, a *sequent* is an expression of the form $\Gamma \vdash A$, where Γ is a finite set of formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and A is a formula from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$. We write $\vdash A$ instead of $\emptyset \vdash A$. Moreover, we write Γ, Γ' for $\Gamma \cup \Gamma'$ and A for $\{A\}$ on the left-hand side of a sequent.

The intended meaning of the sequent $\Gamma \vdash A$ is that the formula A is a logical consequence of the formulas Γ. There are several sensible notions of logical consequence in the case where formulas can be classified as both-true-and-false. The notion underlying $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is precisely defined in Section 4. It corresponds to the intuitive idea that one can draw conclusions that are not false from premises that are not false. Sequents are proved by (natural deduction) proofs obtained by using the rules of inference given below.

3.2 Rules of inference

The sequent-style natural deduction proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ consists of the inference rules given in Table 1. In this table, x is a syntactic variable ranging over all variable symbols from $\mathcal{V}$, t_1, t_2, and t are syntactic variables ranging over all terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, and A_1, A_2, A_3, and A are syntactic variables ranging over all formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$. Double lines indicate a two-way inference rule.

Table 1. Natural deduction proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$

$\boxed{\mathrm{I}}\ \dfrac{}{\Gamma, A \vdash A}$	$\boxed{\mathrm{EM}}\ \dfrac{}{\Gamma \vdash A \lor \neg A}$
$\boxed{\mathsf{T}\text{-I}}\ \dfrac{}{\Gamma \vdash \neg \mathsf{F}}$	$\boxed{\mathsf{F}\text{-E}}\ \dfrac{\Gamma \vdash \mathsf{F}}{\Gamma \vdash A}$
$\boxed{\land\text{-I}}\ \dfrac{\Gamma \vdash A_1 \quad \Gamma \vdash A_2}{\Gamma \vdash A_1 \land A_2}$	$\boxed{\land\text{-E}}\ \dfrac{\Gamma \vdash A_1 \land A_2}{\Gamma \vdash A_i}$ for $i = 1, 2$
$\boxed{\lor\text{-I}}\ \dfrac{\Gamma \vdash A_i}{\Gamma \vdash A_1 \lor A_2}$ for $i = 1, 2$	$\boxed{\lor\text{-E}}\ \dfrac{\Gamma \vdash A_1 \lor A_2 \quad \Gamma, A_1 \vdash A_3 \quad \Gamma, A_2 \vdash A_3}{\Gamma \vdash A_3}$
$\boxed{\supset\text{-I}}\ \dfrac{\Gamma, A_1 \vdash A_2}{\Gamma \vdash A_1 \supset A_2}$	$\boxed{\supset\text{-E}}\ \dfrac{\Gamma \vdash A_1 \supset A_2 \quad \Gamma \vdash A_1}{\Gamma \vdash A_2}$
$\boxed{\forall\text{-I}}\ \dfrac{\Gamma \vdash A}{\Gamma \vdash \forall x \bullet A}\ \dagger$	$\boxed{\forall\text{-E}}\ \dfrac{\Gamma \vdash \forall x \bullet A}{\Gamma \vdash [x := t]A}$
$\boxed{\exists\text{-I}}\ \dfrac{\Gamma \vdash [x := t]A}{\Gamma \vdash \exists x \bullet A}$	$\boxed{\exists\text{-E}}\ \dfrac{\Gamma \vdash \exists x \bullet A_1 \quad \Gamma, A_1 \vdash A_2}{\Gamma \vdash A_2}\ \ddagger$
$\boxed{=\text{-I}}\ \dfrac{}{\Gamma \vdash t = t}$	$\boxed{=\text{-E}}\ \dfrac{\Gamma \vdash t_1 = t_2 \quad \Gamma \vdash [x := t_1]A}{\Gamma \vdash [x := t_2]A}$
$\boxed{\neg\text{-M}}\ \begin{array}{c} \Gamma \vdash \neg\neg A \\ \hline\hline \Gamma \vdash A \end{array}$	$\boxed{\land\text{-M}}\ \begin{array}{c} \Gamma \vdash \neg(A_1 \land A_2) \\ \hline\hline \Gamma \vdash \neg A_1 \lor \neg A_2 \end{array}$
$\boxed{\lor\text{-M}}\ \begin{array}{c} \Gamma \vdash \neg(A_1 \lor A_2) \\ \hline\hline \Gamma \vdash \neg A_1 \land \neg A_2 \end{array}$	$\boxed{\supset\text{-M}}\ \begin{array}{c} \Gamma \vdash \neg(A_1 \supset A_2) \\ \hline\hline \Gamma \vdash A_1 \land \neg A_2 \end{array}$
$\boxed{\forall\text{-M}}\ \begin{array}{c} \Gamma \vdash \neg\forall x \bullet A \\ \hline\hline \Gamma \vdash \exists x \bullet \neg A \end{array}$	$\boxed{\exists\text{-M}}\ \begin{array}{c} \Gamma \vdash \neg\exists x \bullet A \\ \hline\hline \Gamma \vdash \forall x \bullet \neg A \end{array}$

† restriction on rule ∀-I: $x \notin \mathit{free}(\Gamma)$;
‡ restriction on rule ∃-E: $x \notin \mathit{free}(\Gamma \cup \{A_2\})$.

3.3 Derivations and proofs

In $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$, a *derivation* of a sequent $\Gamma \vdash A$ from a finite set of sequents $\mathcal{H}$ is a finite sequence $\langle s_1, \ldots, s_n \rangle$ of sequents such that s_n equals $\Gamma \vdash A$ and, for each $i \in \{1, \ldots, n\}$, one of the f ollowing conditions holds:

- $s_i \in \mathcal{H}$;
- s_i is the conclusion of an instance of some inference rule from the proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ whose premises are among $s_1, \ldots, s_{i-1}$.

A *proof* of a sequent $\Gamma \vdash A$ is a derivation of $\Gamma \vdash A$ from the empty set of sequents. A sequent $\Gamma \vdash A$ is said to be *provable* if there exists a proof of $\Gamma \vdash A$.

An inference rule that does not belong to the inference rules of some proof system is called a *derived inference rule* if there exists a derivation of the conclusion from the premises, using the inference rules of that proof system, for each instance of the rule.

The difference between CLuNs and $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is that bi-implication is a logical connective in CLuNs and must be defined as an abbreviation in $\mathrm{LPQ}^{\supset,\mathsf{F}}$. In [1], a proof system of CLuNs is presented which is formulated as a Hilbert system. Removing the axiom schemas A≡1, A≡2, and A≡3 from this proof system and taking formulas of the form $A_1 \equiv A_2$ in this proof system as abbreviations yields a proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ formulated as a Hilbert system. Henceforth, this proof system will be referred to as the H proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ and the proof system presented in Section 3.2 will be referred to as the ND proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}$.

3.4 FOCL(Σ)

In FOCL, the same assumptions about symbols are made as in $\mathrm{LPQ}^{\supset,\mathsf{F}}$ and the notion of a signature is defined as in $\mathrm{LPQ}^{\supset,\mathsf{F}}$. The languages of FOCL($\Sigma$) and $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ are the same. A natural deduction proof system of FOCL(Σ) can be obtained by adding the following inference rule to the ND proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$:

$$\boxed{\mathrm{C}}\,\frac{\Gamma \vdash A_1 \quad \Gamma \vdash \neg A_1}{\Gamma \vdash A_2}\;.$$

This proof system is known to be sound and complete. There exist better known alternatives to it, but this proof system is arguably the most appropriate one in this paper.

In Section 5, the sequents of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ will be translated to sequents of FOCL(Σ') (Σ' is a particular signature related to Σ). The translation concerned has the property that what can be derived remains the same after translation. This implies that the inference rules of the proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ become derived inference rules of the above-mentioned proof system of FOCL(Σ') after translation. Thus, the translation provides a logical justification for the inference rules of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$. A model-theoretic justification is afforded by the interpretation given in Section 4.

4 Interpretation of Terms and Formulas of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$

The proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is based on the interpretation of the terms and formulas of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ presented below: the inference rules preserve validity under this interpretation. The interpretation is given relative to a structure and an assignment. First, the notion of a structure and the notion of an assignment are introduced. Next, the interpretation of the terms and formulas of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ is presented.

4.1 Structures

The terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and the formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ are interpreted in structures which consist of a non-empty domain of individuals and an interpretation of every symbol in the signature Σ and the equality symbol. The domain of truth values consists of three values: t (*true*), f (*false*), and b (*both true and false*).

A structure $\mathbf{A}$ of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ consists of:

- a set $\mathcal{U}^{\mathbf{A}}$, the *domain* of $\mathbf{A}$, such that $\mathcal{U}^{\mathbf{A}} \neq \emptyset$ and $\mathcal{U}^{\mathbf{A}} \cap \{\mathsf{t}, \mathsf{f}, \mathsf{b}\} = \emptyset$;
- for each $c \in \mathrm{C}(\Sigma)$,
  $$\text{an element } c^{\mathbf{A}} \in \mathcal{U}^{\mathbf{A}};$$
- for each $n \in \mathbb{N}_1$, for each $f \in \mathrm{F}_n(\Sigma)$,
  $$\text{a function } f^{\mathbf{A}} : \underbrace{\mathcal{U}^{\mathbf{A}} \times \cdots \times \mathcal{U}^{\mathbf{A}}}_{n \text{ times}} \to \mathcal{U}^{\mathbf{A}};$$
- for each $n \in \mathbb{N}_1$, for each $P \in \mathrm{P}_n(\Sigma)$,
  $$\text{a function } P^{\mathbf{A}} : \underbrace{\mathcal{U}^{\mathbf{A}} \times \cdots \times \mathcal{U}^{\mathbf{A}}}_{n \text{ times}} \to \{\mathsf{t}, \mathsf{f}, \mathsf{b}\};$$
- a function $=^{\mathbf{A}} : \mathcal{U}^{\mathbf{A}} \times \mathcal{U}^{\mathbf{A}} \to \{\mathsf{t}, \mathsf{f}, \mathsf{b}\}$ such that, for each $d \in \mathcal{U}^{\mathbf{A}}$,
  $$=^{\mathbf{A}} (d, d) = \mathsf{t} \text{ or } =^{\mathbf{A}} (d, d) = \mathsf{b}.$$

Instead of $w^{\mathbf{A}}$ we write w when it is clear from the context that the interpretation of symbol w in structure $\mathbf{A}$ is meant.

4.2 Assignments

An assignment in a structure $\mathbf{A}$ of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ assigns elements from $\mathcal{U}^{\mathbf{A}}$ to the variable symbols from $\mathcal{V}$. The interpretation of the terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and the formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ in $\mathbf{A}$ is given with respect to an assignment α in $\mathbf{A}$.

Let $\mathbf{A}$ be a structure of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$. Then an *assignment* in $\mathbf{A}$ is a function $\alpha : \mathcal{V} \to \mathcal{U}^{\mathbf{A}}$. For every assignment α in $\mathbf{A}$, variable symbol $x \in \mathcal{V}$ and element $d \in \mathcal{U}^{\mathbf{A}}$, we write $\alpha(x \to d)$ for the assignment α' in $\mathbf{A}$ such that $\alpha'(x) = d$ and $\alpha'(y) = \alpha(y)$ if $y \not\equiv x$.

4.3 Interpretation

The interpretation of the terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ is given by a function mapping term t, structure $\mathbf{A}$ and assignment α in $\mathbf{A}$ to the element of $\mathcal{U}^{\mathbf{A}}$ that is the value of t in $\mathbf{A}$ under assignment α. Similarly, the interpretation of the formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ is given by a function mapping formula A, structure $\mathbf{A}$ and assignment α in $\mathbf{A}$ to the element of $\{\mathsf{t}, \mathsf{f}, \mathsf{b}\}$ that is the truth value of A in $\mathbf{A}$ under assignment α. We write $[\![t]\!]^{\mathbf{A}}_{\alpha}$ and $[\![A]\!]^{\mathbf{A}}_{\alpha}$ for these interpretations.

The interpretation functions for the terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and the formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ are inductively defined in Table 2. In this table, x is a syntactic variable ranging over all variable symbols from $\mathcal{V}$, c is a syntactic variable ranging over all constant symbols from $\mathrm{C}(\Sigma)$, f is a syntactic variable ranging over all function symbols from $\mathrm{F}_n(\Sigma)$ (where n is understood from the context), t_1, ...,

Table 2. Interpretation of the language of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$

$$
\begin{aligned}
\llbracket x \rrbracket^{\mathbf{A}}_{\alpha} &= \alpha(x)\ , \\
\llbracket c \rrbracket^{\mathbf{A}}_{\alpha} &= c^{\mathbf{A}}\ , \\
\llbracket f(t_1,\ldots,t_n) \rrbracket^{\mathbf{A}}_{\alpha} &= f^{\mathbf{A}}(\llbracket t_1 \rrbracket^{\mathbf{A}}_{\alpha},\ldots,\llbracket t_n \rrbracket^{\mathbf{A}}_{\alpha}) \\
\\
\llbracket \mathsf{F} \rrbracket^{\mathbf{A}}_{\alpha} &= \mathsf{f}\ , \\
\llbracket t_1 = t_2 \rrbracket^{\mathbf{A}}_{\alpha} &= {=^{\mathbf{A}}}\ (\llbracket t_1 \rrbracket^{\mathbf{A}}_{\alpha},\llbracket t_2 \rrbracket^{\mathbf{A}}_{\alpha})\ , \\
\llbracket P(t_1,\ldots,t_n) \rrbracket^{\mathbf{A}}_{\alpha} &= P^{\mathbf{A}}(\llbracket t_1 \rrbracket^{\mathbf{A}}_{\alpha},\ldots,\llbracket t_n \rrbracket^{\mathbf{A}}_{\alpha})\ , \\
\llbracket \neg A \rrbracket^{\mathbf{A}}_{\alpha} &= \begin{cases} \mathsf{t} & \text{if } \llbracket A \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f} \\ \mathsf{f} & \text{if } \llbracket A \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{t} \\ \mathsf{b} & \text{otherwise,} \end{cases} \\
\llbracket A_1 \wedge A_2 \rrbracket^{\mathbf{A}}_{\alpha} &= \begin{cases} \mathsf{t} & \text{if } \llbracket A_1 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{t} \text{ and } \llbracket A_2 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{t} \\ \mathsf{f} & \text{if } \llbracket A_1 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f} \text{ or } \llbracket A_2 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f} \\ \mathsf{b} & \text{otherwise,} \end{cases} \\
\llbracket A_1 \vee A_2 \rrbracket^{\mathbf{A}}_{\alpha} &= \begin{cases} \mathsf{t} & \text{if } \llbracket A_1 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{t} \text{ or } \llbracket A_2 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{t} \\ \mathsf{f} & \text{if } \llbracket A_1 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f} \text{ and } \llbracket A_2 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f} \\ \mathsf{b} & \text{otherwise,} \end{cases} \\
\llbracket A_1 \supset A_2 \rrbracket^{\mathbf{A}}_{\alpha} &= \begin{cases} \mathsf{t} & \text{if } \llbracket A_1 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f} \text{ or } \llbracket A_2 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{t} \\ \mathsf{f} & \text{if } \llbracket A_1 \rrbracket^{\mathbf{A}}_{\alpha} \neq \mathsf{f} \text{ and } \llbracket A_2 \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f} \\ \mathsf{b} & \text{otherwise,} \end{cases} \\
\llbracket \forall x \bullet A \rrbracket^{\mathbf{A}}_{\alpha} &= \begin{cases} \mathsf{t} & \text{if, for all } d \in \mathcal{U}^{\mathbf{A}},\ \llbracket A \rrbracket^{\mathbf{A}}_{\alpha(x \to d)} = \mathsf{t} \\ \mathsf{f} & \text{if, for some } d \in \mathcal{U}^{\mathbf{A}},\ \llbracket A \rrbracket^{\mathbf{A}}_{\alpha(x \to d)} = \mathsf{f} \\ \mathsf{b} & \text{otherwise.} \end{cases} \\
\llbracket \exists x \bullet A \rrbracket^{\mathbf{A}}_{\alpha} &= \begin{cases} \mathsf{t} & \text{if, for some } d \in \mathcal{U}^{\mathbf{A}},\ \llbracket A \rrbracket^{\mathbf{A}}_{\alpha(x \to d)} = \mathsf{t} \\ \mathsf{f} & \text{if, for all } d \in \mathcal{U}^{\mathbf{A}},\ \llbracket A \rrbracket^{\mathbf{A}}_{\alpha(x \to d)} = \mathsf{f} \\ \mathsf{b} & \text{otherwise.} \end{cases}
\end{aligned}
$$

t_n are syntactic variables ranging over all terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, P is a syntactic variable ranging over all predicate symbols from $\mathrm{P}_n(\Sigma)$ (where n is understood from the context), and A_1, A_2, and A are syntactic variables ranging over all formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$,

The logical consequence relation of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ is based on the idea that a formula A holds in a structure $\mathbf{A}$ under an assignment α in $\mathbf{A}$ if $\llbracket A \rrbracket^{\mathbf{A}}_{\alpha} \in \{\mathsf{t}, \mathsf{b}\}$.

Let Γ be a finite set of formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and A be a formula from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$. Then A is a *logical consequence* of Γ, written $\Gamma \models A$, iff for all structures $\mathbf{A}$ of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$, for all assignments α in $\mathbf{A}$, $\llbracket A' \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f}$ for some $A' \in \Gamma$ or $\llbracket A \rrbracket^{\mathbf{A}}_{\alpha} \in \{\mathsf{t}, \mathsf{b}\}$.

As mentioned before, the difference between CLuNs and $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is that bi-implication is a logical connective in CLuNs and must be defined as an abbreviation in $\mathrm{LPQ}^{\supset,\mathsf{F}}$. In [1], an interpretation of the formulas of CLuNs is presented whose restriction to formulas without occurrences of the bi-implication connective is essentially the same as the interpretation of the formulas of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ given above. The soundness and completeness properties for the Hilbert proof system of CLuNs proved in [1] directly carry over to $\mathrm{LPQ}^{\supset,\mathsf{F}}$.

Theorem 1. *The ND proof system of* $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ *presented in Section 3.2 is sound and complete, i.e., for each finite set* Γ *of formulas from* $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ *and each formula* A *from* $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, $\Gamma \vdash A$ *is provable in the ND proof system of* $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ *iff* $\Gamma \models A$.

Proof. Because it is known from [1] that these properties hold for the H proof system of $\mathrm{LPQ}^{\supset,\mathsf{F}}$, it is sufficient to prove that, for each finite set Γ of formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and each formula A from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, $\Gamma \vdash A$ is provable in the H system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ iff $\Gamma \vdash A$ is provable in the ND system of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$.

The only if part is straightforwardly proved by induction on the length of the proof of $\Gamma \vdash A$ in the H system, using that (a) for each axiom A' of the H system, $\vdash A'$ can be proved in the ND system and (b) for each inference rule of the H system, a corresponding derived inference rule of the ND system can be found.

The if part is straightforwardly proved by induction on length of the proof of $\Gamma \vdash A$ in the ND system, using that (a) the standard deduction theorem holds for the H system, (b) for each inference rule of the ND system different from I, $\supset$-E, $\forall$-I, and $\exists$-E, there exists a corresponding axiom of the H system, (c) for each of the inference rules $\supset$-E, $\forall$-I, and $\exists$-E, a corresponding derived inference rule of the H system can be found, and (d) $\vdash A \supset A$ can be proved in the H system. ☐

In addition to the notion of logical consequence, the notions of logical equivalence and consistency are semantic notions that are relevant for a paraconsistent logic. The logical equivalence relation is semantically defined as it is semantically defined in classical logic. Let A_1 and A_2 be formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$. Then A_1 is *logically equivalent* to A_2, written $A_1 \Leftrightarrow A_2$, iff for all structures $\mathbf{A}$ of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$, for all assignments α in $\mathbf{A}$, $[\![A_1]\!]^{\mathbf{A}}_{\alpha} = [\![A_2]\!]^{\mathbf{A}}_{\alpha}$. The consistency property is not semantically definable in classical logic. Let A_1 and A_2 be formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$. Then A is *consistent* iff for all structures $\mathbf{A}$ of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$, for all assignments α in $\mathbf{A}$, $[\![A_1]\!]^{\mathbf{A}}_{\alpha} \neq \mathsf{b}$.

Unlike in classical logic, it does not hold that $A_1 \Leftrightarrow A_2$ iff $\vdash A_1 \equiv A_2$, but it holds that $A_1 \Leftrightarrow A_2$ iff $\vdash (A_1 \equiv A_2) \land (\neg A_1 \equiv \neg A_2)$. Moreover, it holds that A is consistent iff $\vdash (A \supset \mathsf{F}) \land (\neg A \supset \mathsf{F})$. In other words, the notions of logical equivalence and consistency can both be internalized in $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$.

5 Embedding of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ into FOCL(Σ)

To give a classical-logic view of $\mathrm{LPQ}^{\supset,\mathsf{F}}$, the terms, formulas and sequents of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ are translated in this section to terms, formulas and sequents, respectively, of FOCL(Σ). The mappings concerned provide a uniform embedding of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ into FOCL(Σ). What can be proved remains the same after translation. Thus, the mappings provide both a classical-logic explanation of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ and a logical justification of its proof system.

5.1 Translation

In the translation, a canonical mapping from symbols of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ to symbols of $\mathrm{FOCL}(\Sigma)$ is assumed. For each $w \in \mathcal{S}\mathit{ym}$, we write $\boldsymbol{w}$ for the symbol to which w is mapped and, for each $W \subseteq \mathcal{S}\mathit{ym}$, we write $\boldsymbol{W}$ for the set $\{\boldsymbol{w} \mid w \in W\}$. The mapping concerned is further assumed to be injective and such that:

- each $x \in \mathcal{V}$ is mapped to an $\boldsymbol{x} \in \boldsymbol{\mathcal{V}}$,
- each $c \in \mathrm{C}(\Sigma)$ is mapped to a $\boldsymbol{c} \in \mathbf{C}(\boldsymbol{\Sigma})$,
- each $f \in \mathrm{F}_n(\Sigma)$ is mapped to an $\boldsymbol{f} \in \mathbf{F}_{\boldsymbol{n}}(\boldsymbol{\Sigma})$,
- each $P \in \mathrm{P}_n(\Sigma)$ is mapped to a $\boldsymbol{P} \in \mathbf{P}_{\boldsymbol{n}}(\boldsymbol{\Sigma})$.

It is also assumed that $\mathsf{U}, \mathsf{B} \in \boldsymbol{\mathcal{P}_1}$, $\mathsf{true}, \mathsf{false}, \mathsf{both} \in \boldsymbol{\mathcal{C}}$, $X \in \boldsymbol{\mathcal{V}}$, and, for each $i \in \mathbb{N}_1$, $X_i \in \boldsymbol{\mathcal{V}}$.

For the translation of terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, one translation function is used:

$$\llparenthesis \_ \rrparenthesis : \mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma) \to \mathcal{T}_{\mathrm{FOCL}}(\boldsymbol{\Sigma})$$

and for the translation of formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, three translation functions are used:

$$\begin{aligned}
\llparenthesis \_ \rrparenthesis^{\mathsf{t}} &: \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma) \to \mathcal{F}_{\mathrm{FOCL}}(\boldsymbol{\Sigma}),\\
\llparenthesis \_ \rrparenthesis^{\mathsf{f}} &: \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma) \to \mathcal{F}_{\mathrm{FOCL}}(\boldsymbol{\Sigma}),\\
\llparenthesis \_ \rrparenthesis^{\mathsf{b}} &: \mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma) \to \mathcal{F}_{\mathrm{FOCL}}(\boldsymbol{\Sigma}).
\end{aligned}$$

For a formula A from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, there are three translations of A to $\mathrm{FOCL}(\boldsymbol{\Sigma})$. $\llparenthesis A \rrparenthesis^{\mathsf{t}}$ is a formula of $\mathrm{FOCL}(\boldsymbol{\Sigma})$ stating that the formula A of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ is true in $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$. Likewise, $\llparenthesis A \rrparenthesis^{\mathsf{f}}$ is a formula of $\mathrm{FOCL}(\boldsymbol{\Sigma})$ stating that the formula A of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ is false in $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ and $\llparenthesis A \rrparenthesis^{\mathsf{b}}$ is a formula of $\mathrm{FOCL}(\boldsymbol{\Sigma})$ stating that the formula A of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ is both true and false in $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$.

The translation functions for the terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and the formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ are inductively defined in Table 3. In this table, x is a syntactic variable ranging over all variable symbols from $\mathcal{V}$, c is a syntactic variable ranging over all constant symbols from $\mathrm{C}(\Sigma)$, f is a syntactic variable ranging over all function symbols from $\mathrm{F}_n(\Sigma)$ (where n is understood from the context), $t_1, \ldots, t_n$ are syntactic variables ranging over all terms from $\mathcal{T}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, P is a syntactic variable ranging over all predicate symbols from $\mathrm{P}_n(\Sigma)$ (where n is understood from the context), and A_1, A_2, and A are syntactic variables ranging over all formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$.

The translation rules strongly resemble the interpretation rules of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ that are given in Section 4: the rules for the mapping $\llparenthesis \_ \rrparenthesis^{\mathsf{t}}$ correspond to the truth-conditions and the rules for the mapping $\llparenthesis \_ \rrparenthesis^{\mathsf{f}}$ correspond to the falsehood-conditions.

A translation for sequents of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ can also be devised:

$$\llparenthesis \Gamma \vdash A \rrparenthesis = \mathrm{Ax}(\Sigma, \Gamma \cup \{A\}) \cup \{\llparenthesis A' \rrparenthesis^{\mathsf{t}} \lor \llparenthesis A' \rrparenthesis^{\mathsf{b}} \mid A' \in \Gamma\} \vdash \llparenthesis A \rrparenthesis^{\mathsf{t}} \lor \llparenthesis A \rrparenthesis^{\mathsf{b}} \ ,$$

Table 3. Translation of the language of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\varSigma)$

$$
\begin{aligned}
\llparenthesis x \rrparenthesis &= \boldsymbol{x}\ , \\
\llparenthesis c \rrparenthesis &= \boldsymbol{c}\ , \\
\llparenthesis f(t_1,\ldots,t_n) \rrparenthesis &= \boldsymbol{f}(\llparenthesis t_1 \rrparenthesis,\ldots,\llparenthesis t_n \rrparenthesis)\ . \\[1em]
\llparenthesis \mathsf{F} \rrparenthesis^{\mathsf{t}} &= \mathsf{F}\ , \\
\llparenthesis t_1 = t_2 \rrparenthesis^{\mathsf{t}} &= \mathsf{eq}(\llparenthesis t_1 \rrparenthesis,\llparenthesis t_2 \rrparenthesis) = \mathsf{true}\ , \\
\llparenthesis P(t_1,\ldots,t_n) \rrparenthesis^{\mathsf{t}} &= \boldsymbol{P}(\llparenthesis t_1 \rrparenthesis,\ldots,\llparenthesis t_n \rrparenthesis) = \mathsf{true}\ , \\
\llparenthesis \neg A \rrparenthesis^{\mathsf{t}} &= \llparenthesis A \rrparenthesis^{\mathsf{f}}\ , \\
\llparenthesis A_1 \land A_2 \rrparenthesis^{\mathsf{t}} &= \llparenthesis A_1 \rrparenthesis^{\mathsf{t}} \land \llparenthesis A_2 \rrparenthesis^{\mathsf{t}}\ , \\
\llparenthesis A_1 \lor A_2 \rrparenthesis^{\mathsf{t}} &= \llparenthesis A_1 \rrparenthesis^{\mathsf{t}} \lor \llparenthesis A_2 \rrparenthesis^{\mathsf{t}}\ , \\
\llparenthesis A_1 \supset A_2 \rrparenthesis^{\mathsf{t}} &= \llparenthesis A_1 \rrparenthesis^{\mathsf{f}} \lor \llparenthesis A_2 \rrparenthesis^{\mathsf{t}}\ , \\
\llparenthesis \forall x \bullet A \rrparenthesis^{\mathsf{t}} &= \forall \boldsymbol{x} \bullet \mathsf{U}(\boldsymbol{x}) \supset \llparenthesis A \rrparenthesis^{\mathsf{t}}\ , \\
\llparenthesis \exists x \bullet A \rrparenthesis^{\mathsf{t}} &= \exists \boldsymbol{x} \bullet \mathsf{U}(\boldsymbol{x}) \land \llparenthesis A \rrparenthesis^{\mathsf{t}}\ , \\[1em]
\llparenthesis \mathsf{F} \rrparenthesis^{\mathsf{f}} &= \mathsf{T}\ , \\
\llparenthesis t_1 = t_2 \rrparenthesis^{\mathsf{f}} &= \mathsf{eq}(\llparenthesis t_1 \rrparenthesis,\llparenthesis t_2 \rrparenthesis) = \mathsf{false}\ , \\
\llparenthesis P(t_1,\ldots,t_n) \rrparenthesis^{\mathsf{f}} &= \boldsymbol{P}(\llparenthesis t_1 \rrparenthesis,\ldots,\llparenthesis t_n \rrparenthesis) = \mathsf{false}\ , \\
\llparenthesis \neg A \rrparenthesis^{\mathsf{f}} &= \llparenthesis A \rrparenthesis^{\mathsf{t}}\ , \\
\llparenthesis A_1 \land A_2 \rrparenthesis^{\mathsf{f}} &= \llparenthesis A_1 \rrparenthesis^{\mathsf{f}} \lor \llparenthesis A_2 \rrparenthesis^{\mathsf{f}}\ , \\
\llparenthesis A_1 \lor A_2 \rrparenthesis^{\mathsf{f}} &= \llparenthesis A_1 \rrparenthesis^{\mathsf{f}} \land \llparenthesis A_2 \rrparenthesis^{\mathsf{f}}\ , \\
\llparenthesis A_1 \supset A_2 \rrparenthesis^{\mathsf{f}} &= \neg \llparenthesis A_1 \rrparenthesis^{\mathsf{f}} \land \llparenthesis A_2 \rrparenthesis^{\mathsf{f}}\ , \\
\llparenthesis \forall x \bullet A \rrparenthesis^{\mathsf{f}} &= \exists \boldsymbol{x} \bullet \mathsf{U}(\boldsymbol{x}) \land \llparenthesis A \rrparenthesis^{\mathsf{f}}\ , \\
\llparenthesis \exists x \bullet A \rrparenthesis^{\mathsf{f}} &= \forall \boldsymbol{x} \bullet \mathsf{U}(\boldsymbol{x}) \supset \llparenthesis A \rrparenthesis^{\mathsf{f}}\ , \\[1em]
\llparenthesis A \rrparenthesis^{\mathsf{b}} &= \neg(\llparenthesis A \rrparenthesis^{\mathsf{t}} \lor \llparenthesis A \rrparenthesis^{\mathsf{f}})\ .
\end{aligned}
$$

where $\mathrm{Ax}(\varSigma, \varGamma \cup \{A\})$ consists of the following formulas:

- $\mathsf{true} \neq \mathsf{false} \land \mathsf{true} \neq \mathsf{both} \land \mathsf{false} \neq \mathsf{both}$;
- $\forall X \bullet \mathsf{B}(X) \equiv (X = \mathsf{true} \lor X = \mathsf{false} \lor X = \mathsf{both})$;
- $\exists X \bullet \mathsf{U}(X)$;
- $\forall X \bullet \neg(\mathsf{U}(X) \equiv \mathsf{B}(X))$;
- $\forall X_1,\ldots,X_n \bullet \mathsf{U}(X_1) \land \ldots \land \mathsf{U}(X_n) \supset \mathsf{U}(\boldsymbol{f}(X_1,\ldots,X_n))$ for each $f \in \mathrm{F}_n(\varSigma)$, for each $n \in \mathbb{N}_1$;
- $\forall X_1,\ldots,X_n \bullet \mathsf{U}(X_1) \land \ldots \land \mathsf{U}(X_n) \supset \mathsf{B}(\boldsymbol{P}(X_1,\ldots,X_n))$ for each $P \in \mathrm{P}_n(\varSigma)$, for each $n \in \mathbb{N}_1$;
- $\forall X_1, X_2 \bullet \mathsf{U}(X_1) \land \mathsf{U}(X_2) \supset \mathsf{B}(\mathsf{eq}(X_1, X_2))$;
- $\forall X \bullet \mathsf{eq}(X, X) = \mathsf{true} \lor \mathsf{eq}(X, X) = \mathsf{both}$;
- $\mathsf{U}(\boldsymbol{x})$ for each $x \in \mathit{free}(\varGamma \cup \{A\})$.

$\mathrm{Ax}(\varSigma, \varGamma \cup \{A\})$ contains formulas asserting that the domain of truth values contains exactly three elements, the domain of individuals contains at least one element and is disjoint from the domain of truth values, application of a function yields an element from the domain of individuals, application of a predicate, including the equality predicate, yields an element from the domain of truth values, application of the equality predicate does not yield false if the arguments are identical, and free variables are always elements from the domain of individuals.

5.2 Embedding

An important property of the translation of sequents of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ to sequents of $\mathrm{FOCL}(\boldsymbol{\Sigma}\cup\{\mathsf{true},\mathsf{false},\mathsf{both},\mathsf{B},\mathsf{U},\mathsf{eq}\})$ presented above is that what can be proved remains the same after translation. This means that the translation provides a uniform embedding of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ into $\mathrm{FOCL}(\boldsymbol{\Sigma}\cup\{\mathsf{true},\mathsf{false},\mathsf{both},\mathsf{B},\mathsf{U},\mathsf{eq}\})$.

Theorem 2. *For each finite set Γ of formulas from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$ and each formula A from $\mathcal{F}_{\mathrm{LPQ}^{\supset,\mathsf{F}}}(\Sigma)$, $\Gamma \vdash A$ is provable in* $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ *iff $\llparenthesis \Gamma \vdash A \rrparenthesis$ is provable in* $\mathrm{FOCL}(\boldsymbol{\Sigma}\cup\{\mathsf{true},\mathsf{false},\mathsf{both},\mathsf{B},\mathsf{U},\mathsf{eq}\})$.

Proof. The only if part is easily proved by induction over the length of a proof of $\Gamma \vdash A$ and case distinction on the last inference rule applied, using that the ND proof system for $\mathrm{FOCL}(\boldsymbol{\Sigma}\cup\{\mathsf{true},\mathsf{false},\mathsf{both},\mathsf{B},\mathsf{U},\mathsf{eq}\})$ described in Section 3.4 contains all inference rules of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$.

The if part is proved making use of Theorem 1. Let $\mathbf{A}$ be a structure of $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$. Then $\mathbf{A}$ can be transformed in a natural way into a structure $\mathbf{A}^*$ of $\mathrm{FOCL}(\boldsymbol{\Sigma}\cup\{\mathsf{true},\mathsf{false},\mathsf{both},\mathsf{B},\mathsf{U},\mathsf{eq}\})$ with the following properties: $\llbracket A \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{t}$ iff $\llbracket \llparenthesis A \rrparenthesis^{\mathsf{t}} \rrbracket^{\mathbf{A}^*}_{\alpha} = \mathsf{t}$, $\llbracket A \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{f}$ iff $\llbracket \llparenthesis A \rrparenthesis^{\mathsf{f}} \rrbracket^{\mathbf{A}^*}_{\alpha} = \mathsf{t}$, and $\llbracket A \rrbracket^{\mathbf{A}}_{\alpha} = \mathsf{b}$ iff $\llbracket \llparenthesis A \rrparenthesis^{\mathsf{b}} \rrbracket^{\mathbf{A}^*}_{\alpha} = \mathsf{t}$ (for all assignments α in $\mathbf{A}$). Now assume that $\mathbf{A}$ is a counter-model for $\Gamma \vdash A$. Then, for its above-mentioned properties, $\mathbf{A}^*$ is a counter-model for $\llparenthesis \Gamma \vdash A \rrparenthesis$. From this, by Theorem 1, the if part follows immediately. □

The translation of sequents extends to inference rules in the obvious way.

Corollary 1. *The translation of the inference rules of the presented proof system of* $\mathrm{LPQ}^{\supset,\mathsf{F}}(\Sigma)$ *are derived inference rules of the proof system of* FOCL $(\boldsymbol{\Sigma}\cup\{\mathsf{true},\mathsf{false},\mathsf{both},\mathsf{B},\mathsf{U},\mathsf{eq}\})$ *described in Section 3.4.*

6 Discussion

In this section, properties the propositional fragment of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ and the different ways in which $\mathrm{LPQ}^{\supset,\mathsf{F}}$ is close to classical logic are briefly discussed.

6.1 On the propositional fragment of $\mathbf{LPQ}^{\supset,\mathsf{F}}$

In [8], the propositional fragment of $\mathrm{LPQ}^{\supset,\mathsf{F}}$, called $\mathrm{LP}^{\supset,\mathsf{F}}$, is presented. In that paper, it is among other things shown that $\mathrm{LP}^{\supset,\mathsf{F}}$ is the only three-valued paraconsistent propositional logic whose logical consequence relation has all properties proposed as desirable for such logics in the literature and whose logical equivalence relation satisfies the identity, annihilation, idempotent, and commutative laws for conjunction and disjunction, the double negation law for negation, and two laws that uniquely characterize its implication connective.

Because closeness to classical logic is generally considered important, the above-mentioned properties of the logical equivalence relation concerning conjunction, disjunction, and negation should arguably also be taken as desirable for paraconsistent propositional logics. Unlike in classical logic, it does not hold in three-valued paraconsistent logics that logical equivalence is the same as logical consequence and its inverse. This necessitates taking desirable properties of the logical equivalence relation explicitly into account. If the above-mentioned properties concerning conjunction, disjunction, and negation are taken into account, then the number of 'ideal' three-valued paraconsistent propositional logics reduces from 8192 to 16, among which LP$^{\supset,\mathsf{F}}$ (cf. [8]).

In [2], an application of LP$^{\supset,\mathsf{F}}$ can be found. The properties of the logical equivalence relation that are essential for that application, among which most, but not all, properties considered desirable above, reduces the number of ideal three-valued paraconsistent propositional logics that are applicable even to 1, namely LP$^{\supset,\mathsf{F}}$. This strengthens the impression that among the paraconsistent logics that deserves most attention are LP$^{\supset,\mathsf{F}}$ and its first-order extensions. However, the question arises whether a paraconsistent logic is really needed to deal with contradictory sets of formulas. The embedding of LPQ$^{\supset,\mathsf{F}}$ into FOCL given in this paper shows that it can be dealt with in classical logic but in a much less convenient way.

6.2 On ways in which LPQ$^{\supset,\mathsf{F}}$ is close to classical logic

LPQ$^{\supset,\mathsf{F}}$ is a paraconsistent logic whose logical consequence relation and logical equivalence relation have all properties proposed as desirable for such logics. These properties are all related to closeness to classical logic. Moreover, LPQ$^{\supset,\mathsf{F}}$ has no connective that is foreign to classical logic and the inference rules of its natural deduction proof system are all known from classical logic:

- except for the inference rules concerning the negation connective, the inference rules are the ones found in all natural deduction proof systems for classical logic;
- the inference rules concerning the negation connective are a rule that corresponds to the law of the excluded middle and rules that correspond to the de Morgan's laws for all connectives and quantifiers;
- the rule corresponding to the law of the excluded middle is also found in natural deduction proof systems for classical logic and the rules corresponding to the de Morgan's laws are well-known derived rules of natural deduction proof systems for classical logic.

This means that natural deduction reasoning in the setting of LPQ$^{\supset,\mathsf{F}}$ differs from classical natural deduction reasoning only by slightly different, but classically justifiable, reasoning about negations.

7 Concluding Remarks

The paraconsistent logic $\mathrm{LPQ}^{\supset,\mathsf{F}}$ has been presented. A sequent-style natural deduction proof system has been given for this logic. A natural deduction proof system has not been given before in the literature for one of the logics that are essentially the same as $\mathrm{LPQ}^{\supset,\mathsf{F}}$. In addition to the model-theoretic justification of the proof system, a logical justification by means of an embedding into classical logic has been given. Thus, a classical-logic view of $\mathrm{LPQ}^{\supset,\mathsf{F}}$ has been provided. It appears that a classical-logic view of a paraconsistent logic has not been given before.

A classical-logic view of a paracomplete logic has been given before in [7], following the same approach as in the current paper. It is likely that this approach works for all truth-functional finitely-valued logics. The approach concerned is reminiscent of the method, described in [3], to reduce the many-valued interpretation of the formulas of a truth-functional finitely-valued logic to a two-valued interpretation.

References

1. Batens, D., de Clercq, K.: A rich paraconsistent extension of full positive logic. Logique et Analyse 185–188, 227–257 (2004)
2. Bergstra, J.A., Middelburg, C.A.: Contradiction-tolerant process algebra with propositional signals. Fundamenta Informaticae 153(1–2), 29–55 (2017)
3. Caleiro, C., Carnielli, W.A., Coniglio, M.E., Marcos, J.: Two's company: The humbug of many logical values. In: Beziau, J.Y. (ed.) Logica Universalis. pp. 169–199. Birkhäuser Verlag, Basel (2005)
4. Carnielli, W.A., Marcos, J., de Amo, S.: Formal inconsistency and evolutionary databases. Logic and Logical Philosophy 8, 115–152 (2000)
5. Coniglio, M.E., Figallo-Orellano, A., Golzio, A.C.: First-order swap structures semantics for some logics of formal inconsistency. Journal of Logic and Computation, doi:10.1093/logcom/exaa027 (2020)
6. D´Ottaviano, I.M.L.: The completeness and compactness of a three-valued first-order logic. Revista Colombiana de Matemáticas 19, 77–94 (1985)
7. Jones, C.B., Middelburg, C.A.: A typed logic of partial functions reconstructed classically. Acta Informatica 31(5), 399–430 (1994)
8. Middelburg, C.A.: On the strongest three-valued paraconsistent logic contained in classical logic. `arXiv:1702.03414v5 [cs.LO]` (2017)
9. Picollo, L.: Truth in a logic of formal inconsistency: How classical can it get? Logic Journal of the IGPL, doi:10.1093/jigpal/jzy059 (2018)
10. Priest, G.: The logic of paradox. Journal of Philosophical Logic 8(1), 219–241 (1979)