A Tableau System for Quasi-Hybrid Logic

Diana Costa and Manuel A. Martins$^{(\boxtimes)}$

CIDMA - Department Mathematics, University Aveiro, Aveiro, Portugal
{dianafcosta,martins}@ua.pt

Abstract. Hybrid logic is a valuable tool for specifying relational structures, at the same time that allows defining accessibility relations between states, it provides a way to nominate and make mention to what happens at each specific state. However, due to the many sources nowadays available, we may need to deal with contradictory information. This is the reason why we came with the idea of Quasi-hybrid logic, which is a paraconsistent version of hybrid logic capable of dealing with inconsistencies in the information, written as hybrid formulas.

In [5] we have already developed a semantics for this paraconsistent logic. In this paper we go a step forward, namely we study its proof-theoretical aspects. We present a complete tableau system for Quasi-hybrid logic, by combining both tableaux for Quasi-classical and Hybrid logics.

1 Introduction

Hybrid logic [1] is the simplest tool for the description of relational structures: it allows establishing accessibility relations between states and furthermore, nominating and making mention to what happens at specific states.

Unfortunately, we may collect contradictory information due to the many sources nowadays available. This is the reason why we came with the idea of Quasi-hybrid logic [5], which is a paraconsistent version of hybrid logic, thus capable of dealing with the inconsistencies in the information, written as formulas in hybrid logic. This kind of logic is useful for comparing the amount of inconsistency among databases, and has proved to be applicable in a wide range of real-life applications, namely we have studied how can inconsistencies relate to the health care flow of a patient [4], and we are currently working in robotics in order to create a robot which uses a paraconsistent reasoning to determine its movements and actions.

This work proposes to introduce proof-theoretical aspects of QH logic. We aimed to combine both tableaux for quasi-classical and hybrid logics, [3,8] respectively, which resulted in a new tableau system as desired.

Classically, tableau systems rely on a backwards reasoning where we start with a formula whose validity we want to prove. A tableau, *i.e.*, a tree, is created using some predefined rules, and whose starting point is the negation of the formula we are investigating. If we come to a point where each branch of the tree contains both a formula of the form φ and a formula of the form $\neg\varphi$, we

say that the tableau is closed and verify that there are no counter-models for the original formula, thus it is proved that the formula is valid.

In our paraconsistent setting, we will consider a database Δ, and a query ψ whose satisfiability will be verified in the bistructures (introduced in Sect. 3) that satisfy the formulas in the database. Analogously to the classical case, we start with Δ and ψ^*, where ψ^* will be defined later (in particular it will be a satisfaction statement). Our tableau is constructed using strong rules (which yield disjunctive syllogism) for formulas in Δ and a weaker version (which rejects DS) for ψ^*. If we end up with a tableau which is closed, *i.e.*, in which every branch has a formula of the form φ and φ^*, we can conclude that φ is true in every bistructure that satisfies Δ.

2   The Basic Hybrid Language

We start by presenting the simplest form of hybrid logic: the *basic hybrid language*, $\mathcal{H}(@)$. The basic hybrid language introduces nominals and the satisfaction operator into the propositional modal logic. Although being a simple extension, it carries great power in terms of expressivity.

Definition 1. *Let* $L = \langle \mathrm{Prop}, \mathrm{Nom} \rangle$ *be a* hybrid similarity type *where* Prop *is a set of* propositional symbols *and* Nom *is a set disjoint from* Prop. *We use* $p, q, r, etc.$ *to refer to the elements in* Prop. *The elements in* Nom *are called* nominals *and we typically write them as* $i, j, k, etc.$. *The set of well-formed formulas over L,* $\mathrm{Form}_@(L)$, *is defined by the following grammar:*

$$WFF := i \mid p \mid \neg\varphi \mid \varphi \vee \psi \mid \varphi \wedge \psi \mid \Diamond\varphi \mid \Box\varphi \mid @_i\varphi$$

For any nominal i, $@_i$ *is called a* satisfaction operator, *and for a formula* φ, $@_i\varphi$ *is called a* satisfaction statement.

Given a hybrid similarity type $L = \langle \mathrm{Prop}, \mathrm{Nom} \rangle$, a *hybrid structure* $\mathcal{H}$ over L is a tuple (W, R, N, V) such that:

- W is a non-empty set called *domain*, whose elements are called *states* or *worlds*.
- R is a binary relation on W and is called the *accessibility relation*.
- $N : \mathrm{Nom} \to W$ is a function called *hybrid nomination* that assigns nominals to elements in W. For any nominal i, $N(i)$ is the element of W named by i.
- V is a *hybrid valuation*, which means that V is a function with domain Prop and range $Pow(W)$ such that $V(p)$ tells us at which states (if any) each propositional symbol is true.

The pair (W, R) is called the *frame* underlying $\mathcal{H}$ and $\mathcal{H}$ is said to be a structure based on this frame.

The satisfaction relation, which is defined as follows, is a generalization of Kripke-style satisfaction.

Definition 2 (Satisfaction). *The relation of local satisfaction $\models$ between a hybrid structure $\mathcal{H} = (W, R, N, V)$, a state $w \in W$ and a hybrid formula is recursively defined by:*

1. $\mathcal{H}, w \models i$ *iff* $w = N(i)$;
2. $\mathcal{H}, w \models p$ *iff* $w \in V(p)$;
3. $\mathcal{H}, w \models \neg\varphi$ *iff it is false that* $\mathcal{H}, w \models \varphi$;
4. $\mathcal{H}, w \models \varphi \wedge \psi$ *iff* $\mathcal{H}, w \models \varphi$ *and* $\mathcal{H}, w \models \psi$;
5. $\mathcal{H}, w \models \varphi \vee \psi$ *iff* $\mathcal{H}, w \models \varphi$ *or* $\mathcal{H}, w \models \psi$;
6. $\mathcal{H}, w \models \Diamond\varphi$ *iff* $\exists w' \in W(wRw'$ *and* $\mathcal{H}, w' \models \varphi)$;
7. $\mathcal{H}, w \models \Box\varphi$ *iff* $\forall w' \in W(wRw' \Rightarrow \mathcal{H}, w' \models \varphi)$;
8. $\mathcal{H}, w \models @_i\varphi$ *iff* $\mathcal{H}, w' \models \varphi$, *where* $w' = N(i)$;

If $\mathcal{H}, w \models \varphi$ we say that φ is satisfied *in $\mathcal{H}$ at w. If φ is satisfied at all states in a structure $\mathcal{H}$, we write $\mathcal{H} \models \varphi$. If φ is satisfied at all states in all structures based on a frame $\mathcal{F}$, then we say that φ is* valid *on $\mathcal{F}$ and we write $\mathcal{F} \models \varphi$. If φ is valid on all frames, then we simply say that φ is* valid *and we write $\models \varphi$. For $\Delta \subseteq \text{Form}_@(L)$, we say that $\mathcal{H}$ is a* model *of Δ iff for all $\theta \in \Delta, \mathcal{H} \models \theta$.*

Definition 3. *A formula $\bar{\varphi} \in \text{Form}_@(L)$ is said to be* (logically) equivalent *to $\varphi \in \text{Form}_@(L)$ iff for every hybrid structure $\mathcal{H} = (W, R, N, V)$, for all $w \in W$,*

$$\mathcal{H}, w \models \varphi \Leftrightarrow \mathcal{H}, w \models \bar{\varphi}.$$

It is easy to see that boolean connectives have the usual properties, and that $\Box\varphi$ is equivalent to $\neg\Diamond\neg\varphi$.

We define the notion of negation normal form of a formula (*i.e.*, formulas in which the negation symbol occurs immediately before propositional symbols and/or nominals) for hybrid logic and we establish an analogous result to the one in [2] for classical propositional logic that states that any modal formula is logically equivalent to one in the negation normal form.

Definition 4. *Let $L = \langle \text{Prop}, \text{Nom} \rangle$ be a hybrid similarity type. A formula is said to be in* negation normal form, *for short* NNF, *if negation only appears directly before propositional variables and/or nominals. The set of* NNF *formulas over L, $\text{Form}_{\text{NNF}(@)}(L)$, is recursively defined as follows:*
For $p \in \text{Prop}, \ i \in \text{Nom}$,

1. $p, i, \neg p, \neg i$ *are in* NNF;
2. *If $\varphi, \ \psi$ are formulas in* NNF, *then $\varphi \vee \psi, \ \varphi \wedge \psi$ are in* NNF;
3. *If φ is in* NNF, *then $\Box\varphi, \ \Diamond\varphi$ are in* NNF;
4. *If φ is in* NNF, *then $@_i\varphi$ is in* NNF.

The next proposition shows that we do not lose generality by considering just formulas in negation normal form.

Proposition 1 [5]**.** *Every formula* $\varphi \in \mathrm{Form}_{@}(L)$ *is logically equivalent to a formula* $\bar{\varphi} \in \mathrm{Form}_{\mathrm{NNF}(@)}(L)$.

The *negation normal form* of a formula is defined just as in classical propositional logic. A recursive procedure that puts formulas in negation normal form $nnf : \mathrm{Form}_{@}(L) \rightarrow \mathrm{Form}_{\mathrm{NNF}(@)}(L)$, is set as usual. For example:

$$nnf(l) \stackrel{def}{=} l, \text{if } l \text{ is a literal,}$$

$$nnf(\neg(\psi_1 \wedge \psi_2)) \stackrel{def}{=} nnf(\neg\psi_1) \vee nnf(\neg\psi_2) \quad \text{and} \quad nnf(\neg\Box\psi) \stackrel{def}{=} \Diamond nnf(\neg\psi)$$

3   Paraconsistency in Hybrid Logic

In this section we study paraconsistency in Hybrid logic following an approach inspired by the work of Grant and Hunter ([6,7]).

First of all, we define a *Quasi-hybrid (*QH*) Basic Logic*. The assumption in [6] is that all formulas are in *Prenex Conjunctive Normal Form*; in QH logic we will assume henceforth that all formulas are in *Negation Normal Form*. This assumption does not lead to loss of generality since any hybrid formula is equivalent to a formula in negation normal form (cf. Proposition 1).

Next, concepts of bistructure, decoupled and strong satisfaction and QH model will be presented. We define the paraconsistent diagram of a bistructure.

3.1   Quasi-Hybrid Basic Logic

As already mentioned, we will assume that all formulas are in negation normal form, *i.e.*, given a hybrid similarity type $L = \langle \mathrm{Prop}, \mathrm{Nom} \rangle$, the set of formulas is $\mathrm{Form}_{\mathrm{NNF}(@)}(L)$.

Definition 5. *Let* θ *be a formula in* NNF. *We define the* complementation operation $\sim$ *from* $\sim \theta := nnf(\neg\theta)$.

The $\sim$ operator is not part of the object hybrid similarity type but it makes some definitions clearer.

Recall that a *hybrid structure* for a hybrid similarity type L is a tuple (W, R, N, V). However, in order to accommodate contradictions in a model, we will use two valuations for propositions: V^+ and V^-.

Definition 6. *A* hybrid bistructure *is a tuple* (W, R, N, V^+, V^-), *where* (W, R, N, V^+) *and* (W, R, N, V^-) *are hybrid structures.*

The map V^+ is interpreted as the acceptance of a propositional symbol, and V^- as the rejection. This is formalized in the definition for decoupled satisfaction.

Definition 7. *For a hybrid bistructure* $E = (W, R, N, V^+, V^-)$ *we define a satisfiability relation* $\models_d$ *called* decoupled satisfaction *at* $w \in W$ *for propositional symbols and nominals as follows:*

1. $E, w \models_d p$ *iff* $w \in V^+(p)$;
2. $E, w \models_d i$ *iff* $w = N(i)$;
3. $E, w \models_d \neg p$ *iff* $w \in V^-(p)$;
4. $E, w \models_d \neg i$ *iff* $w \neq N(i)$;

Since we allow both a propositional symbol and its negation to be simultaneously satisfied and also allow both to be non-satisfied, we have decoupled, at the level of the structure, the link between a formula and its complement. In contrast, if a classical hybrid structure satisfies a propositional symbol at some world, it is forced to not satisfy its complement at that world.

This decoupling gives us the basis for a semantics for paraconsistent reasoning. Paraconsistency involves a tradeoff; in order to allow contradictions, one of the following three principles must be abandoned: disjunction introduction, disjunctive syllogism, and transitivity. In this approach, we chose to keep the disjunctive syllogism and transitivity and discard disjunction introduction.

In Quasi-hybrid logic, "*or*" statements involve an intensional disjunction. Such a disjunction is one whose satisfaction entails not merely that at least one of the disjuncts is the case, but also that if one of the disjuncts were not the case, then the other one would be the case.

Definition 8. *A satisfiability relation* $\models_s$ *called* strong satisfaction, *is defined as follows:*

1. $E, w \models_s p$ *iff* $E, w \models_d p$;
2. $E, w \models_s \neg p$ *iff* $E, w \models_d \neg p$;
3. $E, w \models_s i$ *iff* $E, w \models_d i$;
4. $E, w \models_s \neg i$ *iff* $E, w \models_d \neg i$;
5. $E, w \models_s \theta_1 \vee \theta_2$ *iff* $[E, w \models_s \theta_1$ *or* $E, w \models_s \theta_2]$ *and* $[E, w \models_s \sim \theta_1 \Rightarrow E, w \models_s \theta_2]$ *and* $[E, w \models_s \sim \theta_2 \Rightarrow E, w \models_s \theta_1]$;
6. $E, w \models_s \theta_1 \wedge \theta_2$ *iff* $E, w \models_s \theta_1$ *and* $E, w \models_s \theta_2$;
7. $E, w \models_s \Diamond \theta$ *iff* $\exists w'(wRw' \ \& \ E, w' \models_s \theta)$;
8. $E, w \models_s \Box \theta$ *iff* $\forall w'(wRw' \Rightarrow E, w' \models_s \theta)$;
9. $E, w \models_s @_i \theta$ *iff* $E, w' \models_s \theta$ *where* $w' = N(i)$;

We define *strong validity* as follows: $E \models_s \theta$ iff for all $w \in W$, $E, w \models_s \theta$.

We say that E is a *quasi-hybrid model* of Δ iff for all $\theta \in \Delta, E \models_s \theta$ and we write $E \models_s \Delta$.

4   A Tableau for Quasi-Hybrid Logic

In this section we discuss a decision procedure for Quasi-hybrid logic, based on a tableau system. This new tableau system is a fusion between the tableau system for Quasi-classical logic introduced in [8], and the tableau system for Hybrid logic proposed in [3].

We will consider a database Δ of hybrid formulas that express real situations where inconsistencies may appear at some states, and we will check if a query φ is a consequence of the database, *i.e.*, we will want to check if every bistructure that strongly validates all formulas in Δ also validates φ weakly.

We will restrict our attention to formulas which are satisfaction statements.

4.1   Tableau Rules for QH Logic

We will start by introducing some definitions that will be useful later when we explain the construction of the tableau. We present the rules and a theorem for checking soundness.

Definition 9. *We define* weak satisfaction $\models_w$ *as strong satisfaction* ($\models_s$), *except for the case of disjunction, which we will consider as a classical disjunction:*

$$E, w \models_w \theta_1 \vee \theta_2 \ iff \ E, w \models_w \theta_1 \ or \ E, w \models_w \theta_2$$

The reader may observe that for any $\theta \in \mathrm{Form}_{\mathrm{NNF}(@)}(L)$, $E, w \models_s \theta$ implies $E, w \models_w \theta$. And that, by contraposition, $\not\models_w \subseteq \not\models_s$.

Similarly to the definition of strong validity, we define *weak validity* as follows: $E \models_w \theta$ iff for all $w \in W$, $E, w \models_w \theta$.

From now on, we will restrict our attention to satisfaction statements.

Definition 10 (Quasi-Hybrid Consequence Relation). *Let Δ be a set of satisfaction statements called* database*, and φ be a satisfaction statement, called* query*. We say that φ is a* consequence *of Δ in quasi-hybrid logic* if and only if*, for all bistructures E which are quasi-hybrid models of Δ, φ is weakly valid.*

Formally,

$$\Delta \models_{\mathrm{QH}} \varphi \ iff \ \forall E \, (E \models_s \Delta \Rightarrow E \models_w \varphi)$$

Before introducing the tableau-based proof procedure, some definitions are required, namely:

Definition 11. *Given a hybrid similarity type $L = \langle \mathrm{Prop}, \mathrm{Nom} \rangle$, we denote the set of satisfaction statements over L as $L_@$.*

We duplicate the set of satisfaction statements by considering starred copies. The extended set is denoted by $L_@^*$ and is defined as: $L_@^* = L_@ \cup \{\varphi^* \mid \varphi \in L_@\}$.

The satisfaction of the new formulas φ^* is in some sense the complementary of the satisfaction of φ.

Definition 12. *We extend both weak and strong satisfaction relations to starred formulas as follows:*

$E, w \models_s \varphi^* \ iff \ E, w \not\models_s \varphi$
$E, w \models_w \varphi^* \ iff \ E, w \not\models_w \varphi$

Weak and strong validity of starred formulas are defined in the natural way.

We can now introduce two types of decomposition rules to be used in our QH semantic tableau: *strong rules* and *weak rules*, strong rules are applied to non-starred formulas, and weak rules are applied to starred formulas. The tableau system will be denoted T_{QH}.

Tableau Rules:

Strong rules (S-rules)

- For connectives and operators:

$$\frac{@_i(\alpha \vee \beta)}{(@_i(\sim \alpha))^* \mid @_i\beta} \ (\vee_1) \qquad \frac{@_i(\alpha \vee \beta)}{(@_i(\sim \beta))^* \mid @_i\alpha} \ (\vee_2) \qquad \frac{@_i(\alpha \vee \beta)}{@_i\alpha \mid @_i\beta} \ (\vee_3)$$

$$\frac{@_i(\alpha \wedge \beta)}{@_i\alpha, @_i\beta} \ (\wedge) \qquad \frac{@_i@_j\alpha}{@_j\alpha} \ (@) \qquad \frac{@_i\Box\alpha, @_i\Diamond t}{@_t\alpha} \ (\Box)$$

$$\frac{@_i\Diamond\alpha}{@_i\Diamond t, @_t\alpha} \ (\Diamond)(\mathrm{i})$$

- For nominals:

$$\frac{}{@_i i} \ (Ref)(\mathrm{ii}) \qquad \frac{@_a c, @_a\varphi}{@_c\varphi} \ (Nom_1)(\mathrm{iii}) \qquad \frac{@_a c, @_a\Diamond b}{@_c\Diamond b} \ (Nom_2)$$

Weak rules (W-rules)

- For connectives and operators:

$$\frac{(@_i(\alpha \vee \beta))^*}{(@_i\alpha)^*, (@_i\beta)^*} \ (\vee^*) \qquad \frac{(@_i(\alpha \wedge \beta))^*}{(@_i\alpha)^* \mid (@_i\beta)^*} \ (\wedge^*)$$

$$\frac{(@_i@_j\alpha)^*}{(@_j\alpha)^*} \ (@^*) \qquad \frac{(@_i\Box\alpha)^*}{@_i\Diamond t, (@_t\alpha)^*} \ (\Box^*)(\mathrm{iv}) \qquad \frac{(@_i\Diamond\alpha)^*, @_i\Diamond t}{(@_t\alpha)^*} \ (\Diamond^*)$$

$$\frac{(@_i\Box\neg t)^*}{@_i\Diamond t} \ (\Box^*_{\neg i})$$

- For nominals:

$$\frac{@_a c, (@_a\varphi)^*}{(@_c\varphi)^*} \ (Nom_1^*)(\mathrm{iii}) \qquad \frac{@_a c, (@_a\Box b)^*}{(@_c\Box b)^*} \ (Nom_2^*)$$

(i) t is a new nominal, α is not a nominal.
(ii) for i in the branch.
(iii) for φ a propositional variable/nominal.
(iv) t is a new nominal, α is not of the form $\neg j$, for j a nominal.

The strong and weak rules for nominals, together with $(\vee_1), (\vee_2), (\Box)$ and $(\Diamond^*)$ are called *non-destructive rules*. The remaining are called *destructive*.

The star in the formulas can be seen as a kind of meta-negation; the weak rules, which involve starred formulas, can thus be viewed as duals of the strong ones, except for the case where we obtain the classical disjunction.

Next theorem states that T_{QH} is sound:

Theorem 1 (Soundness). *The tableau rules are sound in the following sense:*

- *for any r-rule* $\dfrac{\Lambda}{\Sigma}$, *any bistructure E and any state* $w \in W$, $E, w \models_r \Lambda$ *implies* $E, w \models_r \Sigma$.
- *for any r-rule* $\dfrac{\Lambda}{\Sigma \mid \Gamma}$, *any bistructure E and any state* $w \in W$, $E, w \models_r \Lambda$ *implies* $E, w \models_r \Sigma$ *or* $E, w \models_r \Gamma$,

for Λ; Σ *and* Γ *lists of formulas in* $L_@^*$ *and* $r \in \{s, \text{w}\}$.

Proof. The proof can be obtained by checking each rule.

4.2   Properties of the Tableau System and its Construction

The idea of a tableau is that we apply the rules previously introduced to root formulas and to formulas which occur in the tableau after the application of a rule. This indiscriminate way of applying rules leads to infinite tableaux, where there may be repeated formulas and loops, thus we must find a systematic construction that terminates a tableau and allows us to take some conclusions from it.

Definition 13. *We say that a formula* $\chi \in L_@^*$ *is a* strong occurrence/s-occurs *if it is the result of applying a strong rule. Analogously we say that* χ *is a* weak occurrence/w-occurs *if it is the result of applying a weak rule. A formula occurs if it s-occurs or w-occurs.*

Definition 14. *The notion of a* subformula *is defined by the following conditions:*

- φ *is a subformula of* φ;
- *if* $\psi \wedge \theta$ *or* $\psi \vee \theta$ *is a subformula of* φ, *then so are* ψ *and* θ;
- *if* $@_a\psi$, $\Box\psi$, *or* $\Diamond\psi$ *is a subformula of* φ, *then so is* ψ.

The tableau system T_{QH} satisfies the following quasi-subformula property:

Theorem 2 (Quasi-Subformula Property). *If a formula* $@_a\varphi$ *s-occurs in a tableau where* φ *is not a nominal and* φ *is not of the form* $\Diamond b$, *then* φ *is a subformula of a root formula. If a formula* $(@_a\varphi)^*$ *w-occurs in a tableau, then* φ *is a subformula of the premise in the applied rule.*

Proof. The proof can be obtained by checking each rule.

Definition 15. *Let* Θ *be a branch of a tableau and let* Nom^Θ *be the set of nominals occurring in the formulas of* Θ. *Define a binary relation* $\sim_\Theta$ *on* Nom^Θ *by* $a \sim_\Theta b$ *if and only if the formula* $@_ab$ *occurs on* Θ.

Definition 16. *Let* b *and* a *be nominals occurring on a branch* Θ *of a tableau in* T_{QH}. *The nominal* a *is said to be* included in the nominal b *with respect to* Θ *if the following holds:*

– *for any subformula φ of a root formula, if the $@_a\varphi$ s-occurs on Θ, then $@_b\varphi$ also s-occurs on Θ; and*
– *if $(@_a\varphi)^*$ w-occurs on Θ, then $(@_b\varphi)^*$ also w-occurs on Θ.*

If a is included in b with respect to Θ, and the first occurrence of b on Θ is before the first occurrence of a, then we write $a \sqsubseteq_\Theta b$.

Definition 17 (Tableau Construction). *Given a database Δ of satisfaction statements and a query $@_a\varphi$ of QH, one wants to verify if $@_a\varphi$ is a consequence of Δ. In order to do so, we define by induction a sequence $\tau_0, \tau_1, \tau_2, \cdots$ of finite tableaux in T_{QH}, each of which is embedded in its successor.*

Let τ_0 be the finite tableau constituted by the formulas in Δ and $(@_a\varphi)^*$. τ_{n+1} is obtained from τ_n if it is possible to apply an arbitrary rule to τ_n with the following three restrictions:

1. *If a formula to be added to a branch by applying a rule already occurs on the branch, then the addition of the formula is simply omitted.*
2. *After the application of a destructive rule to a formula occurrence φ on a branch, it is recorded that the rule was applied to φ with respect to the branch and the rule will not again be applied to φ with respect to the branch or any extension of it.*
3. *The existential rules $(\Diamond, \Box^*)$ are not applied to a formula occurrence $@_a\Diamond\varphi$ or $(@_a\Box\varphi)^*$ on a branch Θ if there exists a nominal b such that $a \sqsubseteq_\Theta b$.*

Note that due to the first restriction, a formula cannot occur more than once on a branch. Also note that no information is recorded about applications of nondestructive rules. The conditions on applications of the existential rules $(\Diamond, \Box^*)$ in the third restriction are the loop-check conditions. The intuition behind loopchecks is that an existential rule is not applied in a world if the information in that world can be found already in an ancestor world. Hence, the introduction of a new world by the existential rule is blocked.

A branch is *closed* iff there is a formula ψ for which ψ and ψ^* are in that branch; we use the symbol $\times$ to mark a closed branch. A T_{QH} tableau is *closed* iff every branch is closed. A branch is *open* if it is not closed and there are no more rules to apply; we use the symbol $\odot$ to mark an open branch. A tableau is *open* if it has an open branch.

A *terminal tableau* is a tableau where the rules have been exhaustively used *i.e.*, there are no more rules applicable to the tableau obeying the restrictions in Definition 17.

Henceforth, θ is a branch of a terminal tableau.

Definition 18. *Let U be the subset of Nom^Θ containing any nominal a having the property that there is no nominal b such that $a \sqsubseteq_\Theta b$. Let $\approx$ be the restriction of $\sim_\Theta$ to U.*

Note that U contains all nominals present in the root formulas since they are the first formulas of the branch Θ. Observe that Θ is closed under the rules (Ref) and (Nom1), so both $\sim_\Theta$ and $\approx$ are equivalence relations.

Given a nominal a in U, we let $[a]_\approx$ denote the equivalence class of a with respect to $\approx$ and we let $U/\approx$ denote the set of equivalence classes.

Definition 19. *Let R be the binary relation on U defined by aRc if and only if there exists a nominal $c' \approx c$ such that one of the following two conditions is satisfied:*

1. *The formula $@_a \Diamond c'$ occurs on Θ.*
2. *There exists a nominal d in Nom^Θ such that the formula $@_a \Diamond d$ occurs on Θ and $d \subseteq_\Theta c'$.*

Note that the nominal d referred to in the second item in the definition is not an element of U. It follows from Θ being closed under the rule (Nom2) that R is compatible with $\approx$ in the first argument and it is trivial that R is compatible with $\approx$ in the second argument. We let $\bar{R}$ be the binary relation on $U/\approx$ defined by $[a]_\approx \bar{R} [c]_\approx$ if and only if aRc.

Definition 20. *Let $\bar{N} : U \to U/\approx$ be defined as $\bar{N}(a) = [a]_\approx$.*

Definition 21. *Let V^+ be the function that to each ordinary propositional symbol assigns the set of elements of U where that propositional variable occurs, i.e., $a \in V^+(p)$ iff $@_a p$ occurs on Θ. Analogously, let V^- be the function that to each ordinary propositional symbol assigns the set of elements of U where the negation of that propositional variable occurs, i.e., $a \in V^-(p)$ iff $@_a \neg p$ occurs on Θ.*

We let $V_\approx^+$ be defined by $V_\approx^+(p) = \{[a]_\approx \mid a \in V^+(p)\}$. We define $V_\approx^-$ analogously: $V_\approx^-(p) = \{[a]_\approx \mid a \in V^-(p)\}$.

Given a branch Θ, let $M^\Theta = (U/\approx, \bar{R}, \bar{N}, V_\approx^+, V_\approx^-)$. We will omit the reference to the branch in M^Θ if it is clear from the context.

Theorem 3 (Model Existence). *Assume that the branch Θ is open. For any satisfaction statement $@_a \varphi$ which contains only nominals from U, the following conditions hold:*

(i) *If $@_a \varphi$ s-occurs on Θ, then $M, [a]_\approx \models_s \varphi$*
(ii) *If $@_a \varphi$ w-occurs on Θ, then $M, [a]_\approx \models_w \varphi$*
(iii) *If $(@_a \varphi)^*$ s-occurs on Θ, then $M, [a]_\approx \not\models_s \varphi$.*
(iv) *If $(@_a \varphi)^*$ w-occurs on Θ, then $M, [a]_\approx \not\models_w \varphi$.*

Proof. The proof is by induction on the structure of φ.

– $\varphi = i$, i a nominal
(i) $@_a i$ s-occurs on Θ, then $[a]_\approx = [i]_\approx$, hence $M, [a]_\approx \models_s i$.
(ii) $@_a i$ never w-occurs.
(iii) $(@_a i)^*$ s-occurs on Θ, then, since the branch is open, $@_a i$ does not occur on Θ, in particular, it does not s-occur. So, $[a]_\approx \neq [i]_\approx$, then $M, [a]_\approx \not\models_s i$.
(iv) $(@_a i)^*$ w-occurs on Θ, then analogously to the previous case, $@_a i$ does not occur on Θ, in particular, it does not w-occur. So, $[a]_\approx \neq [i]_\approx$, then $M, [a]_\approx \not\models_w i$.

– $\varphi = \neg i$, i a nominal
The proof is analogous to the case $\varphi = i$.

– $\varphi = p$, p a propositional variable
(i) $@_a p$ s-occurs on Θ, then $a \in V_{\approx}^{+}(p)$, thus $M, [a]_{\approx} \models_s p$.
(ii) $@_a p$ never w-occurs.
(iii) $(@_a p)^*$ s-occurs on Θ, then, since the branch is open, $@_a p$ does not occur on Θ, in particular, it does not s-occur. It also means that $a \notin V_{\approx}^{+}(p)$ so, $M, [a]_{\approx} \not\models_s p$.
(iv) $(@_a p)^*$ w-occurs on Θ; analogously to the previous case, considering that $@_a p$ does not w-occur we get that $M, [a]_{\approx} \not\models_{\mathrm{w}} p$.

– $\varphi = \neg p$, p a propositional variable
Each case will be analogous to the one in $\varphi = p$, only with the difference that we now consider $V_{\approx}^{-}(p)$ instead of $V_{\approx}^{+}(p)$.

– $\varphi = \phi \wedge \psi$, ϕ, ψ formulas
(i) $@_a(\phi \wedge \psi)$ s-occurs on Θ, then, by applying the S-rule $(\wedge)$, $@_a\phi$ and $@_a\psi$ s-occur on Θ. By the induction hypothesis, $M, [a]_{\approx} \models_s \phi$ and $M, [a]_{\approx} \models_s \psi$. Thus, $M, [a]_{\approx} \models_s \phi \wedge \psi$.
(ii) $@_a(\phi \wedge \psi)$ never w-occurs.
(iii) $(@_a(\phi \wedge \psi))^*$ s-occurs on Θ, then, by applying the W-rule $(\wedge^*)$, $(@_a\phi)^*$ or $(@_a\psi)^*$ w-occur on Θ. Hence, by induction hypothesis, $M, [a]_{\approx} \not\models_{\mathrm{w}} \phi$ or $M, [a]_{\approx} \not\models_{\mathrm{w}} \psi$. Thus, $M, [a]_{\approx} \not\models_{\mathrm{w}} \phi \wedge \psi$. Therefore, $M, [a]_{\approx} \not\models_s \phi \wedge \psi$.
(iv) $(@_a(\phi \wedge \psi))^*$ w-occurs on Θ; follow an analogous approach to (iii).

– $\varphi = @_i\phi$, ϕ a formula
(i) $@_a@_i\phi$ s-occurs on Θ, then, by applying the S-rule $(@)$, $@_i\phi$ s-occurs on Θ. By the induction hypothesis, $M, [i]_{\approx} \models_s \phi$; thus, by the definition of satisfiability, $M, [a]_{\approx} \models_s @_i\phi$.
(ii) $@_a@_i\phi$ never w-occurs.
(iii) $(@_a@_i\phi)^*$ s-occurs on Θ, then, by applying the W-rule $(@^*)$, $(@_i\phi)^*$ w-occurs on Θ. By the induction hypothesis, $M, [i]_{\approx} \not\models_{\mathrm{w}} \phi$. Thus, $M, [i]_{\approx} \not\models_s \phi$ and it follows that $M, [a]_{\approx} \not\models_s @_i\phi$.
(iv) $(@_a@_i\phi)^*$ w-occurs on Θ; follow an analogous approach to (iii).

– $\varphi = \phi \vee \psi$, ϕ, ψ formulas
(i) $@_a(\phi \vee \psi)$ s-occurs on Θ, then, since we can apply three S-rules, namely $(\vee_1, \vee_2, \vee_3)$, one may obtain 8 new branches, represented as follows:
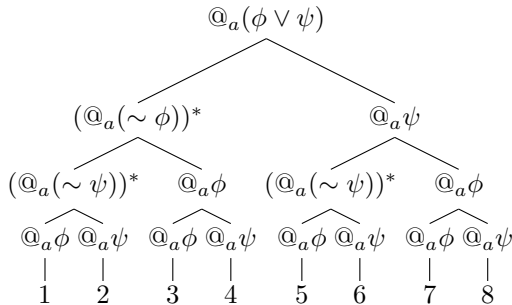
Let us check what happens at each branch:

1. $(@_a(\sim \phi))^*$, $(@_a(\sim \psi))^*$ and $@_a\phi$ s-occur.
   By induction hypothesis:
   $M, [a]_\approx \not\models_s \sim \phi$,
   $M, [a]_\approx \not\models_s \sim \psi$, and
   $M, [a]_\approx \models_s \phi$.

   Recall that $M, [a]_\approx \models_s \phi \vee \psi$
   $\Leftrightarrow (M, [a]_\approx \models_s \phi \text{ or } M, [a]_\approx \models_s \psi)$
   and $(M, [a]_\approx \models_s \sim \phi \Rightarrow M, [a]_\approx \models_s \psi)$
   and $(M, [a]_\approx \models_s \sim \psi \Rightarrow M, [a]_\approx \models_s \phi)$
   Thus, under the conditions in branch number 1, it is verified that
   $M, [a]_\approx \models_s \phi \vee \psi$.

2. $(@_a(\sim \phi))^*$, $(@_a(\sim \psi))^*$ and $@_a\psi$ s-occur.
   Thus,
   $M, [a]_\approx \not\models_s \sim \phi$,
   $M, [a]_\approx \not\models_s \sim \psi$, and
   $M, [a]_\approx \models_s \psi$.
   Using the same approach as before, one has that $M, [a]_\approx \models_s \phi \vee \psi$.

3. $(@_a(\sim \phi))^*$, and $@_a\phi$ s-occur.
   Thus,
   $M, [a]_\approx \not\models_s \sim \phi$, and
   $M, [a]_\approx \models_s \phi$.

   Then, $M, [a]_\approx \models_s \phi \vee \psi$.

4. $(@_a(\sim \phi))^*$, $@_a\phi$ and $@_a\psi$ s-occur.
   Thus,
   $M, [a]_\approx \not\models_s \sim \phi$,
   $M, [a]_\approx \models_s \phi$, and
   $M, [a]_\approx \models_s \psi$.

   So, $M, [a]_\approx \models_s \phi \vee \psi$.

5. $@_a\psi$, $(@_a(\sim \psi))^*$ and $@_a\phi$ s-occur.
   Analogously to branch number 4, $M, [a]_\approx \models_s \phi \vee \psi$.

6. $@_a\psi$ and $(@_a(\sim \psi))^*$ s-occur.
   Analogously to branch number 3, $M, [a]_\approx \models_s \phi \vee \psi$.

7. and 8. $@_a\phi$ and $@_a\psi$ s-occur.
   Thus, $M, [a]_\approx \models_s \phi \vee \psi$.

We can thus conclude that, if $@_a(\phi \vee \psi)$ s-occurs, then $M, [a]_\approx \models_s \phi \vee \psi$.

(ii) $@_a(\phi \vee \psi)$ never w-occurs.

(iii) $(@_a(\phi \vee \psi))^*$ s-occurs on Θ, then by applying rule $(\vee^*)$, we obtain that $(@_a\phi)^*$ and $(@_a\psi)^*$ w-occur. By induction hypothesis, $M, [a]_\approx \not\models_w \phi$ and $M, [a]_\approx \not\models_w \psi$; thus, $M, [a]_\approx \not\models_w \phi \vee \psi$. Therefore $M, [a]_\approx \not\models_s \phi \vee \psi$.

(iv) $(@_a(\phi \vee \psi))^*$ w-occurs on Θ, analogous to the previous case.

- $\varphi = \Diamond\psi$, where:
  - ψ is a nominal t

    For the first two cases we have that:

    (i) $@_a\Diamond t$ s-occurs on Θ, then $[a]_\approx\bar{R}[t]_\approx$. By definition of satisfiability, $M, [a]_\approx \models_s \Diamond t$.

    (ii) $@_a\Diamond t$ w-occurs on Θ; it follows from the previous explanation that $M, [a]_\approx \models_w \Diamond t$.

  - ψ is not a nominal

    For the first two cases we have that:

    (i) $@_a\Diamond\psi$ s-occurs on Θ, then by the application of the S-rule $(\Diamond)$, $@_a\Diamond t$ and $@_t\psi$ s-occur, for t a new nominal.
      * if $t \in U$, $[a]_\approx\bar{R}[t]_\approx$ and $M, [a]_\approx \models_s \Diamond\psi$.
      * if $t \notin U$, $\exists d$ such that $t \subseteq_\Theta d$. Assume that there is no e such that $d \subseteq_\Theta e$, *i.e.*, $d \in U$. By Theorem 2, the formula ψ is a subformula of a root formula, so $@_d\psi$ s-occurs on Θ. By induction hypothesis, $M, [d]_\approx \models_s \psi$ and $[a]_\approx\bar{R}[d]_\approx$. Thus yielding that $M, [a]_\approx \models_s \Diamond\psi$.

    (ii) $@_a\Diamond\psi$ never w-occurs.

  - The last two cases are applicable for either ψ a nominal or not:

    (iii) $(@_a\Diamond\psi)^*$ s-occurs on Θ.

    We want to prove that $M, [a]_\approx \not\models_s \Diamond\psi$, *i.e.*, that for all $[c]_\approx$ such that $[a]_\approx\bar{R}[c]_\approx$, $M, [c]_\approx \not\models_s \psi$.

    By definition, $[a]_\approx\bar{R}[c]_\approx$ implies that $\exists c' : c' \approx c$ that satisfies one of the following two conditions:
      * $@_a\Diamond c'$ occurs.

        Then, by the W-rule $(\Diamond^*)$, $(@_{c'}\psi)^*$ w-occurs on Θ. By induction, $M, [c']_\approx \not\models_w \psi$.

        Since $[c']_\approx = [c]_\approx$, then $[a]_\approx\bar{R}[c']_\approx$. Thus, $M, [a]_\approx \not\models_w \Diamond\psi$, which implies that $M, [a]_\approx \not\models_s \Diamond\psi$.

      * $\exists d \in \text{Nom}^\Theta$ such that $@_a\Diamond d$ occurs and $d \subseteq_\Theta c'$.

        By the application of the W-rule $(\Diamond^*)$, $(@_d\psi)^*$ occurs on Θ. By Theorem 2, ψ is a subformula of the premise in the applied rule.

        Since $d \subseteq_\Theta c'$, $(@_{c'}\psi)^*$ occurs on Θ.

        By induction we conclude that $M, [c']_\approx \not\models_w \psi$, thus $M, [a]_\approx \not\models_w \Diamond\psi$ and finally, $M, [a]_\approx \not\models_s \Diamond\psi$.

    (iv) $(@_a\Diamond\psi)^*$ w-occurs on Θ. The proof is the same as the previous one, stopping at the point where $M, [a]_\approx \not\models_w \Diamond\psi$.

– $\varphi = \Box\psi$

In order to prove the first two cases, consider ψ an arbitrary formula.

(i) $@_a\Box\psi$ s-occurs on Θ.

We want to prove that $M, [a]_\approx \models_s \Box\psi$, $i.e.$, that for all $[c]_\approx$ such that $[a]_\approx \bar{R}[c]_\approx$, $M, [c]_\approx \models_s \psi$.

As you may verify, this is a similar proof to the one made for $(@_a\Diamond\psi)^*$:

By definition, $[a]_\approx \bar{R}[c]_\approx$ implies that $\exists c' : c' \approx c$ that satisfies one of the following two conditions:

- $@_a\Diamond c'$ occurs.
  Then, by the S-rule $(\Box)$, $@_{c'}\psi$ s-occurs on Θ. By induction, $M, [c']_\approx \models_s \psi$. Since $[c']_\approx = [c]_\approx$, then $[a]_\approx \bar{R}[c']_\approx$. Thus, $M, [a]_\approx \models_s \Box\psi$.

- $\exists d \in \text{Nom}^\Theta$ such that $_a\Diamond d$ occurs and $d \subseteq_\Theta c'$.
  By the application of the S-rule $(\Box)$, $@_d\psi$ s-occurs on Θ. By Theorem 2, ψ is a subformula.
  Since $d \subseteq_\Theta c'$, $@_{c'}\psi$ occurs on Θ.
  By induction we conclude that $M, [c']_\approx \models_s \psi$, thus $M, [a]_\approx \models_s \Box\psi$.

(ii) $@_a\Box\psi$ never w-occurs.

The last two cases require a separation between ψ of the form $\neg i$, with i a nominal, and not of the form described.

Let us consider:

- ψ of the form $\neg i$, with i a nominal.
  (iii) $(@_a\Box\neg i)^*$ s-occurs on Θ, thus, by the rule $(\Box^*_{\neg i})$ which implies that $(@_a\Diamond i)$ w-occurs.
  An occurrence of $(@_a\Diamond i)$ means that $[a]_\approx \bar{R}[i]_\approx$. So, by definition, $M, [a]_\approx \models_w \Diamond i$, which entails that $M, [a]_\approx \not\models_w \Box\neg i$. Therefore, $M, [a]_\approx \not\models_s \Box\neg i$.
  (iv) $(@_a\Box\neg i)^*$ w-occurs on Θ follows the same approach.

- ψ otherwise.
  (iii) $(@_a\Box\psi)^*$ s-occurs on Θ, thus by application of the W-rule $(\Box^*)$, $@_a\Diamond t$ and $(@_t\psi)^*$ w-occur, for t a new nominal.
      * if $t \in U$, $[a]_\approx \bar{R}[t]_\approx$. By induction hypothesis, $M, [t]_\approx \not\models_w \psi$. Thus $M, [a]_\approx \not\models_w \Box\psi$. To conclude, $M, [a]_\approx \not\models_s \Box\psi$.
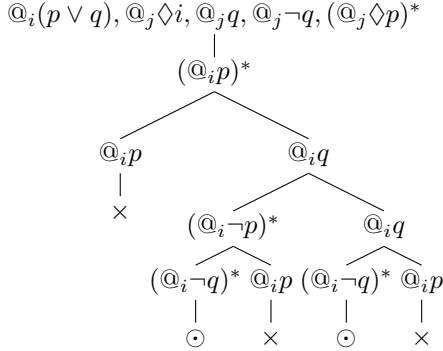   * if $t \notin U$, $\exists d$ such that $t \subseteq_\Theta d$. Assume that there is no e such that $d \subseteq_\Theta e$, $i.e.$, $d \in U$. By Theorem 2, the formula ψ is a subformula of a root formula, so $(@_d\psi)^*$ w-occurs on Θ. By induction hypothesis, $M, [d]_\approx \not\models_w \psi$ and $[a]_\approx \bar{R}[d]_\approx$. Thus yielding that $M, [a]_\approx \not\models_w \Box\psi$, which, in its turn, yields $M, [a]_\approx \not\models_s \Box\psi$.
  (iv) $(@_a\Box\psi)^*$ s-occurs on Θ; the proof is the same as before, stopping at $M, [a]_\approx \not\models_w \Box\psi$, which already gives the desired result.

From this theorem together with the soundness theorem we have the following decision procedure:

Decision Procedure: *Given a database Δ and a query $@_a\varphi$ whose consequence from Δ we want to decide, let τ_n be a terminal tableau generated by the tableau construction algorithm. If the tableau is closed, then $@_a\varphi$ is a consequence of Δ. Analogously, if the tableau is open, then $@_a\varphi$ is not a consequence of Δ.*
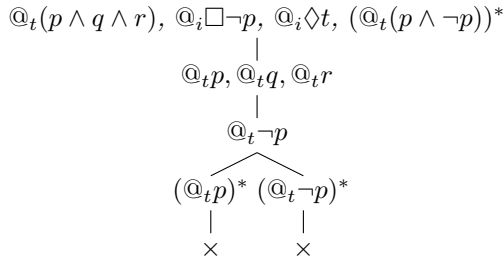
Example 1. *Let $\Delta = \{@_i(p \vee q), @_j\Diamond i, @_j q, @_j \neg q\}$ be a database and consider a query $\varphi = @_j \Diamond p$. Let us decide if φ is a consequence of Δ using the tableau procedure described:*

$$@_i(p \vee q), @_j\Diamond i, @_j q, @_j\neg q, (@_j\Diamond p)^*$$
$$|$$
$$(@_i p)^*$$

$$@_i p \qquad\qquad @_i q$$
$$|$$
$$\times$$

$$(@_i\neg p)^* \qquad\qquad @_i q$$

$$(@_i\neg q)^* \quad @_i p \quad (@_i\neg q)^* \quad @_i p$$
$$| \qquad\quad | \qquad\quad | \qquad\quad |$$
$$\odot \qquad\quad \times \qquad\quad \odot \qquad\quad \times$$

Since the tableau has some open branches, it means that the tableau is open, thus φ is not a consequence of Δ. A counter-model is the bistructure with two worlds i and j where only $@_i q$, $@_j q$, $@_j\neg q$ and $@_j\Diamond i$, $@_i\Box\neg i$, $@_j\Box\neg j$ are satisfied, which clearly is a bistructure that satisfies Δ, but where φ is not true.

We can easily verify that the database has an inconsistency, as it contains both $@_j q$ and $@_j\neg q$. If we were to evaluate φ in hybrid logic (usual version, non-paraconsistent), then φ would trivially follow.

Example 2. *Let $\Delta = \{@_t(p \wedge q \wedge r), @_i\Box\neg p, @_i\Diamond t\}$ be a database and consider a query $\varphi = (@_t(p \wedge \neg p))$. Let us decide if φ is a consequence of Δ using the tableau procedure described:*

$$@_t(p \wedge q \wedge r), \; @_i\Box\neg p, \; @_i\Diamond t, \; (@_t(p \wedge \neg p))^*$$
$$|$$
$$@_t p, @_t q, @_t r$$
$$|$$
$$@_t\neg p$$

$$(@_t p)^* \quad (@_t\neg p)^*$$
$$| \qquad\qquad |$$
$$\times \qquad\qquad \times$$

Note that the database has an inconsistency and the query itself is inconsistent. However, from the tableau procedure we verify that, since it is closed, φ is a consequence of Δ.

These examples show the mechanism of the tableau construction in action. Both examples yield an inconsistency in the database, but it is possible to still reach sensible conclusions about the validity of the query, thanks to the paraconsistent aspect of the tableau system presented.

5 Conclusion

Paraconsistent logics have been around for some decades, and their importance is justified by the unavoidable occurrence of inconsistencies in data and knowledge management. Inconsistent information can appear everywhere, and for many reasons. Contradictory information may arise in systems which are safety critical, such as health systems, aviation systems and many others.

After introducing Quasi-hybrid logic [5], a paraconsistent version of hybrid logic capable of dealing with contradictory information under the form of hybrid formulas, the next step was the construction of a proof-theoretical system for this new logic and the study of properties such as completeness, decidability, *etc.*

The challenge here was the combination of two different tableau systems already existent in literature, one for Quasi-classical logic and another for Hybrid logic. We ended up with a tableau system for Quasi-hybrid logic, described in this work, which ends with some examples.

This is clearly a step forward in the study of QH logic, which we aim to continue exploring, namely by:

- investigating inconsistency at the level of nominals, by allowing two nominations: a positive one for nominals, and a negative one for the negation of nominals. This way, we are able to handle the possibility of receiving information of the form $@_i j$ and $@_i \neg j$;
- accounting for inconsistency in modalities, for example by allowing both $@_i \Diamond j$ and $@_i \Box \neg j$ to be true;
- studying paraconsistency in the context of strong Priorean logic;
- use a combination of all of the above to study reactive deontic logics and switch graphs, where the base accessibility relation changes when its edges are traversed, in the context of QH logic.

Paraconsistency has already been considered in the context of modal logic (e.g. [9,10]). It will be interesting to compare our system with the hybrid systems obtained by adding nominals and the satisfaction operator to those systems.

References

1. Blackburn, P.: Representation, reasoning, and relational structures: a hybrid logic manifesto. Log. J. IGPL **8**(3), 339–365 (2000)
2. Boolos, G.S., Burgess, J.P., Jeffrey, R.C.: Computability and Logic, 5th edn. Cambridge University Press, Cambridge (2007)
3. Braüner, T.: Hybrid Logic and Its Proof-Theory. Springer, Heidelberg (2010)
4. Costa, D., Martins, M.: Inconsistencies in health care knowledge. In: 2014 IEEE 16th International Conference on e-Health Networking, Applications and Services (Healthcom), pp. 37–42, October 2014
5. Costa, D., Martins, M.: Paraconsistency in hybrid logic. Accepted in the Journal of Logic and Computation (2016). http://sweet.ua.pt/martins/documentos/preprint_2014/phl14.pdf
6. Grant, J., Hunter, A.: Measuring inconsistency in knowledge bases. J. Intell. Inf. Syst. **27**(2), 159–184 (2006)
7. Grant, J., Hunter, A.: Analysing inconsistent first-order knowledge bases. Artif. Intell. **172**(8–9), 1064–1093 (2008)
8. Hunter, A.: A semantic tableau version of first-order quasi-classical logic. In: Benferhat, S., Besnard, P. (eds.) ECSQARU 2001. LNCS (LNAI), vol. 2143, pp. 544–555. Springer, Heidelberg (2001)
9. Odintsov, S.P., Wansing, H.: Modal logics with Belnapian truth values. J. Appl. Non-Class. Logics **20**(3), 279–304 (2010)
10. Rivieccio, U., Jung, A., Jansana, R.: Four-valued modal logic: Kripke semantics and duality. J. Logic Comput. (2015). http://logcom.oxfordjournals.org/citmgr?gca=logcom%3Bexv038v1. doi:10.1093/logcom/exv038