Mathematical Structures in Computer Science (2025), **35**, e14, pp. 1–42
doi:10.1017/S0960129525100170

CAMBRIDGE UNIVERSITY PRESS

PAPER

Paraconsistent transition structures: compositional principles and a modal logic

Juliana Cunha[1,2], Alexandre Madeira[1] and Luis S. Barbosa[2]

[1]CIDMA, University of Aveiro, Aveiro, Portugal and [2]HASLab INESC-TEC, University of Minho, Braga, Portugal
Corresponding author: Alexandre Madeira; Email: madeira@ua.pt

(Received 1 August 2023; revised 7 January 2025; accepted 15 May 2025)

Abstract

Often in Software Engineering, a modeling formalism has to support scenarios of inconsistency in which several requirements either reinforce or contradict each other. Paraconsistent transition systems are proposed in this paper as one such formalism: states evolve through two accessibility relations capturing weighted evidence of a transition or its absence, respectively. Their weights come, parametrically, from a residuated lattice. This paper explores both i) a category of these systems, and the corresponding compositional operators and ii) a modal logic to reason upon them. Furthermore, two notions of *crisp* and *graded* simulation and bisimulation are introduced in order to relate two paraconsistent transition systems. Finally, results of modal invariance, for specific subsets of formulas, are discussed.

Keywords: Paraconsistent transition systems; paraconsistent logic; modal logic; simulation; bisimulation

1. Introduction

In classical logic, propositions are ascribed to exactly one truth value: true or false. While this bivalent framework has been indispensable in numerous mathematical and logical applications, the inherent simplicity and rigidity may fall short in capturing the intricacies present in real-world scenarios. For instance, in Software Engineering, it is common to encounter application scenarios where requirements either reinforce or contradict each other. One such scenario comes from current practice in quantum computation in the context of NISQ (*Noisy Intermediate-Scale Quantum*) technology (Preskill 2018) in which levels of decoherence of quantum memory need to be articulated with the length of the circuits to assess program quality. Similar challenges emerge in AI applications and data engineering.

The foundation of our work lies in a recent paper (Cruz et al. 2022a), where the authors introduced a weighted transition systems that records, for each transition, a pair of weights, a positive and a negative one. Informally, the positive weight captures the degree of effectiveness (‘*presence*’), and the negative weight captures the degree of impossibility (‘*absence*’) of a transition. This allows models to capture both *vagueness*, whenever both weights sum less than 1, as usual for example, in fuzzy systems, and *inconsistency*, when their sum exceeds 1. This last feature motivates the qualifier *paraconsistent* borrowed from the work on paraconsistent logic (Jaśkowski 1969; da Costa et al. 2007), which accommodates inconsistency in a controlled way, treating inconsistent information as potentially informative. Such logics were originally developed in Latin America in the decades of 1950 and 1960, primarily through the influential works of F. Asenjo and Newton da Costa. Quickly, however, paraconsistent logic attracted attention in the international community

© The Author(s), 2025. Published by Cambridge University Press.

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

and the original scope of mathematical applications broadened out as witnessed in a well-known book emphasizing the engineering potential of paraconsistency (Akama 2016). A number of other applications have emerged to themes from robotics, quantum mechanics and quantum information theory in works by Abe et al. (2007), Chiara and Giuntini (2000), Agudelo and Carnielli (2010) and da Costa and Krause (2014).

Paraconsistent transition structures and their composition were further studied in the preliminary conference communication (Cruz et al. 2022b) that this paper extends. The research program in Cruz et al. (2022b) continued here in two directions. First a suitable notion of morphism for paraconsistent labeled transition systems (PLTS) was introduced leading to the definition of the corresponding category and its algebra. Then, notions of trace for PLTS were discussed, as well as, notions of *crisp* (i.e. bivalent) simulation and bisimulation to relate two paraconsistent transition systems.

This paper starts by revising this work, namely lifting the residuated structure underlying the (parametric) domain of weights to the twisted structure which enables joint computation of positive and negative weights. We adopt the framework outlined in Bou et al. (2009) and focus on many-valued (paraconsistent) modal logics defined over commutative integral residuated lattices over a set A of possible truth values. The motivation behind these systems is to combine modal and graded reasoning, with existing research on many-valued logic addressing problems related to decidability, complexity (Caicedo et al. 2017), and axiomatizability of a (many-valued) dynamic logic (Madeira et al. 2016). Further contributions such as (Badia et al. 2023) involve investigating frame definability in finitely valued modal logics and Marti and Metcalfe (2014) delves into the expressivity of modal languages in a many-valued setting. Subsequent work in Badia and Olkhovikov (2020) demonstrated that no compact extension of these many-valued logic, displays both the Tarski Union Property and strong invariance for the proposed notion of *crisp* bisimulation introduced in Marti and Metcalfe (2014).

Some of the previously cited works (Marti and Metcalfe 2014; Badia et al. 2023) concentrate on crisp many-valued logics, where accessibility is a binary relation and proposition valuation is many-valued, parametric to a residuated lattice A. While other works (Caicedo and Rodríguez 2015; Jain et al. 2019) delve into fuzzy Kripke models, where both propositions and the accessibility relation are many-valued, taking values in the standard Gödel algebra $[0, 1]$. Our approach aims to combine these perspectives by working with models that are parametric to a residuated lattice A, where the accessibility and valuation relation are many-valued, taking values in $A \times A$. We believe that this approach not only admits different instances to better suit each modeling problem at hand but also provides a nuanced perspective, allowing transitions and propositions to represent consistent, vague, or inconsistent information.

This paper extends (Cruz et al. 2022b) with three main contributions:

(1) A notion of *graded* simulation and bisimulation, which enables comparing PLTS in a (formal) paraconsistent way;
(2) A modal (minimal) logic over the corresponding Kripke models, in which, in the tradition of so-called Hennessy-Milner logics, modalities are indexed by transition labels, or actions;
(3) A number of results on modal preservation by both crisp and graded simulations and bisimulations.

Alternatively to the concept of simulation or bisimulation as a *crisp* relation, typically used in many-valued logics (Marti and Metcalfe 2014; Jain et al. 2019; Cruz et al. 2022b), we study the concept of simulation and bisimulation as paraconsistent relations themselves. That is, a relation associating to each pair of states a pair of weights witnessing, respectively, the level of confidence or rejection on such a relationship. This leads to a less strict notion of simulation and bisimulation, aligning with the inherent uncertainty and inconsistencies possibly present in PLTS. Our work

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

draws inspiration from Nguyen (2022), where a logical characterization of fuzzy bisimulations in fuzzy modal logics is given. Thus, we extend this approach exploring logical characterizations of paraconsistent bisimulation in a paraconsistent modal logic over a general residuated lattice.

Preliminary results on classical soundness and the introduction of graded soundness for paraconsistent Kripke models are presented in § 5.2. In contrast to classical soundness, graded soundness allows for premises and conclusions that may involve vagueness or contradiction. A similar motivation can be found in Caicedo and Rodríguez (2010), where the authors consider a many-valued version of Kripke semantics in which both the propositions, denoted by function e, and accessibility relations, denoted by S, are infinitely many-valued in the standard Gödel algebra $[0, 1]$. For a Gödel-Kripke model $GK = \langle W, S, e\rangle$ where W is a set of worlds, the authors define set $e(x, T) = \{e(x, \varphi) : \varphi \in T\}$ which corresponds to a collection of valuations, that is, values in $[0, 1]$. A graded notion of logical consequence arises naturally as

$$T \models_{\leq GK} \varphi \text{ iff for any } GK\text{-model } M \text{ and any world } x \text{ in } M, \inf e(x, T) \leq e(x, \varphi)$$

Then it is proved soundness for the $\Box$-fragment and $\diamond$-fragment. Soundness is stated as $T \vdash_{\mathcal{G}_\Box} \varphi$ implies $T \models_{\leq GK} \varphi$, where $\mathcal{G}_\Box$ is the $\Box$-fragment. Our approach to defining graded soundness for paraconsistent models is similar. However, the main difference lies in studying theories where the classical notion of logical consequence is not equivalent to the graded one.

This work is part of a broader research agenda, of which we highlight references on specification theory and the stepwise implementation process (Sannella 2012) applied to paraconsistent transition systems and their corresponding processes (Cunha et al. 2023a,b). An application example of these systems to optimisation of quantum circuits, briefly mentioned in Cruz et al. (2022b), is further discussed in reference Barbosa and Madeira (2023).

Paper Structure. Section 2 revisits the concept of a PLTS and characterizes the twisted structure for the joint computation of positive and negative weights. The whole construction is parametric on a residuated lattice from which weights are taken. The compositional construction of (pointed) PLTS are characterized in § 3 by exploring the relevant category, following G. Winskel and M. Nielsen's 'recipe' (Winskel and Nielsen 1995). In § 4, PTLS are extended to (paraconsistent) Kripke models over which notions of crisp and graded simulation and bisimulation are studied. A modal logic for paraconsistent transition structures is introduced in § 5, alongside preliminary results on soundness and modal preservation for crisp and graded simulation and bisimulation. Finally, in § 6, we explore possible applications of PLTS. Section 7 concludes and points out a number of future research directions.

2. Paraconsistent Labelled Transition Systems

In line with the work documented in Bou et al. (2009), our constructions are parametric on a residuated lattice over a set A of possible truth values, which in the classical modal logic corresponds to a Boolean algebra. Our work extends this framework to the paraconsistent case, introducing the concept of a paraconsistent transition system where transitions are represented by pairs of weights $(a, b) \in A \times A$. We further introduce the notion of an $\boldsymbol{A}$*-twisted algebra* to manipulate pairs of weights, laying the foundation for the subsequent sections.

Formally, a *paraconsistent labeled transition system* (PLTS) incorporates two accessibility relations, classified as positive and negative, respectively, which characterize each transition in opposite ways: one represents the evidence of its presence and other the evidence of its absence. Both relations are weighted by elements of a residuated lattice $\boldsymbol{A} = \langle A, \wedge, \vee, \odot, \rightarrow, 1, 0\rangle$, where, $\langle A, \wedge, \vee, 1, 0\rangle$ is a lattice, $\langle A, \odot, 1\rangle$ is a monoid, and operation $\odot$ is the left adjoint to $\rightarrow$, its *residuum*, that is for all $a, b, c \in A$,

$$a \odot b \leq c \text{ iff } b \leq a \rightarrow c$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

We will focus on a particular class of complete residuated lattices that are bounded by a maximal 1 and a minimal element 0, respectively, and where lattice meet ($\wedge$) and monoidal composition ($\odot$) coincide. As these two operators coincide, we will henceforth omit one when working with the residuated lattices. Consequently, the adjunction above can be rephrased as

$$a \wedge b \leq c \text{ iff } b \leq a \rightarrow c \tag{1}$$

Furthermore, a pre-linearity condition is enforced

$$(a \rightarrow b) \vee (b \rightarrow a) = 1 \tag{2}$$

A residuated lattice obeying pre-linearity is known as a *MTL-algebra* (Esteva and Godo 2001). With a slight abuse of nomenclature, the designation *iMTL-algebra*, from *integral MTL-algebra*, will be used in the sequel for the class of semantic structures considered, i.e., complete, prelinear, residuated lattices in which $\wedge$ and $\odot$ coincide.

The Gödel algebra $G = \langle [0, 1], \min, \max, \min, \rightarrow, 1, 0\rangle$ is an example of such a structure that will be used in the sequel. Operators max and min retain the usual definitions, whereas implication is given by

$$a \rightarrow b = \begin{cases} 1, & \text{if } a \leq b \\ b, & \text{otherwise} \end{cases}$$

The following Lemma builds upon [Nguyen (2022), Lemma 2.1] to introduce a number of properties of these structures that will be useful in the paper. Since most of the assertions of this lemma are well known and their proofs straightforward, we omit the proof here and refer the reader instead to Appendix A for the proofs of Properties (12), (13), (14), and (16). The remaining properties are proved in Nguyen (2022).

Lemma 1. *Let* $\boldsymbol{A} = \langle A, \wedge, \vee, , \rightarrow, 1, 0\rangle$ *be an iMTL-algebra over a nonempty set A. The following properties hold, for any* $a,\ a',\ b,\ b',\ c \in A$

$$a \leq a' \text{ iff } (a \rightarrow a') = 1 \tag{3}$$

$$a \leq a' \text{ and } b \leq b' \text{ implies } a \wedge b \leq a' \wedge b' \tag{4}$$

$$a' \leq a \text{ and } b \leq b' \text{ implies } a \rightarrow b \leq a' \rightarrow b' \tag{5}$$

$$(a \leftrightarrow a') \wedge (b \leftrightarrow b') \leq (a \wedge b) \leftrightarrow (a' \wedge b') \tag{6}$$

$$(a \leftrightarrow a') \wedge (b \leftrightarrow b') \leq (a \vee b) \leftrightarrow (a' \vee b') \tag{7}$$

$$a \rightarrow (b \rightarrow c) = b \rightarrow (a \rightarrow c) \tag{8}$$

$$a \rightarrow (b \rightarrow c) = (a \wedge b) \rightarrow c \tag{9}$$

$$a \wedge (b \vee b') = (a \wedge b) \vee (a \wedge b') \tag{10}$$

$$a \vee (b \wedge b') = (a \vee b) \wedge (a \vee b') \tag{11}$$

$$a \wedge (a \rightarrow c) \leq c \tag{12}$$

$$(a \rightarrow c) \wedge (b \rightarrow c) = (a \vee b) \rightarrow c \tag{13}$$

$$(a \rightarrow b) \wedge (a \rightarrow c) = a \rightarrow (b \wedge c) \tag{14}$$

$$a \wedge (b \rightarrow c) \leq (a \rightarrow b) \rightarrow c \tag{15}$$

$$a \wedge (b \rightarrow c) \leq b \rightarrow (a \wedge c) \tag{16}$$

The following definition introduces the paper's basic semantic structure, which is to a large extent, independent of the particular residuated lattice chosen.

Definition 1. *A paraconsistent labeled transition system (PLTS) over a iMTL-algebra* $\boldsymbol{A}$, *and a set of atomic actions* Act, *is a structure* $\langle W, R\rangle$ *where,*

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

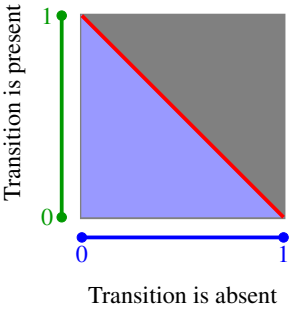
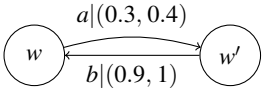


Figure 1. The vagueness-inconsistency square.

- *W is a non-empty set of states*
- *$R = (R_a \colon W \times W \to A \times A)_{a \in \mathrm{Act}}$ is an* Act*-indexed family of functions. Given any pair of states $(w_1, w_2) \in W \times W$ and an atomic action $a \in \mathrm{Act}$, relation R_a assigns a pair $(\alpha, \beta) \in A \times A$ such that α weights the evidence of the transition from w_1 to w_2 occurring through action a and β weights the evidence of the transition being absent.*

For any pair of weights $\omega = (\alpha, \beta) \in A \times A$, ω^+ denotes α, referred to as the *positive weight*, and ω^- denotes β, referred to as the *negative weight*.

Example 1. *A PLTS over the set $\{a, b\}$ of atomic actions, taking weights from Gödel algebra G is depicted below*



Note that $R_a(w, w')$ represents vague information, because the sum of the positive and negative weights for the corresponding transition is less than 1. On the other hand, $R_b(w', w)$ represents inconsistent information.

Actually, the road from *vague* to *inconsistent* transitions is represented in Figure 1 which depicts all pairs of weights $(\alpha, \beta) \in [0, 1] \times [0, 1]$ in a Gödel algebra.

These pairs of weights jointly express different behaviors:

- *inconsistency*, when the positive and negative weights are contradictory, that is, they sum to some value greater then 1; this corresponds to the upper triangle in Figure 1, filled in grey.
- *vagueness*, when the sum is less than 1, corresponding to the lower, periwinkle triangle in Figure 1.
- *consistency*, when the sum is exactly 1, which means that the measures of the factors enforcing or preventing a transition are complementary, corresponding to the red line in Figure 1.

This discussion can be lifted to arbitrary iMTL-algebras over A through the introduction of a metric space over A^2, as proposed in our previous work (Cruz et al. 2022a).

Jointly operating with pairs of (positive and negative) weights is essential for defining a (modal) logic over PLTS. The main conceptual tool for this is the notion of a *twisted algebra* (Kracht 1998).

Definition 2. *Given an iMTL-algebra $\boldsymbol{A} = \langle A, \wedge, \vee, 1, 0, \to \rangle$ a $\boldsymbol{A}$-twisted algebra*

$$\mathcal{A} = \langle A \times A, \overset{\wedge}{\wedge}, \overset{\vee}{\vee}, \Rightarrow, /\!\!/ \rangle$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

is defined as follows:

- $(\alpha,\beta)\overset{\wedge}{\wedge}(\alpha',\beta') = (\alpha\wedge\alpha',\beta\vee\beta')$
- $(\alpha,\beta)\overset{\vee}{\vee}(\alpha',\beta') = (\alpha\vee\alpha',\beta\wedge\beta')$
- $(\alpha,\beta)\Rightarrow(\alpha',\beta') = ((\alpha\to\alpha')\wedge(\beta'\to\beta),\alpha\wedge\beta')$
- $/\!\!/(\alpha,\beta) = (\beta,\alpha)$

As usual, equivalence is defined by abbreviation,

$$(\alpha,\beta)\Leftrightarrow(\alpha',\beta') \;=\; ((\alpha,\beta)\Rightarrow(\alpha',\beta'))\overset{\wedge}{\wedge}((\alpha',\beta')\Rightarrow(\alpha,\beta))$$

The order in $\boldsymbol{A}$ *is lifted to* $\mathcal{A}$ *as* $(a,b)\preccurlyeq(c,d)$ iff $a\leq c$ and $b\geq d$

The problem with the definition above is that the original $\to$, $\wedge$ adjunction in $\boldsymbol{A}$ does not lift to $\mathcal{A}$. A counterexample in the Gödel algebra falsifies the envisaged condition

$$(\alpha,\beta)\overset{\wedge}{\wedge}(\alpha',\beta') \preccurlyeq (\alpha'',\beta'') \text{ iff } (\alpha,\beta) \preccurlyeq (\alpha',\beta')\Rightarrow(\alpha'',\beta'')$$

Indeed, let $(\alpha,\beta)=(0.9,0.6)$, $(\alpha',\beta')=(0.9,0.9)$ and $(\alpha'',\beta'')=(1,0.8)$. Clearly,

$$(0.9,0.6)\overset{\wedge}{\wedge}(0.9,0.9) \;=\; (0.9\wedge 0.9, 0.6\vee 0.9) \;=(0.9,0.9) \preccurlyeq (1,0.8)$$

However,

$$(0.9,0.6)\not\preccurlyeq(0.9,0.9)\Rightarrow(1,0.8) \;=\; (0.9\to 0.8\wedge 1\to 0.9, 0.9\wedge 0.8) \;=\; (0.8,0.8)$$

The adjunction is recovered, however, replacing $\overset{\wedge}{\wedge}$ by

$$(\alpha,\beta)\otimes(\alpha',\beta') \;=\; (\alpha\wedge\alpha', \alpha\to\beta'\wedge\alpha'\to\beta)$$

Thus,

Lemma 2. *The operator* $\otimes$ *as defined above has* $\Rightarrow$ *as its residuum, that is as a right adjoint:*

$$(\alpha,\beta)\otimes(\alpha',\beta') \preccurlyeq (\alpha'',\beta'') \text{ iff } (\alpha,\beta) \preccurlyeq (\alpha',\beta')\Rightarrow(\alpha'',\beta'') \tag{17}$$

Proof. We start by simplifying both sides of equivalence (17):

$$\begin{aligned}
&(\alpha,\beta)\otimes(\alpha',\beta') \preccurlyeq (\alpha'',\beta'')\\
&\text{iff}\;\; (\alpha\wedge\alpha', (\alpha\to\beta')\wedge(\alpha'\to\beta)) \preccurlyeq (\alpha'',\beta'')\\
&\text{iff}\;\; \alpha\wedge\alpha' \leq \alpha'' \text{ and } (\alpha\to\beta')\wedge(\alpha'\to\beta)\geq\beta''
\end{aligned}$$

and,

$$\begin{aligned}
&(\alpha,\beta) \preccurlyeq (\alpha',\beta')\Rightarrow(\alpha'',\beta'')\\
&\text{iff}\;\; (\alpha,\beta) \preccurlyeq ((\alpha'\to\alpha'')\wedge(\beta''\to\beta'),\alpha'\wedge\beta'')\\
&\text{iff}\;\; \alpha \leq (\alpha'\to\alpha'')\wedge(\beta''\to\beta') \text{ and } \beta\geq\alpha'\wedge\beta''
\end{aligned}$$

Let us now prove that, if $(\alpha,\beta)\otimes(\alpha',\beta')\preccurlyeq(\alpha'',\beta'')$ then $(\alpha,\beta)\preccurlyeq(\alpha',\beta')\Rightarrow(\alpha'',\beta'')$. By hypothesis $\alpha'\wedge\alpha\leq\alpha''$ using (1), $\alpha\leq\alpha'\to\alpha''$. Also by hypothesis, $\alpha\to\beta'\geq\beta''$, using (1) $\alpha\wedge\beta''\leq\beta'$, that is the same as writing $\beta''\wedge\alpha\leq\beta'$, using (1) again $\alpha\leq\beta''\to\beta'$. Since $\wedge$ is monotone we prove that $\alpha\leq(\alpha'\to\alpha'')\wedge(\beta''\to\beta')$. Finally, by hypothesis $\alpha'\to\beta\geq\beta''$ by (1), $\alpha'\wedge\beta''\leq\beta$.

For the converse implication, if $(\alpha,\beta)\preccurlyeq(\alpha',\beta')\Rightarrow(\alpha'',\beta'')$ then $(\alpha,\beta)\otimes(\alpha',\beta')\preccurlyeq(\alpha'',\beta'')$, note that, by hypothesis $\alpha\leq\alpha'\to\alpha''$ using (1), $\alpha'\wedge\alpha\leq\alpha''$. Again, $\beta\geq\alpha'\wedge\beta''$ using (1), $\beta''\leq\alpha'\to\beta$ also by hypothesis $\alpha\leq\beta''\to\beta'$ using (1), $\beta''\wedge\alpha\leq\beta'$ which is the same as $\alpha\wedge\beta''\leq\beta'$, using (1), $\beta''\leq\alpha\to\beta'$. Since $\wedge$ is monotone, $\beta''\leq\alpha\to\beta'\wedge\alpha'\to\beta$. □

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

From now on, we will consider that any twisted algebra, as in Definition 2, is enriched with operator $\otimes$, that is for any iMTL-algebra $\boldsymbol{A}$ its $\boldsymbol{A}$-twisted algebra is $\mathcal{A} = \langle A \times A, \overset{\wedge}{\wedge}, \overset{\vee}{\vee}, \Rightarrow, /\!\!/, \otimes \rangle$.

Finally, the following crucial lemma introduces a number of properties relating pairs of weights that will be used in the sequel. The following assertions follow directly from the properties of Lemma 1. To streamline the presentation and ensure the article is self-contained, we provide the proof of the following lemma in Appendix B.

Lemma 3. *Let* $\mathcal{A} = \langle A \times A, \overset{\wedge}{\wedge}, \overset{\vee}{\vee}, \Rightarrow, /\!\!/, \otimes \rangle$ *be a* $\boldsymbol{A}$*-twisted algebra over* $\boldsymbol{A} = \langle A, \wedge, \vee, 1, 0, \rightarrow \rangle$. *Thus,*

$$\text{if } (a',b') \preccurlyeq (a,b) \text{ and } (c,d) \preccurlyeq (c',d') \text{ then } (a,b) \Rightarrow (c,d) \preccurlyeq (a',b') \Rightarrow (c',d') \tag{18}$$

$$\text{if } (a',b') \preccurlyeq (a,b) \text{ and } (c,d) \preccurlyeq (c',d') \text{ then } (a,b) \otimes (c,d) \preccurlyeq (a',b') \otimes (c',d') \tag{19}$$

$$((a,b) \Leftrightarrow (c,d)) = (/\!\!/(a,b) \Leftrightarrow /\!\!/(c,d)) \tag{20}$$

$$(a,b) \otimes \Big((c,d) \overset{\vee}{\vee} (e,f)\Big) = \Big((a,b) \otimes (c,d)\Big) \overset{\vee}{\vee} \Big((a,b) \otimes (e,f)\Big) \tag{21}$$

$$(a,b) \otimes \Big((c,d) \overset{\wedge}{\wedge} (e,f)\Big) \preccurlyeq \Big((a,b) \otimes (c,d)\Big) \overset{\wedge}{\wedge} \Big((a,b) \otimes (e,f)\Big) \tag{22}$$

$$(a,b) \otimes \Big((c,d) \Rightarrow (e,f)\Big) \preccurlyeq (c,d) \Rightarrow \Big((a,b) \otimes (e,f)\Big) \tag{23}$$

$$(a,b) \Rightarrow \Big((c,d) \Rightarrow (e,f)\Big) \preccurlyeq \Big((a,b) \otimes (c,d)\Big) \Rightarrow (e,f) \tag{24}$$

$$(a,b) \otimes \Big((c,d) \Rightarrow (e,f)\Big) \preccurlyeq \Big((a,b) \Rightarrow (c,d)\Big) \Rightarrow (e,f) \tag{25}$$

$$(a,b) \Rightarrow \Big((c,d) \Rightarrow (e,f)\Big) \preccurlyeq (c,d) \Rightarrow \Big((a,b) \Rightarrow (e,f)\Big) \tag{26}$$

$$\Big((a,b) \Leftrightarrow (a',b')\Big) \overset{\wedge}{\wedge} \Big((c,d) \Leftrightarrow (c',d')\Big) \preccurlyeq \Big((a,b) \overset{\wedge}{\wedge} (c,d)\Big) \Leftrightarrow \Big((a',b') \overset{\wedge}{\wedge} (c',d')\Big) \tag{27}$$

$$\Big((a,b) \Leftrightarrow (a',b')\Big) \overset{\wedge}{\wedge} \Big((c,d) \Leftrightarrow (c',d')\Big) \preccurlyeq \Big((a,b) \overset{\vee}{\vee} (c,d)\Big) \Leftrightarrow \Big((a',b') \overset{\vee}{\vee} (c',d')\Big) \tag{28}$$

$$\Big((a,b) \Rightarrow (a',b')\Big) \overset{\wedge}{\wedge} \Big((c,d) \Rightarrow (c',d')\Big) \preccurlyeq \Big((a,b) \overset{\vee}{\vee} (c,d)\Big) \Rightarrow \Big((a',b') \overset{\vee}{\vee} (c',d')\Big) \tag{29}$$

$$\Big((a,b) \Rightarrow (a',b')\Big) \overset{\wedge}{\wedge} \Big((c,d) \Rightarrow (c',d')\Big) \preccurlyeq \Big((a,b) \overset{\wedge}{\wedge} (c,d)\Big) \Rightarrow \Big((a',b') \overset{\wedge}{\wedge} (c',d')\Big) \tag{30}$$

3. New PLTS from Old

This section introduces the relevant operators by exploring the structure of the category of $\mathsf{Pt}_{\boldsymbol{A}}$ of pointed PLTS, that is whose objects are PLTS with a distinguished initial state. Formally, a *pointed PLTS* is $\langle W, i, R, \mathrm{Act} \rangle$, where $\langle W, R, V \rangle$ is a PLTS, $i \in W$ is the initial state and Act is a set of action symbols. As any other mathematical, or computational structure, pointed PLTS over a iMTL-algebra $\boldsymbol{A}$ live in a category whose morphisms respect, as one would expect, the structure of both accessibility relations, that is, given two pointed PLTS $T_1 = \langle W_1, i_1, R_1, \mathrm{Act} \rangle$, $T_2 = \langle W_2, i_2, R_2, \mathrm{Act} \rangle$ over an iMTL-algebra $\boldsymbol{A}$ a *morphism* from T_1 to T_2 is a function $h : W_1 \to W_2$ such that, for any $a \in \mathrm{Act}$, $(R_1)_a(w,v) \preccurlyeq (R_2)_a(h(w), h(v))$. Note that pointed PLTS and their morphisms form a category, with composition and identities borrowed from Set.

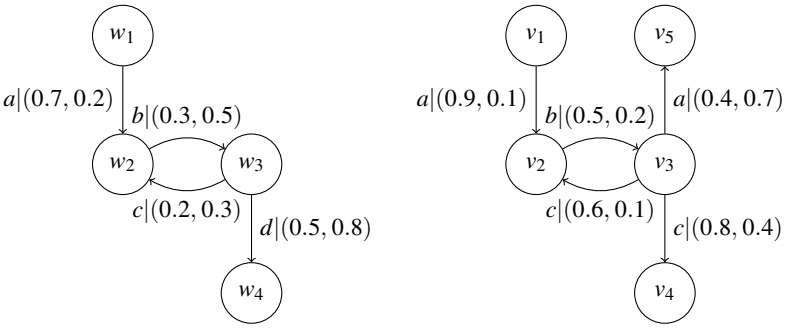
Our concern in this section is to come up with a number of meaningful operators to compose PLTS capturing the usual compositional concepts in process algebra, namely parallel and

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

concurrent composition, choice, interface relabeling and restriction, and prefix as a form of sequential composition.

A quite standard way to do this is by defining a category of PLTS by imposing the corresponding algebraic structure upon the category Set of sets and set theoretical functions, and then seeking for the relevant compositional operators as standard constructions in this category. This avoids the introduction of *ah hoc* definitions and brings for free a number of properties for some operators (e.g. the ones coming from the definition of limit and colimit this category inherits from Set). In a sense, to use a popular aphorism in category theory, our aim is *to find rather than inventing*.This method was pioneered by Nielsen and Winskel on their seminal paper on models of concurrency Winskel and Nielsen (1995).

Example 2. *Consider the two PLTS over the Gödel algebra* $M_1 = \langle W_1, R_1, V_1\rangle$ *and* $M_2 = \langle W_2, R_2, V_2\rangle$ *depicted bellow over the set* $\{a, b, c, d\}$ *of atomic actions. Function* $h = \{w_1 \mapsto v_1, w_2 \mapsto v_2, w_3 \mapsto v_3\}$ *is a morphism from* M_1 *to* M_2.



In this setting, new pointed PLTS can be built compositionally. This section introduces the relevant operators, always parametric on the underlying iMTL-algebra $\boldsymbol{A}$, by exploring the structure of the category of $\mathsf{Pt}_{\boldsymbol{A}}$ of *pointed* PLTS over $\boldsymbol{A}$, that is whose objects are PLTS with a distinguished initial state and arrows in $\mathsf{Pt}_{\boldsymbol{A}}$ are allowed between PLTS with different sets of action symbols therefore generalizing the informal definition of morphism given above as follows:

Definition 3. *Let* $T_1 = \langle W_1, i_1, R_1\rangle$ *and* $T_2 = \langle W_2, i_2, R_2\rangle$ *be two pointed PLTS over the set of atomic actions* Act_1 *and* Act_2, *respectively. A morphism in* $\mathsf{Pt}_{\boldsymbol{A}}$ *from* T_1 *to* T_2 *consists on a pair of functions* $(\sigma : W_1 \to W_2, \lambda : \mathrm{Act}_1 \to_\perp \mathrm{Act}_2)$ *such that*[1] $\sigma(i_1) = i_2$ *and for any* $a \in \mathrm{Act}_1$,

$$(R_1)_a(w, w') \preccurlyeq (R_2)^{\perp}_{\lambda(a)}(\sigma(w), \sigma(w'))$$

For an accessibility relation R, $R^{\perp} = R \cup R_{\perp}$ *with* $R_{\perp}(w, w) = (1, 0)$ *for any state* $w \in W$, *denotes* R *enriched with idle transitions in each state.*

Clearly $\mathsf{Pt}_{\boldsymbol{A}}$ forms a category, whose composition in inherited from Set. Hence, as usual in categories of algebraic structures, most constructions in this Section build on the corresponding constructions in Set.

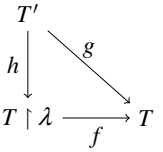
An algebra of PLTS typically includes some form of parallel composition, disjoint union, restriction, relabeling and prefixing, as one is used to from the process algebra literature (Baeten et al. 2010). Accordingly, these operators are defined along the lines proposed by Winskel and Nielsen (1995), for the standard, more usual case.

Restriction. The restriction operator is intended to control the interface of a transition system, preserving, in the case of a PLTS, the corresponding positive and negative weights. Formally,

Definition 4. *Let* $T = \langle W, i, R, \mathrm{Act}\rangle$ *be pointed PLTS and* $\lambda : \mathrm{Act}' \to \mathrm{Act}$ *be an inclusion morphism. The restriction of* T *to* λ, $T \restriction \lambda$, *is a PLTS* $\langle W, i, R'\rangle$ *over* Act' *such that, for any* $w, v \in W$ *and* $a \in \mathrm{Act}'$, $R'_a(w, v) = R_a(w, v)$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

There is a morphism $f = (1_W, \lambda)$ from $T \restriction \lambda$ to T, and a functor $P : \mathsf{Pt}_{\boldsymbol{A}} \to \mathsf{Set}_\perp$ which sends a morphism $(\sigma, \lambda) : T \to T'$ to the partial function $\lambda : \mathrm{Act}' \to \mathrm{Act}$. Clearly, f is the Cartesian lifting of morphism $P(f) = \lambda$ in $\mathsf{Set}_\perp$. Being Cartesian means that for any $g : T' \to T$ in $\mathsf{Pt}_{\boldsymbol{A}}$ such that $P(g) = \lambda$ there is a unique morphism h such that $P(h) = 1_{\mathrm{Act}'}$ making the following diagram to commute:



Relabeling. In the same group of *interface-modifier* operators, is *relabeling*, which renames the labels of a PLTS according to a total function $\lambda : \mathrm{Act} \to \mathrm{Act}'$.

Definition 5. *Let $T = \langle W, i, R, \mathrm{Act}\rangle$ be a pointed PLTS and $\lambda : \mathrm{Act} \to \mathrm{Act}'$ be a total function. The relabeling of T according to λ is denoted by $T\{\lambda\}$ and it is the PLTS $\langle W, i, R'\rangle$ over Act' where for any $w, v \in W$ and $a \in \mathrm{Act}$, $R'_{\lambda(a)}(w, v) = R_a(w, v)$.*

Dually to the previous case, there is a morphism $f = (1_W, \lambda)$ from T to $T\{\lambda\}$ which is the cocartesian lifting of λ $(= P(f))$.

Parallel Composition. The product of two pointed PLTS combines their state spaces and includes all *synchronous* transitions, triggered by the simultaneous occurrence of an action of each component, as well as *asynchronous* ones in which a transition in one component is paired with an *idle* transition and valuations, labeled by $\perp$, in the other. Formally,

Definition 6. *Let $T_1 = \langle W_1, i_1, R_1, \mathrm{Act}_1\rangle$ and $T_2 = \langle W_2, i_2, R_2, \mathrm{Act}_2\rangle$ be two pointed PLTS. Their parallel composition $T_1 \times T_2$ is the PLTS $\langle W_1 \times W_2, (i_1, i_2), R, \mathrm{Act}_1 \times_\perp \mathrm{Act}_2\rangle$ where,*

$$\mathrm{Act}_1 \times_\perp \mathrm{Act}_2 = \{(a, \perp) \mid a \in \mathrm{Act}_1\} \cup \{(\perp, b) \mid b \in \mathrm{Act}_2\} \cup \{(a, b) \mid a \in \mathrm{Act}_1, b \in \mathrm{Act}_2\}$$

Such that, $R_{(a,b)}((w_1, w_2), (v_1, v_2)) = (\alpha, \beta)$ if and only if, $(R_1)^\perp_a(w_1, v_1) = (\alpha_1, \beta_1)$ and $(R_2)^\perp_b(w_2, v_2) = (\alpha_2, \beta_2)$ and $(\alpha, \beta) = (\alpha_1, \beta_1) \overset{\wedge}{\wedge} (\alpha_2, \beta_2)$.

Lemma 4. *Parallel composition is the product construction in $\mathsf{Pt}_{\boldsymbol{A}}$.*

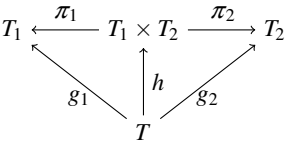
Proof. Let $T = \langle W, i, R, \mathrm{Act}\rangle$ and $T_1 = \langle W_1, i_1, R_1, \mathrm{Act}_1\rangle$ be two pointed PLTS and let us define $T_1 \times T_2 = \langle W_1 \times W_2, (i_1, i_2), R', \mathrm{Act}'\rangle$ where $\mathrm{Act}' = \mathrm{Act}_1 \times_\perp \mathrm{Act}_2$ according to Definition 6. The projections $\pi_k : W_1 \times W_2 \to T_k$ defined as $\pi_k(w_1, w_2) = w_k$ for $k = 1, 2$ are morphisms in the $\mathsf{Pt}_{\boldsymbol{A}}$ category because $(R')_{(a,b)}((w_1, w_2), (v_1, v_2)) = (\alpha, \beta)$ if and only if, $(R_1)^\perp_a(\pi_1(w_1, w_2), \pi_1(v_1, v_2)) = (\alpha_1, \beta_1)$ and $(R_2)^\perp_b(\pi_2(w_1, w_2), \pi_2(v_1, v_2)) = (\alpha_2, \beta_2)$ and from $(\alpha, \beta) = (\alpha_1, \beta_1) \overset{\wedge}{\wedge} (\alpha_2, \beta_2)$ follows $(\alpha, \beta) \preccurlyeq (\alpha_1, \beta_1)$ and $(\alpha, \beta) \preccurlyeq (\alpha_2, \beta_2)$.

In the diagram below let $g_i = (\sigma_i, \lambda_i)$, for $i = 1, 2$, and define h as $h = (\langle\sigma_1, \sigma_2\rangle, \langle\lambda_1, \lambda_2\rangle)$, where $\langle f_1, f_2\rangle(x) = (f_1(x), f_2(x))$ is the universal arrow in a product diagram in Set. Clearly, h lifts universality to $\mathsf{Pt}_{\boldsymbol{A}}$, as the unique arrow making the diagram to commute. It remains show it is indeed an arrow in the category. By Definition 6, for each $R_a(w, w') = (\alpha, \beta)$, there is a transition $(R_1)^\perp_{\lambda_1(a)}(\sigma_1(w), \sigma_1(w')) = (\alpha_1, \beta_1)$ such that $\alpha \leq \alpha_1$ and $\beta \geq \beta_1$; and also a transition $R_2{}^\perp_{\lambda_2(a)}(\sigma_2(w), \sigma_2(w')) = (\alpha_2, \beta_2)$ such that $\alpha \leq \alpha_1$ and $\beta \geq \beta_2$. Moreover, there is a transition

$$R'_{\langle\lambda_1,\lambda_2\rangle(a)}(\langle\sigma_1, \sigma_2\rangle(w), \langle\sigma_1, \sigma_2\rangle(w')) = (\alpha_1, \beta_1) \overset{\wedge}{\wedge} (\alpha_2, \beta_2)$$

Thus, there is a transition $R'_{\langle\lambda_1,\lambda_2\rangle(a)}(\langle\sigma_1, \sigma_2\rangle(w), \langle\sigma_1, \sigma_2\rangle(w')) = (\alpha', \beta')$, for any $R_a(w, w') = (\alpha, \beta)$, such that $\alpha \leq \alpha'$ and $\beta \geq \beta'$. Furthermore, $\langle\sigma_1, \sigma_2\rangle(i) = (\sigma_1(i), \sigma_2(i)) = (i_1, i_2)$. This establishes h as a $\mathsf{Pt}_{\boldsymbol{A}}$ morphism.

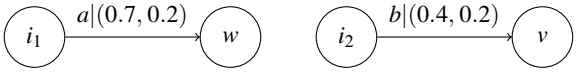
https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press



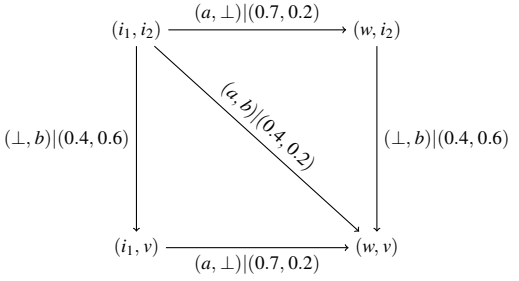
□

The following example illustrates the product of two pointed PLTS.

Example 3. *Consider the two PLTS, T_1 and T_2 depicted below.*



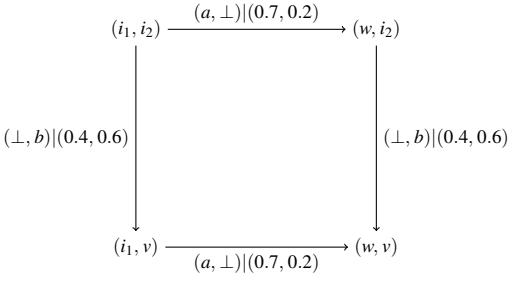
Their product T is the PLTS



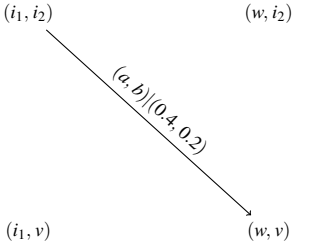
A suitable combination of parallel composition and restriction may enforce different synchronization disciplines. For example, *interleaving* or *asynchronous product* $T_1||T_2$ is defined as $(T_1 \times T_2) \restriction \lambda$ with the inclusion $\lambda : \mathrm{Act} \to \mathrm{Act}_1 \times_\bot \mathrm{Act}_2$ for $\mathrm{Act} = \{(a, \bot) \mid a \in \mathrm{Act}_1\} \cup \{(\bot, b) \mid b \in \mathrm{Act}_2\}$. This results in a PLTS $\langle W_1 \times W_2, (i_1, i_2), R, \mathrm{Act}\rangle$ such that for any $a \in \mathrm{Act}$, if $(R_2)_a(w, w') = (\alpha, \beta)$ then $(R_1)_a(w, w') = (\alpha, \beta)$.

Similarly, the *synchronous product* $T_1 \otimes T_2$ is also defined as $(T_1 \times T_2) \restriction \lambda$, taking now $\mathrm{Act} = \{(a, b) \mid a \in \mathrm{Act}_1 \text{ and } b \in \mathrm{Act}_2\}$ as the domain of λ.

Example 4. *Interleaving and synchronous product of T_1 and T_2 as in Example 3, are depicted below.*



$T_1||T_2$



$T_1 \otimes T_2$

Sum. The sum of two pointed PLTS corresponds to their nondeterminisitic composition: the resulting PLTS behaves as either of its components. Formally,

Definition 7. *Let $T_1 = \langle W_1, i_1, R_1, \mathrm{Act}_1\rangle$ and $T_2 = \langle W_2, i_2, R_2, \mathrm{Act}_2\rangle$ be two pointed PLTS. Their sum $T_1 + T_2$ is the PLTS $\langle W, (i_1, i_2), R, \mathrm{Act}\rangle$ with $\mathrm{Act} = \mathrm{Act}_1 \cup \mathrm{Act}_2$ and*

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

- $W = (W_1 \times \{i_2\}) \cup (\{i_1\} \times W_2)$
- $R_a((w_1, w_2), (v_1, v_2)) = (\alpha, \beta)$ *if and only if there is a transition* $(R_1)_a(w_1, v_1) = (\alpha, \beta)$ *or there is a transition* $(R_2)_a(w_2, v_2) = (\alpha, \beta)$

Sum is actually a coproduct in Pt_A (the proof follows the argument used for the product case), making $T_1 + T_2$ dual to $T_1 \times T_2$.

Example 5. *The sum* $T_1 + T_2$*, for* T_1, T_2 *defined as in Example 3 is given by*

$$(i_1, v) \xleftarrow{b|(0.4,\,0.6)} (i_1, i_2) \xrightarrow{a|(0.7,\,0.2)} (w, i_2)$$

Prefixing. As a limited form of sequential composition, prefix appends to a pointed PLTS a new initial state and a new transition to the previous initial state, after which the system behaves as the original one.

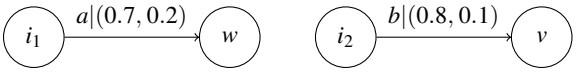
Definition 8. *Let* $T = \langle W, i, R, \mathrm{Act}\rangle$ *be a pointed PLTS. Consider* w_{new} *a fresh state identifier not in* W*. Given an action* $a \notin \mathrm{Act}$*, and* $\alpha, \beta \in A$*, the prefix* $a|(\alpha, \beta)T$ *is defined as*

$$\langle W \cup \{w_{new}\}, w_{new}, R', \mathrm{Act} \cup \{a\}\rangle$$

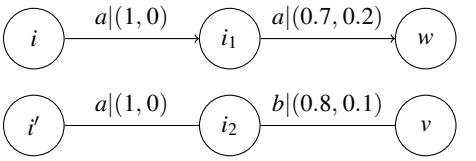
where $R'_b(w, v) = R_b(w, v)$ *if* $b \in \mathrm{Act}$ *and* $R'_a(w_{new}, i) = (\alpha, \beta)$*.*

Since it is not required that the prefixing label is distinct from the ones in the original system, prefixing does not extend to a functor in Pt_A, as illustrated in the counterexample below. This is obviously the case for a category of classical labeled transition systems as well. In both cases, however, prefix extends to a functor if the corresponding categories are restricted to action-preserving morphisms, that is, in which the action component of a morphism is always an inclusion

Example 6. *Consider two pointed PLTS* T_1 *and* T_2



connected by a morphism $(\sigma, \lambda) : T_1 \to T_2$ *such that* $\sigma(i_1) = i_2$*,* $\sigma(w) = v$ *and* $\lambda(a) = b$*. Now consider the prefixes* $a|(1, 0)T_1$ *and* $a|(1, 0)T_2$ *depicted below.*



Clearly, a mapping from the actions in $a|(1, 0)T_1$ *to the actions in* $a|(1, 0)T_1$ *does not exist so neither exists a morphism between the two systems.*

Functorial Extensions. Other useful operations between pointed PLTS, typically acting on transitions' positive and negative weights, and whose definition depends on the specific residuated lattice chosen, can be defined functorially in Pt_A. An example involving a pointed PLTS defined over a Gödel algebra is an operation that uniformly increases or decreases the value of the positive (or the negative, or both) weight in all transitions. Let

$$\alpha \oplus \beta = \begin{cases} 1 \text{ if } \alpha + \beta \geq 1 \\ 0 \text{ if } \alpha + \beta \leq 0 \\ \alpha + \beta \text{ otherwise} \end{cases}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Thus,

Definition 9. *Let* $T = \langle W, i, R, \text{Act}\rangle$ *be a pointed PLTS. Taking* $v \in [-1, 1]$, *the positive v-approximation* $T^{+}_{\oplus v}$ *is a pointed PLTS* $\langle W, i, R', \text{Act}\rangle$ *where for any relation* $R_a(w, w') = (\alpha, \beta)$ *we define* $R'_a(w, w') = (\alpha \oplus v, \beta)$.

The definition extends to a functor in $\mathsf{Pt}_{\boldsymbol{A}}$ which is the identity in morphisms. Similar operations can be defined to act on the negative accessibility relation or both.

Another useful operation removes all transitions in a pointed PLTS for which the positive accessibility relation is below a certain value and the negative accessibility relation is above a certain value. Formally,

Definition 10. *Let* $T = \langle W, i, R, \text{Act}\rangle$ *be a pointed PLTS. For* $p, n \in [0, 1]$, *the purged PLTS* $T_{p\uparrow\downarrow n}$ *is defined as* $\langle W, i, R', \text{Act}\rangle$ *where*

$$R'_a(w, w') = \begin{cases} R_a(w, w') & \text{if } (p, m) \preccurlyeq R_a(w, w') \\ (0, 1) & \text{otherwise} \end{cases}$$

Clearly, the operation extends to a functor in $\mathsf{Pt}_{\boldsymbol{A}}$, mapping morphisms to themselves.

4. From PLTS to Paraconsistent Kripke Models: Crisp and Graded Bisimulations

Having introducing PLTS and their algebra, this section goes another step ahead. Stating and proving properties of such systems entails the need for a suitable modal logic to explore their (double) transition structure. We start in this section by endowing PLTS with propositions that take values on $A \times A$, therefore providing a corresponding notion of a paraconsistent Kripke model. Two approaches to defining simulation and bisimulation over them, one crisp, as in the original paper (Cruz et al. 2022b), and another graded are discussed. We start with the following definition:

Definition 11. *A paraconsistent Kripke model over a iMTL-algebra* $\boldsymbol{A}$, *a set of atomic actions* Act *and a set of proposition symbols* Prop *is a structure* $\langle W, R, V\rangle$ *where* $\langle W, R\rangle$ *is a paraconsistent transition system and* $V : W \times \text{Prop} \to A \times A$ *is a valuation function that for a proposition* $p \in \text{Prop}$ *and state* $w \in W$, $V(w, p) = (\alpha, \beta) \in A \times A$ *means that* α *weights the evidence of* p *holding at state* w *and* β *weights the evidence of* p *not holding at* w.

Paraconsistent Kripke models can be related by simulation or bisimulation relations, either crisp or graded, as discussed in the sequel. As before, all constructions are parametric in the underlying iMTL-algebra $\boldsymbol{A}$, whose explicit reference is, from now on, omitted.

Crisp Simulation and Bisimulation

Definition 12. *Let* $M = \langle W, R, V\rangle$ *and* $M' = \langle W', R', V'\rangle$ *be two paraconsistent models over a set of action symbols* Act *and another of propositions* Prop. *A relation* $S \subseteq W \times W'$ *is a simulation provided that, for all* $(w, w') \in S$, $p \in \text{Prop}$ *and* $a \in \text{Act}$

- $V(w, p) \preccurlyeq V'(w', p)$, *and*
- $R_a(w, v) = (\alpha, \beta)$ *then exists* $v' \in W'$, $\gamma, \delta \in A$ *such that* $R'_a(w', v') = (\gamma, \delta)$ *and* $R_a(w, v) \preccurlyeq R'_a(w', v')$ *and* $(v, v') \in S$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Whenever one restricts in the definition above to the existence of values γ (resp. δ) such that $\gamma \geq \alpha$ (resp. $\delta \leq \beta$), the corresponding simulation is called *positive* (resp. *negative*).

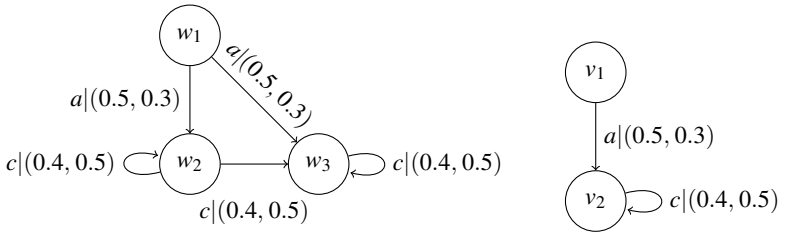
Example 7. *Recall the PLTS depicted above over the empty set of proposition symbols. The following relation is a crisp simulation,*

$$S = \{(w_1, v_1), (w_2, v_2), (w_3, v_2), (w_4, v_3), (w_5, v_4)\}$$

Definition 13. *Let $M = \langle W, R, V\rangle$ and $M' = \langle W', R', V'\rangle$ be two paraconsistent models over the set of actions* Act *and the set of propositions* Prop. *A relation $B \subseteq W \times W'$ is a bisimulation if for $(w, w') \in B$, $p \in$ Prop and $a \in$ Act the following conditions hold:*

- $V(w, p) = V'(w', p)$, *and*
- *for any $v \in W$ such that $R_a(w, v) = (\alpha, \beta)$, there is $v' \in W'$ such that $R'_a(w', v') = (\alpha, \beta)$ and $(v, v') \in S$*
- *for any $v' \in W'$ such that $R'_a(w', v') = (\alpha, \beta)$, there is $v \in W$ such that $R_a(w, v) = (\alpha, \beta)$ and $(v, v') \in S$*

Example 8. *Consider the two paraconsistent models depicted below.*



with $V_1(w, p) = V_2(v_1, p)$ and $V_1(w_2, p) = V_1(w_3, p) = V_2(v_2, p)$ for any proposition symbol p. The relation $B = \{(w_1, v_1), (w_2, v_2), (w_3, v_3)\}$ is a crisp bisimulation.

Graded Simulation and Bisimulation

When comparing paraconsistent transition structures, a more intuitive approach consists of treating bisimulation as a graded relation, encompassing both positive and negative weights. This relation, we refer to as *paraconsistent bisimulation*, follows the lines of work documented in Nguyen (2022) for fuzzy modal logic. This subsection generalizes (Nguyen 2022) approach for paraconsistent modal logic and discusses its, slightly weird, expressivity.

Definition 14. *Let $M = \langle W, R, V\rangle$ and $M' = \langle W', R', V'\rangle$ be two paraconsistent models over a iMTL-algebra $\boldsymbol{A}$ and a set Act of actions. A relation $S^G : W \times W' \to A \times A$ is a **graded simulation** if the following conditions hold for all $p \in$ Prop, $a \in$ Act and all possible values of free variables:*

$$S^G(w, w') \preccurlyeq (V(w, p) \Rightarrow V'(w', p)) \tag{31}$$

$$\exists v' \in W',\ (S^G(w, w') \otimes R_a(w, v)) \preccurlyeq (R'_a(w', v') \otimes S^G(v, v')) \tag{32}$$

It is worth noticing that the crisp simulation is not a particular case of the graded simulation. The following two examples discuss this issue.

Example 9. *Consider the following paraconsistent model with one proposition p:*



https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

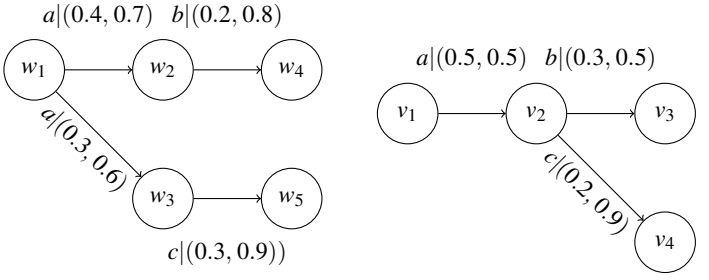
with $V(w,p) = V'(w',p) = (1,0)$. *Since* $V(w,p) \preccurlyeq V'(w',p)$ *and* $R_a(w,w) \preccurlyeq R'_a(w',w')$ *relation* $S = \{(w,w')\}$ *is a crisp simulation, i.e., it is possible to informally write* $S(w,w') = (1,0)$ *since there is complete evidence that states* w *and* w' *are similar and absolute minimal evidence that they are not. Notice that* $S(w,w') = (1,0)$ *satisfies conditions (31) and (32), that is,*

$$(1,0) \preccurlyeq V(w,p) \Rightarrow V'(w',p) = (1,0)$$

$$(1,0) \otimes (1,1) \preccurlyeq (1,0) \otimes (1,0)$$

Thus, $S(w,w') = (1,0)$ *is also a graded simulation. Since any graded simulation is bounded by the pair* $(1,0)$ *we have that* $S(w,w') = (1,0)$ *is the biggest graded simulation between the two paraconsistent models. Actually, while the notion of a crisp simulation is bivalent, that is, either states are or are not similar, in a graded simulation that is not the case. It is often possible to define other graded simulations that satisfy all conditions. For example, the relation* $S'(w,w') = (0.5, 0.5)$ *is also a graded simulation between the models that satisfies (31) and (32).*

Example 10. *Consider now the following paraconsistent model assuming* $(1,0)$ *as the value of proposition* p *in all states.*



The following relation is a graded simulation

$$S^G(w_1,v_1) = S^G(w_2,v_2) = S^G(w_4,v_3) = S^G(w_5,v_4) = (1,0)$$
$$S^G(w_3,v_2) = (0.2,1)$$

Transitions from state w_3 *to* w_5 *and from state* v_2 *to* v_4 *are similar with equal negative weights, i.e.* $R_c^-(w_3,w_5) = R'^-_c(v_2,v_4)$. *However, despite the transitions being similar, as per the definition of a crisp simulation, there is no crisp simulation* S *such that* $(w_3,v_2) \in S$. *However, when considering a graded simulation, a different perspective emerges. There is complete evidence that the states are not similar, as witnessed by the crisp relation, while simultaneously indicating some evidence,* 0.2, *of their similarity.*

Let us take this question a little further and investigate under which conditions graded simulation coincides with the crisp simulation. Consider paraconsistent models $M = \langle W, R, V\rangle$, $M' = \langle W', R', V'\rangle$ and a crisp simulation S between them. Consider any pair of states $(w,w') \in W \times W'$ such that $(w,w') \in S$. Is there a graded simulation $S^G : W \times W' \to A \times A$ such that $S^G(w,w') = (1,0)$? In other words, if two states are related by a crisp simulation, is there a graded simulation relating them with weight $(1,0)$, that is, is there complete evidence that the states are similar and absolute minimal evidence that they are not similar?

For some proposition symbol p and action symbol a, consider $V(w,p) = (\gamma,\delta)$, $V'(w',p) = (\gamma',\delta')$, $R_a(w,v) = (\alpha,\beta)$ and $R'_a(w',v') = (\alpha',\beta')$. Since $(w,w') \in S$, it follows that $(\gamma,\delta) \preccurlyeq (\gamma',\delta')$ and $(\alpha,\beta) \preccurlyeq (\alpha',\beta')$, as well as, $(v,v') \in S$. Let us now evaluate if it is possible to have a graded simulation such that $S^G(w,w') = (1,0)$, or without loss of generality, $S^G(w,w') \preccurlyeq (1,0)$. Relation S^G must satisfy condition (31),

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

$$
\begin{aligned}
S^G(w,w') &\preccurlyeq V(w,p) \Rightarrow V'(w',p)\\
&\{\text{defn of } \Rightarrow\}\\
&= ((\gamma \to \gamma') \wedge (\delta' \to \delta), \gamma \wedge \delta')\\
&\{\text{Hypothesis: } \gamma \leq \gamma' \text{ and } \delta \geq \delta'\}\\
&= (1, \gamma \wedge \delta')
\end{aligned}
$$

Thus, from condition (31), $S^G(w,w') \preccurlyeq (1,0)$ implies $\gamma = 0$ or $\delta' = 0$. Relation S^G must also satisfy condition (32),

$$S^G(w,w') \otimes R_a(w,v) \preccurlyeq R'_a(w',v') \otimes S^G(v,v')$$

Using adjunction (17), $S^G(w,w') \preccurlyeq R_a(w,v) \Rightarrow (R'_a(w',v') \otimes S^G(v,v'))$. Thus,

$$
\begin{aligned}
S^G(w,w') &\preccurlyeq R_a(w,v) \Rightarrow (R'_a(w',v') \otimes S^G(v,v'))\\
&\{S^G(v,v') \preccurlyeq (1,0) \text{ by (19) and (18)}\}\\
&\preccurlyeq R_a(w,v) \Rightarrow (R'_a(w',v') \otimes (1,0))\\
&\{\text{defn of } \otimes\}\\
&= R_a(w,v) \Rightarrow (\alpha', (\alpha' \to 0) \wedge \beta')\\
&\{\text{defn of } \Rightarrow\}\\
&= ((\alpha \to \alpha') \wedge (((\alpha' \to 0) \wedge \beta') \to \beta), \alpha \wedge \alpha' \to 0 \wedge \beta')\\
&\{\text{Hypothesis: } \alpha \leq \alpha' \text{ and } \beta' \leq \beta\}\\
&= (1, \alpha \wedge (\alpha' \to 0) \wedge \beta')
\end{aligned}
$$

Thus, from condition (32), $S^G(w,w') \preccurlyeq (1,0)$ implies $\alpha = 0$ or $\beta' = 0$ or $\alpha' > 0$.

In conclusion, if $(w,w') \in S$ there is a graded simulation S^G such that $S^G(w,w') = (1,0)$ only when $V^+(w,p) = 0$ or $V'^-(w',p) = 0$ and either $R_a^+(w,v) = 0$ or $R'^-_a(w',v') = 0$ or $R'^+_a(w',v') > 0$.

This discussion sheds light on a somehow surprising conclusion: that identity fails to be a simulation. Actually, consider the following paraconsistent models,



with $V(w,p) = V'(w',p) = (0.2, 0.2)$. The relation $S^G(w,w') = (1,0)$ is not a graded simulation because $(1,0) \not\preccurlyeq (0.2,0.2) \Rightarrow (0.2,0.2) = (1,0.2)$

Definition 15. *Let $M = \langle W, R, V\rangle$ and $M' = \langle W', R', V'\rangle$ be paraconsistent models over a iMTL-algebra **A** and a set Act of actions. A relation $B^G : W \times W' \to A \times A$ is a **graded bisimulation** if the following conditions hold for all $p \in$ Prop, $a \in$ Act and all possible values of free variables:*

$$B^G(w,w') \preccurlyeq (V(w,p) \Leftrightarrow V'(w',p)) \tag{33}$$

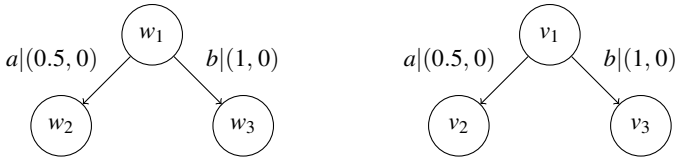
$$\exists v' \in W',\ (B^G(w,w') \otimes R_a(w,v)) \preccurlyeq (R'_a(w',v') \otimes B^G(v,v')) \tag{34}$$

$$\exists v \in W,\ (B^G(w,w') \otimes R'_a(w',v')) \preccurlyeq (R_a(w,v) \otimes B^G(v,v')) \tag{35}$$

Again a crisp bisimulation is not, in general, a particular case of a graded one, as illustrated in the following examples.

Example 11. *Consider the paraconsistent models depicted below where $V(w_1,p) = V'(v_1,p) = (0,1)$, $V(w_2,p) = V'(v_2,p) = (1,0)$ and $V(w_3,p) = V'(v_3,p) = (0,0)$.*

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press



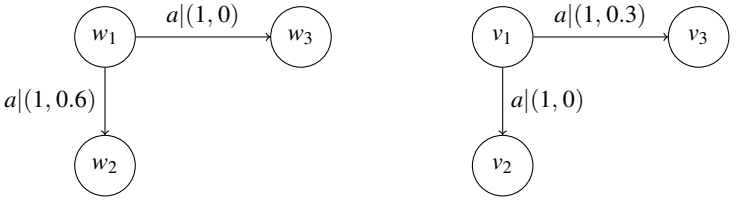
Clearly, $B = \{(w_1, v_1), (w_2, v_2), (w_3, v3)\}$ *is a crisp bisimulation. Informally,* B *can be written as a graded bisimulation* $B^G : W \times W \to A \times A$ *where* $B^G(w_1, v_1) = B^G(w_2, v_2) = B^G(w_3, v_3) = (1, 0)$. *Notice that,* B^G *satisfies condition (33),*

$$(1,0) \preccurlyeq (0,1) \Leftrightarrow (0,1) = (1,0)$$
$$(1,0) \preccurlyeq (1,0) \Leftrightarrow (1,0) = (1,0)$$
$$(1,0) \preccurlyeq (0.5,0) \Leftrightarrow (0.5,0) = (1,0)$$

and conditions (34) and (35),

$$(1,0) \otimes (0.5,0) = (0.5,0) \preccurlyeq (0.5,0) \otimes (1,0) = (0.5,0)$$
$$(1,0) \otimes (1,0) = (1,0) \preccurlyeq (1,0) \otimes (1,0) = (1,0)$$

Example 12. *For an example in the opposite direction, consider*



where $V(w_1, p) = V'(v_1, p) = (1, 1)$, $V(w_2, p) = V'(v_3, p) = (1, 0.5)$ *and* $V(w_3, p) = V'(v_2, p) = (0.8, 0.8)$. *A graded bisimulation* B^G *between the above paraconsistent models can be computed as follows:*

- *From condition (33),*

$$B^G(w_1, v_1) \preccurlyeq (1,1) \Leftrightarrow (1,1) = (1,1)$$
$$B^G(w_2, v_2) \preccurlyeq (1,0.5) \Leftrightarrow (0.8,0.8) = (0.5,0.8)$$
$$B^G(w_3, v_3) \preccurlyeq (0.5,0.8)$$
$$B^G(w_2, v_3) \preccurlyeq (1,0.5)$$
$$B^G(w_3, v_2) \preccurlyeq (1,0.8)$$

- *From condition (34), on the other hand,*

$$B^G(w_1, v_1) \otimes R_a(w_1, w_2) \preccurlyeq R'_a(v_1, v_2) \otimes B^G(w_2, v_2) \preccurlyeq (1,0) \otimes (0.5,0.8) = (0.5,0)$$

  Thus, $B^{G+}(w_1, v_1) \wedge 1 = B^{G+}(w_1, v_1) \leq 0.5$. *Similarly,*

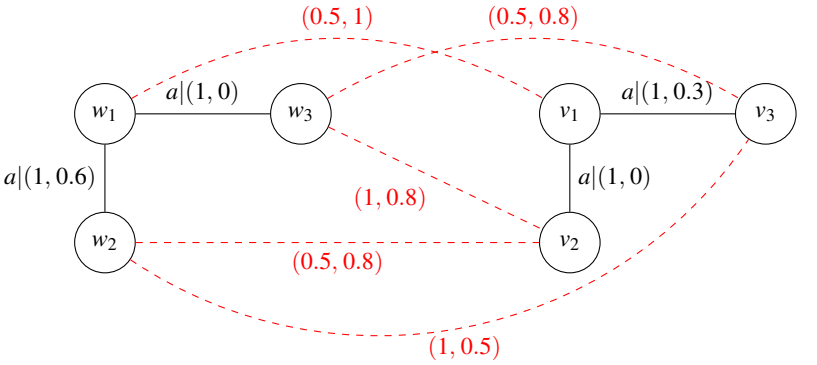
$$B^G(w_1, v_2) \otimes R_a(w_1, w_2) \preccurlyeq R'_a(v_2, v_2) \otimes B^G(w_2, v_2) \preccurlyeq (0,1) \otimes (0.5,0.8) = (0,1)$$

  Thus, $B^G(w_1, v_2) = (0, 1)$. *A similar reasoning yields* $B^G(w_1, v_3) = (0, 1)$.
- *Finally, from condition (35), we conclude that* $B^G(w_2, v_1) = B^G(w_3, v_1) = (0, 1)$.

Therefore, one possible graded bisimulation between the paraconsistent models is $B^G(w_1, v_1) = (0.5, 1)$, $B^G(w_2, v_2) = (0.5, 0.8)$, $B^G(w_2, v_3) = (1, 0.5)$, $B^G(w_3, v_2) = (1, 0.8)$ *and* $B^G(w_3, v_3) =$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

$(0.5, 0.8)$ *since it satisfies conditions (33)-(35). The figure below depicts this graded bisimulation through red dashed lines.*



Once again, we may ask which conditions are necessary for crisp and graded bisimulations to match. Consider paraconsistent models $M = \langle W, R, V\rangle$, $M' = \langle W', R', V'\rangle$ and a crisp bisimulation B between them. Consider any pair of states $w \in W$, $w' \in W'$ such that $(w, w') \in B$. Is there a graded bisimulation $B^G : W \times W' \to A \times A$ such that $B^G(w, w') = (1, 0)$?

If $(w, w') \in B$ there is a graded bisimulation B^G such that $B^G(w, w') = (1, 0)$ only if $V^+(w, p) = 0$ or $V^-(w, p) = 0$ and either $R_a^+(w, v) = 0$ or $R'^-_a(w', v') = 0$ or $R'^+_a(w', v') > 0$ and either $R'^+_a(w', v') = 0$ or $R_a^-(w, v) = 0$ or $R_a^+(w, v) > 0$.

5. A Modal Logic for Paraconsistent Models

The Logic

We are now able to introduce a modal logic $\mathrm{P}(\boldsymbol{A})$ to specify properties of paraconsistent structures parametric on an iMTL-algebra $\boldsymbol{A}$.

Definition 16. *Given an iMTL-algebra* $\boldsymbol{A}$, *a set of proposition symbols* Prop *and a set of action symbols* Act, *the set of sentences* $\mathsf{Sen}(\mathrm{Prop}, \mathrm{Act})$ *of* $\mathrm{P}(\boldsymbol{A})$ *is generated by the following grammar:*

$$\varphi := \bot \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle a\rangle\varphi$$

where $p \in \mathrm{Prop}$ *and* $a \in \mathrm{Act}$. *As usual, the following abbreviations are considered:* $\top = \neg\bot$, $\varphi \vee \varphi' = \neg(\neg\varphi \wedge \neg\varphi')$, $\varphi \triangleright \varphi' = \neg\varphi \vee \varphi' = \neg(\varphi \wedge \neg\varphi')$, $\varphi \diamond \varphi' = (\varphi \triangleright \varphi') \wedge (\varphi' \triangleright \varphi)$ *and* $[a]\varphi = \neg\langle a\rangle\neg\varphi$.

For a given paraconsistent model M, which the underlying iMTL-algebra is $\boldsymbol{A}$, the satisfaction relation in $\mathrm{P}(\boldsymbol{A})$ associates to M and a sentence φ a pair of weights $(\alpha, \beta) \in A \times A$. As before, the positive weight α stands for the evidence that φ holds in M while the negative weight β stands for the evidence of the opposite fact.

Definition 17. *Let* $\boldsymbol{A} = \langle A, \wedge, \vee, 1, 0, \to\rangle$ *be an iMTL-algebra and* $M = \langle W, R, V\rangle$ *a paraconsistent model. The satisfaction relation,*

$$\models : M \times \mathsf{Sen}(\mathrm{Prop}, \mathrm{Act}) \to A \times A$$

is given by

$$(M \models \varphi) = \bigwedge_{w \in W} (M, w \models \varphi)$$

where relation $\models$ *is recursively defined as follows,*

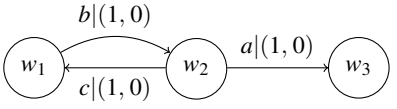
https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

- $(M, w \models \bot) = (0, 1)$
- $(M, w \models p) = V(w, p)$
- $(M, w \models \neg\varphi) = /\!\!/(M, w \models \varphi)$
- $(M, w \models \varphi \wedge \varphi') = (M, w \models \varphi) \overset{\wedge}{\wedge} (M, w \models \varphi')$
- $(M, w \models \langle a \rangle \varphi) = \overset{\vee}{\underset{v \in W}{\vee}} \Big(R_a(w, v) \otimes (M, v \models \varphi) \Big)$

Informally, sentence φ is evaluated at each state of the model M and (a, b) is the result of the conjunction of each of these evaluations. Hence, a sentence φ is said to be *valid* if, for any state $w \in W$, $(M, w \models \varphi) = (1, 0)$, that is there is complete evidence that φ holds at state w and absolute minimal evidence that it does not hold. Note how the operators characterizing the twist-structure introduced earlier in the paper reappear in the satisfaction relation to take care of pairs of weights being computed.

The next example emphasizes the fact that paraconsistent models generalize transition structures in which the information is consistent, that is, in which the transition relation and the valuation function are either $(1, 0)$ or $(0, 1)$.

Example 13. *Consider the set* $\mathrm{Act} = \{a, b, c\}$ *of action symbols and the set* $\mathrm{Prop} = \{p\}$ *of propositions. Let* M *be the following paraconsistent model over the Gödel algebra with* $V(w_1, p) = V(w_3, p) = (1, 0)$ *and* $V(w_2, p) = (0, 1)$.

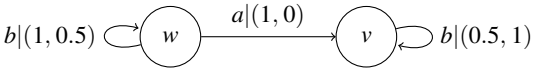


Then,

$$
\begin{aligned}
(M, w_2 \models \langle c \rangle p \wedge [a] \neg p) &= \Big(M, w_2 \models \langle c \rangle p \Big) \overset{\wedge}{\wedge} \Big(M, w_2 \models [a] \neg p \Big) \\
&= \Big((R_c(w_2, w_1) \otimes (M, w_1 \models p) \Big) \overset{\wedge}{\wedge} \Big(M, w_2 \models \neg \langle a \rangle \neg \neg p \Big) \\
&= \Big((1, 0) \otimes (1, 0) \Big) \overset{\wedge}{\wedge} /\!\!/ \Big(M, w_2 \models \langle a \rangle p \Big) \\
&= (1 \wedge 1, 1 \to 0 \wedge 1 \to 0) \overset{\wedge}{\wedge} /\!\!/ \Big(R_a(w_2, w_3) \otimes (M, w_3 \models p) \Big) \\
&= (1, 0) \overset{\wedge}{\wedge} /\!\!/ \Big((1, 0) \otimes (0, 1) \Big) = (1, 0) \overset{\wedge}{\wedge} /\!\!/ (0, 1) \\
&= (1 \wedge 1, 0 \vee 0) \\
&= (1, 0)
\end{aligned}
$$

Informally, this can be read as follows: at state w_2 *there is a transition through action* c *to a state where* p *holds and all transitions through action* a *reach a state where* p *does not hold. As expected, the result is a consistent pair of weights* $(1, 0)$*, that is, there is complete evidence,* 1*, that this sentence holds at state* w_2 *and absolute non evidence,* 0*, evidence that it does not hold.*

Example 14. *Consider another paraconsistent model over the Gödel algebra, with* $\mathrm{Act} = \{a, b\}$ *and* $\mathrm{Prop} = \{p, q, r\}$



https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

with, $V(w,p) = (1,1)$, $V(w,q) = V(v,p) = (0,0.5)$, $V(w,r) = V(v,r) = (0.5,0.5)$ *and* $V(v,q) = (0,0)$. *Then,*

$$\begin{aligned}(M, w \models r \triangleright (p \vee q)) &= (M, w \models \neg r \vee (p \vee q)) \\ &= (M, w \models \neg(\neg\neg r \wedge \neg(p \vee q))) \\ &= /\!\!/ \left(M, w \models r \wedge \neg(p \vee q)\right) \\ &= /\!\!/ \left((M, w \models r) \overset{\wedge}{\wedge} (M, w \models \neg(p \vee q))\right) \\ &= /\!\!/ \left(V(w,r) \overset{\wedge}{\wedge} /\!\!/(M, w \models p \vee q)\right) \\ &= /\!\!/ \left(V(w,r) \overset{\wedge}{\wedge} (M, w \models \neg p) \overset{\wedge}{\wedge} (M, w \models \neg q)\right) \\ &= /\!\!/ \left(V(w,r) \overset{\wedge}{\wedge} /\!\!/V(w,p) \overset{\wedge}{\wedge} /\!\!/V(w,q)\right) \\ &= /\!\!/ \left((0.5,0.5) \overset{\wedge}{\wedge} (1,1) \overset{\wedge}{\wedge} (0.5,0)\right) \\ &= /\!\!/ (0.5 \wedge 1 \wedge 0.5, 0.5 \vee 1 \vee 0) = /\!\!/(0.5,1) \\ &= (1,0.5)\end{aligned}$$

Similarly, $(M, v \models r \triangleright (p \vee q)) = (0.5,0)$. *Therefore,*

$$M \models r \triangleright (p \vee q) = (1,0.5) \overset{\wedge}{\wedge} (0.5,0) = (0.5,0.5)$$

giving 0.5 *evidence that the sentence* $r \triangleright (p \vee q)$ *holds and does not hold in this paraconsistent model.*

Properties

This subsection presents some preliminary results on soundness of paraconsistent Kripke models. The crucial ingredient is that we generalize the classical notion of soundness, where both premises and conclusions are consistently true, and instead, introduce a graded notion of soundness, where both premises and conclusions may encompass vagueness and contradictions.

As stated in the introduction, a similar motivation can be found in Caicedo and Rodríguez (2010) where the authors introduce a notion of logical consequence, denoted by $\models_{\leq \mathrm{GK}}$, for Gödel Kripke models GK. They then proceed to prove the soundness of the $\Box$-fragment (and similarly of the $\Diamond$-fragment). However, it is important to note that the main results highlighted in Caicedo and Rodríguez (2010) pertain to countable theories, where the notions of classical logical consequence $\models_{GK}$ and $\models_{\leq \mathrm{GK}}$ are equivalent. Our approach for paraconsistent Kripke models is similar, as we also consider classical and graded notions of soundness, the latter of which we term as *g-soundness*. This graded variant adjusts the classical definition of soundness to accommodate vague and inconsistent evaluations. It is evident that *g-soundness* is a more relaxed notion than classical soundness, thus implying that g-soundness entails classical soundness. However, the converse is not true, leading us to explore soundness results in Kripke paraconsistent models where g-soundness and classical soundness are not equivalent. In summary, the main results documented are:

- Modus Ponens (MP) and Modal Generalization (Gen) rules are classically sound but not g-sound;
- Distribution of the possibility operator over $\vee$ and $\wedge$, rules (36) and (37), are sound and g-sound.
- Distribution of the necessity operator over $\triangleright$, rule (K), is neither sound nor g-sound.

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Let us start by revisiting the classical notion of soundness and then introduce its graded counterpart, extending upon the classical understanding. Following that, we will conduct a preliminary investigation into the properties of soundness for paraconsistent Kripke models.

Definition 18. *A rule*

$$\frac{\varphi_1 \ \dots \ \varphi_n}{\varphi}$$

is sound if for any paraconsistent Kripke structure $M = \langle W, R, V\rangle$,

$$\text{if } (M \models \varphi_i) = (1,0) \text{ for } i \in \{1,\dots,n\}, \text{ then } (M \models \varphi) = (1,0)$$

The following, alternative, notion of soundness accommodates vague and inconsistent evaluations, implying that from the evidence of truth and falsity of premises, it is possible to draw conclusions whose evidence of truth is higher and of falsity lower.

Definition 19. *A rule*

$$\frac{\varphi_1 \ \dots \ \varphi_n}{\varphi}$$

is g-sound if for any paraconsistent Kripke structure $M = \langle W, R, V\rangle$,

$$\bigwedge_{i=1}^{n} (M \models \varphi_i) \preccurlyeq (M \models \varphi)$$

The following theorem is crucial for subsequent steps because it assures that g-soundness generalizes classical soundness.

Theorem 1. *If a rule is g-sound, then it is sound.*

Proof. Consider a g-sound rule whose premises are $\varphi_1 \ \dots \ \varphi_n$ and conclusion φ. We must prove that this rule is sound. Let us fix a paraconsistent Kripke structure $M = \langle W, R, V\rangle$ such that the premises are consistently true, that is, $(M \models \varphi_i) = (1,0)$ for any $i \in \{1,\dots,n\}$. By hypothesis

$$(M \models \varphi_1) \wedge (M \models \varphi_2) \wedge \dots \wedge (M \models \varphi_n) \preccurlyeq (M \models \varphi)$$

Since the premises are consistently true it follows, $(1,0) \preccurlyeq (M \models \varphi)$. Trivially, this implies that $(M \models \varphi) = (1,0)$. Therefore, any rule that is sound is a particular case of a rule that is g-sound whenever the premises are consistently true. □

We proceed by examining classical and graded soundness results in paraconsistent model and present counterexamples for rules that are not sound. Note that by Theorem 1, g-soundness implies classical soundness, but the converse does not hold in general.

Theorem 2. *The following rules*

$$\frac{\varphi,\ \varphi \rhd \varphi'}{\varphi'} \qquad \textbf{(MP)}$$

$$\frac{\varphi}{[a]\varphi} \qquad \textbf{(Gen)}$$

are sound.

Proof. Let $M = \langle W, R, V\rangle$ be any paraconsistent model.

- For (MP) let us assume that for any state $w \in W$,

$$(M, w \models \varphi) = (1,0) \text{ and } (M, w \models \varphi \rhd \varphi') = (1,0)$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

By definition of $\models$,

$$(M, w \models \varphi \triangleright \varphi') = (M, w \models \neg\varphi) \overset{\vee}{\vee} (M, w \models \varphi') = (1, 0)$$

By hypothesis $(M, w \models \neg\varphi) = /\!\!/(M, w \models \varphi) = (0, 1)$. Finally, using the definition of $\overset{\vee}{\vee}$, it follows that $(M, w \models \varphi') = (1, 0)$.

- For (Gen) let us assume that $(M, w \models \varphi) = (1, 0)$ for any $w \in W$. Then,

$$\begin{aligned}
&M, w \models [a]\varphi \\
=&\{\text{defn. } [a]\varphi = \neg\langle a\rangle\neg\varphi\} \\
&M, w \models \neg\langle a\rangle\neg\varphi \\
=&\{\text{defn. of } \models\} \\
&/\!\!/\left(\overset{\vee}{\vee}_{v\in W}\left(R_a(w, v) \otimes (M, v \models \neg\varphi)\right)\right) \\
=&\{\text{Hypothesis}\} \\
&/\!\!/\left(\overset{\vee}{\vee}_{v\in W}\left(R_a(w, v) \otimes (0, 1)\right)\right) \\
=&\{\text{defn. of } \otimes\} \\
&/\!\!/ (0, 1) \\
=&\{\text{defn. of } /\!\!/\} \\
&(1, 0)
\end{aligned}$$

□

Theorem 3. *The following rules*

$$\frac{\langle a\rangle(\varphi \vee \varphi')}{\langle a\rangle\varphi \vee \langle a\rangle\varphi'} \tag{36}$$

$$\frac{\langle a\rangle(\varphi \wedge \varphi')}{\langle a\rangle\varphi \wedge \langle a\rangle\varphi'} \tag{37}$$

are g-sound.

Proof. Let $M = \langle W, R, V\rangle$ be any paraconsistent model.

- For (36) note that for any $w \in W$,

$$\begin{aligned}
&M, w \models \langle a\rangle(\varphi \vee \varphi') \\
=&\{\text{defn. of } \models\} \\
&\overset{\vee}{\vee}_{v\in W}\left(R_a(w, v) \otimes (M, v \models \varphi \vee \varphi')\right) \\
=&\{\text{defn. of } \models\} \\
&\overset{\vee}{\vee}_{v\in W}\left(R_a(w, v) \otimes \left((M, v \models \varphi) \overset{\vee}{\vee} (M, v \models \varphi')\right)\right) \\
=&\{\text{Property (21)}\} \\
&\overset{\vee}{\vee}_{v\in W}\left(\left(R_a(w, v) \otimes (M, v \models \varphi)\right) \overset{\vee}{\vee} \left(R_a(w, v) \otimes (M, v \models \varphi')\right)\right) \\
=&\{\text{Distribution of } \overset{\vee}{\vee}\} \\
&\overset{\vee}{\vee}_{v\in W}\left(R_a(w, v) \otimes (M, v \models \varphi)\right) \overset{\vee}{\vee} \overset{\vee}{\vee}_{v\in W}\left(R_a(w, v) \otimes (M, v \models \varphi')\right) \\
=&\{\text{defn. of } \models\} \\
&M, w \models \langle a\rangle\varphi \vee \langle a\rangle\varphi'
\end{aligned}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Hence, $(M \models \langle a \rangle(\varphi \vee \varphi')) \preccurlyeq (M \models \langle a \rangle\varphi \vee \langle a \rangle\varphi')$.

- For (37) note that for any $w \in W$,

$$
\begin{aligned}
& M, w \models \langle a \rangle(\varphi \wedge \varphi') \\
=& \{\text{defn. of } \models\} \\
& \bigvee_{v \in W}^{\vee} \Big(R_a(w,v) \otimes (M, v \models \varphi \wedge \varphi') \Big) \\
=& \{\text{defn. of } \models\} \\
& \bigvee_{v \in W}^{\vee} \Big(R_a(w,v) \otimes \Big((M, v \models \varphi) \overset{\wedge}{\wedge} (M, v \models \varphi') \Big) \Big) \\
\preccurlyeq& \{\text{Property (22)}\} \\
& \bigvee_{v \in W}^{\vee} \Big(\Big(R_a(w,v) \otimes (M, v \models \varphi) \Big) \overset{\wedge}{\wedge} \Big(R_a(w,v) \otimes (M, v \models \varphi') \Big) \Big) \\
=& \{\text{Distribution of } \overset{\vee}{\vee} \text{ over } \overset{\wedge}{\wedge}\} \\
& \bigvee_{v \in W}^{\vee} \Big(R_a(w,v) \otimes (M, v \models \varphi) \Big) \overset{\wedge}{\wedge} \bigvee_{v \in W}^{\vee} \Big(R_a(w,v) \otimes (M, v \models \varphi') \Big) \\
=& \{\text{defn. of } \models\} \\
& (M, w \models \langle a \rangle\varphi) \overset{\wedge}{\wedge} (M, w \models \langle a \rangle\varphi') \\
=& \{\text{defn. of } \models\} \\
& M, w \models \langle a \rangle\varphi \wedge \langle a \rangle\varphi'
\end{aligned}
$$

Hence, $(M \models \langle a \rangle(\varphi \wedge \varphi') \preccurlyeq (M \models \langle a \rangle\varphi \wedge \langle a \rangle\varphi')$. □

Corollary 1. *The following rules*

$$\frac{\langle a \rangle(\varphi \vee \varphi')}{\langle a \rangle\varphi \vee \langle a \rangle\varphi'} \qquad \frac{\langle a \rangle(\varphi \wedge \varphi')}{\langle a \rangle\varphi \wedge \langle a \rangle\varphi'}$$

are sound.

Proof. This corollary is a direct consequence of Theorems 3 and 1. □

We conclude this subsection by revisiting the rules outlined in Theorem 2 alongside axiom (**K**) and show how these rules are not g-sound.

Lemma 5. *The following rules*

$$\frac{\varphi,\ \varphi \rhd \varphi'}{\varphi'} \quad (\mathbf{MP}) \qquad \frac{\varphi}{[a]\varphi} \quad (\mathbf{Gen}) \qquad \frac{[a](\varphi \rhd \varphi')}{[a]\varphi \rhd [a]\varphi'} \quad (\mathbf{K})$$

are not g-sound.

Proof.

- Consider a paraconsistent model $M = \langle W, R, V \rangle$ such that at some state $w \in W$, $(M, w \models \varphi) = (0.3, 0.4)$ and $(M, w \models \varphi') = (0.2, 0)$. Then,

$$(M \models \varphi) \overset{\wedge}{\wedge} (M, w \models \varphi \rhd \varphi') = (0.3, 0.4) \overset{\wedge}{\wedge} \Big((0.4, 0.3) \overset{\vee}{\vee} (0.2, 0) \Big) = (0.3, 0.4)$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

However,

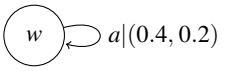
$$(0.3, 0.4) \not\preccurlyeq (0.2, 0)$$

.

Hence, the following assertion does not always hold.

$$\Big((M \models \varphi) \overset{\wedge}{\wedge} (M, w \models \varphi \triangleright \varphi')\Big) \preccurlyeq (M, w \models \varphi')$$

- For modal generalization consider the following paraconsistent model $M = \langle W, R, V \rangle$ depicted below.



with $(M, w \models \varphi) = (0.3, 0.5)$, for some sentence φ. Then,

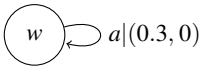
$$\begin{aligned} & M, w \models [a]\varphi \\ =& M, w \models \neg\langle a\rangle\neg\varphi \\ =& /\!\!/ (R_a(w, w) \otimes /\!\!/(M, w \models \varphi)) \\ =& /\!\!/ ((0.4, 0.2) \otimes (0.5, 0.3)) \\ =& (0.2, 0.4) \end{aligned}$$

However,

$$(M, w \models \varphi) \not\preccurlyeq (M, w \models [a]\varphi)$$

.

- Finally, we explore the soudness of axiom (K) proving it is not classically sound by providing a counterexample such that $(M, w \models [a](\varphi \triangleright \varphi')) = (1, 0)$ but $(M, w \models [a]\varphi \triangleright [a]\varphi') \neq (1, 0)$. Consider the paraconsistent model $M = \langle W, R, V \rangle$ depicted below.



with $(M, w \models \varphi') = (0, 0.5)$ and $(M, w \models \varphi') = (0, 0)$, for some sentences φ and φ'. Then,

$$\begin{aligned} & M, w \models [a](\varphi \triangleright \varphi') \\ =& M, w \models \neg\langle a\rangle(\varphi \wedge \neg\varphi') \\ =& /\!\!/ \Big(R_a(w, w) \otimes ((M, w \models \varphi) \overset{\wedge}{\wedge} /\!\!/(M, w \models \varphi'))\Big) \\ =& /\!\!/ ((0.3, 0) \otimes (0 \wedge 0, 0.5 \vee 0)) \\ =& (1, 0) \end{aligned}$$

However,

$$\begin{aligned} & M, w \models [a]\varphi \triangleright [a]\varphi' \\ =& M, w \models \langle a\rangle\neg\varphi \vee \neg\langle a\rangle\neg\varphi' \\ =& (M, w \models \langle a\rangle\neg\varphi) \overset{\vee}{\vee} (M, w \models \neg\langle a\rangle\neg\varphi') \\ =& \Big(R_a(w, w) \otimes /\!\!/(M, w \models \varphi)\Big) \overset{\vee}{\vee} /\!\!/\Big(R_a(w, w) \otimes /\!\!/(M, w \models \varphi')\Big) \\ =& ((0.3, 0) \otimes (0.5, 0)) \overset{\vee}{\vee} /\!\!/((0.3, 0) \otimes (0, 0)) \\ =& (0.3, 0) \overset{\vee}{\vee} (0, 0) \\ =& (0.3, 0) \end{aligned}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Hence, the rule

$$\frac{[a](\varphi \triangleright \varphi')}{[a]\varphi \triangleright [a]\varphi'}$$

is not sound and by the contrapositive of Theorem 1 it is also not g-sound. □

Validity Preservation Over Simulation

This section addresses another question relevant for the characterization of the proposed logic: the preservation of modal validity over different forms of simulation in paraconsistent models. We start with the crisp case.

Theorem 4. *Let $M = \langle W, R, V\rangle$ and $M' = \langle W', R', V'\rangle$ be two paraconsistent models over a iMTL-algebra $\boldsymbol{A}$, and $S \subseteq W \times W'$ be a crisp simulation. Then, for every $(w, w') \in S$ and any sentence $\varphi \in \mathsf{Sen}^{+}(\mathrm{Prop}, \mathrm{Act})$:*

$$(M, w \models \varphi) \preccurlyeq (M', w' \models \varphi) \tag{38}$$

where $\mathsf{Sen}^{+}(\mathrm{Prop}, \mathrm{Act})$ *is the positive fragment of* $\mathrm{P}(\boldsymbol{A})$.

Proof. The proof is by induction over the structure of sentences.

- Case $\bot$ is trivial, using the definition of $\models$ $(M, w \models \bot) = (0, 1) = (M', w' \models \bot)$. Thus,
$$(M, w \models \bot) \preccurlyeq (M', w' \models \bot)$$
- For $p \in \mathrm{Prop}$ the result follows from the fact that S is a simulation hence, $V(w, p) \preccurlyeq V'(w', p)$, using the definition of $\models$, $(M, w \models p) \preccurlyeq (M', w' \models p)$.
- For $\varphi_1 \wedge \varphi_2$ we reason,
$$\begin{aligned}
&(M, w \models \varphi_1 \wedge \varphi_2)\\
&= \{\text{defn of } \models\}\\
&\quad (M, w \models \varphi_1) \mathbin{\overset{\wedge}{\wedge}} (M, w \models \varphi_2)\\
&\preccurlyeq \{\text{Induction Hypothesis twice}\}\\
&\quad (M', w' \models \varphi_1) \mathbin{\overset{\wedge}{\wedge}} (M', w' \models \varphi_2)\\
&= \{\text{defn of } \models\}\\
&\quad (M', w' \models \varphi_1 \wedge \varphi_2)
\end{aligned}$$
- For $\langle a\rangle\varphi$,
$$\begin{aligned}
&M, w \models \langle a\rangle\varphi\\
&= \{\text{defn of } \models\}\\
&\quad \mathop{\overset{\vee}{\vee}}_{v \in W} \Big(R_a(w, v) \otimes (M, v \models \varphi) \Big)\\
&\preccurlyeq \{\star\}\\
&\quad \mathop{\overset{\vee}{\vee}}_{v' \in W'} \Big(R'_a(w', v') \otimes (M', v' \models \varphi) \Big)\\
&= \{\text{defn of } \models\}\\
&\quad M', w' \models \langle a\rangle\varphi
\end{aligned}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

The step labeled by $*$ in the last proof, is justied as follows: since S is a simulation, there exists $v' \in W'$ such that $R_a(w,v) \preccurlyeq R'_a(w',v')$ and by hypothesis $(M, v \models \varphi) \preccurlyeq (M', v' \models \varphi)$. Using (19),

$$R_a(w,v) \otimes (M, v \models \varphi) \preccurlyeq R'_a(w',v') \otimes (M', v' \models \varphi)$$

Since $\bigvee\limits_{\vee}$ is monotone, we conclude that,

$$\bigvee_{v \in W} \left(R_a(w,v) \otimes (M, v \models \varphi) \right) \preccurlyeq \bigvee_{v' \in W'} \left(R'_a(w',v') \otimes (M', v' \models \varphi) \right)$$

□

Not all sentences are, however, preserved by a crisp simulation. Typical examples are $\neg\varphi$, $\varphi_1 \triangleright \varphi_2$ and $[a]\varphi$, as shown by the following counterexamples. Consider the following paraconsistent models $M = \langle W, R, V \rangle$ and $M' = \langle W', R', V' \rangle$.



where $R_a(w,w) \preccurlyeq R'_a(w',w')$ and $V(w,p) = (0.3, 0.8) \preccurlyeq (0.6, 0.7) = V'(w',p)$ and $V(w,q) = (0.1, 0.4) \preccurlyeq (0.2, 0.4) = V'(w',q)$. Hence, $S = \{(w,w')\}$ is a simulation.

- For $\neg p$ we have that $(M, w \models \neg p) = /\!/ V(w,p) = (0.8, 0.3)$ and $(M', w' \models \neg p) = /\!/ V'(w',p) = (0.7, 0.6)$. That is,

$$(M, w \models \neg p) \not\preccurlyeq (M', w' \models \neg p)$$

- For $p \triangleright q$ we have that $(M, w \models p \triangleright q)^- = 0.3$ and $(M', w' \models p \triangleright q)^- = 0.4$. Thus,

$$(M, w \models p \triangleright q) \not\preccurlyeq (M', w' \models p \rightarrow q)$$

- For $[a]p$, we have that $(M, w \models [a]p)^- = 0$ and $(M', w' \models [a]p)^- = 0.7$. Thus,

$$(M, w \models [a]p) \not\preccurlyeq (M', w' \models [a]p)$$

Theorem 5. *Let $M = \langle W, R, V \rangle$ and $M' = \langle W', R', V' \rangle$ be two paraconsistent models over a iMTL-algebra $\boldsymbol{A}$ and $S^G : W \times W' \to A \times A$ be a graded simulation. Then, for every $w \in W$, $w' \in W'$ and any sentence $\varphi \in \mathsf{Sen}^+(\mathrm{Prop}, \mathrm{Act})$:*

$$S^G(w,w') \preccurlyeq \left((M, w \models \varphi) \Rightarrow (M', w' \models \varphi) \right) \tag{39}$$

where $\mathsf{Sen}^+(\mathrm{Prop}, \mathrm{Act})$ *is the positive fragment of* $\mathrm{P}(\boldsymbol{A})$.

Proof. The proof is by induction over the structure of sentences.

- Case $\bot$ is trivial. By definition of $\models$ and $\Rightarrow$

$$(M, w \models \bot) \Rightarrow (M', w' \models \bot) = (0,1) \Rightarrow (0,1) = (1,0)$$

and, since $S^G(w,w') \in A \times A$ we have that $S^G(w,w') \preccurlyeq (1,0)$.

- For $p \in \mathrm{Prop}$, recall that S^G is a graded simulation and therefore satisfies condition (31). That is,

$$S^G(w,w') \preccurlyeq V(w,p) \Rightarrow V'(w',p)$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Using the definition of $\models$

$$S^G \preccurlyeq (M, w \models p) \Rightarrow (M', w' \models p)$$

- For $\varphi_1 \wedge \varphi_2$ (where $\varphi_1,\ \varphi_2 \in \mathsf{Sen}^+(\mathrm{Prop}, \mathrm{Act})$) the proof is as follows:

$$\begin{aligned}
&(M, w \models \varphi_1 \wedge \varphi_2) \Rightarrow (M', w' \models \varphi_1 \wedge \varphi_2)\\
=\ &\{\text{defn } \models\}\\
&(M, w \models \varphi_1 \mathbin{\overset{\wedge}{\wedge}} M, w \models \varphi_2) \Rightarrow (M', w' \models \varphi_1 \mathbin{\overset{\wedge}{\wedge}} M', w' \models \varphi_2)\\
\succcurlyeq\ &\{\text{Property (30)}\}\\
&(M, w \models \varphi_1 \Rightarrow M', w' \models \varphi_1) \mathbin{\overset{\wedge}{\wedge}} (M, w \models \varphi_2 \Rightarrow M', w' \models \varphi_2)\\
\succcurlyeq\ &\{\text{Induction hypothesis and } \overset{\wedge}{\wedge} \text{ is idempotent}\}\\
&S^G(w, w')
\end{aligned}$$

- For $\langle a\rangle\varphi$, consider, without loss of generality, a state $v \in W$ such that

$$(M, w \models \langle a\rangle\varphi) \ =\ R_a(w, v) \otimes M, v \models \varphi \tag{40}$$

Since S^G is a graded simulation, by (32) there is $v' \in W'$ such that the following inequality is satisfied.

$$S^G(w, w') \otimes R_a(w, v) \preccurlyeq R'_a(w', v') \otimes S^G(v, v') \tag{41}$$

We will prove that,

$$S^G(w, w') \preccurlyeq (M, w \models \langle a\rangle\varphi) \Rightarrow (R'_a(w', v') \otimes M', v' \models \varphi) \tag{42}$$

Observe that,

$$(R'_a(w', v') \otimes M', v' \models \varphi) \preccurlyeq (M', w' \models \langle a\rangle\varphi) \tag{43}$$

Thus, by proving (42) resorting to property (18), we prove the case $\langle a\rangle\varphi$. Since S^G is a graded simulation, $S^G(v, v') \preccurlyeq (M, v \models \varphi) \Rightarrow (M', v' \models \varphi)$ through (18) in (41),

$$S^G(w, w') \otimes R_a(w, v) \preccurlyeq R'_a(w', v') \otimes ((M, v \models \varphi) \Rightarrow (M', v' \models \varphi))$$

Using Property (17),

$$S^G(w, w') \preccurlyeq R_a(w, v) \Rightarrow \Big(R'_a(w', v') \otimes ((M, v \models \varphi) \Rightarrow (M', v' \models \varphi))\Big)$$

Using Property (23),

$$S^G(w, w') \preccurlyeq R_a(w, v) \Rightarrow \Big((M, v \models \varphi) \Rightarrow (R'_a(w', v') \otimes (M', v' \models \varphi))\Big)$$

Using Property (24)

$$S^G(w, w') \preccurlyeq \Big(R_a(w, v) \otimes (M, v \models \varphi)\Big) \Rightarrow \Big(R'_a(w', v') \otimes (M', v' \models \varphi)\Big)$$

Given (40), (43) and (18), we conclude that

$$S^G(w, w') \preccurlyeq (M, w \models \langle a\rangle\varphi) \Rightarrow (M', w' \models \langle a\rangle\varphi)$$

□

Again some formulas are not preserved by a graded simulation. Such are the cases of $\neg\varphi$, $\varphi_1 \triangleright \varphi_2$, $[a]\varphi$. Counterexamples are provided below for each case.

- For $\neg\varphi$ consider the following paraconsistent models

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press



with $V(w,p) = (0, 0.3)$ and $V'(w',p) = (0.3, 0)$. Relation $S^G(w,w') = (1,0)$ is a graded simulation, since it satisfies conditions (31) and (32). Note that,

$$(M, w \models \neg p) \Rightarrow (M', w' \models \neg p) = (0.3, 0) \Rightarrow (0, 0.3) = (0, 0.3)$$

However,

$$S(w,w') = (1,0) \not\preccurlyeq (0, 0.3)$$

.

- For $\varphi_1 \triangleright \varphi_2$ consider again the above paraconsistent models now with $V(w,p) = V(w,q) = (0.5, 0.5)$ and $V'(w',p) = (1, 0.5)$ and $V'(w',q) = (0.5, 1)$. Relation $S^G(w,w') = (1, 0.5)$ is a graded simulation. Note that,

$$\begin{aligned}
&(M, w \models p \triangleright q) \Rightarrow (M', w' \models p \triangleright q)\\
&= /\!\!/ \left((M, w \models p) \overset{\wedge}{\wedge} /\!\!/ (M, w \models q) \right) \Rightarrow /\!\!/ \left((M', w' \models p) \overset{\wedge}{\wedge} /\!\!/ (M', w' \models q) \right)\\
&= /\!\!/ \left((0.5, 0.5) \overset{\wedge}{\wedge} (0.5, 0.5) \right) \Rightarrow /\!\!/ \left((1, 0.5) \overset{\wedge}{\wedge} (1, 0.5) \right)\\
&= (0.5, 0.5) \Rightarrow (0.5, 1)\\
&= (0.5, 0.5)
\end{aligned}$$

However,

$$S^G(w,w') = (1, 0.5) \not\preccurlyeq (0.5, 0.5)$$

.

- Finally, for $[a]\varphi$ consider the following two paraconsistent models



with $V(w,p) = V'(w',p) = (1,0)$, $V(v,p) = (0.7, 0.6)$ and $V'(v',p) = (0.1, 0.1)$. Relation $S^G(w,w') = (1, 0.1)$ and $S^G(v,v') = (0.1, 0.1)$ is a graded simulation. Note that,

$$\begin{aligned}
(M, w \models [a]p) \Rightarrow (M', w' \models [a]p) &= (M, w \models \neg\langle a\rangle\neg p) \Rightarrow (M', w' \models \neg\langle a\rangle\neg p)\\
&= /\!\!/ (M, w \models \langle a\rangle\neg p) \Rightarrow /\!\!/ (M', w' \models \langle a\rangle\neg p)\\
&= /\!\!/ (R_a(w,v) \otimes /\!\!/ V(v,p)) \Rightarrow /\!\!/ (R'_a(w',v') \otimes /\!\!/ V'(v',p))\\
&= (1, 0.1) \Rightarrow (0.1, 0.1)\\
&= (0.1, 0.1)
\end{aligned}$$

However,

$$S(w,w') = (1, 0.1) \not\preccurlyeq (0.1, 0.1)$$

.

Another interesting observation is that a disjunctive sentence $\varphi_1 \vee \varphi_2$ is preserved by graded simulations, eventhough it is defined in terms of $\neg$ and $\wedge$, with $\neg$ is not preserved. Let us start by showing that,

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

$$(M, w \models \varphi_1 \vee \varphi_2) = (M, w \models \varphi_1) \overset{\vee}{\vee} (M, w \models \varphi_2) \tag{44}$$

Let $(M, w \models \varphi_1) = (\alpha, \beta)$ and $(M', w' \models \varphi_2) = (\alpha', \beta')$ with $\alpha,\ \alpha',\ \beta,\ \beta' \in A$.

$$\begin{aligned} M, w \models \varphi_1 \vee \varphi_2 &= M, w \models \neg(\neg\varphi_1 \wedge \neg\varphi_2) \\ &= /\!\!/ (M, w \models \neg\varphi_1 \wedge \neg\varphi_2) \\ &= /\!\!/ (/\!\!/ (M, w \models \varphi_1) \overset{\wedge}{\wedge} /\!\!/ (M, w \models \varphi_2)) \\ &= /\!\!/ ((\beta, \alpha) \overset{\wedge}{\wedge} (\beta', \alpha')) \\ &= /\!\!/ (\beta \wedge \beta', \alpha \vee \alpha') \\ &= (\alpha \vee \alpha', \beta \wedge \beta') \\ &= (M, w \models \varphi_1) \overset{\vee}{\vee} (M, w \models \varphi_2) \end{aligned}$$

Thus, the proof of Theorem 5 for $\varphi_1 \vee \varphi_2$ can be written as follows

$$\begin{aligned} & (M, w \models \varphi_1 \vee \varphi_2) \Rightarrow (M', w' \models \varphi_1 \vee \varphi_2) \\ =\ & \{(44)\} \\ & \left((M, w \models \varphi_1) \overset{\vee}{\vee} (M, w \models \varphi_2)\right) \Rightarrow \left((M', w' \models \varphi_1) \overset{\vee}{\vee} (M', w' \models \varphi_2)\right) \\ \succcurlyeq\ & \{(29)\} \\ & \left((M, w \models \varphi_1) \Rightarrow (M', w' \models \varphi_1)\right) \overset{\wedge}{\wedge} \left((M, w \models \varphi_2) \Rightarrow (M', w' \models \varphi_2)\right) \\ \succcurlyeq\ & \{\text{Induction hypothesis and } \overset{\wedge}{\wedge} \text{ idempotent}\} \\ & S^G(w, w') \end{aligned}$$

Modal Invariance

Finally, modal invariance is discussed with respect to both crisp and graded bisimulation.

Theorem 6. *Let $M = \langle W, R, V\rangle$, $M' = \langle W', R', V'\rangle$ be two paraconsistent models over a iMTL-algebra $\boldsymbol{A}$, and $B \subseteq W \times W'$ be a crisp bisimulation. Then, for every $(w, w') \in B$ and any sentence $\varphi \in \mathsf{Sen}(\mathrm{Prop}, \mathrm{Act})$,*

$$(M, w \models \varphi) = (M', w' \models \varphi) \tag{45}$$

Proof. The proof is by induction over the structure of sentences.

- The case $\bot$ is trivial from the definition of $\models$, $(M, w \models \bot) = (0, 1) = (M', w' \models \bot)$.
- For $p \in \mathrm{Prop}$, since B is a crisp bisimulation, it follows that $V(w, p) = V'(w', p)$, and by definition of $\models$, conclude $(M, w \models p) = (M', w' \models p)$.
- For $\varphi_1 \wedge \varphi_2$ the proof is as follows,

$$\begin{aligned} & M, w \models \varphi_1 \wedge \varphi_2 \\ =\ & \{\text{defn of } \models\} \\ & (M, w \models \varphi_1) \overset{\wedge}{\wedge} (M, w \models \varphi_2) \\ =\ & \{\text{Induction Hypothesis twice}\} \\ & (M', w' \models \varphi_1) \overset{\wedge}{\wedge} (M', w' \models \varphi_2) \\ =\ & \{\text{defn of } \models\} \\ & M', w' \models \varphi_1 \wedge \varphi_2 \end{aligned}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

- For $\langle a\rangle\varphi$,

$$\begin{aligned}
& M, w \models \langle a\rangle\varphi \\
= & \{\text{defn of } \models\} \\
& \bigvee_{v\in W} \Big(R_a(w,v) \otimes (M, v \models \varphi) \Big) \\
= & \{(\star\star)\} \\
& \bigvee_{v'\in W'} \Big(R'_a(w',v') \otimes (M', v' \models \varphi) \Big) \\
= & \{\text{defn of } \models\} \\
& M', w' \models \langle a\rangle\varphi
\end{aligned}$$

Step $(\star\star)$ is proved as follows: B being a bisimulation, there exists $v \in W$ such that $R'_a(w', v') = R_a(w, v)$ and, by hypothesis, $(M', v' \models \varphi) = (M, v \models \varphi)$. Therefore,

$$R'_a(w', v') \otimes (M', v' \models \varphi) = R_a(w, v) \otimes (M, v \models \varphi)$$

Since $\bigvee$ is monotone, $\bigvee_{v'\in W'} \Big(R'_a(w', v') \otimes (M', v' \models \varphi) \Big) = \bigvee_{v\in W} \Big(R_a(w, v) \otimes (M, v \models \varphi) \Big)$ □

Theorem 7. *Let $M = \langle W, R, V\rangle$ and $M' = \langle W', R', V'\rangle$ be two paraconsistent models over a iMTL-algebra $\mathbf{A}$, and $B^G : W \times W' \to A \times A$ be a graded bisimulation. Then, for every $w \in W$, $w' \in W'$ and any sentence $\varphi \in \mathsf{Sen}(\mathrm{Prop}, \mathrm{Act})$,*

$$B^G(w, w') \preccurlyeq \Big((M, w \models \varphi) \Leftrightarrow (M', w' \models \varphi) \Big) \tag{46}$$

Proof. The proof is again by induction over the structure of sentences.

- Case $\bot$ is trivial,

$$B^G(w, w') \preccurlyeq (M, w \models \bot) \Leftrightarrow (M', w' \models \bot) = (0, 1) \Leftrightarrow (0, 1) = (1, 0)$$

- For $p \in \mathrm{Prop}$, since B^G is a graded bisimulation it satisfies condition (33), that is

$$B^G(w, w') \preccurlyeq V(w, p) \Leftrightarrow V'(w', p)$$

Using the definition of $\models$, we conclude

$$B^G \preccurlyeq (M, w \models p) \Leftrightarrow (M', w' \models p)$$

- For $\neg\varphi$,

$$\begin{aligned}
& (M, w \models \neg\varphi) \Leftrightarrow (M, w \models \neg\varphi) \\
= & \{\text{defn of } \models\} \\
& /\!\!/ (M, w \models \varphi) \Leftrightarrow /\!\!/ (M, w \models \varphi) \\
= & \{\text{Property (20)}\} \\
& (M, w \models \varphi) \Leftrightarrow (M, w \models \varphi) \\
\succcurlyeq & \{\text{Induction hypothesis}\} \\
& B^G(w, w')
\end{aligned}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

- For $\varphi_1 \wedge \varphi_2$,

$$\begin{aligned}
&\left((M, w \models \varphi_1) \mathbin{\hat{\wedge}} (M, w \models \varphi_2)\right) \Leftrightarrow \left((M', w' \models \varphi_1) \mathbin{\hat{\wedge}} (M', w' \models \varphi_2)\right) \\
&\succcurlyeq \{(27)\} \\
&\left((M, w \models \varphi_1) \Leftrightarrow (M', w' \models \varphi_1)\right) \mathbin{\hat{\wedge}} \left((M, w \models \varphi_2) \Leftrightarrow (M', w' \models \varphi_2)\right) \\
&\succcurlyeq \{\text{induction hypothesis and } \hat{\wedge} \text{ idempotent}\} \\
&B^G(w, w')
\end{aligned}$$

- For $\langle a \rangle \varphi$ we need to verify the following two inequalities,

$$B^G(w, w') \preccurlyeq (M, w \models \langle a \rangle \varphi) \Rightarrow (M', w' \models \langle a \rangle \varphi) \tag{47}$$

$$B^G(w, w') \preccurlyeq (M', w' \models \langle a \rangle \varphi) \Rightarrow (M, w \models \langle a \rangle \varphi) \tag{48}$$

The proof for (47) is equal to the one presented for case $\langle a \rangle \varphi$ in Theorem 5. Similarly, one can prove (48). Since B^G is a graded bisimulation, there is $v \in W$ such that the following inequality is satisfied

$$B^G(w, w') \otimes R'_a(w', v') \preccurlyeq R_a(w, v) \otimes B^G(v, v') \tag{49}$$

We will prove that for $v \in W$,

$$B^G(w, w') \preccurlyeq (M', w' \models \langle a \rangle \varphi) \Rightarrow (R_a(w, v) \otimes M, v \models \varphi) \tag{50}$$

Observe that,

$$(R_a(w, v) \otimes M, v \models \varphi) \preccurlyeq (M, w \models \langle a \rangle \varphi) \tag{51}$$

Thus, by proving (50) using property (18), we verify (48). Since B^G is a graded simulation, $B^G(v, v') \preccurlyeq (M', v' \models \varphi) \Rightarrow (M, v \models \varphi)$. Using (18) in (49), entails

$$B^G(w, w') \otimes R'_a(w', v') \preccurlyeq R_a(w, v) \otimes ((M', v' \models \varphi) \Rightarrow (M, v \models \varphi))$$

Then, by (17),

$$B^G(w, w') \preccurlyeq R'_a(w', v') \Rightarrow \left(R_a(w, v) \otimes ((M', v' \models \varphi) \Rightarrow (M, v \models \varphi))\right)$$

Using (23),

$$B^G(w, w') \preccurlyeq R'_a(w', v') \Rightarrow \left((M', v' \models \varphi) \Rightarrow (R_a(w, v) \otimes (M, v \models \varphi))\right)$$

Using (24),

$$B^G(w, w') \preccurlyeq \left(R'_a(w', v') \otimes (M', v' \models \varphi)\right) \Rightarrow \left(R_a(w, v) \otimes (M, v \models \varphi)\right)$$

By (18),

$$B^G(w, w') \preccurlyeq (M', w' \models \langle a \rangle \varphi) \Rightarrow (M, w \models \langle a \rangle \varphi)$$

Finally, by (47) and (48) and the fact that $\hat{\wedge}$ is monotone,

$$B^G(w, w') \preccurlyeq (M, w \models \langle a \rangle \varphi) \Leftrightarrow (M', w' \models \langle a \rangle \varphi)$$

□

6. An Application to Robotics

In this section, we take the initial steps toward exploring a potential application of paraconsistent transition systems in robotics. This inspiration is drawn from the work documented in

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

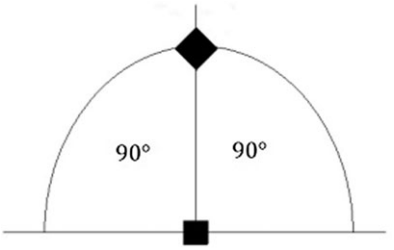


Figure 2. The angles of vision of each sensor of the robot (Costa and Martins 2016).

Abe et al. (2007) and its adaptation in Costa and Martins (2016), where paraconsistent logics are employed to determine the movements of a robot. The main goal is to investigate how the robot, which may receive contradictory information regarding the presence (or absence) of objects in its environment, navigates along a specific path.

While sensors for object detection remain internally consistent, discrepancies may arise between them, particularly if their methods for object detection are different. Furthermore, sensors are susceptible to contradictions due to various factors, such as hardware limitations that restrict the accuracy in measuring to a certain degree, and the fact that measurements themselves can be affected by outside conditions. Consequently, it is not uncommon information collected by one sensor to contradict that collected by another. On the other hand, combining various types of sensors is a common practice to enhance reliability and accuracy in obstacle detection and avoidance.

These challenges motivate our approach to incorporate various types of sensors, consider external conditions and use each sensor's characteristics in determining confidence in object detection along different navigation paths. To illustrate this, let us work with 4 sensors: two *ultrasonic sensors* that measure distance to an obstacle by using ultrasonic waves, and two *infrared reflective sensors* that detect the presence of an obstacle by measuring the amount of reflected infrared light. There will be two sets of sensors: one called the set of "positive sensors" that comprises one sensor of each type (one ultrasonic and one infrared), while the other is called the set "of negative sensors" and includes the remaining sensors of each type. Both sets of sensors evaluate if a path is free of obstacles or not. Then each set provides a value in the interval [0, 1]. Such value represents, for the "positive sensors", its certainty that there is no obstacle and the path is *clear*, while for the "negative sensors" it represents its certainty that there is an obstacle and the path is *obstructed*.

Certainty is a crucial aspect in object detection, where the system not only identifies objects but also quantifies its confidence in the accuracy of identification. Various approaches can be adopted to measure certainty, such as considering limitations in the precision of measurements, disregarding a sensor's output in the presence of environmental conditions that affect measurements, and setting different threshold levels for activation and deactivation to maintain stability in the output due to noise or minor fluctuations.

It is also possible to take into consideration different placements of the sensors. For instance, in Costa and Martins (2016), the approach involved a robot with two sensors covering a total amplitude of 180°: one on the left covering 90° to the left, and the other on the right covering 90° to the right, as illustrated in Figure 2. Thus, at specific points along the path certain sensors may be temporarily disregarded or have default values adopted to increase confidence in detection.

In our example, the robot is capable of movement in four directions (left, right, up, and down) and is equipped with two sets of different types of sensors. To simplify the problem, as the robot navigates through a map with four checkpoints, a condition is imposed that prohibits the robot to return to a previously visited checkpoint (resulting in paths with maximum length of 3). We will also assume a non adaptive approach since we will only focus on walks that have one or two moves. However, it is possible to consider that at each state the robot has more information, and

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

makes decisions based on them. The checkpoints are represented by states in a paraconsistent model and transitions between these states represent the robot's movement from one checkpoint to another. The transition labels indicate the information given by the set of "positive sensors" and the "negative sensors" to move in that direction. Thus, each path will entail a pair (α, β), where α indicates the certainty degree up to which the "positive sensors" detected the path is clear, and β indicates the certainty degree up to which the "negative sensors" detected the path is obstructed.

As before, for any given pair (α, β), if $\alpha + \beta = 1$, we face a scenario where all sensors are consistent with each other, if $\alpha + \beta < 1$, the sensors are consistent but lack sufficient information together to draw a conclusion and if $\alpha + \beta > 1$, the sensors disagree with each other. For example, a label such as

(1) $(1, 0)$ indicates complete agreement between all of the sensors that the path is clear, allowing the robot to move freely in that direction;

(2) $(0, 1)$ indicates unanimous consensus between all of the sensors that the path is not clear, restricting the robot from moving in that direction;

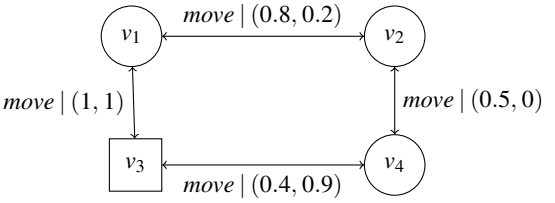
(3) $(1, 1)$ suggests complete disagreement between both sensors, leading to the paradoxical situation where they are simultaneously certain that the path is clear and not clear.

(4) $(0, 0)$ reflects minimal certainty from both sensors regarding the path's status as free or obstructed.

Scenarios of inconsistency, as in 3, or vagueness, as in 4, may be more challenging to picture than others. However, it is conceivable that under external unfavorable conditions, such as adverse weather, sensors could either demonstrate complete disagreement or lack confidence in their measurements.

Additionally, the proposition *Obst* serves as a crisp valuation indicating the presence of an obstacle at a given state, with a valuation of $(1, 0)$, or the absence of an obstacle, with a valuation of $(0, 1)$.

Let us consider the following map:



If the robot is initially positioned in state v_1, the path to v_2 is reportedly clear by the "positive sensors" with a certainty of 0.8, and it is reported as not clear by the "negative sensors" with a certainty of 0.2. The only state that contains an obstacle is state v_3 and therefore is pictured differently in the model.

Let us evaluate a few sentences and explore the robot's behavior in the paraconsistent model.

	v_1	v_2	v_3	v_4
$Obst \rightarrow [move]\bot$	$(1, 0)$	$(1, 0)$	$(0, 1)$	$(1, 0)$
$\neg Obst \wedge \langle move\rangle \neg Obst$	$(0.8, 0)$	$(0.8, 0)$	$(0, 1)$	$(0.5, 0)$
$\neg Obst \wedge \langle move\rangle \neg Obst \wedge \langle move\rangle \langle move\rangle \neg Obst$	$(0.5, 0)$	$(0, 1)$	$(0, 1)$	$(0.5, 0)$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

- The sentence $Obst \rightarrow [move]\bot$ indicates that whenever a state has an obstacle, the robot should not be able to move. As expected, this sentence at states v_1, v_2 and v_4 is consistently true since they do not have obstacles. However, at state v_3 the evaluation is consistently false. Eventhough, state v_3 has an obstacle, there is positive evidence that it is possible to move to another state.

$$
\begin{aligned}
& M, v_3 \models Obst \rightarrow [move]\bot \\
&= M, v_3 \models Obst \rightarrow \neg\langle move\rangle \top \\
&= M, v_3 \models \neg\,(Obst \wedge \langle move\rangle \top) \\
&= /\!/ \left((M, v_3 \models Obst) \overset{\wedge}{\wedge} (M, v_3 \models \langle move\rangle \top) \right) \\
&= /\!/ \left(V(v_3, Obst) \overset{\wedge}{\wedge} ((R_{move}(v_3, v_1) \otimes (1,0)) \overset{\vee}{\vee} (R_{move}(v_3, v_4) \otimes (1,0)) \right) \\
&= /\!/ \left((1,0) \overset{\wedge}{\wedge} ((1,0) \overset{\vee}{\vee} (0.4,0) \right) \\
&= (0,1)
\end{aligned}
$$

  The problem arises because both $R^{+}_{move}(v_3, v_1)$ and $R^{+}_{move}(v_3, v_4)$ are both different from 0. If, at that state, external and internal conditions for the sensors were improved to the extent that the positive sensors indicate the path is clear with 0 certainty, and the negative sensors report the path is obstructed with the same certainty level, that is, $R_{move}(v_3, v_1) = (0, 1)$ and $R_{move}(v_3, v_4) = (0, 0.9)$. Then,

$$
\begin{aligned}
& M, v_3 \models Obst \rightarrow [move]\bot \\
&= /\!/ \left((M, v_3 \models Obst) \overset{\wedge}{\wedge} (M, v_3 \models \langle move\rangle \top) \right) \\
&= /\!/ \left(V(v_3, Obst) \overset{\wedge}{\wedge} ((R_{move}(v_3, v_1) \otimes (1,0)) \overset{\vee}{\vee} (R_{move}(v_3, v_4) \otimes (1,0)) \right) \\
&= /\!/ \left(V(v_3, Obst) \overset{\wedge}{\wedge} (((0,1) \otimes (1,0)) \overset{\vee}{\vee} ((0,0.9) \otimes (1,0)) \right) \\
&= /\!/ \left((1,0) \overset{\wedge}{\wedge} ((0,1) \overset{\vee}{\vee} (0,0.9) \right) \\
&= (0.9,0)
\end{aligned}
$$

  Only with such changes would the robot's behavior at that state align with the expected result, that is, the robot at state v_3 has (almost) complete certainty that in the presence of an obstacle, it cannot move.
- The sentence $\neg Obst \wedge \langle move\rangle \neg Obs$ indicates that at a state with no obstacle it is possible to move to another state where there is still no obstacle. At state v_4, there is less positive certainty of movement than at states v_1 and v_2. Thus, the evaluation of the sentence at v_4 is $(0.5, 0)$, while at v_1 and v_2 it is $(0.8, 0)$. Given that $(0.5, 0) \preccurlyeq (0.8, 0)$ when planning the robot's path, we might take the starting position at state v_1 instead of v_4. This is because there is overall more certainty that the first move from state v_1 will not encounter an obstacle.
- The sentence $\neg Obst \wedge \langle move\rangle \neg Obst \wedge \langle move\rangle \langle move\rangle \neg Obst$ indicates that at a state with no obstacle, it is possible to move two steps without ever encountering an obstacle. This sentence evaluation is consistently false at state v_3, which has an obstacle, and at state v_2 because every walk that starts at v_2 and takes two steps will always lead to v_3. Finally, at states v_1 and v_4, there is a 0.5 certainty that taking two steps will lead to a path without obstacles, and there is a 0 certainty that taking two steps will encounter an obstacle. This situation represents a

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

vague scenario where the sum of the weights is less than 1. This is because transitions from v_1 to v_2 to v_4 and vice versa always present positive evidence smaller than 1, so they do not have complete certainty that it is possible to move. However, they present 0 or close to 0 negative evidence that the transitions will be prevented from occurring due to obstacle detection.

This example shows how paraconsistent models and their logics can be used in practice to model complex, weakly consistent situations, and enabling reasoning about them. It is worth noting that this example can be enhanced by incorporating dynamic operators from dynamic logic extended to the paraconsistent setting, as introduced in our prior work (Cunha et al. 2023a). Reference Barbosa and Madeira (2023) reports on another application in a totally different domain, that of quantum circuits analysis in the presence of decoherence.

7. Conclusions and Future Work

This paper studied a new kind of labeled transition structures able to capture both *vagueness* and *inconsistency* in software modeling scenarios. Two main research directions were pursued. First, the structure of a category of (pointed) PLTS was explored to define a number of useful operators to build such systems in a compositional way. Then, a modal logic, whose modalities are indexed by transition labels, was proposed. Preliminary results of classical and graded soundness were explored. This graded variant adjusts the classical definition of soundness to accommodate vague and inconsistent evaluations. Thus, a rule of the form:

$$\frac{\varphi_1 \ \ldots \ \varphi_n}{\varphi}$$

is *g-sound* if for any paraconsistent Kripke model $M = \langle W, R, V \rangle$,

$$\bigwedge_{i=1}^{n} (M \models \varphi_i) \preccurlyeq (M \models \varphi)$$

This implies that from the evidence of truth and falsity of premises, it is possible to draw conclusions whose evidence of truth is higher and evidence of falsity is lower. Remarkably, while g-soundness entails classical soundness, the reverse is not true. This lack of equivalence prompts further investigation in § 5.2 of classical and graded soundness in paraconsistent Kripke models.

Current work seeks to develop an axiomatic system and prove its completeness and soundness for the paraconsistent logic discussed. Drawing inspiration from research on many-valued complete logics (Caicedo and Rodríguez 2010), we anticipate achieving similar results, at least for the positive component of the paraconsistent logic presented in this paper. Our approach so far has been to separately explore and establish the completeness and soundness of the positive and negative components of the logic, with the aim of later combining them for overall results. This strategy offers greater flexibility, as some statements are theorems in the positive component but not in the negative, indicating that we may be certain that a statement will always hold (positively it is a theorem), but the evidence for it not holding may not be exactly complementary (negatively it is not a theorem); the converse can also occur.

From this ongoing work, preliminary findings include the observation that the inference rule *Modus Ponens* preserves truth in the positive component, that is, if the premises hold with positive certainty 1, then the conclusions also hold with positive certainty 1, but not in the negative component, meaning that premises may have negative certainty, but the conclusions do not necessarily follow. Additionally, we have found that the modal logic is not normal, as it does not include the schema $\Box(A \to B) \to (\Box A \to \Box B)$.

This paper also introduced both crisp and graded notions of simulation and bisimulation of the corresponding (paraconsistent) Kripke models were also characterized and a number of modal preservation results proved. It remains to consider other notions of equivalence from coalgebraic

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

trace equivalence (Urabe and Hasuo 2018) to logic-induced bisimulations (Beohar et al. 2023) and their corresponding preservation results.

A lot remains to be done, namely in what concerns the development of the logic's proof-theoretic perspective, which is not addressed here. Such is crucial from a Software Engineering point of view, as a major step towards (semi-)automatic support to reasoning about such complex, often weird, but actually quite common phenomena. The development of a proper specification theory Sannella (2012) is also worth to explore, following some preliminary work by the authors documented in Cunha et al. (2023a,b).

The detailed characterization of different application scenarios, from AI models to quantum computation and robotics, constitutes another main challenge for the future, once the basic mathematical structure has been completely unveiled.

Acknowledgment. This work is supported by the FCT - Portuguese Foundation of Science and Technology with grant 2024.04652.BD and with projects PTDC/CCI-COM/4280/2021, UIDB/04106/2025 and Bansky - A paraconsistent inference engine to support research in age-related macular degeneration.

Note

1 Notation $\lambda : \mathrm{Act}_1 \rightarrow_\perp \mathrm{Act}_2$ stands for the totalization of a partial function by mapping to $\perp$ all elements of Act_1 for which the function is undefined.

References

Abe, J. M., Torres, C. R., Lambert-Torres, G., da Silva Filho, J. I. and Martins, H. G. (2007). Paraconsistent autonomous mobile robot emmy III. In: Lambert-Torres, G., Abe, J. M., da Silva Filho, J. I. and Martins, H. G. (eds.) *Advances in Technological Applications ofLogical and Intelligent Systems, Selected Papers from the Sixth Congress on Logic Applied to Technology,LAPTEC 2007, Unisanta, Santa Cecilia University, Santos, Brazil, November 21–23, 2007, volume 186 ofFrontiers in Artificial Intelligence and Applications*, IOS Press, 236–258.

Agudelo, J. C. and Carnielli, W. A. (2010). Paraconsistent machines and their relation to quantum computing. *Journal of Logic and Computation*. **20** (2) 573–595.

Akama, S. (ed.) (2016) *Towards Paraconsistent Engineering, Volume 110 of Intelligent Systems Reference Library*, Springer.

Badia, G., Caicedo, X. and Noguera, C. (2023). Frame definability in finitely valued modal logics. *Annals of Pure and Applied Logic*. **174** (7) 103273.

Badia, G. and Olkhovikov, G. K. (2020). A lindström theorem in many-valued modal logic over a finite mtl-chain. *Fuzzy Sets and Systems*. **388**) 26–37.

Baeten, J. C. M., Basten, T. and Reniers, M. A. (2010). *Process Aalgebra: Eequational theories of communicating processes. Cambridge Tracts in Theoretical Computer Science*, **50**, Cambridge University Press.

Barbosa, L. S. and Madeira, A. (2023). Capturing qubit decoherence through paraconsistent transition systems. In: Chiba, S., Cong, Y. and Boix, E. G., *Companion Proceedings of the 7th International Conference on the Art, Science, and Engineering of Programming, Programming 2023, Tokyo,Japan, March 13–17, 2023*, ACM, 109–110.

Beohar, H., Gurke, S., König, B. and Messing, K. (2023). Hennessy–Milner theorems via Galois connections. In: Klin, B. and Pimentel, E., *31st EACSL Annual Conference on Computer ScienceLogic (CSL 2023), volume 252 of Leibniz International Proceedings in Informatics (LIPIcs), Dagstuhl, Germany, 2023*, Schloss Dagstuhl – Leibniz-Zentrum f¨ur Informatik, 12:1–12:18.

Bou, F., Esteva, F., Godo, L. and Rodríguez, R. O. (10 2009). On the minimum many-valued modal logic over a finite residuated lattice. *Journal of Logic and Computation*. **21** (5) 739–790.

Caicedo, X., Metcalfe, G., Rodríguez, R. O. and Rogger, J. (2017). Decidability of order-based modal logics. *Journal of Computer and System Sciences*. **88** 53–74.

Caicedo, X. and Rodríguez, R. O. (2010). Standard gödel modal logics. *Studia Logica*. **94** (2) 189–214.

Caicedo, X. and Rodríguez, R. O. (2015). Bi-modal gödel logic over [0, 1]-valued kripke frames. *Journal of Logic and Computation*. **25** (1) 37–55.

Chiara, M. L. D. and Giuntini, R. (2000). Paraconsistent ideas in quantum logic. *Synthese*. **125** (1-2) 55–68.

Costa, D. and Martins, M. A. (2016). Intelligent-based robot to deal with contradictions. *2016 International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 199–204.

Cruz, A., Madeira, A. and Barbosa, L. S. (2022). A logic for paraconsistent transition systems. In: Indrzejczak, A. and Zawidzki, M. (eds.) *Proceedings of the 10th International Conference on Non-ClassicalLogics. Theory and Applications, NCL 2022, Łódź, Poland, 14–18 March 2022, Volume 358 of EPTCS*, 270–284.

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Cruz, A., Madeira, A. and Barbosa, L. S. (2022). Paraconsistent transition systems. In: Nantes-Sobrinho, D. and Fontaine, P. (eds.) *Proceedings 17th International Workshop on Logical and SemanticFrameworks with Applications, LSFA 2022, Belo Horizonte, Brazil (hybrid), 23–24 September 2022, Volume 376 of EPTCS*, 3–15.

Cunha, J., Madeira, A. and Barbosa, L. S. (2023). Stepwise development of paraconsistent processes. In: David, C. and Sun, M. (eds.) *Theoretical Aspects of Software Engineering - 17th InternationalSymposium, TASE 2023, Bristol, UK, July 4–6, 2023, Proceedings, Volume 13931 of Lecture Notes in Computer Science*, Springer, 327–343.

Cunha, J., Madeira, A. and Barbosa, L. S. (2023) Structured specification of paraconsistent transition systems. In: Hojjat, H. and brahám, E. Á. (eds.) *Fundamentals of Software Engineering - 10thInternational Conference, FSEN 2023, Tehran, Iran, May 4–5, 2023, Revised Selected Papers, Volume 14155 of Lecture Notes in Computer Science*, Springer, pp. 1–17.

da Costa, N. C. A. and Krause, D. (2014). Physics, inconsistency, and quasi-truth. *Synthese*. **191** (13) 3041–3055.

da Costa, N. C. A., Krause, D. and Bueno, O. (2007) Paraconsistent logics and paraconsistency. In: Jacquette, D. (ed.), *Handbook of the Philosophy of Science (Philosophy of Logic)*, Elsevier, 791–911.

Esteva, F. and Godo, L. (2001). Monoidal t-norm based logic: towards a logic for left-continuous t-norms. *Fuzzy Sets and Systems*. **124** (3) 271–288.

Jain, M., Madeira, A. and Martins, M. A. (2019). A fuzzy modal logic for fuzzy transition systems. Felty, A. P. and Marcos, J. (eds.) *Proceedings of the 14th Workshop on Logical and Semantic Frameworkswith Applications, LSFA 2019, Natal, Brazil, August, 2019, volume 348 of Electronic Notes in Theoretical Computer Science*, Elsevier, 85–103.

Jaśkowski, S. (1969). Propositional calculus for contradictory deductive systems. *Studia Logica*. **24** (1) 143–157.

Kracht, M. (1998). On extensions of intermediate logics by strong negation. *Journal of Philosophical Logic*. **27** (1) 49–73.

Madeira, A., Neves, R. and Martins, M. A. (2016). An exercise on the generation of many-valued dynamic logics. *Journal of Logical and Algebraic Methods in Programming*. **85** (5) 1011–1037.

Marti, M. and Metcalfe, G. (2014) A hennessy-milner property for many-valued modal logics. In: Goré, R., Kooi, B. P. and Kurucz, A. (eds.), *Advances in Modal Logic 10, Invited and Contributed Papers From the Tenth Conference On "Advances in Modal Logic," Held in Groningen, The Netherlands, August 5–8, 2014*, College Publications, 407–420.

Nguyen, L. A. (2022). Logical characterizations of fuzzy bisimulations in fuzzy modal logics over residuated lattices. *Fuzzy Sets and Systems*. **431** 70–93.

Preskill, J. (2018). Quantum computing in the nisq era and beyond. *Quantum*. **2** (79) 87–95.

Sannella, A. T. D. (2012). *Foundations of Algebraic Specification and Formal Software Development*, Springer-Verlag.

Urabe, N. and Hasuo, I. (2018). Coalgebraic infinite traces and kleisli simulations. *Logical Methods in Computer Science*. **14** (3).

Winskel, G. and Nielsen, M. (1995) Models for concurrency. In: *Handbook of Logic in Computer Science (Vol. 4): Semantic Modelling*, Abramsky, S., Gabbay, D. M. and Maibaum, T. S. E. (eds.) Oxford Science Publications, 1–148.

Appendix A. Proof of Lemma 1

We will focus on the proofs of properties (12), (13), (14), and (16). The proofs for the remaining properties can be found in Nguyen (2022).

- Property (12) is a direct consequence of (1) applied to the trivial assertion $a \to c \leq a \to c$
- To prove Property (13) we start by proving

$$(a \to c) \wedge (b \to c) \leq (a \vee b) \to c \tag{52}$$

and then

$$(a \to c) \wedge (b \to c) \geq (a \vee b) \to c \tag{53}$$

Trivially, $c \geq c \wedge (b \to c)$ and $c \geq c \wedge (a \to c)$ and since $\vee$ is monotone, $c \geq (c \wedge (b \to c)) \vee (c \wedge (a \to c))$. Then, Property (12) entails $c \geq a \wedge (a \to c)$ and $c \geq b \wedge (b \to c)$, leading to

$$c \geq (a \wedge (a \to c) \wedge (b \to c)) \vee (b \wedge (b \to c) \wedge (a \to c))$$

as $\wedge$ and $\vee$ are monotone as well. Since $\wedge$ is associative, property (10) entails $c \geq (a \vee b) \wedge ((a \to c) \wedge (b \to c))$. Finally, using Property (1) we have (52). To prove (53) we start with the trivial assertions $a \leq a \vee b$ and $b \leq a \vee b$. By Property (5), $(a \vee b) \to c \leq a \to c$ and $(a \vee b) \to c \leq b \to c$. Finally, since $\wedge$ is monotone and idempotent we prove (53).

- To prove property (14) we first start by proving $(a \to b) \wedge (a \to c) \leq a \to (b \wedge c)$ and then $a \to (b \wedge c) \leq (a \to b) \wedge (a \to c)$. According to property (12), $a \wedge (a \to b) \leq b$ and $a \wedge (a \to c) \leq c$. Since $\wedge$ is monotone and idempotent, $a \wedge (a \to b) \wedge (a \to c) \leq b \wedge c$. Using

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

(1), entails $(a \to b) \wedge (a \to c) \leq a \to (b \wedge c)$. Moreover, trivially $b \wedge c \leq b$ and $b \wedge c \leq c$ by Property (5) $a \to (b \wedge c) \leq a \to b$ and $a \to (b \wedge c) \leq a \to c$. Finally, given that $\wedge$ is monotone and idempotent $a \to (b \wedge c) \leq (a \to b) \wedge (a \to c)$

- To prove property (16), observe that, by (12), $(b \to c) \wedge b \leq c$. Since $\wedge$ is monotone and associative $b \wedge a \wedge (b \to c) \leq (c \wedge a)$. Using (1) it follows that $a \wedge (b \to c) \leq b \to (c \wedge a)$.

Appendix B. Proof of Lemma 3

- To prove Property (18) we need to prove $(a \to c) \wedge (d \to b) \leq (a' \to c') \wedge (d' \to b')$ and $a \wedge d \geq a' \wedge d'$. Let us assume that $a' \leq a$, $c \leq c'$, $b' \geq b$ and $d \geq d'$. By Property (5), $(a \to c) \leq (a' \to c')$ and $(d \to b) \leq (d' \to b')$. Since $\wedge$ is monotone, $(a \to c) \wedge (d \to b) \leq (a' \to c') \wedge (d' \to b')$. Finally, by hypothesis and the fact that $\wedge$ is monotone, $a \wedge d \geq a' \wedge d'$.
- To prove Property (19) we need to prove that $a \wedge c \leq a' \wedge c'$ and that $(a \to d) \wedge (c \to b) \geq (a' \to d') \wedge (c' \to b')$. By hypothesis $a \leq a'$ and $c \leq c'$, since $\wedge$ is monotone $a \wedge c \leq a' \wedge c'$. Also by hypothesis, $b \geq b'$ and $d \geq d'$, by Property (5), $(a \to d) \wedge (c \to b) \geq (a' \to d') \wedge (c' \to b')$.
- To prove Property (20) note that

$$(a, b) \Leftrightarrow (c, d) = ((a \leftrightarrow c) \wedge (d \leftrightarrow b), (a \wedge d) \vee (c \wedge b))$$
$$/\!\!/ (a, b) \Leftrightarrow /\!\!/ (c, d) = (b, a) \Leftrightarrow (d, c) = ((b \leftrightarrow d) \wedge (a \leftrightarrow c), (c \wedge b) \vee (a \wedge d))$$

  Since $\wedge$ and $\vee$ are associative, the pair $(a, b) \Leftrightarrow (c, d)$ is equal to the pair $/\!\!/ (a, b) \Leftrightarrow /\!\!/ (c, d)$.
- To prove Property (21) note that

$$(a, b) \otimes ((c, d) \overset{\vee}{\vee} (e, f)) = (a \wedge (c \vee e), a \to (d \wedge f) \wedge (c \vee e) \to b)$$
$$((a, b) \otimes (c, d)) \overset{\vee}{\vee} ((a, b) \otimes (e, f)) = ((a \wedge c) \vee (a \wedge e), a \to d \wedge c \to b \wedge a \to f \wedge e \to b)$$

  By Property (10), it follows that $a \wedge (c \vee e) = (a \wedge c) \vee (a \wedge e)$ and by Property (13) and (14) it follows $a \to (d \wedge f) \wedge (c \vee e) \to b = a \to d \wedge c \to b \wedge a \to f \wedge e \to b$.
- To prove Property (22) note that $(c, d) \overset{\wedge}{\wedge} (e, f) \preccurlyeq (c, d)$ and $(c, d) \overset{\wedge}{\wedge} (e, f) \preccurlyeq (e, f)$. Using Property (19),

$$(a, b) \otimes \left((c, d) \overset{\wedge}{\wedge} (e, f) \right) \preccurlyeq (a, b) \otimes (c, d)$$
$$(a, b) \otimes \left((c, d) \overset{\wedge}{\wedge} (e, f) \right) \preccurlyeq (a, b) \otimes (e, f)$$

  Finally, since the operator $\overset{\wedge}{\wedge}$ is monotone it implies that,

$$(a, b) \otimes \left((c, d) \overset{\wedge}{\wedge} (e, f) \right) \preccurlyeq \left((a, b) \otimes (c, d) \right) \overset{\wedge}{\wedge} \left((a, b) \otimes (e, f) \right)$$

- To prove Property (23) note that,

$$\begin{aligned}(a, b) \otimes ((c, d) \Rightarrow (e, f)) &= (a, b) \otimes ((c \to e) \wedge (f \to d), c \wedge f) \\ &= (a \wedge (c \to e) \wedge (f \to d), (a \to (c \wedge f)) \wedge ((c \to e \wedge f \to d) \to b)) \\ (c, d) \Rightarrow ((a, b) \otimes (e, f)) &= (c, d) \Rightarrow (a \wedge e, (a \to f) \wedge (e \to b)) \\ &= ((c \to (a \wedge e)) \wedge (((a \to f) \wedge (e \to b)) \to d), c \wedge (a \to f) \wedge (e \to b))\end{aligned}$$

  Thus, we want to prove

$$a \wedge (c \to e) \wedge (f \to d) \leq (c \to (a \wedge e)) \wedge ((a \to f \wedge e \to b) \to d) \tag{54}$$
$$(a \to (c \wedge f)) \wedge (((c \to e) \wedge (f \to d)) \to b) \geq c \wedge (a \to f) \wedge (e \to b) \tag{55}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

The proof of inequality (54) follows as

$$\begin{aligned}
& a \wedge (c \to e) \wedge (f \to d) \\
= & \{\wedge \text{ is idempotent and associative}\} \\
& a \wedge (c \to e) \wedge a \wedge (f \to d) \\
\leq & \{(16)\} \\
& (c \to (a \wedge e)) \wedge a \wedge (f \to d) \\
\leq & \{(15)\} \\
& (c \to (a \wedge e)) \wedge ((a \to f) \to d) \\
\leq & \{\text{since } (a \to f) \geq (a \to f) \wedge (e \to b) \text{ and } (5)\} \\
& (c \to (a \wedge e)) \wedge (((a \to f) \wedge (e \to b)) \to d)
\end{aligned}$$

For inequality (55) consider

$$\begin{aligned}
& (a \to (c \wedge f)) \wedge (((c \to e) \wedge (f \to d)) \to b) \\
\geq & \{(16)\} \\
& c \wedge (a \to f) \wedge (((c \to e) \wedge (f \to d)) \to b) \\
\geq & \{\text{since } c \to e \geq (c \to e) \wedge (f \to d) \text{ by } (5)\} \\
& c \wedge (a \to f) \wedge ((c \to e) \to b) \\
\geq & \{(15)\} \\
& c \wedge (a \to f) \wedge c \wedge (e \to b) \\
= & \{\wedge \text{ is idempotent}\} \\
& c \wedge (a \to f) \wedge (e \to b)
\end{aligned}$$

- To prove Property (24) note that,

$$\begin{aligned}
(a,b) \Rightarrow ((c,d) \Rightarrow (e,f)) &= (a,b) \Rightarrow ((c \to e) \wedge (f \to d), c \wedge f) \\
&= ((a \to ((c \to e) \wedge (f \to d))) \wedge ((c \wedge f) \to b), a \wedge c \wedge f) \\
((a,b) \otimes (c,d)) \Rightarrow (e,f) &= (a \wedge c, (a \to d) \wedge (c \to b)) \Rightarrow (e,f) \\
&= (((a \wedge c) \to e) \wedge (f \to ((a \to d) \wedge (c \to b))), a \wedge c \wedge f)
\end{aligned}$$

Thus, we want to prove that,

$$(a \to ((c \to e) \wedge (f \to d))) \wedge ((c \wedge f) \to b) \leq ((a \wedge c) \to e) \wedge (f \to ((a \to d) \wedge (c \to b)))$$

The proof follows as,

$$\begin{aligned}
& (a \to ((c \to e) \wedge (f \to d))) \wedge ((c \wedge f) \to b) \\
\leq & \{(5), \text{ and } \wedge \text{ is monotone}\} \\
& (a \to (c \to e)) \wedge (a \to (f \to d)) \wedge ((c \wedge f) \to b) \\
= & \{(9)\} \\
& ((a \wedge c) \to e) \wedge (a \to (f \to d)) \wedge ((c \wedge f) \to b) \\
= & \{(8)\} \\
& ((a \wedge c) \to e) \wedge (f \to (a \to d)) \wedge ((c \wedge f) \to b) \\
= & \{(9)\} \\
& ((a \wedge c) \to e) \wedge (f \to (a \to d)) \wedge (f \to (c \to b)) \\
\leq & \{(14)\} \\
& ((a \wedge c) \to e) \wedge (f \to ((a \to d) \wedge (c \to b)))
\end{aligned}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

- To prove Property (25) let us start by noting that,

$$\begin{aligned}(a,b)\otimes((c,d)\Rightarrow(e.f)) &= (a,b)\otimes((c\to e)\wedge(f\to d),c\wedge f)\\ &= (a\wedge(c\to e)\wedge(f\to d),(a\to(c\wedge f))\wedge(((c\to e)\\ &\quad\wedge(f\to d))\to b))\\ ((a,b)\Rightarrow(c,d))\Rightarrow(e,f) &= ((a\to c)\wedge(d\to b),a\wedge d)\Rightarrow(e,f)\\ &= ((((a\to c)\wedge(d\to b))\to e)\wedge(f\to(a\wedge d)),(a\to c)\\ &\quad\wedge(d\to b)\wedge f)\end{aligned}$$

  which entails the need to prove the following inequalities,

$$a\wedge(c\to e)\wedge(f\to d)\ \leq\ (((a\to c)\wedge(d\to b))\to e)\wedge(f\to(a\wedge d)) \tag{56}$$

$$(a\to(c\wedge f))\wedge(((c\to e)\wedge(f\to d))\to b)\ \geq\ (a\to c)\wedge(d\to b)\wedge f \tag{57}$$

  Thus, for (56),

$$\begin{aligned}&a\wedge(c\to e)\wedge(f\to d)\\ =\ &\{\wedge\text{ is idempotent and associative}\}\\ &a\wedge(c\to e)\wedge a\wedge(f\to d)\\ \leq\ &\{(15)\}\\ &((a\to c)\to e)\wedge a\wedge(f\to d)\\ \leq\ &\{(5)\}\\ &(((a\to c)\wedge(d\to b))\to e)\wedge a\wedge(f\to d)\\ \leq\ &\{(16)\}\\ &(((a\to c)\wedge(d\to b))\to e)\wedge(f\to(a\wedge d))\end{aligned}$$

  and, for (57),

$$\begin{aligned}&(a\to(c\wedge f))\wedge(((c\to e)\wedge(f\to d))\to b)\\ \geq\ &\{(16)\}\\ &f\wedge(a\to c)\wedge(((c\to e)\wedge(f\to d))\to b)\\ \geq\ &\{(5)\}\\ &f\wedge(a\to c)\wedge((f\to d)\to b)\\ \geq\ &\{(16)\text{ and }\wedge\text{ idempotent}\}\\ &f\wedge(a\to c)\wedge(d\to b)\end{aligned}$$

- To prove (26) notice that,

$$\begin{aligned}(a,b)\Rightarrow((c,d)\Rightarrow(e,f)) &= (a,b)\Rightarrow((c\to e)\wedge(f\to d),c\wedge f)\\ &= ((a\to((c\to e)\wedge(f\to d)))\wedge((c\wedge f)\to b),a\wedge c\wedge f)\\ (c,d)\Rightarrow((a,b)\Rightarrow(e,f)) &= (c,d)\Rightarrow((a\to e)\wedge(f\to b),a\wedge f)\\ &= ((c\to((a\to e)\wedge(f\to b)))\wedge((a\wedge f)\to d),a\wedge c\wedge f)\end{aligned}$$

  Thus, we prove

$$(a\to((c\to e)\wedge(f\to d)))\wedge((c\wedge f)\to b)\ \leq\ (c\to((a\to e)\wedge(f\to b)))\wedge((a\wedge f)\to d)$$

  as follows,

$$\begin{aligned}&(c\to((a\to e)\wedge(f\to b)))\wedge((a\wedge f)\to d)\\ =\ &\{(9)\}\\ &(c\to((a\to e)\wedge(f\to b)))\wedge(a\to(f\to d))\\ \geq\ &\{(14)\}\end{aligned}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

$$
\begin{aligned}
& (c \to (a \to e)) \wedge (c \to (f \to b)) \wedge (a \to (f \to d)) \\
= & \{(8)\} \\
& (a \to (c \to e)) \wedge (c \to (f \to b)) \wedge (a \to (f \to d)) \\
= & \{(9)\} \\
& (a \to (c \to e)) \wedge ((c \wedge f) \to b) \wedge (a \to (f \to d)) \\
\geq & \{+(5)\} \\
& (a \to ((c \to e) \wedge (f \to d))) \wedge ((c \wedge f) \to b) \wedge (a \to ((c \to e) \wedge (f \to d))) \\
= & \{\wedge textidempotent\} \\
& (a \to ((c \to e) \wedge (f \to d))) \wedge ((c \wedge f) \to b)
\end{aligned}
$$

- To prove Properties (27) and (28) observe that,

$$
\begin{aligned}
& ((a, b) \Leftrightarrow (a', b')) \overset{\wedge}{\wedge} ((c, d) \Leftrightarrow (c', d')) \\
= & ((a \leftrightarrow a') \wedge (b \leftrightarrow b'), (a \wedge b') \vee (a' \wedge b)) \overset{\wedge}{\wedge} ((c \leftrightarrow c') \wedge (d \leftrightarrow d'), (c \wedge d') \vee (c' \wedge d)) \\
= & ((a \leftrightarrow a') \wedge (b \leftrightarrow b') \wedge (c \leftrightarrow c') \wedge (d \leftrightarrow d'), (a \wedge b') \vee (a' \wedge b) \vee (c \wedge d') \vee (c' \wedge d))
\end{aligned}
$$

$$
\begin{aligned}
& ((a, b) \overset{\wedge}{\wedge} (c, d)) \Leftrightarrow ((a', b') \overset{\wedge}{\wedge} (c', d')) \\
= & (a \wedge c, b \vee d) \Leftrightarrow (a' \wedge c', b' \vee d') \\
= & ((a \wedge c) \leftrightarrow (a' \wedge c') \wedge (b \vee d) \leftrightarrow (b' \vee d'), (a \wedge c \wedge (b' \vee d')) \vee (a' \wedge c' \wedge (b \vee d)))
\end{aligned}
$$

and,

$$
\begin{aligned}
& ((a, b) \overset{\vee}{\vee} (c, d)) \Leftrightarrow ((a', b') \overset{\vee}{\vee} (c', d')) \\
= & (a \vee c, b \wedge d) \Leftrightarrow (a' \vee c', b' \wedge d') \\
= & (((a \vee c) \leftrightarrow (a' \vee c')) \wedge ((b \wedge d) \leftrightarrow (b' \wedge d')), ((a \vee c) \wedge b' \wedge d') \vee ((a' \vee c') \wedge b \wedge d))
\end{aligned}
$$

Therefore, to prove (27) it is enough to verify the following inequalities,

$$(a \leftrightarrow a') \wedge (b \leftrightarrow b') \wedge (c \leftrightarrow c') \wedge (d \leftrightarrow d') \leq ((a \wedge c) \leftrightarrow (a' \wedge c')) \wedge ((b \vee d) \leftrightarrow (b' \vee d')) \tag{58}$$

$$(a \wedge b') \vee (a' \wedge b) \vee (c \wedge d') \vee (c' \wedge d) \geq (a \wedge c \wedge (b' \vee d')) \vee (a' \wedge c' \wedge (b \vee d)) \tag{59}$$

The inequality (58) follows from applying (6) twice

$$
\begin{aligned}
& (a \leftrightarrow a') \wedge (b \leftrightarrow b') \wedge (c \leftrightarrow c') \wedge (d \leftrightarrow d') \\
\leq & \{(6) \text{ twice and } \wedge \text{ monotone}\} \\
& ((a \wedge c) \leftrightarrow (a' \wedge c')) \wedge ((b \wedge d) \leftrightarrow (b' \wedge d'))
\end{aligned}
$$

For (59)

$$
\begin{aligned}
& ((a \vee c) \wedge b' \wedge d') \vee ((a' \vee c') \wedge b \wedge d) \\
= & \{(10)\} \\
& (a \wedge b' \wedge d') \vee (c \wedge b' \wedge d') \vee (a' \wedge b \wedge d) \vee (c' \wedge b \wedge d) \\
\leq & \{\vee \text{ and } \wedge \text{ monotone}\} \\
& (a \wedge b') \vee (c \wedge d') \vee (a' \wedge b) \vee (c' \wedge d)
\end{aligned}
$$

- To prove (28) the following inequalities have to be shown

$$(a \leftrightarrow a') \wedge (b \leftrightarrow b') \wedge (c \leftrightarrow c') \wedge (d \leftrightarrow d') \leq ((a \vee c) \leftrightarrow (a' \vee c')) \wedge ((b \wedge d) \leftrightarrow (b' \wedge d')) \tag{60}$$

$$(a \wedge b') \vee (a' \wedge b) \vee (c \wedge d') \vee (c' \wedge d) \geq ((a \vee c) \wedge b' \wedge d') \vee ((a' \vee c') \wedge b \wedge d) \tag{61}$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

Inequality (60) follows from applying (7) twice and the fact that $\wedge$ is monotone. The inequality (61) follows by

$$\begin{aligned}
&((a \vee c) \wedge b' \wedge d') \vee ((a' \vee c') \wedge b \wedge d) \\
=& \{(10)\} \\
&(a \wedge b' \wedge d') \vee (c \wedge b' \wedge d') \vee (a' \wedge b \wedge d) \vee (c' \wedge b \wedge d) \\
\leq& \{\vee \text{ monotone}\} \\
&(a \wedge b') \vee (c \wedge d') \vee (a' \wedge b) \vee (c' \wedge d)
\end{aligned}$$

- To prove Properties (29) and (30) start by noting that,

$$\begin{aligned}
&((a, b) \Rightarrow (a', b')) \overset{\wedge}{\wedge} ((c, d) \Rightarrow (c', d')) \\
=& ((a \to a') \wedge (b' \to b), a \wedge b') \overset{\wedge}{\wedge} ((c \to c') \wedge (d' \to d), c \wedge d') \\
=& ((a \to a') \wedge (b' \to b) \wedge (c \to c') \wedge (d' \to d), (a \wedge b') \vee (c \wedge d'))
\end{aligned}$$

For (29) reason

$$\begin{aligned}
&((a, b) \overset{\vee}{\vee} (c, d)) \Rightarrow ((a', b') \overset{\vee}{\vee} (c', d')) \\
=& (a \vee c, b \wedge d) \Rightarrow (a' \vee c', b' \wedge d') \\
=& (((a \vee c) \to (a' \vee c')) \wedge ((b' \wedge d') \to (b \wedge d)), (a \vee c) \wedge b' \wedge d')
\end{aligned}$$

Thus, proving (29) amounts to show the following inequalities

$$(a \to a') \wedge (b' \to b) \wedge (c \to c') \wedge (d' \to d) \leq ((a \vee c) \to (a' \vee c')) \wedge ((b' \wedge d') \to (b \wedge d)) \tag{62}$$

$$(a \wedge b') \vee (c \wedge d') \geq (a \vee c) \wedge b' \wedge d' \tag{63}$$

Let us start by proving (62)

$$\begin{aligned}
&(a \to a') \wedge (b' \to b) \wedge (c \to c') \wedge (d' \to d) \\
\leq& \{(5)\} \\
&(a \to (a' \vee c')) \wedge (c \to (a' \vee c')) \wedge ((b' \wedge d') \to b) \wedge ((b' \wedge d') \to d) \\
\leq& \{(13) \text{ and } (14)\} \\
&((a \vee c) \to (a' \vee c')) \wedge ((b' \wedge d') \to (b \wedge d))
\end{aligned}$$

The prove of inequality (63) follows as,

$$\begin{aligned}
&(a \vee c) \wedge b' \wedge d' \\
=& \{(10)\} \\
&(a \wedge b' \wedge d') \vee (c \wedge b' \wedge d') \\
\leq& \{\vee \text{ monotone}\} \\
&(a \wedge b') \vee (c \wedge d')
\end{aligned}$$

- To prove (30) note that

$$\begin{aligned}
((a, b) \overset{\wedge}{\wedge} (c, d)) \Rightarrow ((a', b') \overset{\wedge}{\wedge} (c', d')) &= (a \wedge c, b \vee d) \Rightarrow (a' \wedge c', b' \vee d') \\
&= (((a \wedge c) \to (a' \wedge c')) \wedge ((b' \vee d') \to (b \vee d)), a \wedge c \wedge (b' \vee d'))
\end{aligned}$$

Thus, we need to prove the following inequalities,

$$(a \to a') \wedge (b' \to b) \wedge (c \to c') \wedge (d' \to d) \leq ((a \wedge c) \to (a' \wedge c')) \wedge ((b' \vee d') \to (b \vee d)) \tag{64}$$

Thus, to prove (64) note that

$$(a \wedge b') \vee (c \wedge d') \geq a \wedge c \wedge (b' \vee d') \tag{65}$$

$$(a \to a') \wedge (b' \to b) \wedge (c \to c') \wedge (d' \to d)$$

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press

$$
\begin{aligned}
&\leq \{\text{Property (5)}\}\\
&\quad ((a \wedge c) \to a') \wedge ((a \wedge c) \to c') \wedge (b' \to (b \vee d)) \wedge (d' \to (b \vee d))\\
&\leq \{\text{Property (13) and (14)}\}\\
&\quad ((a \wedge c) \to (a' \wedge c')) \wedge ((b' \vee d') \to (b \vee d))
\end{aligned}
$$

Finally, (64) follows from

$$
\begin{aligned}
&\quad a \wedge c \wedge (b' \vee d')\\
&= \{\text{Property (10)}\}\\
&\quad (a \wedge c \wedge b') \vee (a \wedge c \wedge d')\\
&\leq \{\vee \text{ is monotone}\}\\
&\quad (a \wedge b') \vee (c \wedge d')
\end{aligned}
$$

Cite this article: Cunha J, Madeira A and Barbosa LS (2025). Paraconsistent transition structures: compositional principles and a modal logic. *Mathematical Structures in Computer Science* **35**(e14), 1–42. https://doi.org/10.1017/S0960129525100170

https://doi.org/10.1017/S0960129525100170 Published online by Cambridge University Press