Three new genuine five-valued logics

Mauricio Osorio[1] and Claudia Zepeda[2]

[1] Universidad de las Américas-Puebla,
[2] Benemérita Universidad Atónoma de Puebla
{osoriomauri,czepedac}@gmail.com

Abstract. We introduce three 5-valued paraconsistent logics that we name FiveASP1, FiveASP2 and FiveASP3. Each of these logics is genuine and paracomplete. FiveASP3 was constructed with the help of Answer Sets Programming. The new value is called e attempting to model the notion of ineffability. If one drops e from any of these logics one obtains a well known 4-valued logic introduced by Avron. If, on the other hand one drops the "implication" connective from any of these logics, one obtains Priest logic FDEe. We present some properties of these logics.

Keywords: many-valued logics, genuine paraconsistent logic, ineffability.

1 Introduction

Belnap [16] claims that a 4-valued logic is a suitable framework for computerized reasoning. Avron in [3,2,4,1] supports this thesis. He shows that a 4-valued logic naturally express true, false, inconsistent or uncertain information. Each of these concepts is represented by a particular logical value. Furthermore in [3] he presents a sound and complete axiomatization of a family of 4-valued logics.

On the other hand, Priest argues in [22] that a 4-valued logic models very well the four possibilities explained before, but here in the context of Buddhist meta-physics, see for instance [26]. This logic is called FDE, but such logic fails to satisfy the well known Modus Ponens inference rule. If one removes the implication connective in this logic, it corresponds to the corresponding fragment of any of the logics studied by Avron. Priest then extends FDE to a 5-valued logic named FDEe, see [23]. This new logic has a new valued e with the aim to represent the notion of ineffability, but FDEe lacks of an implication connective.

Some authors claim that many arguments formulated in Buddhist texts correspond to such well recognized rules of inference as Modus Ponens, constructive dilemma and categorical syllogism (also known as hypothetical syllogism) among other rules of inference, see [21,14]. Having an implication that does not satisfy Modus Ponens or removing the implication connective, can be consider as a kind of weakness of FDEe, see [21].

However Priest makes a remarkable work by studying in great detail the work of the buddhist texts from the Pali Canon to the MMK by Nāgārjuna and beeing able to model their way of reasoning in terms of modern non-classical logics [22,23,24,25].

A main issue is to represent the notion of ineffability. The fifth value, e, then, is the value of ineffable.

There is a complex but well known phenomenon that often arises when a philosophy argues that there are limits to thought/language, and tries to justify this view by giving reasons as to why there are things about which one cannot think/talk—in the process appearing to give the lie to the claim. In poetry we also find a similar situation: the need to talk about extreme situations, which somehow we can not talk [8,17]. Priest is concerned with that phenomenon. According to him, Buddhist philosophy has resources to address this kind of issue much less present in Western traditions. Buddhist logicians consider that there are four possibilities: only true, only false, both true and false, and finally neither true or false. Later developments add a fifth possibility: ineffability[3]. Of course, one might be skeptical that such ideas can be made logically respectable. Priest shows how to accomplish this task with some tools from contemporary non-classical logic. His work is impeccable, but as stated earlier, we consider prudent to extend FDEe logic with an "implication" connective that at least satisfies Modus Ponens.

For the nature of this work is desirable to consider the use of paraconsistent logics [10]. In addition recent work on these logics considers also some useful relative new properties, namely genuineness and paracompleteness, see [7,15,20,18,4]. Arguments in favor of rejecting the law of non-contradiction have been supported more recently by the research done on paraconsistent logics and the applications they have encountered, in particular, in artificial intelligence. Paraconsistent logics accept inconsistencies without presenting the problem that once a contradiction has been derived, then any proposition follows as a consequence, as is the case of classical logic.

We introduce three paraconsistent (genuine and paracomplete) logics that are constructed based on the combination of two logics: BDEe (by Priest) and a version of Four due to Avron that we call it $BL_{\supset}$. The main point is to add an implication to FDEe that satisfies (at least) Modus Ponens. Furthermore, as a second contribution (a minor one) we briefly explain how to use Answer Set Programming (ASP) [13] to construct one of our logics, namely FiveASP3. According to our experience, we know that always it is very useful to have software tools that help us to analyze logics. One of these tools is the ASP tool called clasp[4], which computes the answer sets of logic programs. ASP is a declarative knowledge representation and logic programming language, it has been used to develop different approaches in the areas of planning, logical agents and artificial intelligence [6,12].

[3] As far as the authors know, buddhist texts never talk explicitly about five possibilities, as they actually mention four cases. However, Priest shows that buddhist narratives assume this sort of incommensurable fifth, see [22,23,24,25].

[4] http://potassco.sourceforge.net

2 Background

In this section, we present two of the more common ways of defining a logic, and provide examples. In Section 2.1 we define a logic from the semantical point of view, particularly via muti-valued systems. On the other hand, in Section 2.2, we present one axiomatic formal system for logic $BL_{\supset}$, provided by Avron in [5].

2.1 Multi-valued logics

A way to define a logic is by means of truth values and interpretations. Multi-valued systems generalize the idea of using the truth tables that are used to determine the validity of formulas in classical logic. It has been suggested that multi-valued systems should not count as logics; on the other hand pioneers such as Łukasiewicz considered such multi-valued systems as alternatives to the classical framework. Like other authors do, we prefer to give to multi-valued systems the benefit of the doubt about their status as logics.

The core of a multi-valued system is its *domain* of values D, where some of such values are special and identified as *designated*. Connectives (e.g. $\wedge$, $\vee$, $\rightarrow$, $\neg$) are then introduced as operators over D according to the particular definition of the logic. An *interpretation* is a function $I\colon \mathcal{L} \rightarrow D$ that maps atoms to elements in the domain. The application of I is then extended to arbitrary formulas by mapping first the atoms to values in D, and then evaluating the resulting expression in terms of the connectives of the logic. A formula is said to be a *tautology* if, for every possible interpretation, the formula evaluates to a designated value. The most simple example of a multi-valued logic is classical logic where: $D = \{0, 1\}$, 1 is the unique designated value, and connectives are defined through the usual basic truth tables.

Not all multi-valued logics must have the four connectives mentioned before, in fact classical logic can be defined in terms of two of those connectives $\neg, \wedge$ (primitive connectives), and the other two (non-primitive) can be defined in terms of $\neg, \wedge$. In case of a logic having the implication connective, it is desirable that it preserves tautologies, in the sense that if $x, x \rightarrow y$ are tautologies, then y is also a tautology. This restriction enforces the validity of Modus Ponens in the logic.

Since we will be working with several logics, we will use subindices next to the connectives to specify to which logic they correspond, for example $\neg_K$ corresponds to the connective $\neg$ of Kleene's logic. In those cases where the given logic is understood from the context, we drop such subindexes. Objects 0, 1, 2 and 3 are part of the semantics of logics studied in this paper and were chosen only for convenience, it does not correspond to natural numbers.

A logic satisfies the principle of explosion (EFQ)) if $x, \neg x \models y$. A logic is paraconsistent if it rejects the principle of explosion. A logic satisfies the principle of non-contradiction (PNC) if $\models \neg(x \wedge \neg x)$. A logic is genuine if it rejects the principle of non-contradiction. A logic satisfies the law of excluded middle if $\models x \vee \neg x$. A logic is paracomplete if it rejects the law of excluded middle.

Kleene's 3-valued logic. The Kleene's 3-valued logic, denote here by K, is defined in [5]. Kleene's logic is a 3-valued logic with truth values in the domain $D = \{0, 1, 3\}$, where 3 is the only designated value[5]. Conjunction and disjunction are defined as the min and max functions respectively, namely $\alpha \wedge \beta = min(\alpha, \beta)$, and $\alpha \vee \beta = max(\alpha, \beta)$. The connectives $\rightarrow_K$ and $\neg_K$ are defined according to the tables given in Table 1. It is important to mention that in this paper we use the implication of Kleene as defined by Avron in [5].

$\rightarrow_K$	0	1	3
0	3	3	3
1	3	3	3
3	0	1	3

x	$\neg_K x$
0	3
1	1
3	0

Table 1. Truth tables of connectives $\rightarrow$ and $\neg$ in Kleene's logic.

The basic 3-valued paraconsistent logic PAC. We consider the domain of the logic PAC as $D = \{0, 2, 3\}$, this logic is a 3-valued paraconsistent logic with 2 and 3 as designated values (We take this domain with the purpose of PAC becomes a fragment of $BL_{\supset}$ logic). The connectives $\neg$, $\wedge$ and $\vee$ have exactly the same properties as those of the logic K. Table 2 shows the truth tables of connectives $\neg_{PAC}$ and $\rightarrow_{PAC}$ [5].

$\rightarrow_{PAC}$	0	2	3
0	3	3	3
2	0	2	3
3	0	2	3

x	$\neg_{PAC} x$
0	3
2	2
3	0

Table 2. Truth tables of connectives $\rightarrow$, $\neg$ in PAC.

Logic $BL_{\supset}$. This logic is a 4-valued logic with truth values in the domain $D = \{0, 1, 2, 3\}$ where 2 and 3 are the designated values. The connectives $\wedge$ and $\vee$, as usually, correspond to the *greatest lower bound* (Glb) and the *least upper bound* (Lub), respectively. The connectives $\neg$ and $\rightarrow$ are defined according to the truth tables given in Table 3.

[5] The reason for considering this domain is that these values and the behavior of its connectives coincide with part of the logic $BL_{\supset}$.

$\rightarrow_{BL_\supset}$	0	1	2	3
0	3	3	3	3
1	3	3	3	3
2	0	1	2	3
3	0	1	2	3

x	$\neg_{BL_\supset} x$
0	3
1	1
2	2
3	0

Table 3. Truth tables of connectives $\rightarrow$ and $\neg$ in logic $BL_\supset$.

The logic $BL_\supset$ is represented in Fig. 1. Note that if we consider only the values 0, 1 and 3 (right part of the Fig. 1) we obtain the Kleene's logic while if we take the values 0, 2 and 3 (left part of the Fig. 1) we have the PAC logic.

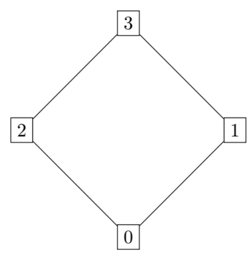

Fig. 1. Lattice logic $BL_\supset$

As A. Avron mentions in [3], $BL_\supset$ is *interlaced* [6] and hence satisfies $1 \wedge 2 = 0$ and $1 \vee 2 = 3$. As a consequence of this result, we can take $D = \{1, 2\}$. However, for simplicity we use $D = \{0, 1, 2, 3\}$.

2.2 The system HBL

Let us consider HBL, a formal axiomatic theory for $BL_\supset$ [3] formed by the primitive logical connectives: $\neg, \rightarrow, \wedge$ and $\vee$. We also consider one logical connective defined in terms of the primitive ones:

$$\alpha \leftrightarrow \beta := (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$$

the well-formed formulas are constructed as usual, the axiom schemas are:

I1 $\alpha \rightarrow (\beta \rightarrow \alpha)$
I2 $(\alpha \rightarrow (\beta \rightarrow \gamma)) \rightarrow ((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma))$
I3 $((\alpha \rightarrow \beta) \rightarrow \alpha) \rightarrow \alpha$
C1 $(\alpha \wedge \beta) \rightarrow \alpha$

[6] This means that each one of $\wedge$, and $\vee$ is monotonic with respect to both $\leq_t$ and $\leq_k$ [3]

C2	$(\alpha \wedge \beta) \to \beta$
C3	$\alpha \to (\beta \to (\alpha \wedge \beta))$
D1	$\alpha \to (\alpha \vee \beta)$
D2	$\beta \to (\alpha \vee \beta)$
D3	$(\alpha \to \gamma) \to ((\beta \to \gamma) \to (\alpha \vee \beta \to \gamma))$
N1	$\neg(\alpha \vee \beta) \leftrightarrow \neg\alpha \wedge \neg\beta$
N2	$\neg(\alpha \wedge \beta) \leftrightarrow \neg\alpha \vee \neg\beta$
N3	$\neg\neg\alpha \leftrightarrow \alpha$
N4	$\neg(\alpha \to \beta) \leftrightarrow \alpha \wedge \neg\beta$

and as the only inference rule: Modus Ponens

$$\frac{\alpha \quad \alpha \to \beta}{\beta}$$

Logic $BL_{\supset}$ is sound and complete with respect to this axiomatization.

Theorem 1. *[3][Soundness and Completeness]*

$$\Gamma \vdash_{BL_{\supset}} \alpha \ if \ and \ only \ if \ \Gamma \models_{BL_{\supset}} \alpha.$$

2.3 First Degree Entailment system

This subsection is a summary of some material from [23] that we need to borrow for the definition of our logics.

First Degree Entailment (FDE) is a system of logic defined by Priest that can be set up in many ways, but one of these is as a 4-valued logic whose values are t (true only), f (false only), b (both), and n (neither). Negation maps t to f, vice versa, n to itself, and b to itself. Conjunction is greatest lower bound, and disjunction is least upper bound. The set of designated values, D, is $\{b, t\}$ The four corners of truth and the FDE logic seem like a correct match.

FDE can be characterised by the following sound and complete rule system, where a double line indicates a two-way rule, and overlining indicates discharging an assumption[7]:

$$\frac{\frac{A,B}{A \wedge B}}{} \qquad \frac{A \wedge B}{A(B)}$$

$$\frac{A(B)}{A \vee B} \qquad \frac{A \vee B \quad \overline{A} \ldots C \quad \overline{B} \ldots C}{C}$$

$$\frac{\neg(A \wedge B)}{\neg A \vee \neg B} \qquad \frac{\neg(A \vee B)}{\neg A \wedge \neg B} \qquad \frac{\neg\neg A}{A}$$

Now we move to FDEe, a 5-valued logic that incorporates the notion of ineffability. According to Priest, technically, the obvious thought is to add a new value, e, to our existing four $\{t, f, b, n\}$), expressing this new status.

Since e is the status of claims such that neither they nor their negations should be accepted, it should obviously not be designated. Thus, we still have

[7] The paper [23] downloaded from the Priest's home page has a typo. It says: $\frac{\neg(A \vee B)}{\neg A \vee \neg B}$ instead of: $\frac{\neg(A \vee B)}{\neg A \wedge \neg B}$.

that same designated values. Priest addresses the following major question: How are the connectives to behave with respect to e?

Both e and n are the values of things that are, in some sense, neither true nor false, but they need to behave differently if the two are to represent distinct alternatives. The simplest suggestion is to take e to be such that whenever any input has the value e, so does the output: e-in/e-out.

The logic that results by modifying FDE in this way is obviously a sub-logic of it. It is a proper sub-logic. It is not difficult to check that all the rules of FDE are designation-preserving except the rule for disjunction-introduction, which is not, as an obvious counter-model shows. However, replace this with the rules:

$$\frac{\varphi(A) \;\; C}{A \vee C} \quad \frac{\varphi(A) \;\; C}{\neg A \vee C} \quad \frac{\varphi(A) \;\; \psi(B) \;\; C}{(A \wedge B) \vee C}$$

where $\varphi(A)$ and $\psi(B)$ are any sentences containing A and B. Call these the φ *Rules*, and call this system FDE_φ. FDE_φ is sound and complete with respect to the semantics.

3 Our 5-valued logics

We present three logics that are constructed based on the combination of two logics: BDEe (by Priest) and $BL_\supset$.

The core of the three logics is based on the following assumptions.

We have 5 values: $\{0, 1, 2, 3, e\}$. FDEe uses $\{f, n, b, t, e\}$ instead. The designated values are $\{2, 3\}$ as FDEe. $\{0, 1, 2, \}$ defines a lattice where $0 < 1$, $0 < 2$, $1 < 3$, $2 < 3$. The connective $\vee$ is the *lub*, while $\wedge$ is the glb.

Since e is interpreted as ineffable then $X \; op \; e = e \; op \; X = e$, where $X \in \{0, 1, 2, 3, e\}$. With respect to negation $(-)$, we have $-0 = 3$, $-3 = 0$, $-1 = 1$, $-2 = 2$, $-e = e$.

Implication is as defined by Avron for the subdomain $\{0, 1, 2, 3\}$, namely: $X \to Y = 3$ when X is not designated, $X \neq e$, $Y \neq e$. While $X \to Y = Y$ when X is designated, $Y \neq e$.

The next two expressions are yet undefined $e \to X$ and $e \to X$ for X in $\{0, 1, 2, 3, e\}$.

Notice that the sublogic defined in the subdomain $\{0, 1, 2, 3\}$ corresponds exactly to $BL_\supset$ logic. Also the sublogic in the domain $\{0, 1, 2, 3, e\}$ but eliminating the implication connective correspond exactly to FDEe.

3.1 FiveASP1

This logic tries to stay very close to FDEe. We define $e \to X = X \to e = e$ for $X \in \{0, 1, 2, 3, e\}$, and we name this logic as FiveASP1.

Theorem 2. *FiveASP1 is a paraconsistent, genuine and paracomplete logic.*

Proof (sketch). Directly using truth tables. For example, to prove that it is paracomplete, it is enough to evaluate the formulas with $val(X) = 1$, for every atom X.

FDEe admits no tautologies. FiveASP1 is faithful in this aspect to FDEe and hence we have the following result.

Theorem 3. *FiveASP1 admits no tautologies.*

Proof (sketch). Evaluating each atom in any formula with e, then the final evaluation is e which is not designated.

Theorem 4. *FiveASP satisfies:*

1. *Modus ponens and Hipothetical sylogism.*
2. *All inference rules of FDEe.*

Proof (sketch). (case 1) They are proven by contradiction using truth tables. (case 2) They are proven by construction, since the three logics behave as logic FDEe regarding the connectives $\wedge$, $\vee$ and negation, and that logic satisfies such inference rules.

3.2 FiveASP2

This logic is somehow the "middle way" between logics FiveASP1 and FiveASP3 (to be introduced soon). It only changes "$e \to e = e$" (in FiveASP1) to "$e \to e = 3$" (in FiveASP2), in order to allow some basic tautologies. Recall that the notion of ineffable is to some extend paradoxical, we can not talk about something ineffable but actually we do it in order to convey a given major message (at least partially). Here, we have that: if X is ineffable and is true and only true that $X \to Y$, then (our logic claims that) Y is ineffable.

We define $e \to X = X \to e = e$ for $X \in \{0, 1, 2, 3\}$ and $e \to e = 3$ and we name this logic as FiveASP2.

Theorem 5. *FiveASP2 is a paraconsistent, genuine and paracomplete logic.*

Proof (sketch). Directly using truth tables.

We can observe that FiveASP2 satisfies some well known tautologies, as $X \to X$ and De Morgan laws, among some of them. Hence, we have the following theorem.

Theorem 6. *FiveASP2 admits some tautologies.*

Proof (sketch). FiveASP2 accept the mentioned tautologies in the phrase previous to this theorem and they are proven directly.

Theorem 7. *FiveASP2 satisfies:*

1. *Modus Ponens and Hipothetical Sylogism.*
2. *All inference rules of FDEe.*

Proof (sketch). (case 1) They are proven by contradiction using truth tables. (case 2) They are proven by construction, since the three logics behave as logic FDEe regarding the connectives $\wedge$, $\vee$ and negation, and that logic satisfies such inference rules.

$\to_{FiveASP3}$	0	1	2	3	4
0	3	3	3	3	3
1	3	3	3	3	3
2	0	1	2	3	1
3	0	1	2	3	1
4	3	3	3	3	3

Table 4. Truth table of connective $\to$ in logic FiveASP3.

3.3 FiveASP3

This logic is kind of pragmatic and the connective "Implication" is a kind of metalinguistic connective [9], we name this logic as FiveASP3. We do not allow to have two values X, Y such that $X \to Y = e$, and hence FiveASP3 is the other extreme case with respect to FiveASP1 logic. Note that this logic gains many well know tautologies. This logic complies with the tautologies I1-I3, C1-C3, D3, N1-N4 (see section 2.2). We consider this logic potentially useful in Artificial Intelligence applied to art (literature). The connective $\to$ is defined according to the truth table given in Table 4.

Theorem 8. *FiveASP3 is a paraconsistent, genuine and paracomplete logic.*

Proof (sketch). Directly using truth tables.

We can observe that FiveASP3 satisfies many well known tautologies, as $X \to X$ and De Morgan laws, the standard two implication rules for positive logic among some of them. Hence, we have the following theorem.

Theorem 9. *FiveASP3 admits some tautologies.*

Proof (sketch). FiveASP3 accepts the mentioned tautologies in the phrase previous to this theorem and they are proven directly.

Theorem 10. *FiveASP3 satisfies:*

1. *Modus Ponens and Hipothetical Sylogism.*
2. *All inference rules of FDEe.*

Proof (sketch). (case 1) They are proven by contradiction using truth tables. (case 2) They are proven by construction, since the three logics behave as logic FDEe regarding the connectives $\wedge$, $\vee$ and negation, and that logic satisfies such inference rules.

Recall that ASP is logic programming that allows to write the specification of a problem (defining only the "what" with no concern to the "how"). On this regard, it follows a generate and test strategy. In this case our concern is to define an implication operator that satisfies certain given constraints. To define such implication operator we write:

$$1\{impl(X, Y, Z) : v(Z)\}1 : -v(x), v(Y).$$

Meaning that given the domain for X, Y (in this case $v(X)$, $v(Y)$) we define a function "impl" with co-domain Z (defined by $v(Z)$).

The rest of the code provides basic definitions such as the domain, the designated values, etc. and also some constraints (test part) such as for example:

$$: -not\ \ impl(Y, X, 3), v1(Y), v1(X), notdes(Y).$$

This constraint says that our implication operator should behaved as $BL_{\supset}$, namely that if the first argument of the implication operator is not designated (and belongs to the subdomian of this 4-valued logic) then our operator should evaluate to 3. In the appendix A, the complete program code with some further comments is presented.

4 Conclusions

We introduce three 5-valued logics by combining FDEe and $BL_{\supset}$. The three of them satisfy the following properties: they are paraconsistent, genuine and paracomplete logics. They satisfy Modus Ponens and Hypothetical Syllogism as well as all inference rules of FDEe. More research needs to be done to understand these logics and to find further mathematical properties of each of them. We also believe that our logics or some extensions of them could be used to represent/understand complex poems where inconsistent, uncertain and/or ineffable beliefs are considered.

It is also interesting to consider extending these logics by defining a possibilistic version of each of our three logics, see [11,19]. In this way we could (perhaps) gain a new dimension of expressibility of some notions.

References

1. Ofer Arieli and Arnon Avron. Logical bilattices and inconsistent data. In *Proceedings of the Ninth Annual Symposium on Logic in Computer Science (LICS '94), Paris, France, July 4-7, 1994*, pages 468–476, 1994.
2. Ofer Arieli and Arnon Avron. Four-valued diagnoses for stratified knowledge-bases. In *Computer Science Logic, 10th International Workshop, CSL '96, Annual Conference of the EACSL, Utrecht, The Netherlands, September 21-27, 1996, Selected Papers*, pages 1–17, 1996.
3. Ofer Arieli and Arnon Avron. Reasoning with logical bilattices. *Journal of Logic, Language and Information*, 5(1):25–63, 1996.
4. Ofer Arieli and Arnon Avron. Four-valued paradefinite logics. *Studia Logica*, 105(6):1, 2017.
5. Arnon Avron. Natural 3-valued logics–characterization and proof theory. *Journal of Symbolic Logic*, 56(1):3–4, 03 1991.
6. Chitta Baral and Michael Gelfond. Logic programming and knowledge representation. *Journal of Logic Programming*, 19:73–148, 1994.

7. Jean-Yves Beziau. Two genuine 3-valued paraconsistent logics. In *Towards Paraconsistent Engineering*, pages 35–47. Springer, 2016.
8. E. Black. Mouthlessness and ineffability in world war i poetry and the waste land. *War, Literature and the Arts: An International Journal of the Humanities*, 25:1–17, 2013.
9. John Corcoran. Meanings of implication. *Diálogos. Revista de Filosofía de la Universidad de Puerto Rico*, 9(24):59–76, 1973.
10. Newton C. A. da Costa, Décio Krause, and Otávio Bueno. *Paraconsistent Logics and Paraconsistency*, pages 791–911. Elsevier, 2007.
11. Didier Dubois and Henri Prade. *Possibility Theory - An Approach to Computerized Processing of Uncertainty*. Springer, 1988.
12. Michael Gelfond and Marcelo Balduccini. Diagnostic reasoning with A-Prolog. *TPLP*, 3(4-5):425–461, 2003.
13. Michael Gelfond and Vladimir Lifschitz. The Stable Model Semantics for Logic Programming. In R. Kowalski and K. Bowen, editors, *5th Conference on Logic Programming*, pages 1070–1080. MIT Press, 1988.
14. Brendan S. Gillon. An early budhist text on logic: Fang Bian Xin Lun. *Argumentation*, 1(22):15–25, 2008.
15. A. Hernández-Tello, J. Arrazola Ramírez, and M. Osorio Galindo. Mouthlessness and ineffability in world war i poetry and the waste land. *Log. Univers.*, 11(4):507—-524, 2017.
16. Dunn J.M. and Epstein G., editors. *Modern Uses of Multiple-Valued Logic*, volume 2 of *Episteme (A Series in the Foundational, Methodological, Philosophical, Psychological,Sociological, and Political Aspects of the Sciences, Pure and Applied)*, chapter A Useful Four-Valued Logic. Springer, Dordrecht, 1977.
17. Lowry Nelson. The rhetoric of ineffability: Toward a definition of mystical poetry. *Comparative Literature*, 8(4):323–336, 1956.
18. M. Osorio, J.L. Carballido, and C. Zepeda. Sp3b as an extension of c_1. *South American Journal of Logic*, 4(1):1–27, 2018.
19. Mauricio Osorio and Juan Carlos Nieves. Pstable semantics for possibilistic logic programs. In *MICAI 2007: Advances in Artificial Intelligence, 6th Mexican International Conference on Artificial Intelligence, Aguascalientes, Mexico, November 4-10, 2007, Proceedings*, pages 294–304, 2007.
20. Y. Petrukhin. Generalized correspondence analysis for three-valued logics. *Log. Univers.*, 12(3):423–460, 2018.
21. G Priest, K Tanaka, Y Deguchi, and J Garfield, editors. *The Moon Points Back*, chapter Nāgārjuna's logic. Oxford University Press, 2015.
22. Graham Priest. The logic of the catuskoti. *Comparative Philosophy*, 1(2):24–54, 2010.
23. Graham Priest. None of the above: The catuskoti in indian buddhist logic. In *New Directions in Paraconsistent Logic*, pages 517–527. Springer, 2015.
24. Graham Priest. Speaking of the ineffable, east and west. *European Journal of Analytic Philosophy*, 11(2):6–20, 2015.
25. Graham Priest. *The Fifth Corner of Four. An Essay on Buddhist Metaphysics and the Catuskoti*. Oxford University Press, 2019.
26. Eric Swanson and Mingyur Rinpoche Yongey. *Joyful Wisdom: Embracing Change and Finding Freedom*. Three Rivers Press, 2010.

A Appendix

`FiveASP3.clasp` is a program that builds a logic of five values from two logic, one from Priest and another from Avron. The logic of Priest has five values but does not include implication connective. Avron's logic has only four values but includes all the connectives. We verify known tautologies, among these: $(x \wedge y) \to x$, $(x \wedge y) \to y$, $x \to (y \to (x \wedge y))$, $(x \vee y) \to ((x \to z) \to ((y \to z)) \to z)$, $y \leftrightarrow \neg\neg y$, Pierce formula $((x \to y) \to x) \to x$.

```
#show impl/3.

v1(0..3). v(e).  v(X):- v1(X).   des(2..3).

%% Definition AND (Priest)
and(X,X,X) :- v1(X).   and(0,X,0) :- v1(X).   and(X,0,0):- v1(X).
and(X,3,X) :- v1(X).   and(3,X,X) :- v1(X).   and(1,2,0).
and(2,1,0).            and(e,X,e) :- v(X).    and(X,e,e) :- v(X).

%% Definition OR (Priest)
or(X,X,X):- v1(X).   or(0,X,X):- v1(X).   or(X,0,X):- v1(X).
or(X,3,3):- v1(X).   or(3,X,3):- v1(X).   or(1,2,3).
or(2,1,3).           or(e,X,e) :- v(X).


% impl (Any function of 2 arguments)
 1 { impl(X,Y,Z) : v(Z) } 1 :- v(X), v(Y).

% Restrictions for the implication of Avron (4-valued logic)
:- not impl(Y,X,3), v1(Y), v1(X), not des(Y).
:- not impl(Y,X,X), v1(Y), v1(X), des(Y).

% Traditional definition of equivalence
equ(X,Y,Z) :- impl(X,Y,L), impl(Y,X,R), and(L,R,Z).

% Definition of negation according to Priest
neg(0,3).
neg(3,0).
neg(1,1).
neg(2,2).
neg(e,e).

%% inference rules MP
:- impl(X,Y,Z), des(X), des(Z), v(Y), not des(Y).

%% axioms 1,2 impl,
%X -> ( Y -> X)
:- impl(Y,X,Z), impl(X,Z,R), v(R), not des(R).

% (X -> (Y -> Z))  -> ( ( X -> Y) -> (X -> Z)
```

```
:- impl(Y,Z, L1), impl(X,L1,L), impl(X,Y,R1), impl(X,Z,R2), impl(R1,R2,R),
impl(L,R,S), v(S), not des(S).

%% -(A ->B) <-> (A & -B)
:- impl(A,B,L), neg(L,L1), neg(B,B1), and(A,B1,R), equ(L1,R,Q),
not des(Q), v(Q).

%%% FINISHES Basic construction.
%There are exactly 4 logics that meet the above.

%% We eliminate the one that evaluate e for values at {0,1,2,3} x {e}
% to contrast with logic fiveASP1. The result is a single logic.
:- impl(X,e,e), v1(X).
```