The N_5' Logic

José Arrazola[1], Mauricio Osorio[2], and Eduardo Ariza[1]

[1]Benemérita Universidad Autónoma de Puebla
aveariza@hotmail.com, arrazola@fcfm.buap.mx

.

[2]Universidad de las Américas - Puebla
osoriomauri@gmail.com

Abstract. We introduce a new 5-valued logic that we call N_5'. This logic extends GLukG, a paraconsistent logic recently introduced [10]. We show that N_5' is sound with respect to GLukG.

1 Introduction

GLukG paraconsistent logic has been investigated for a short period of time [5,10]. However, it has demonstrated important qualities for the study of other logics. It also has relevant attributes in knowledge representation [11]. Although, this work does not address this last point. On this paper, we investigate how to add a strong negation connective to GLukG logic.

The logic that is of our interest is a 5-valued logic that is sound with respect to GLukG. In this case we use two kinds of negations, one is the strong negation represented by the $\sim$ symbol and the other one is the default negation represented by the $\neg$ symbol. In the literature we can find several logical extension, through a strong negation operator, but few of them are hardly paraconsistent logics satisfy a theorem of substitution, to see [3]. To summarize, the main contribution of the paper is the proposal of the paraconsistent 5-valued logic called N_5' that satisfies the following suitable properties: (1) It is a conservative extension of GLukG logic. (2) It satisfies the substitution theorem.

The structure of our paper is as follows. Section 2 describes the general background needed for the paper including the definition of GLukG logic. On Section 3 we present a Hilbert-style axiomatization for GLukG which is a slight variation of the one presented in [10]. On Section 4 we present Theorem 4, our main result, which establishes that our logic N_5' is a conservative extension of Nelson's N_5. Finally, on Section 5 we present our conclusions and we address the future work.

We assume that the reader has some familiarity with basic logic such as chapter one in [9].

2 Background

We first introduce the syntax of logic formulas considered in this paper. Then we present a few basic definitions of how logics can be built to interpret the meaning of such formulas in order to, finally, give a brief introduction to several of the logics that are relevant for the results of our later sections.

2.1 Syntax of formulas

We consider a formal (propositional) language built from: an enumerable set $\mathcal{L}$ of elements called *atoms* (denoted a, b, c, $\ldots$); the binary connectives $\wedge$ (*conjunction*), $\vee$ (*disjunction*) and $\rightarrow$ (*implication*); and the unary connective $\neg$ (*negation*). Formulas (denoted α, β, γ, $\ldots$) are constructed as usual by combining these basic connectives together with the help of parentheses.

We also use $\alpha \leftrightarrow \beta$ to abbreviate $(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$ and $\alpha \leftarrow \beta$ to abbreviate $\beta \rightarrow \alpha$. It is useful to agree on some conventions to avoid the use of so many parenthesis in writing formulas. This will make the reading of complicated expressions easier. First, we may omit the outer pair of parenthesis of a formula. Second, the connectives are ordered as follows: $\neg, \wedge, \vee, \rightarrow$, and $\leftrightarrow$, and parentheses are eliminated according to the rule that, first, $\neg$ applies to the smallest formula following it, then $\wedge$ is to connect the smallest formulas surrounding it, and so on.

A *theory* is just a set of formulas and, in this paper, we only consider finite theories. Moreover, if T is a theory, we use the notation $\mathcal{L}_T$ to stand for the set of atoms that occur in the theory T, if $T = \{\phi\}$ we denote $\mathcal{L}_\phi$. A literal l is either an atom or the negation of an atom.

2.2 Logic systems

We consider a *logic* simply as a set of formulas that, moreover, satisfies the following two properties: (i) is closed under modus ponens (i.e. if α and $\alpha \rightarrow \beta$ are in the logic, then so is β) and (ii) is closed under substitution (i.e. if a formula α is in the logic, then any other formula obtained by replacing all occurrences of an atom b in α with another formula β is still in the logic). The elements of a logic are called *theorems* and the notation $\vdash_X \alpha$ is used to state that the formula α is a theorem of X (i.e. $\alpha \in X$). We say that a logic X is *weaker than or equal to* a logic Y if $X \subseteq Y$, similarly we say that X is *stronger than or equal to* Y if $Y \subseteq X$.

Hilbert style proof systems There are many different approaches that have been used to specify the meaning of logic formulas or, in other words, to define *logics*. In Hilbert style proof systems, also known as axiomatic systems, a logic is specified by giving a set of axioms (which is usually assumed to be closed by substitution). This set of axioms specifies, so to speak, the 'kernel' of the logic. The actual logic is obtained when this 'kernel' is closed with respect to the inference rule of modus ponens. The notation $\vdash_X F$ for provability of a logic formula F in the logic X is usually extended within Hilbert style systems; given a theory T, we use $T \vdash_X F$ to denote the fact that the formula F can be derived from the axioms of the logic and the formulas contained in T by a sequence of applications of modus ponens[1]. Recall that, in all these definitions, the logic connectives are parameterized by some underlying logic, e.g. the expression $\vdash_X (F_1 \wedge \cdots \wedge F_n) \to F$ actually stands for $\vdash_X (F_1 \wedge_X \cdots \wedge_X F_n) \to_X F$.

C_ω logic [6] is defined by the following set of axioms:

Pos1	$a \to (b \to a)$
Pos2	$(a \to (b \to c)) \to ((a \to b) \to (a \to c))$
Pos3	$a \wedge b \to a$
Pos4	$a \wedge b \to b$
Pos5	$a \to (b \to (a \wedge b))$
Pos6	$a \to (a \vee b)$
Pos7	$b \to (a \vee b)$
Pos8	$(a \to c) \to ((b \to c) \to (a \vee b \to c))$
C_ω**1**	$a \vee \neg a$
C_ω**2**	$\neg\neg a \to a$

Note that the first 8 axioms somewhat constraint the meaning of the $\to$, $\wedge$ and $\vee$ connectives to match our usual intuition. It is a well known result that in any logic satisfying axioms **Pos1** and **Pos2**, and with *modus ponens* as its unique inference rule, the *Deduction Theorem* holds [9].

Multivalued logics An alternative way to define the semantics for a logic is by the use of truth values and interpretations. Multivalued logics generalize the idea of using truth tables that are used to determine the validity of formulas in classical logic. The core of a multivalued logic is its *domain* of values $\mathcal{D}$, where some of such values are special and identified as *designated*. Logic connectives (e.g. $\wedge$, $\vee$, $\to$, $\neg$) are then introduced as operators over $\mathcal{D}$ according to the particular definition of the logic.

An *interpretation* is a function $I\colon \mathcal{L} \to \mathcal{D}$ 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 $\mathcal{D}$, and then evaluating the resulting expression in terms of the connectives of the logic (which are defined over $\mathcal{D}$). 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 multivalued logic is classical logic where: $\mathcal{D} = \{0, 1\}$, 1 is the unique designated value, and connectives are defined through the usual basic truth tables. If X is any logic, we write $\models_X \alpha$ to denote that α is a tautology in the logic X. We say that α is a logical consequence of a set of formulas $\Gamma = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ (denoted by $\Gamma \models_X \alpha$) if $\bigwedge \Gamma \to \alpha$ is a tautology, where $\bigwedge \Gamma$ stands for $\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n$.

Note that in a multivalued logic, so that it can truly be a *logic*, the implication connective has to satisfy the following property: for every value $x \in \mathcal{D}$, if there is a designated value $y \in \mathcal{D}$ such that $y \to x$ is designated, then x must also be a designated value. This restriction enforces the validity of modus ponens in the logic. The inference rule of substitution holds without further conditions because of the functional nature of interpretations and how they are evaluated.

Given a theory T, we define the negation of the theory $\neg T$ as $\{\neg F \mid F \in T\}$ (the negation symbol is parameterized with respect to some given logic). For

[1] We drop the subscript X in $\vdash_X$ when the given logic is clear from the context.

any pair of theories T and U, we use $T \vdash_X U$ to state that $T \vdash_X F$ for every formula $F \in U$.

3 Axiomatization of GLukG

We present the Hilbert-style axiomatization of GLukG that is a slight (equivalent) variant of the one presented in [10]. This logic has three primitive logical connectives, namely $\mathcal{GL} := \{\rightarrow, \wedge, \neg\}$. *GLukG*-formulas (or *GL*-formulas) are formulas built from these connectives in the standard form. We also have three defined connectives:

1. $\alpha \vee \beta := ((\alpha \rightarrow \beta) \rightarrow \beta) \wedge ((\beta \rightarrow \alpha) \rightarrow \alpha)$.
2. $-\alpha := \alpha \rightarrow (\neg\alpha \wedge \neg\neg\alpha)$.
3. $\alpha \leftrightarrow \beta := (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$.

GLukG Logic has all the axioms of C_ω logic plus the following:

E1 $(\neg\alpha \rightarrow \neg\beta) \leftrightarrow (\neg\neg\beta \rightarrow \neg\neg\alpha)$
E2 $\neg\neg(\alpha \rightarrow \beta) \leftrightarrow ((\alpha \rightarrow \beta) \wedge (\neg\neg\alpha \rightarrow \neg\neg\beta))$
E3 $\neg\neg(\alpha \wedge \beta) \leftrightarrow (\neg\neg\alpha \wedge \neg\neg\beta)$
E4 $(\beta \wedge \neg\beta) \rightarrow (- - \alpha \rightarrow \alpha)$

Note that Classical logic is obtained from GLukG by adding to the list of axioms any of the following formulas: $\alpha\rightarrow\neg\neg\alpha, \alpha\rightarrow(\neg\alpha\rightarrow\beta), (\neg\beta\rightarrow\neg\alpha)\rightarrow(\alpha\rightarrow\beta)$. It is shown in [11]. On the other hand, $-\alpha \rightarrow \neg\alpha$ is a theorem in GLukG the "$-$" connective is called strong negation. Some experimental work as well as the experience with answer set programming suggested to consider a more useful negation connective for NMR that we will introduce in the following section. See [17] to understand this point in the context of ASP.

Theorem 1 ([11]). *For every formula α, α is a tautology in G_3' iff α is a theorem in* GLukG*.*

In this paper we consider the *standard* substitution, here represented with the usual notation: $\varphi[\alpha/p]$ will denote the formula that results from substituting the formula α for the atom p, wherever it occurs in φ. Recall the recursive definition: if φ is atomic, then $\varphi[\alpha/p]$ is α when φ equals p, and φ otherwise. Inductively, if φ is a formula $\varphi_1\#\varphi_2$, for any binary connective $\#$. Then $\varphi[\alpha/p]$ will be $\varphi_1[\alpha/p]\#\varphi_2[\alpha/p]$. Finally, if φ is a formula of the form $\neg\varphi_1$, then $\varphi[\alpha/p]$ is $\neg\varphi_1[\alpha/p]$.

3.1 Multivalued logic

It is very important to note that GLukG can also be presented as a multivalued logic. Such presentation is given in [13], where GLukG is called G_3'. In this form it is defined through a 3-valued logic with truth values in the domain $\mathcal{D} = \{0, 1, 2\}$ where 2 is the designated value. The evaluation function of the logic connectives is then defined as follows: $x \wedge y = \min(x, y)$; $x \vee y = \max(x, y)$; and the $\neg$ and $\rightarrow$

connectives are defined according to the truth tables given in Table 1. We write $\models \alpha$ to denote that the formula α is a tautology, namely that α evaluates to 2 (the designated value) for every valuation. We say that α is a logical consequence of a set of formulas $\Gamma = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ (denoted by $\Gamma \models \alpha$) if $\bigwedge \Gamma \to \alpha$ is a tautology, where $\bigwedge \Gamma$ stands for $\varphi_1 \wedge \varphi_2 \wedge \ldots \wedge \varphi_n$.

In this paper we keep the notion G'_3 to refer to multivalued logic just defined and we use the notion GLukG to refer as the Hilbert system defined at the beginning of this section.

x	$\neg x$
0	2
1	2
2	0

$\to$	0	1	2
0	2	2	2
1	0	2	2
2	0	1	2

Table 1. Truth tables of connectives in G'_3.

Theorem 2 (Substitution theorem for G'_3-logic [10]). *Let α, β and ψ be GL-formulas and let p be an atom. If $\alpha \leftrightarrow \beta$ is a tautology in G'_3 then $\psi[\alpha/p] \leftrightarrow \psi[\beta/p]$ is a tautology in G'_3.*

Corollary 1 ([10]). *Let α, β and ψ be GL-formulas and let p be an atom. If $\alpha \leftrightarrow \beta$ is a theorem in* GLukG *then $\psi[\alpha/p] \leftrightarrow \psi[\beta/p]$ is a theorem in* GLukG.

4 Main Results

We present $N5'$, a 5-valued logic. We will use the set of values $\{-2, -1, 0, 1, 2\}$. Valid formulas evaluate to 2. The connectives $\wedge$ and $\vee$ correspond to the *min* and *max* functions in the usual way. For the other connectives, the associated truth tables are as follows:

$\to$	-2	-1	0	1	2
-2	2	2	2	2	2
-1	2	2	2	2	2
0	2	2	2	2	2
1	-1	-1	0	2	2
2	-2	-1	0	1	2

$\neg$	
-2	2
-1	2
0	2
1	2
2	-2

$\sim$	
-2	2
-1	1
0	0
1	-1
2	-2

We have defined 5 logical connectives, namely $N_c := \{\to, \wedge, \vee, \neg, \sim\}$. N-formulas are built from these set of connectives. If α always evaluates to the designated value then it is called a tautology.

Remark 1. Observed the following:

1. $\models\sim(\alpha\to\beta)\leftrightarrow\alpha\wedge\sim\beta$.
2. $\models\sim(\alpha\wedge\beta)\leftrightarrow\sim\alpha\vee\sim\beta$.
3. $\models\sim(\alpha\vee\beta)\leftrightarrow\sim\alpha\wedge\sim\beta$.
4. $\models\alpha\leftrightarrow\sim\sim\alpha$.
5. $\models\sim\neg\alpha\leftrightarrow\neg\neg\alpha$.
6. $\models\sim\alpha\to\neg\alpha$.

7.– $\models\neg\sim\alpha\to((\neg\sim\alpha\to\alpha)\to\alpha)$.
8.– $\models\sim\alpha\to((\neg\sim\alpha\to\alpha)\to\neg\alpha)$.
9.– $\not\models\sim\alpha\to((\neg\sim\alpha\to\alpha)\to\alpha)$.
10.– $\not\models\sim\alpha\to((\neg\sim\alpha\to\alpha)\to\neg\sim\alpha)$.
11.– $\not\models\neg\alpha\to((\neg\sim\alpha\to\alpha)\to\neg\sim\alpha)$.
12.– $\not\models\neg\alpha\to((\neg\sim\alpha\to\alpha)\to\sim\alpha)$.

What is important about this remark is that these formulae have the same structure than those theorems of Nelson introduced previously in [3] for the construction of extensions, with strong negation, of intuitionistic logic; In fact, formulae $1-6$ show the classical-like behavior of strong negation, in particular, formula 6 honors the adjective *strong*.

Theorem 3. *Given α and β two formulae, then:*
1. *If α and $\alpha\to\beta$ are tautologies, then β is also a tautology.*
2. *If α is a tautology, then $\neg\neg\alpha$ is also a tautology.*

Proof. The proof of 1 is straightforward; If α and $\alpha\to\beta$ are tautologies, then they always evaluate to the designated value and, looking at the implication table given for N_5', we see that (since α evaluates to 2) there is only one case in which $\alpha\to\beta$ evaluates to 2, and it is precisely when β evaluates to 2. This concludes the proof of 1.
To prove 2, we see that if α evaluates to 2, then $\neg\neg\alpha$ also evaluates to 2.

Note: N_5' logic is in some way similar to N_5 logic (see [3] for more information on N_5). The only difference is that in N_5, $\neg 1=-1$, but in N_5', $\neg 1=2$. Moreover, with N_5' logic we can express N_5 logic.

Theorem 4. *N_5' logic can express N_5 logic.*

Proof. It suffices to see that we can express the constant $\perp$ of N_5 by the formula $\neg a\wedge\neg\neg a$ of N_5', where a is any formula. Hence, we can also express the $\neg\alpha$ formula of N_5 by $\alpha\to\perp$ in N_5'.

Lemma 1. *Let I be an interpretation (based on G_3'). Let J_I be an interpretation (based on N_5') defined for every atom a as follows:*

$$J_I(a)=\begin{cases} I(a) & \text{if } I(a)>0\\ -2 & \text{otherwise}\end{cases}$$

Then, for every GL-formula α,

$$J_I(\alpha)=\begin{cases} I(\alpha) & \text{if } I(\alpha)>0\\ <0 & \text{otherwise}\end{cases}$$

Proof. The proof is by induction on the number n of ocurrences of $\to,\wedge,\vee,\neg,\sim$: if $n=0$, α is just an atom a, the proof is followed by definition of J_I. Assume now that lemma holds for all $j\le n$.
Case 1. α is $\neg\beta$. Then β has fewer than n connectives.

1a. $0 < I(\alpha)$ then $I(\alpha) = 2$, hence $I(\beta) = 1$ or $I(\beta) = 0$. If $I(\beta) = 1$, by inductive hypothesis, we have that $J_I(\beta) = I(\beta) = 1$ so $J_I(\alpha) = J_I(\neg\beta) = 2 = I(\alpha)$.
1b. $I(\beta) = 0$, this subcase is analogous to subcase 1a.
Case 2. α is $\beta \to \gamma$. Then β and γ have fewer than n connectives.
2a. $0 < I(\alpha)$ then $I(\alpha) = 1$ or $I(\alpha) = 2$.
i) Assume $1 = I(\alpha)$, so $I(\beta) = 2$ and $I(\gamma) = 1$. By inductive hypothesis, $J_I(\beta) = I(\beta) = 2$ and $J_I(\gamma) = I(\gamma) = 1$ thus $J_I(\alpha) = 1 = I(\alpha)$.
ii) If $I(\alpha) = 2$ is worked analogously the previous subcase.
2b. $I(\alpha) = 0$, we have by inductive hypothesis $0 < I(\beta) = J_I(\beta)$ and $I(\gamma) = J_I(\gamma) < 0$, so $J_I(\alpha) < 0$.
Case 3. α is $\beta \wedge \gamma$. Then β and γ have fewer than n connectives.
3a. $0 < I(\alpha)$ then by inductive hypothesis $0 < I(\alpha) = min\{I(\beta), I(\gamma)\} = min\{J_I(\beta), J_I(\gamma)\} = J_I(\alpha)$.
3b. $I(\alpha) \leq 0$ is similar to subcase 3a.
Finally, the case α is $\beta \vee \gamma$ is proved analogously.

A very important result is that N_5' logic is a conservative extension of G_3' logic, as the following theorem shows.

Theorem 5. *For every GL-formula α, α is a tautology in N_5' iff α is a tautology in G_3'.*

Proof. Suppose that α is a tautology in G_3', then by Theorem 1, α is a theorem in GLukG. We proceed by induction on the length of the proof of α. The base case is immediate from the fact that all GLukG axioms are tautologies in N_5'. We suppose that any proposition, that has a proof with at most n steps, satisfies the induction hypothesis.

Let α be a proposition whose proof requires exactly n steps and let be $B_1, \ldots, B_n$, a proof of α. We are done if $B_n = \alpha$ is a GLukG axiom. If α is a Modus Ponens consequence, then by Theorem 3, if β and $\beta \to \alpha$ are tautologies, then α is tautology. This observation and the inductive hypothesis finish the proof.

For the other implication we suppose that α is not a tautology in G_3', then there exists an interpretation I (based on G_3') such that $I(\alpha) \neq 2$. By Lemma 1, there exists J_I, an interpretation based on N_5', such that $J_I(\alpha) \neq 2$. Hence, α is not a tautology of N_5', as desired.

4.1 Substitution

A particular feature of our N_5' logic is that the symbol $\leftrightarrow$ does not define a *congruential relation* on formulas, note that it can be the case that $\alpha \leftrightarrow \beta$ is a tautology, but $\sim\alpha \leftrightarrow \sim\beta$ doesn't. A particular example is the following: Take α_1 to be $\sim (a \to b)$ and α_2 to be $a \wedge \sim b$. Clearly $\alpha_1 \leftrightarrow \alpha_2$ is a tautology, but $\sim \alpha_1 \leftrightarrow \sim \alpha_2$ does not (take $I(a) = I(b) = 1$). This property also holds in N_5.

Thus, when we refer to *equivalence* of formulas, we will have to be more precise and make some particular considerations. The term *weak equivalence* will mean that $\alpha \leftrightarrow \beta$ is a tautology. There is a stronger notion of equivalence of **N**-formulas, which we will call *N_5'-equivalence*, and it holds when both $\alpha \leftrightarrow \beta$

and $\sim\alpha \leftrightarrow \sim\beta$ are tautologies. For this purpose, we define a new connective $\Leftrightarrow$. We write $\alpha \Leftrightarrow \beta$ to denote the formula: $(\alpha \leftrightarrow \beta) \wedge (\sim\alpha \leftrightarrow \sim\beta)$. The reader can easily verify that $\alpha \leftrightarrow \beta$ is a tautology iff for every valuation v, $v(\alpha) > 0$ implies $v(\alpha) = v(\beta)$, while $\alpha \Leftrightarrow \beta$ is a tautology iff for every valuation v, $v(\alpha) = v(\beta)$. This can be seen in the following truth tables:

$\leftrightarrow$	-2	-1	0	1	2
-2	2	2	2	-1	-2
-1	2	2	2	-1	-1
0	2	2	2	0	0
1	-1	-1	0	2	1
2	-2	-1	0	1	2

$\Leftrightarrow$	-2	-1	0	1	2
-2	2	1	0	-1	-2
-1	1	2	0	-1	-1
0	0	0	2	0	0
1	-1	-1	0	2	1
2	-2	-1	0	1	2

Theorem 6 (Basic Substitution theorem). *Let α, β and ψ be* **N**-*formulas and let p be an atom. If $\alpha \Leftrightarrow \beta$ is a tautology then $\psi[\alpha/p] \leftrightarrow \psi[\beta/p]$ is a tautology.*

Proof. If $\alpha \Leftrightarrow \beta$ is tautology then for every v, an N-valuation, $v(\alpha) = v(\beta)$ (see the comments at the end of the previous paragraph). Therefore, $v(\psi[\alpha/p]) = \psi[\beta/p])$ so, $v(\psi[\alpha/p] \leftrightarrow \psi[\beta/p]) = 2$.

To be able to apply standard substitution we require N_5'-equivalence of formulas to hold. However, in certain cases this condition may be too strong. We are also interested in the particular cases where weak equivalence of formulas suffices for substituting. The first such case is when substitution is not done inside the scope of a $\sim$ symbol.

Lemma 2. *Let α, β and ψ be* **N**-*formulas and let p be an atom such that p does not occur in ψ within the scope of a $\sim$ symbol. If $\alpha \leftrightarrow \beta$ is a tautology then $\psi[\alpha/p] \leftrightarrow \psi[\beta/p]$ is a tautology.*

Proof. It is of high importance to remember (see the comments at the end of the first paragraph) that: $\theta \leftrightarrow \eta$ is a tautology iff for every valuation v, $v(\theta) > 0$ implies $v(\theta) = v(\eta)$. By structural induction:
Base case:

If $\psi = q$, q an atom, we have:

$$\psi[\alpha/p] \leftrightarrow \psi[\beta/p] = \begin{cases} q \leftrightarrow q \text{ if } p \neq q \\ \alpha \leftrightarrow \beta \text{ if } p = q, (hypothesis) \end{cases}$$

therefore it is a tautology.
We suppose that the inductive case is satisfied:
If $\psi = \neg\varphi$

By inductive hypothesis, for every $\mathbf{N}_5'$-valuation, $\varphi[\alpha/p] \leftrightarrow \varphi[\beta/p]$ is a tautology. If $0 < v(\varphi[\alpha/p])$ then $v(\neg\varphi[\alpha/p]) = v(\neg\varphi[\beta/p])$, therefore $\psi[\alpha/p] \leftrightarrow \psi[\beta/p]$ is a tautology.

Remark 2. If $v(\varphi_1) \leq 0$ and $v(\varphi_2) \leq 0$ implies $v(\varphi_1 \to \varphi_2) = 2$

If $\psi = \varphi_1 \wedge \varphi_2$

Suppose that $0 < v((\varphi_1 \wedge \varphi_2)[\alpha/p])$ then, $0 < \min\{v(\varphi_i[\alpha/p])\}_{i=1,2}$. Hence, by the inductive hypothesis $v(\psi_i[\alpha/p]) = v(\psi_i[\beta/p])$ for $i = 1, 2$. Then $v(\psi[\alpha/p]) = \min\{v(\varphi_i[\alpha/p])\}_{i=1,2} = \min\{v(\varphi_i[\beta/p])\}_{i=1,2} = v(\psi[\beta/p])$. Therefore $\psi[\alpha/p] \leftrightarrow \psi[\beta/p]$ is a tautology.

$\psi = \varphi_1 \to \varphi_2$

Remember that by inductive hypothesis we have $\varphi_1[\alpha/p] \leftrightarrow \varphi_1[\beta/p]$ and $\varphi_2[\alpha/p] \leftrightarrow \varphi_2[\beta/p]$ are tautologies.

Suppose that $0 < v(\psi[\alpha/p]) = v(\varphi_1[\alpha/p] \to \varphi_2[\alpha/p])$.

We have two cases in which we can apply the inductive hypothesis:

1. $v(\varphi_1[\alpha/p]) > 0$ then by inductive hypothesis, we have $v(\varphi_1[\alpha/p]) = v(\varphi_1[\beta/p])$, then $0 < v(\varphi_2[\alpha/p])$, again by inductive hypothesis $v(\varphi_2[\alpha/p]) = v(\varphi_2[\beta/p])$, therefore $v(\varphi_1[\alpha/p] \to \varphi_2[\alpha/p]) = v(\varphi_1[\beta/p] \to \varphi_2[\beta/p])$.
2. $v(\varphi_2[\alpha/p]) > 0$ then $v(\varphi_2[\alpha/p]) = v(\varphi_2[\beta/p])$. Since $\varphi_1[\alpha/p] \leftrightarrow \varphi_1[\beta/p]$ and $\varphi_2[\alpha/p] \leftrightarrow \varphi_2[\beta/p]$ are tautologies, then we have that both $\varphi_1[\alpha/p]$ and $\varphi_1[\beta/p]$ are negative or null (not necessarily in the same case). Finally $v(\varphi_1[\alpha/p] \to \varphi_2[\alpha/p]) = v(\varphi_1[\beta/p] \to \varphi_2[\beta/p])$.

4.2 Standard form

We present the notion of a standard form of a formula.

Definition 1. *We define the function S: **N**-formulae $\to$ **N**-formulae as follows: If a is an atom and φ is an **N**-formula, then*

$$\begin{array}{llll} S(a) & = a, & S(\alpha \to \beta) & = S(\alpha) \to S(\beta), \\ S(\neg a) & = \neg a, & S(\alpha \wedge \beta) & = S(\alpha) \wedge S(\beta), \\ S(\sim a) & = \sim a, & S(\sim (\alpha \to \beta)) & = S(\alpha) \wedge S(\sim \beta), \\ S(\sim \neg \alpha) & = S(\neg\neg\alpha), & S(\sim (\alpha \wedge \beta)) & = S(\sim \alpha) \vee S(\sim \beta), \\ S(\neg \alpha) & = \neg S(\alpha), & S(\sim\sim \alpha) & = S(\alpha). \end{array}$$

Definition 2 (Standard Form). *An **N**-formula φ is said to be in* standard form *if $S(\varphi) = \varphi$*

Intuitively a formula is in standard form if it has all occurrences of the $\sim$ connective just in front of an atom.

Example 1. Take the formula $\varphi := \sim (a \to \neg b) \wedge \sim c$. Then its standard form is $S(\varphi) := a \wedge b \wedge \sim c$.

Lemma 3. *For any **N**-formula φ, φ is a tautology in N_5' iff $S(\varphi)$ is a tautology in N_5'.*

Proof. By structural induction:
Base case: It is vacuously true.

We suppose that the inductive case is satisfied:

$\psi = \neg\varphi$ is a tautology iff φ evaluates to $-2, -1, 0$ or 1 iff, by inductive hypothesis, $2 = \neg S(\varphi) = S(\psi)$.

$\psi = \varphi_1 \wedge \varphi_2$ is a tautology iff φ_1 and φ_2 have the same value, by inductive hypothesis, $S(\psi) = S(\varphi_1) \wedge S(\varphi_2)$ is tautology.

If $\psi = \varphi_1 \rightarrow \varphi_2$ is tautology, we have two cases:
1.- if φ_1 has any of the values $-2, -1$ or 0 then by inductive hypothesis also $S(\varphi_1)$ the same value; hence it does not matter the value of value $S(\varphi_2)$ and $S(\psi) = S(\varphi_1 \rightarrow \varphi_2) = S(\varphi_1) \rightarrow S(\varphi_2)$ is a tautology, and reciprocally
2.- if the value of φ_1 is 1 or 2 then φ_2 has the same value, respectively; hence, by the inductive hypothesis we have $S(\psi)$ is tautology.

If $\psi = \sim \varphi$ is a tautology then by remark 1 and the inductive hypothesis we have that $S(\psi)$ is a tautology.

Finally, if $\psi = \sim \neg\varphi$ is a tautology, then $S(\psi) = S(\sim \neg\varphi) = S(\neg\neg\varphi) = S(\varphi)$, but $\sim \neg\varphi \rightarrow \varphi$ is a tautology, then φ is tautology, hence by inductive hypothesis $S(\varphi)$ is a tautology.

5 Conclusions and Future Work

We introduce a family of paraconsistent logics extended by a strong paraconsistent negation operator. We study one particular logic, that we call N_5', and we show that it is sound with respect to N-GLukG. For future work one can consider the formal construction of non-monotonic semantics based on N_5'. This seems to be an easy task thanks to the experience of the construction of the answer set semantics based on N_5 logic.

References

1. A. Avron. *Natural 3-valued Logics, Characterization and Proof Theory.* Journal of Symbolic Logic 56, 276-294, 1991.
2. C. Baral. *Knowledge Representation, reasoning and declarative problem solving with Answer Sets.* Cambridge University Press, Cambridge, 2003.
3. M. Kracht. On extensions of intermediate logics by strong negation evaluation. *The Journal of Philosophical Logic*, 27(1):49–73, 1998.
4. J. Nieves, M. Osorio and U. Cortes. Preferred extensions as stable models. *Theory and Practice of Logic Programming*, 2008.
5. W. A. Carnielli and J. Marcos. A taxonomy of **C**-Systems. In *Paraconsistency: The Logical Way to the Inconsistent, Proceedings of the Second World Congress on Paraconsistency (WCP 2000)*, number 228 in Lecture Notes in Pure and Applied Mathematics, pages 1–94. Marcel Dekker, Inc., 2002.
6. N. da Costa. On the theory of inconsistent formal systems. *Notre Dame Journal of Formal Logic*, 15(4):497–510, 1974.
7. Minsky M. A framework for representing knowledge, The Psychology of computer vision. *Mcgrawn Hill*, 1975.

8. M. Gelfond and V. 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.
9. E.Mendelson. *Introduction to Mathematical Logic*. Wadsworth, Belmont, CA, third edition, 1987.
10. M. Osorio and J. L. Carballido. Brief study of G'$_3$ logic. *To appear in Journal of Applied Non-Classical Logic*, 18(4):79–103, 2009.
11. M. Osorio, J.Arrazola, and J. L. Carballido. Logical weak completions of paraconsistent logics. *Journal of Logic and Computation*, Published on line on May 9, 2008.
12. M. Osorio, J. A. Navarro, J. Arrazola, and V. Borja. Ground nonmonotonic modal logic S5: New results. *Journal of Logic and Computation*, 15(5):787–813, 2005.
13. M. Osorio, J. A. Navarro, J. Arrazola, and V. Borja. Logics with common weak completions. *Journal of Logic and Computation*, 16(6):867–890, 2006.
14. D. Pearce. Back and Forth Semantics for Normal, Disjunctive and Extended Logic programs. *Joint Conference on Declarative Programming, APPIA-GULP-PRODE*, 329-342, 1998.
15. D. Pearce. Stable Inference as Intuitionistic Validity. *Journal of Logic Programming*, 38:79–91, 1999.
16. D. Pearce. From here to there: Stable Negation in Logic Programming. *What is the negation?*, Kluver academic publishers, 161-181, 1998.
17. M.Ortiz and M. Osorio. Strong Negation and Equivalence in the Safe Belief Semantics. In *Journal of Logic and Computation* , 499 - 515, April, 2007.