AUTOMATED DEDUCTION WITH NON CLASSICAL NEGATIONS

Arthur Buchsbaum Tarcisio Pequeno
Laboratório de Inteligência Artificial
Universidade Federal do Ceará
Departamento de Computação
Campus do PICI / Bloco 910
60455–760 – Fortaleza/CE – Brasil
Tels: 55 85 2671513 – 55 85 2264419
Fax: 55 85 2231333
Emails: arthur@lia.ufc.br – tarcisio@lia.ufc.br

Abstract

Three basic alternatives to classical negation, represented by the paraconsistent calculus C_1, the paracomplete calculus P_1 and the non alethic calculus N_1, are treated. For each one of these calculi a tableau system is given. A general sketch of the proof of soundness and completeness of these tableaux against their correspondent calculi is provided.

1. Introduction

Two of the three basic principles taken by the founders of logic as self evident and adopted ever since for the construction of standard logical systems are related to negation: the *non contradiction law* and the *law of excluded middle*. This is a reason why most of non standard logics, even when they are not directly concerned with revising negation, are going to affect negation behavior by not endorsing some of its classical properties. This happens, for instance, to *relevant systems* with respect to non contradiction to the extent that contradictory theories in relevant logic may not be pathological and to admit models. It also happens to *intuitionistic logic* with respect to excluded middle.

More concerned with negation, there are several non standard logics whose explicit aim is exactly to question the universality and necessity of the above mentioned principles. This is done at least up to the point of demonstrating the feasibility of constructing non trivial logical systems relaxing those classical negation properties, in [5,8,7]. This kind of non classical logics can be generally divided into three families: the *paraconsistent logics*, which relax non contradiction, the *paracomplete logics*, relaxing excluded middle, and *non alethic logics*, relaxing both.

Our concern here is the study of automated procedures for these three types of logical systems. Since classical negation properties play a central and very special role for the conception of automated deduction methods, interesting questions are immediately raised. How refutational proof procedures are going to work for logics in which contradictory theories are no longer unsatisfiable? How tableau expansion rules should be changed in order to work for these logics? What are sound conditions for closure of tableau branches?

These questions are faced here in a very concrete fashion by the presentation of proof procedures for representatives of each one of those logical families. Those kind of logical systems have been intensely studied by da Costa and his collaborators.

We choose then as representative three of the most popular logics developed by his group: the *paraconsistent calculus* C_1, the *paracomplete calculus* P_1, and the *non alethic calculus* N_1. For the sake of generality and simplicity, tableau-like procedures have been adopted in this presentation.

2. Basic Elements of Tableau Systems

In order to generalize tableau method, making it applicable to non standard logics, its basic elements are identified in [1]. These elements are:

- *Initialization Function* — it sets up a formula derived from the proposed theorem on the root of the tableau, in order to prepare for the unsatisfiability test provided by tableau procedure. Frequently it is just the negation of a given formula, but it could be some other suitable transformation. This *initial tableau* starts the process of tableau development which proceed by recurrent application of *expansion rules*.
- *Expansion Rules* — these rules are the *core* of tableau method. They promote branching in the tableaux by performing analysis of the formulas occurring in them. This analysis leads to *irreducible forms* (or *literals*), formulas which are not in the domain of any expansion rule.
- *Closure Conditions* — these are special conditions that, when occurring in a branch of a tableau, cause the closure of this branch. The goal of a tableau procedure is exactly to close all the branches of the tableau, therefore proving the proposed theorem.

To define a particular tableau procedure is to define these three elements. All the logics to be treated here have in common the fact that classical negation is definable in them. This is not a special feature of this particular sample. It is a very widespread property (if not a universal one) of this kind of logics that classical negation is either definable in, or belonging to a conservative extension of them. To *define* classical negation is to express a connective which satisfies all theorems that negation does in classical logic. This includes the *axiom of absurd*, which is crucial to a refutational method. So, the initialization of the tableau will be always done by a *defined* negation of the proposed theorem. In spite of this apparent uniformity, it will be seen that this definition varies substantially from logic to logic.

Defined negation will also play a role for the *closure criteria*, but it does not solve entirely our problem. To deal with a non standard logic is to deal with a peculiar set of tautologies. So, the tableau expansion rules, the core of a system of tableaux, as we have said, must be rebuilt in accordance to this new set.

Non standard logics will also exhibit peculiar *irreducible forms*, i.e., peculiar sets of *literals*. When a tableau system is based on a recursive semantics, i.e., a semantics in which the truth value of any complex formula is entirely determined by the truth values of its atomic components, the closure conditions may be expressed in terms of literals. This is not the case for the logics treated here. No recursive semantics are known to date to these logics [6,8,7]. So, the closure conditions are expressed in terms of general formulas. Paraconsistent, paracomplete and non alethic logics with recursive semantics are given in [2].

3. The Paraconsistent Calculus C_1

A *paraconsistent logic* is a logic in which an inconsistent theory is not necessarily the set of all formulas in the language. In other words, a logic in which the presence of a contradiction does not *trivialize* a theory. Semantically, this means that in a paraconsistent logic an inconsistent theory could have a model. A trivial theory, on the other hand, will be an unsatisfiable one. Da Costa has sometimes, as in [5], translated this condition into a requirement stating that $\neg(A \wedge \neg A)$, which in a sense express *internally* the non contradiction law, should not be a logical theorem. This is not in fact a necessary condition for the attainment of paraconsistency, as it is attested in [2] and [9], but it is anyway respected by the family C_n of paraconsistent calculi. C_1 is the first member of this family. Its axiomatics is the following:

$A \to (B \to A)$;
$(A \to B) \to (A \to (B \to C)) \to (A \to C)$;
$A, A \to B \,/\, B$;
$A \wedge B \to A$;
$A \wedge B \to B$;
$A \to (B \to A \wedge B)$;
$A \to A \vee B$;
$B \to A \vee B$;
$(A \to C) \to (B \to C) \to (A \vee B \to C)$;
$A \vee \neg A$;
$\neg\neg A \to A$;
$B^o \to ((A \to B) \to ((A \to \neg B) \to \neg A))$;
$A^o \wedge B^o \to (A \to B)^o \wedge (A \wedge B)^o \wedge (A \vee B)^o$.

A° is shorthand for $\neg(A \wedge \neg A)$.

In C_1 the schemes $\neg(A \wedge \neg A)$ and $(A \wedge \neg A) \to B$ are not theorems.

Classical negation can be defined in C_1 by:

$$\sim A \equiv \neg A \wedge A^\circ$$

The connective "~", as defined above, holds all properties of classical negation in C_1.

The following formulas are among the theorems of C_1 that played an important role for the design of rules for C_1 tableaux:

$\vdash_{C_1} \neg A \leftrightarrow \sim A \vee \sim A^\circ$;
$\vdash_{C_1} A^\circ \leftrightarrow \sim A \vee \sim\neg A$;
$\vdash_{C_1} \neg\neg A \to A$;

$\vdash_{C_1} \sim\neg A \to A;$
$\vdash_{C_1} \neg\sim A \leftrightarrow A;$
$\vdash_{C_1} \neg A^{\circ} \leftrightarrow \sim A^{\circ}.$

We present below a tableau system for C_1 (this system has been presented before in [3]). The tableau is initialized by the classical negation (~) of the given formula.

The tableau expansion rules are the following (the symbol "#" is used to represent anyone of the connectives "$\to$", "$\wedge$", or "$\vee$"):

$$\frac{A \to B}{\sim A \mid B} \qquad \frac{A \wedge B}{\begin{array}{c} A \\ | \\ B \end{array}} \qquad \frac{A \vee B}{A \mid B} \qquad \frac{\sim(A \to B)}{\begin{array}{c} A \\ | \\ \sim B \end{array}} \qquad \frac{\sim(A \wedge B)}{\sim A \mid \sim B} \qquad \frac{\sim(A \vee B)}{\begin{array}{c} \sim A \\ | \\ \sim B \end{array}} \qquad \frac{\sim\sim A}{A}$$

$$\frac{\neg\neg A}{A} \qquad \frac{\sim\neg A}{A} \qquad \frac{\neg\sim A}{A} \qquad \frac{A^{\circ}}{\sim A \mid \sim\neg A} \qquad \frac{\sim(A^{\circ})}{\begin{array}{c} A \\ | \\ \neg A \end{array}} \qquad \frac{\neg(A^{\circ})}{\begin{array}{c} A \\ | \\ \neg A \end{array}} \qquad \frac{\neg(A \# B)}{\sim(A \# B) \mid \sim(A^{\circ}) \mid \sim(B^{\circ})}$$

The rules for $\sim\sim A$, $\neg\sim A$, $\sim(A^{\circ})$ or $\neg(A^{\circ})$ are given for the sake of computational convenience, but are not essential for completeness of this tableau system with respect to C_1. According to these rules, literal forms in C_1 are: p, ~p and $\neg$p, where p is an atomic formula.

A branch is closed in this tableau if A and ~A occurs in it. The presence of one of the following formulas in a branch is sufficient condition for its closure: $\sim((A \wedge \neg A)^{\circ})$, $\sim(A^{\circ\circ})$ or $\sim((\sim A)^{o})$. However, these additional criteria are not essential for completeness.

4. The Paracomplete Calculus P_1

A *paracomplete logic*, as defined in [8], is a logic in which two formulas, A and $\neg$A, can be both false. It fits our intuition about all kinds of epistemic attitudes such as knowledge, perception, belief, etc. We don't know many things, thence sometimes we can't affirm or deny a property of a given object that we don't perceive perfectly. The axiomatics of P_1 is the following:

$A \to (B \to A);$
$(A \to B) \to (A \to (B \to C)) \to (A \to C);$
$A, A \to B / B;$
$A \wedge B \to A;$
$A \wedge B \to B;$
$A \to (B \to A \wedge B);$
$A \to A \vee B;$
$B \to A \vee B;$
$(A \to C) \to (B \to C) \to (A \vee B \to C);$
$((A \to B) \to A) \to A;$
$A \to (\neg A \to B);$
$A \to \neg\neg A;$
$\neg(A \wedge \neg A);$
$A^{*} \to ((A \to B) \to ((A \to \neg B) \to \neg A));$
$A^{*} \wedge B^{*} \to (A \to B)^{*} \wedge (A \wedge B)^{*} \wedge (A \vee B)^{*}.$

A^* is shorthand for $A \vee \neg A$ in P_1. This schema is not a theorem in P_1.

Classical negation can be introduced in P_1 by the following definition:

$$\sim A \equiv A \rightarrow \neg A$$

The connective "~", as defined above, bears all properties of classical negation in P_1.

The following theorems are among the ones which played a fundamental role in the design of tableau rules for P_1:

$\vdash_{P_1} \neg A \leftrightarrow {\sim}A \wedge A^*$;

$\vdash_{P_1} {\sim}\neg A \leftrightarrow A \vee {\sim}A^*$;

$\vdash_{P_1} \neg{\sim}A \leftrightarrow A$;

$\vdash_{P_1} {\sim}\neg\neg A \rightarrow {\sim}A$.

The initialization will be given again by the use of classical negation. Expansion rules are the following:

$$\begin{array}{c} \underline{A \rightarrow B} \\ {\sim}A \mid B \end{array} \qquad \begin{array}{c} \underline{A \wedge B} \\ A \\ \vert \\ B \end{array} \qquad \begin{array}{c} \underline{A \vee B} \\ A \mid B \end{array} \qquad \begin{array}{c} \underline{{\sim}(A \rightarrow B)} \\ A \\ \vert \\ {\sim}B \end{array} \qquad \begin{array}{c} \underline{{\sim}(A \wedge B)} \\ {\sim}A \mid {\sim}B \end{array} \qquad \begin{array}{c} \underline{{\sim}(A \vee B)} \\ {\sim}A \\ \vert \\ {\sim}B \end{array}$$

$$\begin{array}{c} \underline{{\sim}{\sim}A} \\ A \end{array} \qquad \begin{array}{c} \underline{\neg{\sim}A} \\ A \end{array} \qquad \begin{array}{c} \underline{{\sim}\neg{\sim}A} \\ {\sim}A \end{array} \qquad \begin{array}{c} \underline{{\sim}\neg\neg A} \\ {\sim}A \end{array} \qquad \begin{array}{c} \underline{\neg A} \\ {\sim}A \end{array} \qquad \begin{array}{c} \underline{\qquad \neg(A \# B) \qquad} \\ A \# B \mid {\sim}(A^*) \mid {\sim}(B^*) \end{array}$$

The rules for ~~A, ¬~A or ~¬~A are not essential for completeness. The literals in this system are p, ~p and ~¬p, where p is an atomic formula.

A branch is closed in this tableau if it has two formulas of the forms A and ~A, or a formula of the form $\sim\neg(A \wedge \neg A)$ occurring in it.

5. The Non Alethic Calculus N_1

A *non alethic logic* is a logic combining paraconsistent with paracomplete features, by relaxing non contradiction and excluded middle at the same time. It has been defined in da Costa [7], as a logic in which two contradictory formulas, A and ¬A, can be both true, or both false.

The axiomatics of N_1 is the following:

$A \rightarrow (B \rightarrow A)$;

$(A \rightarrow B) \rightarrow (A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow C)$;

$A, A \rightarrow B / B$;

$A \wedge B \rightarrow A$;

$A \wedge B \rightarrow B$;

$A \rightarrow (B \rightarrow A \wedge B)$;

$A \rightarrow A \vee B$;

$B \rightarrow A \vee B$;

$(A \rightarrow C) \rightarrow (B \rightarrow C) \rightarrow (A \vee B \rightarrow C)$;

$((A \rightarrow B) \rightarrow A) \rightarrow A$;

$A^\circ \rightarrow (A \rightarrow \neg\neg A) \wedge (A \rightarrow (\neg A \rightarrow B))$;

$A^* \rightarrow (\neg\neg A \rightarrow A)$;

$A^* \wedge B^\circ \to ((A \to B) \to ((A \to \neg B) \to \neg A));$
$A^\circ \wedge B^\circ \to (A \to B)^\circ \wedge (A \wedge B)^\circ \wedge (A \vee B)^\circ \wedge (\neg A)^\circ;$
$A^* \wedge B^* \to (A \to B)^* \wedge (A \wedge B)^* \wedge (A \vee B)^* \wedge (\neg A)^*;$
$A^\circ \vee A^*.$

The shorthands for A° in C_1 and for A^* in P_1 are still adopted in N_1.

The schemes $\neg(A \wedge \neg A)$, $(A \wedge \neg A) \to B$ and $A \vee \neg A$ are not theorems in N_1.

The classical negation can be defined in N_1 by:

$${\sim}A \equiv A \to (\neg A \wedge A^\circ)$$

The connective "~", as defined above, has all the properties of classical negation at N_1.

The following theorems have been taken into account for the elaboration of rules for N_1 tableaux:

$\vdash_{N_1} \neg A \wedge A^\circ \to {\sim}A;$
$\vdash_{N_1} {\sim}\neg A \wedge A^* \to A;$
$\vdash_{N_1} \neg{\sim}A \leftrightarrow A;$
$\vdash_{N_1} A^\circ \leftrightarrow {\sim}A \vee {\sim}\neg A;$
$\vdash_{N_1} \neg A^\circ \leftrightarrow {\sim}A^\circ.$

Initialization is done again by classical negation. Expansion rules of the tableau for N_1 are:

$$\frac{A \to B}{{\sim}A \mid B} \qquad \frac{A \wedge B}{\begin{array}{c}A\\ \mid\\ B\end{array}} \qquad \frac{A \vee B}{A \mid B} \qquad \frac{{\sim}(A \to B)}{\begin{array}{c}A\\ \mid\\ {\sim}B\end{array}} \qquad \frac{{\sim}(A \wedge B)}{{\sim}A \mid {\sim}B} \qquad \frac{{\sim}(A \vee B)}{\begin{array}{c}{\sim}A\\ \mid\\ {\sim}B\end{array}} \qquad \frac{{\sim}{\sim}A}{A}$$

$$\frac{\neg\neg A}{{\sim}\neg A \mid A^\circ} \qquad \frac{\neg{\sim}A}{A} \qquad \frac{{\sim}\neg{\sim}A}{{\sim}A} \qquad \frac{{\sim}\neg\neg A}{\neg A \mid {\sim}A^*} \qquad \frac{A^\circ}{{\sim}A \mid {\sim}\neg A} \qquad \frac{{\sim}\neg(A^\circ)}{{\sim}A \mid {\sim}\neg A}$$

$$\frac{{\sim}(A^\circ)}{\begin{array}{c}A\\ \mid\\ \neg A\end{array}} \qquad \frac{\neg(A^\circ)}{\begin{array}{c}A\\ \mid\\ \neg A\end{array}} \qquad \frac{\neg(A \# B)}{{\sim}(A \# B) \mid {\sim}(A^\circ) \mid {\sim}(B^\circ)} \qquad \frac{\neg(A \# B)}{{\sim}(A \# B) \mid {\sim}(A^\circ) \mid {\sim}(B^\circ)}$$

The rules for ${\sim}{\sim}A$, $\neg{\sim}A$, ${\sim}\neg{\sim}A$, A°, $\neg(A^\circ)$ or ${\sim}\neg(A^\circ)$ are not essential for completeness. The literals in this system are p, ~p, ¬p, or ~¬p, where p is an atomic formula.

An occurrence of A and ~A in a branch is the condition for its closure.

6. Soundness and Completeness

A proof of the soundness and completeness of the given tableaux, against their correspondent calculi, is sketched bellow. It is presented in terms of a uniform routine capable to prove that a tableau system S_C is sound and complete with respect to ***C***, being ***C*** one of the calculi C_1, P_1 and N_1.

The concept of *trivial theory* will play a key role in the proof. A *trivial theory* is a theory which proves anything. It corresponds to the set of all well formed formulas in the language where it is expressed. Semantically, a trivial theory would be one which does not have a model. So, logical consequence becomes vacuously universal. For classical logic, inconsistent and trivial theories are synonymous, but this is not generally the case for the logics presented here. So, the concept of *trivialization* will play the rule that the concept of *inconsistency* plays for the classical case, both of them corresponding semantically to *unsatisfiability*.

The following lemma establishes that the rules in $S_{\boldsymbol{C}}$ are *conservative*, in the sense that their application preserves trivialization.

Trivialization Lemma: Let Φ be a set of wff's in $\boldsymbol{C}$, and A a formula of Φ. Let $\Delta_1,\ldots,\Delta_n$ be the sets of formulas in the branches resulting from the application of a rule of $S_{\boldsymbol{C}}$ to A. If $\Phi \cup \Delta_1,\ldots,\Phi \cup \Delta_n$ are all *trivial*, then so is Φ.

Proof:

For each i = 1,...,n, let $\sim\Delta_i$ be the disjunction of classical negations (as defined for each calculus) of the elements of Δ_i.Since $\Phi \cup \Delta_i$ is trivial,we have that

$$\Phi \vdash_{\boldsymbol{C}} \sim\Delta_i,$$

and then

$$\Phi \vdash_{\boldsymbol{C}} \bigwedge_{n}^{i=1} \sim\Delta_i$$

(this can easily be verified for the given calculi).

We have that

$$\bigwedge_{n}^{i=1} \sim\Delta_i \vdash_{\boldsymbol{C}} \sim A,$$

but then

$$\Phi \vdash_{\boldsymbol{C}} \sim A \text{ (by transitivity)},$$

and so $\Phi \cup A$ is trivial. Since $A \in \Phi$, it follows that Φ is trivial.

As an example, consider the rule for $\sim\neg(A \# B)$ in $S_{\mathbf{P_1}}$. If $\Phi \cup \{A \# B\}$, $\Phi \cup \{\sim A^*\}$ and $\Phi \cup \{\sim B^*\}$ are trivial, then $\Phi \vdash_{\mathbf{P_1}} \sim(A \# B)$, $\Phi \vdash_{\mathbf{P_1}} A^*$ and $\Phi \vdash_{\mathbf{P_1}} B^*$, and thence $\Phi \vdash_{\mathbf{P_1}} \sim(A \# B) \wedge A^* \wedge B^*$. But we have that $\sim(A \# B) \wedge A^* \wedge B^* \vdash_{\mathbf{P_1}} \neg(A \# B)$, therefore $\Phi \vdash_{\mathbf{P_1}} \neg(A \# B)$, and so $\Phi \cup \{\sim\neg(A \# B)\}$ is trivial. Finally, if $\sim\neg(A \# B) \in \Phi$, then Φ is trivial.

●

For the next theorem we need a suitable definition of the *degree* of a tableau, in order to perform induction on tableau development.

We say that $(T_i)_{i\in I}$ is a *tableau development sequence* in $S_{\boldsymbol{C}}$ if the following conditions are satisfied:

- $I = \mathbb{N}$ or $I = \{0,...,n\}$, for some $n \in \mathbb{N}$;
- T_0 is the initial tableau for some collection of formulas;
- for all $i \in I$ such that $i + 1 \in I$, T_{i+1} is obtained from T_i by one application of some rule of $S_{\boldsymbol{C}}$.

A natural number i_0 is said to be the *degree* of a tableau T' if there exists a tableau development sequence $(T_i)_{i\in I}$ such that $T' = T_{i_0}$. Observe that a tableau T' cannot have two distinct degrees, otherwise there would be two different quantities of marked nodes, which is an absurd.

Soundness Theorem: If there is a confutation for Φ, then Φ is trivial.
Proof:
Let n be the degree of the existing confutation for Φ. We will show, by induction on n, that Φ is trivial.
If $n = 0$, it means that the confutation has a single branch containing all the formulas of Φ. Since it is closed, the proof reduces to the verification that the given *closure conditions* are in fact *trivialization conditions*. In S_{P_1}, for instance, that means to prove that $A \wedge {\sim}A \vdash_{\mathbf{P_1}} B$ and that ${\sim}\neg(A \wedge \neg A) \vdash_{\mathbf{P_1}} B$, which is straightforward.
For an induction step, let $\mathcal{R}$ be the rule applied to obtain the tableau of degree 1 at the beginning of the development sequence generating the given confutation. Let $r_1,\ldots,r_p$ be the branches resulting from this application, A the formula to which $\mathcal{R}$ was applied, and finally $\Delta_1,\ldots,\Delta_p$ the collections of formulas in $r_1,\ldots,r_p$.
For each $i = 1,\ldots p$, a confutation for $\Phi \cup \Delta_i$, with degree less than n, can be obtained by incorporating the development following r_i, in the confutation for Φ, to the initial tableau for $\Phi \cup \Delta_i$. Thence, by the induction hipothesis, $\Phi \cup \Delta_i$ is trivial. Therefore, by the trivialization lemma, Φ is trivial.

●

Corollary (Soundness restated): If there is a confutation for $\Phi \cup \{{\sim}A\}$, then $\Phi \vdash_{\boldsymbol{C}} A$.
Proof:
If there is a confutation for $\Phi \cup \{{\sim}A\}$, then, by the soundness theorem, $\Phi \cup \{{\sim}A\}$ is trivial, therefore $\Phi \vdash_{\boldsymbol{C}} A$.

●

To prove the completeness of $S_{\boldsymbol{C}}$ some lemmas and definitions are in order.

We say that two formulas are *complementary* if the presence of both of them in a branch cause the branch to be closed. For reasonable tableau systems, as the ones presented here, if two formulas A and A' are complementary to a third one and there is a confutation to $\Phi \cup \{A\}$, then there is a confutation to $\Phi \cup \{A'\}$.

Complementarity Lemma: Let Φ be a set of wff's of $\boldsymbol{C}$, A a formula of Φ, and $\Delta_1,\ldots,\Delta_n$ the collections of formulas of the branches resulting from the applications of rules of $S_{\boldsymbol{C}}$ to A and to ~A, in a tableau development sequence. If there are confutations for $\Phi \cup \{A\}$ and for $\Phi \cup \{{\sim}A\}$, then, for all $i = 1,\ldots,n$, there exists a confutation for $\Phi \cup \{\Delta_i\}$.
Proof:
Let B be one of the formulas A or ~A. Without loosing generality, we can consider Δ_i coming from an application of a rule of $S_{\boldsymbol{C}}$ to B. Then a confutation T for $\Phi \cup \{\Delta_i\}$ can be obtained by imitating the branch of the confutation for $\Phi \cup \{B\}$, whose collection of formulas is Δ_i. The unique way to use B in T would be to close the branches in which a formula B', complementary of B, occurs. But, according to the observation done above, there is a confutation T' for $\Phi \cup \{B'\}$. Therefore, such branches can be closed the same way B' occurring branches in T' are closed.

●

Elimination Lemma: If there are confutations for $\Phi \cup \{A\}$ and for $\Phi \cup \{\sim A\}$, then there is a confutation for Φ.
Proof:
It can be proved by recursion on the structure of a formula A.
If A is a literal, a confutation T for Φ can be constructed by following the existing confutation for $\Phi \cup \{A\}$, except when A is employed. Since literals are irreducible forms (do not occur in the domain of any rule), the only way to use A in T would be to close branches in which ~A occurs. But, since a confutation T' for $\Phi \cup \{\sim A\}$ is also available, such branches can be closed the same way ~A occurring branches in T' are closed.
For the induction step, consider the application of a rule to a formula A. By the complementarity lemma, for all i = 1,...,n, there exists a confutation for $\Phi \cup \{\Delta_i\}$. By this last assertion, using the induction hipothesis, we can conclude that there is a confutation for Φ.
For example, consider S_{P_1}. Let A be of the form $\neg(B \# C)$. If there are confutations for $\Phi \cup \{\neg(B \# C)\}$ and for $\Phi \cup \{\sim\neg(B \# C)\}$, then, by the complementarity lemma, there are confutations for $\Phi \cup \{(B \# C)\}$ and for $\Phi \cup \{\sim(B \# C)\}$, thence, by induction hipothesis, there exists a confutation for Φ.

●

We are now able to prove a theorem stating that the employment of *excluded middle rules* (*em-rules*) could be eliminated from tableau developments. An *em-rule* is a function, applicable to all wff's, which maps any formula into a two branched tree with a branch containing a formula and the other its classical negation.

Elimination Theorem: If there is a confutation for Φ employing em-rules, then there is a confutation for Φ without using those rules.
Proof:
Let T be the existing confutation for Φ and k the highest level in which an em-rule occurs in T:

$$\begin{array}{ccc} & \vdots & \\ & A_k & \\ B & | & \sim B \\ \vdots & & \vdots \end{array}$$

Before this application, A_k was a leaf in a branch, say $\Psi = \{A_0,...,A_k\}$. Since this is the last point an em-rule is used, there are confutations for $\Psi \cup \{B\}$ and $\Psi \cup \{\sim B\}$, without the use of such rules, and, by the elimination lemma, there is a confutation for Ψ, which is em-rule free. This procedure could be repeated finitely many times, until an em-rule free confutation for Φ is obtained.

●

The central result can now be easily demonstrated.

Completeness Theorem: If $\Phi \vdash_{\boldsymbol{C}} A$, then there is a confutation for $\Phi \cup \{\sim A\}$.
Proof:
The proof is done by induction on theorems of Φ in ***C***.
For each logical axiom α of ***C***, we must verify that there is a confutation for $\Phi \cup \{\sim\alpha\}$.
Obviously, for each $\phi \in \Phi$ there is a confutation for $\Phi \cup \{\sim\phi\}$.

Finally the induction step through modus ponens. Assume that there are confutations for $\Phi \cup \{\sim B\}$ and for $\Phi \cup \{\sim(B \to C)\}$. A confutation for $\Phi \cup \{\sim C\}$ could be constructed in the following way:

$$\begin{array}{ccc} & \Phi \cup \{\sim C\} & \\ B \to C & & \sim(B \to C) \\ \begin{array}{cc} \sim B & \underline{C} \\ \vdots & \end{array} & & \vdots \end{array}$$

Since we have confutations for $\Phi \cup \{\sim B\}$ and for $\Phi \cup \{\sim(B \to C)\}$, the first and third branches will eventually be closed. The branch in the middle is closed by $\sim C$, located in the initial tableau.

By the elimination lemma, the em-rule employed for $B \to C$ and $\sim(B \to C)$ can be eliminated. So, there is a confutation, without em-rules, for $\Phi \cup \{\sim C\}$.

●

7. Commentaries and Conclusions

The logics studied here present three basic alternatives to negation. C_1 has a *paraconsistent negation*, a negation weaker than classical one and such that an inconsistent theory, in terms of this negation, is not necessarily the set of all wff's. In other words, in C_1 a formula A and its negation could be simultaneously the case. The logic P_1 is a kind of *dual* of C_1 by introducing a *paracomplete negation*, a negation such that A and $\neg A$ may simultaneously not be the case. Finally, the *non alethic negation* of N_1 accumulate both non classical features of paraconsistency and paracompleteness.

The logics we choose here to illustrate how deviant negations may affect methods of reasoning by tableau are a sort of *first generation alternative negation calculi*. They are among the most traditional and representative calculi developed by da Costa to express his ideas about alternative negations.

These three logics can be extended by definition to include a negation which behaves classically. The tableaux for them have in fact been written in these extensions. So, in all of them we have been allowed to use classical negation to initialize the test for unsatisfiability provided by the method of tableaux, and to express conditions for the closure of branches. For the logics treated here these conditions are the same of classical tableau, with the exception of P_1, for which an additional condition is required for completeness. Supplementary conditions can be added for the two other calculi, but just for the sake of computational convenience. This simplicity of closure conditions is not always the case for logics featuring non classical negation. We designed a paraconsistent logic called LEI to formalize the notion of *epistemic inconsistency* [9]. Although classical negation is definable in LEI, its tableau, given in [4], is far more complex then the ones given here, inclusive with respect to closure conditions.

The logics presented here also have in common the fact that there are no recursive semantics available to them. We have developed a collection of logics which we call respectively LI (for *logic for inconsistency*), PCL (for *paracomplete logic*) and NALL (for *non alethic logic*). These are a sort of *second generation calculi*. Although being respectively paraconsistent, paracomplete and non alethic, they are, in a sense, stronger than da Costa's ones. Recursive semantics have been provided for these calculi in [2]. Based on these semantics, tableaux can be given for them. In those tableaux, closure conditions may be completely expressed in terms of literals.

In order to enhance the credibility of the solutions given here, sketches of the soundness and completeness proofs have been added. For the sake of space and simplicity we aimed to present them in a general form applicable, under minor adaptations, to the three given calculi.

References

[1] Buchsbaum, Arthur & Pequeno, Tarcisio, *O Método dos Tableaux Generalizado e sua Aplicação ao Raciocínio Automático em Lógicas Não Clássicas*, “O que nos faz pensar” – nº 3, pp. 81-96, September 1990.

[2] Buchsbaum, Arthur & Pequeno, Tarcisio, *Uma Família de Lógicas Paraconsistentes e/ou Paracompletas com Semânticas Recursivas*, Monografias em Ciência da Computação nº 5/91, Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, 1991.

[3] Buchsbaum, Arthur & Pequeno, Tarcisio, *A Reasoning Method for a Paraconsistent Logic*, Studia Logica 52/2, May 1993.

[4] Correa, Marcelo & Buchsbaum, Arthur & Pequeno, Tarcisio, *Sensible Inconsistent Reasoning: A Tableau System for LEI*, Technical Notes of AAAI Fall Symposium on Automated Deduction in Non-Standard Logics, October 1993.

[5] da Costa, Newton C. A., *On the Theory of Inconsistent Formal Systems*, Notre Dame Journal of Formal Logic 15, pp. 497-510, 1974.

[6] da Costa, Newton C. A., *A Semantical Analysis of the Calculi* C_n, Notre Dame Journal of Formal Logic 18, pp. 621-630, 1977.

[7] da Costa, Newton C. A., *Logics that are both Paraconsistent and Paracomplete*, Rendiconti dell'Accademia Nazionale dei Linzei, vol. 83, pp. 29-32, 1989.

[8] da Costa, Newton C. A. & Marconi, Diego, *A Note on Paracomplete Logic*, Rendiconti dell'Accademia Nazionale dei Linzei, vol. 80, pp. 504-509, 1986.

[9] Pequeno, Tarcisio & Buchsbaum, Arthur, *The Logic of Epistemic Inconsistency*, Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning, Morgan Kaufmann, pp. 453-460, 1991.