filosofia unisinos | Unisinos Journal of Philosophy

ISSN 1984-8234

Filosofia Unisinos
Unisinos Journal of Philosophy
23(1): 1-10, 2022 | e23105

Unisinos – doi: 10.4013/fsu.2022.231.05

Article

Tableau method of proof for Peirce's three-valued propositional logic[1]

Método de prova por tablôs para a lógica proposicional trivalorativa de Peirce

José Renato Salatiel
https://orcid.org/0000-0001-5858-1248
Universidade Federal do Espírito Santo, Programa de Pós-Graduação em Filosofia, Vitória/ES, Brasil. Email: jrsalatiel@hotmail.com.

ABSTRACT

Peirce's triadic logic has been under discussion since its discovery in the 1960s by Fisch and Turquette. The experiments with matrices of three-valued logic are recorded in a few pages of unpublished manuscripts dated 1909, a decade before similar systems have been developed by logicians. The purposes of Peirce's work on such logic, as well as semantical aspects of his system, are disputable. In the most extensive work about it, Turquette suggested that the matrices are related in dual pairs of axiomatic Hilbert-style systems. In this paper, we present a simple tableau proof for a fragment of Peirce three-valued logic, called $\mathbf{P}_{3'}$ based on similar approaches in many-valued literature. We demonstrated that this proof is sound and complete. Besides that, taking the false as the only undesignated value and adding non-classical negations to the calculus, we can explore paraconsistent and paracompleteness theories into $\mathbf{P}_3$.

Keywords: Charles S. Peirce, many-valued logics, theory of proof, tableau method.

[1] This paper presents partial results of Postdoctoral research in Philosophy at the University of São Paulo (FFLCH-USP), named "Studies in Charles S. Peirce's three-valued logic" (2021-2022).

RESUMO

A lógica triádica de Peirce tem sido motivo de debates desde sua descoberta nos anos 1960, por Fisch e Turquette. Os experimentos com matrizes de lógicas trivalentes são registrados em poucas páginas de manuscritos inéditos datados de 1909, uma década antes de sistemas similares terem sido descobertos por lógicos. Os motivos do trabalho de Peirce nessa lógica, assim como aspectos semânticos de seu sistema, são discutíveis. No mais extensivo trabalho a respeito, Turquette sugeriu que as matrizes estão relacionadas em pares duais de sistemas axiomáticos de estilo hilbertiano. Neste artigo apresentamos uma prova simples por tablôs para um fragmento da lógica trivalente de Peirce, chamada $\mathbf{P}_3$, baseada em abordagens similares na literatura sobre lógicas trivalorativas. Demonstramos que tal prova é consistente e completa. Além disso, tomando o falso como único valor não-designado e adicionando negações não-clássicas, podemos explorar teorias de paraconsistência e paracompletude em $\mathbf{P}_3$.

Palavras-chave: Charles S. Peirce, lógicas trivalorativas, teoria da prova, método de tablôs.

Introduction

A few pages of Charles Peirce's unpublished manuscripts, dated from January to February 1909, is the subject of a lasting controversy among researchers of his work. The document contains what is now recognized as one of the first systems of many-valued propositional calculus, which precedes similar logics later discovered by Łukasiewicz, Post, Kleene, and Bochvar (Peirce, 1909).

In these notes, Peirce generalized the matrix method for classical logic, discovery by himself years ago, to a system with three truth-values: *verum* (V), *falsum* (F), and the "limit" (L), which means not determinately true or false. He then defined two groups of unary and binary operators, which seem to be negations, conjunctions, and disjunctions in modern logic.

But there is no textual evidence in these pages about the notions of logical consequence or material implication. Besides that, Peirce left no detail about the motivation of these experiments in his lifetime.

Three pages of Peirce's "Logic Notebook" were published by Fisch and Turquette in 1966. Turquette, in the next two decades, published articles in which triadic logic is regarded as axiomatic Hilbert-style systems related by dualistic relations between the connectives (1976, 1978, and 1981). However, besides some philosophical discussion suggested by Lane (1999) and, recently, Odland (2021), almost no work dealt with the problems of Peirce matrices since Turquette.

In Salatiel (2021), we proposed that Peirce's matrices for three-valued logic can be read, in fact, as logical devices for three different systems. Two of them were later developed by contemporary literature, but one of them, from this point called $\mathbf{P}_3$, is an original three-valued calculus and similar to paraconsistent theories. These conclusions are closely related to Belikov's (2021) recent article, which presented proof in natural deduction for Peirce's triadic logic, with no material implication operator.

In this paper, we aim to go a step further and propose a simple and generic tableau method of proof for Peirce $\mathbf{P}_3$-three-valued propositional logic, which is demonstrated to be sound and complete. This approach is based on the techniques developed by Carnielli (1987) and Hähnle (1990). To conclude, we address a brief note on paraconsistent and paracomplete views of Peirce's triadic logic.

1. Syntax and Semantic of System P_3

The $\mathbf{P_3}$ calculus is a three-valued generalization of the two-valued classic propositional logic. This logic is generated taken the set {Φ, Ψ, -} as primitive logical connectives. The symbols Φ and Ψ correspond respectively to binary operators for disjunction and conjunction. The bar connective (-) is the same as Łukasiewicz's three-valued negation.[2]

The remaining matrices that we found in Peirce's notebook (Peirce, 1909, seq. 640 v.), with two other types of disjunction and conjunction, gives rise to other well-known three-valued logics known as Łukasiewicz ($\mathbf{L_3}$)/Kleene (strong $\mathbf{K_3}$) and Bochvar (internal $\mathbf{B_3}$)/Kleene (weak $\mathbf{K_3}$) (Salatiel, 2021). Because of this, we focus on $\mathbf{P_3}$, whose main characteristics is that formula takes the truth-value L (limit) only when the components of the formula take L as well. In this section, we presented the syntax and semantic of $\mathbf{P_3}$ in a standard way (Burris; Sankappanavar, 1981; Wójcicki, 1988).

Definition 2.1: Let $\mathcal{L}$ be a propositional language defined as an abstract algebra $\mathcal{L}$ = <*For*, f_1, f_2, …, f_m>, where *For* stands for a set of formulas of $\mathcal{L}$ and {f_1, f_2, …., f_m} is a finite non-empty set of n-arity operations on *For*, called connectives. In the following, we will use the symbols ¬, ∨, ∧, and → for, respectively, negation, disjunction, conjunction, and implication. The Greek letters, such as α, β, and γ, will denote a scheme of formulas, while the uppercase Greek letters stand for a set of formulas. Now let *Var* = {P, Q, R…} be a non-empty set of symbols called propositional variables (or atomic formulas), such that *Var* ⊆ *For*.

Then we suppose an algebra $\mathcal{A}$ of the same type of $\mathcal{L}$ and *Val* is a non-empty set of truth-values {1, ½, 0}, that we will use instead of V, L and F. Considering that each f_i denotes a semantic correlate of $\mathcal{L}$, such that $\mathcal{A}$= <*Val*, f_{1A}, f_{2A}, f_{3A} … f_{nA}>, we define the homomorphism and a logical matrix.

Definition 2.2: Suppose that $\mathcal{A}$ is a subalgebra of $\mathcal{L}$, such that $\mathcal{A} \subseteq \mathcal{L}$. A mapping *h:* $\mathcal{L} \longrightarrow \mathcal{A}$ is called a homomorphism from $\mathcal{A}$ into $\mathcal{B}$, such that, for i = 1, …, m, if the following holds: $h(f_{i\,\mathcal{L}}(\alpha_1, \ldots, \alpha_n)) = f_{i\,\mathcal{A}}(h(\alpha_1), \ldots, h(\alpha_n))$, for $f \in \mathcal{L}$.

Definition 2.3: We called a logical matrix for $\mathcal{L}$ the pair $\mathcal{M}$ = <$\mathcal{A}$, $\mathcal{D}$>, where $\mathcal{D}$ is a non-empty proper set of *Val* which denotes the designated truth-values of $\mathcal{M}$. Now we can define a $\mathcal{M}$-valuation, i.e., a valuation with respect to $\mathcal{M}$, is a function

$$v(\mathcal{M}): For \longrightarrow Val$$

extended inductively to all functions of *For* as follows:

1. $v(\neg\alpha) = f_{\neg}(v(\alpha))$;
2. $v(\alpha \vee \beta) = f_{\vee}(v(\alpha), v(\beta))$;
3. $v(\alpha \wedge \beta) = f_{\wedge}(v(\alpha), v(\beta))$;
4. $v(\alpha \rightarrow \beta) = f_{\rightarrow}(v(\alpha), v(\beta))$.

Definition 2.4: Let Γ be a set of formulas, such that Γ ⊆ *For*, and α be a formula, such that α ∈ Γ. Given a matrix $\mathcal{M}$, a propositional $\mathcal{M}$-valuation of $\mathcal{L}$ is *model* of the formula α, denoted by the expression $v \vDash_{\mathcal{M}} \alpha$ (which means that α is $\mathcal{M}$-*satisfiable*), if $v(\alpha) \in \mathcal{D}$, for all α ∈ Γ.

Definition 2.5: Let $\mathcal{M}$ be a matrix, then α is a $\mathcal{M}$-consequence of Γ, $\Gamma \vDash_{\mathcal{M}} \alpha$ if every model of Γ is a model of α too.

Definition 2.6: $\mathbf{P_3}$ is induced by a logical matrix defined as an ordered triple:

$$\mathcal{M}\mathbf{P_3} = <\{1, ½, 0\}, \neg, \vee, \wedge, \rightarrow, \{1, ½\}>$$

whose connectives ¬, ∨ and ∧ can be defined just as the classical logic:

1. $v(\neg i) = 1 - v(i)$;
2. $v(i \vee j) = \max\{v(i), v(j)\}$;

[2] In the following, we will use standard notation, instead of symbols introduced by Peirce.

3. $v\ (i \wedge j) = \min \{v\ (i), v\ (j)\}$.

The set of connectives {¬, ∨} or {¬, ∧} can be taken as primitives and → is defined as (¬α ∨ β) or ¬(α ∧ ¬β). Based on the fact that every function of the system can be defined by the primitive set of connectives, it is easy to see that $\mathbf{P}_3$ is functionally complete. These definitions resulting in the following matrices:

	¬
1	0
½	½
0	1

∧	1	½	0
1	1	1	0
½	1	½	0
0	0	0	0

∨	1	½	0
1	1	1	1
½	1	½	0
0	1	0	0

→	1	½	0
1	1	0	0
½	1	½	0
0	1	1	1

FIGURE 1: truth-tables for $\mathbf{P}_3$ connectives.

However, there is no agreement about the definition of the designated truth-values of Peirce's triadic logic. Turquette (1981) considers that $\mathcal{D}$ = {1, ½} is most suitable to calculus, in this case, a proof in Hilbert-axiomatic type. Recently, Belikov (2021) pointed that such a choice contradicts the truth-value "gap" approach that Peirce expresses in his manuscript, which says that the truth-value L is not determinate truth nor determinate false (Peirce, 1909, seq. 645). But the choice for $\mathcal{D}$ = {1} turn L false anyway.

We think, otherwise, that making the singleton {0} the only non-designated truth-value is the best way to prove some theorems of the classical logic. Besides that, that composition of the set $\mathcal{D}$ = {1, ½} is close to Peirce's intention, when he said that triadic logic "does not conflict with Dyadic Logic" (Peirce, seq. 645).

2. Tableau for P_3

The method of analytic tableau for classical logic was first generalized to many-valued logics by Carnielli (1987), based on the first approach by Surma (1977). Another improvement was made by Hähnle (1990, 1999), which use a truth-value set of signs to make the tableau rules less redundant.

In this paper, we proceed with the method and basic notions, such as branch, node, and signed formulas, as defined for propositional logic by Smullyan (2009) and Fitting (1996). For basic proof of soundness and completeness, we will follow the steps suggested by Howson (1997). In the case of three-valued logic, the introduction of a third value ½ between true and false will demand a change in notions of close and open branches.

Definition 3.1: Let $\mathcal{L}_3$ be a propositional language of $\mathbf{P}_3$, and *Val* and $\mathcal{D}$ symbols as defined above. The set of signs can be defined as $\mathbf{S} = \{S_0, S_{½}, S_1\}$, which we will replace by the letters F, L, and V. Then, considering X and Y schemes of formulas and Si as a sign, the string of symbols S_i (X) and S_i (Y) are called signed formulas of $\mathcal{L}_3$.

Definition 3.2: Let X be a formula $F(X_1, \ldots, X_m)$, where F is an n-ary connective. A tableau rule is a function π_{iF} which assigns a tree to a signed formula $Si(X) \in \mathcal{L}$. This tree has in its first node, represented by the symbol "°", a formula $S_1\ (F\ (X_1, \ldots, X_m))$, that we call premise, and in the next nodes, the formulas $\{S_{ji}\ (X_{ij})\ _{\circ}, \ldots, {}_{\circ}\ S_{jt}\ (X_{it})\}$ is called the consequence of the premise.

Then a $\mathbf{P}_3$ propositional tableau for a formula $S_i\ (F(X_1, \ldots, X_m))$ is a tree that has in its first node this very formula, and whose other nodes are generated by the application of the function π_{iF}. The rules for $\mathbf{P}_3$ will be the following:

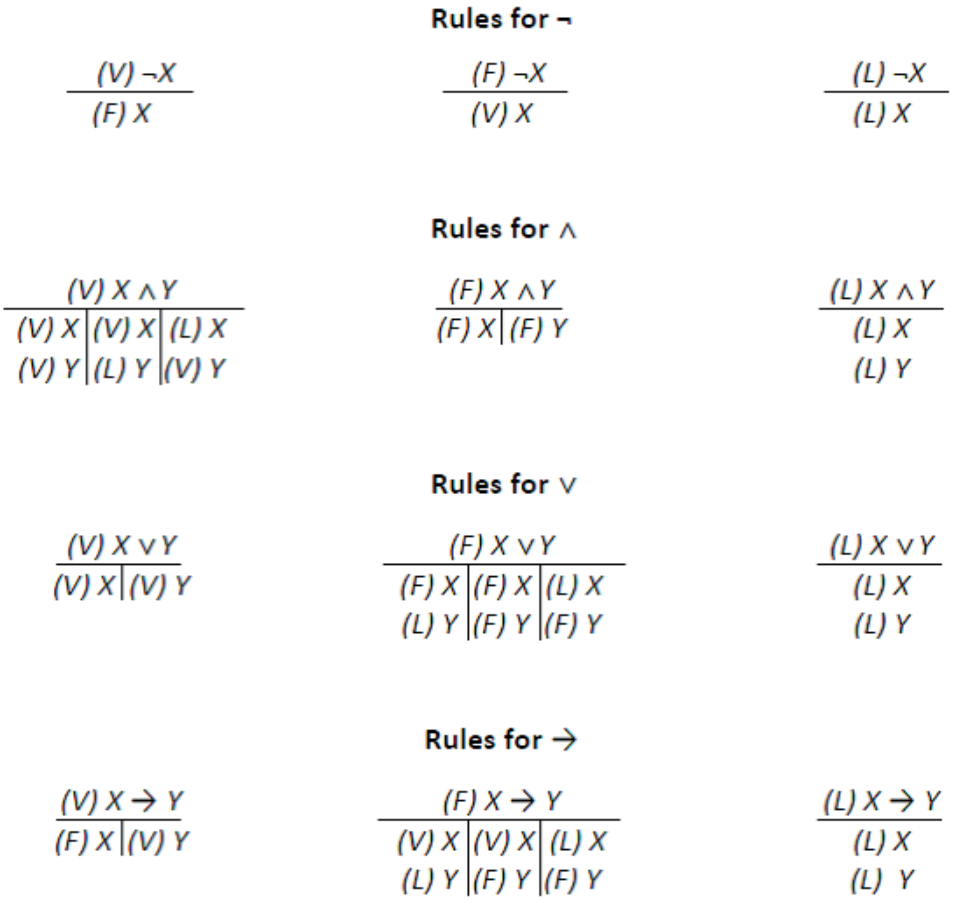

FIGURE 2: Tableau rules for the $\mathbf{P}_3$ system.

These rules can be brought together in a uniform notation, following the suggestion of Smullyan (2009) into three groups. The first one, represented by the formula α has just one branch, as a result of the application of the rules for ∨, ∧, and → to signed formulas. The other two, represented by the formulas β and γ, have respectively two and three branches as a result of the application of the rules. We have as a result the following diagrams:

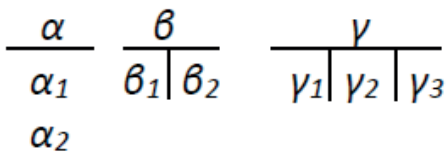

FIGURE 3: Diagrams of tableau rules for $\mathbf{P}_3$.

Definition 3.3: A tableau branch **B** is called closed if these conditions hold: (i) there is a node that contains both formulas S_i (X) and S_j (X) for $i \neq j$; or (ii) the branch contains a non-atomic formula X, such that F (X) ∈ **B** and F (¬X) ∈ **B**; otherwise the tableau is called open.

Let's see some examples of tableau proof for theorems of $\mathbf{P}_3$. We use the symbol "*" to indicate a close branch. First, we prove that (P → Q) → (¬P ∨ Q) is a theorem of $\mathbf{P}_3$, showing that there is a closed tableau with (F) (P → Q) → (¬P ∨ Q) in the root.

Example 3.1: (P → Q) → (¬P ∨ Q)

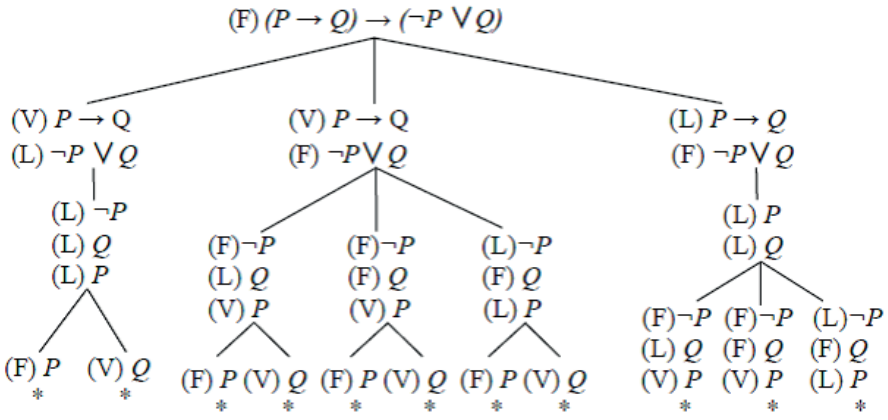

FIGURE 4: Tableau for (P → Q) → (¬P ∨ Q).

Next, we prove that the formula (P → (Q → P) is not a theorem of $\mathbf{P}_3$, as shown by the tree with an open branch.

Example 3.2: (P → (Q → P)

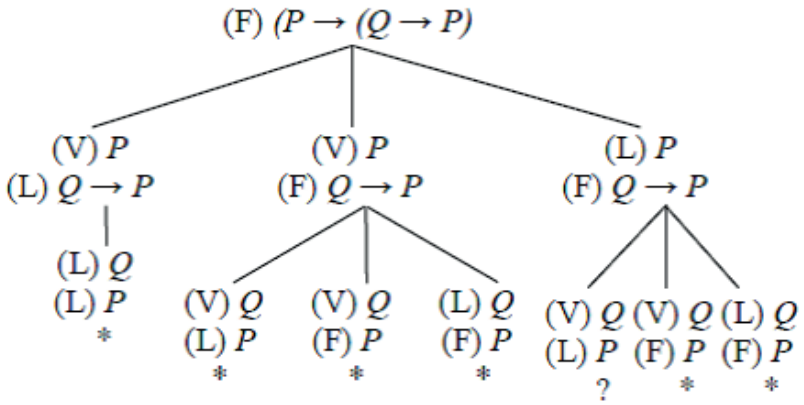

FIGURE 5: Tableau for (P → (Q → P).

Notice that the symbol "?" indicates that the branch is open. Immediately, the same open branch gives us a counter model of the formula: *v*(P)= L and *v*(Q)= V.

3. Soundness and completeness proofs

The proof of the soundness and completeness for the $\mathbf{P}_3$ system will be presented in this section, based on Howson (1997).

Lemma 4.1: Let B be a branch on a tableau $\mathcal{T}$. Then, if all the formulas on B are true under *v*, i.e., if α ∈ **B** then *v*(α) ∈ $\mathcal{D}$, then B is open. In this case, the formulas are said to be *satisfiable*.

Proof: Suppose **B** is a closed branch on a tableau $\mathcal{T}$. Then it would contain a formula α and its negation ¬ α**,** such that it is not true under an interpretation υ, i.e., α would be not satisfiable.

Theorem 4.1 (Soundness): Let Γ be a finite set of formulae in $\mathbf{P}_3$. If Γ is satisfiable by an interpretation υ, then a finished and totally developed tableau $\mathcal{T}$, generated by the application of the rules on

Γ, will display at least an open branch B. In other words, the tableau method for $\mathbf{P}_3$ is soundness in the sense that no formula and his negation are both provable in such a system (case in which we would have a closed branch).

Proof: Suppose a tableau $\mathcal{T}$ generated from Γ such that all the formulas in Γ are true under an interpretation υ. Then, by Lemma 1, we will be able to find an open branch **B** in $\mathcal{T}$. There are two cases:

(i) if **B** contains at least a literal (α or ~ α), then **B** is a finished open branch, such that $v(\alpha) \in \mathcal{D}$, for $\alpha \subseteq \Gamma$;

(ii) if **B** contains a non-literal, it has an extension **B'** generated by applying the rules for binary connectives. For example, suppose we have (V)X ∧ Y. Immediately we add to the branch (V)X and (V)Y; (L)X and (V)Y; and (V)X and (L)Y. Either way, the formula will be true under *v*. Now suppose (F)X ∧ Y. Then we have (F)X or (F)Y, and the formula will be true under an interpretation *v* too. In either case, (i) or (ii), the formulae on **B'** are true under *v*. Therefore, by the Lemma 1, the branch is open and the formulas are satisfiable. On the other hand, if **B'** is not a finished branch, then will be other branches as his extension in the tableau, **B'**, **B''**, **B'''**, etc. However, the tableau will be finished in an open branch at some point, as stated by our hypothesis.

For the next lemmas, we will associate to each connective a *weight*, represented by an integer, in the following way: the weight of a unary connective is 1 and a weight of a binary connective is 2. In this way, a degree of a formula X is the sum of the weights of his connectives. For example, the degree of the formula (P ∨ ¬P) is 3.

Lemma 4.2: For any formula X, the degrees of the formulas $\{\alpha_1, \alpha_2\}$, $\{\beta_1, \beta_2\}$, and $\{\gamma_1, \gamma_2, \gamma_3\}$ are, each of them, less than the degree of X itself. It is easy to check, by noticing that the application of the three rules for the schemas α, β, and γ always eliminate a binary connective of the tree.

Lemma 4.3: Let Γ be any set of formulas of $\mathbf{P}_3$ and k and integer. Suppose that: (i) all formulas of degree ≤ *k* in Γ have a propriety **P**; (ii) X is a formula of degree > *k*, then, if all formulas of lower degree have **P**, X has P too. Therefore, all the formulas of Γ have **P**.

Proof: Suppose that (i) holds. Then, all the formulas of degree ≤ *k* in Γ have **P**. Let *k'* be the smallest number greater than k, such that there is a formula of degree *k'* in Γ. Now we go to the next level, applying the same method to the formulas of Γ, to conclude that any formula of degree has **P**. Therefore, we demonstrate that (ii), all the formulas of Γ have **P**.

Theorem 4.2 (Completeness): Let Γ be a finite set of formulas of $\mathbf{P}_3$. If $\mathcal{T}$ is a finished open tree, in the sense that it contains at least one open branch, then the origin of Γ is satisfiable.

Proof: Suppose that Γ has a finished open tree and **B** is an open branch. Then we show that all the formulas on **B** are satisfiable under a valuation *v*.

Step 1: First, assume that literals are the only formulas that have degrees 0 or 1. Then, since ***b*** is an open branch, there is at least one literal on **B**. So the lowest degree of the formulas of Γ is 0 or 1. By supposition, all formulas on degree 0 or 1 on **B** are literals and are satisfiable under an interpretation *v*.

Step 2: Let X be any formula on **B** of degree >1. We assume, by hypothesis, that all formulas on b of degree > 1 in satisfiable under *v*, then X is satisfiable under v. There are three cases:

Case 1: X is a formula α. Then, by Lemma 4.1, both α_1 and α_2 are lower degrees than X. Therefore, by hypothesis, both are satisfiable under *v*, and, by theorem, so does X. The same goes for **Case 2** (X is a formula β) and **Case 3** (X is a formula γ). Hence, we proved that all formulas on Γ are satisfiable under *v*.

4. Paraconsistent and paracomplete theories

In this section, we address a brief note on paraconsistent and paracomplete approaches to $\mathbf{P}_3$ calculus. A logic is paraconsistent if the Principle of Explosion, in general, does not hold. In other words, for any formula α and β, is true that $\alpha \wedge \neg \alpha \nvdash \beta$. Dual to paraconsistent, a logic is paracomplete if the Principle of Excluded Middle generally does not hold. That is, is true that $\beta \nvdash \alpha \vee \neg\alpha$ (see Carnielli; Coniglio, 2016).

Then, we can prove theorems of both logics, paraconsistent and paracomplete, if we add to $\mathbf{P}_3$ matrix the following unary operators:

	∘
1	1
½	0
0	1

	~
1	0
½	0
0	1

FIGURE 6: Truth-tables for ∘ and ~.

The operator of consistency "∘" and the strong (intuitionistic) negation "~" can be defined, respectively, as

$\circ \coloneqq (\alpha \rightarrow \bot) \vee (\neg\alpha \rightarrow \bot)$

$\sim \coloneqq (\alpha \rightarrow \bot)$

Finally, we add the following tableau rules, where "*" indicate a tableau closure:

Rules for ∘

$$\frac{(V)\ \circ X}{(V)\ X \mid (F)\ X} \qquad \frac{(F)\ \circ X}{(L)\ X} \qquad \frac{(L)\ \circ X}{*}$$

Rules for ~

$$\frac{(V)\ {\sim}X}{(F)\ X} \qquad \frac{(F)\ {\sim}X}{(V)\ X \mid (L)\ Y} \qquad \frac{(L)\ {\sim}X}{*}$$

FIGURE 7: Tableau rules for ∘ and ~.

It is easy now to prove theorems of paraconsistent and paracomplete logics in $\mathbf{P}_3$ propositional calculus presented in this paper. As we noted early (Salatiel, 2021), the addition of the operator of consistency to the Peirce matrix corresponds to a three-value paraconsistent logic, later called **Ciore,** presented by Carnielli *et all* (2000).

Example 5.1: (∘P ∧ ∘Q) → ∘(P ∨ Q)

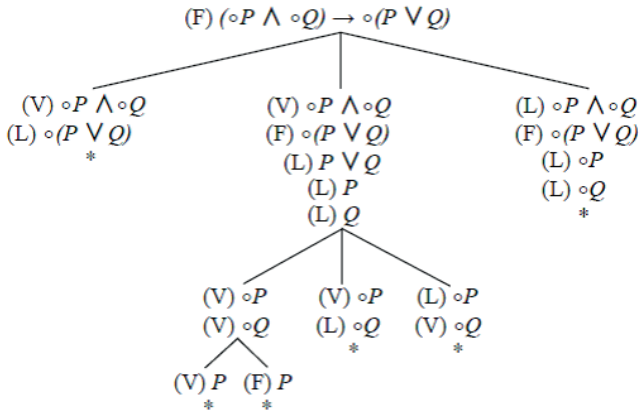

FIGURE 8: Tableau for (∘P ∧ ∘Q) → ∘(P ∨ Q).

Conclusion

In this paper, we suggested a tableau method for three-valued propositional logic based on the Peirce set of operators {Φ, Ψ, -}, which we called the $\mathbf{P}_3$ system. The tableau procedure was based on early works on many-valued systems by Carnielli (1987) and Hähnle (1990, 1999). We were able to obtain an efficient calculus to prove several theorems of classical logic, which is sound and complete.

It is difficult to understand Peirce's motivation from his unpublished manuscripts. What we proposed here is a simple way to interpret his matrices with a third truth-value L. In this path, we have to take a side, mainly concerning semantic aspects of $\mathbf{P}_3$. Peirce's choice about designated values, for example, it's not clear. Nevertheless, based on his intention to make his triadic logic as close as possible to classical propositional calculus, it would be more interesting to take false as the only undesignated value. Another important choice is about the use of material implication that reflects a functionally complete system.

Finally, we showed that this $\mathbf{P}_3$ system can be turned into both paraconsistent and paracomplete logics, simply by adding two unary operators: consistency and intuitionistic negations. Actually, Peirce's paraconsistent three-valued logic is akin to some systems from the family of the Logics of Formal Inconsistency.

References

BELIKOV, A. 2021. Peirce's triadic logic and its (overlooked) connexive expansion. *Logic and Logical Philosophy*, (online first articles). Available at: https://apcz.umk.pl/czasopisma/index.php/LLP/article/view/LLP.2021.007. Accessed 16 June 2021.

BURRIS, S.; SANKAPPANAVAR, H. P. 1981. *A Course in Universal Algebra*. New York, Springer-Verlag.

CARNIELLI, W. 1987.Systematization of finite many-valued logics through the method of tableaux. *The Journal of Symbolic Logic*, **52**(2): 473-493.

CARNIELLI, W.; CONIGLIO, M. E. 2016. *Paraconsistent Logic*: consistent, contradiction and negation. Dordrecht, Springer.

CARNIELLI, W.; MARCOS, J.; DE AMO, S. 2000. Formal inconsistency and evolutionary databases. *Logic and logical philosophy*, **8**(8): 115-152.

FISCH, M.; TURQUETTE, A. 1966. Peirce's triadic logic. *Transactions of the Charles S. Peirce Society*, **2**(2): 71-85.

FITTING, M. 1996. *First-Order and Automated Theorem Proving*. 2 ed. New York, Springer-Verlag.

HÄHNLE, R. 1990. Towards an efficient tableau proof procedure for multiple-valued logics. In *Proceedings Workshop Computer Science Logic*. Heidelberg, Springer, p. 248-260.

HÄHNLE, R. 1999. Tableaux for many-valued logic. In: M. D'AGOSTINO et al. (Eds.) *Handbook of Tableau Methods*. Netherlands, Springer, pp. 529-580.

HOWSON, C. 1997. *Logic with Trees*: an introduction to symbolic logic. London and New York, Routledge.

LANE, R. 1999. Peirce's triadic logic revisited. *Transactions of the Charles S. Peirce Society*, **XXXV**(2): 284-311.

ODLAND, B. C. 2021. Peirce's triadic logic: continuity, modality, and L. Unpublished master's thesis. University of Calgary, Calgary, AB. Available at: http://hdl.handle.net/1880/112238. Accessed 10 April 2021.

PEIRCE, C. S. 1909. *The Logic Notebook (1865-1909)*. Charles Sanders Peirce Papers MS Am 1632 (339). Houghton Library, Harvard University, Cambridge, Mass. Available at: https://nrs.harvard.edu/urn-3:F-

HCL.Hough:3686182. Accessed 10 April 2021.

SALATIEL, J. R. 2021. On a new approach to Pierce's three-value propositional logic. Manuscript submitted for publication.

SMULLYAN, R. 2009. *First-Order Logic*. New York, Dover Publications.

SURMA, S. 1977. An algorithm for axiomatizing every finite logic. In: D. RINE (Ed.). *Computer Science and Multiple-Valued Logics*: theory and applications. Amsterdam, New York and Oxford, North-Holland Publishing Company, p. 137-143.

TURQUETTE, A. 1976. Minimal axioms for Peirce's triadic logic. *Zeitschrift für mathe- matische Loßik und Grundlagen der Mathematik*, **22**: 169-176.

_____. 1978. Alternative axioms for Peirce's triadic logic. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, **24**: 443-444.

_____. 1981. Quantification for Peirce's preferred system of triadic logic. *Studia Logica*, **40**: 373-382.

WÓJCICKI, R. 1988. *Theory of Logical Calculi*: basic theory of consequence operations. Dordrecht, Springer.

Submitted on July 22, 2021.

Accepted on September 08, 2021.