4. Semantics: From BA to Boundary Logic.

"When logically analyzed, order turns out to be… inconceivable and incomprehensible to us unless we had the idea… expressed by the term 'negation'. Thus it is that negation, which is always also something intensely positive, not only aids us in giving order to life, and in finding order in the world, but logically determines the very essence of order." Royce (1917: 540)

LoF (pp. 113-17) shows how the CTV and the elementary Boolean algebra of sets are possible interpretations (models) of the pa. Before showing how the pa translates the CTV, I first sketch some facts about the key players in the CTV, the truth functors (functors for short). A functor has an arity $n \in N$. Because B has cardinality 2, there are 2^{2^n} possible functors with arity n; in particular, there are 16 binary functors. Six of these map a and b into one of $\{a, \neg a, b, \neg b, \mathsf{T}, \mathsf{F}\}$ and will not detain us. The remaining 10 binary functors are $\{\wedge, \vee, \rightarrow, \leftarrow, \leftrightarrow\}$ and their negations (see Table 4-2). There are $2^{2^1}=4$ possible unary functors; of these, only $\neg a$ need be considered. There are $2^{2^0}=2$ 0-ary functors; these are T and F by convention. All functors of arity>2 are redundant, because any formula employing such functors is equivalent to a formula whose functors have arity of at most two (Epstein 1995: §II.J.3).

Table 4-1. Some Interpretations of the pa.

Interpretation	*Primal*		*Dual*
Key Binary Functor	Alternation	Conditional	Conjunction
Implied EA Functor Pair	$\{\vee,\neg\}$	$\{\rightarrow,\mathsf{F}\}$ or $\{\rightarrow,\neg\}$	$\{\wedge,\neg\}$
pa *Equivalent*	{((··)),(·)}	{(·)·,(())} or {(·)·,(·)}	{((··)),(·)}
Representation of:			
Alternation	*ab*		(*a'b'*)
Conjunction	(*a'b'*)		*ab*
Conditional	*a'b*		(*ab'*)
Antecedents[31]	1910: *PM*, 6.11	1879: Frege, 1.1 1885: Peirce, 3.11 1956: Church, 1.4c, $\mathbf{P_1}$	1892: Johnson 1897: Peirce EG
Recent Examples	Halmos & Givant (1998: §§8,13)	Machover (1996: §7.6) Bostock (1997: §5.2)	Quine (1982: §1)

It would seem that there are 5+1+2=8 essential truth functors. In fact, there is ample redundancy among these, in that some of these 8 can be defined in terms of the remainder. If for any CTV formula, there exists an equivalent CTV formula in which only a subset of these 8 functors appears, the members of that subset are termed *expressively adequate* (abbreviated EA) or *truth-functionally complete* (Bostock 1997: §§2.7, 2.9).[32]

31. Numbers refer to systems in Prior (1962: Appendix I). For more re systems with $\{\rightarrow,\mathsf{F}\}$ primitive, cf. Prior (1962: §I.III.1). Systems based on $\{\rightarrow,\neg\}$ are quite standard, e.g., Church's (1956: §20) $\mathbf{P_2}$, Epstein's (1995: 408) **PC**, and Mendelson's (1997: 35) **L**. Systems with $\{\wedge,\neg\}$ primitive include Johnson's (1892), discussed in §5.1.2 below, the modal logics of C. I. Lewis (11.1), and ones proposed independently by Rosser and Sobocinski (6.3). For more on historical axiom systems, see §5.1.2 below. §6.0 discusses Peirce's existential graphs.

32. For more on connectives, axiomatics, etc., see the references under *Calculus of Truth Values* in the Bibliographic Postscript.

For the purpose at hand, the primitive basis of CTV (e.g., DeLong 1971: 107) consists of the primitive values T and F, and any EA set of functors. *Boundary logic* results from a one-to-one correspondence between the BA and EA sets of functors. Among the binary functors, $\vee$, $\wedge$, and $\leftrightarrow$, commute and associate, just as juxtaposition does in the pa. Denial, $\neg$, is a unary functor whose scope is set by brackets, which is exactly the way $(\cdot)$ works in the pa. We shall see in section 4 that $\{\vee,\neg\}$ and $\{\wedge,\neg\}$ are EA; the upshot is two of the three interpretations of the pa shown in Table 4-1.

I now establish a correspondence between the PA and conventional logic, beginning with the assumption $()\Leftrightarrow$T. Table 2-1b immediately reveals that the semantics of (α) are identical to those of $\neg\alpha$, namely $\neg$T$\Leftrightarrow$F (A2) and $\neg$F$\Leftrightarrow$T. Thus emerges the most salient fact about boundary logic: *just as an empty boundary denotes a primitive value, a negation with an empty scope denotes a truth value.* $()() = ()$ (A1) and $\bot\bot = \bot$ in Table 2-1a imply that juxtaposition is idempotent. The two remaining cells of Table 2-1a reveal that juxtaposition commutes, as discussed in §3.2. Hence by virtue of the PA, $\alpha\beta$ can interpret either $\alpha\vee\beta$ or $\alpha\wedge\beta$ and the road to a CTV translation of Table 2-1 is now clear.[33]

Since Frege and Peirce, the predominant primitive CTV connective has been the conditional, for which the current notation is '$\rightarrow$'.[34] The well-known equivalence $a\rightarrow b \Leftrightarrow \neg a\vee b$ suggests the translation $a\rightarrow b \Leftrightarrow a'b$. Then note that $a\rightarrow$F $\Leftrightarrow a'\bot = a'$; thus the pa also translates the EA set $\{\rightarrow,$F$\}$. Table 2-1b now translates as T$\rightarrow\bot$ and its converse, $\bot\rightarrow$T. Other possible translations of the pa into the CTV are discussed in §4.2. Table 4-1 summarizes this section.

4.1. Duality.

§3.3 concludes that *B* is partially ordered. Since *B* has only two members, they must also be comparable, else *B* cannot be ordered at all. Hence either $\bot\leq()$ or $()\leq\bot$ must be the case. $()=\bot$ leads to triviality; hence the inequalities must hold strictly. Thus far, I have tacitly assumed $\bot\leq()$. Somewhat arbitrarily, I refer to the semantics that flow from the assumption $\bot\leq()$ as the *primal interpretation*. $()\leq\bot$ is equally sensible and gives rise to the *dual interpretation*. That logic and Boolean algebra can be carried out under either interpretation gives rise to *duality*. To switch interpretations, *mutatis mutandis*, is known as *dualisation*. Duality is little more than a semantic implication of *B* being an ordered set.[35] Duality, unmentioned in *LoF*, is a compelling reason for an explicit symbol denoting the unmarked state.

---

33. 3.3.3 and 3.3.4 also suggest that the CTV with primitive $\{\vee,\neg\}$ is a model for the pa.

34. "Conditional" is Quine's (1982: §3) word, who rightly prefers it to the "material implication" of *PM*, because $a\rightarrow b$ does *not* translate "*a* implies *b*", but rather "if *a* then *b*". Better yet, $a\rightarrow b$ should be read as a synonym for $a\leq b$, where *a* and *b* are members of some ordered set.

35. Let *S* be a set partially ordered by the relation ρ, with $a,b\in S$. Then *S* is also partially ordered by the relation ρ', such that $b\rho' a$ is true whenever $a\rho b$ is true; this is the *principle of duality* for posets (Donnellan 1968: Th. 13; Stoll 1974: 193-94). Let S=B, ρ=$\leq$, and ρ'=$\geq$, and the duality of CTV and Ba follows. Ba is typically formulated such that $1\in B$ and $1'\equiv 0\in B$, with $0\neq 1$. BA duality follows from $(())\neq()$ being a trivial consequence of A2, and from $(())\neq() \Leftrightarrow 0\neq 1$. The nearest *LoF* gets to the subject of this footnote is the first complete paragraph on p. 113.

An interpretation begins with a one-to-one correspondence, and there are two possible one-to-one correspondences between B and {T,F}. Thus far, I have assumed T⇔(). All interpretations based on this assumption result in the same pa translation of any given CTV formula. The dual interpretation begins with T⇔⊥. Juxtaposition now denotes conjunction, and (ab') the conditional. The dual of {∨,¬} is {∧,¬}. Under the dual interpretation, Table 2-1a is now the table for Boolean multiplication, and Boolean and numerical multiplication yield the same result when the carrier is assumed to be {0,1}. Since the two cells in Table 2-1b form a dual pair, duality suggests that only one of these cells is strictly necessary.

Perhaps surprisingly, Table 2-1 and T1-T7 hold under both interpretations. Likewise, the rules defining and simplifying PA and pa formulae do not change. Hence the syntax of BA is invariant under dualisation. Dualising the PA does, however, reverse the mapping of () into T, and ⊥ into F.

Matters are a bit more involved for the pa, because dualization alters the semantics of juxtaposition. The semantics of a formal system are known as its *truth definition* or *Boolean valuation* (Smullyan 1968: §I.2, Def. 1). Recall that an atomic valuation (3.1.3) assigns one of () or ⊥ to every variable. The pa then enjoys the following very simple truth definition:

4.1.1. Definition. *A Boolean valuation for* BA. Let ϕ,δ be metalogical notation for BA formulae, and let the value of ϕ be |ϕ|, given some atomic valuation. All molecular formulae then evaluate to either () or ⊥ by recursive application of two elementary rules: |(ϕ)|=(|ϕ|), and |δϕ|=max[|ϕ|,|δ|], where max[(),⊥]=() [=⊥] under the primal [dual] interpretation.[36]

This truth definition follows trivially from Table 2-1. A tautology can now be defined as a formula whose value is invariant to atomic valuation. Moreover, ϕ=δ is a tautological equivalence if |ϕ|=|δ| holds for all atomic valuations. Further definitions:

4.1.2. Definition (adapted from Halmos and Givant 1998, §22): Let $\alpha = \alpha\langle a_1,...,a_n\rangle$ be a formula containing the atomic formulae $a_1,...,a_n$. α is the *primal*, $(\alpha\langle a_1,...,a_n\rangle)$ the *complement*, $\alpha\langle a'_1,...,a'_n\rangle$ the *contradual*, and $\alpha^D = (\alpha\langle a'_1,...,a'_n\rangle)$ the *dual*. The dual of the dual is the primal; hence a primal and its dual are known as a *dual pair*.

4.1.3 answers the following question: if the equation $\alpha = \phi$ holds under some or all atomic valuations, what is true of α^D and ϕ^D?

4.1.3. Duality Theorem. If $\alpha = \phi$, then $\alpha^D = \phi^D$.

Proof. See §A.9.

The Duality Theorem is the basis for the *duality principle* characterising Boolean algebra and CTV, to the effect that the dual of a tautology is also a tautology. Keeping in mind that α in 4.1.2 is a tautology if α=() or ⊥ for all possible valuations of $a_1,...,a_n$, 4.1.3 says

36. For more on truth definitions, see Bostock (1997: §§2.4, 3.4) and Hodges (2001: §3).

more. If a formula or equation is tautologous under some interpretation, then its contradual and dual are also tautologies under that same interpretation.[37]

There is no syntactical or proof-theoretic ground for preferring one interpretation over the other. Spencer-Brown preferred the primal interpretation, claiming that *a'b* is a more economical represention of the conditional than (*ab'*) (*LoF*, p. 113-14),[38] and I share that preference. Yet I agree with Prior who wrote:

> "…'and' and 'not' are the only operators which are quite unambiguously truth functional in ordinary speech; truth functional interpretations of other ordinary-speech connectives all wear at times an air of artificiality." Prior (1962: 254).

We shall se in §6.0 that C. S. Peirce, too, came to prefer the dual interpretation.

4.2. From the pa to the CTV.

"*…everything in pp. 98-126 of* Principia Mathematica *can be rewritten without formal loss in the one symbol* '()'*… Allowing some 1500 symbols to the page, this represents a reduction of the mathematical noise-level by a factor of more than 40,000.*" LoF, p. 117.

Table 4-2 translates, assuming the primal interpretation, the ten nontrivial CTV binary connectives into the pa. Each row of Table 4-2 contains a dual pair; hence the connectives can be grouped into two groups of five, I and II, with each group being the dual of the other. Connectives sharing the same numerical identifier (shown in the two middle columns) can be derived from each other via negation. Let *a** stand for either *a* or *a'*; *a** is a *literal*. The *simple* connectives are those that can be described by *a*b** or the duals thereof; these are *ab*, *a'b*, *ab'*, *a'b'*, and their duals.

Note that the assignment of () to either T [⊥≤()] or F [()≤⊥] is arbitrary. But once a choice is made, the pa representation of all connectives is determined. Table 4-2 translates *ab* as $a \vee b$, and its dual, (*a'b'*), as $a \wedge b$, both as per the first column of Table 4-1. Likewise, either *a'b'* or (*ab*) translates the Sheffer stroke, *a*|*b*. These translations renders obvious that | can be read as "not and" and as "if *a* then not *b*"; the latter reading perhaps suggests more strongly the peculiar expressive power of the Sheffer stroke. The duality of *a'b'* and (*ab*) points to "not or" as the semantic dual of the Sheffer stroke.[39]

Table 4-2. The 10 Nontrivial Binary Connectives (Functors).	
Primal	**Dual**

37. Quine's (1982: §12) second law of duality is my definition 3.4.1. His third law is α[*true*=()] ⇔ α^D [*true*=⊥] in this paragraph; the fourth law is $(\alpha \to \phi) \leftrightarrow (\phi^D \to \alpha^D)$. The fifth law is the Duality Theorem. On duality, see also Bostock (1997: §2.10).

38. Spencer-Brown makes too much of this, especially if one downplays the conditional in favor of conjunction/alternation. Moreover, while (*ab'*) has more symbols than *a'b*, (*ab'*)=(()) is equivalent to *b'a*=(), which has no more symbols than *a'b*=().

39. The misconception that the pa is little more than a new notation for the Sheffer stroke (Grattan-Guiness 2001: 557; Wolfram 2002: 1173) may stem from hasty readings of Appendix 1 of *LoF*.

Name	*Logic*	*Sets*	**pa**			**pa**	*Sets*	*Logic*	*Name*
Alternation	$a \vee b$	$a \cup b$	ab	1	5	$(a'b')$	$a \cap b$	$a \wedge b$	Conjunction
Conditional	$a \rightarrow b$	$a \subseteq b$	$a'b$	2	4	(ab')	$b \sim a$	$\neg[a \leftarrow b]$	Difference
Symmetric Difference	$\neg[a \leftrightarrow b]$	$a \Delta b$	$(a'b)(ab')$	3a	3a	$((a'b)(ab'))$	$a \subseteq b \subseteq a$	$a \leftrightarrow b$	Biconditional
			$((a'b')(ab))$	3b	3b	$(a'b')(ab)$			
Converse	$a \leftarrow b$	$a \supseteq b$	ab'	4	2	$(a'b)$	$a \sim b$	$\neg[a \rightarrow b]$	Difference
Sheffer stroke	$a \mid b$	$\overline{a \cap b}$	$a'b'$	5	1	(ab)	$\overline{a \cup b}$	$a \downarrow b$	NOR

Note. Each row contains a *dual pair*. Items with same number are negation pairs. The 6 remaining binary connectives are uninteresting, because they map $\{a,b\}$ into one of a, b, $\neg a$, $\neg b$, **T**, and **F**.

The meaning of CTV duality should now be clear: for any statement α, there exists an equivalent statement α^D derived by interchanging $\wedge$ and $\vee$, $\rightarrow$ and $\neg\leftarrow$, $\leftrightarrow$ and $\neg\leftrightarrow$, | and $\downarrow$, and T and F. More generally, under either interpretation, the pa representation of conjunction is the dual of the pa representation of alternation, and the same dual relation holds for $\rightarrow$ and $\neg\leftarrow$.

A given pa formula enjoys a multiplicity of CTV translations, revealing the ample redundancy inherent in the CTV. For instance, take the well-known De Morgan's laws, $\neg(a \vee b) \leftrightarrow (\neg a \wedge \neg b)$ and $\neg(a \wedge b) \leftrightarrow (\neg a \vee \neg b)$. The pa translations, $(ab)=(ab)$ and $a'b'=a'b'$, render trivial that these laws are true and form a dual pair.

Because the algebra of sets is a model for the Boolean algebra **2**, it is also a model for the pa. Let U be the universal set, $a,b \in$ U, and $\varnothing$ be the null set. Then the columns headed by "Sets" show how the algebra of sets and the pa are equivalent.

Table 4-3 translates the *LoF* consequences into CTV notation, using Table 4-2 as the key. For each *LoF* consequence, Table 4-3 also supplies a name, if the conventional literature provides one, and the number of the corresponding tautology in Kalish et al (1980: §II.11) (KMM), an unusually comprehensive list of tautologies. *LoF* says very little about how J1-C9 relate to the extant literature on logic and Boolean algebra. J2, C1, C3, and C5 should be very familiar. J1 is a form of the Law of Excluded Middle (LEM)[40]; C2 is Johnson's (1892: 342) Law of Exclusion; C4 is the biconditional corresponding to an axiom in conditional form proposed by Peirce (W5: 162-90) in 1885. C6 is the Law of Elaboration or Development, so called by Bostock (1997: 41). C7 and C9 are not well known. If a and b on either side of C9 were to trade places, the two sides of C9 would then form a dual pair.[41]

Table 4-3.
The *LoF* Initials and Consequences, Conventionally Translated.

40. *Intuitionist logic* is based on axioms (Bostock 1997: 235, (1)-(9) & EFQ) from which the LEM (and J1, C1) cannot be proved. Finding a boundary version of the intuitionist CTV would be an interesting exercise.

41. *PM* (*2–*5) proves many tautologies including the following ones from *LoF*: J1, 3.24; J2, 4.41; C1, 4.13; C2, 2.621 & 2.67; C5, 4.25; C6, 4.42; C8, 4.4. Other lists of tautologies include Rosser (1953: Theorem VI.6.1), Carnap (1958: T8-2, T8-6), Wolf (1998: Appendix 3), and Cori & Lascar (2000: §1.2.3). In Boolean algebra J1 is known as complementarity; C3 is union; C5, idempotence; De Morgan's laws, dualization. For more on the relation between J1-C5 and conventional logic and Ba, see §A.7.

LoF	Conventional Logic	Name	KMM
J1	$\neg(a\to a) \leftrightarrow \bot$	Law of Contradiction	59
J2	$(a\vee r)\wedge(b\vee r) \leftrightarrow (a\wedge b)\vee r$	Distributive Law	62
C1	$\neg(\neg a) \leftrightarrow a$	Law of Involution	110
C2	$(b\vee a)\to a \leftrightarrow b\to a$	Law of Exclusion	73
C3	$\mathsf{T}\vee a \leftrightarrow \mathsf{T}$	T is the *unit element* for alternation	
C4	$(a\to b)\to a \leftrightarrow a$	Peirce's Law	23
C5	$a\vee a \leftrightarrow a$	Law of Tautology for alternation	47
C6	$(a\wedge b)\vee(a\wedge\neg b) \leftrightarrow a$	Law of Elaboration	68
C7	$[(a\to b)\wedge\neg c] \leftrightarrow \neg[(a\vee c)\wedge(b\to c)]$		
C9	$[(a\to r)\wedge(r\to\neg b)]\leftrightarrow\neg[(a\vee r)\wedge(r\to b)]$		

LoF invokes J1-C3 104 times, C4-C9 15 times, and C5-C8 a mere 6 times. *LoF* invokes C2 more often than any other consequence, C1 excepted. C2 allows a subformula to be copied into and erased from any subspace deeper than the shallowest instance of itself (with the proviso that a subformula cannot be copied into a part of itself). While not a standard part of elementary logic, C2 is a powerful tool for simplifying formulae and a trivial corollary of the Ba Consistency Principle, 3.3.5.[42]

It could be argued that all I have done thus far is to dispose of the truth functors by means of the following very elementary reasoning. Alternation and conjunction commute and associate; hence mere juxtaposition suffices to notate either. Brackets are then free to notate negation. It is well known that negation and one of conjunction or alternation are EA. Hence brackets are the only explicitly truth functional notation required. QED. But here lies a bit more mathematical meat than meets the eye, because juxtaposition and enclosure can be eliminated in favor of a single two place functor, $(\cdot)\cdot$ or $\cdot(\cdot)$, and the constant $\bot$=(()). To see this, note that ab = $((a))b$ [C1] = $((a)\boldsymbol{b})b$ [C2] and (a) = $(a)\bot$ [A2]. In the terminology of universal algebra (Abbott 1969: §2-5), the pa is a $\langle B, (\cdot)\cdot,\bot\rangle$ algebra of type $\langle 2,0\rangle$,[43] as well as a $\langle B,\cdot\cdot,(\cdot),()\rangle$ algebra of type $\langle 2,1,0\rangle$.

The *normal form* (NF) of a pa formula runs as follows. Let the pa formula α contain n variables so that $\alpha = f(a_1,...,a_n)$. The NF is a formula equivalent to α having the form:

(#) $$(a_i^*...)_j \Leftrightarrow \bigvee_j[\bigwedge_i a_{ij}^*].$$

All variables in (#) appear as literals. The NF can be seen as the analog of a polynomial in ordinary algebra.

$(a_i^*...)_j$ is the *jth disjunct*. The ranges of the indices i and j begin with 1 and are finite; otherwise, these ranges are deliberately unspecified, if only because the NF is not unique. Also, either i or j may in some cases not exceed 1. If the jth disjunct is $(\bot)$, then the entire NF degenerates to (); if it is (()), the jth disjunct vanishes. It is easier to parse a NF if the variables in each disjunct appear in lexicographic order, moving from left to

42. See §A.6 for more on C2.

43. A model of this algebra is Church's (1956) $\mathbf{P_1}$; see Table 4-1. Interpreting juxtaposition as 'product' and enclosure as 'inverse,' the pa is a monoid with identity element $\bot$ (because $a\bot=a$), and would be a commutative Abelian group if $(a)a=\bot$ were the case.

right. This reordering is allowed because the variables in any disjunct can be reordered at will, but is not a mathematical imperative.

Given any Ba/CTV formula, there exists an equivalent formula resembling the rhs of (#), namely a series of subformulae linked by alternation. Each of these subformulae in turn consists of literals linked by conjunction. This is the disjunctive normal form (DNF), closely related to the NF; the dual of the DNF is the conjunctive normal form (CNF). *LoF* is silent about the well known Ba/CTV result that there exists a CNF/DNF dual pair equivalent to any formula. In the pa, the distinction between the DNF and the CNF is merely semantic.[44]

A Historical Digression on Notation.
What is here enclosed in parentheses, *LoF* would place under $\rceil$, the 'mark'; e.g., I write $(a)b$ and (ab) where *LoF* writes $\overline{ab}\rceil$ and $\overline{a}\rceil b$. Martin Gardner (*Scientific American* 1980 (2): 14) deemed *LoF*'s notation "eccentric." LoF's notation has antecedents. Peirce (4.12-20), written in 1880 but not published until 1933, proposed to notate Ba with concatenation, interpreted as NAND, and brackets. This notation is that of this paper, but for his limitation of concatenation to a binary scope. Kauffman (2001) points out that Peirce later fused the overbar (denoting Boolean complementation) to the Boolean '+' (OR) to create a symbol Peirce called the "sign of illation," closely resembling the '$\rceil$' of *LoF* and having identical semantics.[45] Peirce saw that his sign of illation sufficed for Boolean algebra and syllogistic logic. Kauffman cited Peirce (1976: 106-15), an excerpt from a manuscript titled "Qualitative Logic," written in 1886 but not published in full until 1993 (*W*5: 323-71). Kauffman also notes that the '$\cdot\lceil\cdot\;\cdot$' notation of Nicod (1917) likewise has the functionality of '$\overline{\cdot\;\cdot}\rceil\cdot$', but does not mention that Nicod on occasion commuted '$a\overline{\lceil b}$' to '$\overline{b}\rceil a$'. Some authors (e.g., Curry 1963) write $\rceil a$ in place of $\neg a$.

4.3. The Metatheory of the pa.
T14 through T18 lay down the metatheory of the pa. Proofs are in §A.8.

T14. Let α be a formula such d^*_α>2. Then α can be transformed, by taking steps, into an equivalent formula β such that d^*_β=2.

Remarks.
1. In *LoF*, T14 does no more than help prove T15 and T17.

2. The proof relies on the ability of C7 to transform any subformula of depth 3 into an equivalent sub-formula of depth 2. Invoking C7 repeatedly, beginning at each point in α with depth=d^*_α–3, transforms α into an equivalent formula with maximum depth≤2. The Appendix proof views a pa formula as an ordered tree; the *LoF* proof does not.

44. For more on the CNF and DNF, see Quine (1982: §10), Bostock (1997: §2.6), Halmos & Givant (1998: §38), and Cori & Lascar (2000: §1.3.2). Bostock defines the DNF so that each disjunct includes all *n* variables, in which case *i* in (#) necessarily ranges over 1 to *n*. He does this so that the truth table corresponding to α can be easily recovered from the DNF. This stipulation is unnecessary here, because truth tables play no essential role in the pa.

45. Peirce's manifold contributions to mathematics, logic, and semiotics inform Kauffman's discussion in other ways.

3. Read from left to right, both C4 and C9 can also be seen as depth reduction tools. C4 [C9] reduces a subformula of depth 2 [3] to one of depth 0 [2].

4. Note that no formula in J1-C9 is more than two parentheses deep, the left side of C7 and C9 excepted.

T15. Let the pa formula $\alpha\langle v\rangle$ contain more than two instances of the variable v. Then $\alpha\langle v\rangle$ can be transformed, by taking steps, into an equivalent formula $\beta\langle v\rangle$, such that $\beta\langle v\rangle$ contains at most two instances of v.

Remark. In *LoF*, T15 only serves to prove T17. T15 is essentially a simple form of the following well-known theorem of Boolean algebra (Hohn 1966: 229, Lemma 2), which I recast into pa notation as follows:
Let f be a truth function whose arguments are $x_1,...,x_n$. Then $f(x_1,...,x_i,...,x_n)=$ $(f(x_1,...,(),...,x_n)'x_i')\ (f(x_1,...,\perp,...,x_n)'x_i)$, $\forall i$, $1\leq i\leq n$.

T14 and T15 together guarantee that every pa formula has an NF equivalent, whose depth does not exceed 2 and that contains at most two instances of any given variable.

T16. If two or more formulae are equivalent in every case of one variable, they are equivalent.
Remarks.
1. I propose to restate this enigmatic theorem as "Let the variable v appear in one or both of the formulae α and β, and let $v_i=|v|$. Let $\alpha\langle v=v_i\rangle$ and $\beta\langle v=v_i\rangle$ be $\alpha\langle v\rangle$ and $\beta\langle v\rangle$ with v set to v_i. If $\alpha\langle v=()\rangle=\beta\langle v=()\rangle$ and $\alpha\langle v=\perp\rangle=\beta\langle v=\perp\rangle$, then $\alpha=\beta$." The converse is also true.
2. *LoF* maintains that T16 justifies a decision procedure described in §5.3.[46]
3. Erasing every instance of a variable is equivalent to setting that variable equal to $\perp$. Hence T16 has another implication, unmentioned in *LoF* (and elsewhere): a tautology remains a tautology when every instance of any variable is erased.
4. Prior (1962: §I.III.4) shows that the CTV can be derived from a metalogical axiom of the form $(\alpha\langle v=()\rangle)(\alpha\langle v=\perp\rangle)\,\alpha\langle v\rangle=()$. This is effectively T16.

T17. The pa is complete.
Remark. A logic is *complete* if for any tautology α, one of α or $\neg\alpha$ can be proved from the axioms/initials, using the inference rules. The pa inference rules are in fact R1 and R2, although *LoF* does not make this explicit. Moreover, if a formula simplifies to a member of *B*, T17 also implies that there exists a corresponding tautology in the pa.

If a logic is *sound*, then there does not exists a formula α such that both α and $\neg\alpha$ can both be proved in that logic. If both α and α' were provable in the pa, then $(\alpha'\alpha)=()$, contradicting J1. Hence the soundness of the pa follows from its com-

46. T16 is Cole's (1968: 346) rule R2. *LoF* (p. xvii) acknowledges that T16 resembles a lemma in Quine's (1938) proof that the CTV is complete. *LoF* neglects to mention Quine's later invention of TVA, which is essentially identical to the *LoF* decision procedure for which T16 is the main justification. Prior's (1962: 53, (3); 58-60) re-exposition of Quine's proof includes proving a lemma that is essentially T16. The *LoF* proof of T16 (restated in §A.8) is vastly easier than either Quine's or Prior's.

pleteness. If there exists a formula α such that both α=() and ¬α=(), then all formulae of that logic can be equated to ().[47] In short, a logic both sound and complete is such that for any statement α, α *is a tautology* ⇄ α *is provable*.

While T17 is arguably the most important (meta)theorem of the BA, it is not an unexpected result because, as we shall see, the CTV and the Boolean algebra **2** are both models of the pa, and the completeness of these models is well established. Moreover, the completeness asserted by T17 is of the strong sort (*LoF*, p. 119), in that a pa demonstration not only yields a tautology because the initials are tautologous (this is DeLong's 1971: 134 *hereditary property*), but adding an initial that cannot be proved from the existing initials renders the pa unsound.

The *LoF* proof of T17 requires all *LoF* consequences except C4 and C5; hence T17 can be seen as the culmination of chapters 1-10 of *LoF*. The proof of T17 resembles Quine's (1938) proof (which *LoF* cites) that the CTV is complete, in that both proofs proceed by strong induction on the total number of variables in hypothetical pa formulae in normal form; this is the only explicit instance of an inductive proof in *LoF*. Crucial to this proof are two facts:

- Every pa formula has an equivalent in normal form, by virtue of T14 and T15;
- A1 [A2] is a tautological equivalence because it is an instance of C3 [C1].[48]

T18. The initials J1 and J2 are independent.

Remark. That is, neither initial can be proved from the other alone. The very concise *LoF* proof of T18 is wholly syntactic, and is predicated on there being only two initials, a signal advantage of the basis J1,J2. However, given that C6 alone suffices to demonstrate J1 and J2 (cf. §5.1 and §A.4), T18 loses some of its luster. On axiom independence, also see Hunter (1971: §36) and Bostock (1997: 195-99).

4.4. The Enigmatic Degeneracy of BA.

---

47. *Dem*. By C2, *b*(*ba*)=*b*(*a*) for any *a*,*b*. Now let *b* be any pa formula whatsoever, and let *a* be a formula such that both *a*=() and *a*=⊥ are demonstrable by hypothesis. The lhs of C2 evaluates as *b*(*ba*) = *b*(*b*()) [since *a*=()] = *b*(()) [C3] = *b* [A2]. The rhs of C2 evaluates as *b*(*a*) = *b*(⊥) [since *a*=⊥] = *b*() [A2] = () [C3]. Hence *b*=(), the desired absurd result. □ On soundness, see Hunter (1971: §§24, 25a,b, 28) and references therein to Church (1956).

48. §A.8 proves T17 and §A.1 demonstrates every consequence that proof requires. §A.8 also includes a pa version of Kneebone's (1963: 48) proof that the CTV is complete, perhaps the simplest proof of this nature extant. Post (1921) was the first to prove the CTV complete (for a summary, see Hunter 1971: §30). For the completeness proofs of Hilbert and Ackerman, and of Quine (1938), see Prior (1962: §§I.II.3, I.III.2). These proofs require over 20 tautologies apiece. Hunter (1971: §§31, 32) restates the proofs of Kalmar and Henkin. Kalmar's proof is the basis for those of Stoll (1963: Th. 9.2.3), Epstein (1995: §§II.L.2-3, II.M.2), and Mendelson (1997: 1.14). These proofs require the Deduction Theorem and at least a dozen lemmas. The proof of T17 merely requires J1-C7 and C9. Henkin's proof has the advantage of yielding the Compactness and Interpolation theorems as corollaries. Nowadays, the preferred approach to proving the CTV complete relies on refutation trees (e.g., Bostock 1997: §§4.6-7; Smullyan 1968: chpt. II). For amusement, see the cryptic proof of Anderson and Belnap (1959), then Hunter's (1971: §37.4) less terse restatement of the same. The reader is invited to peruse this literature and then draw a conclusion about the conciseness of the pa.

The expressively adequate (EA) subsets of the functors in common use are {→, F/¬}, {∧/∨, ¬}, and {↔, F, ∨/∧}. (This last subset I shall not discuss further.) Table 4-1 shows how the resulting four EA functor sets map into BA. This mapping reveals a curious detail: corresponding to each EA functor set is a pair of BA formulae, one involving one boundary, the other two. The question naturally arises as to whether this is true of all nine EA functor sets with two members. Table 4-4 reveals that this is nearly the case. The nine EA functor sets consist of four dual pairs and one self-dual set, {→,¬→}. The first four rows of the Table show how seven of the nine EA functor sets can be derived by inserting 1 or 2 letters into (), and 0 or 2 letters into (()). The fifth row reveals that the dual pair of EA functors involving ↔ cannot be represented in this manner, suggesting that ↔ should be seen as a tacit conjunction of conditionals. Hence I submit that the first four rows of Table 4-4 capture the essence of Ba and the CTV. I leave to the future the further exploration of the symmetries inhering in the two leftmost columns of Table 4-4.

Table 4-4.
Building the Nine EA CTV Functor Pairs from () and (()).

pa		*EA CTV Functor Pairs*			
		primal		*dual*	
(*a*)*b*, (())	2,0	a→b	F	¬(b→a)	T
((*a*)*b*)*b*=((*ab*))=*ab*, (*a*)	2,1	a∨b	¬a	a∧b	¬a
((*b*)*a*), (*a*)	2,1	¬(b→a)	¬a	a→b	¬a
(*a*)*b*, ((*b*)*a*)	2,2	a→b, ¬(b→a)			
(*a*)*b*, ((*a*)*b*)((*b*)*a*)	2,4	a→b	¬(a↔b)	¬(b→a)	a↔b
Source for EA Functor Pairs: Wernick (1942: 132).					

The pa suggests that expressive adequacy requires two capabilities, namely a way of:

- Concatenating subformulae. Let these ways be *a*b* and (*a*b*);
- Enclosing subformulae. Create *a′* in one of the following ways:
  - Invoke it outright;
  - Starting from the Sheffer stroke, (*ab*), set *a*=*b* or (*aa*)= *a′* [C5];
  - *a′*(()) = *a′* [A2]. (()) follows from (*b′a*): either erase *a*,*b*, or let *a*=*b* and J1.

Note that the Sheffer stroke is EA by itself. As () and (()()) denote distinct primitive values, () alone suffices for all of truth functional logic.

Table 4-5. How *a*b*, (*a*b*), and *a′* Yield
The Sheffer Stroke and the Nine EA Functor Pairs.

	Interpretation:		*Needed to Obtain* (*a*):			
	primal	*dual*	*Assume*	*Interpretation*		(*a*) =
Commute						
ab	a∨b	a∧b	*a′*	¬a		---
(*ab*)	a↓b	a | b	---	---		(*aa*)
Do not Commute				*primal*	*dual*	
a′b	a→b	¬[b→a]	*a′*	¬a		---
"	"	"	(())	**F**	**T**	*a′*(())
"	"	"	(*c′a*) †	¬[c→a]	a→c	*a′*(*a′a*)
"	"	"	(*c′a*)(*c′b*)	¬[a↔c]	a↔c	*a′*(*a′a*)(*a′a*)

† Self-dual row.

Tables 4-4 and 4-5 reveal that all possible EA functor pairs can be obtained by inserting letters in certain ways into () alone, or into () and (()). Hence there is a sense in which the members of *B* encapsulate all EA functor pairs. The members of *B* can be seen as the operators (·)· and ((··)·), where '·' indicates a possible location of a letter. BA does not syntactically demarcate operators from operands; the operators (·)· and ((··)·) can be distinguished from the primitive values [operands] () and (()) only in context.

5. Proof and the pa.

"*As a material machine economises the exertion of force, so a symbolic calculus economises the exertion of intelligence. …the more perfect the calculus, the smaller the intelligence compared to the results*." Thus begins Johnson (1892).

What is conventionally referred to as a proof, *LoF* calls a *demonstration*, meaning a sequence of steps showing that two pa formulae, e.g., ϕ and γ, are equivalent; the result is the consequence $\phi=\gamma$. Each step invokes an axiom, initial, or consequence already demonstrated. R1 or R2 are explicitly invoked in demonstrations. A demonstration is carried out entirely *within* an object language, the pa or other formal system. The correctness of a demonstration can be verified algorithmically, at least in principle.[49] In *LoF*, *proof* applies only to (meta)theorems. A proof is necessarily metalinguistic and may draw on any resource from mathematics or logic. There can be no decision procedure for verifying the correctness of a proof.

A *calculation* is a demonstration that methodically eliminates all variables from a pa formula.[50] A calculation may be executed via the following algorithm:

5.0.1. Algorithm.

1. Alter α in a series of steps, justifying each step using one or more initials and consequences, with the objective of finding a formula β equivalent to α, from which all variables have been eliminated;

   Remark. C1, C2, and J1 are especially powerful here. But C4-C6 can also be seen as tools for eliminating redundant variables.

2. If β exists, it will either be a member of *B* or a PA formula which, by T3-T4, can be simplified. Either way, α is a tautology;
3. If β does not exist, then α is satisfiable.

End of Algorithm

A calculation begins with the sign *Cal*. To verify a pa equation of the form $\phi=\gamma$, first calculate $\phi'\gamma$, denoted LR (*left* to *right*), then calculate $\gamma'\phi$, denoted RL (*right* to *left*). If both

49. On distinguishing "proof" from "demonstration," see Quine (1951: 319-22), Machover (1996: 120), and Mendelson (1997: 36, fn †).

50. "Calculation", a word not appearing in *LoF*, is a shortened form of Dijkstra and Scholten's (1990: 21) *calculation proof*, meaning a series of steps that transform a given Boolean expression into the Boolean value *true*.

conditionals can be transformed into the same formula, $\phi=\gamma$ holds by T7. Equivalently, $\phi=\gamma$ holds if a calculation reduces the biconditional $((\phi'\gamma)(\gamma'\phi))$ to (), in one fell swoop.

A demonstration of $\phi=\gamma$ consists of a sequence of formulae, beginning with ϕ. Each formula in the sequence results from a *step*, inferred from one or more preceding formulae in a manner to be discussed later. The demonstration terminates when a step results in the formula γ. *Hilbert demonstration* is the name I propose for an exercise of this nature. A Hilbert demonstration is simply another name for common-garden mathematical proof. An implication of completeness (T17) is that there exists a Hilbert demonstration for any tautology. But T17 gives no clue on how to find that demonstration, other than suggesting that $\phi=\gamma$ should be restated in normal form. Hence if both ϕ and γ are hypothesized from the outset, it is usually easier to verify $\phi=\gamma$ by calculation. Demonstration and calculation are central to the *proof theory* (Bostock 1997: Part II; Sundholm 1983) of the pa. *LoF* does not mention proof theory.

Each step in a demonstration is justified by an annotation. If the demonstration is in column form, the annotations are written to the right of each step. If it is in linear form, the annotations are written after each step and enclosed in square brackets. If a demonstration step requires more than one consequence, these and their substitutions are listed sequentially, separated by semicolons. If a step includes a rearrangement of subformulae, that fact may be annotated by "OI".

If one of J0-C5 justifies a step, the annotation is as follows:

- When J0, J1, C1, C2, or C5 is invoked to delete a subformula or nested parentheses, the annotation for the current line states 'J0', etc. The deleted context is underlined in the *preceding* line;
- When J1, C1, C2, or C5 is invoked to add a subformula or nested parentheses, the added text is shown in bold in the *current* line;
- When J2 is invoked to shift a subformula from a greater depth to a lesser one, the shifted subformula is underlined in the *preceding* line;
- When J2 is invoked to shift a subformula from a lesser depth to a greater one, the shifted subformula is shown in bold in the *current* line.

While the justification for a step may invoke a PA axiom, J1 usually does the work of A2, and A1 is seldom necessary. If a step invokes one of C6-C9, the annotation may be more complicated, building on the fact that BA formulae can be taken as *schemata*, in which case they are stated using upper case letters. A generous reading of R2 allows us to replace any upper case statement letter by a subformula; e.g., C6 is assumed to take the form (A'B')(A'B) = A. Substitutions are notated as per the following example. If the subformulae α and β are substituted for A and B in C6, the annotation is 'C6, α/A, β/B', with the actual values of α and β written using lower case letters.

Time was, all demonstations were of the Hilbert variety, truth tables excepted. During the past 50-odd years, however, logicians (in contrast to mathematicians doing logic) seldom preach or practice Hilbert demonstrations. Instead, the reigning fashion was natural deduction and sequent calculi (both derived from the work of Gentzen in the 1930s), and is now refutation trees (Bostock 1997: §§4.1-4, 6.2, 7.4).

5.1. Variant Sets of Initials.

"*Any finite… selection of statements (preferably true ones, perhaps) is as much a set of postulates as any other. …'postulate' is significant only relative to an act of inquiry; we apply the word to a set of statements… to which we have seen fit to direct our attention.*" Quine (1982: 35)

5.1.1. C2 and J0 as Initials.

J1 and J2 are not the only possible pa initials, nor are they necessarily the most attractive. Because the pa is a Boolean algebra, J1, J2, and OI constitute a postulate set for Boolean algebra. Similarly, the many postulate sets proposed for Boolean algebra (Rudeanu 1963: chpt. 5) become possible sets of pa initials.

Bricken (1986) demonstrated J1 and J2 from C1-C3; hence C1-C3 can serve as initials. They appeal mainly because C1 and C2 alone suffice to justify most calculation steps. C1-C3 are very easily verified by any decision procedure, and C2 is much shorter than J2. Bricken (2002) came up with a more concise set of initials, by demonstrating C1 from C2 and the complement of C3, $(a())=\perp$, taken as an initial. He also replaced C2 with a notational variant of T13 in §3.1 above. I will henceforth refer to T13 as C2. Bricken's $(a())=\perp$ can be replaced with J0, because §A.1 derives J1 and J2 from J0 and C2. Hence J0, C2, and OI form a very economical basis for the CTV and **2**.

J0 and J1 govern the primitive value ().Table 5-1 shows how the initials J0,C2 can be seen as insertion/cancellation rules, with the implicit goal of making all variables vanish:

- *J0 Insert*. The same thing may be written on both sides of an empty boundary.
- *J0 Cancel.* If the *entire* content of a boundary is echoed in the pervasive space, both instances of that content may be erased, leaving an empty boundary.
- *C2 Insert*. Anything outside a boundary may be copied inside the boundary.
- *C2 Cancel*. If any *part* of the content of a boundary is echoed in the pervasive space, that part may be erased.[51]

Table 5-1. The pa in a Nutshell.

Initial	*Content of ()*	*Action*	*Notation*	*Antecedents*
OI	anything	Reorder the contents of a subspace at will.	*abc*=*bca*	Byrne (1946)
Let there be a boundary, and let a *appear outside it.*				
J0	*a*	$() \rightleftarrows (a)a$.	$(a)a = ()$	Natural deduction
C2	*ab*	All instances of *a* but the shallowest are redundant.	$a(b\langle a\rangle)=a(b\langle\perp//a\rangle)$	Rule of (De)Iteration in the Existential Graphs†
† See **2i-e** in Table 6-1.				

Now consider the three possible 2-1 partitions of *aab*. That C2 addresses the case *a*(*ab*) is immediate. J0 addresses (*a*)*ab*, once it is seen that (*a*)*ab*=(*ab*)*ab*=(). (The third case, (*aa*)*b*, equals (*a*)*b* by C5.) When a given subformula appears on both sides of a boundary, the interior instance is always redundant.

51. J0 can be seen as the sole axiom of natural deduction, and as akin to the rule for closing a branch in a tableau (cf. Bostock 1997: chpts. 4,6). C2 can be seen as a paired insertion and elimination rule for $\neg$ and $\vee$ (dually, $\neg$ and $\wedge$) of the sort that is typical of natural deduction. Likewise, A1 [A2] can be seen as a rule for () [$\perp$].

It should now be clear why pa demonstrations here and in *LoF* work J0–C2 very hard. Moreover, consequences beyond these are required in only a few situations. Demonstrating C9 requires C6; calculating it requires C7 as well. The proofs of T14-T18 invoke C9 twice and C7 once. The pa demonstrations in §§3.3 and 5.2 invoke C5 twice and C4 once.

Why did Spencer-Brown chose J2 as an initial? For one thing, J2 plays a key role in lattice theory (§3.3 above) and in Huntington's (1904) classic postulate set for Ba. Moreover, I know of no elementary demonstration of J2 from any postulates excluding J2. A happy consequence of deeming J2 an initial is that all *LoF* demonstrations, C1 and C9 excepted, are arguably trivial. While demonstrating J2 from J0,C2 is a bit involved, this is amply offset by an easy derivation of C1 and by the calculating power, to be demonstrated in §5.2, afforded by J0, C1, and C2 alone.

5.1.2. Some Postulate Sets for Ba.

Table 5-2 is not intended to be comprehensive; rather it includes a variety of axioms sets (bases) for which one or more of the following is the case: they are directly relevant to *LoF*, have been largely ignored in the literature, or otherwise interest me in some way. Prior (1962) includes a comprehensive review of CTV axiom sets proposed before 1960, especially the many sets proposed by Polish logicians. In the remainder of this paper, "*system m.n*" refers to axiom set *m.n* in Prior's Appendix I. (Epstein 1995: 407-9 is a more recent and limited survey.) Any pair of axioms asserting that a connective commutes and associates, I have replaced with OI. The "length" of a basis is the number of BA symbols required to express it. For other details of how I operationalize the notion of the "length" of a basis, see the Note to Table 5-2.

Johnson (*1892*). In a three-part article in *Mind*, the British logician W E Johnson set out a system whose syntax consisted merely of juxtaposed letters with overbars. The BA translation of this syntax is trivial: if α, β are formulae, $(\alpha\beta)$ translates Johnson's $\overline{\alpha\beta}$. He interpreted juxtaposition as conjunction, the overbar as complementation. His axioms were C1, C5, juxtaposition commutes and associates (tantamount to OI), and the Law of Dichotomy, Johnson's name for $(ab)(ab')=a'$, the contradual of Huntington's axiom. Since Dichotomy is C6 with a'/a, followed by C1,2x. I will refer to Dichotomy as C6. §A.4 includes a demonstration of J1, C1, C2, and C5 from C6 alone, thus proving that Johnson's axioms form a set of pa initials, and that C1 and C5 are redundant axioms. Johnson's system has had little impact; Prior (1962) is silent about it, even though Prior cites Johnson repeatedly. Johnson may have been ignored because his rambling discussion lacks the understanding of Ba and CTV that emerged later.

Huntington (*1904*). *LoF* rightly cited this paper, the wellspring of self-aware Boolean axiomatics. Huntington defined Boolean algebra as a set B with at least two members and closed under three operations, two binary and one unary. The binary operations are dual to each other, so that his remaining eight axioms are grouped into four dual pairs. The binary operations commute (B1), have distinct identity elements (B2), distribute over each other (B3), and have inverses defined in terms of the unary operation and the identity elements (B4) (Stoll 1974: §4.1; Eves 1990: 216, 257).

LoF reduces B1-B4 to J1,J2 as follows. Since the pa has both a primal and dual interpretation, four pa initials suffice for Huntington's four dual pairs of axioms. *LoF* 's tacit order irrelevance yields B1. The irrelevance of the unmarked state yields $a\perp=a$, one half of the dual pair B2. J1 and J2 yield B3 and B4. Interpreting Huntington's two binary operations as *ab* and ((*a*)(*b*)), BA likewise satisfies Huntington's axioms: B1, since OI implies *ab*=*ba* and ((*a*)(*b*))=((*b*)(*a*)); B2, since ()≠⊥, *a*⊥=*a*, and ((*a*)(())) = ((*a*)) [A2] = *a* [C1]; B3, by J2 and its dual; and B4, by J0 and J1.

Huntington (*1933, 1933a*) derived his 1904 Ba basis from the basis OI,C6, a nontrivial exercise. Like Johnson, he (1933) at first thought that C5 had to be a postulate, but very soon (1933a) proved otherwise. Kauffman (1990), using the pa, considerably simplified Huntington's result; see §A.4 for details. That J1,J2, J0,C2, and C6 each form a set of pa initials requires demonstrating each of J1,J2 ⊢ C6 (*LoF*), C6 ⊢ J0,C2 (§A.4), and J0,C2 ⊢ J1,J2 (§A.1).

Robbins conjectured that C6 could be replaced by its dual (*4.43 in *PM*). That OI and the dual of C6 constitute a Ba basis eluded proof until McCune (1997) brought powerful automated theorem proving techniques to bear.

Wolfram. Employing computer-intensive search methods he devised, Wolfram (2002: 773, 808, 1151, 1175) found two new single axioms for Ba based on the Sheffer stroke, one of which is (((*bc*)*a*)(*b*((*ba*)*b*))) = *a*.[53] Wolfram's axioms are each 21 symbols long, and are the shortest known single axiom for Ba/CTV, whether based on the Sheffer stroke (system 6.4) or not (systems 1.5, 3.13, 6.14). Wolfram (p. 808) also believes that his single axiom is the shortest known Ba basis, when in fact it is no shorter than either the Robbins-McCune basis, or Bricken's (2002) basis with ()*a*=() replacing (()*a*)=⊥. The shortest known Ba basis is in fact one Wolfram discovered but deemed an also-ran: ((*ab*)(*a*(*bc*)))=*a* and *ab*=*ba*, only 20 symbols long. This basis is the shortest one because the parentheses in the literal translation of the second axiom, (*ab*)=(*ba*), are eliminable, economizing four symbols.

As is always the case with single axiom bases for algebras, Wolfram's single axiom makes for very difficult proofs of simple results. For instance, deriving (pp. 810-11) Sheffer's (1913) Ba axioms from Wolfram's single axiom shown above requires 343 steps, 81 lemmas, and expressions with as many as 128 operators. 42 lemmas are required merely to prove that the Sheffer stroke commutes! Wolfram also ran the following computer "horse race" (p. 1175). For each of eight bases (p. 808), he counted how many steps were required to prove, using proprietary software, each of 582 consequences, each containing no more than 2 variables and 6 instances of the Sheffer stroke. A graphical summary of the results reveals large differences across the eight bases in the number of proof steps required. Sheffer's 1913 basis and Wolfram's ((*ab*)(*a*(*bc*)))=*a*, *ab*=*ba* were more or less tied for fewest proof steps, averaged over the 582 consequences. The Robbins-McCune basis did poorly; its dual, Huntington's (1933a) basis, was not a contestant. Wolfram's two single axiom bases faired the worst of all.

---

53. *Verification*. Let $a|b \Leftrightarrow (ab)$. *Dem*. $(((bc)a)(b((\underline{b}a)\underline{b}))) = (((bc)a)(b\underline{((}a\underline{))}))$ [C2,2x] $= (((bc)\underline{a})(b\underline{a}))$ [C1] $= (\underline{((}bc\underline{))}b')a$ [J2] $= \underline{(b'bc)}a$ [C1; OI] $= \boldsymbol{a}$ [J1].□

Table 5-2. Selected CTV/Ba Axioms, pa Initials, Reexpressed in pa Notation.					
Year	*Author*		*Axioms/Initials*	*Diff.*	*Length*
1885	Peirce	CTV	*a′ab*, C3, ((*a′b*)*a*)*a*, *Syll*, (*a′b′c*)*b′a′c*	3	56
1917	Nicod	CTV	J0, (*a*(*b*))(*cb*)*ac*	4	20
1924	Lukasiewicz-Bernays	CTV	*a′ab*, (*aa*)*a*, (*a′b*)(*ca*)*cb*, (*ab*)*ba*	4	36
1929	Lukasiewicz	CTV	*a′ab*, (*aa*)*a*, (*a′b*)(*b′c*)*a′c* (*Syll*)	4	32
1942	Rosser[54]	CTV	*a′ab*, (*aa*)*a*, (*ab′*)(*bc*)*ca*	4	28
1948	Lukasiewicz-Wajsberg	CTV	C3, ((*a′b*)*r*)(*r′a*)*s′a*	5	25
1956	Church	CTV	PC1, PC2, ((*a′*⊥)⊥)*a*	4	42
1964	Mendelson	CTV	PC1, PC2, (*a′b*)(*ab*)*b*	4	43
1996	Machover	CTV	PC1, PC2, J0, ((*a′b*)*a*)*a*, (*a′a′*)*a′*	2	61
1892	Johnson	na	C1, C5, contradual of C6, OI	2	32
1904	Huntington	Ba	J1, J2, *a*⊥=*a*, *ab*=*ba*	3	37
1913	Sheffer	Ba	C1, *a*(*b′b*)=*a*, *a*(*bc*)=((*b′a*)(*c′a*))	3	36
1933	Huntington	Ba	C6, OI	3	23
1933	Robbins-McCune	Ba	C6 (dual), OI	6	21
1934	Bernstein[55]	Ba	C6 (dual), ((*c′a*)(*b′a*))=*a*(*bc*)	4	34
1969	Abbott	Ba	C3, C4, *ab*=*ba*, (*ab*)*c*=(*ba*)*c*	3	31
1969	*LoF*	pa	J1, J2, OI	1	35
1986	Bricken	pa	J0, C1, C2, OI	1	30
2002	Wolfram	Ba	((*b*((*ab*)*b*))(*a*(*cb*)))=*a*	6	21
"	"	Ba	((*ab*)(*a*(*bc*)))=*a*, *ab*=*ba*	5	20
2002	Bricken	pa	C3, C2, OI	1	21

Note. PC1, PC2 are defined in §5.2.
Length = Number of BA symbols required to state the axioms in BA notation. A primed variable counts as 3 symbols, '()' as 1. (*ab*), not *a′b′*, translates the Sheffer stroke. If an axiom has the form (ϕ)=() or ⊥, or (ϕ)=(γ), I omit the outermost parentheses. A CTV axiom is treated as an equation ending in '=()', adding 2 to the length. I have replaced Ba axioms asserting commutativity and associativity with OI, *abc*=*bca*, and added OI to every pa initial set. However, no basis as published includes OI.
Diff.: A subjective assessment of the ingenuity required to derive J1-C6 in *LoF* from the given basis, with 6 requiring the most ingenuity.

5.1.3. Some CTV Axioms.

A set of pa initials can serve as CTV axioms and vice versa. CTV axioms take the form 'α→β' or can be re-expressed as such. pa initials are of the form 'α=β', and are easier to work with, especially for those whose mathematical habits are those of elementary algebra. The distinction is not essential, however, because an axiom of the form α→β is equivalent to the equation (α)β=().

54. Eves 1990: 256, L′; system 6.3. The translation invokes the dual interpretation, as Rosser's primitive connective is '∧'.

55. Bernstein (1934: 880). *LoF* (p. 107) asserts, without proof or citation, that Bernstein's second axiom and (*a′*(*b′b*))=*a* form a Ba basis.

CTV axiom sets either explicitly state that one of ∧ or ∨ commutes (e.g., *PM*), or are designed in such a way that this can be demonstrated. But even if *ab*=*ba* or the like were to be added to the three sets of pa initials proposed above, no CTV axiom set in Epstein (1995) or Prior (1962) would correspond to any set of Ba basis or pa initials shown in Table 5-2. Logicians, evidently, are not in the habit of reading the Boolean algebra literature. Boolean algebraists are not guilty of the reciprocal sin; see the references in Huntington (1933) and Bernstein (1934). The Ba bases (those of Wolfram and Bernstein excepted), and pa initials seem simpler and more intuitive than the CTV axiom sets.

Nicod (1917: 34) proposed a two axiom basis for sentential logic, formulated using only the Sheffer stroke, read as NAND. The longer of these axioms is more easily understood when re-expressed using the conditional as well as the stroke (*PM*, p. xviii). Invoking the dual interpretation, so that *a* | *b* ⇔ (*ab*) and *a*→*b* ⇔ (*a*(*b*)), and treating any outermost parentheses as redundant, Nicod's axioms are (*a*)*a* and (*a*(*b*))(*cb*)*ac*, the shortest (20) CTV basis in Table 5-2. The substitutions *c*/*a* and (*a*)/*c* reveal the latter axiom to be an instance of *Syll*, (*a'b*)(*b'c*)*a'c*. Nicod then condensed his two axioms into one (*PM*, p. xix) and proved that the latter is a CTV basis by deriving the axioms of *PM* and L from it and a variant of *modus ponens*. Lukasiewicz later simplified Nicod's single axiom into something (not shown in Table 5-2) whose dual pa translation has length 23. While this basis is well-known (system 6.4; Quine 1982: 87), like all single axiom bases, it is of doubtful perspicacity.

Lukasiewicz (system 1.4a; Quine 1982: 85) proposed in 1929 an axiom set that became well-known and that I will call L. L has a straightforward pa interpretation. (*a*$\underline{a}$)*a* is one half of C5. *a'ab* is at once C3 and J0, thinly disguised, and the other half of C5, given *a*=*b*. *Syll* asserts the transitivity of the conditional and the validity of the syllogism in Barbara, that hoary chestnut of traditional logic. *Syll* with *a*=() yields another hoary chestnut, *modus ponens*. Nicod's two axiom set, just discussed, is, in effect, L with (*a*)*ab* omitted. (Note also the similarity of Nicod's basis to Rosser's.) Hence a slight modification of *Syll* renders (*a*)*ab* redundant, a fact apparently heretofore unnoticed.

Lukasiewicz and Bernays, working independently, earlier proposed a set of axioms with (*a'b*)(*ca*)*cb* and (*ab*)*ba* instead of *Syll*, yielding a revised version (system 6.11) of the truth-functional axioms of *PM*.[56] The literature is silent about the similarity between Rosser and Lukasiewicz-Bernays, as well as about how L and Rosser render *PM*'s (*ab*)*ba* redundant. In 1948, Lukasiewicz proved that ((*a'b*)*r*)(*r'a*)*s'a* is the shortest possible single axiom from which all formulae involving '→' alone can be demonstrated (system 2.15d). Wajsberg (system 3.12) had shown in 1937 that adding ()*a* (i.e., C3) and ¬*a* $=_{\mathbf{df}}$ *a*→F to any axiom system adequate for '→' alone results in a CTV axiom system.

5.2. Proof: Worked Examples.

"*…standard university logic problems, which the calculus published in this text renders so easy that we need not trouble ourselves further with them...*" *LoF*, p. viii.

56. For a (nontrivial) demonstration of J2 from these axioms, see Halmos & Givant (1998: 37-8).

Example 1. I now calculate the three standard axioms for the CTV (Bostock 1997: 194).[57]

PC1: $a \to [b \to a]$

Cal.	$a'b'a$	transcription
	$(\boldsymbol{b'}a)b'a$	C2; OI
	$()$	J0. □

PC2: $[a \to [b \to c]] \to [[a \to b] \to [a \to c]]$

Cal.	$(a'b'c)(\underline{a'}b)a'c$	transcription
	$(a'b'c)a'(b)c$	C2; OI
	$()$	J0. □

PC3: $[\neg a \to \neg b] \to [b \to a]$

Cal.	$(\underline{(}a'\underline{)}b')b'a$	transcription
	$(ab')ab'$	C1; OI
	$()$	J0. □

These pa calculations reveal that PC1 and PC2 are C2 in another guise; ditto for PC3 and C1. All three calculations invoke J0 in the final step. PC1-PC3 neither are, nor claim to be, "obvious" and "elementary." Rather, PC1 and PC2 are taken as axioms mainly to facilitate proving the Deduction Theorem (Machover 1996: 7.7.3; also see Table 5-4 below). PC3 is one of many possible axioms governing denial. J1 fills that role in *LoF*; C1 is arguably the simplest.

Example 2. I now present the standard demonstration (e.g., Bostock 1997: 200) of $a \to a$, which is simply J0 in the pa. "MP" abbreviates *modus ponens*; 'MP *m,n*' means "infer β because line *n* asserts α, and line *m* has the form α→β," with *m* and *n* both being less than the number of the line where 'MP *m,n*' appears.

5.2.2. Theorem. $a \to a$

Standard Demonstration.

1. $[a \to ((a \to a) \to a)] \to [(a \to (a \to a)) \to (a \to a)]$	PC2, $a \to a$/A, a/B, a/C
2. $a \to ((a \to a) \to a)$	PC1, $a \to a$/B
3. $(a \to (a \to a)) \to (a \to a)$	MP 1,2
4. $a \to (a \to a)$	PC1, a/B
5. $a \to a$	MP 3,4. □

Example 3. I now calculate four more involved tautologies taken from standard texts. The first two are from Nolt et al (1998: 4.46, 109).

$(p \to q) \leftrightarrow \neg(p \wedge \neg q)$

Dem.	$(p)q = \underline{((}(p)\underline{((}q\underline{))))}$	transcription
	$(p)q = (p)q$	C1, 2x. □

57. PC1-PC3 is Church's (1956: 119) $\mathbf{P_2}$ and Eves's (1990: 256) L″. Eves, citing no one, attributes L″ to Lukasiewicz.

I have taken the liberty of translating '↔' as '='. To someone experienced in the pa, the tautological equivalence of $(p\rightarrow q)\leftrightarrow\neg(p\wedge\neg q)$ is evident at a glance. The next tautology is in the form a clause, a type of argument defined in §5.4 below:

$\neg s\leftrightarrow(\neg p\vee\neg v),\ v\wedge p\vdash s$	From the conjunction of everything to the left of '⊢', infer the alternation of everything to the right of '⊢'.
$(((((sp'v')((p'v')s')))((v'p'))))s$	transcription
$(sp'v')((p'v')s')v'p's$	C1, 3x
$(sp'v')sp'v'((p'v')s')$	OI
$()$	J0. □

I chose the two preceding examples because the corresponding demonstrations in Nolt et al are 18 and 21 lines long, respectively, the longest purely sentential proofs in that text. The next two examples are from Kalish et al (1980: 417, 66f).

$(a\rightarrow b)\rightarrow[(a\wedge b)\leftrightarrow a)]$

$(a'b)((((a'b'))a)(a'(a'b')))$	transcription
$(a'b)((a'ab')(a'(a'b')))$	C1; OI
$(a'b)((a'(a'b')))$	J1
$(a'b)((a'(b')))$	C2
$(a'b)((a'b))$	C1
$()$	J0. □

$[(\neg a\rightarrow r)\wedge(b\rightarrow r)]\leftrightarrow[(a\rightarrow b)\rightarrow r]$

$((((ar)(b'r)))((a'b)r))(((ar)(b'r))(a'b)r)$	
$((ar)(b'r)((a'b)r))((a'(b'))(a'b)r)$	C1; C2,2x
$((ar)(b'r)((a'b)r))((a'b)(a'b)r)$	C1
$((ar)(b'r)((a'b)r))((a'b)r)$	C5
$((ar)(b'r))((a'b)r)$	C2
$\mathbf{r}((a'b))(a'b)r$	J2
$()$	C1; J0. □

I chose these examples because Kalish et al require 32 and 27 lines, respectively, to verify them. The demonstration of the one on the left fills an entire page and is preceded by five pages of discussion. The demonstrations in Nolt et al and Kalish et al are not reproduced here because they require 102 lines in all and invoke natural deduction techniques that are beyond the scope of this paper, typographically as well as logically. The above four pa calculations require a mere 22 steps, only two of which invoke something other than J0-C2 and order irrelevance.

The next example is from MacKay (1989: exercise 9m.4) and involves determining the satisfiability of $((p\leftrightarrow\neg q)\leftrightarrow\neg p)\leftrightarrow\neg q$. To avoid working with a single very long formula, I break up the rightmost biconditional into its constituent conditionals. I also translate $\alpha\leftrightarrow\neg\beta$ as $((\alpha'\beta')(\alpha\beta))$.

$((p\leftrightarrow\neg q)\leftrightarrow\neg p)\rightarrow\neg q$

$(((p'q')(pq)p)(((p'q')(pq))p'))q'$	transcription
$(((p')(q)p)(((p'q')\mathbf{p'q'}(pq))p'))q'$	C2,4x
$(((p')p)(p'))q'$	J1; C2
$((p)(p'))q'$	C2
q'	J1

$\neg q\rightarrow((p\leftrightarrow\neg q)\leftrightarrow\neg p)$

$q((p'q')(pq)p)(((p'q')(pq))p')$	
$q((p'q')(pq)p\mathbf{q})(((p'q'\mathbf{q})(p))p')$	C2,3x
$q(((p))p')$	J1,2x
q	J1. □

As one conditional simplifies to q' and the other to q, their conjunction evaluates to ⊥ by C1 and J1. While this calculation is a bit involved (14 steps), mainly because '↔' lacks a concise pa representation, it requires only two consequences: J1 and C2. By contrast, MacKay's (pp. 368-69) proof is 43 lines long and invokes 11 natural deduction rules.

Example 4. Quine (1982: 69) introduces the DNF as a method for determining satisfiability, and builds his exposition of the DNF around a six page discussion of the formula (1), one he deems "forbidding":

(1) $\neg(((p\rightarrow(\neg s\wedge q))\rightarrow\neg((s\wedge q)\rightarrow p))\wedge\neg(\neg(r\wedge p)\wedge\neg(p\rightarrow s)))$.

I now show how to employ the pa to determine the satisfiability of (1). Because of the prevalence of conjunction in (1), I take the unusual step of translating it using the dual interpretation of the pa, as in that interpretation, $x\wedge y$ translates as xy instead of $(x'y')$.

$(((p(s'q))\underline{((}(sq)p'\underline{))})((rp)(ps')))$	transcription
$(((p(s'q))(sq)p')\chi)$	C1; let $\chi=((rp)(ps'))$
$((\underline{(\boldsymbol{p}'p(s'q))}(sq)p')\chi)$	C2; OI
$(((sq)p')\chi)$	J0
$(((sq)p')((r\underline{p})(\underline{p}s')))$	Expand χ
$(((sq)p')\boldsymbol{p}(r'\underline{(}s'\underline{)}))$	J2
$((\underline{p'\boldsymbol{p}(sq)})p(r's))$	C2; OI; C1; OI
$(p(sr'))$	J0. □

Conclusion: (1) is satisfied when $p\rightarrow s\rightarrow r$ is the case. Note how this technique easily reveals the irrelevance of q, even of all of (1) to the left of the third '$\wedge$'.

Example 5. The following two examples are taken from texts with a contemporary following. Hurley (2000: 415, exercise 19) asks students to verify the clause:

$a\rightarrow(nn')\rightarrow s\vee t,\ t\rightarrow(f\wedge\neg f)\ \therefore\ a\rightarrow s.$

$(a'\underline{(nn')}st)(t'\underline{(ff')})a's$	transcription
$(a'st)\underline{(t')}a's$	J1,2x
$(a'st)a'st$	C1; OI
$()$	J0. □

Hurley's natural deduction proof (p. 653) requires 19 steps and invokes 13 rules.

Lepore's (2003: 131) exercise 8.5.2 asks whether $((p\wedge q\wedge k)\vee\neg r)$ and $(r\rightarrow(\neg q\rightarrow (p\wedge\neg(v\vee\neg j))))$ are equivalent. I follow Lepore in breaking up the problem into two conditionals and then calculating each:

$((p\wedge q\wedge k)\vee\neg r)\rightarrow(r\rightarrow(\neg q\rightarrow(p\wedge\neg(v\vee\neg j))))$

$((p'q'k')\underline{r'})r'q(p'vj')$	transcription
$\underline{(}(p'q'k')\underline{)}r'q(p'vj')$	C2
$\underline{q'q}p'k'r'(p'vj')$	C1; OI
$()$	J0.

$(r\rightarrow(\neg q\rightarrow(p\wedge\neg(v\vee\neg j))))\rightarrow((p\wedge q\wedge k)\vee\neg r)$

$(\underline{r'}q(p'vj'))(p'q'k')r'$	transcription
$(q(\underline{p'}vj'))(\underline{p'}q'k')r'$	C2
$(q(vj'))(q'k')p'r$	J2.
$(qv')(qj)(q'k')p'r$	C7.

The conditional on the right cannot be simplified any further. Hence the two halves of the biconditional do not simplify to the same formula, and the two statements are not equivalent. Note the use of C7 on the right to obtain the NF, which is more nakedly revealing of the inability to proceed further. Lepore's worked answer using refutation trees is 25 lines long, invokes 6 rules, and fills all of his p. 389.

The most spectacular example of this nature I have saved for last. Leblanc and Wisdom's (1976: 395) proof of $[p\vee(q\rightarrow r)]\leftrightarrow[(p\vee q)\rightarrow(p\vee r)]$ is 42 lines long and invokes eight natural deduction rules. Translating the single instance of '↔' as '=', the pa demonstration is utterly trivial: *Dem.* $pq'r = q'pr$ [OI] = $(pq)pr$ [C2]. □
End of Examples

The 12 examples above suggest that pa calculations are much easier than conventional proofs. The enormous gain in simplicity largely stems from working J0-C2 very hard. That the pa accomplishes so much with so little reveals that in practice, the pa is more than merely a new notation for the CTV and **2**.[58]

5.3. Truth Value Analysis.
A different proof procedure, very much in the spirit of the PA, follows from T16: *If formulae are equivalent in every case of one variable, they are equivalent, and conversely*. Let a be a variable and $f\langle a\rangle$ and $g\langle a\rangle$ be formulae containing a. Let $f\langle a\text{=()}\rangle$ mean replacing every instance of a by (), and so on. If $f\langle a\text{=()}\rangle = g\langle a\text{=()}\rangle$ and $f\langle a=\perp\rangle = g\langle a=\perp\rangle$, T16 concludes that $f=g$, regardless of the values of any other variables they may contain. Hence $f=g$ is a tautological equivalence.

Consider the following algorithm for determining the satisfiability of f. Evaluate $f\langle a\text{=()}\rangle$ and $f\langle a=\perp\rangle$; let these be two *branches*. Then note the following facts:

- Setting an unprimed [primed] variable to ⊥ [()] makes the variable vanish;
- Setting an unprimed [primed] variable to () [⊥] results in ();
- If both branches feature the same formula, they terminate;
- If a branch simplifies to () or ⊥, it terminates.

At any stage, a branch may be simplified by invoking a consequence; in this regard, J0, J1, C1, and C3 are especially useful. If a recognisable tautology emerges in any branch, set that branch to () or ⊥ and do not explore it any further. Repeat this procedure, each time selecting the remaining variable with the most instances so as to economize work, so as to form a tree. The algorithm terminates when all branches have terminated.

If all branches of the tree terminate with the same formula, the overall formula is a tautology. If the branches terminate in a mixture of () and ⊥, the formula is satisfiable, with the pattern of () and ⊥ indicating the satisfying atomic valuations. This algorithm sufficiently resembles Quine's (1982: §5) *truth value analysis* that I will appropriate the name, henceforth abbreviating it to TVA.[59]

---

58. I invite the reader to compare the demonstrations in *LoF* and here with those in Nidditch (1962), a book comparable to *LoF* in size and time of writing, also intended for undergraduate instruction, but more conventional in approach. Deferring to intuitionism, Nidditch posits 11 algebraic axioms and the rule *modus ponens*, then proves 4 lemmas and 58 theorems (a category that lumps together what are here called (meta)theorems and consequences).

59. TVA first saw the light of day in the 1950 first edition of Quine (1982). I owe my discovery of TVA to Bostock's (1997: §2.11) elegant treatment thereof, to my knowledge unique among contemporary texts. Prior (1962: 17) includes a fine example of a TVA proof in tree form. N.B. "Truth value analysis" in Kalish et al (1980: §§II.8-9) is an unrelated concept.

Fig. 2 gives, by way of example, a TVA proof of Leibniz's *Praeclarum Theorema*, $[(p\rightarrow r)\wedge(q\rightarrow s)]\rightarrow[(p\wedge q)\rightarrow(r\wedge s)]$:

Fig. 2.
Verifying Leibniz's *Praeclarum Theorema* via Truth Value Analysis.

		(((*p′r*)(*q′s*)))((*p′q′*))(*r′s′*)	
p	(((*r*)(*q′s*)))((*q′*))(*r′s′*)		(((()*r*)(*q′s*)))((()*q′*))(*r′s′*)
q	((*r′s′*))(())(*r′s′*)	(((*r*)(()*s*)))((()))(*r′s′*)	(((()*r*)(*q′s*)))()(*r′s′*) [C3; A2]
	((*r′s′*))(*r′s′*) [A2]	(((*r*)(()*s*)))()(*r′s′*) [A2]	() [C3]
	() [J0]	() [C3]	□

Section 6 below will demonstrate the *Theorema* in three ways.

Verifying both α′β and β′α by TVA amounts to a TVA verification of the equation α=β. It may be easier in practice to subject each of α and β to a separate TVA. Iterate the TVA until the set of formulae terminating the brances of α is the same as the set terminating the corresponding branches of β, in which case the equation if verified. In all other cases, the equation does not hold.

Given a formula with *n* distinct variables, the construction of the corresponding truth table requires evaluating 2^n PA formulae. If *n* does not exceed three or four, a truth table is mechanical but not impractical. Also, by virtue of T3, a truth table and a pa demonstration must yield the same result. But thanks to TVA, any such tedious resort to brute force is unnecessary. One round of the above algorithm, applied to the variable with the most instances, often suffices.[60]

All CTV tautologies can be verified by TVA. Furthermore, the axioms of the CTV are a small subset of the set of all tautologies. From these undisputed facts, Quine (1951: *100; 1982: §13), invoking Herbrand, argues that all CTV tautologies are equally deserving of the honorific title of axiom. (Smullyan 1968: 81, Wolf 1998: 79, and Cori & Lascar 2000: §4.1.1, all concur.) This egalitarian view, however commendable, fails to distinguish the *validation* of a given tautology, for which decision procedures are adequate, from the *discovery* of a new tautology, a process typically requiring inspired trial and error. And a Hilbert proof ultimately requires some unproved assertions, i.e., axioms. We may, if we wish, make those axioms as economical as possible.[61]

5.4. Boundary Logic and the Usual Inference Rules.

---

60. Ascertaining the satisfiability of a formula with *n* distinct variables via truth tables requires evaluating 2^n interpretations. Hence the truth table decision procedure for CTV satisfiability is said to require *exponential time*. Whether there exists a decision procedure for satisfiability that is merely a polynomial function of *n*, (i.e., a procedure executable in *polynomial time*) is a major unsolved problem in computational mathematics; see Hodges (2001: 23-24) and references cited therein. While I submit that TVA is quicker and easier than truth tables, especially when *n* is not large, I cannot claim that executing TVA on a computer would require less than exponential time for any *n*.

61. For a defense of Hilbert proof when a decision procedure is available, see Epstein (1995: §II.K.1).

The only inference rules mentioned thus far have been substitution and replacement, R1 and R2, which suffice for equational logics. First order logic typically includes the truth-functional inference rule *modus ponens* (also known as the *rule of detachment*): from α and $\alpha \rightarrow \beta$, infer β.

All truth functional inference rules other than R1 and R2 can be seen as special cases of a single inference rule Wolf (1998: §§3.5, 4.2) calls *propositional consequence*, PC. Let $\phi_1 \ldots \phi_n$ be *premises*, and χ be a *conclusion.* The premises and conclusion must be statements. PC asserts that an argument from the premises to the conclusion takes the form of a *clause* (Cori & Lascar 2000: §§1.3.2, 4.4.1), a formula consisting of the conjoined premises linked to the conclusion via the conditional. If a clause is a tautology, the conclusion χ follows from the premises and the clause (as well as the associated argument) is *valid*. Hence PC can be stated as:

(1) $\phi_1, \ldots, \phi_n \vdash \chi \Leftrightarrow \vdash [\phi_1 \wedge \ldots \wedge \phi_n] \rightarrow \chi \Leftrightarrow \underline{(}\underline{(}(\phi_1) \ldots (\phi_n)\underline{)}\underline{)}\chi = (\phi_1) \ldots (\phi_n)\chi$ [C1] $= ()$.

A clause is the concatenation of all the premises with the conclusion, with the premises enclosed. While I have tacitly assumed that a clause contains but one conclusion, doing so results in no loss of generality: a clause can contain multiple conclusions, all juxtaposed.

Ascertain the satisfiability of a clause as follows:
- Translate every premise and every conclusion into the pa;
- Enclose each premise, then concatenate the premises and conclusions;
- Invoke C5 to erase all duplicate instances of a subformula within a given subspace;
- Erase all nested parentheses redundant by virtue of C1;
- Work C2 relentlessly to erase redundant instances of variables at different depths;
- Invoke J1 to erase any subformulae of the form $(\alpha'\alpha)$.

If the end result is a primitive value, the clause or its negation is always valid. If the end result is a formula, the clause is valid under those atomic valuations that satisfy that formula.

Table 5-3 shows several inference rules whose clausal representations give rise to tautologies. Greek letters are metalogical, and stand for anything that can be assigned a truth value.

Table 5-3.
Some Common Instances of Propositional Consequence.

Name	ϕ_1	ϕ_2	ϕ_3	χ	Source
Contrapositive	α'	β		$\alpha'\beta$	88
Conjunction	α	β		$(\alpha'\beta')$	88
modus ponens	α	$\alpha'\beta$		β	79
modus tollens	$\alpha'\beta$	β'		α'	88
Biconditional	$\alpha'\beta$	$\beta'\alpha$		$((\alpha'\beta)(\beta'\alpha))$	85
Syllogism	$\alpha^*\beta$	$\beta'\gamma^*$		$\alpha^*\gamma^*$	na
Proof by cases	$\alpha\beta$	$\alpha'\gamma$	$\beta'\gamma$	γ	84

Source: Corresponding page number in Wolf (1998).

The Syllogism: A Digression

Following a tradition dating from the foundation of logic in ancient Greece, restrict each premise and conclusion to one of the forms "[All/Some] *A* are [Not] *B*." The pa translation thereof is as follows: 'all α are β′ is α′β, 'all α are not β′ is α′β′, 'some α are β′ is (α′β′), and 'some α are not β′ is (α′β). "All non-α are β" is αβ, and similarly for the other forms. α and β can be read as metavariables standing for nonempty *sets*, in which case 'all α are β′ may be seen as shorthand for 'all members of set α are also members of set β′, and so on. Let a '*' after a variable stand for a prime, or nothing. A *syllogism* is a clause consisting of two premises and a conclusion, each having one of the forms just described. Medieval tradition assigns to the syllogism (α′β)(β′γ)α′γ the name 'Barbara.'

LoF 's Appendix II concludes that all *valid syllogisms* are variants of Barbara, and hence share a common simple pa structure, but the details of the argument are hard to follow (especially the one culminating in 'Interpretative theorem 2′). If Barbara is valid,[62] then so is (α*β)(β′γ*)α*γ*=(), giving rise to four permutations. Moreover, the validity of this clause does not depend on the order of its three molecular components, giving rise to another 3×2=6 possible permutations, and a total of 6×4=24 permutations. There are indeed 24 valid syllogisms, although the literature, e.g., Quine 1982: §16, is not as unequivocal about this fact as could be desired. At any rate, the pa trivializes what had been for more than 2000 years a rather involved subject. On previous attempts to algebraize the syllogism, see Prior (1962: systems 10.11-6). None of these systems reveal the simplicity and generality of (α*β)(β′γ*)α*γ*, or that manipulating syllogisms requires nothing beyond Boolean algebra.

End of Digression

Table 5-4 presents the usual inference rules of contemporary logic, taken from Machover (1996) and Bostock (1997), along with their boundary justifications. The boundary translation of the syntactic and semantic turnstiles is: prime all objects to the left of the turnstile, then concatenate everything on both sides. Given this translation, the inference rules shown are all trivial pa consequences.

BA allows the molecular subformulae making up a clause to be permuted at will. Since all partitions of the molecular subformulae into premises and conclusions are equivalent; the turnstiles are order irrelevant objects. In particular, (ϕ_n) and ⊢ in (1) can be transposed; the result is the boundary logic equivalent of the Deduction Theorem. Moreover, the validity of a clause does not depend on whether any particular molecular subformula is included among the premises or the conclusion, as long as any formula moved from one side of the turnstile to the other is first enclosed. It should now be clear how and why boundary logic dispenses with all turnstiles.

BS and the Inconsistency Effect are all J0 in another guise; ditto for INT and OI, CON and C5, and Indirect Proof and C1. The Cut Rule is the only rule whose demonstration invokes C2. Its meaning is simpler than may appear: if both φ and φ′ appear in the premises, φ is irrelevant to the conclusion.[63]

---

62. *Cal*. (<u>*a*</u>′*b*)(*b*′<u>*c*</u>)*a*′*c* = (*b*)(*b*′)*a*′*c* [C2,2x] = (*b*′***a*′*c***)*b*′*a*′*c* [C2; OI] = () [J0]. □ Note that validity requires that *b*, the variable not appearing in the conclusion, appear twice in the premises, once primed and once not.

63. This is the unproved Interpretive Theorem 1 in *LoF*, p. 123. The associated discussion does not mention the Cut Rule.

Table 5-4. Some Common Logical Rules and Their Boundary Derivations.			
Name	*Formal Statement*	pa *Derivation*	*Source*†
Bostock's (1997) *Structural Rules*			
Basic Sequents‡	Γ,φ ⊨ φ,Δ	$\Gamma'\underline{\varphi'\varphi}\Delta = (\Gamma'\varphi\Delta)\Gamma'\varphi\Delta$ [C2,2x] = () [J0].	p. 285
INTerchange, L	[Γ,φ,ψ,Δ⊨Θ] → [Γ,ψ,φ,Δ⊨Θ]	(Γ′φ′ψ′Δ′Θ)Γ′ψ′φ′Δ′Θ = () [OI; J0].	§7.1
" , R	[Γ⊨Δ,φ,ψ,Θ]→[Γ⊨Δ,ψ,φ,Θ]	(Γ′ΔφψΘ)Γ′ΔψφΘ =() [OI; J0]	"
CONtraction, L	[Γ,φ,φ ⊨ Δ] → [Γ,φ ⊨ Δ]	$(\Gamma'\underline{\varphi'}\varphi'\Delta)\Gamma'\varphi'\Delta = ()$ [C5; J0].	"
" , R	[Γ ⊨ φ,φΔ] → [Γ ⊨ φ,Δ]	$(\Gamma'\varphi'\Delta)\Gamma'\underline{\varphi'}\varphi'\Delta = ()$ [C5; J0].	"
CUT	[Γ ⊨ φ,Δ] ∧ [Φ,φ ⊨ Θ] → [Γ,Φ ⊨ Δ,Θ]	$\underline{((}(\underline{\Gamma'}\varphi\underline{\Delta})(\underline{\Phi'}\varphi'\underline{\Theta})\underline{))}\Gamma'\Phi'\Delta\Theta = \underline{(\varphi)(\varphi')}\Gamma'\Phi'\Delta\Theta$ [C1; C2,4x] = () [J0].	§2.5.C
Machover's (1996) *Inference Rules*			
Indirect proof, *reductio*	Γ,¬α ⊢ ⊥ ↔ Γ ⊢ α	$\Gamma'\underline{((}\alpha\underline{))}\bot = \Gamma'\alpha$ [C1].	§7.8.9, 15
Deduction Theorem	Γ,α ⊢ β ↔ Γ ⊢ α→β	Γ′α′β = Γ′α′β.	§7.7.2
Inconsistency Effect	[Γ ⊢ ⊥] ⊢ [Γ ⊢ β]	(Γ′⊥)Γ′β = () [A2; J0].	§7.8.6
† Section of Bostock (1997) or Machover (1996) where the rule in question is introduced and discussed. The formal statements are from Bostock (1997: 385). ‡ This replaces Bostock's (§2.5) ASSumptions, and THINning from the left, right. **Note**: L=left; R=right. A lower [upper] case Greek letter denotes a single formula [set of formulae]. A primed upper case letter signifies that each constituent formula is primed.			

I now show by example how the pa simplifies clausal reasoning by reworking Stoll's (1963: 184) Example 4.4.3, reproduced in Table 5-5. The premises are in the third column from the left, in rows where 'p' alone appears in the rightmost column. The conclusion is at the bottom of the third column. 't' in the rightmost column means that an unspecified tautology has been invoked.

Table 5-5. Stoll's (1963) Example 4.4.3.			
{1}	1	¬*C*∧¬*U*	p
{1}	2	¬*U*	1 t
{3}	3	*S*→*U*	p
{1,3}	4	¬*S*	2,3 t
{1}	5	¬*C*	1 t
{1,3}	6	¬*C*∧¬*S*	4,5 t
{1,3}	7	¬(*C*∨*S*)	6 t
{8}	8	(*W*∨*P*)→*I*	p
{9}	9	*I*→(*C*∨*S*)	p
{8,9}	10	(*W*∨*P*)→(*C*∨*S*)	8,9 t
{1,3,8,9}	11	¬(*W*∨*P*)	7,10 t

{1,3,8,9}	12	$\neg W \wedge \neg P$	11 t
{1,3,8,9}	13	$\neg W$	12 t

A reader unversed in natural deduction need take away from Table 5-5 only the comparative opacity of its content. A pa calculation to the same effect follows:

Premises: $(CU) \Leftrightarrow \neg C \wedge \neg U$ *Conclusion*: $(W) \Leftrightarrow \neg W$

$(S)U \Leftrightarrow S \rightarrow U$

$(WP)I \Leftrightarrow (W \vee P) \rightarrow I$

$(I)CS \Leftrightarrow I \rightarrow (C \vee S)$

$\underline{((}CU\underline{))}((S)U)((WP)I)((I)CS)W'$	Equivalent pa clause.
$CU((S)\underline{U})((WP)I)((I)\underline{C}S)W'$	C1
$CU\underline{((}S\underline{))}((WP)I)((I)S)W'$	C2,2x
$CUS((WP)I)((I)\underline{S})W'$	C1
$CUS((WP)I)\underline{((}I\underline{))}W'$	C2
$CUS((WP)\underline{I})IW'$	C1
$CUS\underline{((}WP\underline{))}IW'$	C2
$CUS\underline{W}PI\underline{W'}$	C1
$\underline{W'W}CUSPI$	OI
$()$	J0. □

The pa calculation is, I submit, vastly simpler than Stoll's proof. The latter introduces the four premises gradually; the calculation introduces them all at the outset. The calculation then proceeds in a mechanical way that should be familiar from §5.2; at each step, invoke C2 [C1] to prune redundant variable instances [boundaries]. When a primed and unprimed instance of the same variable appears in the pervasive space, invoke J0 and terminate the calculation. The pa also reveals that any argument from Stoll's premises is valid, if the conclusion includes a primed instance of at least one variable appearing in the premises.

5.4.1. Recapitulation. The primitive basis of the primary algebra (pa) is as follows:

- The PA;
- Variables (statement letters), with or without subscripts ranging over the natural numbers, inserted anywhere in a PA formula. The improper symbols '′' and '…';
- The initial (3.1.5) set $(a)a=()$, $a(ab) = a(b)$, and $abc=bca$. Many other sets are possible;
- The usual inference rules for equational logics, the substitution of equivalents (R1), and the uniform replacement of variables (R2).

The contents of a boundary and of the pervasive space can be rearranged at will. Equivalently, juxtaposition is a tacit connective that commutes and associates. The pa is well suited to a decision procedure resembling Quine's TVA. From the initials, verifiable by TVA, either demonstrate a tautology Hilbert-style, or verify it by calculation. The pa is sound and complete, and has two intended interpretations: the Boolean algebra **2** and the CTV. If () denotes a truth value, the result is boundary logic, in which denial interprets (α), and alternation (*LoF*) or conjunction (dually) interprets juxtaposition.[64]

64. Kauffman (2001) and Bricken (2002) exposit BA and boundary logic in a manner more in philosophical sympathy with *LoF*. Other possible approaches, not pursued here, to a deeper understanding of BA include mereology and topology (Casati and Varzi 1999), and semiotics (Merrell 1982).