Sentential Calculus Review

Syntax

The sentential calculus studies how compound sentences are built out of simple sentences by means of truth-functional connectives. We start with atomic sentences, usually upper case Roman letters, sometimes with Arabic numeral subscripts. SC sentences are built from atomic sentences by means of four connectives: "$\vee$," which we use to translate English "or"; "$\wedge$," which we use to translate English "$\wedge$," "~," with goes with English "not"; and "$\rightarrow$," which pairs with "if,... then." The SC sentences make up the smallest class of expression that:

> contains the atomic sentences;
> contains ~ φ whenever it contains φ; and
> contains $(\varphi \vee \psi)$, $(\varphi \wedge \psi)$, and $(\varphi \rightarrow \psi)$, whenever it contains both φ and ψ.

In other words, if $\mathscr{F}$ is the set of all sets of expressions that satisfy the three conditions, then an expression is an SC sentence if and only if it is a member of every member of $\mathscr{F}$.

The Greek letters aren't part of the formal language. They are variables we use in English to talk about the formal language. A way to give the same definition, without the Greek letters, is this:

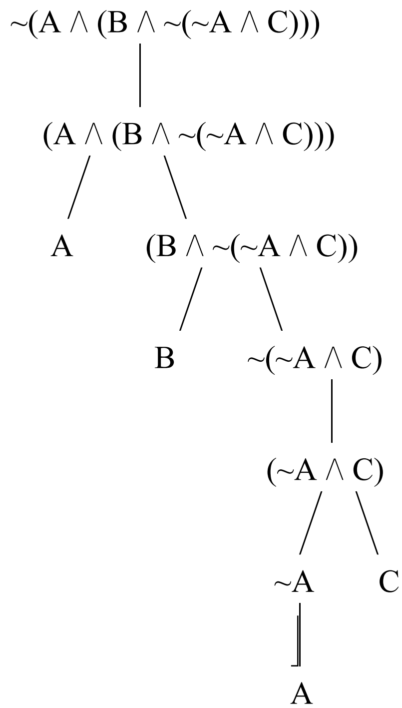
> Every atomic sentence is an SC sentence.
> The result of writing "~" in front of an SC sentence is always an SC sentence.
> Whenever you take two SC sentences, write one of the symbols "$\wedge$," "$\vee$," and "$\rightarrow$" between them, and enclose the result parentheses, what you get is an SC sentence.
> Nothing is an SC sentence unless it's required to be by the three clauses above.

Notice the role that parentheses play in preventing ambiguities. If you combine "A," "B," and "C" by first taking the disjunction of "A" and "B," then taking the conjunction with "C," you get something different from what you'd get by taking the disjunction of "A" with the conjunction of "B" and "C." The first procedure gives you "((A $\vee$ B) $\wedge$ C," whereas the second yields "(A $\vee$ (B $\wedge$ C))." Indeed, we have the following:

> **Unique Readability.** A sentence of the language for the sentential calculus is built up from sentential letters in a unique way.

Unique readability is an important respect in which the formal language is an improvement on English. The English sentence "Jack went up the hill or Jill went up the hill and someone fetched a pail of water" is ambiguous.

The proof of unique readability, which involves a meticulous investigation of the mating habits of parentheses [in brief, they are monogamous and heterosexual], will not be given here, but I will give the idea of the proof. The idea is that we can represent the structure of any given sentence by a finite tree in which each node of the tree is labeled by a sentence. The given sentence is the label of the trunk of tree. Whenever a node is labeled by a conjunction, there will be two other nodes directly beneath the node, each labeled with one of the conjuncts. Similarly for disjunctions, and conditionals. A node labeled by a negation has just one node directly beneath it, labeled by the sentence negated. The leaves of the tree are labeled by atomic sentences. For example, here is the tree associated with the sentence "~(A ∧ (B ∧ ~(~A ∧ C)))":



Unique readability is proved by showing that each sentence is associated with one and only one labeled tree.

Unique readability tells us that the SC sentences fall into five nonoverlapping categories:

atomic sentences.

disjunction: sentences of the form $(\varphi \vee \psi)$; φ and ψ are disjuncts

conjunctions: sentencs of the form $(\varphi \wedge \psi)$; φ and ψ are conjuncts.

conditionals: sentences of the form $(\varphi \rightarrow \psi)$; φ is the antecedent and ψ the consequent.

negations: sentences of the form $\sim \varphi$; φ is the negatum

Semantics

Sentential calculus only looks at how the truth or falsity of compound sentences is determined by the truth or falsity of their simpler components. It has nothing to say about when and wny atomic sentences are true.

> **Definition.** A *truth assignment* for a language for the sentential calculus is a function that assigns a value either true or false, to each sentence.
> :
> **Definition.** A truth assignment $\mathfrak{I}$ for a language for the sentential calculus is a *normal truth assignment* (*N.T.A.*) just in case it satisfies the following conditions:
>
> A conjunction is assigned the value true if and only if both its conjuncts are assigned true.
>
> A disjunction is assigned true if and only if one or both disjuncts are assigned true.
>
> A conditional is assigned ture and only if either its antecedent is assigned false or its consequent is assigned true (or both).
>
> A negation is assigned true if and only if the negatum is assigned false.
>
> **Definition.** A sentence is said to be *true* under a N.T.A. if and only if it's assigned the value 1 by the N.T.A. A sentence that's assigned 0 by the N.T.A. is *false* under the N.T.A.

This definition is handily summed up in the following table:

φ	ψ	$(\varphi \wedge \psi)$	$(\varphi \vee \psi)$	$(\varphi \rightarrow \psi)$	$\sim\varphi$
T	T	T	T	T	F
T	F	F	T	F	F
F	T	F	T	T	T
F	F	F	F	T	T

Using this table, we can determine for even the most complicated SC sentence whether or not it is true under a given N.T.A., provided we know which of the atomic sentence that occur within the sentence are true under the N.T.A. The method is to determine whether a complicated sentence is true by first determining whether its simpler components are true, then applying the table.

For example, to see whether "$(\neg(A \vee B) \rightarrow (A \wedge B))$" is true under an N.T.A. under which "A" and "B" are both true, note that "$(A \vee B)$" is true under the N.T.A., and so "$\neg(A \vee B)$" is false under the N.T.A.. "$(A \wedge B)$" will be true under the N.T.A., so that "$(\neg(A \vee B) \to (A \wedge B))$" will be a conditional with a false antecedent and true consequent, and so true.

Next consider an N.T.A. under which "A" is true and "B" false. Under this N.T.A., "$(A \vee B)$" be true and so "$\neg(A \vee B)$" will be false. "$(A \wedge B)$" will be false, and so "$(\neg(A \vee B) \to (A \wedge B))$" will Be a conditional with a false antecedent and a false consequent, and so again true.

Now consider an N.T.A. under which "A" is false and "B" true. Under this N.T.A., "$(A \vee B)$" will Be true and so "$\neg(A \vee B)$" will be false. "$(A \wedge B)$" will be false, and so, once again, "$(\neg(A \vee B) \to (A \wedge B))$" will be a conditional with a false antecedent and a false consequent, and so true.

Finally, consider an N.T.A. under which "A" and "B" are Both false. Under this N.T.A. "$(A \vee B)$" will be false, so that "$\neg(A \vee B)$" will be true. "$(A \wedge B)$" will be false. Thus under this N.T.A. "$(\neg(A \vee B) \to (A \wedge B))$" will have a true antecedent and false consequent, and so it will be false.

Our results are nicely summarized in the following table:

A	B	$(\neg(A \vee B) \to (A \wedge B))$			
T	T	F	T	T	T
T	F	F	T	T	F
F	T	F	T	T	F
F	F	T	F	F	F

This use of so-called *truth tables* to display the conditions under which a given sentence is true will prove to be extremely useful.

> **Definition.** An sentence is a *tautology* (or is *valid*) if and only if it is true under every N.T.A.. A sentence is a *contradiction* (or is *inconsistent*) if and only if it is false under every . A sentence is *indeterminate* (or *mixed*) if and only if it is true under some N.T.A.s and false under others.

Examples: "$((P \to Q) \vee (Q \to R))$" is a tautology, as we can see by examining the following table; the main connective is "$\vee$," and there are all "1"s under the main connective:

P	Q	R	((P → Q)	∨	(Q → R))
T	T	T	T	T	T
T	T	F	T	T	F
T	F	T	F	T	T
T	F	F	F	T	T
F	T	T	T	T	T
F	T	F	T	T	F
F	F	T	T	T	T
F	F	F	T	T	T

"¬((P → Q) ∨ (Q → R))" is a contradiction, as we can see from the following table; here the main connective is "¬," and there are all "0"s under the main connective:

P	Q	R	~	((P → Q)	∨	(Q → R))
T	T	T	F	T	T	T
T	T	F	F	T	T	F
T	F	T	F	F	T	T
T	F	F	F	F	T	T
F	T	T	F	T	T	T
F	T	F	F	T	T	F
F	F	T	F	T	T	T
F	F	F	F	T	T	T

"((P → Q) ∧ (Q → R))" is indeterminate, as we see from the following truth table, in which there are both "1"s and "0"s beneath the main connective, which is "∧":

P	Q	R	((P → Q)	∧	(Q → R))
T	T	T	T	T	T
T	T	F	T	F	F
T	F	T	F	F	T
T	F	F	F	F	T
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	T	T	T
F	F	F	T	T	T

In general, whether a sentence is true under a given N.T.A. is determined by determining which of the sentential letters the sentence contains are true, and which are false. Once you've determined which atomic sentences are true, all the values that are assigned the other sentences by a given NTA are settled; just work your way up the tree. If a given sentence contains n sentential letters, there will be 2^n ways to assign truth value to the sentential letters that appear in it, and so to test whether a sentence is a tautology, we have only to examine each of these 2^n possibilities. We can organize this investigation efficiently by writing a truth table.

> **Definitions.** A sentence φ is a *tautological consequence* of a set of sentences Γ iff φ is true under every NTA under which all the members of Γ are true.We also say that Γ *tautologically entails* or *tautologically implies* φ. If there is a NTA under which all the members of Γ are true, we say that Γ is truth-functionally consistent. Two sentences are *tautologically equivalent* iff they are true under precisely the same NTAs

We write $(\varphi \leftrightarrow \psi)$ as an abbreviation for $((\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi))$. φ and ψ are tautologically equivalent if and only if $(\varphi \leftrightarrow \psi)$ is tautological.

A distinction to keep in mind: Truth and falsity for SC sentences are relative notions. An SC sentence is true under a particular NTA or false under a particular NTA It makes no sense to say, simply, that an SC sentence is true or false. The notions of tautology, contradiction, logical equivalence, and implication are absolute notions. To be a tautology is to be true under every N.T.A.. That a sentence is a tautology is a property of the sentence, not relative to any particular N.T.A.. It makes no sense to say that a sentence is a tautology under an N.T.A., or that it is a contradiction under an N.T.A., or that one sentence implies another under an N.T.A., or that one is logically equivalent to another under an N.T.A..⊠

Compactness

There is only one big theorem in the sentential calculus. This is it:

> **Compactness Theorem.** A set Γ of SC sentences is tautologically entails a sentence χ iff some finite subset of Γ tautologically entails χ.

The notion of tautological entailment is a formalization of an informal notion of entailment according to which a set of sentences Γ entails χ iff it isn't possible for all the members of Γ to be true and χ false. Compactness doesn't hold for this informal notion of entailment, if we're talking about sentences of English. The sentences "There is at least one star," "There are at least two stars," "There are at least three stars," and so on, together entail "There are finitely many stars," but no finite subset does.

Before beginning the proof, let me describe the connection between this theorem and the notion of compactness employed by topology. Let us stipulate that a set S of N.T.A.s is *closed* just in case there is a set of sentence Γ such that S = the set of N.T.A.s under which every member of Γ is true. It is straightforward to verify that this defines a topology on the set of N.T.A.s. What the Compactness Theorem tells us is that this topology is compact.

Proof: The right-to-left direction is obvious. To prove the left-to-right direction, assume that no finite subset of Γ tautologically entails χ. So for each finite subset of Γ, there is a NTA under with all the members of the finite subset are true and χ is false. *Prima facie* it could happen that we have to use different NTAs for different sets. If fact, this doesn't happen. There is one NTA under with χ is false that verifies all the members of Γ. The Compactness Theorem is a one-size-fits-all theorem.

It is a little easier to work with sets of sentences than with truth assignments. For that reason, we make the following:

> **Definition.** A set of sentences Ω is a *complete story* just in case it satisfies the following five conditions, for any φ and ψ:
> a) $(\varphi \wedge \psi) \in \Omega$ iff $\varphi \in \Omega$ and $\psi \in \Omega$.
> b) $(\varphi \vee \psi) \in \Omega$ iff $\varphi \in \Omega$ or $\psi \in \Omega$ (or both).
> c) $(\varphi \rightarrow \psi) \in \Omega$ iff $\varphi \notin \Omega$ or $\psi \in \Omega$ (or both).
> d) $\sim \varphi \in \Omega$ iff $\varphi \notin \Omega$.

Clearly, a set of sentences Ω is a complete story if and only if there is a N.T.A. under which all and only the members of Ω are true. Thus the question whether χ is a tautological consequence of Γ can be reformulated as the question whether there is a complete story that contains all the members of Γ and excludes χ.

The strategy for proving compactness is this: We assume that no finite subset of Γ tautologically entails χ. We use this fact to build a complete story that contains Γ and excludes χ.

We are going to find a set of sentences Ω containing Γ with the following two properties:

> 1) No finite subset of Ω tautologically entails χ.
> 2) No set of sentences that properly contains Ω has property 1).

> **Claim.** Any set of sentences with properties 1) and 2) is a complete story.

We'll verify a). b) through d) are similar.

Suppose the $(\varphi \wedge \psi)$ is in Ω but φ isn't in Ω. Then by 2), some finite subset of $\Omega \cup \{\varphi\}$ tautologically entails χ. But any sentence tautologically entialed by $\Omega \cup \{\varphi\}$ is tautologically entailed by $\Omega \cup \{(\varphi \wedge \psi)\}$, which is the same as Ω. Contradiction. The hypothesis that $(\varphi \wedge \psi)$ is in Ω and ψ is not leads to a contradiction the same way.

Now suppose that φ and ψ are both in Ω but $(\varphi \wedge \psi)$ is not. There there is a finite subset Λ of Ω such that $\Lambda \cup \{(\varphi \wedge \psi)\}$ tautologically entails χ. $\Lambda \cup \{\varphi, \psi\}$ is a finite subset of Ω, so by 1) there is a NTA under which all the members of $\Lambda \cup \{\varphi, \psi\}$ are true and χ is false. This will be a NTA under which all the members $\Lambda \cup \{(\varphi \wedge \psi)\}$ are true and χ is false. Contradiction.

Out plan is to start with Γ, which satisfies condition 1), and build up a set of sentences $\Omega \supseteq \Gamma$ that satisfies 2) as well. We build Ω isn stages. We enumerate all the sentences of the language, in some sort of alphabetical order, as ζ_0, ζ_1, ζ_2, ζ_3, and so on. Form an infinite sequence $\Gamma_0, \Gamma_1, \Gamma_2, \Gamma_3, \ldots$ of sets of sentences, as follows:

$\Gamma_0 = \Gamma$. By hypothesis, Ω satisfies 1).

Given Γ_n satisfying condition 1). If $\Gamma_n \cup \{\zeta_n\}$ satisfies condition 1), let Γ_{n+1} be $\Gamma_n \cup \{\zeta_n\}$. Otherwise, $\Gamma_{n+1} = \Gamma_n$.

Let Ω be the union of the Γ_ns. Ω satisfies 1) and 2).

Our proof assumes that the language we're talking about is countable, that is, that it is possible to arrange the sentences of the language in an infinite list, $\zeta_0, \zeta_1, \zeta_2, \zeta_3$,.... The way we've defined "language for the sentential calculus," there are mathematically possible languages that aren't countable. The theorem still holds for uncountable language. The proof invokes Zorn's Lemma.. The same holds for most of the other theorems we'll prove.

Corollary. A set of sentences Ω is tautologically consistent iff every finite subset of Γ is tautologically consistent.

Proof: Take χ to be "(P $\wedge$ ~ P)," which we'll abbreviate "$\perp$."⊠

Corollary (Lindenbaum). A sentence χ is a tautological consequence of a set of sentences Γ if and only there are elements $\gamma_1, \gamma_2,..., \gamma_n$ of Γ such that $(\gamma_1 \rightarrow (\gamma_2 \rightarrow ... (\gamma_n \rightarrow \chi)...))$ is a tautology

Proof: (⇐) If $(\gamma_1 \rightarrow (\gamma_2 \rightarrow ... (\gamma_n \rightarrow \chi)...))$ is a tautology, it's in every complete story, so any complete story that also contains the γ_is will contain χ

(⇒) If χ is a tautological consequence of Γ, χ is a tautological consequence of some finite subset $\{\gamma_1, \gamma_2,..., \gamma_n\}$ of Γ. So $(\gamma_1 \rightarrow (\gamma_2 \rightarrow ... (\gamma_n \rightarrow \chi)...))$ is a tautology.⊠

Complete stories are important for us, because we can identify complete stories with possible worlds. We gives us that result that a sentence is tautological iff it is true in all possible worlds. A sentence is tautologically consistent iff it is true in at least one possible world. This is a metaphysically benign way to develop possible-world semantics (due mainly to Carnap, *Meaning and Necessity* (Chicago: 1947)). "There are fire-breathing dragons" is consistent, so there is a complete story in which "There are fire-breating dragons" is true, so "There are fire-breathing dragons" is true in some possible world. This way of thinking works for logical necessity. It doesn't work for metaphysical necessity. The most straightforward development of possible-world semantics for metaphysical seems to require us to postulate the existence of a magical place with hobbits and fire-breating dragons.

Extension Theorem

Theorem. Every function assigning truth values to the atomic sentences can be extended to a NTA in a unique way.

Given a function $\mathfrak{I}$ assigning values, true or false, to the atomic sentences, a *partial NTA* extending $\mathfrak{I}$ will be a function $\mathfrak{K}$, defined on a set of sentences, meeting the following conditions:

> Every atomic sentence is in the domain of $\mathfrak{K}$ and is assigned the same value by $\mathfrak{K}$ and by $\mathfrak{I}$.
>
> If a disjunction is in the domain of $\mathfrak{K}$, both disjuncts are in its domain, and the disjunction is assigned the value true iff one or both disjuncts are.
>
> Similar clauses for conjunctions, conditionals, and negations.
>
> **Lemma.** Two partial NTAs extending $\mathfrak{I}$ agree in the value they assign to any sentence in both their domains.

Proof: Define the *complexity* of a sentence as follows: An atomic sentence has complexity 0. A disjunction, conjunction, or conditional has complexity 1 greater that the maximum of the complexity of its two components. A negation has complexity 1 + the complexity of the negatum

Suppose the lemma is false, so there are partial NTAs $\mathfrak{F}$ and $\mathfrak{G}$ extending $\mathfrak{I}$ that assign different values to some sentence in their common domaing. Let χ be a sentence of least complexity on which they disagree. There are five cases:

Case 1. χ is atomic. Can't happen, since $\mathfrak{F}(\chi) = \mathfrak{I}(\chi) = \mathfrak{G}(\chi)$.

Case 2. χ is a disjunction $(\varphi \vee \psi)$. φ and ψ are assigned the same values by $\mathfrak{F}$ and by $\mathfrak{G}$. So the following are equivalent:

> $(\varphi \vee \psi)$ is declared true by $\mathfrak{F}$
> Either φ or ψ is declared true by $\mathfrak{F}$
> Either φ or ψ is declared true by $\mathfrak{G}$
> $(\varphi \vee \psi)$ is declared true by $\mathfrak{G}$

Contr adiction.

Cases 3-5. Similar. ⊠

It follows from the lemma that the union of all the partial NTAs extending $\mathfrak{I}$ is a partial NTA extending $\mathfrak{I}$. Call it $\mathfrak{K}$. We want to show that every sentence is in the domain of $\mathfrak{K}$. Suppose otherwise, and let χ be a sentence of least complexity that isn't in the domain of $\mathfrak{K}$. There are five cases.

Case 1. χ is atomic. Can't happen, because $\mathfrak{K}$ extends $\mathfrak{I}$.

Case 2. χ is a disjunction. If we extend $\mathfrak{K}$ by adding χ to its domain, declaring it true iff one or both disjuncts are declared true by $\mathfrak{K}$, we get a partial NTA extending $\mathfrak{I}$ that strictly includes $\mathfrak{K}$, which is absurd.

Cases 3-5. Similar.

A partial NTA whose domain is the entire set of sentences is just an NTA. So $\mathfrak{K}$ is the function we want. That it's the only function that does the job follows from the lemma. ⊠

State Descriptions, Disjunctive Normal Form, and Expressive Completeness

We have already learned how to find the truth table for a given SC sentence. We now want to attack the opposite problem: Given a truth table, how can we find a sentence that has that truth table?

Under any N.T.A., exactly one of the following eight SC sentences is true: "(A ∧ (B ∧ C))," "(A ∧ (B ∧ ~ C))," "(A ∧ (~ B ∧ C))," "(A ∧ (~B ∧ ~C))," "(~A ∧ (B ∧ C))," "(~ A ∧ (B ∧ ~C))," "(~ A ∧ (~ B ∧ C))," and "(~ A ∧ (~ B ∧ ~ C))." These eight sentences are said to be the *state descriptions* for "A," "B," and "C," since they completely describe the state of the world with respect to these three sentences. Each of the state descriptions is associated with a line of the truth table, as follows:

A	B	C	Associated state description
1	1	1	(A ∧ (B ∧ C))
1	1	0	(A ∧ (B ∧ ~ C))
1	0	1	(A ∧ (~ B ∧ C))
1	0	0	(A ∧ (~ B ∧ ~ C))
0	1	1	(~ A ∧ (B ∧ C))
0	1	0	(~ A ∧ (B ∧ ~ C))
0	0	1	(~ A ∧ (~ B ∧ C))
0	0	0	(~ A ∧ (~ B ∧ ~ C))

A state description is true at the line of the truth table it's associated with, and nowhere else. We can think of state descriptions as mini-worlds, ways the world might be as far as the members of a certain finite set of atomic sentences is concerned.

With the notion of state description in hand, the solution to our problem of getting a sentence with a given truth table is simple: We take our sentence to be the disjunction whose disjuncts are the state descriptions associated with those lines of the truth table where "1"s appear.

Example: Find an SC sentence whose truth table is this:

A	B	C	
1	1	1	0
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	0
0	1	0	1
0	0	1	0
0	0	0	0

Solution: "1" appears in the truth table at line two, three, and six, with which the associated state descriptions are "(A ∧ (B ∧ ~C))," "(A ∧ (~ B ∧ C))," and "(~ A ∧ (B ∧ ~ C))," respectively. So the sentence we want is the disjunction of those three state descriptions, namely: "((A ∧ (B ∧ ~ C)) ∨ ((A ∧ (~ B ∧ C)) ∨ (~ A ∧ (B ∧ ~ C))))."

Example: Find an SC sentence whose truth table is this:

A	B	C	
1	1	1	0
1	1	0	0
1	0	1	0
1	0	0	1
0	1	1	0
0	1	0	0
0	0	1	0
0	0	0	0

Solution: Here there is only one "1," so we don't need to form a disjunction. We just take the single state description corresponding to the line where "1" appears, namely, "(A ∧ (~ B ∧ ~ C))."

Exception: Our procedure tells us to form the disjunction of the state descriptions associated with lines of the truth table where "1"s appear. But what do we do if there aren't any "1"s? How do we find a sentence with the truth table

A	B	C	
1	1	1	0
1	1	0	0
1	0	1	0
1	0	0	0
0	1	1	0
0	1	0	0
0	0	1	0
0	0	0	0

Solution: This is easy. We just use "$(A \wedge \neg A)$."

The procedure we have been developing is perfectly general. Given atomic sentences α_1, α_2, ..., α_n, a truth table for α_1, α_2, ..., α_n will have 2^n rows, one row for each possible assignment of "1"s and "0"s to α_1, α_2, ..., α_n. Given a column of 2^n "0"s and "1"s, we want to find a sentence that has that column as its truth table. With each line of the truth table, we associate a state description, a conjunction of 2^n conjuncts whose i^{th} conjunct is either α_i or $\neg\alpha_i$, depending on whether "1" or "0" appears under α_i in that line. As our sentence whose truth table is the given column, we take the disjunction of the state descriptions of the state descriptions associated with positions in which "1" appears, assuming there is more than one such position. If "1" appears in only one position, we take our sentence to be the state description associated with that position. If there are no "1"s at all, we take our sentence to be $(\alpha_1 \wedge \neg\alpha_1)$.

In every case, the sentence we obtain by our procedure has a special form. The sentence we get is a disjunction of one or more sentences each of which is a conjunction of one or more atomic sentences or negated atomic sentences. A sentence that has this form is said to be in disjunctive normal form. Thus we have shown that, for any given truth table, we can find a sentence in disjunctive normal form that has that truth table.

Given any sentence, we can find a logically equivalent sentence in disjunctive normal form by writing out the truth table for the given sentence, then using the procedure given above to find a sentence in disjunctive normal form that has that truth table.

A sentence in disjunctive normal form contains only the connectives "$\vee$," "$\wedge$," and "~." Thus we see that, for any given truth table, we can find a sentence that contains only the connectives "$\vee$," "$\wedge$," and "~" that has that truth table. A set S of sentential connectives is said to be *expressively complete* iff, for any given truth table, you can find a sentence containing only connectives from S that has that truth table. Thus we see that {"$\vee$,""$\wedge$,""~"} is expressively complete.

Now "$\vee$" can be defined in terms of "$\wedge$" and "$\neg$," since $(\varphi \vee \psi)$ is logically equivalent to $\sim(\sim \varphi \wedge \sim\psi)$. This tells us that, for any given SC sentence, we can find a logically equivalent sentence whose only connectives are "$\wedge$" and "~" by the following procedure: First find a sentence in disjunctive normal form which is logically equivalent to the given sentence. Next look within the sentence in disjunctive normal form for a subsentence of the form $(\varphi \vee \psi)$; replace this subsentence with $\sim(\sim \varphi \wedge \sim \psi)$. Continue eliminating "$\vee$"s until they're all gone. This gives us a sentence whose only connectives are "$\wedge$" and "~." So {"$\wedge$," "~"} is expressively complete.

Similarly, for any given truth table, we can find a sentence with that truth table which contains only the connectives "$\vee$" and "~," by first finding a sentence in disjunctive normal form that has that truth table, then replacing each subsentence of the form $(\chi \wedge \theta)$ by $\sim(\sim \chi \vee \sim \theta)$. Hence {"$\vee$,""~"} is expressively complete.

{"→," "~"} is expressively complete, since we can write $(\varphi \vee \psi)$ as $(\sim \varphi \rightarrow \psi)$, and we can write $(\chi \wedge \theta)$ as $\sim (\chi \rightarrow \sim\theta)$.

By contrast, {"↔," "~"} isn't expressively complete, since any four-line truth table constructed out of "↔" and "~" will have "T"s on an even number of lines.

Similarly, the set {"∨,""∧,""→,""↔"} is not expressively complete, as we can see by observing that any sentence constructed out of these connectives is sure to have "T" at the top line of its truth table.

We might want to extend the sentential calculus by adding connectives "NOR" and "NAND" with the following truth tables:

φ	ψ	(φ NOR ψ)	(φ NAND ψ)
T	T	F	F
T	F	F	T
F	T	F	T
F	F	T	T

There is no need to extend the language this way. We can define "NOR" and "NAND":

$(\varphi \text{ NOR } \psi) =_{\text{Def}} \sim (\varphi \vee \psi)$.

$(\varphi \text{ NAND } \psi) =_{\text{Def}} \sim (\varphi \wedge \psi)$.

There is a dual to disjunctive normal form called *conjunctive normal form* (CNF): a conjunction of one or more conjuncts, each of which is a disjunction of atomic sentences and negated atomic sentences. For each sentence χ, we can find a logically equivalent sentence in CNF by the following procedure:

Find a sentence ξ in disjunctive normal form that is logically equivalent to $\neg \chi$.

Apply de Morgan's laws repeatedly to $\sim \xi$ to push the negation sign inside, as far as it will go. De Morgan's laws tell us that $\sim (\varphi \vee \psi)$ is logically equivalent to $(\sim \varphi \wedge \sim \psi)$ and that $\sim (\varphi \wedge \psi)$ is logically equivalent ot $(\sim \varphi \vee \sim \psi)$.

Eliminate double negations.

As an example, let χ be "$(P \leftrightarrow (Q \vee R))$." Writing $\sim \chi$ in DNF gives us this:

$$((P \wedge (\sim Q \wedge \sim R)) \vee ((\sim P \wedge (Q \wedge R)) \vee ((\sim P \wedge (Q \wedge \sim R)) \vee (\sim P \wedge (\sim Q \wedge R)))))$$

Negating this an pushing the negation sign inside, we get this:

$$(\sim (P \wedge (\sim Q \wedge \sim R)) \wedge (\sim (\sim P \wedge (Q \wedge R)) \wedge (\sim (\sim P \wedge (Q \wedge \sim R)) \wedge \sim (\sim P \wedge (\sim Q \wedge R))))).$$

Pushing further:

$$((\sim P \vee (\sim \sim Q \vee \sim \sim R)) \wedge ((\sim \sim P \vee (\sim Q \vee \sim R)) \wedge ((\sim \sim P \vee (\sim Q \vee \sim \sim R)) \wedge ((\sim \sim P \vee ((\sim \sim Q \vee \sim R))))).$$

Eliminating double negations gives us CNF:

$$(\sim P \vee (Q \vee R)) \wedge ((P \vee (\sim Q \vee \sim R)) \wedge ((P \vee (\sim Q \vee R)) \wedge ((P \vee ((Q \vee \sim R))))).$$

Sentential calculus was advertised as the study of how simple sentences can be combined by means of truth-functional connectives, but it looks like something much more specialized, the study of how simple sentences can be combined by means of the four connectives “$\vee$,” “$\wedge$,” “$\rightarrow$,” and “$\sim$.” We now see that this wasn’t false advertising after all. The effect of any truth-functional connective can be achieved using the connectives we already have.