



## Three Logical Theories

John Corcoran

*Philosophy of Science*, Vol. 36, No. 2. (Jun., 1969), pp. 153-177.

Stable URL:

<http://links.jstor.org/sici?sici=0031-8248%28196906%2936%3A2%3C153%3ATLT%3E2.0.CO%3B2-M>

*Philosophy of Science* is currently published by The University of Chicago Press.

---

Your use of the JSTOR archive indicates your acceptance of JSTOR's Terms and Conditions of Use, available at <http://www.jstor.org/about/terms.html>. JSTOR's Terms and Conditions of Use provides, in part, that unless you have obtained prior permission, you may not download an entire issue of a journal or multiple copies of articles, and you may use content in the JSTOR archive only for your personal, non-commercial use.

Please contact the publisher regarding any further use of this work. Publisher contact information may be obtained at <http://www.jstor.org/journals/ucpress.html>.

Each copy of any part of a JSTOR transmission must contain the same copyright notice that appears on the screen or printed page of such transmission.

---

The JSTOR Archive is a trusted digital repository providing for long-term preservation and access to leading academic journals and scholarly literature from around the world. The Archive is supported by libraries, scholarly societies, publishers, and foundations. It is an initiative of JSTOR, a not-for-profit organization with a mission to help the scholarly community take advantage of advances in technology. For more information regarding JSTOR, please contact [support@jstor.org](mailto:support@jstor.org).

## THREE LOGICAL THEORIES\*

JOHN CORCORAN

*State University of New York at Buffalo*

This study concerns logical systems considered as theories. By searching for the problems which the traditionally given systems may reasonably be intended to solve, we clarify the rationales for the adequacy criteria commonly applied to logical systems. From this point of view there appear to be three basic types of logical systems: those concerned with logical truth; those concerned with logical truth and with logical consequence; and those concerned with deduction *per se* as well as with logical truth and logical consequence. Adequacy criteria for systems of the first two types include: effectiveness, soundness, completeness, Post completeness, "strong soundness" and strong completeness. Consideration of a logical system as a theory of deduction leads us to attempt to formulate two adequacy criteria for systems of proofs. The first deals with the concept of rigor or "gaplessness" in proofs. The second is a completeness condition for a system of proofs. An historical note at the end of the paper suggests a remarkable parallel between the above hierarchy of systems and the actual historical development of this area of logic.

This paper is mainly a contribution to the philosophy of the science of logic. Our primary purpose is to apply a certain methodological principle to logical systems *considered as theories*. The principle may be stated as follows: The form of a completed theory is partly determined by the nature of the subject matter of the theory and partly by the specific problems that the theory is intended to solve. On the basis of the problems they may reasonably be intended to solve we distinguish three types of logical systems. The first includes the early logistic systems. The second includes the logical systems of a later period where proofs from arbitrary sets of assumptions play a dominant role. The third type includes natural deduction systems which have been appearing recently. (For examples of systems of the first type see [20] and [11]; for a clear example of the second see [6]; the systems of [19] and [1] are of the third type.)

This classification of logical systems has two important properties. In the first place it is a trichotomy: every logical system is in exactly one of the three classes. In the second place it is hierarchical in the following sense. The problems dealt with by the first type of system are also dealt with by the second which treats of additional problems as well. The same is true of the second with respect to the third. This hierarchy of systems is completely general in that there are three types of propositional logics, three types of first order logics, three types of second order logics, three types of modal logics and so on. Everything that will be said is intended to apply to any and all of these logics. However, in order to provide a simple example of each type of system we will restrict part of the discussion to a fragment of the logic of sentences (propositional logic).

For the purposes of the examples we assume a language  $L$  which has in its logical vocabulary the symbol  $\supset$  having the truth-functional meaning of 'if . . . then . . .'

\* Received March, 1967.

the symbol  $\sim$  having the truth-functional meaning of 'not,' and parentheses for grouping.  $L$  may indeed contain symbols for 'and,' 'or,' 'iff,' quantifiers of any kind, modal operators, etc. All we are assuming for the purposes of the examples is the above and that under each interpretation of the language (in which the meanings of  $\supset$  and  $\sim$  remain fixed) each sentence is either true or false and not both. Thus  $L$  could be a formal language of set theory, geometry, biology, etc. or a formal language made up for purely logical purposes.

### 1. Logical Truth: Logistic Systems.

*"It is possible . . . to give in advance a description of all 'true' logical propositions."*

WITTGENSTEIN

Among the true sentences of a language one may distinguish a subclass of logically true sentences. The logically true sentences are those which are true in virtue of the meanings of the logical symbols only. More specifically, the logically true sentences are those which are true under all interpretations of the language. (By an interpretation we mean an interpretation in which the logical symbols have their normal meanings and in which all of the grammatical distinctions are preserved: proper nouns are interpreted as individuals, one-placed predicates as sets of individuals, etc.) One reasonable aim of a logical theory is the codification of all logically true sentences. This aim is especially reasonable for a logician who believes that all laws of mathematics are logically true; once he has a codification of all logically true sentences he need "only" check to see if all mathematical laws are in his system.

Logical systems intended primarily to codify logically true sentences are here called *logistic systems*. The system of Principia Mathematica [20] is of this type although at the time of its inception Russell was not aware of the above idea of logical truth (cf. his remarks in [4], pp. 221–222). All of Lewis' systems [11] are logistic systems.

Usually a logistic system consists in a set of sentences called axioms together with some production rules which generate further sentences from the axioms. The sentences actually codified by the system are the axioms together with the sentences generated from the axioms by means of the rules—all of these are called theorems.

In our example we will codify only those logically true sentences of  $L$  whose logical truth turns on the meanings of ' $\supset$ ' and ' $\sim$ .' This will be done indirectly. We will codify schemas or sentence forms all of whose *instances* are logically true sentences. In general, the schemas are *not* sentences of  $L$  but their instances are. For example  $(P \supset P)$  is definitely not a sentence of  $L$  if  $L$  happens to be the language of geometry. However, if  $a$  is a sentence of geometry then  $(a \supset a)$  is again a sentence of geometry and, moreover  $(a \supset a)$  is an instance of  $(P \supset P)$ . All instances of  $(P \supset P)$  are logically true so  $(P \supset P)$  is the sort of schema that we will codify. On the other hand, since some instances of  $(P \supset Q)$  are logically true and some are not we will want to avoid codifying  $(P \supset Q)$ .

For definiteness we define the *schemas* as any of the schematic letters  $P, Q, R, P_1, Q_1, R_1, P_2, \dots$  and any formula built up from these using  $\supset$  and  $\sim$  with parentheses being used for grouping. A schema which has only logically true sentences

as instances is called a *law of logic*. Some laws of logic, like  $(P \supset (Q \supset P))$ , are of the ‘if . . . then . . .’ form and some, like  $\sim \sim (P \supset P)$ , are not. Those of the ‘if . . . then . . .’ form are especially important because of their roles in making valid inferences: given sentences  $a$  and  $b$ , from  $a$  using the above law we may validly infer  $(b \supset a)$ . Some logicians regard this proper subclass of the class of logical laws as the principal subject of logic ([2], pp. 2 and 3).

The logistic system will codify a class of schemas called *theorems*. These theorems are intended to include all and only the logical laws in question. The class of theorems is defined by taking some logical laws as axioms and by giving operations by which the others may be obtained from the axioms.

The logistic system  $LS$  is defined by the following five definitions.


Definition: The following three schemas are *axioms of LS*

- (A1)  $(P \supset (Q \supset P))$   
 (A2)  $((P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R)))$   
 (A3)  $((\sim P \supset \sim Q) \supset ((\sim P \supset Q) \supset P))$

Definition: A schema  $\phi$  is defined to be *obtained from* a schema  $\delta$  by *substitution* iff  $\phi$  is the result of replacing all occurrences of a single schematic letter in  $\delta$  by a schema  $\eta$  or by performing the above operation simultaneously on a number of schematic letters.

For example: every schema  $\phi$  is obtained from the schema  $P$  by substitution because we may replace all occurrences of  $P$  (only one in this case) by  $\phi$ . In particular,  $\sim P$  is obtained by substitution from  $P$ . Needless to say, substitution is *not* a sound rule of inference but it *is* a way of generating logical laws from logical laws: e.g. substituting  $(P \supset \sim P)$  for  $Q$  in A1 produces  $(P \supset ((P \supset \sim P) \supset P))$  which is another logical law.

Definition: A schema  $\delta$  is said to be *obtained from* schemas  $\phi$  and  $\eta$  by *detachment* iff  $\eta = (\phi \supset \delta)$ . (In other words,  $\delta$  is obtained from  $\phi$  and  $(\phi \supset \delta)$  by detachment.)

For example,  $(Q \supset P)$  is obtained from  $P$  and  $(P \supset (Q \supset P))$  by detachment. This operation is immediately recognized as a sound rule of inference but the fact that it is a sound rule of inference is irrelevant to the blem at hand—i.e. to the problem of codifying all schemas all of whose instances are logically true sentences of the appropriate sort.

Now we may define the derivations in  $LS$ . Derivations are lists of schemas which may be offered as evidence that certain schemas are codified by the system. Derivations correspond to proofs which involve only logical reasoning except that the principles used (1) are not sound and (2) do not correspond to normal reasoning.

Definition: A *derivation in LS* is a finite sequence of schemas beginning with an axiom and such that each subsequent schema in the list *either* is an axiom *or* is obtained from a previous schema by means of substitution *or* is obtained from two previous schemas by detachment.

Below is an example of a derivation. This derivation establishes that  $(P \supset P)$  is a

“theorem” of *LS*. Next to each line we indicate the rule according to which that line was entered.

- |     |                                                                                                     |                                                             |
|-----|-----------------------------------------------------------------------------------------------------|-------------------------------------------------------------|
| (1) | $(P \supset (Q \supset P))$                                                                         | A1                                                          |
| (2) | $(P \supset ((P \supset P) \supset P))$                                                             | substituting $(P \supset P)$ for $Q$ in (1)                 |
| (3) | $((P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R)))$                         | A2                                                          |
| (4) | $((P \supset ((P \supset P) \supset P)) \supset ((P \supset (P \supset P)) \supset (P \supset P)))$ | substituting $(P \supset P)$ for $Q$ and $P$ for $R$ in (3) |
| (5) | $((P \supset (P \supset P)) \supset (P \supset P))$                                                 | detachment (2) (4)                                          |
| (6) | $(P \supset (P \supset P))$                                                                         | substituting $P$ for $Q$ in (1)                             |
| (7) | $(P \supset P)$                                                                                     | detachment (5) (6)                                          |

Now we define the theorems as the schemas actually codified by the system.

**Definition:** A *theorem of LS* is the last schema of some derivation in *LS*.

Thus the axioms of *LS* are theorems as they each form the last line of a one-line derivation in *LS*. In addition we have as theorems the schemas of lines (2), (4), (5), (6), and (7).

There are two obvious adequacy conditions to be placed on any logistic system. First, it must be *sound*, i.e. all sentences codified must be logically true. In *LS* all theorems must be laws of logic. To demonstrate soundness of *LS* it is sufficient to show that the axioms are all laws of logic and that the rules of substitution and detachment serve to generate from laws of logic only laws of logic. In the second place it is desirable that the logistic system be *complete*, i.e. that it codify all logically true sentences. In *LS* it is desirable that every law of logic be a theorem. By using methods developed by Post [16] *LS* can be shown to be sound and complete. (An easier approach to completeness of *LS*: show that all theorems of the system *L* ([14], p. 31) are theorems of *LS* and then use Kalmar’s idea ([14], p. 36).)

If a logistic system is not sound then among its theorems are sentences which are not logically true. Thus an unsound logistic system would be abandoned (or modified). However, an incomplete logistic system is still of value. Indeed many logistic systems are inherently incomplete but they still serve the purpose of codifying logically true sentences—they just do not codify all of them.

How could a logistic system be inherently incomplete? It would seem that we could continue to add as new axioms any laws of logic found not derivable. This is certainly true. But there is no guarantee that after adding a finite number of new axioms we would achieve completeness. Indeed the logical truths of a language may be so complex that there is no effective method even for specifying an infinite set of axioms which would make the system complete.

As a limiting case, what is to prevent taking all logical laws as axioms to begin with? This suggestion is, in general, its own *reductio ad absurdum*. Would we be able to regard ourselves as having *codified* the logical truths of a language by merely uttering an incantation such as “let the logical truths be hereby codified”?

In order to regard a system as a codification the axioms must be given in such a way as to enable us to decide *whether or not* any given sentence (or schema) in question is an axiom. Moreover, given any finite list of such sentences (or schemas)

we must be able to decide whether or not the last was obtained from the others by one of the operations. In short, we must be able to decide of any finite list of sentences (or schemas) whether or not it is a derivation.

We may formulate a third condition as follows: The derivations of a logistic system must be decidable, i.e. the axioms must be decidable and the operations must be decidable. Unless this condition held we would have no reason for regarding a system as a codification at all. A system whose derivations are decidable we shall call *effective*.

(We remark parenthetically that there are logistic systems—codifications of logical truths—which do not have the axioms-plus-rules form. The concept of effective formulated above would therefore have to be modified—but some such condition must be satisfied by any system which is claimed to be a codification.)

There is a fourth adequacy condition for logistic systems which was formulated by Post ([16], p. 173). Consider a complete and sound logistic system, say  $S$ . If a new axiom  $\phi$  were added to  $S$  making a new system  $S'$  then one of two things would necessarily happen. First if  $\phi$  is logically true it would already have been a theorem of  $S$  so  $S'$  is sound and complete but it has a redundant axiom. Second if  $\phi$  is not logically true then  $S'$  is unsound—one of its theorems (viz.  $\phi$ ) is not logically true. If the generating power of  $S$  were “rich enough” then the one unsound theorem would “destroy” the system by allowing all sentences to be generated, in particular (e.g. in  $LS$ ), both  $P$  and  $\sim P$ . If this happened we might say that the system  $S$  is so complete that (properly) new axioms can not be added. This is the idea of *Post completeness*. Post also developed methods by which  $LS$  can be shown to be Post complete. To get an idea of why  $LS$  is Post complete, consider adding  $P$  as new axiom—immediately every schema (including  $\sim P$ ) becomes a theorem. The same thing results from adding  $(Q \supset P)$ : by substituting any axiom, say  $A1$ , for  $Q$  in  $(Q \supset P)$  we get  $(A1 \supset P)$  and then by detachment from  $A1$  and  $(A1 \supset P)$  we get  $P$ . From  $P$  all schemas are obtained by substitution. For  $LS$  this happens whenever *any* properly new axiom is added, i.e. whenever we add as an axiom a schema not already a theorem. This is what it means for  $LS$  to be Post complete.

The adequacy conditions for logistic systems may be summarized as follows. Every logistic system must be effective and sound, i.e. must be a codification and must codify only logically true sentences. An “ideal” logistic system is also complete (codifies all logically true sentences) and Post complete (so complete that properly new axioms can not be added). Notice that completeness is obviously a desirable property of logistic system whereas, although it is not unreasonable to want Post completeness, the lack of Post completeness is not a deficiency in a logistic system.

## 2. Valid Arguments: Consequence Systems.

*“The central problem of logic that has motivated much of its development from Aristotle’s syllogisms down to the mathematical logic of the present day is: Given any sentences  $Q, P_1, P_2, \dots$ , state precisely under what circumstances the sentence  $Q$  is a logical consequence of the sentences  $P_1, P_2, \dots$ ”*

PAUL GILMORE

A second and more comprehensive aim of a logical theory is the codification of all valid arguments. This aim is reasonable not only for the logicist who must recognize the value of valid arguments in law, philosophy, science, etc. but also for all other logicians. An argument is a system of sentences one of which is called the *conclusion* and the rest called *premises*. We may represent any argument as  $S \therefore p$  where  $S$  is a set of sentences and  $p$  is a sentence. An argument  $S \therefore p$  is valid if the truth of the premises guarantees the truth of the conclusion or, put another way, if it is impossible for the conclusion to be false while all of the premises are true. In terms of interpretations we may define an argument  $S \therefore p$  as valid (following Tarski [19], pp. 309–420) iff every interpretation making all sentences in  $S$  true also makes  $p$  true. It follows that  $S \therefore p$  is valid iff there is no interpretation making all sentences in  $S$  true and also making  $p$  false.

The conclusion of a valid argument is commonly said to be a *logical consequence* of the premises. For this reason we use the term *consequence system* to refer to a logical system which is intended to codify not only logically true sentences but also valid arguments. Some logicians ([13], p. 2; [12], p. v) regard the investigation of logical consequences as the principal concern of logic. Systems given in these references are examples of *consequence systems*.

A consequence system codifies the logically true sentences of a language in much the same manner as a logicist system. The valid arguments are codified by giving rules of inference by which *the logical consequences of a given set of sentences are produced from the given set*. Needless to say the rules of inference must always produce from a given set of sentences only logical consequences of it. Thus substitution can not be used. Detachment does produce from two sentences a third which is a logical consequence.

As an example we will give a consequence system  $CS$  which will codify (1) the same logically true sentences of  $L$  as codified by  $LS$  and (2) the valid arguments of  $L$  whose validity involves only the meanings of  $\supset$  and  $\sim$ . The latter will be done by codifying valid argument forms. An *argument form* is a system of schemas one of which is called *the conclusion* and the rest called *premises*. We may represent an argument form as  $\Sigma \therefore \phi$  where  $\Sigma$  is a set of schemas and  $\phi$  is a schema. An argument form is *valid* if all of its instances are valid arguments.

The following five definitions define the consequence system  $CS$ .

**Definition:** The *axioms of CS* are all schemas of the following forms

- (B1)  $(\phi \supset (\delta \supset \phi))$   
 (B2)  $((\phi \supset (\delta \supset \eta)) \supset ((\phi \supset \delta) \supset (\phi \supset \eta)))$   
 (B3)  $((\sim \phi \supset \sim \delta) \supset ((\sim \phi \supset \delta) \supset \phi))$

Thus  $CS$  has an infinite number of axioms whereas  $LS$  had only three. This definition enables the incorporation of the sound “part” of substitution and the avoidance of the unsound “part.” Our only generating operation is detachment defined as above.

**Definition:** (definition of detachment as above).

A derivation in a consequence system is a “record” of a production of a conclu-

sion from premises. Derivations correspond to normal, rigorous proofs from premises except that principles used to construct a derivation are not necessarily normal or rigorous.

**Definition:** A *derivation of  $\phi$  from  $\Sigma$*  is a finite sequence of schemas (1) beginning either with a premise in  $\Sigma$  or with an axiom and (2) such that each subsequent schema *either* is a premise in  $\Sigma$ , *or* is an axiom, *or* is obtained from two previous schemas in the sequence by means of detachment and (3) which ends with  $\phi$ .

Logical laws are codified by derivations from the empty set of premises. This is exactly the method by which logical laws are codified in a logistic system except that in a logistic system there are derivations *only* from the empty set.

**Definition:** A *theorem of CS* is the last schema of a derivation from  $\Lambda$ , the empty set.

Valid arguments are codified by derivable argument forms.

**Definition:** An argument form  $\Sigma \therefore \phi$  is *derivable in CS* iff there is a derivation of  $\phi$  from  $\Sigma$ .

Notice that CS codifies arguments having an infinite number of premises. Any consequence system must do this in order to make any claim to comprehensiveness because there are many sciences which have an infinite number of axioms and a consequence system certainly should be able to codify the valid arguments in science. More generally, because infinite sets of sentences do indeed have logical consequences, a consequence system should be able to codify arguments having an infinite number of premises.

As a result of our general definitions there are argument forms having no premises at all. There are derivable argument forms of the form  $\Lambda \therefore \phi$ . This may seem to be an artifact of the system. But, by the definition we have that  $\phi$  is a theorem iff  $\Lambda \therefore \phi$  is derivable. This accords nicely with Leibniz's observation that a logically true sentence is one which is derivable on the basis of logic alone without using any other principles.

Below are a few examples of derivations. We will head the derivation with the argument form being derived and give on the right the rules used.

$\Lambda \therefore (P \supset P)$

- |                                                                                                         |                     |
|---------------------------------------------------------------------------------------------------------|---------------------|
| (1) $(P \supset ((P \supset P) \supset P))$                                                             | schema in form (B1) |
| (2) $((P \supset ((P \supset P) \supset P)) \supset ((P \supset (P \supset P)) \supset (P \supset P)))$ | schema in form (B2) |
| (3) $((P \supset (P \supset P)) \supset (P \supset P))$                                                 | detachment (1) (2)  |
| (4) $(P \supset (P \supset P))$                                                                         | schema in form (B1) |
| (5) $(P \supset P)$                                                                                     | detachment (3) (4)  |

$[P] \therefore P$

- |         |                  |
|---------|------------------|
| (1) $P$ | premise in $[P]$ |
|---------|------------------|



$[P, (P \supset Q), (Q \supset R)] \therefore R$

(1) $P$	premise in $[P, (P \supset Q), (Q \supset R)]$
(2) $(P \supset Q)$	premise in $[P, (P \supset Q), (Q \supset R)]$
(3) $(Q \supset R)$	premise in $[P, (P \supset Q), (Q \supset R)]$
(4) $Q$	detachment (1), (2)
(5) $R$	detachment (3), (4)

In order adequately to serve the purposes (1) of codifying all and only logically true sentences of a language and (2) of codifying all and only valid arguments of a language, a consequence system must satisfy four conditions. In the first and second places it must be complete and sound in the above senses. Third, it must be *strongly complete*, i.e. it must codify all valid arguments. Finally, it must be *strongly sound*, i.e. it must codify only valid arguments.

As far as *CS* is concerned, the discussion of soundness and completeness for *LS* carries over with only minor changes. To establish strong soundness it is sufficient to show that all of the axioms are logical laws and that detachment always produces from two schemas a third which is a logical consequence. The simplest proof of strong completeness for *CS* is probably one which follows Henkin's ideas [6].

Post completeness is clearly not an appropriate adequacy condition for a consequence system as if the addition of any properly new axiom would destroy the system then the system would fail to be strongly sound. In *S* we should not expect  $\sim P$  to be derivable from  $P$ .

It goes without saying that a consequence system must be effective (in an appropriate sense). Otherwise we could not regard it as codifying anything.

Both logistic systems and consequence systems, as we have seen, each have one kind of derivation. The basic difference between the two types of systems lies in the nature of their respective kinds of derivations. The derivations of a logistic system must be "based on" the axioms of the system whereas in a consequence system the derivations are derivations from arbitrary sets of premises. It is this difference which makes the consequence system able to codify valid arguments as well as logical truths. The logistic system, on the other hand, is appropriate only to the task of codifying logical truths.

There have been attempts to use logistic systems as codifications of valid arguments (see e.g. [11], p. 341 or [7], p. 23). If this were to be done by introducing derivations from assumptions then, of course, it would be more reasonable to say that the system has been modified to a consequence system. Let us digress briefly to consider how logistic systems may be thought of as codifying valid arguments without introducing derivations from assumptions.

In *LS*, for example, we could use theorems of the form  $(\phi \supset \delta)$  to codify argument forms with one premise  $[\phi] \therefore \delta$ . But this is unsatisfactory as it codifies only arguments with one premise. The difficulty can be partially circumvented by use of the principle of the corresponding conditional which relates valid argument to logically true sentences. In order to state this principle neatly we need to assume a language  $L'$  which is like  $L$  except that it also contains  $\&$  in its truth-functional

meaning. Assume also that  $LS$  is modified so that it codifies all logical truths of  $L'$  turning on  $\supset$ ,  $\sim$ , or  $\&$ . For example,  $((P \& Q) \supset P)$  would be a theorem of this new logistic system which we will call  $LS'$ .

The principle of the corresponding conditional relates the validity of an argument having a *nonempty*, finite set of premises to the logical truth of the conditional sentence whose antecedent is the conjunction of the premises and whose consequent is the conclusion. More precisely:

$[\phi_1, \phi_2, \dots, \phi_n] \therefore \delta$  is valid iff  $((\phi_1 \& \phi_2 \& \dots \& \phi_n) \supset \delta)$  is logically true.

Thus we may take arguments with non-empty finite sets of premises to be codified by the corresponding conditional. Moreover, arguments having no premises may be codified by their conclusions since we have:

$\Lambda \therefore \delta$  is valid iff  $\delta$  is logically true.

Thus all finite arguments are codified within a logistic system. To codify an infinite argument we may use a conditional whose antecedent is a conjunction of *some* of the premises and whose consequent is the conclusion. For example if  $H$  is the set of all schematic letters then the valid argument form  $H \therefore P$  would be codified by  $(P \supset P)$  since  $[P]$  is a subset of  $H$ .

This arrangement would not be adequate in case there were an infinite set of premises (1) which implied a conclusion but (2) which was such that no finite subset also implied the same conclusion. But since derivations are only of finite length, the consequence system would not be able to codify such an argument either.

Thus a logistic system can indeed be used to codify valid arguments as well as logical truths. But this is like cutting pie with a pencil—the tool is not appropriate to the task but can still be used if one does not mind a messy job. In general, the derivations of the corresponding conditional will be much longer and more difficult to discover than the derivation of the conclusion from the premises. Moreover, there are many interesting relations among valid arguments which are nicely manageable in a consequence system but almost impossibly complicated in a logistic system. On this point one may compare the development of the deduction theorem in a consequence system (e.g. [14], p. 32) with its development in a comparable logistic system (e.g. [3], pp. 86–89).

Probably the most troublesome difficulty with trying to treat valid arguments in a system not suited for the task is a conceptual one. If we take an argument  $[\phi_1, \phi_2, \dots, \phi_n] \therefore \delta$  as codified by  $((\phi_1 \& \phi_2 \dots \& \phi_n) \supset \delta)$  then there is a tendency to confuse 'implies' with ' $\supset$ .' It is a social fact that mathematicians, needing a short word to express  $\supset$ , use the word 'implies.' This fact does nothing to resolve the confusion but actually seems to lend authority to maintaining it. The obvious facts resolving the confusion (e.g. that ' $\supset$ ' is in  $L$  whereas 'implies' need not be) tend to be overlooked when one is working in a logistic system.

In summary we reiterate that logistic systems are naturally thought of as concerned only with logical truth whereas consequence systems treat logical consequence as well as logical truth. Since the consequence systems treat a more comprehensive

class of problems the adequacy conditions for a consequence system are more comprehensive than those for the logistic system. In addition to being sound a consequence system must also be strongly sound. For a consequence system, not only is completeness desirable but strong completeness is as well. Moreover, although it is not unreasonable to expect a logistic system to be Post complete no adequate consequence system can have this property as it conflicts with strong soundness. Finally, although valid arguments can be codified in a system which does not have derivations from premises the details of the codification are unnecessarily complicated, inefficient and the opposite of enlightening.

### 3. Proofs: Deductive Systems

*“Proofs . . . had to exist before the structure of a proof could be logically analysed; and this analysis . . . must have rested . . . on a large body of mathematical writings. In other words, logic, so far as we mathematicians are concerned, is no more and no less than the grammar of the language which we use, a language which had to exist before the grammar could be constructed.”*

BOURBAKI

In addition to codifying the logically true sentences and the valid arguments, a logician may be concerned to codify the proofs of a language. In other words, one may wish to examine the steps of reasoning involved in showing that a conclusion follows from premises. In order to get an idea of what the proofs of a language are one may examine a number of them in some existing corpus of deductive reasoning (like geometry).

By examining a number of proofs one is almost forced to conclude that proofs are finite discourses structured in accordance with certain rules which may be called rules of inference. Thus we may codify the proofs of a language by specifying the rules of inference and how they are applied.

In logic we do not aim to codify all proofs which occur but we idealize in two directions. In the first place we do not wish to codify mistakes so we require that rules of inference be sound, i.e. that they lead from premises to conclusions which actually follow. In the second place we do not want to codify great leaps of logical intuition—we want our rules to be absolutely simple, i.e. we want to codify the proofs which have the maximum amount of logical detail. In short we wish to codify the logically rigorous proofs.

As an example we will consider the problem of codifying the proofs of  $L$  and only those which turn on the meanings of  $\supset$  and  $\sim$ . We will codify them by codifying *proof forms* which are discourse schemas all of whose instances are proofs involving only the meanings of  $\supset$  and  $\sim$ . One reasonable candidate for proof forms is the derivations of  $CS$ . But there is one obvious obstacle to taking these as forms of rigorous proofs even though they are sound.

The writing of axioms (especially, long and intricate ones) is neither a principle actually found in normal reasoning nor is it rigorous. The writing of axioms as in the derivations of  $CS$  is not rigorous because *more logical detail can reasonably be requested and easily supplied*. Thus, a proof which cited such an axiom would contain a logical gap.

Derivations from premises are rejected despite the fact that they are sound and this is as it should be as it is one thing for an inference to be sound and quite another for it to be logically simple, i.e. have no “gaps” in the reasoning. One may soundly infer a very complicated theorem of geometry directly from the axioms but one would hardly regard this as proving the theorem from the axioms.

(a) *Principles of Reasoning*

Below we shall survey in thought the kinds of principles actually found in normal reasoning. Our aim is *not* that of characterizing normal or natural reasoning—natural reasoning is certainly *not* rigorous. However, as we will see below, there are rigorous principles frequently used in normal reasoning. We will use some of these principles to set up a deductive system which codifies not only logical truths and valid arguments but also proofs which may, with good reason, be called rigorous.

In surveying the proofs of geometry (or any other body of deductive reasoning) we find many kinds of inference—some sound and, most likely, some unsound, some “gapless” and some which could bear having some additional logical detail filled-in. Of the gapless or rigorous principles of proof we find three kinds.

The first type of principle to be noticed will most likely be an *immediate inference rule*. An immediate inference rule is a rule which permits the writing of a new line in a proof on the basis of one or two previous lines. Detachment is of this sort. Another is the principle by which we infer  $P$  from  $(P \ \& \ Q)$ , for example. An immediate inference rule which will be of importance below is called repetition. This rule permits one to repeat an earlier line. (Note that we are not using the word ‘immediate’ in the sense of “obvious.” Many of the nonimmediate inferences mentioned here are also obvious.)

The next principle to be noticed in such a survey is the rule which permits us to enter as a line in a proof one of the assumptions from which we wish to draw a conclusion—provided that we clearly note that it is an assumption (and not a conclusion on the basis of steps above it). This rule we call *assumption* and put it in a class by itself.

Notice that apart from the rule permitting the introduction of axioms a consequence system generally has only immediate inference rules and the rule of assumption. Since the introduction of logical axioms generally makes gaps in a proof we are not regarding principles of this sort as rigorous. The third type of rule to be noticed provides the key not only to giving rigorous proofs of the complex axioms of *CS* (which are never seen in normal reasoning) but also to giving rigorous proofs of simple axioms like  $(P \vee \sim P)$  and  $\sim(P \ \& \ \sim P)$  which do occur in normal reasoning. The third type of rigorous principle of proof is ubiquitous in normal reasoning but generally does not occur in consequence systems. It might even be maintained that it is the absence of this third type of rule which makes derivations in consequence systems unnatural and unrigorous.

The third type of principle of proofs we call a *structural rule*. The distinguishing feature of a structural rule is that it permits inference *not from previous lines* in a

proof but *on the basis of a pattern of reasoning*. Consider, for example, the following “proof” of  $((P \& Q) \supset (Q \& P))$  (from the empty set):

- Assume (1)  $(P \& Q)$ . From (1) we have  
 (2)  $Q$ . Again from (1) we have  
 (3)  $P$ . From (2) and (3) we have  
 (4)  $(Q \& P)$ .  
 Thus (5)  $((P \& Q) \supset (Q \& P))$ .

Notice that lines (2), (3), and (4) all depend on line (1) and all were immediate inferences from previous lines. Line (5), however, does not depend on line (1) nor on any other line or lines above it. Line (5) was inferred on the basis of a pattern of reasoning. On the basis of reasoning from  $(P \& Q)$  to  $(Q \& P)$  we inferred  $((P \& Q) \supset (Q \& P))$  and not as depending on any previous lines.

In general the principle of conditionalization permits the inference of  $(\phi \supset \delta)$  on the basis of reasoning from  $\phi$  to  $\delta$  and not as depending on  $\phi$  but only, perhaps, on *other previous assumptions* used in the reasoning. There are two structural rules which cover indirect proof. The first is called *negation introduction*: on the basis of reasoning from  $\phi$  to  $\delta$  and reasoning from  $\phi$  to  $\sim\delta$  we conclude  $\sim\phi$ . The second negation is called *negation elimination*: on the basis of reasoning from  $\sim\phi$  to  $\delta$  and the reasoning from  $\sim\phi$  to  $\sim\delta$  we conclude  $\phi$ . These are the principles behind “reductio” type proofs.

Below we will give “gapless” proofs of the three axioms of *LS* by using the rules: assumption (*A*), repetition (*R*), detachment or  $\supset$ -elimination ( $\supset E$ ), conditionalization or  $\supset$ -introduction ( $\supset I$ ), negation-elimination ( $\sim E$ ) and negation-introduction ( $\sim I$ ). We will use a plus sign (+) to mark assumptions and we will box off patterns of reasoning. For simplicity we require that each pattern of reasoning contain only one assumption and that the assumption be the first schema in the pattern.

(A1)	(1)	<div style="border: 1px solid black; padding: 2px; display: inline-block; vertical-align: middle;"> <div style="border: 1px solid black; padding: 2px; display: inline-block; vertical-align: middle;"> <math>+Q</math> <math>P</math> </div> </div>	$A$
	(2)	$+Q$	$A$
	(3)	$P$	$R$
	(4)	$(Q \supset P)$	$\supset I$
	(5)	$(P \supset (Q \supset P))$	$\supset I$

The idea of this proof is as follows. Since we want to prove  $(P \supset (Q \supset P))$  we could try assuming  $P$  and then reasoning to  $(Q \supset P)$ . Thus we assume  $P$  (line 1). Since we now want to prove  $(Q \supset P)$  from our assumption we could try assuming  $Q$  and reasoning to  $P$ . Thus we assume  $Q$  (line 2).  $P$  follows by repetition from our assumption  $P$  (line 3). We now have a pattern of reasoning from  $Q$  to  $P$ —so we box it off and conclude  $(Q \supset P)$  (line 4). Now we have a pattern of reasoning from  $P$  to  $(Q \supset P)$ —so we box it off and write the conclusion (line 5).

Notice that when we said that a box can contain one assumption we meant only one *unboxed* assumption—once some reasoning is boxed off the individual schemas in it no longer function as individual steps of the proof—inferences can no longer be made from them as they are now part of a pattern of reasoning.

(A2)	$+(P \supset (Q \supset R))$ <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: 80%;"> <math display="block">+(P \supset Q)</math> <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: 60%;"> <math display="block">+P</math> <math display="block">Q</math> <math display="block">Q \supset R</math> <math display="block">R</math> </div> <math display="block">(P \supset R)</math> </div> $((P \supset Q) \supset (P \supset R))$	$A$  $A$  $A$ $\supset E$ $\supset E$ $\supset E$  $\supset I$  $\supset I$
	$((P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R)))$	$\supset I$

(A3)	$+(\sim P \supset \sim Q)$ <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: 80%;"> <math display="block">+(\sim P \supset Q)</math> <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: 60%;"> <math display="block">+\sim P</math> <math display="block">Q</math> </div> <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: 60%;"> <math display="block">+\sim P</math> <math display="block">\sim Q</math> </div> <math display="block">P</math> </div> $((\sim P \supset Q) \supset P)$	$A$  $A$  $A$ $\supset EE$  $A$ $\supset E$  $\sim EE$  $\supset I$
	$((\sim P \supset \sim Q) \supset ((\sim P \supset Q) \supset P))$	$\supset I$

These proofs contain the maximum amount of logical detail. They are absolutely rigorous. Each time a new line was written in the proof there was no additional logical detail which could be asked for and none which could (nonredundantly) be supplied.

The reason that such apparently extravagant claims can be made is, very roughly: each lengthening of the proofs by the addition of a new line involved *either* an assumption *or* the introduction or elimination of exactly one logical symbol.

(b) *Deductive Systems*

We use the term deductive system to indicate a logical system which codifies logical truths, valid arguments and (sound, rigorous) proofs. The term deductive is chosen to indicate that the most comprehensive purpose of such a system is the codification of deduction. (Note: recently the term ‘deductive system’ has been used in a sense somewhat like the above but in the past it was used synonymously with the modern technical term ‘theory’ to denote a class of sentences which contains all of its own consequences relative to a consequence system.)

Below we will give as an example of a deductive system a system  $DS$  which (1) codifies the same logical truths of  $L$  as codified by  $LS$  (and  $CS$ ), which (2) codifies the same valid arguments of  $L$  as codified by  $CS$  and which (3) codifies the (sound, rigorous) proofs of  $L$  whose “correctness” turns on the symbols  $\supset$  and  $\sim$ . The latter will be accomplished by the codification of *proof forms* ( $PF$ s) which are discourse forms all of whose instances are proofs of the above sort. The forms referred to above as proofs of (A1), (A2), and (A3) are not themselves proofs in  $L$  because the symbols  $P$ ,  $Q$ , and  $R$  are schemas. But, when  $P$ ,  $Q$ , and  $R$  are replaced by sentences of  $L$  the results are instances of the forms and these instances indeed are proofs.

In order to define a deductive system which codifies proofs, valid arguments and logical truths one may (1) define proofs, (2) define proofs from a set of premises  $S$  to a conclusion  $p$ , (3) define a derivable argument as one for which there is a proof from its premises to its conclusion, and finally (4) define a theorem as a sentence  $p$  such that  $\Lambda \therefore p$  is a derivable argument.

Taking the term ‘*plussed schema*’ to indicate a schema preceded by a  $+$  sign and taking the term ‘*boxed schema*’ to indicate a schema enclosed in a box we offer the following four definitions as defining  $DS$ . In the inductive definition of  $PF$  we will indicate the clauses corresponding to the various principles of inference by using the abbreviation of the name of the principle.

Definition: A  $PF$  of the deductive system  $DS$  is defined:

- (i) (A) (a) a plussed schema is a  $PF$ 
  - (b) every  $PF$  may be extended to another  $PF$  by adding any plussed schema
- (ii) (R) every  $PF$  may be extended to another  $PF$  by adding any unboxed schema already occurring in it omitting the plus if the schema to be added was plussed.
- (iii) ( $\supset E$ ) any  $PF$  containing unboxed  $\phi$  and  $(\phi \supset \delta)$  (plussed or not) may be extended by adding  $\delta$ .
- (iv) any  $PF$  ending with a sequence which (1) begins with a plussed unboxed schema and (2) contains no other (unboxed) plussed schema may be extended to another  $PF$  by boxing off the latter sequence provided that it ends in an unboxed and unplussed schema.
- (v) ( $\supset I$ ) any  $PF$  ending with a boxed sequence which (1) begins with  $+\phi$  and (2) ends with  $\delta$  may be extended to another  $PF$  by adding  $(\phi \supset \delta)$ .
- (vi) ( $\sim I$ ) any  $PF$  ending in two successive boxed sequences one of which begins

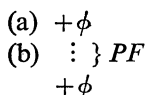
with  $+\phi$  and ends in  $\delta$  and the other of which begins with  $+\phi$  and ends with  $\sim\delta$  may be extended by adding  $\sim\phi$ .

- (vii) ( $\sim E$ ) any *PF* ending in two successive boxed sequences one of which begins with  $+\sim\phi$  and ends with  $\delta$  and the other of which begins with  $+\sim\phi$  and ends with  $\sim\delta$  may be extended by adding  $\phi$ .
- (viii) only those things constructible by a finite number of applications of the above rules are *PFs*.

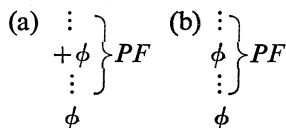
Note that there is no rule corresponding to the introduction of axioms. This logical system contains no axioms. The first clause ((i)a) says that every *PF* begins with an assumption. Clause (iv) is a rule for boxing off patterns of reasoning. The remaining clauses specify how a given proof may be lengthened by the addition of a new schema. These rules are diagrammed below.

Diagrams of Rules for Proof Forms in *DS*

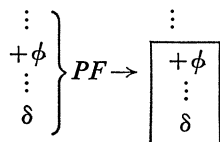
(i) Assumption: *A*



(ii) Repetition: *R*

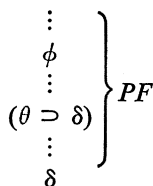


(iv) Boxing off patterns of reasoning

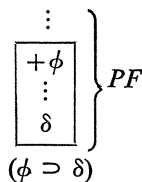


Note: The part of the proof being boxed off can contain only one unboxed assumption (plussed schema).

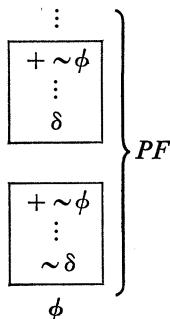
(iii) Detachment:  $\supset E$



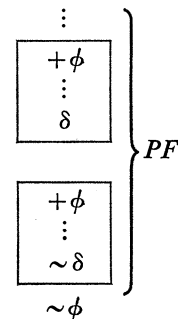
(v) Conditionalization:  $\supset I$



(vii) Reductio:  $\sim E$



(vi) Reductio:  $\sim I$





Now we define what it means for a *PF* to be a *PF* of  $\phi$  from  $\Sigma$  where  $\phi$  is a schema and  $\Sigma$  is a set of schemas.

**Definiton:** A *PF*, say  $\alpha$ , is a *PF* of  $\phi$  from  $\Sigma$  iff  $\phi$  is the last line of  $\alpha$  and  $\phi$  is unboxed and every unboxed plussed schema in  $\alpha$  is a member of  $\Sigma$ .

As an example we give a *PF* of  $(P \supset R)$  from  $[(P \supset (Q \supset R)), (P \supset Q)]$

+ $(P \supset (Q \supset R))$   
 + $(P \supset Q)$

+ $P$
$Q$
$(Q \supset R)$
$R$

$(P \supset R)$

By extending this *PF* as above we obtain a *PF* of (A2) and from the empty set, i.e. we obtain what corresponds to a proof of (A2) based on logical principles alone. In *CS* a derivation of (A2) from the empty set is merely a one line derivation:

$$((P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R)))$$

This seems to epitomize a gap in reasoning. If we ask for more logical detail then we must go to a system like *DS* as in *CS* there is no way to fill-in the gap without creating other gaps.

**Definition:**  $\Sigma \therefore \phi$  is a *derivable argument form* in *DS* iff there is a *PF* from  $\Sigma$  to  $\phi$ .

**Definition:**  $\phi$  is a *theorem* of *DS* iff  $\Lambda \therefore \phi$  is a derivable argument form in *DS*.

As far as this particular deductive system is concerned, soundness, completeness, strong soundness and strong completeness can easily be inferred from the strong soundness and strong completeness of *CS*. Since completeness and soundness follow, respectively, from strong completeness and strong soundness it is sufficient to prove these latter properties of *DS* from the properties of *CS*. Both proofs are instructive.

To show that *DS* is strongly complete we must show that every valid argument form is derivable in *DS*. From the strong completeness of *CS* we know that every valid argument form is derivable in *CS*. Thus it is sufficient to show that every argument form derivable in *CS* is also derivable in *DS*. Let  $\Sigma \therefore \phi$  be derivable in *CS*. Then there is a derivation of  $\phi$  from  $\Sigma$  in *CS*. Let  $\alpha$  be such a derivation. If we can show how to modify  $\alpha$  so that it becomes a *PF* of  $\phi$  from  $\Sigma$  in *DS* we are finished. Now  $\alpha$  may already be a *PF* of  $\phi$  from  $\Sigma$ . In this case nothing need be shown. But if  $\alpha$  is not a *PF* then there must be axioms in it—because gaps occur in derivations when and only when axioms are introduced; detachment and assumption are the only other ways of making derivations. *Replace each axiom in  $\alpha$  by a *PF* of that axiom* (as in the examples above). Thus  $\alpha$  is changed to  $\alpha'$  and  $\alpha'$  is now a *PF* as all gaps have been eliminated. Thus every argument form derivable in *CS* is also derivable in *DS*. Thus if *CS* is strongly complete so is *DS*.

To show that *DS* is strongly sound we must show that every derivable argument

form is valid. From the strong soundness of *CS* we know that every argument form derivable in *CS* is valid. Thus if we can show that every argument form derivable in *DS* is also derivable in *CS* we will be finished. What we must show is that the simple patterns of reasoning permitted by applications of *R*,  $\supset I$ ,  $\sim E$ , and  $\sim I$  in *DS* can be carried out by “chunks” of derivations in *CS*. In standard terminology, we must show that *R*,  $\supset I$ ,  $\sim E$ , and  $\sim I$  are *derived rules of inference* ([3], p. 83) in *CS*. To show that  $\supset I$  is a derived rule of inference in *CS* is to prove Herbrand’s famous deduction theorem, viz.: if there is a derivation in *CS* of  $\delta$  from assumptions  $\Sigma$  and  $\phi$  then there is a derivation in *CS* of  $(\phi \supset \delta)$  from  $\Sigma$  alone. In other words, if  $\Sigma' \therefore \delta$  is derivable in *CS* then so is  $\Sigma \therefore (\phi \supset \delta)$  where  $\Sigma'$  contains  $\phi$  and  $\Sigma$  is obtained from  $\Sigma'$  by deleting  $\phi$ . Actually, to show that  $\supset I$  is a derived rule the proof must do a bit more; the proof must give a *uniform procedure* which, when applied to a derivation of  $\delta$  from  $\Sigma$  and  $\phi$ , produces a derivation of  $(\phi \supset \delta)$  from  $\Sigma$  above. This can be done very easily by well-known methods (e.g. as in [12], p. 32). Similar methods can be used to show that *R*,  $\sim E$ , and  $\sim I$  are derived rules. As an example let us see how repetition (*R*) can be obtained as a derived rule. We will give a uniform method which will produce from a given derivation  $\alpha$  having a line  $\phi$  another derivation  $\alpha'$  among whose lines all lines of  $\alpha$  are found and whose *last line* is  $\phi$ . To the given derivation add a derivation of  $(\phi \supset \phi)$  (as above) and to this derivation add  $\phi$ . The result is a derivation because the last line  $\phi$  was obtained by detachment from the postulated earlier line  $\phi$  and the line  $(\phi \supset \phi)$ . Using these results one can show that any argument form derivable in *DS* can also be derived in *CS* and (by the strong soundness of *CS*) is, therefore, valid.

Actually it is easier, but perhaps less instructive, to prove the strong soundness of *DS* directly by mathematical induction on the length of a *PF*.

Thus *DS* is sound, complete, strongly sound and strongly complete. Also, needless to say, *DS* is effective.

If *DS* is to be thought of as more than a mere consequence system there must be stronger adequacy conditions satisfied by its derivations (*PFs*). Otherwise we have no *precisely articulated reason* for regarding it as anything but a different kind of consequence system. It is not enough that its derivations are sound and effective; they must be rigorous as well, i.e. they must contain a maximal amount of relevant logical detail. Intuitively, of course *PFs* satisfy this condition—but since the concept of a rigorous *PF* has not been precisely defined we can not prove that *PFs* are rigorous.

Here we find ourselves in the same kind of position as that occupied by logicians before Tarski defined the concept of logical consequence or valid argument: they knew intuitively that some of their systems were strongly sound but lacking the definition of validity they could not offer proof.

Assuming that we understand what is meant by the term rigorous we notice that since *DS* is intended to codify rigorous proofs it is desirable that *DS* codify *all* of them. We may define a deductive system to be *deductively complete* if it codifies *all* rigorous proofs (of the appropriate kind). Now the need for a definition of a rigorous proof (or rigorous *PF* in the case of *DS*) becomes all the more pressing because we would like to be able to show either that *DS* is deductively complete or that it is not. Lacking the definition we can only resort to plausibility arguments.

We suggest that *DS* is *not* deductively complete because it fails to codify the following two kinds of rigorous proofs: (1) proofs from  $\Sigma$  to  $\phi$  which, after the assumptions in  $\Sigma$  have been entered, infer  $\phi$  on the basis of a pattern of reasoning beginning with a new assumption  $\sim\phi$  and ending with  $\phi$ ; (2) proofs from  $\Sigma$  to  $\sim\phi$  which, after the assumptions in  $\Sigma$  have been entered, infer  $\sim\phi$  on the basis of a pattern of reasoning beginning with a new assumption  $\phi$  and ending with  $\sim\phi$ .

Examples of these respective kinds of proofs follow.

- |                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                              |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>(1) <math>[(\sim P \supset P)] \therefore P</math></p> <p>1. <math>+(\sim P \supset P)</math> (in <math>\Sigma</math>)</p> <p>2. <math>\boxed{+\sim P}</math> (<math>\sim\phi</math>)</p> <p>3. <math>\boxed{P}</math></p> <p>4. <math>P</math></p> | <p>(2) <math>[(P \supset \sim P)] \therefore \sim P</math></p> <p>1. <math>+(P \supset \sim P)</math> (in <math>\Sigma</math>)</p> <p>2. <math>\boxed{+P}</math> (<math>\phi</math>)</p> <p>3. <math>\boxed{\sim P}</math></p> <p>4. <math>\sim P</math></p> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Proofs of these sorts can easily be seen as “abbreviations” of *PFs* in *DS*. Put (a) (below) between lines 3 and 4 of (1) and put (b) (below) between lines 3 and 4 of (2)—the results are *PFs* in *DS*.

- |                                                             |                                                   |
|-------------------------------------------------------------|---------------------------------------------------|
| (a) $\boxed{\begin{matrix} +\sim P \\ \sim P \end{matrix}}$ | (b) $\boxed{\begin{matrix} +P \\ P \end{matrix}}$ |
|-------------------------------------------------------------|---------------------------------------------------|

However, the principles of reasoning in (1) and (2) as they stand are sound and are part of every (classical) mathematician’s reasoning. Moreover, intuitively these proofs are rigorous.

In general, a deductive system codifies logical truths, valid arguments and sound, rigorous proofs. Thus it is necessary that a deductive system be effective, sound, strongly sound and rigorous. Moreover, it is desirable that a deductive system be complete, strongly complete and deductively complete. As we have seen, the rationale for each of these conditions is to be found in the intended purposes of a deductive system.

In table I we have summarized a comparison of the three types of systems.

Table I  
Comparison of Types of Systems

Logical systems	Concern	Necessary properties	Desirable properties
Logistic	Codification Logical truth	Effective Sound	Complete (Post complete)
Consequence	Codification Logical truth Valid arguments	Effective Sound Strongly sound	Complete Strongly complete
Deductive	Codification Logical truth Valid argument Proof	Effective Sound Strongly sound Rigorous	Complete Strongly complete Deductively complete

In passing we note two things. First: some logicians may find it desirable to formulate concepts of maximal strong completeness and maximal deductive completeness which bear, respectively, to strong completeness and deductive completeness a relation like that which Post completeness bears to completeness. Second: in his thoughtful history of logic ([10], p. 535), Kneale has suggested that logic should be concerned to codify rules of inference. This suggestion entails a fourth type of logical system naturally to be referred to as an *inferential system*. In this connection *DS* could be misleading because in codifying valid arguments *by* codifying valid argument forms it seems to be codifying rules of inference. Notice, however (1) that no structural rules of inference could be thought of as mere valid argument forms and (2) that although we have defined the concept of valid argument form we have not formulated a general definition of a rule of inference.

#### 4. Rigorous Rules and Structural Rules.

*Rigorous Deduction:* Above we have emphasized that logic is concerned to codify proofs as well as logical truths and valid arguments. We have suggested that the term 'deductive system' be used to indicate a logical system which is intended to accomplish these three aims. We also suggested that a deductive system should not aim to codify all proofs of a language but only the sound and rigorous ones. The codification of mistakes in reasoning and the codification of great leaps of logical intuition are both eschewed.

Thus in the discussion of purposes of deductive systems and, consequently, in the formulation of appropriate adequacy conditions for them, the terms 'sound' and 'rigorous' are prominent. As a result of Tarski's explication of the concept of logical consequence the concept of soundness is no longer problematic. However, the concept of *rigor* lacks a similar careful explication. One purpose of this section is to suggest lines along which such an explication may be constructed.

In investigation of deductive systems a definition of "a rigorous proof" is needed. In such studies it is desirable to be able to answer two types of questions. First, is a deductive system rigorous, i.e. is every proof codified by a given system a rigorous proof? Second, is a deductive system deductively complete, i.e. is every rigorous proof actually codified by the system? Below we shall not concern ourselves directly with the concept of a *rigorous proof* but rather with the related concept of a *rigorous rule of inference*. Moreover, we will restrict our discussion to rules used for inference in the strict sense, i.e. we will exclude from consideration the rules of assumption and repetition.

In the first place a rule should be sound in order for it to be regarded as rigorous. Moreover, it should be effective in order to be regarded as a rule at all. Over and above soundness and effectiveness there seem to be two distinct ideas involved in rigor. In the first place a rigorous rule should issue in proofs which contain a maximum amount of relevant logical detail—a rigorous rule should not permit gaps in reasoning. In the second place a rigorous rule should be simple in some sense.

We suggest that a maximum amount of logical detail is achieved by rules which either introduce or eliminate exactly one occurrence of a logical symbol. For example, detachment eliminates one occurrence of  $\supset$  and conditionalization intro-

duces one occurrence, while *modus tollens* (from  $\sim\delta$  and  $(\phi \supset \delta)$  infer  $\sim\phi$ ) eliminates one occurrence of  $\sim$  and one occurrence of  $\supset$  while introducing one occurrence of  $\sim$ . Contraposition (on the basis of a pattern of reasoning from  $\sim\delta$  to  $\sim\phi$  infer  $(\phi \supset \delta)$ ) eliminates two occurrences of  $\sim$  and introduces one occurrence of  $\supset$ . The rules of double negation (from  $\sim\sim\phi$  infer  $\phi$ , from  $\phi$  infer  $\sim\sim\phi$ ) also “turn on” two symbol occurrences.

Simplicity, on the other hand, seems to be exemplified by rules which involve only one *kind* of logical symbol. Modus ponens, detachment, and negation-introduction and negation-elimination have both of these properties, i.e. each of these rules (1) either introduces or eliminates exactly one occurrence of a logical symbol and (2) involves only one kind of logical symbol. With the exception of the double negation rules which involve only  $\sim$ , the other rules just mentioned above have neither of these properties. It might be true that every rule introducing or eliminating exactly one occurrence of a logical symbol also involves in its application (or statement) only one logical symbol.

As a sketch of an explication of the concept of a rigorous rule of inference we offer the following:

A rule of inference is rigorous iff (1) it is effective, (2) it is sound, (3) it introduces or eliminates exactly one occurrence of a logical symbol (and not both), and (4) its application involves only one kind of logical symbol.

As an aside we want to point out that the condition of involving only one logical symbol has some interesting consequences. In the development of a deductive system, assumption and repetition may be put down first. Thereafter each logical symbol may be treated in complete independence of the other symbols. For each logical symbol in the language under consideration one may introduce one introduction rule and one elimination rule without reference to any other logical symbol in the language.

Since in the specification of the rules of the deductive system there is no interaction among the logical symbols it is possible to regard a complex deductive system as a product of simple, mutually independent deductive systems. Each of these simple systems consists in one elimination rule, one introduction rule, the rule of assumption and the rule of repetition. *DS* is the product of a simple system for negation and simple system for ‘if . . . then.’ Questions of completeness and strong completeness can be asked of each simple system. By factoring a complex system into simple systems it might be possible to isolate the “causes” of various undesirable features such as undecidability, incompleteness, or noncompactness.

Another possible application of this idea is in providing a more intuitive point of view for understanding the result of Hiž [8] concerning the construction of a consequence system based on one set of logical symbols by translating the rules of a system based on another. Hiž indicates by example that by translating a complete system one may obtain an incomplete system. The underlying idea is that in translating a rule of inference soundness is preserved but “deductive power” is not.

In a rigorous system the deductive power may be thought of as the ability to introduce a symbol as a principal symbol and the ability to eliminate a symbol from

the position of a principal symbol. Since every definition of a logical symbol involves at least two other logical symbols, the translation of any rigorous system must result in an unrigorous system. For example, if we translate detachment using  $(P \supset Q) =_{df} \sim(P \& \sim Q)$  then we get as an elimination rule the following: from  $P$  and  $\sim(P \& \sim Q)$  infer  $Q$ . Thus by translating the rigorous rule of detachment we obtain a rule which eliminates two occurrences of  $\sim$  and one occurrence of  $\&$  and which, moreover, involves in its statement two different logical symbols. Furthermore, while detachment has the power of eliminating  $\supset$  from a principal position the new rule can eliminate  $\&$  only when it is embedded in the complex structure  $\sim(P \& \sim Q)$ ! Translating conditionalization we get the rule: on the basis of a pattern of reasoning from  $P$  to  $Q$  infer  $\sim(P \& \sim Q)$ . This rule is not rigorous. Moreover it is inherently weaker than conditionalization because it does not permit the introduction of  $\&$  into the principal position.

Consider the deductive system based on the connectives  $\&$  and  $\sim$  whose rules are the same as for *DS* except for the two translations above. Because the  $\&$ -elimination rule only eliminates  $\&$  from the context  $\sim(P \& \sim Q)$  one is led to conjecture that in this system one cannot prove

$$[(P \& Q)] \therefore P$$

Indeed, the following two matrices provide a sound three-valued semantics for this system in which an interpretation of  $P$  as 1 and  $Q$  as 1 makes  $(P \& Q)$  true (designated) and  $Q$  false (undesignated).

$\&$	2	1	0	$\sim$
2*	2	0	0	0
1	0	2	0	2
0	0	0	0	2

Another interesting application is to experimentation in modal logics. Given a nonmodal logic based on a language having modal operators one may form the modal rules by introducing two rigorous rules for each operator. The modal problems may be thought of in isolation from the rest of the system. Where applicable, this approach greatly facilitates experimentation with modal systems.

It should be emphasized that these considerations are secondary to the main purpose of this paper which was to investigate logical systems from the viewpoint of the problems which they may reasonably be thought of as solving. Logistic systems are obviously intended to codify logical truths. Consequence systems are obviously intended to codify valid arguments. The conditions that these two types of systems must satisfy in order to be completely adequate are already well-known. Deductive systems are obviously intended for an additional purpose, viz., the codification of proofs. Being faced with the problem of formulating the conditions of adequacy of deductive systems we were led to conjecture the existence of simple and "gapless" proofs. This, in turn, issued in the formulation of the concept of rigor introduced above.

Our formulation of the concept of rigor is defective in at least two ways.

In the first place, although the basic idea “rigorous” involves effectiveness, soundness, simplicity and maximality of relevant logical detail and, in fact, our definition has components corresponding to each—our definition still seems to be too specific. What we should like to have is a more general concept. In fact, we should like to have a concept of *rigor* which (1) is so general that it would require a mathematical proof to show that the system *DS* is rigorous and which (2) obviously characterizes the idea.

Secondly, and perhaps unimportantly, we have not given a mathematically precise definition of what it means to “introduce” or to “eliminate” a logical symbol. The intuitive idea is perfectly clear and it is easy to give for each symbol an ad hoc definition of introducing or eliminating it. But it would be much better to have a general definition which did not refer to any specific symbol and from which the ad hoc definitions follow.

These deficiencies are not germane to the main purpose of the paper. In regard to our present purpose it is enough that we have brought to light the necessity for some precise formulation of adequacy conditions for the concept of proof—over and above mere soundness and effectiveness. Moreover, it should also be emphasized that the above suggested applications for our idea of “rigorous” still stand regardless of whether the idea proves appropriate to its intended purpose.

*Structural Rules of Inference:* The distinction between immediate inference rules and structural inference rules was introduced above. Assumption is not being considered a rule of inference in this discussion because it does not, in itself, permit inferences. Each inference rule, whether discussed above or not, *either* is applied on the basis of a fixed number of lines *or* is applied on the basis of a pattern of reasoning whose length can not be fixed in advance. For example, detachment is always applied to two lines whereas conditionalization is applied on the basis of reasoning from  $\phi$  to  $\delta$  and this reasoning may be of any finite length. Similarly, existential generalization is always applied to one line and has no dependence on the rest of the proof in which it is applied, whereas existential instantiation permits the inference of  $\delta$  (containing no free  $x$ ) in a proof (none of whose assumptions contain free  $x$ ) on the basis of: (1) a line  $(\exists x) \phi$  and (2) a pattern of reasoning from  $\phi$  to  $\delta$ . Coordinately, universal instantiation permits inference on the basis of a single line, whereas in order to apply universal generalization the entire proof must be considered. (See inside covers of [1].)

For a rule of inference to be structural it seems to be necessary and sufficient that the number of lines on the basis of which the rule can be applied is not fixed. This provides a criterion for classifying rules as structural.

By going to a corpus of deductive reasoning or by unselfconsciously writing some proofs oneself (if one happens to be in the habit of proving things) it is possible to discover the rationale of the patterns of reasoning involved in these rules. The important idea in structural rules is the pattern of reasoning—the necessary and sufficient conditions given above are mere grammatical criteria.

Apparently, the structural nature of some rules of inference has not been explicitly recognized before. Indeed, it might be possible to explain the difficulties

experienced by competent logicians (see Foreword to Revised Edition [17] and also [15]) in trying to formulate certain structural rules by noticing a tendency to express structural rules in the form of immediate inference rules with restrictions.

### 5. Conclusion.

*Another Hierarchy:* It should be pointed out that the hierarchy of systems discussed above is completely independent of the following hierarchy: propositional logic, monadic predicate logic, predicate logic, second order logic, etc. This latter hierarchy concerns the depth of logical analysis: there are sentences analyzable as logical truths in second order logic but not so analyzable in predicate logic, there are sentences analyzable as logical truths in predicate logic but not so analyzable in monadic predicate logic, similarly for monadic logic compared to propositional logic. Below are examples.

- (1) Everything has a property.  
 Second order:  $(\forall x)(\exists P)Px$   
 Predicate:  $(\forall x)Hx$
- (2) If there is a cause of everything then everything has a cause.  
 Predicate:  $((\exists x)(\forall y) Cxy \supset (\forall y)(\exists x)Cxy)$   
 Monadic:  $((\exists x) Cx \supset (\forall y)Hy)$
- (3) If all men are Greek, then if Socrates is a man, Socrates is Greek.  
 Monadic:  $((\forall x)(Mx \supset Gx) \supset (Ms \supset Gs))$   
 Propositional:  $(A \supset (M \supset G))$

This hierarchy, of course, extends upward and can be repeated for modal logics. The problems raised above were raised *in general* (although we took our examples from a very simple propositional logic). Needless to say, all of the observations concerning logical systems apply at each level of the hierarchies just mentioned.

*Hierarchy of Logical Systems:* We have seen three increasingly complex types of logical systems. The logistic systems were seen to be appropriate to the problem of codifying logical truths. The consequence systems codified logical truths but were also appropriate to the task of codifying valid arguments although this task was performable by logistic systems only with the greatest difficulty. Finally, the deductive system codified proofs in addition to logical truths and valid arguments.

Paralleling the increasing comprehensiveness of the three types of systems are the increasingly comprehensive adequacy conditions for the systems. The ideal logistic system is sound and complete. The ideal consequence system is strongly sound and strongly complete. Since soundness and completeness are special cases respectively of strong soundness and strong completeness the consequence systems are inherently more comprehensive than the logistic systems. Moreover, the distinction between these two types of systems is underlined by the fact that a logistic system might reasonably be expected to satisfy a condition which conflicts with strong soundness—viz. Post completeness. For clear rationales of the adequacy conditions for logistic systems and consequence systems one may refer to their respective intended aims.



By considering actual deductive practice we are led to distinguish between valid arguments and proofs. Thus we may set as a reasonable aim the codification of proofs and use the term 'deductive system' to denote a system which has this purpose. From an informal but thoughtful point of view we are led to reject consequence systems as codifications of proofs for two reasons—in the first place, they treat intuitively rigorous proofs (e.g. those using conditionalization or existential instantiation) as having gaps; in the second place, derivations from premises in consequence systems themselves have unmistakable gaps. *Moreover, necessary conditions generally agreed upon for consequence systems, viz. strong soundness and effectiveness, have no reference to any idea of rigor.* Thus we are faced with the problem of formulating a mathematical condition (1) that proofs must satisfy in order to be rigorous and (2) which goes beyond mere strong soundness and effectiveness.

For the purposes of this conclusion it is beside the point whether the concept of rigor introduced above is correct. On the basis of (1) the fact that codification of proofs is a reasonable aim and (2) the obvious but unscientific fact that consequence systems do not fulfill this purpose—we conclude that a third type of system is needed.

*Historical Remarks:* We would like to conclude with a brief consideration of the relationship between the above hierarchy of logical systems and the actual historical development of them in modern times. This consideration can not be thought of as a sketch of the history of modern logic; the questions considered above constitute an important but relatively small part of modern logic.

Within this part of the history of logic we focus on two interrelated themes: first, the construction of logical systems; second, the formulation of adequacy conditions for the systems. Obviously, the conclusions we would like to be able to make are the following:

(1) In regard to systems themselves; logistic systems, consequence systems, and deductive systems were developed sequentially in that order.

(2) In regard to adequacy conditions; first the concepts of completeness, soundness and Post completeness were formulated, next strong soundness and strong completeness, finally the concept of rigorous was dealt with.

Our criterion for priority is date-of-publication; use of a more sophisticated criterion would involve considerable historical research.

The earliest systems were certainly logistic systems. For Frege and Russell, the motivation was the belief that all theorems of mathematics are "logical laws." In 1934 the first investigations into the nature of proof were published ([10], pp. 538 and 539). One is due to Gentzen, the other to Jaskowski. Subsequent development in this area has been based on these two works ([10], p. 539 and [1], p. v). In order to establish that consequence systems were developed before 1934 it is enough to cite Tarski's important methodological study of consequence systems which was first published in 1930 ([19], pp. 30–37).

The adequacy conditions for logistic systems were formulated by Post [16] in 1920. Post discussed soundness, completeness and Post completeness (though not

by these names). Apparently strong soundness and strong completeness were first discussed in print by Abraham Robinson in 1951 ([18], Theorem 3.1.2 and Theorem 3.2.2, respectively). This is a truly surprising conclusion in view of the importance of these concepts; however, it is probably true that many logicians had used these concepts in their thinking and in their teaching before 1951. In this regard it is worth noting that theorems trivially implying the strong completeness of first order logic were published by Gödel [5] in 1931 and by Henkin [6] in 1949. But neither of them mention strong completeness despite the fact that this theorem is conceptually more interesting than the theorems they actually proved. As far as I have been able to determine, apart from this paper, there is no published consideration of any conditions for proofs over and above effectiveness and soundness.

*Acknowledgments:* It is a pleasure to acknowledge interesting and stimulating discussions of an earlier version of this paper with my colleagues, Mr. James Munz and Mr. William Snively. I would also like to thank the following students for various kinds of contributions to this work: Uwe Henke, Nicolas Noviello, Howard Wasserman, and George Weaver. This work was originally written as a lecture given in December, 1966 to the Department of Philosophy, University of Pennsylvania.

## REFERENCES

- [1] Anderson, J. M., and Johnstone, H. W., *Natural Deduction*, Belmont, California, 1962.
- [2] Bochenski, I. M., *A History of Formal Logic* (tr. Thomas, Ivo), Notre Dame, Indiana, 1961.
- [3] Church, A., *Introduction to Mathematical Logic*, Princeton, 1956.
- [4] Copi, I. M., and Gould, J. A., *Readings on Logic*, New York, 1964.
- [5] Gödel, K., "Die Vollständigkeit der Axiome des logischen Funktionenkalküls," *Monatshefte für Mathematik und Physik*, vol. xxxvii, 1930, p. 349.
- [6] Henkin, L., "The completeness of the first order functional calculus," *Journal of Symbolic Logic*, vol. 14, 1949, p. 159.
- [7] Hilbert, D., and Ackermann, W., *Principles of Mathematical Logic* (tr. Hammond, Leckie, and Steinhardt), New York, 1950.
- [8] Hiž, H., "A warning about translating axioms," *American Mathematical Monthly*, vol. LXV, 1958, p. 613.
- [9] Kalish, D., and Montague, R., *Logic: Techniques of Formal Reasoning*, New York, 1964.
- [10] Kneale, W., and Kneale, M., *The Development of Logic*, Oxford, 1962.
- [11] Lewis, C. I., and Langford, C. H., *Symbolic Logic*, 2nd ed., New York, 1959.
- [12] Lightstone, A. H., *The Axiomatic Method*, Prentice-Hall, Englewood Cliffs, N.J., 1964.
- [13] Mates, B., *Elementary Logic*, New York, 1965.
- [14] Mendelson, Elliot, *Introduction to Mathematical Logic*, Princeton, 1964.
- [15] Parry, W. T., "Comments on a variant form of natural deduction," *Journal of Symbolic Logic*, vol. 30, 1965, p. 119.
- [16] Post, E. L., "Introduction to general theory of elementary propositions," *American Journal of Mathematics*, vol. 43, 1921, p. 163.
- [17] Quine, W. V. O., *Methods of Logic* (revised edition), New York, 1959.
- [18] Robinson, A., *On the Metamathematics of Algebra*, Amsterdam, 1951.
- [19] Tarski, A., *Logic, Semantics and Metamathematics* (tr. Woodger, J. H.), Oxford, 1956.
- [20] Whitehead, A. N., and Russell, B., *Principia: Mathematica to 56*, Cambridge, 1962.

## LINKED CITATIONS

- Page 1 of 1 -



You have printed the following article:

### **Three Logical Theories**

John Corcoran

*Philosophy of Science*, Vol. 36, No. 2. (Jun., 1969), pp. 153-177.

Stable URL:

<http://links.jstor.org/sici?sici=0031-8248%28196906%2936%3A2%3C153%3ATLT%3E2.0.CO%3B2-M>

---

*This article references the following linked citations. If you are trying to access articles from an off-campus location, you may be required to first logon via your library web site to access JSTOR. Please visit your library's website or contact a librarian to learn about options for remote access to JSTOR.*

## References

### <sup>6</sup> **The Completeness of the First-Order Functional Calculus**

Leon Henkin

*The Journal of Symbolic Logic*, Vol. 14, No. 3. (Sep., 1949), pp. 159-166.

Stable URL:

<http://links.jstor.org/sici?sici=0022-4812%28194909%2914%3A3%3C159%3ATCOTFF%3E2.0.CO%3B2-9>

### <sup>15</sup> **Comments on a Variant Form of Natural Deduction**

William Tuthill Parry

*The Journal of Symbolic Logic*, Vol. 30, No. 2. (Jun., 1965), pp. 119-122.

Stable URL:

<http://links.jstor.org/sici?sici=0022-4812%28196506%2930%3A2%3C119%3ACOA VFO%3E2.0.CO%3B2-1>

### <sup>16</sup> **Introduction to a General Theory of Elementary Propositions**

Emil L. Post

*American Journal of Mathematics*, Vol. 43, No. 3. (Jul., 1921), pp. 163-185.

Stable URL:

<http://links.jstor.org/sici?sici=0002-9327%28192107%2943%3A3%3C163%3AITAGTO%3E2.0.CO%3B2-T>

**NOTE:** *The reference numbering from the original has been maintained in this citation list.*