An Introduction to Non-Classical Logic

From If to Is

Second Edition

GRAHAM PRIEST

University of Melbourne

and

University of St Andrews

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE UNIVERSITY PRESS
Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore, São Paulo

Cambridge University Press
The Edinburgh Building, Cambridge CB2 8RU, UK
Published in the United States of America by Cambridge University Press, New York

www.cambridge.org
Information on this title: www.cambridge.org/9780521854337

First published in print format 2008

ISBN-13 978-0-511-39361-7 eBook (EBL)

ISBN-13 978-0-521-85433-7 hardback

ISBN-13 978-0-521-67026-5 paperback

1 Classical Logic and the Material Conditional

1.1 Introduction

1.1.1 The first purpose of this chapter is to review classical propositional logic, including semantic tableaux. The chapter also sets out some basic terminology and notational conventions for the rest of the book.

1.1.2 In the second half of the chapter we also look at the notion of the conditional that classical propositional logic gives, and, specifically, at some of its shortcomings.

1.1.3 The point of logic is to give an account of the notion of validity: what follows from what. Standardly, validity is defined for inferences couched in a formal language, a language with a well-defined vocabulary and grammar, the *object language*. The relationship of the symbols of the formal language to the words of the vernacular, English in this case, is always an important issue.

1.1.4 Accounts of validity themselves are in a language that is normally distinct from the object language. This is called the *metalanguage*. In our case, this is simply mathematical English. Note that 'iff' means 'if and only if'.

1.1.5 It is also standard to define two notions of validity. The first is *semantic*. A valid inference is one that *preserves truth*, in a certain sense. Specifically, every interpretation (that is, crudely, a way of assigning truth values) that makes all the premises true makes the conclusion true. We use the metalinguistic symbol '$\models$' for this. What distinguishes different logics is the different notions of interpretation they employ.

1.1.6 The second notion of validity is *proof-theoretic*. Validity is defined in terms of some purely formal procedure (that is, one that makes reference only to the symbols of the inference). We use the metalinguistic symbol '$\vdash$' for this notion of validity. In our case, this procedure will (mainly) be one employing tableaux. What distinguish different logics here are the different tableau procedures employed.

1.1.7 Most contemporary logicians would take the semantic notion of validity to be more fundamental than the proof-theoretic one, though the matter is certainly debatable. However, given a semantic notion of validity, it is always useful to have a proof-theoretic notion that corresponds to it, in the sense that the two definitions always give the same answers. If every proof-theoretically valid inference is semantically valid (so that $\vdash$ entails $\models$) the proof-theory is said to be *sound*. If every semantically valid inference is proof-theoretically valid (so that $\models$ entails $\vdash$) the proof-theory is said to be *complete*.

1.2 The Syntax of the Object Language

1.2.1 The symbols of the object language of the propositional calculus are an infinite number of propositional parameters:[1] $p_0, p_1, p_2, \ldots$; the connectives: $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), $\supset$ (material conditional), $\equiv$ (material equivalence); and the punctuation marks: (,).

1.2.2 The (well-formed) formulas of the language comprise all, and only, the strings of symbols that can be generated recursively from the propositional parameters by the following rule:

If A and B are formulas, so are $\neg A$, $(A \vee B)$, $(A \wedge B)$, $(A \supset B)$, $(A \equiv B)$.

1.2.3 I will explain a number of important notational conventions here. I use capital Roman letters, $A, B, C, \ldots$, to represent arbitrary formulas of the object language. Lower-case Roman letters, p, q, r, $\ldots$, represent arbitrary,

[1] These are often called 'propositional variables'.

but distinct, propositional parameters. I will always omit outermost parentheses of formulas if there are any. So, for example, I write $(A \supset (B \vee \neg C))$ simply as $A \supset (B \vee \neg C)$. Upper-case Greek letters, $\Sigma, \Pi, \ldots$, represent arbitrary sets of formulas; the empty set, however, is denoted by the (lower case) ϕ, in the standard way. I often write a finite set, $\{A_1, A_2, \ldots, A_n\}$, simply as $A_1, A_2, \ldots, A_n$.

1.3 Semantic Validity

1.3.1 An *interpretation* of the language is a function, ν, which assigns to each propositional parameter either 1 (true), or 0 (false). Thus, we write things such as $\nu(p) = 1$ and $\nu(q) = 0$.

1.3.2 Given an interpretation of the language, ν, this is extended to a function that assigns every formula a truth value, by the following recursive clauses, which mirror the syntactic recursive clauses:[2]

$\nu(\neg A) = 1$ if $\nu(A) = 0$, and 0 otherwise.
$\nu(A \wedge B) = 1$ if $\nu(A) = \nu(B) = 1$, and 0 otherwise.
$\nu(A \vee B) = 1$ if $\nu(A) = 1$ or $\nu(B) = 1$, and 0 otherwise.
$\nu(A \supset B) = 1$ if $\nu(A) = 0$ or $\nu(B) = 1$, and 0 otherwise.
$\nu(A \equiv B) = 1$ if $\nu(A) = \nu(B)$, and 0 otherwise.

1.3.3 Let Σ be any set of formulas (the premises); then A (the conclusion) is a *semantic consequence* of Σ ($\Sigma \models A$) iff there is no interpretation that makes all the members of Σ true and A false, that is, every interpretation that makes all the members of Σ true makes A true. '$\Sigma \not\models A$' means that it is not the case that $\Sigma \models A$.

1.3.4 A is a *logical truth* (*tautology*) ($\models A$) iff it is a semantic consequence of the empty set of premises ($\phi \models A$), that is, every interpretation makes A true.

[2] The reader might be more familiar with the information contained in these clauses when it is depicted in the form of a table, usually called a *truth table*, such as the one for *conjunction* displayed:

$\wedge$	**1**	**0**
1	1	0
0	0	0

1.4 Tableaux

1.4.1 A *tree* is a structure that looks, generally, like this:[3]

$$
\begin{tikzcd}
& \cdot \arrow[d] & & \\
& \cdot \arrow[dl] \arrow[dr] & & \\
\cdot \arrow[d] & & \cdot \arrow[dl] \arrow[dr] & \\
\cdot & \cdot & & \cdot
\end{tikzcd}
$$

The dots are called *nodes*. The node at the top is called the *root*. The nodes at the bottom are called *tips*. Any path from the root down a series of arrows as far as you can go is called a *branch*. (Later on we will have trees with infinite branches, but not yet.)

1.4.2 To test an inference for validity, we construct a tableau which begins with a single branch at whose nodes occur the premises (if there are any) and the negation of the conclusion. We will call this the *initial list*. We then apply rules which allow us to extend this branch. The rules for the conditional are as follows:

$$
\begin{tikzcd}
& A \supset B \arrow[dl] \arrow[dr] & \\
\neg A & & B
\end{tikzcd}
\qquad
\begin{tikzcd}
\neg(A \supset B) \arrow[d] \\
A \arrow[d] \\
\neg B
\end{tikzcd}
$$

The rule on the right is to be interpreted as follows. If we have a formula $\neg(A \supset B)$ at a node, then every branch that goes through that node is extended with two further nodes, one for A and one for $\neg B$. The rule on the left is interpreted similarly: if we have a formula $A \supset B$ at a node, then every branch that goes through that node is split at its tip into two branches; one contains a node for $\neg A$; the other contains a node for B.

[3] Strictly speaking, for those who want the precise mathematical definition, it is a partial order with a unique maximum element, x_0, such that for any element, x_n, there is a unique finite chain of elements $x_n \leq x_{n-1} \leq \cdots \leq x_1 \leq x_0$.

1.4.3 For example, to test the inference whose premises are $A \supset B$, $B \supset C$, and whose conclusion is $A \supset C$, we construct the following tree:

$$
\begin{array}{ccccccc}
 & & & A \supset B & & & \\
 & & & \downarrow & & & \\
 & & & B \supset C & & & \\
 & & & \downarrow & & & \\
 & & & \neg(A \supset C) & & & \\
 & & & \downarrow & & & \\
 & & & A & & & \\
 & & & \downarrow & & & \\
 & & & \neg C & & & \\
 & & \swarrow & & \searrow & & \\
 & \neg A & & & & B & \\
\swarrow & \downarrow & & & & \downarrow & \searrow \\
\neg B & C & & & & \neg B & C \\
\times & \times & & & & \times & \times
\end{array}
$$

The first three formulas are the premises and negated conclusion. The next two formulas are produced by the rule for the negated conditional applied to the negated conclusion; the first split on the branch is produced by applying the rule for the conditional to the first premise; the next splits are produced by applying the same rule to the second premise. (Ignore the '$\times$'s: we will come back to those in a moment.)

1.4.4 The other connectives also have rules, which are as follows.

$$
\begin{array}{c}
\neg\neg A \\
\downarrow \\
A
\end{array}
$$

$$
\begin{array}{ccc}
 & A \vee B & \\
\swarrow & & \searrow \\
A & & B
\end{array}
\qquad
\begin{array}{c}
\neg(A \vee B) \\
\downarrow \\
\neg A \\
\downarrow \\
\neg B
\end{array}
$$

$$
\begin{array}{ccc}
 & \neg(A \wedge B) & \\
\swarrow & & \searrow \\
\neg A & & \neg B
\end{array}
\qquad
\begin{array}{c}
A \wedge B \\
\downarrow \\
A \\
\downarrow \\
B
\end{array}
$$

$$
\begin{array}{ccc}
 & A \equiv B & \\
\swarrow & & \searrow \\
A & & \neg A \\
\downarrow & & \downarrow \\
B & & \neg B
\end{array}
\qquad
\begin{array}{ccc}
 & \neg(A \equiv B) & \\
\swarrow & & \searrow \\
A & & \neg A \\
\downarrow & & \downarrow \\
\neg B & & B
\end{array}
$$

Intuitively, what a tableau means is the following. If we apply a rule to a formula, then if that formula is true in an interpretation, so are the formulas below on at least one of the branches that the rule generates. (Of course, there may be only one such branch.) This is a useful mnemonic for remembering the rules. It must be stressed, though, that officially the rules are purely formal.

1.4.5 A tableau is *complete* iff every rule that can be applied has been applied. By applying the rules over and over, we may always construct a complete tableau. In the present case, the branches of a completed tableau are always finite,[4] but in the tableaux of some subsequent chapters they may be infinite.

1.4.6 A branch is *closed* iff there are formulas of the form A and $\neg A$ on two of its nodes; otherwise it is *open*. A closed branch is indicated by writing an $\times$ at the bottom. A *tableau* itself is closed iff every branch is closed; otherwise it is open. Thus the tableau of 1.4.3 is closed: the leftmost branch contains A and $\neg A$; the next contains A and $\neg A$ (and C and $\neg C$); the next contains B and $\neg B$; the rightmost contains C and $\neg C$.

1.4.7 A is a proof-theoretic consequence of the set of formulas $\Sigma (\Sigma \vdash A)$ iff there is a complete tree whose initial list comprises the members of Σ and the negation of A, and which is closed. We write $\vdash A$ to mean that $\phi \vdash A$,

[4] This is not entirely obvious, though it is not difficult to prove.

that is, where the initial list of the tableau comprises just $\neg A$. '$\Sigma \nvdash A$' means that it is not the case that $\Sigma \vdash A$.[5]

1.4.8 Thus, the tree of 1.4.3 shows that $A \supset B, B \supset C \vdash A \supset C$. Here is another, to show that $\vdash ((A \supset B) \wedge (A \supset C)) \supset (A \supset (B \wedge C))$. To save space, we omit arrows where a branch does not divide.

$$
\begin{array}{ccccccc}
 & & & \neg(((A \supset B) \wedge (A \supset C)) \supset (A \supset (B \wedge C))) & & & \\
 & & & (A \supset B) \wedge (A \supset C) & & & \\
 & & & \neg(A \supset (B \wedge C)) & & & \\
 & & & (A \supset B) & & & \\
 & & & (A \supset C) & & & \\
 & & & A & & & \\
 & & & \neg(B \wedge C) & & & \\
 & & \swarrow & & \searrow & & \\
 & & \neg B & & \neg C & & \\
 & \swarrow & \downarrow & & \downarrow & \searrow & \\
\neg A & & B & & \neg A & B & \\
\times & & \times & & \times & \downarrow \quad \searrow & \\
 & & & & & \neg A & C \\
 & & & & & \times & \times
\end{array}
$$

Note that when we find a contradiction on a branch, there is no point in continuing it further. We know that the branch is going to close, whatever else is added to it. Hence, we need not bother to extend a branch as soon as it is found to close. Notice also that, wherever possible, we apply rules that do not split branches before rules that split branches. Though this is not essential, it keeps the tableau simpler, and is therefore useful practically.

1.4.9 In practice, it is also a useful idea to put a tick at the side of a formula once one has applied a rule to it. Then one knows that one can forget about it.

[5] There may, in fact, be several completed trees for an inference, depending upon the order of the premises in the initial list and the order in which rules are applied. Fortunately, they all give the same result, though this is not entirely obvious. See 1.14, problem 5.

1.5 Counter-models

1.5.1 Here is another example, to show that $(p \supset q) \vee (r \supset q) \nvdash (p \vee r) \supset q$.

$$
\begin{array}{c}
(\mathbf{p} \supset \mathbf{q}) \vee (\mathbf{r} \supset \mathbf{q}) \\
\neg((\mathbf{p} \vee \mathbf{r}) \supset \mathbf{q}) \\
(\mathbf{p} \vee \mathbf{r}) \\
\neg \mathbf{q} \\
\swarrow \qquad\qquad\qquad \searrow
\end{array}
$$

$$
\begin{array}{cccccccccc}
 & & & & (\mathbf{p} \supset \mathbf{q}) & & (r \supset q) & & & \\
 & & & \swarrow & \downarrow & & \downarrow & \searrow & & \\
 & & \neg \mathbf{p} & & q & & \neg r & & q & \\
 & \swarrow & \downarrow & & \times & & \downarrow & \searrow & & \times \\
p & & \mathbf{r} & & & & p & & r & \\
\times & & & & & & & & \times &
\end{array}
$$

The tableau has two open branches. The leftmost one is emphasised in bold for future reference.

1.5.2 The tableau procedure is, in effect, a systematic search for an interpretation that makes all the formulas on the initial list true. Given an open branch of a tableau, such an interpretation can, in fact, be read off from the branch.[6]

1.5.3 The recipe is simple. If the propositional parameter, p, occurs at a node on the branch, assign it 1; if $\neg p$ occurs at a node on the branch, assign it 0. (If neither p nor $\neg p$ occurs in this way, it may be assigned anything one likes.)

1.5.4 For example, consider the tableau of 1.5.1 and its (bolded) leftmost open branch. Applying the recipe gives the interpretation, ν, such that $\nu(r) = 1$, and $\nu(p) = \nu(q) = 0$. It is simple to check directly that $\nu((p \supset q) \vee (r \supset q)) = 1$ and $\nu((p \vee r) \supset q) = 0$. Since p is false, $p \supset q$ is true, as is $(p \supset q) \vee (r \supset q)$. Since r is true, $p \vee r$ is true; but q is false; hence, $(p \vee r) \supset q$ is false.

[6] If one thinks of constructing a tableau as a search procedure for a counter-model, then the soundness and completeness theorems constitute, in effect, a proof that the procedure always gives the right result, that is, which *verifies* the algorithm in question.

1.5.4a Note that the tableau of 1.4.8 shows that *any* inference of the form in question is valid. That is, A, B and C can be *any* formulas. To show that an inference is invalid, we have to construct a counter-model, and this means assigning truth values to *particular* formulas. This is why the example just given uses 'p', 'q' and 'r', not 'A', 'B' and 'C'. One may say that an inference expressed using schematic letters ('A's and 'B's) is invalid, but this must mean that there are some formulas that can be substituted for these letters to make it so. Thus, we may write $A \nvDash B$, since $p \nvDash q$. But note that this does not rule out the possibility that some inferences of that form are valid, e.g., $p \vDash q \vee \neg q$.

1.5.5 As one would hope, the tableau procedure we have been looking at is sound and complete with respect to the semantic notion of consequence, i.e., if Σ is a finite set of sentences, $\Sigma \vdash A$ iff $\Sigma \models A$. That is, the search procedure really works. If there is an interpretation that makes all the formulas on the initial list true, the tableau will have an open branch which, in effect, specifies one. And if there is no such interpretation, every branch will close. These facts are not obvious. The proof is in 1.11.[7]

1.6 Conditionals

1.6.1 In the remainder of this chapter, we look at the notion of conditionality that the above, classical, semantics give us, and at its inadequacy. But first, what is a conditional?

1.6.2 Conditionals relate some proposition (the *consequent*) to some other proposition (the *antecedent*) on which, in some sense, it depends. They are expressed in English by 'if' or cognate constructions:

If the bough breaks (then) the cradle will fall.
The cradle will fall if the bough breaks.
The bough breaks only if the cradle falls.

[7] The restriction to finite Σ is due to the fact that tableaux have been defined only for finite sets of premises. It is possible to define tableaux for infinite sets of premises as well (not putting all the premises at the start, but introducing them, one by one, at regular intervals down the branches). If one does this, the soundness and completeness results generalise to arbitrary sets of premises. We will take up this matter again in Chapter 12 (Part II), where the matter assumes more significance.

1.10.5 Now, suppose that $\neg A \vee B$ is true. Then from this and A we can deduce B, by the *disjunctive syllogism*: $A, \neg A \vee B \vdash B$. Hence, by (*), 'If A then B' is true.

1.10.6 We will come back to this argument in a later chapter. For now, just note the fact that it uses the disjunctive syllogism.

1.11 *Proofs of Theorems

1.11.1 Definition: Let ν be any propositional interpretation. Let b be any branch of a tableau. Say that ν is *faithful* to b iff for every formula, A, on the branch, $\nu(A) = 1$.

1.11.2 Soundness Lemma: If ν is faithful to a branch of a tableau, b, and a tableau rule is applied to b, then ν is faithful to at least one of the branches generated.

Proof:
The proof is by a case-by-case examination of the tableau rules. Here are the cases for the rules for $\supset$. The other cases are left as exercises. Suppose that ν is faithful to b, that $\neg(A \supset B)$ occurs on b, and that we apply a rule to it. Then only one branch eventuates, that obtained by adding A and $\neg B$ to b. Since ν is faithful to b, it makes every formula on b true. In particular, $\nu(\neg(A \supset B)) = 1$. Hence, $\nu(A \supset B) = 0$, $\nu(A) = 1$, $\nu(B) = 0$, and so $\nu(\neg B) = 1$. Hence, ν makes every formula on b true. Next, suppose that ν is faithful to b, that $A \supset B$ occurs on b, and that we apply a rule to it. Then two branches eventuate, one extending b with $\neg A$ (the left branch); the other extending b with B (the right branch). Since ν is faithful to b, it makes every formula on b true. In particular, $\nu(A \supset B) = 1$. Hence, $\nu(A) = 0$, and so $\nu(\neg A) = 1$, or $\nu(B) = 1$. In the first case, ν is faithful to the left branch; in the second, it is faithful to the right. ■

1.11.3 Soundness Theorem: For finite Σ, if $\Sigma \vdash A$ then $\Sigma \models A$.

Proof:
We prove the contrapositive. Suppose that $\Sigma \not\models A$. Then there is an interpretation, ν, which makes every member of Σ true, and A false – and hence makes $\neg A$ true. Now consider a completed tableau for the inference. ν is faithful to the initial list. When we apply a rule to the list, we can, by the

Soundness Lemma, find at least one of its extensions to which ν is faithful. Similarly, when we apply a rule to this, we can find at least one of *its* extensions to which ν is faithful; and so on. By repeatedly applying the Soundness Lemma in this way, we can find a whole branch, b, such that ν is faithful to every initial section of it. (An initial section is a path from the root down the branch, but not necessarily all the way to the tip.) It follows that ν is faithful to b itself, but we do not need this fact to make the proof work. Now, if b were closed, it would have to contain some formulas of the form B and $\neg B$, and these must occur in some initial section of b. But this is impossible since ν is faithful to this section, and so it would follow that $\nu(B) = \nu(\neg B) = 1$, which cannot be the case. Hence, the tableau is open, i.e., $\Sigma \nvdash A$. ■

1.11.4 Definition: Let b be an open branch of a tableau. The interpretation *induced* by b is any interpretation, ν, such that for every propositional parameter, p, if p is at a node on b, $\nu(p) = 1$, and if $\neg p$ is at a node on b, $\nu(p) = 0$. (And if neither, $\nu(p)$ can be anything one likes.) This is well defined, since b is open, and so we cannot have both p and $\neg p$ on b.

1.11.5 Completeness Lemma: Let b be an open complete branch of a tableau. Let ν be the interpretation induced by b. Then:

if A is on b, $\nu(A) = 1$
if $\neg A$ is on b, $\nu(A) = 0$

Proof:
The proof is by induction on the complexity of A. If A is a propositional parameter, the result is true by definition. If A is complex, it is of the form $B \wedge C$, $B \vee C$, $B \supset C$, $B \equiv C$, or $\neg B$. Consider the first case, and suppose that $B \wedge C$ is on b. Since b is complete, the rule for conjunction has been applied to it. Hence, both B and C are on the branch. By induction hypothesis, $\nu(B) = \nu(C) = 1$. Hence, $\nu(B \wedge C) = 1$, as required. Next, suppose that $\neg(B \wedge C)$ is on b. Since the rule for negated conjunction has been applied to it, either $\neg B$ or $\neg C$ is on the branch. By induction hypothesis, either $\nu(B) = 0$ or $\nu(C) = 0$. In either case, $\nu(B \wedge C) = 0$, as required. The cases for the other binary connectives are similar. For $\neg$: suppose that $\neg B$ is on b. Then, since the result holds for B, by the induction hypothesis, $\nu(B) = 0$. Hence, $\nu(\neg B) = 1$. If $\neg\neg B$ is on b, then so is B, by the rule for double negation. By induction hypothesis, $\nu(B) = 1$, so $\nu(\neg B) = 0$. ■

1.11.6 COMPLETENESS THEOREM: For finite Σ, if $\Sigma \models A$ then $\Sigma \vdash A$.

Proof:
We prove the contrapositive. Suppose that $\Sigma \nvdash A$. Consider a completed open tableau for the inference, and choose an open branch. The interpretation that the branch induces makes all the members of Σ true, and A false, by the Completeness Lemma. Hence, $\Sigma \not\models A$. ■

1.12 History

The propositional logic described in this chapter was first formulated by Frege in his *Begriffsschrift* (translated in Bynum, 1972) and Russell (1903). Semantic tableaux in the form described here were first given in Smullyan (1968). The issue of how to understand the conditional is an old one. Disputes about it can be found in the Stoics and in the Middle Ages. Some logicians at each of these times endorsed the material conditional. For an account of the history, see Sanford (1989). The defence of the material conditional in terms of conversational rules first seems to have been suggested by Ajdukiewicz (1956). The idea was brought to prominence by Grice (1989, chs. 1–4). The argument for distinguishing between the indicative and subjunctive conditionals was first given by Adams (1970). The examples of 1.9 are taken from a much longer list given by Cooper (1968). The argument of 1.10 was given by Faris (1968).

1.13 Further Reading

For an introduction to classical logic based on tableaux, see Jeffrey (1991), Howson (1997) or Restall (2006). For a number of good papers discussing the connection between material, indicative and subjunctive conditionals, see Jackson (1991). For further discussion of the examples of sec1.9, see Routley, Plumwood, Meyer and Brady (1982, ch. 1).

1.14 Problems

1. Check the truth of each of the following, using tableaux. If the inference is invalid, read off a counter-model from the tree, and check directly that it makes the premises true and the conclusion false, as in 1.5.4.

7 Many-valued Logics

7.1 Introduction

7.1.1 In this chapter, we leave possible-world semantics for a time, and turn to the subject of propositional many-valued logics. These are logics in which there are more than two truth values.

7.1.2 We have a look at the general structure of a many-valued logic, and some simple but important examples of many-valued logics. The treatment will be purely semantic: we do not look at tableaux for the logics, nor at any other form of proof procedure. Tableaux for some many-valued logics will emerge in the next chapter.

7.1.3 We also look at some of the philosophical issues that have motivated many-valued logics, how many-valuedness affects the issue of the conditional, and a few other noteworthy issues.

7.2 Many-valued Logic: The General Structure

7.2.1 Let us start with the general structure of a many-valued logic. To simplify things, we take, henceforth, $A \equiv B$ to be defined as $(A \supset B) \wedge (B \supset A)$.

7.2.2 Let $\mathcal{C}$ be the class of connectives of classical propositional logic $\{\wedge, \vee, \neg, \supset\}$. The classical propositional calculus can be thought of as defined by the structure $\langle \mathcal{V}, \mathcal{D}, \{f_c; c \in \mathcal{C}\}\rangle$. $\mathcal{V}$ is the set of truth values $\{1,0\}$. $\mathcal{D}$ is the set of *designated* values $\{1\}$; these are the values that are preserved in valid inferences. For every connective, c, f_c is the truth function it denotes. Thus, $f_\neg$ is a one-place function such that $f_\neg(0) = 1$ and $f_\neg(1) = 0$; $f_\wedge$ is a two-place function such that $f_\wedge(x, y) = 1$ if $x = y = 1$, and $f_\wedge(x, y) = 0$ otherwise; and so on. These functions can be (and often are) depicted in the following 'truth tables'.

$f_\neg$	
1	0
0	1

$f_\wedge$	1	0
1	1	0
0	0	0

7.2.3 An interpretation, ν, is a map from the propositional parameters to $\mathcal{V}$. An interpretation is extended to a map from all formulas into $\mathcal{V}$ by applying the appropriate truth functions recursively. Thus, for example, $\nu(\neg(p\wedge q)) = f_\neg(\nu(p \wedge q)) = f_\neg(f_\wedge(\nu(p), \nu(q)))$. (So if $\nu(p) = 1$ and $\nu(q) = 0$, $\nu(\neg(p \wedge q)) = f_\neg(f_\wedge(1, 0)) = f_\neg(0) = 1$.) Finally, an inference is semantically valid just if there is no interpretation that assigns all the premises a value in $\mathcal{D}$, but assigns the conclusion a value not in $\mathcal{D}$.

7.2.4 A many-valued logic is a natural generalisation of this structure. Given some propositional language with connectives $\mathcal{C}$ (maybe the same as those of the classical propositional calculus, maybe different), a logic is defined by a structure $\langle \mathcal{V}, \mathcal{D}, \{f_c; c \in \mathcal{C}\}\rangle$. $\mathcal{V}$ is the set of truth values: it may have any number of members (≥ 1). $\mathcal{D}$ is a subset of $\mathcal{V}$, and is the set of designated values. For every connective, c, f_c is the corresponding truth function. Thus, if c is an n-place connective, f_c is an n-place function with inputs and outputs in $\mathcal{V}$.

7.2.5 An interpretation for the language is a map, ν, from propositional parameters into $\mathcal{V}$. This is extended to a map from all formulas of the language to $\mathcal{V}$ by applying the appropriate truth functions recursively. Thus, if c is an n-place connective, $\nu(c(A_1, \ldots, A_n)) = f_c(\nu(A_1), \ldots, \nu(A_n))$. Finally, $\Sigma \models A$ iff there is no interpretation, ν, such that for all $B \in \Sigma$, $\nu(B) \in \mathcal{D}$, but $\nu(A) \notin \mathcal{D}$. A is a logical truth iff $\phi \models A$, i.e., iff for every interpretation $\nu(A) \in \mathcal{D}$.

7.2.6 If $\mathcal{V}$ is finite, the logic is said to be *finitely many-valued*. If $\mathcal{V}$ has n members, it is said to be an n-valued logic.

7.2.7 For any finitely many-valued logic, the validity of an inference with finitely many premises can be determined, as in the classical propositional calculus, simply by considering all the possible cases. We list all the possible combinations of truth values for the propositional parameters employed.

Then, for each combination, we compute the value of each premise and the conclusion. If, in any of these, the premises are all designated and the conclusion is not, the inference is invalid. Otherwise, it is valid. We will have an example of this procedure in the next section.

7.2.8 This method, though theoretically adequate, is often impractical because of exponential explosion. For if there are m propositional parameters employed in an inference, and n truth values, there are n^m possible cases to consider. This grows very rapidly. Thus, if the logic is 4-valued and we have an inference involving just four propositional parameters, there are already 256 cases to consider!

7.3 The 3-valued Logics of Kleene and Łukasiewicz

7.3.1 In what follows, we consider some simple examples of the above general structure. All the examples that we consider are 3-valued logics. The language, in every case, is that of the classical propositional calculus.

7.3.2 A simple example of a 3-valued logic is as follows. $\mathcal{V} = \{1, i, 0\}$. 1 and 0 are to be thought of as *true* and *false*, as usual. i is to be thought of as *neither true nor false*. $\mathcal{D}$ is just $\{1\}$. The truth functions for the connectives are depicted as follows:

$f_\neg$	
1	0
i	i
0	1

$f_\wedge$	1	i	0
1	1	i	0
i	i	i	0
0	0	0	0

$f_\vee$	1	i	0
1	1	1	1
i	1	i	i
0	1	i	0

$f_\supset$	1	i	0
1	1	i	0
i	1	i	i
0	1	1	1

Thus, if $\nu(p) = 1$ and $\nu(q) = i$, $\nu(\neg p) = 0$ (top row of $f_\neg$), $\nu(\neg p \vee q) = i$ (bottom row, middle column of $f_\vee$), etc.

7.3.3 Note that if the inputs of any of these functions are classical (1 or 0), the output is exactly the same as in the classical case. We compute the other entries as follows. Take $A \wedge B$ as an example. If A is false, then, whatever B is, this is (classically) sufficient to make $A \wedge B$ false. In particular, if B is neither true nor false, $A \wedge B$ is false. If A is true, on the other hand, and B is neither true nor false, there is insufficient information to compute the (classical) value of $A \wedge B$; hence, $A \wedge B$ is neither true nor false. Similar reasoning justifies all the other entries.

7.3.4 The logic specified above is usually called the (strong) Kleene 3-valued logic, often written K_3.[1]

7.3.5 The following table verifies that $p \supset q \models_{K_3} \neg q \supset \neg p$:

p	q	$p \supset q$	$\neg q$	$\supset$	$\neg p$
1	1	**1**	0	**1**	0
1	i	**i**	i	**i**	0
1	0	**0**	1	**0**	0
i	1	**1**	0	**1**	i
i	i	**i**	i	**i**	i
i	0	**i**	1	**i**	i
0	1	**1**	0	**1**	1
0	i	**1**	i	**1**	1
0	0	**1**	1	**1**	1

In the last three columns, the first number is the value of $\neg q$; the last number is that of $\neg p$, and the central number (printed in bold) is the value of the whole formula. As can be seen, there is no interpretation where the premise is designated, that is, has the value 1, and the conclusion is not.

7.3.6 In checking for validity, it may well be easier to work backwards. Consider the formula $p \supset (q \supset p)$. Suppose that this is undesignated. Then it has either the value 0 or the value i. If it has the value 0, then p has the value 1 and $q \supset p$ has the value 0. But if p has the value 1, so does $q \supset p$. This situation is therefore impossible. If it has the value i, there are three possibilities:

p	$q \supset p$
1	i
i	i
i	0

The first case is not possible, since if p has the value 1, so does $q \supset p$. Nor is the last case, since if p has the value i, $q \supset p$ has value either i or 1. But the

[1] Weak Kleene logic is the same as K_3, except that, for every truth function, if any input is i, so is the output.

second case is possible, namely when both p and q have the value i. Thus, $\nu(p) = \nu(q) = i$ is a counter-model to $p \supset (q \supset p)$, as a truth-table check confirms. So $\not\models_{K_3} p \supset (q \supset p)$.

7.3.7 A distinctive thing about K_3 is that the law of excluded middle is not valid: $\not\models_{K_3} p \vee \neg p$. (Counter-model: $\nu(p) = i$.) However, K_3 is distinct from intuitionist logic. As we shall see in 7.10.8, intuitionist logic is not the same as any finitely many-valued logic.

7.3.8 In fact, K_3 has no logical truths at all (7.14, problem 3)! In particular, the law of identity is not valid: $\not\models_{K_3} p \supset p$. (Simply give p the value i.) This may be changed by modifying the middle entry of the truth function for $\supset$, so that $f_\supset$ becomes:

$f_\supset$	1	i	0
1	1	i	0
i	1	1	i
0	1	1	1

(The meaning of $A \supset B$ in K_3 can still be expressed by $\neg A \vee B$, since this has the same truth table, as may be checked.) Now, $A \supset A$ always takes the value 1.

7.3.9 The logic resulting from this change is one originally given by Łukasiewicz, and is often called $Ł_3$.

7.4 *LP* and *RM*$_3$

7.4.1 Another 3-valued logic is the one often called *LP*. This is exactly the same as K_3, except that $\mathcal{D} = \{1, i\}$.

7.4.2 In the context of *LP*, the value i is thought of as *both true and false*. Consequently, 1 and 0 have to be thought of as *true and true only*, and *false and false only*, respectively. This change does not affect the truth tables, which still make perfectly good sense under the new interpretation. For example, if A takes the value 1 and B takes the value i, then A and B are both true; hence, $A \wedge B$ is true; but since B is false, $A \wedge B$ is false. Hence, the value of $A \wedge B$ is i. Similarly, if A takes the value 0, and B takes the value i, then A

and B are both false, so $A \wedge B$ is false; but only B is true, so $A \wedge B$ is not true. Hence, $A \wedge B$ takes the value 0.

7.4.3 However, the change of designated values makes a crucial difference. For example, $\models_{LP} p \vee \neg p$. (Whatever value p has, $p \vee \neg p$ takes either the value 1 or i. Thus it is always designated.) This fails in K_3, as we saw in 7.3.7.

7.4.4 On the other hand, $p \wedge \neg p \not\models_{LP} q$. Counter-model: $\nu(p) = i$ (making $\nu(p \wedge \neg p) = i$), $\nu(q) = 0$. But $p \wedge \neg p$ can never take the value 1 and so be designated in K_3. Thus, the inference is valid in K_3.

7.4.5 A notable feature of *LP* is that *modus ponens* is invalid: $p, p \supset q \not\models_{LP} q$. (Assign p the value i, and q the value 0.)

7.4.6 One way to rectify this is to change the truth function for $\supset$ to the following:

$f_\supset$	1	i	0
1	1	0	0
i	1	i	0
0	1	1	1

(As in 7.3.8, the meaning of $A \supset B$ in *LP* can still be expressed by $\neg A \vee B$.) Now, if A and $A \supset B$ have designated values (1 or i), so does B, as a moment checking the truth table verifies.

7.4.7 This change gives the logic often called RM_3.

7.5 Many-valued Logics and Conditionals

7.5.1 Further details of the properties of $\wedge$, $\vee$ and $\neg$ in the logics we have just met will emerge in the next chapter. For the present, let us concentrate on the conditional.

7.5.2 In past chapters, we have met a number of problematic inferences concerning conditionals. The following table summarises whether or not they hold in the various logics we have looked at. (A tick means *yes*; a cross means *no*.)

		K_3	$Ł_3$	LP	RM_3
(1)	$q \models p \supset q$	√	√	√	×
(2)	$\neg p \models p \supset q$	√	√	√	×
(3)	$(p \wedge q) \supset r \models (p \supset r) \vee (q \supset r)$	√	√	√	√
(4)	$(p \supset q) \wedge (r \supset s) \models (p \supset s) \vee (r \supset q)$	√	√	√	√
(5)	$\neg(p \supset q) \models p$	√	√	√	√
(6)	$p \supset r \models (p \wedge q) \supset r$	√	√	√	√
(7)	$p \supset q, q \supset r \models p \supset r$	√	√	×	√
(8)	$p \supset q \models \neg q \supset \neg p$	√	√	√	√
(9)	$\models p \supset (q \vee \neg q)$	×	×	√	×
(10)	$\models (p \wedge \neg p) \supset q$	×	×	√	×

(1) and (2) we met in 1.7, and (3)–(5) we met in 1.9, all in connection with the material conditional. (6)–(8) we met in 5.2, in connection with conditional logics. (9) and (10) we met in 4.6, in connection with the strict conditional. The checking of the details is left as a (quite lengthy) exercise. For K_3, a generally good strategy is to start by assuming that the premises take the value 1 (the only designated value), and recall that, in K_3, if a conditional takes the value 1, then either its antecedent takes the value 0 or the consequent takes the value 1. For $Ł_3$, it is similar, except that a conditional with value 1 may also have antecedent and consequent with value i. For LP, a generally good strategy is to start by assuming that the conclusion takes the value 0 (the only undesignated value), and recall that, in LP, if a conditional takes the value 0, then the antecedent takes the value 1 and the consequent takes the value 0. For RM_3, it is similar, except that if a conditional has value 0, the antecedent and consequent may also take the values 1 and i, or i and 0, respectively. And recall that classical inputs (1 or 0) always give the classical outputs.

7.5.3 As can be seen from the number of ticks, the conditionals do not fare very well. If one's concern is with the ordinary conditional, and not with conditionals with an enthymematic *ceteris paribus* clause, then one may ignore lines (6)–(8). But all the logics suffer from some of the same problems as the material conditional. K_3 and $Ł_3$ also suffer from some of the problems that the strict conditional does. In particular, even though (10) tells us that $(p \wedge \neg p) \supset q$ is not valid in these logics, contradictions still entail everything, since $p \wedge \neg p$ can never assume a designated value. By contrast, this is not true of *LP* (as we saw in 7.4.4), but this is so only because *modus ponens* is invalid, since $(p \wedge \neg p) \supset q$ is valid, as (10) shows. (*Modus ponens* is valid for the other logics, as may easily be checked.) About the best of the bunch is RM_3.

7.5.4 But there are quite general reasons as to why the conditional of any finitely many-valued logic is bound to be problematic. For a start, if disjunction is to behave in a natural way, the inference from A (or B) to $A \vee B$ must be valid. Hence, we must have:

(i) if A (or B) is designated, so is $A \vee B$

Also, $A \equiv A$ ought to be a logical truth. (Even if A is neither true nor false, for example, it would still seem to be the case that *if* A then A, and so, that A iff A.) Hence:

(ii) if A and B have the same value, $A \equiv B$ must be designated (since $A \equiv A$ is).

Note that both of these conditions hold for all the logics that we have looked at, with the exception of K_3, for which (ii) fails.

7.5.5 Now, take any n-valued logic that satisfies (i) and (ii), and consider $n+1$ propositional parameters, $p_1, p_2, \ldots, p_{n+1}$. Since there are only n truth values, in any interpretation, two of these must receive the same value. Hence, by (ii), for some j and k, $p_j \equiv p_k$ must be designated. But then the disjunction of all biconditionals of this form must also be designated, by (i). Hence, this disjunction is logically valid.

7.5.6 But this seems entirely wrong. Consider $n + 1$ propositions such as 'John has 1 hair on his head', 'John has 2 hairs on his head', . . ., 'John has $n + 1$ hairs on his head'. Any biconditional relating a pair of these would appear to be false. Hence, the disjunction of all such pairs would also appear to be false – certainly not logically true.

7.6 Truth-value Gluts: Inconsistent Laws

7.6.1 Let us now turn to the issue of the philosophical motivations for many-valued logics and, in particular, the 3-valued logics we have met. Typically, the motivations for those logics that treat i as both true and false (a *truth-value glut*), like *LP* and RM_3, are different from those that treat i as neither true nor false (a *truth-value gap*), like K_3 and $Ł_3$. Let us start

8 First Degree Entailment

8.1 Introduction

8.1.1 In this chapter we look at a logic called *first degree entailment* (*FDE*). This is formulated, first, as a logic where interpretations are *relations* between formulas and standard truth values, rather than as the more usual *functions*. Connections between *FDE* and the many-valued logics of the last chapter will emerge.

8.1.2 We also look at an alternative possible-world semantics for *FDE*, which will introduce us to a new kind of semantics for negation.

8.1.3 Finally, we look at the relation of all this to the explosion of contradictions, and to the disjunctive syllogism.

8.2 The Semantics of *FDE*

8.2.1 The language of *FDE* contains just the connectives $\wedge$, $\vee$ and $\neg$. $A \supset B$ is defined, as usual, as $\neg A \vee B$.

8.2.2 In the classical propositional calculus, an interpretation is a function from formulas to the truth values 0 and 1, written thus: $\nu(A) = 1$ (or 0). Packed into this formalism is the assumption (usually made without comment in elementary logic texts) that every formula is either true or false; never neither, and never both.

8.2.3 As we saw in the last chapter, there are reasons to doubt this assumption. If one does, it is natural to formulate an interpretation, not as a function, but as a relation between formulas and truth values. Thus, a formula may relate to 1; it may relate to 0; it may relate to both; or it may relate to neither. This is the main idea behind the following semantics for *FDE*.

8.2.4 Note that it is now very important to distinguish between being false in an interpretation and not being true in it. (There is, of course, no difference in the classical case.) The fact that a formula is false (relates to 0) does not mean that it is untrue (it may also relate to 1). And the fact that it is untrue (does not relate to 1) does not mean that it is false (it may not relate to 0 either).

8.2.5 An *FDE* interpretation is a relation, ρ[1] between propositional parameters and the values 1 and 0. (In mathematical notation, $\rho \subseteq \mathcal{P} \times \{1, 0\}$, where $\mathcal{P}$ is the set of propositional parameters.) Thus, $p\rho 1$ means that p relates to 1, and $p\rho 0$ means that p relates to 0.

8.2.6 Given an interpretation, ρ, this is extended to a relation between all formulas and truth values by the recursive clauses:

$A \wedge B\rho 1$ iff $A\rho 1$ and $B\rho 1$
$A \wedge B\rho 0$ iff $A\rho 0$ or $B\rho 0$

$A \vee B\rho 1$ iff $A\rho 1$ or $B\rho 1$
$A \vee B\rho 0$ iff $A\rho 0$ and $B\rho 0$

$\neg A\rho 1$ iff $A\rho 0$
$\neg A\rho 0$ iff $A\rho 1$

Note that these are exactly the same as the classical truth conditions, stripped of the assumption that truth and falsity are exclusive and exhaustive. Thus, a conjunction is true (under an interpretation) if both conjuncts are true (under that interpretation); it is false if at least one conjunct is false, etc.

8.2.7 As an example of how these conditions work, consider the formula $\neg p \wedge (q \vee r)$. Suppose that $p\rho 1$, $p\rho 0$, $q\rho 1$ and $r\rho 0$, and that ρ relates no parameter to anything else. Since p is true, $\neg p$ is false; and since p is false, $\neg p$ is true. Thus $\neg p$ is both true and false. Since q is true, $q \vee r$ is true; and since q is not false, $q \vee r$ is not false. Thus, $q \vee r$ is simply true. But then, $\neg p \wedge (q \vee r)$ is true, since both conjuncts are true; and false, since the first conjunct is false. That is, $\neg p \wedge (q \vee r)\rho 1$ and $\neg p \wedge (q \vee r)\rho 0$.

[1] Not to be confused with the reflexive ρ of normal modal logics.

8.2.8 Semantic consequence is defined, in the usual way, in terms of truth preservation, thus:

$$\Sigma \models A \text{ iff for every interpretation, } \rho, \text{ if } B\rho 1 \text{ for all } B \in \Sigma \text{ then } A\rho 1$$

and:

$$\models A \text{ iff } \phi \models A, \text{ i.e., for all } \rho, \ A\rho 1$$

8.3 Tableaux for *FDE*

8.3.1 Tableaux for *FDE* can be obtained by modifying those for the classical propositional calculus as follows.

8.3.2 Each entry of the tableau is now of the form $A, +$ or $A, -$. Intuitively, A, $+$ means that A is true, $A, -$ means that it isn't. As we noted in 8.2.4, and as with intuitionist logic (6.4.1), $\neg A, +$ no longer means the same, intuitively, as $A, -$.

8.3.3 To test the claim that $A_1, \ldots, A_n \vdash B$, we start with an initial list of the form:

$$\begin{array}{c} A_1, + \\ \vdots \\ A_n, + \\ B, - \end{array}$$

8.3.4 The tableaux rules are as follows:

$$\begin{array}{ccc} A \wedge B, + & \qquad & A \wedge B, - \\ \downarrow & & \swarrow \quad \searrow \\ A, + & & A, - \quad B, - \\ B, + & & \end{array}$$

$$\begin{array}{ccc} A \vee B, + & \qquad & A \vee B, - \\ \swarrow \quad \searrow & & \downarrow \\ A, + \quad B, + & & A, - \\ & & B, - \end{array}$$

$$\begin{array}{ccc} \neg(A \wedge B), + & \qquad & \neg(A \wedge B), - \\ \downarrow & & \downarrow \\ \neg A \vee \neg B, + & & \neg A \vee \neg B, - \end{array}$$

$$\begin{array}{cc} \neg(A \vee B), + & \neg(A \vee B), - \\ \downarrow & \downarrow \\ \neg A \wedge \neg B, + & \neg A \wedge \neg B, - \end{array}$$

$$\begin{array}{cc} \neg\neg A, + & \neg\neg A, - \\ \downarrow & \downarrow \\ A, + & A, - \end{array}$$

The first two rules speak for themselves: if $A \wedge B$ is true, A and B are true; if $A \wedge B$ is not true, then one or other of A and B is not true. Similarly for the rules for disjunction. The other rules are also easy to remember, since $\neg(A \wedge B)$ and $\neg A \vee \neg B$ have the same truth values in *FDE*, as do $\neg(A \vee B)$ and $\neg A \wedge \neg B$, and $\neg\neg A$ and A. (De Morgan's laws and the law of double negation, respectively.)

8.3.5 Finally, a branch of a tableau closes if it contains nodes of the form $A, +$ and $A, -$.

8.3.6 For example, the following tableau demonstrates that $\neg(B \wedge \neg C) \wedge A \vdash (\neg B \vee C) \vee D$:

$$\begin{array}{c} \neg(B \wedge \neg C) \wedge A, + \\ (\neg B \vee C) \vee D, - \\ \neg(B \wedge \neg C), + \\ A, + \\ \neg B \vee \neg\neg C, + \\ \neg B \vee C, - \\ D, - \\ \neg B, - \\ C, - \\ \swarrow \qquad\qquad \searrow \\ \begin{array}{cc} \neg B, + & \neg\neg C, + \\ \times & C, + \\ & \times \end{array} \end{array}$$

The third and fourth lines come from the first, by the rule for true conjunctions. The next line comes from the third by De Morgan's laws. The next two lines come from the second by the rule for untrue disjunctions, which is then applied again, to get the next two lines. The branching arises because of the rule for true disjunctions, applied to line five. The left branch is now closed because of $\neg B, -$ and $\neg B, +$; an application of double negation then closes the righthand branch.

8.3.7 Here is another example, to show that $p \wedge (q \vee \neg q) \nvdash r$:

$$\begin{array}{c} p \wedge (q \vee \neg q), + \\ r, - \\ p, + \\ q \vee \neg q, + \\ \swarrow \qquad \searrow \\ q, + \qquad \neg q, + \end{array}$$

8.3.8 Counter-models can be read off from open branches in a simple way. For every parameter, p, if there is a node of the form $p, +$, set $p\rho 1$; if there is a node of the form $\neg p, +$, set $p\rho 0$. No other facts about ρ obtain.

8.3.9 Thus, the counter-model defined by the righthand branch of the tableau in 8.3.7 is the interpretation ρ, where $p\rho 1$ and $q\rho 0$ (and no other relations hold). It is easy to check directly that this interpretation makes the premises true and the conclusion untrue.

8.3.10 The tableaux are sound and complete with respect to the semantics. This is proved in 8.7.1–8.7.7.

8.4 *FDE* and Many-valued Logics

8.4.1 Given any formula, A, and any interpretation, ρ, there are four possibilities: A is true and not also false, A is false and not also true, A is true and false, A is neither true nor false. If we write these possibilities as 1, 0, b and n, respectively, this makes it possible to think of *FDE* as a 4-valued logic.

8.4.2 The truth conditions of 8.2.6 give the following truth tables:

$f_\neg$	
1	0
b	b
n	n
0	1

$f_\wedge$	1	b	n	0
1	1	b	n	0
b	b	b	0	0
n	n	0	n	0
0	0	0	0	0

$f_\vee$	1	b	n	0
1	1	1	1	1
b	1	b	1	b
n	1	1	n	n
0	1	b	n	0

The details are laborious, but easy enough to check. Thus, suppose that A is n and B is b. Then it is not the case that A and B are both true; hence, $A \wedge B$ is not true. But B is false; hence, $A \wedge B$ is false. Thus, $A \wedge B$ is false but not true, 0. Since B is true, $A \vee B$ is true; and since A and B are not both false, $A \vee B$ is not false. Hence, $A \vee B$ is true and not false, 1. The other cases are left as an exercise.

8.4.3 An easy way to remember these values is with the following diagram, the 'diamond lattice':

$$\begin{tikzcd} & 1 & \\ b \arrow[ur] & & n \arrow[ul] \\ & 0 \arrow[ul] \arrow[ur] & \end{tikzcd}$$

The conjunction of any two elements, x and y, is their greatest lower bound, that is, the greatest thing from which one can get to both x and y going up the arrows. Thus, for example, $b \wedge n = 0$ and $b \wedge 1 = b$. The disjunction of two elements, x and y, is the least upper bound, that is, the least thing from which one can get to both x and y going down the arrows. Thus, for example, $b \vee n = 1$, $b \vee 1 = 1$. Negation toggles 0 and 1, and maps each of n and b to itself.[2]

8.4.4 Since validity in *FDE* is defined in terms of truth preservation, the set of designated values is $\{1, b\}$ (true only, and both true and false).

8.4.5 This is not one of the many-valued logics that we met in the last chapter, but two of the ones that we did meet there are closely related to *FDE*.

8.4.6 Suppose that we consider an *FDE* interpretation that satisfies the constraint:

Exclusion: for no p, $p\rho 1$ and $p\rho 0$

[2] In fact, this structure is more than a mnemonic. The lattice is one of the most fundamental of a group of structures called 'De Morgan lattices', which can be used to give a different semantics for *FDE*.

i.e., no propositional parameter is both true and false. Then it is not difficult to check that the same holds for every sentence, A.[3] That is, nothing takes the value b.

8.4.7 The logic defined in terms of truth preservation over all interpretations satisfying this constraint is, in fact, K_3. For if we take the above matrices, and ignore the rows and columns for b, we get exactly the matrices for K_3 (identifying n with i). (In K_3, $A \supset B$ can be defined as $\neg A \vee B$, as we observed in 7.3.8.)

8.4.8 K_3 is sound and complete with respect to the tableaux of the previous section, augmented by one extra closure rule: a branch closes if it contains nodes of the form $A, +$ and $\neg A, +$. (This is proved in 8.7.8.) Here, for example, is a tableau showing that $p \wedge \neg p \vdash_{K_3} q$. (The tableau is open in *FDE*.)

$$\begin{array}{c} p \wedge \neg p, + \\ q, - \\ p, + \\ \neg p, + \\ \times \end{array}$$

Counter-models are read off from open branches of tableaux in exactly the same way as in *FDE*.

8.4.9 Suppose, on the other hand, that we consider an *FDE* interpretation that satisfies the constraint:

Exhaustion: for all p, either $p\rho 1$ or $p\rho 0$

i.e., every propositional parameter is either true or false – and maybe both. Then it is not difficult to check that, again, the same holds for every sentence, A.[4] That is, nothing takes the value n.

[3] *Proof*: The proof is by an induction over the complexity of sentences. Suppose that it is true for A and B; we show that it is true for $\neg A$, $A \wedge B$ and $A \vee B$. Suppose that $\neg A\rho 1$ and $\neg A\rho 0$; then $A\rho 0$ and $A\rho 1$, contrary to supposition. Suppose that $A \wedge B\rho 1$ and $A \wedge B\rho 0$; then $A\rho 1$ and $B\rho 1$, and either $A\rho 0$ or $B\rho 0$; hence, either $A\rho 1$ and $A\rho 0$, or the same for B. Both cases are false, by assumption. The argument for $A \vee B$ is similar.

[4] *Proof*: The proof is by an induction over the complexity of sentences. Suppose that it is true for A and B; we show that it is true for $\neg A$, $A \wedge B$ and $A \vee B$. Suppose that either $A\rho 1$ or $A\rho 0$; then either $\neg A\rho 0$ or $\neg A\rho 1$. Since $A\rho 1$ or $A\rho 0$, and $B\rho 1$ or $B\rho 0$, then either $A\rho 1$ and $B\rho 1$, and so $A \wedge B\rho 1$; or $A\rho 0$ or $B\rho 0$, and so $A \wedge B\rho 0$. The argument for $A \vee B$ is similar.

8.4.10 The logic defined by truth preservation over all interpretations satisfying this constraint is, in fact, *LP*. For if we take the matrices of 8.4.2 and ignore the rows and columns for n, we get exactly the matrices for *LP* (identifying b with i). (Again, in *LP*, $A \supset B$ can be defined as $\neg A \vee B$, as we observed in 7.4.6.)

8.4.11 *LP* is sound and complete with respect to the tableaux of the previous section, augmented by one extra closure rule: a branch closes if it contains nodes of the form $A, -$ and $\neg A, -$. (This is proved in 8.7.9.) Here, for example, is a tableau showing that $p \vdash_{LP} q \vee \neg q$. (The tableau is open in *FDE*.)

$$\begin{array}{c} p, + \\ q \vee \neg q, - \\ q, - \\ \neg q, - \\ \times \end{array}$$

Counter-models are read off from open branches of tableaux by employing the following rule: if $p, -$ is not on the branch (and so, in particular, if $p, +$ is), set $p\rho 1$; and if $\neg p, -$ is not on the branch (and so, in particular, if $\neg p, +$ is), set $p\rho 0$.

8.4.12 Finally, and of course, if an interpretation satisfies both *Exclusion* and *Exhaustion*, then for every p, $p\rho 0$ or $p\rho 1$, but not both, and the same follows for arbitrary A. In this case, we have what is, in effect, an interpretation for classical logic. Adding the closure rules for K_3 and *LP* to those of *FDE*, therefore gives us a new tableau procedure for classical logic.

8.4.13 Since all K_3 interpretations are *FDE* interpretations, and all *LP* interpretations are *FDE* interpretations, *FDE* is a sub-logic of K_3 and *LP*. It is a proper sub-logic of each, as the tableaux of 8.4.8 and 8.4.11 show.

8.4a Relational Semantics and Tableaux for $Ł_3$ and RM_3

8.4a.1 Before we move on to a different kind of semantics for *FDE*, it is worth noting that the semantics for $Ł_3$ and RM_3 can be reformulated in a relational fashion as well. The only difference from K_3 and *LP* (respectively) concerns the appropriate conditional.

8.4a.2 For $Ł_3$, we consult the truth table of 7.3.8, and recall that i is n – that is, neither true (relates to 1) nor false (relates to 0). It is not difficult to check that:

$A \supset B\rho 1$ iff $A\rho 0$ or $B\rho 1$ or (none of $A\rho 1$, $A\rho 0$, $B\rho 1$, $B\rho 0$)
$A \supset B\rho 0$ iff $A\rho 1$ and $B\rho 0$

8.4a.3 For *LP*, we consult the truth table of 7.4.6, and recall that i is b – that is, both true (relates to 1) and false (relates to 0). It is not difficult to check that:

$A \supset B\rho 1$ iff it is not the case that $A\rho 1$ or it is not the case that $B\rho 0$ or ($A\rho 1$ and $A\rho 0$ and $B\rho 1$ and $B\rho 0$)
$A \supset B\rho 0$ iff $A\rho 1$ and $B\rho 0$

8.4a.4 In virtue of these truth conditions, it is straightforward to give tableaux systems for the two logics. The tableaux for $Ł_3$ are the same as those for K_3, with the additional rules for $\supset$:

$$
\begin{array}{ccc}
 & A \supset B, + & \\
\swarrow & \downarrow & \searrow \\
\neg A, + & B, + & A \vee \neg A, - \\
 & & B \vee \neg B, -
\end{array}
\qquad
\begin{array}{cc}
\multicolumn{2}{c}{A \supset B, -} \\
\swarrow & \searrow \\
A, + & \neg B, + \\
B, - & \neg A, -
\end{array}
$$

$$
\begin{array}{c}
\neg(A \supset B), + \\
\downarrow \\
A, + \\
\neg B, +
\end{array}
\qquad
\begin{array}{cc}
\multicolumn{2}{c}{\neg(A \supset B), -} \\
\swarrow & \searrow \\
A, - & \neg B, -
\end{array}
$$

8.4a.5 The tableaux for RM_3 are the same as those for *LP*, with the additional rules for $\supset$:

$$
\begin{array}{ccc}
 & A \supset B, + & \\
\swarrow & \downarrow & \searrow \\
A, - & \neg B, - & A \wedge \neg A, + \\
 & & B \wedge \neg B, +
\end{array}
\qquad
\begin{array}{cc}
\multicolumn{2}{c}{A \supset B, -} \\
\swarrow & \searrow \\
A, + & \neg B, + \\
B, - & \neg A, -
\end{array}
$$

$$
\begin{tikzcd}[column sep=small]
\neg(A \supset B), + \arrow[d] \\
A, + \\
\neg B, +
\end{tikzcd}
\qquad
\begin{tikzcd}[column sep=small]
& \neg(A \supset B), - \arrow[dl] \arrow[dr] & \\
A, - & & \neg B, -
\end{tikzcd}
$$

8.4a.6 The tableau systems are sound and complete with respect to the appropriate semantics. (See 8.10, problem 11.)

8.5 The Routley Star

8.5.1 We now have two equivalent semantics for *FDE*, a relational semantics and a many-valued semantics.[5] For reasons to do with later chapters, we should have a third. This is a two-valued possible-world semantics, which treats negation as an intensional operator; that is, as an operator whose truth conditions require reference to worlds other than the world at which truth is being evaluated.

8.5.2 Specifically, we assume that each world, w, comes with a mate, w^*, its *star world*, such that $\neg A$ is true at w if A is false, not at w, but at w^*. If $w = w^*$ (which may happen), then these conditions just collapse into the classical conditions for negation; but if not, they do not. The star operator is often described with a variety of metaphors; for example, it is sometimes described as a reversal operator; but it is hard to give it and its role in the truth conditions for negation a satisfying intuitive interpretation.

8.5.3 Formally, a *Routley interpretation* is a structure $\langle W, *, \nu \rangle$, where W is a set of worlds, $*$ is a function from worlds to worlds such that $w^{**} = w$, and ν assigns each propositional parameter either the value 1 or the value 0 at each world. ν is extended to an assignment of truth values for all formulas by the conditions:

$\nu_w(A \wedge B) = 1$ if $\nu_w(A) = 1$ and $\nu_w(B) = 1$; otherwise it is 0.
$\nu_w(A \vee B) = 1$ if $\nu_w(A) = 1$ or $\nu_w(B) = 1$; otherwise it is 0.
$\nu_w(\neg A) = 1$ if $\nu_{w^*}(A) = 0$; otherwise it is 0.

[5] At least, they are equivalent given the standard set-theoretic reasoning employed in the reformulation. Such reasoning employs classical logic, however, and in a set theory based on a paraconsistent logic it may fail. See Priest (1993).

Note that $\nu_{w^*}(\neg A) = 1$ iff $\nu_{w^{**}}(A) = 0$ iff $\nu_w(A) = 0$. In other words, given a pair of worlds, w and w^*, each of A and $\neg A$ is true exactly once. Validity is defined in terms of truth preservation over all worlds of all interpretations.

8.5.4 Appropriate tableaux for these semantics are easy to construct. Nodes are now of the form $A, +x$ or $A, -x$, where x is either i or $i^\#$, i being a natural number. (In fact, i will always be 0, but we set things up in a slightly more general way for reasons to do with later chapters.) Intuitively, $i^\#$ represents the star world of i. Closure occurs if we have a pair of the form $A, +x$ and $A, -x$. The initial list comprises a node $B, +0$ for every premise, B, and $A, -0$, where A is the conclusion. The tableau rules are as follows, where x is either i or $i^\#$, and whichever of these it is, $\bar{x}$ is the other.

$$
\begin{array}{c}
A \wedge B, +x \\
\downarrow \\
A, +x \\
B, +x
\end{array}
\qquad
\begin{array}{ccc}
 & A \wedge B, -x & \\
\swarrow & & \searrow \\
A, -x & & B, -x
\end{array}
$$

$$
\begin{array}{ccc}
 & A \wedge B, +x & \\
\swarrow & & \searrow \\
A, +x & & B, +x
\end{array}
\qquad
\begin{array}{c}
A \vee B, -x \\
\downarrow \\
A, -x \\
B, -x
\end{array}
$$

$$
\begin{array}{c}
\neg A, +x \\
\downarrow \\
A, -\bar{x}
\end{array}
\qquad
\begin{array}{c}
\neg A, -x \\
\downarrow \\
A, +\bar{x}
\end{array}
$$

8.5.5 Here are tableaux demonstrating that $\neg(B \wedge \neg C) \wedge A \vdash (\neg B \vee C) \vee D$ and $p \wedge (q \vee \neg q) \nvdash r$:

$$
\begin{array}{c}
\neg(B \wedge \neg C) \wedge A, +0 \\
(\neg B \vee C) \vee D, -0 \\
(\neg B \vee C), -0 \\
D, -0 \\
\neg B, -0 \\
C, -0 \\
B, +0^\# \\
\downarrow
\end{array}
$$

$$\begin{array}{ccc} & \neg(B \wedge \neg C), +0 & \\ & A, +0 & \\ & B \wedge \neg C, -0^{\#} & \\ \swarrow & & \searrow \\ B, -0^{\#} & & \neg C, -0^{\#} \\ \times & & C, +0 \\ & & \times \end{array}$$

Line two is pursued as far as possible. Then line one is pursued to produce closure.

$$\begin{array}{ccc} & p \wedge (q \vee \neg q), +0 & \\ & r, -0 & \\ & p, +0 & \\ & q \vee \neg q, +0 & \\ \swarrow & & \searrow \\ q, +0 & & \neg q, +0 \\ & & q, -0^{\#} \end{array}$$

8.5.6 To read off a counter-model from an open branch: $W = \{w_0, w_{0^\#}\}$ (there are only ever two worlds); $w_0^* = w_{0^\#}$ and $(w_{0^\#})^* = w_0$. (W and $*$ are always the same, no matter what the tableau.) ν is such that if $p, +x$ occurs on the branch, $\nu_{w_x}(p) = 1$, and if $p, -x$ occurs on the branch, $\nu_{w_x}(p) = 0$. Thus, the counter-model defined by the righthand open branch of the second tableau of 8.5.5 has $\nu_{w_0}(p) = 1$, $\nu_{w_0}(r) = 0$ and $\nu_{w_{0^\#}}(q) = 0$. It is easy to check directly that this interpretation does the job. Since q is false at $w_{0^\#}$, $\neg q$ is true at w_0, as, therefore, is $q \vee \neg q$; but p is true at w_0, hence $p \wedge (q \vee \neg q)$ is true at w_0. But r is false at w_0, as required.

8.5.7 The soundness and completeness of this tableau procedure is proved in 8.7.10–8.7.16.

8.5.8 It is not at all obvious that the $*$ semantics are equivalent to the relational semantics, but it is not too difficult to establish this. Essentially, it is because a relational interpretation, ρ, is equivalent to a pair of worlds, w and w^*. Specifically, the relation and the worlds do exactly the same job when they are related by the condition:

$\nu_w(p) = 1$ iff $p\rho 1$
$\nu_{w^*}(p) = 0$ iff $p\rho 0$

for all parameters, p. The proof of the equivalence is given in 8.7.17 and 8.7.18.

8.6 Paraconsistency and the Disjunctive Syllogism

8.6.1 As we have seen (8.4.8 and 8.4.11), both of the following are false in *FDE*: $p \models q \vee \neg q$, $p \wedge \neg p \models q$. This is essentially because there are truth-value gaps (for the former) and truth-value gluts (for the latter). In particular, then, *FDE* does not suffer from the problem of explosion (4.8).

8.6.2 A logic in which the inference from p and $\neg p$ to an arbitrary conclusion is not valid is called *paraconsistent*. *FDE* is therefore paraconsistent, as is *LP* (7.4.4).

8.6.3 It is not only explosion that fails in *FDE* (and *LP*). The disjunctive syllogism (*DS*) is also invalid: $p, \neg p \vee q \not\models_{FDE} q$. (Relational counter-model: $p\rho 1$ and $p\rho 0$, but just $q\rho 0$.)

8.6.4 This is a significant plus. We have seen the *DS* involved in two problematic arguments: the argument for the material conditional of 1.10, and the Lewis argument for explosion of 4.9.2. We can now see that these arguments do not work, and (at least one reason) why.[6]

8.6.5 Note, also, that the *DS* is just *modus ponens* for the material conditional. Since this fails, we have another argument against the adequacy of the material conditional to represent the real conditional.

8.6.6 The failure of the *DS* has also been thought by some to be a significant minus. First, it is claimed that the DS is intuitively valid. For if $\neg p \vee q$ is true, either $\neg p$ or q is true. But, the argument continues, if p is true, this rules out the truth of $\neg p$. Hence, it must be q that is true. But once one countenances the possibility of truth-value gluts, this argument is patently wrong. The truth of p does not rule out the truth of $\neg p$: both may hold. From this perspective, the inference is intuitively invalid.

8.6.7 A more persuasive objection is that we frequently use, and seem to need to use, the *DS* to reason, and we get the right results. Thus, we know

[6] For good measure, the argument of 4.9.3 for the validity of the inference from A to $B \vee \neg B$ is also invalid in *FDE*, since $p \not\models (p \wedge q) \vee (p \wedge \neg q)$, as may be checked.

that you are either at home or at work. We ascertain that you are not at home, and infer that you are at work – which you are. If the *DS* is invalid, this form of reasoning would seem to be incorrect.

8.6.8 If the *DS* fails, then the inference about being at home or work is not deductively valid. It may be perfectly legitimate to use it, none the less. There are a number of ways of spelling this idea out in detail, but at the root of all of them is the observation that when the *DS* fails, it does so because the premise p involved is a truth-value glut. If the situation about which we are reasoning is consistent – as it is, presumably, in this case – the *DS* cannot lead us from truth to untruth. So it is legitimate to use it. This fact will underwrite its use in most situations we come across, since consistency is, arguably, the norm.

8.6.9 In the same way, if we have some collection, X, one cannot infer from the fact that some other collection, Y, is a proper subset of X that it is smaller.[7] But provided that we are working with collections that are finite, this inference is perfectly legitimate: violations can occur only when infinite sets are involved.

8.6.10 Thus, this objection can also be set aside.

8.7 *Proofs of Theorems

8.7.1 The soundness and completeness proofs for the relational semantics for *FDE* modify those for classical logic (1.11).

8.7.2 Definition: Let ρ be any relational interpretation. Let b be any branch of a tableau. ρ is *faithful* to b iff for every node, $A, +$, on the branch, $A\rho 1$, and for every node, $A, -$, on the branch, it is not the case that $A\rho 1$.

8.7.3 Soundness Lemma: If ρ is faithful to a branch of a tableau, b, and a tableau rule is applied to b, then ρ is faithful to at least one of the branches generated.

[7] For example, the set of all natural numbers is the same size as the set of all even numbers, as can be seen by making the following correlation:

$$\begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & \dots \\ \updownarrow & \updownarrow & \updownarrow & \updownarrow & \updownarrow & \\ 0 & 2 & 4 & 6 & 8 & \dots \end{array}$$

Proof:

The proof is by a case-by-case examination of the tableau rules. First, the rules for $\wedge$. Suppose that we apply the rule for $A \wedge B, +$; then since ρ is faithful to the branch, $A \wedge B\rho 1$. Hence, $A\rho 1$ and $B\rho 1$. Hence, ρ is faithful to the extended branch. Next, suppose that we apply the rule for $A \wedge B, -$; then since ρ is faithful to the branch, it is not the case that $A \wedge B\rho 1$. Hence, either it is not the case that $A\rho 1$ or it is not the case that $B\rho 1$. Hence, ρ is faithful to either the left branch or the right branch. The argument for $\vee$ is similar. For the other rules, it is easy to check that in *FDE*, $\neg(A \wedge B)$ is true under an evaluation iff $\neg A \vee \neg B$ is true; the same goes for $\neg(A \vee B)$ and $\neg A \wedge \neg B$, and $\neg\neg A$ and A. (Details are left as an exercise.) The cases for the other rules follow simply from these facts. ■

8.7.4 SOUNDNESS THEOREM FOR *FDE*: For finite Σ, if $\Sigma \vdash A$ then $\Sigma \models A$.

Proof:

The proof follows from the Soundness Lemma in the usual way. ■

8.7.5 DEFINITION: Let b be an open branch of a tableau. The interpretation *induced* by b is the interpretation, ρ, such that for every propositional parameter, p:

$p\rho 1$ iff $p, +$ occurs on b
$p\rho 0$ iff $\neg p, +$ occurs on b

8.7.6 COMPLETENESS LEMMA: Let b be an open completed branch of a tableau. Let ρ be the interpretation induced by b. Then:

if $A, +$, occurs on b, then $A\rho 1$
if $A, -$ occurs on b, then it is not the case that $A\rho 1$
if $\neg A, +$, occurs on b, then $A\rho 0$
if $\neg A, -$ occurs on b, then it is not the case that $A\rho 0$

Proof:

The proof is by an induction on the complexity of A. If A is a propositional parameter, p: if $p, +$ occurs on b, then $p\rho 1$ by definition. If $p, -$ occurs on b, then $p, +$ does not occur on b, since it is open. Hence, by definition, it is not the case that $p\rho 1$. The cases for 0 are similar. For $B \wedge C$: if $B \wedge C, +$ occurs on b, then $B, +$ and $C, +$ occur on b. By induction hypothesis, $B\rho 1$ and $C\rho 1$. Hence, $B \wedge C\rho 1$ as required. The argument for $B \wedge C, -$ is similar. If $\neg(B \wedge C), +$

occurs on b, then by applications of a De Morgan rule and a disjunction rule, either $\neg B, +$ or $\neg C, +$ are on b. By induction hypothesis, either $B\rho 0$ or $C\rho 0$. In either case, $B \wedge C\rho 0$. The case for $\neg(B \wedge C), -$ is similar. The argument for $\vee$ is symmetric. This leaves negation. Suppose that $\neg B, +$ occurs on b. Since the result holds for B, $B\rho 0$. Hence, $\neg B\rho 1$, as required. Similarly for $\neg B, -$. If $\neg\neg B, +$ is on b, $B, +$ is on b. Hence, by induction hypothesis, $B\rho 1$, and so $\neg\neg B\rho 1$ as required. The case for $\neg\neg B, -$ is similar. ■

8.7.7 Completeness Theorem for *FDE*: For finite Σ, if $\Sigma \models A$ then $\Sigma \vdash A$.

Proof:
The proof follows from the Completeness Lemma in the usual way. ■

8.7.8 Theorem: The tableau rules of 8.4.8 are sound and complete for K_3.

Proof:
The soundness proof is exactly the same as that for *FDE*. (If the rules are sound with respect to *FDE* interpretations, they are sound with respect to K_3 interpretations, which are a special case.) The completeness proof is also essentially the same. All we have to check, in addition, is that the induced interpretation is a K_3 interpretation. It cannot be the case that $p\rho 1$ and $p\rho 0$, for then we would have both $p, +$ and $\neg p, +$ on b. But this is impossible, or b would be closed by the new closure rule. ■

8.7.9 Theorem: The tableau rules of 8.4.11 are sound and complete for *LP*.

Proof:
The soundness proof is exactly the same as that for *FDE*. (If the rules are sound with respect to *FDE* interpretations, they are sound with respect to *LP* interpretations, which are a special case.) In the completeness proof, the induced interpretation is defined slightly differently, thus:

$p\rho 1$ iff $p, -$ is not on b
$p\rho 0$ iff $\neg p, -$ is not on b

Note that this makes ρ an *LP* interpretation. By the new closure rule, either $p, -$ or $\neg p, -$ is not on b. Hence, either $p\rho 1$ or $p\rho 0$. In the Completeness Lemma, the new definition makes the argument for the basis case different. If $p, +$ occurs on b, then $p, -$ does not occur on b, by the *FDE* closure rule, so $p\rho 1$. If $p, -$ occurs on b, then it is not the case that $p\rho 1$, by definition. The argument for $\neg p$ is the same. The rest of the Completeness Lemma, and the proof of the Completeness Theorem itself, are as usual. ■

8.7.10 The soundness and completeness proofs for the $*$ semantics are variations on those for intuitionist tableaux (6.7). We start off, as usual, with a redefinition of faithfulness.

8.7.11 DEFINITION: Let $\mathcal{I} = \langle W, *, \nu \rangle$ be any Routley interpretation, and b be any branch of a tableau. Then $\mathcal{I}$ is faithful to b iff there is a map, f, from the natural numbers to W, such that:

for every node $A, +x$ on b, A is true at $f(x)$ in $\mathcal{I}$,
for every node $A, -x$ on b, A is false at $f(x)$ in $\mathcal{I}$,

where $f(i^{\#})$ is, by definition, $f(i)^*$.

8.7.12 SOUNDNESS LEMMA: Let b be any branch of a tableau, and $\mathcal{I} = \langle W, *, \nu \rangle$ be any Routley interpretation. If $\mathcal{I}$ is faithful to b, and a tableau rule is applied, then it produces at least one extension, b', such that $\mathcal{I}$ is faithful to b'.

Proof:
Let f be a function which shows $\mathcal{I}$ to be faithful to b. The proof proceeds by a case-by-case consideration of the tableau rules. Suppose we apply the rule to $A \wedge B, +x$, then, by assumption $A \wedge B$ is true at $f(x)$. Thus, A and B are both true at $f(x)$, and so f shows that $\mathcal{I}$ is faithful to b'. If we apply the rule to $A \wedge B, -x$, then, by assumption, $A \wedge B$ is false at $f(x)$. Consequently, A is false at $f(x)$ or B is false at $f(x)$, i.e., f shows that $\mathcal{I}$ is faithful to either the left branch or the right branch. The arguments for the rules for disjunction are also similar. This leaves the rules for negation. Suppose that we apply the rule to $\neg A, +i$. Then, by assumption, $\neg A$ is true at $f(i)$. Hence, A is false at $f(i)^*$, as required. If we apply the rule to $\neg A, +i^{\#}$, then we know that $\neg A$ is true at $f(i)^*$. Hence, A, is false at $f(i)$, as required. The argument for the other negation rule is similar. ■

8.7.13 SOUNDNESS THEOREM: For finite Σ, if $\Sigma \vdash A$ then $\Sigma \models A$.

Proof:
This follows from the Soundness Lemma in the usual way. ■

8.7.14 Definition: Let b be an open branch of a tableau. The interpretation, $\mathcal{I} = \langle W, *, \nu \rangle$, induced by b, is defined as in 8.5.6. $W = \{w_0, w_{0^\#}\}.w_0^* = w_{0^\#}$, $(w_{0^\#})^* = w_0$. ν is such that:

$\nu_{w_x}(p) = 1$ if $p, +x$ is on b
$\nu_{w_x}(p) = 0$ if $p, -x$ is on b

(where x is either 0 or $0^\#$). Since the branch is open, this is well defined. Note also that, by the definition of $*$, $w_x^{**} = w_x$, i.e., the induced interpretation is a Routley interpretation.

8.7.15 Completeness Lemma: Let b be any open completed branch of a tableau. Let $\mathcal{I} = \langle W, *, \nu \rangle$ be the interpretation induced by b. Then:

if $A, +x$ is on b, A is true at w_x
if $A, -x$ is on b, A is false at w_x

Proof:
This is proved by induction on the complexity of A. If A is atomic, the result is true by definition. If $B \wedge C, +x$ occurs on b, then $B, +x$ and $C, +x$ occur on b. By induction hypothesis, B and C are true at w_x. Hence, $B \wedge C$ is true at w_x. If $B \wedge C, -x$ occurs on b, then either $B, -x$, or $C, -x$ occurs on b. By induction hypothesis, B is false at w_x or C is false at w_x. Hence, $B \wedge C$ is false at w_x as required. The cases for disjunction are similar. For negation: if $\neg B, +x$ occurs on b, then $B, -\bar{x}$ occurs on b. By induction hypothesis, B is false at $w_{\bar{x}}$; hence, by the definition of $*$, B is false at w_x^*, that is, $\neg B$ is true at w_x, as required. The other negation rule is the same. ■

8.7.16 Completeness Theorem: For finite Σ, if $\Sigma \models A$ then $\Sigma \vdash A$.

Proof:
The result follows from the Completeness Lemma in the usual fashion. ■

8.7.17 Theorem: If $\Sigma \models A$ under the relational semantics, $\Sigma \models A$ under the Routley semantics.

Proof:
We prove the contrapositive. Suppose that there is a Routley interpretation, $\langle W, *, \nu \rangle$, and a world $w \in W$, which makes all the members of Σ true and

A false (i.e., untrue). Define a relational interpretation, ρ, by the following conditions:

$p\rho 1$ iff $\nu_w(p) = 1$
$p\rho 0$ iff $\nu_{w^*}(p) = 0$

If it can be shown that the displayed conditions hold for all formulas, then the result follows. This is proved by induction on the construction of A. If A is a propositional parameter, the result holds by definition. Suppose that the result holds for B and C. $B \wedge C\rho 1$ iff $B\rho 1$ and $C\rho 1$; iff $\nu_w(B) = 1$ and $\nu_w(C) = 1$, by induction hypothesis; iff $\nu_w(B \wedge C) = 1$. $B \wedge C\rho 0$ iff $B\rho 0$ or $C\rho 0$; iff $\nu_{w^*}(B) = 0$ or $\nu_{w^*}(C) = 0$, by induction hypothesis; iff $\nu_{w^*}(B \wedge C) = 0$, as required. The cases for disjunction are similar. $\neg A\rho 1$ iff $A\rho 0$; iff $\nu_{w^*}(A) = 0$, by induction hypothesis; iff $\nu_w(\neg A) = 1$. $\neg A\rho 0$ iff $A\rho 1$; iff $\nu_w(A) = 1$, by induction hypothesis; iff $\nu_{w^*}(\neg A) = 0$, as required. ■

8.7.18 Theorem: If $\Sigma \models A$ under the Routley semantics, $\Sigma \models A$ under the relational semantics.

Proof:
We prove the contrapositive. Suppose that there is a relational interpretation, ρ, which makes all the members of Σ true and A untrue. Define a Routley interpretation, $\langle W, *, \nu\rangle$, where $W = \{a, b\}$, $a^* = b$ and $b^* = a$, and ν is defined by the conditions:

$\nu_a(p) = 1$ iff $p\rho 1$
$\nu_b(p) = 1$ iff it is not the case that $p\rho 0$

If it can be shown that the displayed condition holds for all formulas, then the result follows. This is proved by induction on the construction of A. If A is a propositional parameter, the result holds by definition. Suppose that the result holds for B and C. $\nu_a(B \wedge C) = 1$ iff $\nu_a(B) = 1$ and $\nu_a(C) = 1$; iff $B\rho 1$ and $C\rho 1$, by induction hypothesis; iff $B \wedge C\rho 1$. $\nu_b(B \wedge C) = 1$ iff $\nu_b(B) = 1$ and $\nu_b(C) = 1$; iff it is not the case that $B\rho 0$ and it is not the case that $C\rho 0$, by induction hypothesis; iff it is not the case that $B \wedge C\rho 0$. The cases for disjunction are similar. $\nu_a(\neg B) = 1$ iff $\nu_{a^*}(B) = 0$; iff $\nu_b(B) = 0$; iff $B\rho 0$ by induction hypothesis; iff $\neg B\rho 1$. $\nu_b(\neg B) = 1$ iff $\nu_{b^*}(B) = 0$; iff $\nu_a(B) = 0$; iff it is not the case that $B\rho 1$, by induction hypothesis; iff it is not the case that $\neg B\rho 0$. ■

8.8 History

The logic *FDE* is the core of a family of relevant logics (which we will meet in later chapters), developed by the US logicians Anderson and Belnap, starting at the end of the 1950s. (Strictly speaking, $A \models_{FDE} B$ iff $A \rightarrow B$ is valid in their system of first degree entailment.) See Anderson and Belnap (1975, esp. ch. 3). The relational semantics were discovered by Dunn in the 1960s as a spin-off from his algebraic semantics for *FDE* (on which, see Anderson and Belnap 1975, sect. 18). He published them only later, however, by which time they had been discovered by others too. The Routley semantics for *FDE* were first given by Richard Routley (later Sylvan) and Val Routley (later Plumwood) in Routley and Routley (1972). There are many paraconsistent logics. *FDE*, *LP* and the relevant logics that we will meet in later chapters constitute one kind. Paraconsistent logics of different kinds were developed by the Polish logician Jaśkowski in 1948 (see Jaśkowski 1969) and the Brazilian logician da Costa in the 1960s (see da Costa 1974). A general history and survey of paraconsistent logics can be found in Priest (2002a).

8.9 Further Reading

On the various semantics for *FDE* covered in this chapter, see Priest (2002a, sects. 4.6 and 4.7); and for a much more detailed account, see Routley, Plumwood, Meyer and Brady (1982, sects. 3.1 and 3.2). For the Routleys' own discussion of the meaning of the star operator, see Routley and Routley (1985). For a defence of the Routley star, see Restall (1999). Discussions of the disjunctive syllogism can be found in Burgess (1983), Mortensen (1983) and Priest (1987, ch. 8).

8.10 Problems

1. Using the tableau procedure of 8.3, determine whether or not the following are true in *FDE*. If the inference is invalid, specify a relational counter-model.
   (*a*) $p \wedge q \vdash p$
   (*b*) $p \vdash p \vee q$
   (*c*) $p \wedge (q \vee r) \vdash (p \wedge q) \vee (p \wedge r)$
   (*d*) $p \vee (q \wedge r) \vdash (p \vee q) \wedge (p \vee r)$

(e) $p \vdash \neg\neg p$
(f) $\neg\neg p \vdash p$
(g) $(p \wedge q) \supset r \vdash (p \wedge \neg r) \supset \neg q$
(h) $p \wedge \neg p \vdash p \vee \neg p$
(i) $p \wedge \neg p \vdash q \vee \neg q$
(j) $p \vee q \vdash p \wedge q$
(k) $p, \neg(p \wedge \neg q) \vdash q$
(l) $(p \wedge q) \supset r \vdash p \supset (\neg q \vee r)$

2. For the inferences of problem 1 that are invalid, determine which ones are valid in K_3 and LP, using the appropriate tableaux.
3. Check all the details omitted in 8.4.2.
4. By checking the truth tables of 8.4.2, note that if A and B have truth value n, then so do $A \vee B, A \wedge B$ and $\neg A$. Infer that if A is any formula all of whose propositional parameters take the value n, it, too, takes the value n. Hence infer that there is no formula, A, such that $\models_{FDE} A$.
5. Similarly, show that if A is a formula all of whose propositional parameters take the value b, then A takes the value b. Hence, show that if A and B have no propositional parameters in common, $A \not\models_{FDE} B$. (Hint: Assign all the parameters in A the value b, and all the parameters in B the value n.)
6. Repeat problem 1 with the $*$ semantics and tableaux of 8.5.
7. Using the $*$ semantics, show that if $A \models_{FDE} B$, then $\neg B \models_{FDE} \neg A$. (Hint: Assume that there is a counter-model for the consequent.) Why is this not obvious with the many-valued or the relational semantics? (Note that contraposition of this kind does not extend to multiple-premise inferences: $p, q \models_{FDE} p$, but $p, \neg p \not\models_{FDE} \neg q$.)
8. Test the validity of the inferences in 7.5.2 using the tableau of this chapter.
9. Under what conditions is it legitimate to employ a deductively invalid inference?
10. *Check the details omitted in 8.7.3.
11. *Show that the tableaux of 8.4a.4 and 8.4a.5 are sound and complete with respect to the semantics of $Ł_3$ and RM_3. (Hint: consult 8.7.8 and 8.7.9.)