Contents

Introduction

“*Necessity is the mother of invention.*” 1

Part I: What is in this book — details.

There are several different types of formal proof procedures that logicians have invented. The ones we consider are: 1) tableau systems, 2) Gentzen sequent calculi, 3) natural deduction systems, and 4) axiom systems. We present proof procedures of each of these types for the most common normal modal logics: S5, S4, B, T, D, K, K4, D4, KB, DB, and also G, the logic that has become important in applications of modal logic to the proof theory of Peano arithmetic. Further, we present a similar variety of proof procedures for an even larger number of regular, non-normal modal logics (many introduced by Lemmon). We also consider some quasi-regular logics, including S2 and S3. Virtually all of these proof procedures are studied in both propositional and first-order versions (generally with and without the Barcan formula). Finally, we present the full variety of proof methods for Intuitionistic logic (and of course Classical logic too).

We actually give two quite different kinds of tableau systems for the logics we consider, two kinds of Gentzen sequent calculi, and two kinds of natural deduction systems. Each of the two tableau systems has its own uses; each provides us with different information about the logics involved. They complement each other more than they overlap. Of the two Gentzen systems, one is of the conventional sort, common in the literature. The other kind of Gentzen system is ”symmetric” in its behavior, and 2
lends itself nicely to use in proofs of Interpolation Theorems. Consequently we are able to give simple, constructive proofs of Craig (and Lyndon) type theorems for most of the logics we consider. Finally, the two kinds of natural deduction systems differ in the way they associate *strict* subordinate derivations with alternate worlds in Kripke models. Of the two systems, one is Fitch-like, while the other has a rather different flavor.

A word about our methods. Although we are studying proof procedures, this is not a book in the constructive proof theory tradition. We generally show our proof procedures do what we say they do by relating them to models — Kripke models. Thus completeness proofs loom large in this work. And that requires a further, and important, comment.

Rather than go through a fresh completeness argument for each new proof procedure we consider, we have instead abstracted the underlying construction to establish, for each logic, a *Model Existence Theorem*, stated in terms of *Consistency Properties* (more on this in part II). Such theorems were originally developed for Classical first-order logic, and have played a significant role for extensions of it, in particular, for certain of the infinitary logics. The nice thing about a Model Existence Theorem is that with a single proof one gets, as easy corollaries, several completeness results, compactness, Löwenheim-Skolem, interpolation (often), decidability (often), and deduction theorems. So Consistency Properties should be seen as the unifying theme of most of this book. We present them as abstractions of the tableau method (again, more in part II). 3

Finally a word about what this book is not. It is not a book that considers the philosophical motivation for various modal logics or for Intuitionistic logic. we recommend Hughes and Cresswell [1968]. And it is not a book on model theory. Of course we use Kripke models, and we manipulate them to some extent, but it is all pretty elementary. It is proof procedures that we are interested in. But Modal and Intuitionistic model theory has gone through a tremendous development over the last decade or so, and the least we can do is recommend a few works to be consulted for details. The following are suitably comprehensive: for modal logic Goldblatt [1976, 1976A], Bowen [1979], and for Intuitionistic logic, Gabbay [1981].

Part II: What is in this book - generalities and background.

In 1935 Gentzen introduced what is now called Gentzen's sequent calculus, for both Classical and Intuitionistic logics. For these proof systems he gave a constructive demonstration of what has since become known as Gentzen's Hauptsatz: that a certain rule of derivation called *cut* could be eliminated without changing the class of provable formulas. From this, consistency results followed easily, as did decidability results for the propositional parts of the logics treated, and also several other results of considerable interest. Once the cut rule was eliminated, Gentzen's proof systems obeyed a *subformula principle*: a proof of X need only involve subformulas of X. This was in marked contrast to the axiomatic proof procedures common at the time (and today too). 4

Since then, Gentzen's methods have been extended to several other logics, with mixed success. The difficulty is that while it is often clear how to formulate a Gentzen sequent calculus for a given logic, one can not always prove a cut-elimination theorem for the resulting system. Indeed, Scott [1973] points out that this is the rule, rather than the exception, "*…in general cut cannot be eliminated. Gentzen's Elimination Theorem holds only for very special relations.*"

Thus Gentzen's methods do not provide anything like a universal approach to logics. Nonetheless, when they do work, they work very well indeed. In this book we show they succeed for a surprising number of well-known modal logics, both normal and regular, both propositional and first-order. And of course we include Classical and Intuitionistic logics too.

There are certain standard logics to which these methods do not apply in as direct a fashion. For the most part they are logics in which the "future can affect the past." For example, consider the logics **B** and **S5**. The Kripke models for these are *symmetric* so, having moved from world Γ to world Δ, we can later return to Γ again. Such things effectively destroy all possibility of a good, simple cut-free Gentzen system. But even for these logics, Gentzen methods can be applied in modified form and much of use results; it is just that the return on our investment is smaller than with those logics that can't "look back."

Since Gentzen's time many technical improvements have been introduced. One tends, in practice, to use a sequent calculus "upsidedown," beginning with the desired result and working upward to axioms. Beth [1956, 1959] developed what amount to 5 Gentzen type systems formulated "upsidedown" from the start. The resulting proof procedures are called *tableau systems*. Beth called attention to the key intuition concerning such things: they are *refutation systems*. A tableau construction is an attempt to refute a given formula; if it fails, the formula intuitively should be valid. This is a fruitful way of thinking about tableau systems.

Beth's ideas were followed up on in Smullyan [1968] for Classical logic. Beth's rather awkward "two-tree" methods were greatly simplified and made more elegant. Shortly after, in Fitting [1969] similar modifications were made to Beth's Intuitionistic tableau system, following the lead of Smullyan. Indeed, the proper way to think of much of the present book is as the extension, to a broad class of logics, of techniques developed in Smullyan [1968] for Classical logic. We have made things self-contained, so it is not necessary to read Smullyan's book first, but it couldn't hurt.

For years, modal logic was hampered in its development by the lack of a good intuitive semantics. Then in the 1960's, what is now called *Kripke semantics* was introduced. Its success in treating the "standard" modal logics made it very popular. Since then its limitations have been investigated too; they are real and significant. Still, for the best-known of the modal logics, it works very well indeed. We take Kripke-style semantics as basic in this book.

Now when Kripke introduced his model theory for modal logics, his completeness proofs were based on semantic tableau constructions of some complexity. Generally, he has not been followed in this. Indeed, in his Journal of Symbolic logic review (Kaplan 6 [1966]) of Kripke's 1963 paper on normal modal logics, the reviewer David Kaplan wrote "*Although the author extracts a great deal of information from his tableau constructions, a completely rigorous development along these lines would be extremely tedious. As a consequence a number of small gaps must be filled by the reader's geometrical intuition, for example in verifying that the construction can be developed so that every line in every tableau will have the appropriate rule applied to it at some point.*" The reviewer refers to the fact that in Kripke's approach some sort of *systematic* tableau procedure is necessary to make sure everything is being done, and it is difficult to completely and accurately describe such a systematic procedure for modal tableaus. Again, later on, he writes, "*The reviewer believes that future research will bring considerably simpler more rigorous proofs which avoid the tableau technique. In fact the interesting half of the main theorem can be established by using the technique of Henkin…*". The reviewer then goes on to outline how the "technique of Henkin", that is, using maximal consistent sets, can help establish the Kripke-style completeness of several *axiomatically* formulated modal logics.

Loosely, the method is as follows. For a given logic L one takes, as possible worlds, all sets of formulas that are maximally consistent relative to L. A suitable Kripke model is created from these and one shows that, if S is one of the worlds in it,

$$X \text{ is a member of } S \qquad \text{if and only if} \qquad X \text{ is true at } S. \tag{$*$}$$

7 Note that the implication goes both ways. We return to this point in a moment. At any rate, from $(*)$, axiomatic completeness follows easily.

The method is indeed simple and elegant, and does not involve any of the elaborate "book-keeping" necessary in a systematic tableau construction following Kripke's approach. Here then is a bit of a dilemma. Many people find tableau methods very pleasant to use when a proof *within* a given logic must be discovered. But tableau completeness proofs are generally messy things. Axiom systems are often difficult to use, and generally give less information about the logic, while completeness proofs for them are quite nice. One feels a natural desire to work in the best of all possible worlds by getting the maximal-consistent-set approach to completeness proofs to work on tableau systems as well as on axiom systems.

Defining a suitable notion of consistency is simple: a set S is consistent, with respect to a given tableau system, if there is no tableau that begins with a finite list of members of S, and goes on to close. That every consistent set, in this sense, can be extended to a *maximal* such set is easy. But then one runs into complications. One can not seem to get that, for a given maximal consistent set S, either a formula X is in S, or its negation is in S; and it is from this that one generally derives knowledge of what *is* in S when supplied with information about what *is not* there. This is a feature of maximal consistent sets that is commonly used in completeness proofs along these lines. We said one could not get this X-or-not-X feature; actually one can when given a tableau cut rule. But adding a cut rule to a tableau system generally destroys its usefulness as a proof-discovery method, while showing cut is eliminable (if it is) is often more baroque 8
than following the details of a systematic tableau construction *a la* Kripke.

Fortunately there is a simple way out of this difficulty: it turns out the difficulty is not a real one. Without cut one does not know the X-or-not-X property of maximal consistent sets. Without that one can not establish ($*$). But one still has enough machinery to show that, on constructing a model from maximal tableau-consistent sets,

$$X \text{ is a member of } S \qquad \text{implies} \qquad X \text{ is true at } S. \qquad\qquad (**)$$

and this is quite enough to get completeness.

In effect, the *maximality* part of the notion of maximal consistency can be weakened, and still get us where we want to go.

Now, if not all the customary features of maximality are really needed to successfully use maximal consistent sets, neither are those of consistency. Following the lead of Smullyan in Classical logic, those features of consistency that are important in proving completeness can be abstracted out to form a notion of *abstract consistency property* (tableau-consistency is, then, just one example of an abstract consistency property). Then the maximal-consistent-set approach can be applied in this abstract setting to give us a proof of the *Model Existence Theorem* for various modal logics. This says: if a set S is consistent with respect to some abstract consistency property, then it is satisfiable in an appropriate Kripke model. The proof of this is little more work than the direct proof of tableau completeness using maximal consistent sets. Indeed, tableau completeness is an easy corollary. But so, generally, are: axiom system completeness; natural deduction system completeness; compactness; deduction theorems; interpolation theorems; and so 9
on. The benefits of abstraction are many indeed.

Our central proof procedure will be tableaus, which should come as no surprise. But we do not follow Kripke's tableau format (which derives from Beth). In Kripke's work, semantic tableau constructions often involved the creation of alternate tableaus, and alternates to these, and so on. It turns out that for many modal logics, one can make do with a single tableau provided one suitably "up-dates" the branches from time to time. This idea makes possible tableau systems that are really quite easy to use, and are the primary proof procedure of the book. Gentzen sequent calculi, as we remarked above, are the ancestors of tableau systems, so it is a simple matter to consider them too. And we introduce two kinds of natural deduction systems, one having close relations with tableau systems, the other with axiom systems. And of course, we consider axiom systems themselves.

Smullyan introduced a system of *uniform notation* into Classical logic, which has been a great convenience. We have extended it to modal logics, and it generally enables us to deal with several related logics at once. Uniform notation has a central position in our approach. Indeed, its features suggested to us the very natural natural deduction systems we use.

Now, the tableau approach of Kripke involved the creation of alternate tableaus, and a systematic tableau construction procedure. All this is avoided in the approach outlined above. But the original ideas have a natural appeal, so we also present a, 10 quite separate, second tableau approach that follows Kripke more closely. But things are greatly simplified by a seemingly trivial device: instead of actually creating the alternate tableaus, we simply create names for them. Then we can use a single tableau, in which we manipulate both formulas and these names. This also leads to some rather efficient proof procedures. And the original tableau completeness proof now works very well, since a systematic tableau construction is simple to describe. Alternately, a maximal-consistent-set approach may be used.

All the techniques just outlined for modal logics are also applied to Intuitionistic logic, yielding easy-to-use proof procedures, simple completeness proofs, and the expected variety of model-theoretic results.

Chapter 1

Background

11

1.1 Propositional Formulas

We define the language common to all the propositional modal logics we will be considering. (Notation varies from book to book, by the way.)

We begin by specifying our *alphabet*.

1) We assume we have the following propositional connectives, modal operators, and logical constants:

$\wedge$	(and)
$\vee$	(or)
$\neg$	(not)
$\supset$	(implies)
$\Box$	(necessary)
$\Diamond$	(possible)
$\bot$	(falsehood)
$\top$	(truth-hood)

2) We suppose we have an infinite supply of *propositional variables*. If the reader chooses to understand this as meaning a *countable list*, we have no objection, though all our results hold for languages of arbitrary cardinality.

We use “P”, “Q”, “R”, ... to stand for propositional variables. Until we get to quantifiers in Chapter 7[][], “variable” will mean propositional variable.

3) We have left and right parentheses, as punctuation symbols.

12 This completes our alphabet.

By an atomic formula we mean any variable, $\bot$ or $\top$.

The notion of *formula* is given by the following four recursive rules:

1) Any atomic formula is a formula.
2) If A is a formula, so is $\neg A$.
3) If A and B are formulas, so are $(A \wedge B)$, $(A \vee B)$, $(A \supset B)$.
4) If A is a formula, so are $\Box A, \Diamond A$.

For this definition of formula, a *unique decomposition theorem* can be shown. It says that for every formula X, exactly one of the following holds:

1) X is atomic,
2) there is a *unique* formula Y and a *unique* symbol S from the list $\neg, \Box, \Diamond$ such that $X = SY$,
3) there are *unique* formulas Y and Z and a *unique* connective B chosen from the list $\wedge, \vee, \supset$ such that $X = (YBZ)$.

We do not prove this here, but we make use of it many times, without explicit mention.

We define the notion of *immediate subformula* as follows:

(1) Atomic formulas have no immediate subformulas.

(2) X is an immediate subformula of $\neg X$, $\Box X$ and $\Diamond X$.

(3) X and Y are immediate sub formulas of $(X \wedge Y)$, $(X \vee Y)$, and $(X \supset Y)$.

The *proper-subformula-of* relation is the transitive closure of the immediate-subformula-of relation. That is, for formulas X and Y, X is a proper subformula of Y if there is a sequence, beginning with X, ending with Y, and in which each term is an immediate subformula of the next. 13

By a *subformula* of X we mean any proper subformula of X, or X itself.

By the *degree* of a formula we mean the number of occurrences of the symbols $\wedge, \vee, \supset, \neg, \Box, \Diamond$ it contains. We will often prove things about formulas by doing an induction on their degrees. Note that the degree of any proper subformula of X must be less than the degree of X. This is the central fact upon which all such induction arguments are based.

Generally, when displaying formulas, we will abbreviate them by leaving off the outermost parentheses. Also we will use square and curly brackets as notational variants of parentheses, to make reading easier.

Concluding Note: As an alternative to the above, we could have defined *formulas* as follows. Let $\mathbb{W}$ be the set of all words (finite strings) over our alphabet. If Ψ is a map from the power set of $\mathbb{W}$ to itself, Ψ is called an operator on $\mathbb{W}$.[1] Now, let Φ be the following particular operator on $\mathbb{W}$:

For a set of words $\mathbb{P}$, $\Phi(\mathbb{P})$ is the set consisting of:

(1) all atomic formulas,

(2) $\neg X, \Box X$ and $\Diamond X$ for all $X \in \mathbb{P}$,

(3) $(X \wedge Y)$, $(X \vee Y)$ and $(X \supset Y)$ for all $X, Y \in \mathbb{P}$.

This operator is *monotone*, that is,

$$\mathbb{P} \subseteq \mathbb{Q} \quad \Longrightarrow \quad \Phi(\mathbb{P}) \subseteq \Phi(\mathbb{Q}).$$

It is a general fact that monotone operators always possess *smallest fixed points.* Well, the set of formulas is the smallest fixed point of the operator Φ constructed above. That is, the set of formulas is that set $\mathbb{F}$ such that 14

$$\Phi(\mathbb{F}) = \mathbb{F} \qquad \text{and} \qquad \Phi(\mathbb{P}) = \mathbb{P} \implies \mathbb{F} \subseteq \mathbb{P}.$$

Of course this amounts to saying the set of formulas is the smallest set $\mathbb{F}$ containing the atomic formulas and closed under the operations

[1] [][] Although I know the symbol Ψ maybe ψ in Fitting's original book, but that symbol is pretty cool!

1) if $X \in \mathbb{F}$ then $\neg X$, $\Box X$ and $\Diamond X$ are in $\mathbb{F}$.

2) if $X, Y \in \mathbb{F}$ then $(X \wedge Y)$, $(X \vee Y)$ and $(X \supset Y)$ are in $\mathbb{F}$.

It is also a general fact that one can prove things about the members of the smallest fixed point of a monotone operator by a kind of generalized induction, as follows. Suppose Ψ is a monotone operator, and suppose that whenever all the members of $\mathbb{R}$ have property P, so do all the members of $\Psi(\mathbb{R})$. Then every member of the smallest fixed point $\mathbb{F}$ of Ψ will have property P. For particular operator Φ defined above, this means we can show every formula has a property P provided we can show

(1) every atomic formula has property P,

(2) if X has property P, so do $\neg X$, $\Box X$, and $\Diamond X$,

(3) if X and Y have property P, so do $(X \wedge Y)$, $(X \vee Y)$, and $(X \supset Y)$.

As a matter of fact, proving things about all formulas using the above technique is essentially the same as doing an induction on degree. The reader is thus free to interpret our proofs in this book in whichever manner seems most congenial.

1.2 Models

1.3 A Unifying Notation

1.4 Classical Semantic Tableaus

29

In Smullyan [1968] a tree proof procedure for Classical logic was extensively developed. It has its origins in Gentzen [1935] and traces its descent through Hintikka [1955], Beth [1959] and Smullyan [1966]. It has the double virtue of being easy to prove things with, and perspicuous to prove things about. In this section we briefly recall the features of the propositional part of it, then in the next chapter we adapt it to certain modal logics.

Chapter 2

Prefixed Tableau Systems