as "I know φ" or "I believe that φ". It is also possible to add on additional modal operators $\Box_i$ and $\Diamond_i$ and provide various interpretations. We prefer to read $\Box$ and $\Diamond$ simply as "box" and "diamond" so as not to prejudge the intended interpretation.

The semantics for a modal language $\mathcal{L}_{\Box,\Diamond}$ is based on a generalization of the structures for classical predicate logic of II.4 known as *Kripke frames*. Intuitively, we consider a collection W of "*possible worlds*". Each world $w \in W$ constitutes a view of reality as represented by a structure $\mathcal{C}(w)$ associated with it. We adopt the notation of forcing from set theory and write $w \Vdash \varphi$ to mean φ is true in the possible world w. (We read $w \Vdash \varphi$ as "w forces φ" or "φ is true at w".) If φ is a sentence of the classical language $\mathcal{L}$, this should be understood as simply asserting that φ is true in the structure $\mathcal{C}(w)$. If $\Box$ is interpreted as necessity, this notion can be understood as truth in all possible worlds; the notion of possibility expressed by $\Diamond$ would then mean truth in some possible world.

Temporal notions, or assertions of the necessity or possibility of some fact φ given some preexisting state of affairs, are expressed by including an accessibility (or successor) relation S between the possible worlds. Thus we write $w \Vdash \Box\varphi$ to mean that φ is true in all possible successor worlds of w or all worlds accessible from w. This is a reasonable interpretation of "φ is necessarily true in world w".

Before formalizing the semantics for modal logic in §2, we give some additional motivation by considering two types of applications to computer science.

The first area of application is to theories of program behavior. Modalities are implicit in the works of Turing [1949, 5.7], Von Neumann [1963, 5.7], Floyd [1967, 5.7], Hoare [1973, 5.7], and Burstall [1972, 5.7] on program correctness. The underlying systems of modal logic were brought to the surface by many later workers. Examples of the logics recently developed for the analysis of programs include algorithmic logic, dynamic logic, process logic, and temporal logic. Here are the primitive modalities of one system, the dynamic logic of sequential programs.

Let α be a sequential (possibly nondeterministic) program, let s be a state of the machine executing α. Let φ be a predicate or property of states. We introduce modal operators $\Box_\alpha$ and $\Diamond_\alpha$ into the description of the execution of the program α with the intended interpretation of $\Box_\alpha\varphi$ being that φ is necessarily or always true after α is executed. The meaning of $\Diamond_\alpha$ is intended to be that φ is sometimes true when α is executed (i.e., there is some execution of α which makes φ true). Thus $\Box_\alpha$ is a modal necessity operator and $\Diamond_\alpha$ is a modal possibility operator.

We can make this language more useful by invoking the ideas of possible worlds as described above. Here the "possible worlds" are the states of the machine and the accessibility relation is determined by the possible execution sequences of the program α. More precisely, we interpret forcing assertions about modal formulas as follows:

$s \Vdash \Box_\alpha\varphi$ asserts that φ is true at any state s' such that there exists a legal execution sequence for α which starts in state s and eventually reaches state s'.

$s \Vdash \Diamond_\alpha\varphi$ asserts that φ is true at (at least) one state s' such that there exists a legal execution sequence for α which starts in state s and eventually reaches state s'.

Thus, the intended accessibility relation, S_α, is that s' is accessible from s, $sS_\alpha s'$, if and only if some execution of program α starting in state s ends in state s'.

We could just as well introduce separate operators $\Box_\alpha$, $\Diamond_\alpha$ for each program α. A modal Kripke semantics could then be developed with distinct accessibility relations S_α for each pair of operators $\Box_\alpha$ and $\Diamond_\alpha$. Such a language is very useful in discussing invariants of programs and, in general, proving their correctness. After all, correctness is simply the assertion that, no matter what the starting state, some situation φ is always true when the execution of α is finished: $\Box_\alpha\varphi$. (See, for example, Goldblatt [1982, 5.6], [1987, 5.6] and Harel [1979, 5.7].)

Many interesting and useful variations on this theme have been proposed. One could, for example, interpret $s \Vdash \Box_\alpha\varphi$ to mean that φ is true at every state s' which can be visited during an execution of α starting at s. In this vein, $s \Vdash \Diamond_\alpha\varphi$ would mean that φ is true at some state s' which is reached during some execution of α starting at s. We have simply changed the accessibility relation and we have what is called process logic. This interpretation is closely related to temporal logic. In temporal logic, $\Box\varphi$ means that φ is always true and $\Diamond\varphi$ means that φ will eventually (or at some time) be true. This logic can be augmented in various ways with other modal operators depending on one's view of time. In a digital sequential machine, it may be reasonable to view time as ordered as are the natural numbers. In this situation, for example, one can introduce a modal operator $\circ$ and read $t \Vdash \circ\,\varphi$ as φ is true at the moment which is the immediate successor of t. Various notions of fairness, for example, can be formulated in these systems (even without $\circ$): every constantly active process will eventually be scheduled (for execution or inspection etc.) — $\Box\varphi(c) \to \Diamond\psi(c)$; every process which is ever active is scheduled at least once — $\varphi(c) \to \Diamond\psi(c)$; every process active infinitely often will be scheduled infinitely often — $\Box\Diamond\varphi(c) \to \Box\Diamond\psi(c)$; etc.. Thus these logics are relevant to analyses of the general behavior and, in particular, the correctness of concurrent or perpetual programs. (Another good reference here is Manna and Waldinger [1985, 5.6].)

A quite different source of applications of modal logic in computer science is in theories of knowledge and belief for AI. Here we may understand $\Box_K\varphi$ as "some (fixed) agent or processor knows φ (i.e., that φ is true)" or $\Box_B\varphi$ as "some (fixed) agent or processor believes φ (i.e., that φ is true)". Again, we may wish to discuss not one processor but many. We can then introduce modal operators such as $\Box_{K,\alpha}\varphi$ to be understood as "processor α knows φ". Thus, for example, $\Box_{K,\alpha}\Box_{K,\beta}\varphi$ says that α knows that β knows φ; $\Box_{K,\alpha}\varphi \to \Box_{K,\beta}\psi$ says that if α knows φ then β knows ψ. (A general reference for AI logics is Turner [1984, 5.6].)

This language clearly allows one to formulate notions about communication and knowledge in distributed or concurrent systems. Another related avenue of investigation considers attempts to axiomatize belief or knowledge in humans as well as machine systems. One can then deduce what other properties of knowledge or belief follow from the axioms. On the basis of such deductions, one may either modify one's epistemological views or change the axioms about knowledge that one is willing to accept. The view of modal logic as a logic of belief or knowledge is particularly relevant to analyses of database management. In this light, it is also closely related to nonmonotonic logic as presented in III.7. (See Halpern and Moses [1985, 5.6] for a survey of logics of knowledge and beliefs and Thayse [1989, 5.6] for a thorough treatment of modal logic aimed at deductive databases and AI applications.)

In the next sections we will give a formal semantics for modal logic (§2) and a tableau style proof system (§3). In §4 we will prove soundness and completeness theorems for our proof system. Many applications of modal logic concern systems in which there are agreed (or suggested) restrictions on the interpretations corresponding to varying views of the properties of necessity, knowledge, time, etc., that one is trying to capture. We devote §5 to the relation between restrictions on the accessibility relation, adding axioms about the modal operators to the underlying logic and adjoining new tableau proof rules. The final section (§6) describes a traditional Hilbert style system for modal logic extending that presented for classical logic in II.8.

2. Frames and Forcing

For technical convenience, we make a couple of modifications to the basic notion of a (modal) language $\mathcal{L}$. First, we omit the connective $\leftrightarrow$ from our formal language and view $\varphi \leftrightarrow \psi$ as an abbreviation for $\varphi \to \psi \wedge \psi \to \varphi$. Second, we assume throughout this chapter that every language $\mathcal{L}$ has at least one constant symbol but no function symbols other than constants. (The elimination of function symbols does not result in a serious loss of expressiveness. We can systematically replace function symbols with relations. The work of a binary function symbol $f(x,y)$, for example, can be taken over by a ternary relation symbol $R_f(x,y,z)$ whose intended interpretation is that $f(x,y) = z$. A formula $\varphi(f(x,y))$ can then be systematically replaced by the formula $\exists z(R_f(x,y,z) \wedge \varphi(z))$.)

We now present the precise notion of a frame used to formalize the semantics of modal logic. As we have explained, a frame consists of a set W of "possible worlds", an accessibility (or successor) relation S between the possible worlds and an assignment of a classical structure $\mathcal{C}(p)$ to each $p \in W$. We have chosen to require that the domains $C(p)$ of the structures $\mathcal{C}(p)$ be monotonic in the successor relation, i.e., if q is a successor world of p, pSq, then $C(p) \subseteq C(q)$. This weak monotonicity requirement is not a serious restriction. As even the atomic predicates are not assumed to be monotonic, i.e., an element c of $C(p)$ can have some property R in $\mathcal{C}(p)$ but not in $\mathcal{C}(q)$. As any object can be declared to no longer be in the domain of a particular database or other predicate. One can provide frame semantics that do not incorporate this restriction but there are many difficulties involved that we wish to avoid. For example, if all objects cease to exist, i.e., some $C(q) = \emptyset$, we have entirely left the realm of classical predicate logic which is formulated only for nonempty domains.

Definition 2.1: Let $\mathcal{C} = (W,\ S,\ \{\mathcal{C}(p)\}_{p\in W})$ consist of a set W, a binary relation S on W and a function which assigns to each p in W a (classical) structure $\mathcal{C}(p)$ for $\mathcal{L}$ (in the sense of Definition II.4.1). To simplify the notation we will write $\mathcal{C} = (W, S, \mathcal{C}(p))$ instead of the more formally precise version, $\mathcal{C} = (W, S, \{\mathcal{C}(p)\}_{p\in W})$. As usual, we let $C(p)$ denote the domain of the structure $\mathcal{C}(p)$. We also let $\mathcal{L}(p)$ denote the extension of $\mathcal{L}$ gotten by adding on a name c_a for each element a of $C(p)$ in the style of the definition of truth in II.4. We write either pSq or $(p,q) \in S$ to denote the fact that the relation S holds between p and q. We also describe this state by saying that q is *accessible* from (or a *successor* of) p. We say that $\mathcal{C}$ is a *frame* for the language $\mathcal{L}$, or simply an *$\mathcal{L}$–frame* if, for every p and q in W, pSq implies that $C(p) \subseteq C(q)$ and the interpretations of the constants in $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ are the same in $\mathcal{C}(p)$ as in $\mathcal{C}(q)$.

We now define the forcing relation for $\mathcal{L}$–frames. While reading the definition and working through the later examples, it may help to keep in mind the following paradigm interpretation: each $p \in W$ is a possible world; pSq means that q is a possible future of p; $p \Vdash \varphi$ means that φ is true in the world p; $\Box\varphi$ means that φ will always be true and $\Diamond\varphi$ means that φ will be true sometime in the future.

Definition 2.2 (*Forcing for frames*): Let $\mathcal{C} = (W,\ S,\ \mathcal{C}(p))$ be a frame for a language $\mathcal{L}$, p be in W and φ be a sentence of the language $\mathcal{L}(p)$. We give a definition of *p forces φ*, written $p \Vdash \varphi$ by induction on sentences φ.

(i) For atomic sentences φ, $p \Vdash \varphi \Leftrightarrow \varphi$ is true in $\mathcal{C}(p)$.

(ii) $p \Vdash (\varphi \to \psi) \Leftrightarrow p \Vdash \varphi$ implies $p \Vdash \psi$.

(iii) $p \Vdash \neg\varphi \Leftrightarrow p$ does not force φ (written $p \nVdash \varphi$).

(iv) $p \Vdash (\forall x)\varphi(x) \Leftrightarrow$ for every constant c in $\mathcal{L}(p)$, $p \Vdash \varphi(c)$.

(v) $p \Vdash (\exists x)\varphi(x) \Leftrightarrow$ there is a constant c in $\mathcal{L}(p)$ such that $p \Vdash \varphi(c)$.

(vi) $p \Vdash (\varphi \wedge \psi) \Leftrightarrow p \Vdash \varphi$ and $p \Vdash \psi$.

(vii) $p \Vdash (\varphi \vee \psi) \Leftrightarrow p \Vdash \varphi$ or $p \Vdash \psi$.

(viii) $p \Vdash \Box\varphi \Leftrightarrow$ for all $q \in W$ such that pSq, $q \Vdash \varphi$.

(ix) $p \Vdash \Diamond\varphi \Leftrightarrow$ there is a $q \in W$ such that pSq and $q \Vdash \varphi$.

If we need to make the frame explicit, we will say that p *forces* φ *in* $\mathcal{C}$ and write $p \Vdash_{\mathcal{C}} \varphi$.

Definition 2.3: Let φ be a sentence of the language $\mathcal{L}$. We say that φ *is forced in the $\mathcal{L}$-frame* $\mathcal{C}$, $\Vdash_{\mathcal{C}} \varphi$, if every p in W forces φ. We say φ is *valid*, $\models \varphi$, if φ is forced in every $\mathcal{L}$-frame $\mathcal{C}$.

Example 2.4: For any sentence φ, the sentence $\Box\varphi \to \neg\Diamond\neg\varphi$ is valid: Consider any frame $\mathcal{C} = (W, S, \mathcal{C}(p))$ and any $p \in W$. We must verify that $p \Vdash \Box\varphi \to \neg\Diamond\neg\varphi$ in accordance with clause (ii) of Definition 2.2. Suppose then that $p \Vdash \Box\varphi$. If $p \nVdash \neg\Diamond\neg\varphi$, then $p \Vdash \Diamond\neg\varphi$ (by (iii)). By clause (x), there is a $q \in W$ such that pSq and $q \Vdash \neg\varphi$. Our assumption that $p \Vdash \Box\varphi$ and clause (ix) then tell us that $p \Vdash \varphi$, contradicting clause (iii). Exercise 1 shows that the converse, $\neg\Diamond\neg\varphi \to \Box\varphi$, is also valid.

Example 2.5: We claim that $\Box\forall x\varphi(x) \to \forall x\Box\varphi(x)$ is valid. If not, there is a frame $\mathcal{C}$ and a p such that $p \Vdash \Box\forall x\varphi(x)$ but $p \nVdash \forall x\Box\varphi(x)$. If $p \nVdash \forall x\Box\varphi(x)$, there is, by clause (iv), a $c \in \mathcal{L}(p)$ such that $p \nVdash \Box\varphi(c)$. There is then, by clause (ix), a $q \in W$ such that pSq and $q \nVdash \varphi(c)$. As $p \Vdash \Box\forall x\varphi(x)$, $q \Vdash \forall x\varphi(x)$ by (ix). Finally, $q \Vdash \varphi(c)$ by (iv) for the desired contradiction. Note that the assumption that the domains $C(p)$ are monotonic, in the sense that $pSq \Rightarrow C(p) \subseteq C(q)$, plays a key role in this argument.

Example 2.6: $\Box\varphi(c) \to \varphi(c)$ is not valid: Consider any frame in which the atomic sentence $\varphi(c)$ is not true in some $\mathcal{C}(p)$ and there is no q such that pSq. In such a frame $p \Vdash \Box\varphi(c)$ but $p \nVdash \varphi(c)$.

Example 2.7: $\forall x\varphi(x) \to \Box\forall x\varphi(x)$ is not valid: Let $\mathcal{C}$ be the frame in which $W = \{p, q\}$, $S = \{(p, q)\}$, $C(p) = \{c\}$, $C(q) = \{c, d\}$, $\mathcal{C}(p) \models \varphi(c)$ and $\mathcal{C}(q) \models \varphi(c) \wedge \neg\varphi(d)$. Now $p \Vdash \forall x\varphi(x)$ but $p \nVdash \Box\forall x\varphi(x)$ as $q \nVdash \varphi(d)$. It is crucial in this example that the domains $C(p)$ of a frame $\mathcal{C}$ are not assumed to all be the same. Modal logic restricted to constant domains is considered in Exercises 4.8.

Note that validity as defined here coincides with that for classical predicate logic for sentences φ with no modal operators (Exercise 10).

Some care must be taken now in the definition of "logical consequence" for modal logic. If one keeps in mind that the basic structure is the entire frame and not the individual worlds within it, one is lead to the following definition:

Definition 2.8: Let Σ be a set of sentences in a modal language $\mathcal{L}$ and φ a single sentence of $\mathcal{L}$. φ *is a logical consequence of* Σ, $\Sigma \models \varphi$, if φ is forced in every $\mathcal{L}$ frame $\mathcal{C}$ in which every $\psi \in \Sigma$ is forced.

Warning: This notion of logical consequence is not the same as requiring that, in every $\mathcal{L}$ frame $\mathcal{C}$, φ is true (forced) at every world w at which every $\psi \in \Sigma$ is forced (Exercise 11). In particular, the deduction theorem (Exercise II.7.6) fails for modal logic as can be seen from Examples 2.7 and 2.9.

Example 2.9: $\forall x\varphi(x) \models \Box\forall x\varphi(x)$: Suppose $\mathcal{C}$ is a frame in which $p \Vdash \forall x\varphi(x)$ for every possible world $p \in W$. If $q \in W$, we claim that $q \Vdash \Box\forall x\varphi(x)$. If not, there would be a $p \in W$ such that qSp and $p \nVdash \forall x\varphi(x)$ contradicting our assumption.

Example 2.10: If φ is an atomic unary predicate, $\Box\varphi(c) \nvDash \Diamond\varphi(c)$: Consider a frame $\mathcal{C}$ in which $S = \emptyset$ and in which $\mathcal{C}(p) \nvDash \varphi(c)$ and so $p \nVdash \varphi(c)$ for every p. In $\mathcal{C}$, every p forces $\Box\varphi(c)$ but none forces $\varphi(c)$ and so none forces $\Diamond\varphi(c)$.

There are other notions of validity (and so of logical consequence) that result from putting further restrictions on the set W of possible worlds or (more frequently) on the accessibility relation S. For example, it is often useful to consider only reflexive and transitive accessibility relations. We will discuss several such alternatives in §5.

One should be aware that although $\Box$ and $\Diamond$ are treated syntactically like propositional connectives, their semantics involves quantification over all possible accessible worlds. $\Box\varphi$ says that, no matter what successor world one might move to, φ will be true there. $\Diamond\varphi$ says that there is some successor world to which one could move and make φ true. The construction of tableaux appropriate to such semantics will involve, of course, the introduction of new worlds and instantiations for elements of old ones.

Exercises

Prove, on the basis of the semantic definition of validity in Definition 2.3, that the following are valid modal sentences.

1. $\neg\Diamond\neg\varphi \to \Box\varphi$ (for any sentence φ).
2. $\forall x\Box\varphi(x) \to \exists x\Box\varphi(x)$ (for any formula $\varphi(x)$ with only x free).

Prove that the following are not, in general, valid modal sentences. Let φ be any modal sentence.

3. $\varphi \to \Diamond\varphi$.
4. $\varphi \to \Box\varphi$.
5. $\Diamond\varphi \to \varphi$.

Verify the following instances of logical consequence for modal sentences φ:

6. $\varphi \models \Box\varphi$.
7. $(\varphi \to \Box\varphi) \models (\Box\varphi \to \Box\Box\varphi)$.

Give frames that demonstrate the following failures of logical consequence:

8. $\Box\varphi \nvDash \varphi$.
9. $(\Box\varphi \to \varphi) \nvDash (\Box\varphi \to \Box\Box\varphi)$.

10. If φ is sentence with no occurrences of $\Box$ or $\Diamond$, prove that validity for φ in the sense of Definition 3.1 coincides with that of II.4.4.
11. We say that φ *is a local consequence of* Σ if, for every $\mathcal{L}$ frame $\mathcal{C} = (W, S, \mathcal{C}(p))$, $\forall p \in W[(\forall \psi \in \Sigma)(p \Vdash \psi) \rightarrow p \Vdash \varphi]$.
    (i) Prove that if φ is a local consequence of Σ then it is a logical consequence of Σ.
    (ii) Prove that the converse of (i) fails, i.e., φ may be a logical consequence of Σ without being a local consequence.

3. Modal Tableaux

We will describe a proof procedure for modal logic based on a tableau style system like that used for classical logic in II.6. In classical logic, the plan guiding tableau proofs is to systematically search for a structure agreeing with the starting signed sentence. We either get such a structure or see that each possible analysis leads to a contradiction. When we begin with a signed sentence $F\varphi$, we thus either find a structure in which φ fails or decide that we have a proof of φ. For modal logic we instead begin with a *signed forcing assertion* $Tp \Vdash \varphi$ or $Fp \Vdash \varphi$ (φ is again a sentence) and try either to build a frame agreeing with the assertion or decide that any such attempt leads to a contradiction. If we begin with $Fp \Vdash \varphi$, we either find a frame in which p does not force φ or decide that we have a modal proof of φ.

The definitions of tableau and tableau proof for modal logic are formally very much like those of II.6 for classical logic. *Modal tableaux* and *tableau proofs* are labeled binary trees. The labels (again called the *entries of the tableau*) are now either *signed forcing assertions* (i.e., labels of the form $Tp \Vdash \varphi$ or $Fq \Vdash \varphi$ for φ a sentence of any given appropriate language) or accessibility assertions pSq. We read $Tp \Vdash \varphi$ as p forces φ and $Fp \Vdash \varphi$ as p does not force φ.

As we are using ordinary predicate logic within each possible world, the atomic tableaux for the propositional connectives $\vee, \wedge, \neg$ and $\rightarrow$ are as in the classical treatment in I.4 or II.6 except that their entries are now signed forcing assertions. The atomic tableaux for the quantifiers $\forall$ and $\exists$ are designed to reflect both the previous concerns in predicate logic as well as our monotonicity assumptions about the domains of possible worlds under the accessibility relation. Thus we still require that only "new" constants be used as witnesses for a true existential sentence or as counterexamples to false universal ones. Roughly speaking, a "new" constant is one for which no previous commitments have been made, e.g., one not in $\mathcal{L}$ or appearing so far in the tableau. Consider, on the other hand, a true universal sentence, $Tp \Vdash \forall x \varphi(x)$. In classical predicate logic we could substitute any constant at all for the universally quantified variable x. Here we can conclude $Tp \Vdash \varphi(c)$ only for constants c which we know to be in $C(p)$ or in $C(q)$ for some world q from which p is accessible, qSp. This idea translates into the requirement that c is in $\mathcal{L}$ or has appeared in a forcing assertion on the path so far which involves p or some q for which qSp has also appeared on the path so far. The point here is that, if qSp and c is in $C(q)$, then by monotonicity it must be in $C(p)$ as well. In the description of modal tableaux, we will refer to these constants as "any appropriate c". Of course, the formal definitions of both "new" and "appropriate" constants are given along with the definition of tableaux.

The other crucial element is the treatment of signed forcing sentences beginning with $\Box$ or $\Diamond$. In classical logic, the elements of the structure built by developing a tableau were the constant symbols appearing on some path of the tableau. We are now attempting to build an entire frame. The p's and q's appearing in the entries of some path P through our tableau will constitute the possible worlds of the frame. We must also specify some appropriate accessibility relation S along each path of the tableau. It is convenient to include this information directly on the path. Thus we allow as entries in the tableau facts of the form pSq for possible worlds p and q that appear in signed forcing assertions on the path up to the entry. Entries of this form will be put on the tableau by some of the atomic tableaux for $\Box$ and $\Diamond$. For example, from $Tp \Vdash \Diamond\varphi$ we can (semantically) conclude that $Tq \Vdash \varphi$ for some q such that pSq. Thus the atomic tableau for $Tp \Vdash \Diamond\varphi$ puts both pSq and $Tq \Vdash \varphi$ on the path for some new q (i.e., one not appearing in the tableau so far). On the other hand, the atomic tableau for $Tp \Vdash \Box\varphi$ will reflect the idea that the meaning of $p \Vdash \Box\varphi$ is that φ is true in every world q such that pSq. It will put on the path the assertion $Tq \Vdash \varphi$ for any appropriate q, i.e., any q for which we already know that pSq by virtue of the fact that pSq has itself appeared on the path so far. In this way, we are attempting to build a suitable frame along every path of the tableau.

We now formally specify the *atomic tableaux*.

Definition 3.1 (*Atomic Tableaux*): We begin by fixing a modal language $\mathcal{L}$ and an expansion to $\mathcal{L}_C$ given by adding new constant symbols c_i for $i \in \mathbf{N}$. We list in Figure 43 the atomic tableaux (for the language $\mathcal{L}$). In the tableaux in the following list, φ and ψ, if unquantified, are any sentences in the language $\mathcal{L}_C$. If quantified, they are formulas in which only x is free.

Warning: In (T$\Box$) and (F$\Diamond$) we allow for the possibility that there is no appropriate q by admitting $Tp \Vdash \Box\varphi$ and $Fp \Vdash \Diamond\varphi$ as instances of (T$\Box$) and (F$\Diamond$) respectively.

The formal definition of tableaux is now quite similar to that for classical logic in II.3.

TAt $Tp \Vdash \varphi$ for any atomic sentence φ and any p		FAt $Fp \Vdash \varphi$ for any atomic sentence φ and any p	
T$\vee$ $Tp \Vdash \varphi \vee \psi$ $Tp \Vdash \varphi \quad Tp \Vdash \psi$	F$\vee$ $Fp \Vdash \varphi \vee \psi$ $Fp \Vdash \varphi$ $Fp \Vdash \psi$	T$\wedge$ $Tp \Vdash \varphi \wedge \psi$ $Tp \Vdash \varphi$ $Tp \Vdash \psi$	F$\wedge$ $Fp \Vdash \varphi \wedge \psi$ $Fp \Vdash \varphi \quad Fp \Vdash \psi$
T$\rightarrow$ $Tp \Vdash \varphi \rightarrow \psi$ $Fp \Vdash \varphi \quad Tp \Vdash \psi$	F$\rightarrow$ $Fp \Vdash \varphi \rightarrow \psi$ $Tp \Vdash \varphi$ $Fp \Vdash \psi$	T$\neg$ $Tp \Vdash \neg\varphi$ $Fp \Vdash \varphi$	F$\neg$ $Fp \Vdash \neg\varphi$ $Tp \Vdash \varphi$
T$\exists$ $Tp \Vdash (\exists x)\varphi(x)$ $Tp \Vdash \varphi(c)$ for some new c	F$\exists$ $Fp \Vdash (\exists x)\varphi(x)$ $Fp \Vdash \varphi(c)$ for any appropriate c	T$\forall$ $Tp \Vdash (\forall x)\varphi(x)$ $Tp \Vdash \varphi(c)$ for any appropriate c	F$\forall$ $Fp \Vdash (\forall x)\varphi(x)$ $Fp \Vdash \varphi(c)$ for some new c
T$\Box$ $Tp \Vdash \Box\varphi$ $Tq \Vdash \varphi$ for any appropriate q	F$\Box$ $Fp \Vdash \Box\varphi$ pSq $Fq \Vdash \varphi$ for some new q	T$\Diamond$ $Tp \Vdash \Diamond\varphi$ pSq $Tq \Vdash \varphi$ for some new q	F$\Diamond$ $Fp \Vdash \Diamond\varphi$ $Fq \Vdash \varphi$ for any appropriate q

Definition 3.2: We continue to use our fixed modal language $\mathcal{L}$ and its extension by constants $\mathcal{L}_C$. We also fix a set $\{p_i : i \in \mathbb{N}\}$ of potential candidates for the p's and q's in our forcing assertions. A *modal tableau* (for $\mathcal{L}$) is a binary tree labeled with signed forcing assertions or accessibility assertions; both sorts of labels are called *entries* of the tableau. The class of modal tableaux (for $\mathcal{L}$) is defined inductively as follows.

(i) Each atomic tableau τ is a tableau. The requirement that c be new in cases (T$\exists$) and (F$\forall$) here simply means that c is one of the constants c_i added on to $\mathcal{L}$ to get $\mathcal{L}_C$ which does not appear in φ. The phrase "any appropriate c" in (F$\exists$) and (T$\forall$) means any constant in $\mathcal{L}$ or in φ. The requirement that q be new in (F$\Box$) and (T$\Diamond$) here means that q is any of the p_i other than p. The phrase "any appropriate q" in (T$\Box$) and (F$\Diamond$) in this case simply means that the tableau is just $Tp \Vdash \Box\varphi$ or $Fp \Vdash \Diamond\varphi$ as there is no appropriate q.

(ii) If τ is a finite tableau, P a path on τ, E an entry of τ occurring on P and τ' is obtained from τ by adjoining an atomic tableau with root entry E to τ at the end of the path P then τ' is also a tableau.

The requirement that c be new in cases (T$\exists$) and (F$\forall$) here means that it is one of the c_i (and so not in $\mathcal{L}$) which do not appear in any entry on τ. The phrase "any appropriate c" in (F$\exists$) and (T$\forall$) here means any c in $\mathcal{L}$ or appearing in an entry on P of the form $Tq \Vdash \psi$ or $Fq \Vdash \psi$ such that qSp also appears on P.

In (F$\Box$) and (T$\Diamond$) the requirement that q be new means that we choose a p_i not appearing in τ as q. The phrase "any appropriate q" in (T$\Box$) and (F$\Diamond$) means we can choose any q such that pSq is an entry on P.

(iii) If $\tau_0, \tau_1, \ldots, \tau_n, \ldots$ is a sequence of finite tableaux such that, for every $n \geq 0$, τ_{n+1} is constructed from τ_n by an application of (ii) then $\tau = \cup\tau_n$ is also a tableau.

As in the previous definitions, we insist that the entry E in clause (ii) formally be repeated when the corresponding atomic tableau is added on to P to guarantee the property corresponding to a classical tableau being finished. The atomic tableaux for which they are actually needed are (F$\exists$), (T$\forall$), (T$\Box$) and (F$\Diamond$). We will, however, generally omit the repetition of the root entry in our examples as a notational convenience. The definition of tableau proofs now follows the familiar pattern.

Definition 3.3 (*Tableau Proofs*)**:** Let τ be a modal tableau and P a path in τ.

(i) P is *contradictory* if, for some forcing assertion $p \Vdash \varphi$, both $Tp \Vdash \varphi$ and $Fp \Vdash \varphi$ appear as entries on P.

(ii) τ is *contradictory* if every path through τ is contradictory.

(iii) τ is a *proof of* φ if τ is a finite contradictory modal tableau with its root node labeled $Fp \Vdash \varphi$ for some p. φ is *provable*, $\vdash \varphi$, if there is a proof of φ.

Note that, as in classical logic, if there is any contradictory tableau with root node $Fp \Vdash \varphi$, then there is one which is finite, i.e., a proof of φ : Just terminate each path when it becomes contradictory. As each path is now finite, the whole tree is finite by König's lemma. Thus, the added requirement that proofs be finite (tableaux) has no affect on the existence of proofs for any sentence. Another point of view is that we could have required that the path P in clause (ii) of the definition of tableaux be noncontradictory without affecting the existence of proofs. Thus, in practice, when attempting to construct proofs we mark any contradictory path with the symbol $\otimes$ and terminate the development of the tableau along that path.

Before dealing with the soundness and completeness of the tableau method for modal logic, we look at some examples of modal tableau proofs. Remember that we are abbreviating the tableaux by generally not repeating the entry which we are expanding. We also number the levels of the tableau on the left and indicate on the right the level of the atomic tableau whose development produced the line.

Example 3.4: There is a natural correspondence between the tableaux of classical predicate logic and those of modal logic beginning with sentences without modal operators. One goes from the modal tableau to the classical one by replacing signed forcing assertions $Tp \Vdash \varphi$ and $Fp \Vdash \varphi$ by the corresponding signed sentences $T\varphi$ and $F\varphi$ respectively. (Formally one must account for the alternate notion of new constant used in II.6.1 when going in the other direction.) Note that this correspondence takes proofs to proofs. (See Exercise 1.)

Example 3.5: $\varphi \to \Box\varphi$, sometimes called the *scheme of necessitation*, is not valid. Figure 44 gives an attempt at a tableau proof.

1	$Fw \Vdash \varphi \to \Box\varphi$	
2	$Tw \Vdash \varphi$	by 1
3	$Fw \Vdash \Box\varphi$	by 1
4	wSv for a new v	by 3
5	$Fv \Vdash \varphi$	by 3

FIGURE 44

This failed attempt at a proof suggests a frame counterexample $\mathcal{C}$ with $W = \{w, v\}$, $S = \{(w, v)\}$ and structures such that φ is true at w but not at v. Such a frame demonstrates that $\varphi \to \Box\varphi$ is not valid as in this frame, w does not force $\varphi \to \Box\varphi$.

Example 3.6: Similarly $\Box\varphi \to \varphi$ is not valid as can be seen from the attempted proof in Figure 45.

1	$Fw \Vdash \Box\varphi \to \varphi$	
2	$Tw \Vdash \Box\varphi$	by 1
3	$Fw \Vdash \varphi$	by 1

FIGURE 45

The frame counterexample suggested here consists of a one world $W = \{w\}$ with empty accessibility relation S and φ false at w. It shows that $\Box\varphi \to \varphi$ is not valid.

Various interpretations of $\Box$ might tempt one to think that $\Box\varphi \to \varphi$ should be valid. For example, probably all philosophers would agree that if φ is necessarily true it in fact is true. On the other hand, most but perhaps not all epistemologists would argue that if I know φ it must also be true. Finally, few people would claim that (for any φ) if I believe φ then φ is true. $\Box\varphi \to \varphi$ is traditionally called "T" or the "knowledge axiom". Under many interpretations of $\Box$, it should be valid. A glance at the attempted proof above shows us that, if we knew that wSw, we could quickly get the desired contradiction. Thus, there is a relation between T and the assumption that the accessibility relation is reflexive. In fact, not only is T valid in all frames with reflexive accessibility relations, but conversely any sentence valid in all such frames can be deduced from T. We will make this correspondence and others like it precise in §5.

Example 3.7: We show in Figure 46 that $\Box(\forall x)\varphi(x) \to (\forall x)\Box\varphi(x)$ is provable.

Note the use of monotonicity in the derivation of lines 6 and 8 corresponding to the semantic argument in Example 2.5.

Example 3.8: Figure 47 gives a tableau proof of

$$\Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi).$$

(This scheme plays an important role in the Hilbert–style systems of modal logic presented in §6.)

Example 3.9: Figure 48 gives an incorrect proof of $\forall x\Box\varphi(x) \to \Box\forall x\varphi(x)$.

1	$Fw \Vdash \Box(\forall x)\varphi(x) \to (\forall x)\Box\varphi(x)$	
2	$Tw \Vdash \Box(\forall x)\varphi(x)$	by 1
3	$Fw \Vdash (\forall x)\Box\varphi(x)$	by 1
4	$Fw \Vdash \Box\varphi(c)$	by 3
5	wSv	by 4
6	$Fv \Vdash \varphi(c)$	by 4
7	$Tv \Vdash (\forall x)\varphi(x)$	by 2, 5
8	$Tv \Vdash \varphi(c)$	by 7
	$\otimes$	by 6, 8

FIGURE 46

The false step occurs at line 7. On the basis of line 6 we can use c for instantiations in forcing assertions about v or any world accessible from v but we have no basis to use it in assertions about w. As in Example 2.7, such a move would be appropriate for an analysis of constant domain frames. (See Exercises 4.8)

Example 3.10: $(\forall x)\neg\Box\varphi \to \neg\Box(\exists x)\varphi$ is not valid. (See Figure 49.)

This is not a proof. With the constant domain $C = \{c, d\}$, and two worlds w, v, with v accessible from w, no atomic sentences true at w and the sentence $\varphi(d)$ true at v, we get a frame counterexample.

As with the semantic definition of logical consequence, one must take care in defining the notion of a modal tableau proof from a set Σ of sentences (which we often call premises). We must match the intuition that we are restricting our attention to frames in which the premises are forced. To do this, we allow the insertion in the tableau of entries of the form $Tp \Vdash \varphi$ for any appropriate possible world p and any $\varphi \in \Sigma$.

1		$Fw \Vdash \Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)$		
2		$Tw \Vdash \Box(\varphi \to \psi)$		by 1
3		$Fw \Vdash \Box\varphi \to \Box\psi$		by 1
4		$Tw \Vdash \Box\varphi$		by 3
5		$Fw \Vdash \Box\psi$		by 3
6		wSv new v		by 5
7		$Fv \Vdash \psi$		by 5
8		$Tv \Vdash \varphi$		by 4, 6
9		$Tv \Vdash \varphi \to \psi$		by 2, 6
10	$Fv \Vdash \varphi$		$Tv \Vdash \psi$	by 9
11	$\otimes$		$\otimes$	by 8, 7

FIGURE 47

Definition 3.11: The definition of *modal tableaux from* Σ, a set of sentences of a modal language called premises, is the same as for simple modal tableaux in Definition 3.2 except that we allow one additional formation rule:

(ii′) If τ is a finite tableau from Σ, $\varphi \in \Sigma$, P a path in τ and p a possible world appearing in some signed forcing assertion on P, then appending $Tp \Vdash \varphi$ to the end of P produces a tableau τ' from Σ.

1	$Fw \Vdash (\forall x)\Box\varphi(x) \to \Box(\forall x)\varphi(x)$		
2	$Tw \Vdash (\forall x)\Box\varphi(x)$		by 1
3	$Fw \Vdash \Box(\forall x)\varphi(x)$		by 1
4	wSv		by 3
5	$Fv \Vdash (\forall x)\varphi(x)$		by 3
6	$Fv \Vdash \varphi(c)$	new c	by 5
7	$Tw \Vdash \Box\varphi(c)$		by 2
8	$Tv \Vdash \varphi(c)$		by 7
	$\otimes$		

FIGURE 48

The notions used to define a tableau proof are now carried over from Definition 3.3 to *tableau proofs from* Σ by simply replacing "tableau" by "tableau from Σ". We write $\Sigma \vdash \varphi$ to denote that φ *is provable from* Σ, i.e., there is a proof of φ from Σ.

Example 3.12: Figure 50 gives a tableau proof of $\Box\forall x\varphi(x)$ from the premise $\forall x\varphi(x)$

Exercises

1. Make precise the correspondence described in Example 3.4 and show that it takes tableau proofs in classical predicate logic to ones in modal logic. Conversely, if τ is a modal tableau proof of a sentence φ of classical logic, describe the appropriate transformation in the other direction and show that it takes τ to a classical proof of φ.

In exercises 2–8, let φ and ψ be any formulas with either no free variables or only x free as appropriate. Give modal tableau proofs of each one.

1	$Fw \Vdash (\forall x)\neg\Box\varphi \to \neg\Box(\exists x)\varphi$		
2	$Tw \Vdash (\forall x)\neg\Box\varphi$		by 1
3	$Fw \Vdash \neg\Box(\exists x)\varphi$		by 1
4	$Tw \Vdash \Box(\exists x)\varphi$		by 3
5	$Tw \Vdash \neg\Box\varphi(c)$		by 2
6	$Fw \Vdash \Box\varphi(c)$		by 5
7	wSv		by 6
8	$Fv \Vdash \varphi(c)$		by 6
9	$Tv \Vdash (\exists x)\varphi$		by 4, 7
10	$Tv \Vdash \varphi(d)$	new d	by 9

FIGURE 49

2. $\neg\Diamond\neg\varphi \to \Box\varphi$.
3. $\forall x\Box\varphi(x) \to \exists x\Box\varphi(x)$.
4. $\Box\Box((\varphi \vee (\forall x)\psi(x)) \to (\forall x)(\varphi \vee \psi(x)))$, x not free in φ.
5. $\neg\Diamond(\neg(\varphi \wedge (\exists x)\psi(x)) \wedge (\exists x)(\varphi \wedge \psi(x)))$, x not free in φ.
6. $\Box(\varphi \vee \neg\psi) \to (\Diamond\psi \to \Diamond\varphi)$.
7. $\Box(\exists x)(\varphi \wedge \psi(x)) \to \Box(\varphi \wedge (\exists x)\psi(x))$, x not free in φ.
8. $\Diamond(\exists x)(\varphi(x) \to \Box\psi) \to \Diamond((\forall x)\varphi(x) \to \Box\psi)$, x not free in ψ.

Give modal tableau proofs of the following:

9. $\varphi \Vdash \Box\varphi$.
10. $(\varphi \to \Box\varphi) \Vdash \Box\varphi \to \Box\Box\varphi$.
11. $\forall x\varphi(x) \Vdash \forall x\Box\varphi(x)$.

1	$Fp \Vdash \Box(\forall x)\varphi(x)$		
2	pSq		by 1
3	$Fq \Vdash (\forall x)\varphi(x)$		by 1
4	$Fq \Vdash \varphi(c)$	new c	by 3
5	$Tq \Vdash (\forall x)\varphi(x)$		premise
6	$Tq \Vdash \varphi(c)$		by 5
	$\otimes$		

FIGURE 50

12. **Theorem on Constants**: Let $\varphi(x_1, ..., x_n)$ be a formula in a modal language $\mathcal{L}$ with all free variables displayed and let $c_1, ..., c_n$ be constant symbols not in $\mathcal{L}$. Show that $\forall x_1 ... \forall x_n \varphi(x_1, ..., x_n)$ is tableau provable iff $\varphi(c_1, ..., c_n)$ is. Argue syntactically to show that given a proof of one of the formulas one can construct a proof of the other.

4. Soundness and Completeness

Our first goal in this section is to show that in modal logic (as in classical logic) provability implies validity. As in the classical soundness theorem (II.7.2), we begin by proving that a frame which "agrees" with the root node of a tableau "agrees" with every entry along some path of the tableau. In the classical case (Definition II.7.1), we construct the path in the tableau and define a structure whose domain consists of the constants c occurring in the signed sentences along this path. In the modal case, we must define a set W of possible worlds and, for each $p \in W$, a structure based on constants occurring on the path. W will consist of the p's occurring in signed forcing assertions along the path. The accessibility relation on W will then be defined by the assertions pSq occurring on the path.

Definition 4.1: Suppose $\mathcal{C} = (V, T, \mathcal{C}(p))$ is a frame for a modal language $\mathcal{L}$, τ is a tableau whose root is labeled with a forcing assertion about a sentence φ of $\mathcal{L}$ and P is a path through τ. Let W be the set of p's appearing in forcing assertions on P and let S be the accessibility relation on W determined by the assertions pSq occurring on P. We say that $\mathcal{C}$ *agrees with* P if there are maps f and g such that

(i) f is a map from W into V which preserves the accessibility relation, i.e., $pSq \Rightarrow f(p)Tf(q)$. (Note that f is not assumed to be one–one.)

(ii) g sends each constant c occurring in any sentence ψ of a forcing assertion $Tp \Vdash \psi$ or $Fp \Vdash \psi$ on P to a constant in $\mathcal{L}(f(p))$. Moreover, g is the identity on constants of $\mathcal{L}$. We also extend g to be a map on formulas in the obvious way: To get $g(\psi)$ simply replace every constant c in ψ by $g(c)$.

(iii) If $Tp \Vdash \psi$ is on P, then $f(p)$ forces $g(\psi)$ in $\mathcal{C}$ and if $Fp \Vdash \psi$ is on P, then $f(p)$ does not force $g(\psi)$ in $\mathcal{C}$.

Theorem 4.2: *Suppose $\mathcal{C} = (V, T, \mathcal{C}(p))$ is a frame for a language $\mathcal{L}$ and τ is a tableau whose root is labeled with a forcing assertion about a sentence φ of $\mathcal{L}$. If $q \in V$ and either*

(i) *$Fr \Vdash \varphi$ is the root of τ and q does not force φ in $\mathcal{C}$*

or

(ii) *$Tr \Vdash \varphi$ is the root of τ and q does force φ in $\mathcal{C}$,*

then there is a path P through τ which agrees with $\mathcal{C}$ with a witness function f (as required in Definition 4.1) that sends r to q.

The proof of this theorem proceeds by induction on the construction of the tableau τ. Before providing the details, we reformulate the result as the standard version of the soundness theorem.

Theorem 4.3 (Soundness, $\vdash \varphi \Rightarrow \models \varphi$): *If there is a (modal) tableau proof of a sentence φ (of modal logic), then φ is (modally) valid.*

Proof (of Soundness): A modal tableau proof of φ is a tableau τ with a root of the form $Fr \Vdash \varphi$ in which every path is contradictory. If φ is not valid, then there is a frame $\mathcal{C} = (V, T, \mathcal{C}(p))$ and a $q \in V$ such that q does not force φ in $\mathcal{C}$. Now apply Theorem 4.2 to get a path P through τ and functions f and g with the properties listed in Definition 4.1. As τ is contradictory, there is a p and a sentence ψ such that both $Tp \Vdash \psi$ and $Fp \Vdash \psi$ occur on P. Definition 4.1(iii) then provides an immediate contradiction. $\Box$

We break the inductive proof of Theorem 4.2 into its component parts. First, there are eighteen base cases corresponding to clause (i) of Definition 3.2 and the eighteen atomic tableaux.

Lemma 4.4: *For each atomic tableau τ which satisfies the hypotheses of Theorem 4.2, there are P, f and g as required in its conclusion.*

There are then sixteen induction cases corresponding to the type of atomic tableau chosen for development in clause (ii) of Definition 3.2. We in fact prove an assertion somewhat stronger than the theorem to facilitate the induction.

Lemma 4.5: *If f and g are witnesses that a path P of a tableau τ agrees with $\mathcal{C}$ and τ' is gotten from τ by an application of clause (ii) of Definition 3.2, then there are extensions P', f' and g' of P, f and g respectively such that f' and g' are witnesses that the path P' through τ' also agrees with $\mathcal{C}$.*

Theorem 4.2 is an easy consequence of these two lemmas so we present its proof before considering the proofs of the lemmas.

Proof (of Theorem 4.2): Lemma 4.4 establishes the theorem for atomic tableau. Lemma 4.5 then proves the theorem for all finite tableaux by induction. In fact, it proves it for infinite tableaux as well: Suppose $\tau = \cup\tau_n$ is an infinite tableau as defined by clause (iii) of Definition 3.2. We begin by applying the appropriate case of Lemma 4.4 to τ_0 to get suitable P_0, f_0 and g_0. We then apply Lemma 4.5 to each τ_n in turn to construct P_n, f_n and g_n. The required P, f and g for τ are then simply the unions of the P_n, f_n and g_n respectively. $\square$

Proof (of Lemma 4.4): We begin by defining $f(p) = r$ and g to be the identity on the constants of $\mathcal{L}_C$. With this choice of f and g, the root itself agrees with $\mathcal{C}$ by the hypothesis of the theorem. This completes the argument for TAt and FAt. The argument needed for each of the other atomic tableaux is precisely the same as the one for the corresponding case of Lemma 4.5. The inductive argument applied to the rest of the atomic tableau then provides the required extensions. (Perhaps technically the degenerate cases of (T$\Box$) and (F$\Diamond$) are exceptions but in those cases the conclusion is precisely the hypothesis.) Thus we have reduced the proof of this lemma to that of Lemma 4.5. $\square$

Proof (of Lemma 4.5): First note that if τ' is gotten by extending τ somewhere other than at the end of P, then the witnesses for τ work for τ' as well. Thus, we may assume that we form τ' by adjoining one of the atomic tableaux at the end of P in τ. We now consider the sixteen cases given by the atomic tableaux of Definition 3.1.

The cases other than (T$\exists$), (F$\forall$), (F$\Box$) and (T$\Diamond$) require no extension of f or g. In each of these cases it is obvious from the induction hypothesis and the corresponding case of the definition of forcing (Definition 2.2) (and the monotonicity assumption on the domains $C(p)$ for cases (F$\exists$) and (T$\forall$)) that one of the extensions of P to a path through τ' satisfies the requirements of the lemma.

We present cases (T$\exists$) and (T$\Diamond$) in detail. Cases (F$\forall$) and (F$\Box$) are similar and are left as Exercise 1. In case (T$\exists$) the entry of P being developed is $Tp \Vdash (\exists x)\varphi(x)$. P', the required extension of P, is the only one possible. It is determined by adding $Tp \Vdash \varphi(c)$ to the end of P. By our induction hypothesis, $f(p) \Vdash_{\mathcal{C}} g((\exists x)\varphi(x))$. By the definition of forcing an existential sentence (2.2(v)), there is a $c' \in \mathcal{L}(p)$ such that $f(p) \Vdash g(\varphi(c'))$. Fix such a c' and extend g to g' by setting $g'(c) = c'$. It is now obvious that P', $f' = f$ and g' satisfy the requirements of the lemma, i.e., f' and g' witness that P' agrees with $\mathcal{C}$.

Finally, in case (T$\Diamond$) the entry of P being developed is $Tp \Vdash \Diamond\varphi$. The required extension of P to P' is the only possible one. It is determined by adding both pSq and $Tq \Vdash \varphi$ onto the end of P. By our induction hypothesis, $f(p) \Vdash_{\mathcal{C}} g(\Diamond\varphi)$. As $g(\Diamond\varphi) = \Diamond g(\varphi)$, there is, by the definition of forcing for $\Diamond$ (2.2(x)), a $q' \in V$ such that $f(p)Tq'$ and $q' \Vdash g(\varphi)$. Fix such a q' and extend f to f' by setting $f'(q) = q'$. It is now obvious that P', f' and $g' = g$ satisfy the requirements of the lemma, i.e., f' and g' witness that P' agrees with $\mathcal{C}$. $\square$

Our next goal is to prove that the tableau method of proof is complete for modal logic. We will define a procedure (like that of II.6.9) for constructing the appropriate complete systematic tableau starting with a given signed forcing assertion at its root. We will then prove that, for any noncontradictory path P through this tableau, we can build a frame $\mathcal{C}$ which agrees with P. Thus if our systematic procedure applied to any forcing assertion $Fp \Vdash \varphi$ fails to produce a modal tableau proof of φ, then we will have built a frame in which φ is not forced and so demonstrated that φ is not valid.

We begin by extending the notion of a reduced entry and a finished tableau (Definition II.6.7) to modal logic.

Recall that $c_1, \ldots, c_n, \ldots$ is a list of all the constants of our expanded language $\mathcal{L}_C$ and p_1, p_2, ... a list of our stock of possible worlds. For convenience we assume that c_1 is in $\mathcal{L}$.

Definition 4.6: Let $\tau = \cup\tau_n$ be a tableau, P a path in τ, E an entry on P and w the i^{th} occurrence of E on P (i.e., the i^{th} node on P labeled with E).

(i) w is *reduced on P* if one of the following four situations hold:

(1) E is not of the form of the root of an atomic tableau of type (F$\exists$), (T$\forall$), (T$\Box$) or (F$\Diamond$) (of Definition 3.1) and, for some j, τ_{j+1} is gotten from τ_j by an application of rule (ii) of Definition 3.2 to E and a path on τ_j which is an initial segment of P. [In this case we say that E occurs on P as the root entry of an atomic tableau.]

(2) E is of the form $Fp \Vdash (\exists x)\varphi(x)$ or $Tp \Vdash (\forall x)\varphi(x)$ (cases (F$\exists$) and (T$\forall$) respectively), there is an $i + 1^{\text{st}}$ occurrence of E on P and either

c_i does not occur in any assertion on P about a possible world q such that qSp occurs on P or

$Fp \Vdash \varphi(c_i)$ or $Tp \Vdash \varphi(c_i)$ is an entry on P.

(3) E is of the form $Tp \Vdash \Box\varphi$ or $Fp \Vdash \Diamond\varphi$ (cases (T$\Box$) and (F$\Diamond$) respectively), there is an $i+1^{\text{st}}$ occurrence of E on P and either

pSp_i is not an entry on P or

$Tp_i \Vdash \varphi$ or $Fp_i \Vdash \varphi$ is an entry on P

(4) E is of the form pSq.

(ii) τ is *finished* if every occurrence of every entry on τ is reduced on every noncontradictory path containing it. It is *unfinished* otherwise.

As in the treatment of classical tableaux, a signed forcing assertion of the form $Tp \Vdash (\forall x)\varphi(x)$ must be instantiated for each constant c_i in our language before we can say that we have finished with it. Here, in addition, if p forces $\Box\varphi$ then φ must be forced by every successor q of p. We can now show that there is a finished tableau with any given signed forcing assertion as its root by constructing the appropriate complete systematic tableau. We use the same scheme as in the classical case based on $\leq_{LL}$, the level–lexicographic ordering, introduced in Definition II.6.8.

Lemma 4.7: *Suppose w is the i^{th} occurrence of an entry E on a path P of a tableau τ and is reduced on P in τ. If τ' is an extension of τ and P' is an extension of P to a path in τ', the only way w could fail to be reduced on P' in τ' is if*

(i) *E is of the form $Fp \Vdash \exists x\varphi(x)$ ($Tp \Vdash \forall x\varphi(x)$) and c_i does not occur in any assertion on P about a possible world q such that qSp occurs on P but c_i does occur in such an assertion on P' and $Fp \Vdash \varphi(c_i)$ ($Tp \Vdash \varphi(c_i)$) is not an entry on P'; or*

(ii) *E is of the form $Tp \Vdash \Box\varphi$ ($Fp \Vdash \Diamond p$) and pSp_i occurs on P' but not on P and $Fp_i \Vdash \varphi$ ($Tp_i \Vdash \varphi$) does not occur on P'.*

Proof: This claim is obvious from the definitions. $\Box$

Definition 4.8: We define the *complete systematic modal tableau* (the CSMT) *starting with a sentence φ* by induction as follows.

(i) τ_0 is the atomic tableau with root $Fp_1 \Vdash \varphi$. This atomic tableau is uniquely specified by requiring that in cases (F$\exists$) and (T$\forall$) we use the constant c_1, in cases (T$\exists$) and (F$\forall$) we use c_i for the least allowable i and in cases (F$\Box$) and (T$\Diamond$) we use the least p_i not occurring in the root. (Note that in cases (T$\Box$) and (F$\Diamond$) the tableau consists of just the root entry. It is finished and constitutes our CSMT.)

At stage n we have, by induction, a tableau τ_n. If τ_n is finished, we terminate the construction. Otherwise, we let w be the level–lexicographically

least node of τ_n which contains an occurrence of an entry E which is unreduced on some noncontradictory path P of τ_n. We now extend τ_n to a tableau τ_{n+1} by applying one of the following three procedures:

(ii) If E is not of the form occurring in the root node of case (F$\exists$), (T$\forall$), (T$\Box$) or (F$\Diamond$), we adjoin the atomic tableau with root E to the end of every noncontradictory path in τ that contains w. For E of type (T$\exists$) or (F$\forall$) we use the least constant c_j not yet appearing in the tableau. If E is of type (F$\Box$) or (T$\Diamond$), we choose p_j for q where j is least such that p_j does not appear in the tableau.

(iii) If E is of type (F$\exists$) or (T$\forall$) and w is the i^{th} occurrence of E on P, we adjoin the corresponding atomic tableau with c_j as the required c, where j is least such that c_j is appropriate and $Fp \Vdash \varphi(c_j)$ or $Tp \Vdash \varphi(c_j)$, respectively, does not appear as an entry on P. We take c to be c_1 if there is no such c_j.

(iv) If E is of type (T$\Box$) or (F$\Diamond$) and w is the i^{th} occurrence of E on P, we adjoin the corresponding atomic tableau with q_j as the required q where j is least such that q_j is appropriate and $Tq_j \Vdash \varphi$ or $Fq_j \Vdash \varphi$, respectively, does not appear as an entry on P. If $Tq_j \Vdash \varphi$ (or $Fq_j \Vdash \varphi$) already appears on P for every appropriate q_j, we simply repeat the assertion $Tq_j \Vdash \varphi$ (or $Fq_j \Vdash \varphi$), where j is least such that q_j is appropriate. (There is at least one appropriate q_j by the assumption that E is not reduced on P.)

The union τ of the sequence of tableaux τ_n is the CSMT starting with φ.

Note that in general a CSMT will be an infinite tableau (even if S is finite). The crucial point is that it is always a finished tableau.

Lemma 4.9: *If p_iSp_j appears as an entry on a* CSMT *then $i < j$.*

Proof: New possible worlds p_j and new instances of the accessibility relation appear on a CSMT $\tau = \cup\tau_n$ only when an entry of the form p_iSp_j is put on the tableau by an application of clause (ii) of Definition 3.8 for cases (F$\Box$) or (T$\Diamond$). Thus they are put on the CSMT in numerical order and so when p_iSp_j is put on the tree, $i < j$ as required. $\Box$

Proposition 4.10: *Every* CSMT *is finished.*

Proof: Consider any entry E and any unreduced occurrence w of E on a noncontradictory path P of the given CSMT τ. (If there is no such w, τ is finished by definition.) Suppose that E makes a forcing assertion about some possible world p_m, w is the i^{th} occurrence of E on P and that there are n nodes of T which are level–lexicographically less than w. Let k be large enough so that

(i) The occurrence w of E is in τ_k.

(ii) p_mSp_i is on P in τ_k if it is on P at all.

(iii) If any assertion about a possible world p_j (for which p_jSp_m occurs on P) and the constant c_i appear on P, then p_jSp_m and some occurrence of an assertion involving both p_j and c_i occurs on P in τ_k.

Note that by Lemma 4.9 there are only finitely many p_j which are relevant to (iii) and so we can find a k large enough to accommodate them all.

It is clear from the definition of the CSMT that we must reduce w on P by the time we form τ_{k+n+1}. Moreover, once reduced in this way, w remains reduced forever by Lemma 4.7. Thus every occurrence of each entry on a noncontradictory path in τ is reduced, as required. □

We can now prove a completeness theorem by showing that the CSMT beginning with $Fp_1 \Vdash \varphi$ is either a proof or supplies us with a frame counterexample.

Theorem 4.11: *Suppose that $\tau = \cup\tau_n$ is a* CSMT *and P is a noncontradictory path in τ. We define a frame $\mathcal{C} = (W, S, \mathcal{C}(p))$ associated with P as follows:*

W is the set of all p_i appearing in forcing assertions on P. S is the set of all pairs (p_i, p_j) such that p_iSp_j appears on P.

For each $p_i \in S$, $\mathcal{C}(p_i)$ is defined by induction on i as the set consisting of all the constants of $\mathcal{L}$ and all other constants appearing in forcing assertions $Tq \Vdash \psi$ or $Fq \Vdash \psi$ on P such that qSp. (Note that by Lemma 4.10, if p_jSp_i appears on P then $j < i$. Thus $\mathcal{C}(p_i)$ is well defined by induction.)

For each $p \in W$, $\mathcal{C}(p)$ is defined by setting each atomic sentence ψ true in $\mathcal{C}(p)$ if and only if $Tp \Vdash \psi$ occurs on P. (Warning: We are using the convention that every $c \in \mathcal{C}(p)$ is named by itself in $\mathcal{L}(p)$.)

If we let f and g be the identity functions on W and on the set of constants appearing on P, respectively, then they are witnesses that $\mathcal{C}$ agrees with P.

Proof: First note that the clauses of the definition of $\mathcal{C}$ are designed to guarantee that $\mathcal{C}$ is a frame for $\mathcal{L}$ according to Definition 2.1. Just remember that every constant c in $\mathcal{L}(p)$ names itself.

We now wish to prove that (f and g witness that) P agrees with $\mathcal{C}$. We use induction on the depth of sentences φ appearing in forcing assertions on P. The key point in the induction is that, by Proposition 4.9, every occurrence of every entry is reduced on P.

(i) Atomic φ : If $Tp \Vdash \varphi$ appears on P then φ is true in $\mathcal{C}(p)$, and so forced by p. If $Fp \Vdash \varphi$ appears on P then $Tp \Vdash \varphi$ does not appear on P as P is noncontradictory. As this is the only way that p could come to force φ in $\mathcal{C}$, we can conclude that p does not force φ, as required.

The inductive cases are each handled by the corresponding clauses of Definition 4.6 and the definition of forcing (Definition 2.2) together with the induction hypothesis for the theorem. We consider some representative cases and leave the others as exercises.

(ii) The propositional connectives: Suppose φ is built using a connective, e.g., φ is $(\varphi_1 \vee \varphi_2)$. As τ is finished, we know that if $Tp \Vdash \varphi$ occurs on P, then either $Tp \Vdash \varphi_1$ or $Tp \Vdash \varphi_2$ occurs on P. By the induction hypothesis if, say, $Tp \Vdash \varphi_1$ occurs on P then p forces φ_1 and so, by the definition of forcing (Definition 2.2(vii)), p forces φ, as required. Similarly, if $Fp \Vdash \varphi$ occurs on P, then both $Fp \Vdash \varphi_1$ and $Fp \Vdash \varphi_2$ appear on P. Thus, by induction and Definition 2.2(vii), p does not force φ, as required. The other classical propositional connectives are treated similarly. (See Exercise 2.)

(iii) Quantifiers: Suppose φ is of the form $(\forall v)\psi(v)$. If w is the i^{th} occurrence of $Tp \Vdash (\forall v)\psi(v)$ on P, then there is an $i + 1^{\text{st}}$ occurrence of $Tp \Vdash (\forall v)\psi(v)$ on P. Moreover, if $c_i \in C(p)$ then $Tp \Vdash \psi(c_i)$ occurs on P. Thus, if $Tp \Vdash (\forall v)\psi(v)$ appears on P, then $Tp \Vdash \psi(c)$ appears on P for every constant $c \in C(p)$. As the depth of $\psi(c)$ is less than that of $(\forall v)\psi(v)$, p forces $\psi(c)$ for every $c \in C(p)$. Thus p forces $\forall v\psi(v)$ by Definition 2.2(iv).

If $Fp \Vdash (\forall v)\psi(v)$ occurs on P, then again as τ is finished, $Fp \Vdash \psi(c)$ occurs on P for some c. By induction hypothesis, p does not force $\psi(c)$. Thus, by Definition 2.2(iv), p does not force $\forall v\psi(v)$ as required.

The analysis for the existential quantifier is similar and is left as Exercise 3.

(iv) The modal operators: If $Tp \Vdash \Box\varphi$ and pSq appear on P, then $Tq \Vdash \varphi$ appears on P as the tableau is finished. (Note that being finished guarantees that if there is one occurrence of $Tp \Vdash \Box\varphi$, there are infinitely many and so in particular a j^{th} one where $q = p_j$.) Thus q forces φ by induction and p forces $\Box\varphi$ by Definition 2.2(ix).

If $Fp \Vdash \Box\varphi$ appears on P, then both pSq and $Fq \Vdash \varphi$ appear on P for some q. Thus q does not force φ by induction and q is a successor of p by definition. So by Definition 2.2(ix), p does not force $\Box\varphi$.

The cases for $\Diamond$ are similar and are left as Exercise 4. □

We can now state the standard form of the completeness theorem.

Theorem 4.12 (Completeness, $\models \varphi \Rightarrow\ \vdash \varphi$): *If a sentence φ of modal logic is valid (in the frame semantics) then it has a (modal) tableau proof.*

Proof: Suppose φ is valid. Consider the CSMT τ starting with root $Fp_1 \Vdash \varphi$. By definition, every contradictory path of τ is finite. Thus if every path of τ is contradictory, τ is finite by König's Lemma. In particular, if τ is not a tableau proof of φ, it has a noncontradictory path P. Theorem 4.9 then provides a frame $\mathcal{C}$ in which p_1 does not force φ. Thus φ is not valid and we have the desired contradiction. □

It is now routine to extend the soundness and completeness theorems to the modal notions of logical consequence and tableaux deductions from premises.

Theorem 4.13 (Soundness, $\Sigma \vdash \varphi \Rightarrow \Sigma \vDash \varphi$): *If there is a modal tableau proof of φ from a set Σ of sentences, then φ is a logical consequence of Σ.*

Proof: The proof of the basic ingredient (Theorem 4.2) of the soundness theorem (Theorem 4.3) shows that if τ is a tableau from Σ and $\mathcal{C}$ is a frame which forces every $\psi \in \Sigma$ which agrees with the root of τ, then $\mathcal{C}$ agrees with some path P of τ. The only new point is that a tableaux can be extended by adding, for any $\psi \in \Sigma$, the assertion $Tp \Vdash \psi$ to the end of any path mentioning p. The proof of Lemma 4.5 is easily modified to incorporate this difference. As $q \Vdash_{\mathcal{C}} \psi$ for every possible world q of $\mathcal{C}$ (by assumption), the inductive hypothesis of the proof of Lemma 4.5 can be immediately verified in this new case as well. The deduction of the theorem from this result is now the same as that of Theorem 4.3 from Theorem 4.2. □

To prove the completeness theorem for deductions from premises, we need the obvious notion of a CSMT from set of premises.

Definition 4.14 (CSMT): We define the *complete systematic modal tableau* (CSMT) *from a set Σ of sentences* starting with a sentence φ by induction as follows. τ_0 is the atomic tableaux with root $Fp_1 \Vdash \varphi$. For the inductive step, we modify Definition 4.8 in much the same same way that the notion of a CST was modified in Definition II.6.9 to accommodate premises. Let $\Sigma = \{\psi_j \mid j \in \mathbb{N}\}$. At even stages of the construction we proceed as in (i)–(iv) of Definition 4.8 as appropriate. At odd stages $n = 2k+1$, we adjoin $Tp_i \Vdash \psi_j$ for every $i, j < k$ to every noncontradictory path P in τ_n on which p_i occurs. We do not terminate the construction unless, for every $\psi \in \Sigma$, $Tp \Vdash \psi$ appears on every noncontradictory path P on which p is mentioned.

Theorem 4.15 (Completeness, $\Sigma \vDash \varphi \Rightarrow \Sigma \vdash \varphi$): *If φ is a logical consequence of a set Σ of sentences of modal logic, then there is a modal tableau proof of φ from Σ.*

Proof: Suppose φ is a logical consequence of Σ. The argument for Proposition 4.10 still shows that the CSMT from Σ is finished. Its definition also guarantees that, for every $\psi \in \Sigma$, $Tp \Vdash \psi$ appears on any noncontradictory path P on which p is mentioned. The argument for Theorem 4.11 now shows that if the CSMT from Σ with root node $Fp_1 \Vdash \varphi$ is not a tableau proof of φ from Σ, then there is a frame $\mathcal{C}$ in which every $\psi \in \Sigma$ is forced but in which φ is not forced. So we have the desired contradiction. □

It is worth remarking that the particular construction we have given of the CSMT and the accompanying proof of completeness shows that, as far as validity (in modal logic) is concerned, we could restrict our attention to frames in which each possible world has at most finitely many predecessors. Indeed, we could even require the transitive closure of the accessibility relation to have this finite predecessor property. (See Exercises 5–7.) The innocuousness of these restrictions should be compared with the very real

changes in the notion of validity that are effected by other restrictions on the accessibility relation. Various such restrictions, including reflexivity and transitivity, are considered in §5.

Exercises

1. Complete the proof of Lemma 4.5 by considering cases (F$\forall$) and (F$\Box$).
2. Complete part (ii) of the proof of Lemma 4.11 by dealing with the connectives $\neg$, $\wedge$ and $\rightarrow$.
3. Verify the case of the existential quantifier in part (iii) of the proof of Theorem 4.11.
4. Verify the case of $\Diamond$ in part (iv) of the proof of Theorem 4.11.
5. Prove that a modal sentence φ is valid if and only if it is forced in every frame $\mathcal{C} = (W, S, \mathcal{C}(p))$ for which S has the *finite predecessor property*, i.e., for each $p \in W$, the set $\{q \in W \mid qSp\}$ is finite.
6. If S is a binary relation on a set W, the *transitive-reflexive closure* of S, TRC(S), is the intersection of all reflexive and transitive binary relations T on W which contain S. Prove that $\mathrm{TRC}(S) = \cup S_n$ where $S_0 = S$, $S_1 = S \cup \{(p,p) \mid \exists q(pSq \vee qSp)\}$ and $S_{n+1} = S_n \cup \{(p,q) \mid \exists w(pS_n w \wedge wS_n q)\}$.
7. Prove that a modal sentence φ is valid if and only if it is forced in all frames $\mathcal{C} = (W, S, \mathcal{C}(p))$ in which, for each $p \in W$, the set $\{q \in W \mid q\mathrm{TRC}(S)p\}$ is finite.
8. **Constant Domains**: We can modify our conception of modal logic to incorporate the restriction that the domains of all possible worlds be the same. In the Definition 2.1 of a frame we require that $C(p) = C(q)$ for every $p, q \in W$. Definitions 2.2 and 2.3 of forcing and validity remain unchanged. In Definition 3.2 of tableaux we change the notion of "any appropriate c" in (ii) to mean any c in $\mathcal{L}$ or appearing in any entry on P. With these definitions, prove the soundness and completeness theorems for constant domain modal logic.

5. Modal Axioms and Special Accessibility Relations

For many particular applications of modal logic various special types of accessibility relations seem appropriate. For example, in analyzing the behavior of a computing machine one might want to insist that the accessibility relation reflect some aspects of time as seen by the machine. In such situations it might be appropriate to require that the accessibility relation be reflexive or transitive. If one is interested in perpetual processes, one might want to require that every state have a strict successor. From another point of view, some particular intended interpretation of the modal operators might suggest axioms that one might wish to add to modal logic. Thus, for example, if $\Box$ means "it is necessarily true that" or "I know that", one might want to include an axiom scheme asserting $\Box\varphi \to \varphi$ for every sentence φ. On the other hand, if $\Box$ is intended to mean "I believe that", then we might well reject $\Box\varphi \to \varphi$ as an axiom: I can have false beliefs. In the case of modeling belief, however, we might want to incorporate an "introspection" axiom like $\Box\varphi \to \Box\Box\varphi$ (what I believe, I believe I believe) or $\neg\Box\varphi \to \Box\neg\Box\varphi$ (what I don't believe, I believe I don't believe). But if $\Box$ is intended to mean "it is necessarily true that", then the last of these axioms ($\neg\Box\varphi \to \Box\neg\Box\varphi$) is not all compelling.

As it turns out, there are close connections between certain natural restrictions on the accessibility relation in frames and various common axioms for modal logics. Indeed, it is often possible to formulate precise equivalents in the sense that the sentences forced in all frames with a specified type of accessibility relation are precisely the logical consequences of some axiom system. In this section we will present several examples of this phenomena. In the next section we will describe traditional Hilbert style axiom and rule systems for these various modal logics in the style of I.7 and II.8.

Before considering specific examples, we introduce some general notation that will simplify our discussion.

Definition 5.1:

(i) Let $\mathcal{F}$ be a class of frames and φ a sentence of a modal language $\mathcal{L}$. We say that φ is *$\mathcal{F}$–valid*, $\models_{\mathcal{F}} \varphi$, if φ is forced in every frame $\mathcal{C} \in \mathcal{F}$.

(ii) Let F be a rule or a family of rules for developing tableaux, i.e., a rule or set of rules of the form "if τ is a tableau, P a path on τ and τ' is gotten from τ by adding some entry E (of a specified type) to the end of P then τ' is a tableau". The *F–tableaux* are defined by induction as in Definition 3.2 with clause (ii) extended to include the formation rules in F. An F–tableau is an F–tableau proof of a sentence φ if it is finite, has a root node of the form $Fp \Vdash \varphi$, and every path is contradictory. We say that φ is *F–provable* (or *F–tableau provable*), $\vdash_F \varphi$, if it has an F–tableau proof.

Example 5.2 (The knowledge axiom and reflexivity): We saw in Example 3.7 that the scheme $\Box\varphi \to \varphi$ is not valid. This scheme is traditionally called the *knowledge axiom* and is denoted by T. If one is modeling knowledge and is of the opinion that one cannot know φ unless φ is true, then this is a plausible axiom. One might well want to restrict one's attention to its logical consequences. We know one way to handle this restriction proof theoretically: We can consider only tableaux deductions from the set of premises consisting of all (closures of) instances of T.

Syntactically, this approach is fairly unwieldy. It to a large extent vitiates the considerable advantages of the tableaux method over a Hilbert style system. The major problem is that if we are trying to get a clever short proof (rather than using the CSMT), we have no good way of knowing which instances of the axiom to insert at any particular point. The basic tableau method, however, generally leads to proofs in fairly direct and easily predictable ways. In addition, this axiomatic approach gives us little direct insight into any semantics corresponding to T. It is far from obvious how the requirement that every instance of T be forced in every frame being considered can give us a characterization of the appropriate class of frames.

The solution to both of these problems is suggested by the attempt (in Example 3.6) to prove a typical instance of T:

1	$Fw \Vdash \Box\varphi \to \varphi$	
2	$Tw \Vdash \Box\varphi$	by 1
3	$Fw \Vdash \varphi$	by 1

At this point, the attempted proof flounders. It is, however, obvious that if we knew that wSw, we would have a complete proof. This suggests that we incorporate reflexivity into both our semantics and tableau proof rules for T.

Definition 5.3:

(i) $\mathcal{R}$ is the class of all *reflexive frames*, i.e., all frames in which the accessibility relation is reflexive (wSw holds for every $w \in W$).

(ii) R is the *reflexive tableau development rule* which says that, given a tableau τ, we may form a new tableau τ' by adding wSw to the end of any path P in τ on which w occurs.

(iii) $\mathcal{T}$ is the set of universal closures of all instances of the scheme T: $\Box\varphi \to \varphi$.

We claim that these notions correctly capture the import of T in the following precise sense:

Theorem 5.4: *For any sentence φ of our modal language $\mathcal{L}$, the following conditions are equivalent:*

(i) *$\mathcal{T} \models \varphi$, φ is a logical consequence of $\mathcal{T}$.*

(ii) *$\mathcal{T} \vdash \varphi$, φ is a tableau provable from $\mathcal{T}$.*

(iii) *$\models_{\mathcal{R}} \varphi$, φ is forced in every reflexive $\mathcal{L}$-frame.*

(iv) *$\vdash_R \varphi$, φ is provable with the reflexive tableau development rule.*

Note that this theorem not only characterizes the logical consequences of the axiom scheme T, it also includes a soundness (iv $\Rightarrow$ iii) and completeness (iii $\Rightarrow$ iv) theorem for the modal logic with a semantics in which validity means being forced in all reflexive frames and a proof system consisting of the usual construction principles plus R. The equivalence of (i) and (ii) is just the soundness and completeness theorems for deductions from sets of premises (Theorems 4.13 and 4.15). To complete the proof of equivalences we will prove the soundness and completeness results for R (iii $\Leftrightarrow$ iv) and the equivalence of (i) and (iii).

Proof (Soundness, $\vdash_R \varphi \Rightarrow \models_{\mathcal{R}} \varphi$): Suppose τ is an R–tableau with root $Fp \Vdash \varphi$ and $\mathcal{C} = (W, S, \mathcal{C}(p))$ is a reflexive frame. If there is a $p \in W$ such that $p \nVdash \varphi$, then the proof of Theorem 4.2 shows that $\mathcal{C}$ agrees with some path on τ. The only new point is that when the rule R is used to put some entry wSw on τ, it is true in $\mathcal{C}$ by the reflexivity of S. Thus, if τ is an R–proof of φ ($\vdash_R \varphi$), there cannot be a reflexive frame which fails to force φ, i.e., $\models_{\mathcal{R}} \varphi$. □

Proof (Completeness, $\models_{\mathcal{R}} \varphi \Rightarrow \vdash_R \varphi$): We begin with the obvious notion of a *complete systematic reflexive modal tableau*, R–CSMT, with root $Fp_1 \Vdash \varphi$. It is defined as was the CSMT from premises Σ in Definition 4.14 except that at odd stages $n = 2k+1$ we adjoin p_iSp_i to every noncontradictory path P on which p_i occurs, for every $i < k$. If the R–CSMT beginning with $Fp_1 \Vdash \varphi$ is not an R–proof of φ, then there is a noncontradictory path P through it. Once again, the argument for Proposition 4.10 shows that any R–CSMT is finished. Its definition easily implies that pSp appears on every noncontradictory path P on which p occurs. Thus the proof of Theorem 4.11 defines a reflexive frame $\mathcal{C}$ from P which agrees with P. This shows that φ is not $\mathcal{R}$–valid, as required. □

Proof ($\mathcal{T} \models \varphi \Rightarrow \models_{\mathcal{R}} \varphi$): It is immediate from Definition 2.2(ix) of forcing $\Box\varphi$ that every instance of T is forced in every reflexive frame. Thus if φ is forced in every frame in which all instances of T are forced, φ is forced in every reflexive frame, i.e., $\mathcal{T} \models \varphi \Rightarrow \models_{\mathcal{R}} \varphi$.

Alternatively, recall that the attempt at a tableau proof of a typical member $\Box\varphi \to \varphi$ of $\mathcal{T}$ presented at the beginning of this example easily becomes, as we noted, an R–proof. (Note that an arbitrary member of $\mathcal{T}$ is of the form $\forall\vec{x}(\Box\varphi \to \varphi)$. In this case the tableau proof just begins by instantiating the universally quantified variables with new constants. It then proceeds as before.) Thus $\vdash_R \theta$ for every $\theta \in \mathcal{T}$. The soundness theorem just proved for $\vdash_R$ then says that $\models_{\mathcal{R}} \theta$ for every $\theta \in \mathcal{T}$. In particular, if $\mathcal{T} \models \varphi$, φ is forced in every reflexive frame, i.e., $\models_{\mathcal{R}} \varphi$ as required. (Since this alternative proof is easier, it will be mimicked in late examples.) □

Proof ($\models_{\mathcal{R}} \varphi \Rightarrow \mathcal{T} \models \varphi$): Suppose, for the sake of a contradiction, that $\models_{\mathcal{R}} \varphi$ but $\mathcal{T} \nvDash \varphi$. Let τ be the CSMT from $\mathcal{T}$ with root $Fp_1 \Vdash \varphi$. By assumption and Theorem 4.13, τ is not a proof of φ, i.e., there is a noncontradictory path P in τ. Let $\mathcal{C} = (W, S, \mathcal{C}(p))$ be the frame defined from P as in Theorem 4.11. Let $\mathcal{C}' = (W', S', \mathcal{C}'(p))$ be the reflexive closure of $\mathcal{C}$, i.e., $W' = W$ and $\mathcal{C}'(p) = \mathcal{C}(p)$ but $S' = S \cup \{(w,w) \mid p \in W\}$ is the reflexive closure of S. We claim that the reflexive frame $\mathcal{C}'$ agrees with P. Hence $p_1 \nVdash_{\mathcal{C}'} \varphi$ and we have the desired contradiction. □

The proof that $\mathcal{C}'$ agrees with P is the same as that for $\mathcal{C}$ in Theorem 4.11 except that new arguments are needed in the induction step to conclude, from the appearance of $Tp \Vdash \Box\psi$ or $Fp \Vdash \Diamond\psi$ on P, that $p \Vdash_{\mathcal{C}'} \psi$ or $p \nVdash_{\mathcal{C}'} \psi$, respectively. It clearly suffices to prove the following lemma (the case that $p = q$ is the nontrivial one and the one of immediate interest):

Lemma 5.5:

(i) *If $Tp \Vdash \Box\psi$ appears on P and $pS'q$, then $Tq \Vdash \psi$ appears on P.*

(ii) *If $Fp \Vdash \Diamond\psi$ appears on P and $pS'q$, then $Fq \Vdash \psi$ appears on P.*

Proof: First note that if $p \neq q$ then pSq appears on P and the fact that τ is finished yields our conclusion. Suppose then that $p = q$.

(i) Suppose $Tp \Vdash \Box\psi$ appears on P and ψ is of the form $\theta(c_1, \ldots c_n)$ where the c_i displayed are all the constants in ψ not in the original language. The element $\forall x_1 \ldots \forall x_n(\Box\theta(x_1, \ldots, x_n) \to \theta(x_1, \ldots x_n))$ of $\mathcal{T}$ appears on P as τ is a CSMT from $\mathcal{T}$. As τ is finished and $Tp \Vdash \Box\theta(c_1, \ldots c_n)$ appears on P, so does $Tp \Vdash \Box\theta(c_1, \ldots, c_n) \to \theta(c_1, \ldots c_n)$. Thus, $Tp \Vdash \Box\psi \to \psi$ appears on P.) As τ is finished, either $Fp \Vdash \Box\psi$ or $Tp \Vdash \psi$ appear on P. The former would make P contradictory. Thus $Tp \Vdash \psi$ appears on P as required.

(ii) If $Fp \Vdash \Diamond\psi$ appears on P, then (as in (i)) $Tp \Vdash \Box\neg\psi \to \neg\psi$ appears on P. Thus either $Fp \Vdash \Box\neg\psi$ or $Tp \Vdash \neg\psi$ appears on P. In the former case, we eventually reduce $Fp \Vdash \Box\neg\psi$ on P by putting both pSw and $Fw \Vdash \neg\psi$ on P for some w. As τ is finished, the first of these, pSw, combines with $Fp \Vdash \Diamond\psi$ to guarantee that $Fw \Vdash \psi$ appears on P. Similarly, the second, $Fw \Vdash \neg\psi$, guarantees that $Tw \Vdash \psi$ appears on P. Thus in this case, P would be contradictory contrary to our assumption. In the latter case, $Fp \Vdash \psi$ appears on P as required. □

The proof of Theorem 5.4 is now complete. □

Example 5.6 (Introspection and Transitivity): The scheme PI, $\Box\varphi \to \Box\Box\varphi$, was traditionally called "4". It is now often called the *scheme of positive introspection* as it expresses the view that what I believe, I believe I believe. Once again, an attempt at proving a typical instance provides the clue to the appropriate semantics and proof rule. (See Figure 51.)

There is no contradiction. By reading off the true atomic statements from the tableaux, we get a three world frame $\mathcal{C} = (W, S, \mathcal{C}(p))$ with $W = \{w, v, u\}$, $S = \{(v,u), (w,v)\}$, $\mathcal{C}(v) \models \varphi$ and $\mathcal{C}(u), \mathcal{C}(w) \nvDash \varphi$. Such a frame $\mathcal{C}$ is a counterexample to the validity of $\Box\varphi \to \Box\Box\varphi$ as this sentence is not forced at w in $\mathcal{C}$. We can, however, produce a contradiction by adding wSu to the accessibility relation. The key here is transitivity.

Definition 5.7:

(i) $\mathcal{TR}$ is the class of all *transitive frames*, i.e., all frames $\mathcal{C} = (W, S, \mathcal{C}(p))$ in which S is transitive: $wSv \wedge vSu \Rightarrow wSu$.

1	$Fw \Vdash \Box\varphi \to \Box\Box\varphi$	
2	$Tw \Vdash \Box\varphi$	by 1
3	$Fw \Vdash \Box\Box\varphi$	by 1
4	wSv new v	by 3
5	$Fv \Vdash \Box\varphi$	by 3
6	vSu new u	by 5
7	$Fu \Vdash \varphi$	by 5
8	$Tv \Vdash \varphi$	by 2, 4

FIGURE 51

(ii) TR is the *transitive tableau development rule* which says that if wSv and vSu appear on a path P of a tableau τ then we can produce another tableau τ' by appending wSu to the end of P.

(iii) $\mathcal{PI}$ is the set of all universal closures of instances of the scheme PI: $\Box\varphi \to \Box\Box\varphi$.

The situation is now much the same as in Example 5.2.

Theorem 5.8: *For any sentence φ of our modal language $\mathcal{L}$, the following conditions are equivalent:*

(i) $\mathcal{PI} \vDash \varphi$, *$\varphi$ is a logical consequence of $\mathcal{PI}$.*

(ii) $\mathcal{PI} \vdash \varphi$, *$\varphi$ is a tableau provable from $\mathcal{PI}$.*

(iii) $\vDash_{\mathcal{TR}} \varphi$, *$\varphi$ is forced in every transitive $\mathcal{L}$-frame.*

(iv) $\vdash_{\mathrm{TR}} \varphi$, *$\varphi$ is provable with the transitive tableau development rule.*

The proof of this theorem is essentially the same as for the corresponding results for reflexive frames (Theorem 5.4). We of course replace reflexive, R, $\mathcal{R}$ and $\mathcal{T}$ by transitive, TR, $\mathcal{TR}$ and $\mathcal{PI}$, respectively and make the other obvious changes. For example, the TR–CSMT at stage $n = 2k+1$ adjoins

to the end of every noncontradictory path P the entry p_iSp_j if there is some q such that p_iSq and qSp_j appear on P. The only part of the proof that is not essentially the same as that of Theorem 5.4 is the proof that $\vDash_{\mathcal{TR}} \varphi \Rightarrow \mathcal{PI} \vDash \varphi$ and, in particular, the proof of the analog of Lemma 5.5.

Proof ($\vDash_{\mathcal{TR}} \varphi \Rightarrow \mathcal{PI} \vDash \varphi$): Suppose, for the sake of a contradiction, that $\vDash_{\mathcal{TR}} \varphi$ but $\mathcal{PI} \nvDash \varphi$. Let τ be the CSMT from $\mathcal{PI}$ with root $Fp_1 \Vdash \varphi$. By assumption and Theorem 4.13, τ is not a proof of φ. So there is a noncontradictory path P in τ. Let $\mathcal{C} = (W, S, \mathcal{C}(p))$ be the frame defined from P as in Theorem 4.11. Let $\mathcal{C}'(W', S', \mathcal{C}'(p))$ be the transitive closure of $\mathcal{C}$, i.e., $W' = W$ and $\mathcal{C}'(p) = \mathcal{C}(p)$ but S' is the *transitive closure* of S, i.e., $S' \equiv \cup S_n$ where $S_0 = S$ and $S_{n+1} = S_n \cup \{(p,q) \mid \exists w(pS_n\, w \wedge wS_nq)\}$. (Note that it S' is transitive by construction.) We claim that the transitive frame $\mathcal{C}'$ agrees with P. Hence $p_1 \nVdash_{\mathcal{C}'} \varphi$ and we have the desired contradiction. □

The proof that $\mathcal{C}'$ agrees with P is again the same as that for $\mathcal{C}$ in Theorem 4.11 except that new arguments are needed in the induction step to conclude, from the appearance of $Tp \Vdash \Box\psi$ or $Fp \Vdash \Diamond\psi$ on P, that $q \Vdash_{\mathcal{C}'} \psi$ or $q \nVdash_{\mathcal{C}'} \psi$, respectively, for every q such that $pS'q$. Once again, it clearly suffices to prove the following lemma:

Lemma 5.9:

(i) *If $Tp \Vdash \Box\psi$ appears on P and $pS'q$, then $Tq \Vdash \psi$ appears on P.*

(ii) *If $Fp \Vdash \Diamond\psi$ appears on P and $pS'q$, then $Fq \Vdash \psi$ appears on P.*

Proof: We proceed by induction on the stage n at which (p,q) enters S'. For $n = 0$, the conclusion in both cases follows from the fact that τ is finished. Suppose that the lemma holds for $(u,v) \in S_n$ and that $(p,q) \in S_{n+1}$. Let w be such that $(p,w), (w,q) \in S_n$.

(i) If $Tp \Vdash \Box\psi$ appears on P, then (as argued in the proof of Lemma 5.5) so does $Tp \Vdash \Box\psi \to \Box\Box\psi$. As τ is finished, either $Fp \Vdash \Box\psi$ or $Tp \Vdash \Box\Box\psi$ appears on P. As P is not contradictory, it must be $Tp \Vdash \Box\Box\psi$ that appears on P. As $(p,w) \in S_n$, $Tw \Vdash \Box\psi$ appears on P by induction. As $(w,q) \in S_n$ as well, $Tq \Vdash \psi$ also appears on P, as required.

(ii) The dual case is similar and we leave it as Exercise 1. □

Example 5.10 (Negative introspection and Euclidean frames): The next scheme we consider is NI, the *negative introspection scheme* (what I don't believe, I believe I don't believe): $\neg\Box\varphi \to \Box\neg\Box\varphi$. This scheme has traditionally been denoted by "E" (for Euclidean) or "5". Figure 52 gives an attempt at proving an instance of NI.

The key to completing a proof here is to be able to adjoin uSv when we have both wSv and wSu. We could then conclude $Tv \Vdash \varphi$ from line 9 contracting line 6. The story should by now be quite familiar.

1	$Fw \Vdash \neg\Box\varphi \to \Box\neg\Box\varphi$	
2	$Tw \Vdash \neg\Box\varphi$	by 1
3	$Fw \Vdash \Box\neg\Box\varphi$	by 1
4	$Fw \Vdash \Box\varphi$	by 2
5	wSv new v	by 4
6	$Fv \Vdash \varphi$	by 4
7	wSu new u	by 3
8	$Fu \vdash \neg\Box\varphi$	by 3
9	$Tu \Vdash \Box\varphi$	by 8

FIGURE 52

Definition 5.11:

(i) $\mathcal{E}$ is the class of all *Euclidean frames*, i.e., all frames $\mathcal{C} = (W, S, \mathcal{C}(p))$ in which S is *Euclidean*: $wSv \wedge wSu \Rightarrow uSv$.

(ii) E is the *Euclidean tableau development rule* which says that if wSv and wSu appear on a path P of a tableau τ then we can produce another tableau τ' by appending uSv to the end of P.

(iii) $\mathcal{NI}$ is the set of all universal closures of instances of the scheme NI: $\neg\Box\varphi \to \Box\neg\Box\varphi$.

Theorem 5.12: *For any sentence φ of our modal language $\mathcal{L}$, the following conditions are equivalent:*

(i) $\mathcal{NI} \models \varphi$, *$\varphi$ is a logical consequence of $\mathcal{NI}$.*

(ii) $\mathcal{NI} \vdash \varphi$, *$\varphi$ is a tableau provable from $\mathcal{NI}$.*

(iii) $\models_{\mathcal{E}} \varphi$, *$\varphi$ is forced in every Euclidean $\mathcal{L}$-frame.*

(iv) $\vdash_E \varphi$, *φ is provable with the Euclidean tableau development rule.*

The proof of this theorem is essentially the same as for the corresponding results for reflexive or transitive frames (Theorems 5.4 and 5.9). The frame $\mathcal{C}'$ needed in the proof *of* $\models_{\mathcal{E}} \varphi \Rightarrow \mathcal{NI} \models \varphi$ is the *Euclidean closure* of $\mathcal{C}$. It is defined by setting $S' = \cup S_n$, where $S_0 = S$ and $S_{n+1} = \{(p,q) \mid \exists w((w,p), (w,q) \in S_n)\}$. The crucial point is again the analog of Lemma 5.5:

Lemma 5.13:

(i) *If $Tp \Vdash \Box\psi$ appears on P and $pS'q$, then $Tq \Vdash \psi$ appears on P.*

(ii) *If $Fp \Vdash \Diamond\psi$ appears on P and $pS'q$, then $Fq \Vdash \psi$ appears on P.*

Proof: We proceed by induction on the stage n at which (p,q) enters S'. For $n = 0$, the conclusion in both cases follows from the fact that τ is finished. Suppose that the lemma holds for $(u,v) \in S_n$ and that $(p,q) \in S_{n+1}$. Let w be such that (w,p), $(w,q) \in S_n$.

(i) If $Tp \Vdash \Box\psi$ appears on P, then (as before) $Tw \Vdash \neg\Box\psi \to \Box\neg\Box\psi$ appears on P. As τ is finished either $Fw \Vdash \neg\Box\psi$ or $Tw \Vdash \Box\neg\Box\psi$ appears on P. In the former case, $Tw \Vdash \Box\psi$ appears on P and, as required, so does $Tq \Vdash \psi$, by induction. In the latter case, $Tp \Vdash \neg\Box\psi$ would appear on P. As this would guarantee that $Fp \Vdash \Box\psi$ appears on P, it would make P contradictory contrary to our assumption. Thus $Fw \Vdash \neg\Box\psi$ and so $Tq \Vdash \psi$ appear on P as required.

(ii) The dual case is similar and is left as Exercise 2. ☐

Example 5.14 (Serial axioms and frames): Our next example is the scheme traditionally called "D": $\Box\varphi \to \neg\Box\neg\varphi$. It says that if I believe φ then I don't believe $\neg\varphi$. It is now often referred to as the *serial scheme*. Once again we begin with an attempt to prove an instance of D:

1	$Fw \Vdash \Box\varphi \to \neg\Box\neg\varphi$	
2	$Tw \Vdash \Box\varphi$	by 1
3	$Fw \Vdash \neg\Box\neg\varphi$	by 1
4	$Tw \Vdash \Box\neg\varphi$	by 3

FIGURE 53

A frame with an empty accessibility relation will provide a counterexample. The existence of a v such that wSv would allow us to deduce the desired contradiction. (We could conclude $Tv \Vdash \varphi$ from ($T\Box$) and $Fv \Vdash \varphi$ from ($T\Box$ and $F\neg$).)

Definition 5.15:

(i) $\mathcal{SE}$ is the class of all *serial frames*, i.e., all frames $\mathcal{C} = (W, S, \mathcal{C}(p))$ in which there is, for every $p \in W$, a q such that pSq.

(ii) SE is the *serial tableau development rule* which says that if p appears on a path P of a tableau τ then we can produce another tableau τ' by appending pSq to the end of P for a new q.

(iii) $\mathcal{D}$ is the set of all universal closures of instances of the scheme D: $\Box\varphi \to \neg\Box\neg\varphi$.

Theorem 5.16: *For any sentence φ of our modal language $\mathcal{L}$, the following conditions are equivalent:*

(i) $\mathcal{D} \models \varphi$, *$\varphi$ is a logical consequence of $\mathcal{D}$.*

(ii) $\mathcal{D} \vdash \varphi$, *$\varphi$ is a tableau provable from $\mathcal{D}$.*

(iii) $\models_{\mathcal{SE}} \varphi$, *$\varphi$ is forced in every serial $\mathcal{L}$-frame.*

(iv) $\vdash_{\mathrm{SE}} \varphi$, *$\varphi$ is provable with the serial tableau development rule.*

The proof is as before. In this case the crucial point in the verification that $\models_{\mathcal{SE}} \varphi \Rightarrow \mathcal{D} \models \varphi$ is that the frame $\mathcal{C}$ defined from a path P on the CSMT τ from $\mathcal{D}$ is automatically serial. If p appears on P, then so does $Tp \Vdash \Box\psi \to \neg\Box\neg\psi$ for some ψ. As τ is finished, either $Fp \Vdash \Box\psi$ or $Tp \Vdash \neg\Box\neg\psi$ appears on P. In either case one or two more appeals to the fact that τ is finished produces an entry pSq on P for some new q as required. $\Box$

Additional examples of tableau rules and their corresponding classes of frames can be found in Exercises 3–4 and 6.1–4.

Exercises

1. Prove (ii) of Lemma 5.9.
2. Prove (ii) of Lemma 5.13.
3. **Tree frames**: A frame $\mathcal{C}$ is a *tree frame* if its accessibility relation S defines a tree (in the sense of Definition I.1.1) on W by setting $p < q$ iff pSq. Extend Theorem 5.8 by proving that the following additional equivalence can be added to the list:

   (v) φ is forced in every tree frame.

   (Hint: Look at Exercises 4.5–7.)

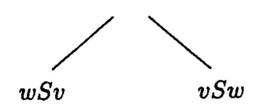


FIGURE 54

4. **Linear frames**: A frame $\mathcal{C}$ is a *linear frame* if its accessibility relation S defines a linear ordering on W by setting $p < q$ iff pSq. The linear tableau rule L says that if the possible worlds w and v appear on a path P of a tableau τ then we can produce another tableau τ' by appending the new tableau (Figure 54) to the end of P. Prove that a sentence φ of a modal language $\mathcal{L}$ is valid in all linear frames if and only if it is provable using the linear tableau rule.

6*. An Axiomatic Approach

We conclude this chapter with a description of a basic Hilbert–style system for modal logic and a brief catalog of some of the standard modal systems with their traditional nomenclature. In line with the view taken in II.8 where we used a language containing only the connectives $\neg$ and $\to$ and the quantifier $\forall$, we here use only the modal operator $\Box$ and view $\Diamond$ as a defined symbol by replacing $\Diamond\varphi$ with $\neg\Box\neg\varphi$.

The proof system for classical logic presented in II.8 can be extended to one called K for modal logic by adding on one new axiom scheme and one new rule:

6.1 Axiom scheme (vi): $\Box(\alpha \to \beta) \to (\Box\alpha \to \Box\beta)$.

6.2 Rule (iii) *Necessitation*: From α infer $\Box\alpha$.

Of course, we now include all instances in our modal language $\mathcal{L}$ of the original axiom schemes (i)–(v) as well as (vi). Similarly, the rules apply to all formulas of $\mathcal{L}$.

The definitions of proofs and proofs from a set Σ of sentences are now the same as in classical logic. It is easy to see that all instances of the axioms of K are valid. (Example 3.5 gives a tableau proof for a typical instance of (vi). The classical tableau proofs for the other axioms remain correct for instances in $\mathcal{L}$ as well. Note that by the validity of an open formula we mean the validity of its universal closure.) As the rules of K also preserve validity, all theorems of this system are valid. (Note that $\alpha \models \Box\alpha$ is immediate even for open α once one remembers that the validity of α means that its universal closure is forced in every frame. Alternatively, one can construct a tableau proof showing that $\forall\vec{x}\alpha(\vec{x}) \Vdash \forall\vec{x}\Box\alpha(\vec{x})$ as in Exercise 3.11.) A standard style completeness theorem for K then shows that it defines the same set of theorems as does our tableau system.

The schemes we have analyzed in §5 are often added to K to produce axiom systems for specialized study. Completeness theorems for each of the Hilbert–style systems listed below can be found in Chapter 9 of Hughes and Cresswell [1984, 4.4]. Thus each has the same theorems as the analogous system of §5.

T is the system consisting of K and the axiom scheme T of Example 5.2: $\Box\varphi \to \varphi$. It is often regarded as the logic of knowledge. As we have seen, its theorems are precisely those sentences forced in all reflexive frames.

S4 is the system consisting of T and the scheme PI of Example 5.6: $\Box\varphi \to \Box\Box\varphi$. It is routine to combine the results of Theorems 5.4 and 5.8 to see that the theorems of S4 are precisely the sentences forced in every reflexive and transitive frame (Exercise 1.) If T is omitted we have the system K4 consisting of K and the scheme PI.

S5 is the system consisting of S4 and the scheme E of Example 5.10: $\neg\Box\varphi \to \Box\neg\Box\varphi$. Its theorems are precisely those sentences forced in every reflexive, transitive, Euclidean frame. As the reflexive, transitive Euclidean relations are precisely the equivalence relations (Exercise 2), the theorems of S5 are those sentences forced in every frame whose accessibility relation is an equivalence relation (Exercise 3). Omitting T gives us the system K5 consisting of K4 and the scheme E.

In fact, we can say somewhat more about S5. Its theorems are the sentences forced in all *complete frames*.

Definition 6.3:

(i) $\mathcal{M}$ is the class of all *complete frames*, i.e., all frames $\mathcal{C} = (W, S, \mathcal{C}(p))$ in which pSq for every $p, q \in W$.

(ii) M is the *complete tableau development rule* which says that if p and q appear on a path P of a tableau τ then we can produce another tableau τ' by appending pSq.

(iii) $\mathcal{S}5$ is the set of all universal closures of instances of the schemes T, PI and NI.

Theorem 6.4: *For any sentence φ of our modal language $\mathcal{L}$, the following conditions are equivalent:*

(i) $\mathcal{S}5 \models \varphi$, *$\varphi$ is a logical consequence of $\mathcal{S}5$.*

(ii) $\mathcal{S}5 \vdash \varphi$, *$\varphi$ is a tableau provable from $\mathcal{S}5$.*

(iii) $\models_{\mathcal{M}} \varphi$, *$\varphi$ is forced in every complete $\mathcal{L}$-frame.*

(iv) $\vdash_M \varphi$, *φ is provable with the complete tableau development rule.*

Of course, this theorem can be proven as were the previous ones (Exercise 4). It can also be deduced (Exercise 6) from the one for frames with equivalence relations (Exercise 3) by the reduction supplied in Exercise 5.

Similar systems have been used (Moore [1984, 5.5] and [1985, 5.5]) as a basis for autoepistemic logic. If extended to have multiple modal operators $\Box_A$, one for each processor A, they are suitable for reasoning about distributed networks of agents (Halpern and Moses [1984, 5.5]).

Exercises

1. Prove that the theorems of S4, i.e., the (logical) consequences of the union of $\mathcal{T}$ and $\mathcal{TR}$, are the ones provable in a system using both the reflexive and transitive tableau rules. These are also the sentences forced in every frame which is both reflexive and transitive.

2. A binary relation S is an *equivalence relation* if it is reflexive (wSw), symmetric ($wSv \Rightarrow vSw$) and transitive ($uSv \wedge vSw \Rightarrow uSw$). Prove that a reflexive, transitive binary relation S is an equivalence relation if and only if it is Euclidean.
3. Prove that the theorems of S5, i.e., the (logical) consequences of the union of $\mathcal{T}$, $\mathcal{TR}$ and $\mathcal{NI}$ are the ones provable in a system using the reflexive, transitive and Euclidean tableau rules. These are also the sentences forced in every frame in which the accessibility relation is an equivalence relation.
4. Give a direct proof of Theorem 6.4.
5. Let $\mathcal{C} = (W, S, \mathcal{C}(p))$ be a frame in which S is an equivalence relation. For $w \in W$, let $[w] = \{p \in W \mid pSw\}$ be the equivalence class of w. If $\mathcal{C}_w$ is the restriction of $\mathcal{C}$ to $[w]$, i.e., the frame $([w], S \restriction [w] \times [w], \mathcal{C}(p))$, then $w \Vdash_{\mathcal{C}} \varphi \Leftrightarrow w \Vdash_{\mathcal{C}_w} \varphi$ for every sentence φ.
6. Use Exercises 3 and 5 to give an alternate proof of Theorem 6.4.

Suggestions for Further Reading

Two good beginning texts on modal logic are Hughes and Cresswell [1984, 4.4] and Chellas [1980, 4.4]. The latter seems to be the common reference in the computer science literature for the basic material. A more advanced text is van Bentham [1983, 4.4] and van Bentham [1988, 4.4] is also useful. The encyclopedic treatment of tableau models in modal logic is Fitting [1983, 4.4]. In particular, this last book contains a proof of the decidability of propositional modal logic in §7 of Chapter 8. Many variations in the definition of both syntax and semantics can also be found there.

Linsky [1971, 4.4] is a good collection of important early articles on modal logic which includes Kripke's pioneering work and various pieces with a philosophical point of view. Some additional background information about applications of modal logic as well as a more comprehensive bibliography can be found in Nerode [1991, 4.1].

Galton [1987, 5.6], Goldblatt [1982, 5.6] and [1987, 5.6] and Turner [1984, 5.6] all stress the uses of various types of modal logics in computer science. Halpern and Moses [1985, 5.6] is a survey of logics of knowledge and belief. Thayse [1989, 5.6] is a thorough presentation of many types of modal logics directed towards deductive databases and AI. Thayse [1991, 5.6] supplies a wide ranging view of applications of these ideas in many areas of AI.

V Intuitionistic Logic

1. Intuitionism and Constructivism

During the past century, a major debate in the philosophy of mathematics has centered on the question of how to regard noneffective or nonconstructive proofs in mathematics. Is it legitimate to claim to have proven the existence of a number with some property without actually being able, even in principle, to produce one? Is it legitimate to claim to have proven the existence of a function without providing any way to calculate it? L. E. J. Brouwer is perhaps the best known early proponent of an extreme constructivist point of view. He rejected much of early twentieth century mathematics on the grounds that it did not provide acceptable existence proofs. He held that a proof of $p \vee q$ must consist of either a proof of p or one of q and that a proof of $\exists x P(x)$ must contain a construction of a witness c and a proof that $P(c)$ is true. At the heart of most nonconstructive proofs lies the law of the excluded middle: For every sentence A, $A \vee \neg A$ is true. Based on this law of classical logic one can prove that $\exists x P(x)$ by showing that its negation leads to a contradiction without providing any hint as to how to find an x satisfying P. Similarly, one can prove $p \vee q$ by proving $\neg(\neg p \wedge \neg q)$ without knowing which of p and q is true.

Example 1.1: We wish to prove that there are two irrational numbers a and b such that a^b is rational. Let $c = \sqrt{2}^{\sqrt{2}}$. If c is rational, then we may take $a = \sqrt{2} = b$. On the other hand, if c is not rational, then $c^{\sqrt{2}} = 2$ is rational and we may take $a = c$ and $b = \sqrt{2}$. Thus, in either case, we have two irrational numbers a and b such that a^b is rational. This proof depends on the the law of the excluded middle in that we assume that either c is rational or it is not. It gives us no clue as to which of the two pairs contain the desired numbers.

Example 1.2: Consider the proof of König's lemma (Theorem I.1.4). We defined the infinite path by induction. At each step we knew by induction that one of the finitely many immediate successors has infinitely many nodes below it. We then "picked" one such successor as the next node in our path. We had proved by induction that a disjunction is true and then simply continued the argument "by cases". As we had not in any way established which successor had infinitely many nodes below it, we have no actual construction of (no algorithm for defining) the infinite path that we proved to exist. Similar considerations apply to our proofs of completeness, compactness and other theorems.

A formal logic that attempts to capture Brouwer's philosophical position was developed by his student Heyting. This logic is called intuitionistic logic. It is an important attempt at capturing constructive reasoning. In particular, the law of the excluded middle is not valid in intuitionistic logic.

A number of paradigms have been suggested for explaining Brouwer's views. Each one can provide models or semantics for intuitionistic logic. One paradigm considers mathematical statements as assertions about our (or someone's) knowledge or possession of proofs. A sentence is true only when we know it to be so or only after we have proven it. At any moment we cannot know what new facts will be discovered or proven later. This interpretation fits well with a number of situations in computer science involving both databases and program verification. In terms of databases, one caveat is necessary. We view our knowledge as always increasing so new facts may be added but no old ones removed or contradicted. This is a plausible view of the advance of mathematical knowledge but in many other situations it is not accurate. Much of the time this model can still be used by simply attaching time stamps to facts. Thus the database records what we knew and when we knew it. The intuitionistic model is then a good one for dealing with deductions from such a database.

In terms of program verification, intuitionistic logic has played a basic role in the development of constructive proof checkers and reasoning systems. A key idea here is that, in accordance with Brouwer's ideas, the proof of an existential statement entails the construction of a witness. Similarly, the proof that for every x there is a y such that $P(x, y)$ entails the construction of an algorithm for computing a value of y from one for x. The appeal of such a logical system is obvious. On a practical level, there are now implementations of large-scale systems which (interactively) provide intuitionistic proofs of such assertions. The systems can then actually extract the algorithm computing the intended function. One then has a verified algorithm since the proof of existence is in fact a proof that the algorithm specified actually runs correctly. One such system is NUPRL developed at Cornell University by R. Constable [1986, 5.6] and others.

In this chapter, we present the basics of intuitionistic logic including a semantics developed by Kripke that reflects the "state of knowledge" interpretation of Heyting's formalism. In addition to the intuitive considerations, the claim that this choice of semantics adequately reflects constructivist reasoning is confirmed by the fact that the following *disjunction* and *existence properties* hold:

Theorem 2.20: *If $(\varphi \vee \psi)$ is intuitionistically valid, then either φ or ψ is intuitionistically valid.*

Theorem 2.21: *If $\exists x \varphi(x)$ is intuitionistically valid, then so is $\varphi(c)$ for some constant c.*

We then develop an intuitionistic proof theory based on a tableau method like that for classical logic and prove the appropriate soundness and completeness theorems. Of course, the completeness theorem converts the above theorems into ones about provability. We can (intuitionistically) prove $\varphi \vee \psi$ only if we can prove one of them. We can prove $\exists x \varphi(x)$ only if we can prove $\varphi(c)$ for some explicit constant c.

The presentation in this chapter is designed to be independent of Chapter IV. Thus there is some overlap of material. For those readers who have read Chapter IV, we supply a guide comparing classical, modal and intuitionistic logics in §6.

2. Frames and Forcing

Our notion of a language is the same as that for classical predicate logic in Chapter II except that we make one modification and two restrictions that simplify the technical details in the development of forcing. The modification is that we formally omit the logical connective $\leftrightarrow$ from our language. We instead view $\varphi \leftrightarrow \psi$ as an abbreviation for $(\varphi \to \psi) \wedge (\psi \to \varphi)$. Our restrictions are on the nonlogical components of our language. We assume throughout this chapter that every language $\mathcal{L}$ has at least one constant symbol but no function symbols other than constants.

We now present a semantics for intuitionistic logic which formalizes the "state of knowledge" interpretation.

Definition 2.1: Let $\mathcal{C} = (R, \leq, \{\mathcal{C}(p)\}_{p \in R})$ consist of a partially ordered set $(R, \leq)$ together with an assignment, to each p in R, of a structure $\mathcal{C}(p)$ for $\mathcal{L}$ (in the sense of Definition II.4.1). To simplify the notation, we will write $\mathcal{C} = (R, \leq, \mathcal{C}(p))$ instead of the more formally precise version, $\mathcal{C} = (R, \leq, \{\mathcal{C}(p)\}_{p \in R})$. As usual, we let $C(p)$ denote the domain of the structure $\mathcal{C}(p)$. We also let $\mathcal{L}(p)$ denote the extension of $\mathcal{L}$ gotten by adding on a name c_a for each element a of $C(p)$ in the style of the definition of truth in II.4. $A(p)$ denotes the set of atomic formulas of $\mathcal{L}(p)$ true in $\mathcal{C}(p)$. We say that $\mathcal{C}$ is a *frame for the language* $\mathcal{L}$, or simply an *$\mathcal{L}$-frame* if, for every p and q in R, $p \leq q$ implies that $C(p) \subseteq C(q)$, the interpretations of the constants in $\mathcal{L}(p) \subseteq \mathcal{L}(q)$ are the same in $\mathcal{C}(p)$ as in $\mathcal{C}(q)$ and $A(p) \subseteq A(q)$.

Often $p \leq q$ is read "*q extends p*", or "*q is a future of p*". The elements of R are called *forcing conditions, possible worlds, or states of knowledge.*

We now define the forcing relation for frames.

Definition 2.2 (Forcing for frames): Let $\mathcal{C} = (R, \leq, \mathcal{C}(p))$ be a frame for a language $\mathcal{L}$, p be in R and φ be a sentence of the language $\mathcal{L}(p)$. We give a definition of *p forces φ*, written $p \Vdash \varphi$ by induction on sentences φ.

(i) For atomic sentences φ, $p \Vdash \varphi \Leftrightarrow \varphi$ is in $A(p)$.
(ii) $p \Vdash (\varphi \to \psi) \Leftrightarrow$ for all $q \geq p$, $q \Vdash \varphi$ implies $q \Vdash \psi$.
(iii) $p \Vdash \neg\varphi \Leftrightarrow$ for all $q \geq p$, q does not force φ .
(iv) $p \Vdash (\forall x)\varphi(x) \Leftrightarrow$ for every $q \geq p$ and for every constant c in $\mathcal{L}(q)$, $q \Vdash \varphi(c)$.
(v) $p \Vdash (\exists x)\varphi(x) \Leftrightarrow$ there is a constant c in $\mathcal{L}(p)$ such that $p \Vdash \varphi(c)$.
(vi) $p \Vdash (\varphi \wedge \psi) \Leftrightarrow p \Vdash \varphi$ and $p \Vdash \psi$.
(vii) $p \Vdash (\varphi \vee \psi) \Leftrightarrow p \Vdash \varphi$ or $p \Vdash \psi$.

If we need to make the frame explicit, we will say that *p forces φ in $\mathcal{C}$* and write $p \Vdash_{\mathcal{C}} \varphi$.

Definition 2.3: Let φ be a sentence of the language $\mathcal{L}$. We say that *φ is forced in the $\mathcal{L}$-frame $\mathcal{C}$* if every p in R forces φ. We say *φ is intuitionistically valid* if it is forced in every $\mathcal{L}$-frame.

Clauses (ii), (iii) and (iv) defining $p \Vdash \varphi \to \psi$, $p \Vdash \neg\varphi$ and $p \Vdash (\forall x)\varphi(x)$ respectively each have a quantifier ranging over elements of the partial ordering, namely "for all q, if $q \geq p$, then ...". Clause (ii) says that p forces an implication $\varphi \to \psi$ only if any greater state of knowledge q which forces the antecedent φ also forces the consequent ψ. This is a sort of permanence of implication in the face of more knowledge. Clause (iii) says p forces the negation of φ when no greater state of knowledge forces φ. This says that $\neg\varphi$ is forced if φ cannot be forced by supplying more knowledge than p supplies. Clause (iv) says p forces a universally quantified sentence only if in all greater states of knowledge all instances of the sentence are forced. This is a permanence of forcing universal sentences in the face of any new knowledge beyond that supplied by p. Another aspect of the permanence of forcing that says the past does not count in forcing, only the future, is given by the following lemma. (Note that the logic of our metalanguage remains classical throughout this chapter. Thus, for example, in clause (ii) "implies" has the same meaning it had in Chapter I.)

Lemma 2.4 (Restriction Lemma): *Let $\mathcal{C} = (R, \leq, \mathcal{C}(p))$ be a frame, let q be in R and let $R_q = \{r \in R \mid r \geq q\}$. Then*

$$\mathcal{C}_q = (R_p\,,\ \leq\,,\ \mathcal{C}(p))$$

is a frame, where $\leq$ and the function $\mathcal{C}(p)$ are restricted to R_q. Moreover, for r in R_q, r forces φ in $\mathcal{C}$ iff r forces φ in $\mathcal{C}_q$.

Proof: By an induction on the length of formulas which we leave as Exercise 7. □

Consider the classical structures $\mathcal{C}(p)$ in an $\mathcal{L}$-frame $\mathcal{C}$. As we go from p to a $q > p$, we go from the classical structure $\mathcal{C}(p)$ associated with p to a (possibly) larger one $\mathcal{C}(q)$ associated with q with more atomic sentences classically true, and therefore fewer atomic sentences classically false.

Clauses (i), (v), (vi) and (vii) for the cases of atomic sentences, "and", "or" and "there exists" respectively are exactly as in the definition of truth in $\mathcal{C}(p)$given in II.4.3. The other clauses have a new flavor and indeed the classical truth of φ in $\mathcal{C}(p)$ and p'*s* forcing φ do not in general coincide. They do, however, in an important special case.

Lemma 2.5 (Degeneracy Lemma): *Let $\mathcal{C}$ be a frame for a language $\mathcal{L}$ and φ a sentence of $\mathcal{L}$. If p is a maximal element of the partial ordering R associated with $\mathcal{C}$, then φ is classically true in $\mathcal{C}(p)$, i.e., $\mathcal{C}(p) \vDash \varphi$, if and only if $p \Vdash \varphi$. In particular, if there is only one state of knowledge p in R, then $\mathcal{C}(p) \Vdash \varphi$ if and only if $p \Vdash \varphi$.*

Proof: The proof procedes by an induction on formulas. For a maximal element p of R the clauses in the definition of $p \Vdash \varphi$ coincide with those in II.4.3 for $\mathcal{C}(p) \vDash \varphi$. In clauses (ii), (iii) and (iv) the dependence on future states of knowledge reduces simply to the classical situation at p. Consider, for example, clause (ii): $p \Vdash \varphi \to \psi \Leftrightarrow (\forall q \geq p)(q \Vdash \varphi$ implies $q \Vdash \psi)$. Since p is maximal in R, $q \geq p$ is the same as $q = p$. Thus clause (ii) reduces to ($p \Vdash \varphi \to \psi$ iff $p \Vdash \varphi$ implies $p \Vdash \psi$) which is the analog as the corresponding clause, II.4.3(v), for classical implication. We leave the verification that all the other clauses are also equivalent as Exercise 8. □

Theorem 2.6: *Any intuitionistically valid sentence is classically valid.*

Proof: By the degeneracy lemma (Lemma 2.5), every classical model is a frame model with a one-element partially ordered set in which forcing and classical truth are equivalent. As a sentence is classically valid if true in all classical models, it is valid if forced in every frame. □

It remains to see which classically valid sentences are intuitionistically valid and which are not. We show how to verify that some classically valid sentences are not intuitionistically valid by constructing frame counterexamples. Before presenting the examples, we want to establish some notational conventions for displaying frames. All the examples below will have orderings which are suborderings of the full binary tree. We can therefore view the associated frames as labeled binary trees with the label of a node p being the structure $\mathcal{C}(p)$, or equivalently, the pair consisting of $C(p)$ and $A(p)$. We will thus draw frames as labeled binary trees in our usual style and display the labels in the form $\langle\, C(p), A(p)\,\rangle$. The theoretical development of tableaux and the proof of their completeness will require somewhat more general trees but we leave that for the next section.

In the examples below of sentences which are not intuitionistically valid (2.7–2.11), φ and ψ will denote atomic formulas of $\mathcal{L}$ with no free variables or only x free as displayed. In each of these examples, $\mathcal{C}(\emptyset)$, the structure associated with the bottom node $\emptyset$ of our partial ordering, will be C with all the constants of $\mathcal{L}$ interpreted as c. We begin with the archetypal classically valid sentence which is not intuitionistically valid.

Example 2.7: As expected, the sentence $\varphi \vee \neg\varphi$ is not intuitionistically valid. Let the frame $\mathcal{C}$ be

$$\begin{tikzcd}
\langle C, \{\varphi\} \rangle \arrow[d, dash] \\
\langle C, \emptyset \rangle
\end{tikzcd}$$

(Thus we have taken C as the domain at both nodes, $\emptyset$ and 0, of the frame.) At the bottom node, no atomic facts are true, i.e., $A(\emptyset)$ is empty. At the upper node 0, we have made the single atomic fact φ true by setting $A(0) = \{\varphi\}$).

Consider now whether or not $\emptyset \Vdash \varphi \vee \neg\varphi$. Certainly $\emptyset$ does not force φ since φ is atomic and not true in $\mathcal{C}(\emptyset)$, i.e., not in $A(\emptyset)$. On the other hand, $0 \Vdash \varphi$ since $\varphi \in A(0)$. Thus $\emptyset$ does not force $\neg\varphi$ since it has an extension 0 forcing φ. So by definition, $\emptyset$ does not force $\varphi \vee \neg\varphi$ and this sentence is not intuitionistically valid.

Example 2.8: The sentence $(\neg\varphi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \varphi)$ is not intuitionistically valid. Let the frame $\mathcal{C}$ be

$$\begin{tikzcd}
\langle C, \{\varphi, \psi\} \rangle \arrow[d, dash] \\
\langle C, \{\psi\} \rangle
\end{tikzcd}$$

Suppose, for the sake of a contradiction, that $\emptyset \Vdash (\neg\varphi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \varphi)$. Then $\emptyset \Vdash (\neg\varphi \rightarrow \neg\psi)$ would imply $\emptyset \Vdash (\psi \rightarrow \varphi)$ by clause (ii) of the definition of forcing (Definition 2.2). Now by clause (iii) of the definition, neither $\emptyset$ nor 0 forces $\neg\varphi$ since φ is in $A(0)$ and so forced at 0. Thus we see that $\emptyset$ does in fact force $(\neg\varphi \rightarrow \neg\psi)$ by applying clause (ii) again and the fact that $\emptyset$ and 0 are the only elements $\geq \emptyset$. On the other hand, $\emptyset$ does not force $(\psi \rightarrow \varphi)$ because $\emptyset$ forces ψ but not φ and so we have our desired contradiction.

Example 2.9: The sentence $(\varphi \rightarrow \psi) \vee (\psi \rightarrow \varphi)$ is not intuitionistically valid. Let the frame $\mathcal{C}$ be

$$\begin{tikzcd}
\langle C, \{\varphi\} \rangle \arrow[dr, dash] & & \langle C, \{\psi\} \rangle \arrow[dl, dash] \\
& \langle C, \emptyset \rangle &
\end{tikzcd}$$

In this frame, $\emptyset$ forces neither φ nor ψ, 0 forces φ but not ψ and 1 forces ψ but not φ. Since there is a node above $\emptyset$, namely 0, which forces φ but not ψ, $\emptyset$ does not force $\varphi \rightarrow \psi$. Similarly, $\emptyset$ does not force $\psi \rightarrow \varphi$. So $\emptyset$ does not force $(\varphi \rightarrow \psi) \vee (\psi \rightarrow \varphi)$.

Example 2.10: The sentence $\neg(\forall x)\varphi(x) \rightarrow (\exists x)\neg\varphi(x)$ is not intuitionistically valid. Let b be anything other than the sole element c of C. Let the frame $\mathcal{C}$ be

$$\begin{tikzcd}
\langle \{b, c\}, \{\varphi(c)\} \rangle \arrow[d, dash] \\
\langle C, \emptyset \rangle
\end{tikzcd}$$

Now by clause (iv) of Definition 2.2, neither $\emptyset$ nor 0 forces $(\forall x)\varphi(x)$ since $b \in C(0)$ but 0 does not force $\varphi(b)$. Thus $\emptyset \Vdash \neg(\forall x)\varphi(x)$. If $\emptyset \Vdash \neg(\forall x)\varphi(x) \rightarrow (\exists x)\neg\varphi(x)$, as it would were our given sentence valid, then $\emptyset$ would also force $(\exists x)\neg\varphi(x)$. By clause (v) of the definition this can happen only if there is a $c \in C$ such that $\emptyset \Vdash \neg\varphi(c)$. As c is the only element of C and $0 \Vdash \varphi(c)$, $\emptyset$ does not force $(\exists x)\neg\varphi(x)$.

Example 2.11: The sentence $(\forall x)(\varphi \vee \psi(x)) \rightarrow \varphi \vee (\forall x)\psi(x)$ is not intuitionistically valid. The required frame is

$$\begin{tikzcd}
\langle \{b, c\}, \{\psi(c), \varphi\} \rangle \arrow[d, dash] \\
\langle C, \{\psi(c)\} \rangle
\end{tikzcd}$$

We first claim that $\emptyset \Vdash (\forall x)(\varphi \vee \psi(x))$. As $\emptyset \Vdash \psi(c)$ and $0 \Vdash \varphi$, combining the clauses for disjunction (vii) and universal quantification (iv) we see that $\emptyset \Vdash (\forall x)(\varphi \vee \psi(x))$ as claimed. Suppose now for the sake of a contradiction that $\emptyset \Vdash (\forall x)(\varphi \vee \psi(x)) \rightarrow \varphi \vee (\forall x)\psi(x)$. We would then have that $\emptyset \Vdash \varphi \vee (\forall x)\psi(x)$. However, $\emptyset$ does not force φ and, as 0 does not force $\psi(b)$, $\emptyset$ does not force $(\forall x)\psi(x)$ either. Thus $\emptyset$ does not force the disjunction $\varphi \vee (\forall x)\psi(x)$, so we have the desired contradiction.

We would now like to give some examples of intuitionistically valid sentences whose validity can be verified directly using the definition of forcing. Before presenting the examples, we will prove a few basic facts about the forcing relation that will be useful for these verifications as well as future arguments. The first is perhaps the single most useful fact about forcing. It expresses the stability of forcing as one moves up in the partial ordering.

Lemma 2.12 (Monotonicity Lemma): *For every sentence φ of $\mathcal{L}$ and every $p, q \in R$, if $p \Vdash \varphi$ and $q \geq p$, then $q \Vdash \varphi$.*

Proof: We prove the lemma by induction on the logical complexity of φ. The inductive hypothesis is not needed to verify the conclusion that $q \Vdash \varphi$ for clauses (i), (ii), (iii) and (iv). The first follows immediately from the definition of a frame and clause (i) itself which defines forcing for atomic sentences. The other clauses define the meaning of (intuitionistic) implication, negation, and universal quantification precisely so as to make this lemma work. We use the induction hypothesis in the verifications of clauses (v), (vi) and (vii) which define forcing for the existential quantifier, conjunction and disjunction respectively.

(i) If φ is atomic and $p \Vdash \varphi$, then φ is in $A(p)$. By the definition of a frame, however, $A(p) \subseteq A(q)$, and so φ is in $A(q)$. Thus, by definition, $q \Vdash \varphi$.

(ii) Suppose $p \Vdash \varphi \to \psi$ and $q \geq p$. We show that $q \Vdash \varphi \to \psi$ by showing that if $r \geq q$ and $r \Vdash \varphi$ then $r \Vdash \psi$. Now $r \geq p$ by transitivity and so our assumptions that $p \Vdash \varphi \to \psi$ and $r \Vdash \varphi$ imply that $r \Vdash \psi$, as required.

(iii) Suppose $p \Vdash \neg\varphi$ and $q \geq p$. We show that $q \Vdash \neg\varphi$ by showing that if $r \geq q$ then r does not force φ. Again by transitivity, $r \geq p$. The definition of $p \Vdash \neg\varphi$ then implies that r does not force φ.

(iv) Suppose $p \Vdash (\forall x)\varphi(x)$ and $q \geq p$. We show that $q \Vdash (\forall x)\varphi(x)$ by showing that, for any $r \geq q$ and any $c \in C(r)$, $r \Vdash \varphi(c)$. Again, $r \geq p$ by transitivity. The definition of $p \Vdash (\forall x)\varphi(x)$ then implies that for any c in $C(r)$, $r \Vdash \varphi(c)$.

(v) Suppose $p \Vdash (\exists x)A(x)$ and $q \geq p$. Then by the definition of forcing there is a c in $C(p)$ such that $p \Vdash \varphi(c)$. By the inductive hypothesis, $q \geq p$ and $p \Vdash \varphi(c)$ imply that $q \Vdash \varphi(c)$. Thus $q \Vdash (\exists x)\varphi(x)$.

(vi) Suppose $p \Vdash (\varphi \wedge \psi)$ and $q \geq p$. Then by the definition of forcing $p \Vdash \varphi$ and $p \Vdash \psi$. By the inductive hypothesis, $q \Vdash \varphi$ and $q \Vdash \psi$. Thus $q \Vdash (\varphi \wedge \psi)$.

(vii) Suppose $p \Vdash (\varphi \vee \psi)$, and $q \geq p$. Then by the definition of forcing either $p \Vdash \varphi$ or $p \Vdash \psi$. By the inductive hypothesis, we get that either $q \Vdash \varphi$ or $q \Vdash \psi$. By the definition of forcing a disjunction, this says that $q \Vdash (\varphi \vee \psi)$. □

Monotonicity says that the addition of new atomic sentences at later states of knowledge q will not change forcing at earlier states of knowledge. This monotone character distinguishes "truth" in an intuitionistic frame from "truth" in "nonmonotonic logics", as discussed in III.7. In those logics, sentences forced at state of knowledge p need not be forced at states of knowledge $q > p$. In frames, as time evolves, we learn new "facts" but never discover that old ones are false.

Lemma 2.13 (Double Negation Lemma): *$p \Vdash \neg\neg\varphi$ if and only if for any $q \geq p$ there is an $r \geq q$ such that $r \Vdash \varphi$.*

Proof: $p \Vdash \neg\neg\varphi$ if and only if every $q \geq p$ fails to force $\neg\varphi$, or equivalently, if and only if every $q \geq p$ has an $r \geq q$ forcing φ. □

Lemma 2.14 (Weak Quantifier Lemma):

(i) *$p \Vdash \neg(\exists x)\neg\varphi(x)$ if and only if for all $q \geq p$ and for all $c \in C(q)$ there is an $r \geq q$ such that $r \Vdash \varphi(c)$.*

(ii) *$p \Vdash \neg(\forall x)\neg\varphi(x)$ if and only if for all $q \geq p$, there exists an $s \geq q$ and a $c \in C(s)$ such that $s \Vdash \varphi(c)$.*

Proof:

(i) This claim follows immediately from the definition.

(ii) $q \Vdash (\forall x)\neg\varphi(x)$ if and only if for all $r \geq q$ and all $c \in C(r)$ there is no $s \geq r$ such that $s \Vdash \varphi(c)$. Thus q does not force $(\forall x)\neg\varphi(x)$ if and only if there is an $r \geq q$ and a $c \in C(r)$ such that for some $s \geq r$, $s \Vdash \varphi(c)$. So $p \Vdash \neg(\forall x)\neg\varphi(x)$ if and only if for all $q \geq p$, there is an $r \geq q$ and a $c \in C(r)$ such that for some $s \geq r$, $s \Vdash \varphi(c)$. By transitivity $s \geq q$ and c is in $C(s)$ as required in the claim. □

We now produce the promised examples of intuitionistic validity. In the following examples (2.15–2.19) φ are ψ are arbitrary sentences.

Example 2.15: $\varphi \to \neg\neg\varphi$ is intuitionistically valid. To see that any p forces $\varphi \to \neg\neg\varphi$ we assume that $q \geq p$ and $q \Vdash \varphi$. We must show that $q \Vdash \neg\neg\varphi$. By the double negation lemma, it suffices to show that for every $r \geq q$ there is an $s \geq r$ such that $s \Vdash \varphi$. By the monotonicity lemma $r \Vdash \varphi$, and so r is the required s.

Example 2.16: $\neg(\varphi \wedge \neg\varphi)$ is intuitionistically valid. To show that any p forces $\neg(\varphi\wedge\neg\varphi)$ we need to show that no $q \geq p$ forces $\varphi\wedge\neg\varphi$, or equivalently no $q \geq p$ forces both φ and $\neg\varphi$. Suppose then that q forces both φ and $\neg\varphi$. Now $q \Vdash \neg\varphi$ means no $r \geq q$ forces φ. Since $q \geq q$, we have both q forces φ and q does not force φ for the desired contradiction.

Example 2.17: $(\exists x)\neg\varphi(x) \to \neg(\forall x)\varphi(x)$ is intuitionistically valid. To see that any p forces $(\exists x)\neg\varphi(x) \to \neg(\forall x)\varphi(x)$, we need to show that if $q \geq p$ and $q \Vdash (\exists x)\neg\varphi(x)$, then $q \Vdash \neg(\forall x)\varphi(x)$. Now $q \Vdash (\exists x)\neg\varphi(x)$ says there is a c in $C(q)$ such that $q \Vdash \neg\varphi(c)$. By monotonicity, any $r \geq q$ forces $\neg\varphi(c)$ as well, so no such r forces $(\forall x)\varphi(x)$, thus $q \Vdash \neg(\forall x)\varphi(x)$. This example should be compared with its contrapositive (Example 2.10) which is classically but not intuitionistically valid.

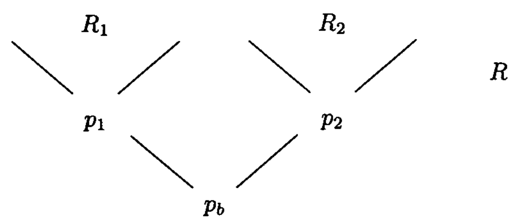
Example 2.18: $\neg(\exists x)\varphi(x) \to (\forall x)\neg\varphi(x)$ is intuitionistically valid. To see that any p forces $\neg(\exists x)\varphi(x) \to (\forall x)\neg\varphi(x)$ we have to show that for any $q \geq p$, if $q \Vdash \neg(\exists x)\varphi(x)$, then $q \Vdash (\forall x)\neg\varphi(x)$. Now $q \Vdash \neg(\exists x)\varphi(x)$ says that, for every $r \geq q$ and every c in $C(r)$, r does not force $\varphi(c)$. By transitivity $s \geq r$ implies $s \geq q$. So for every $r \geq q$ and every c in $C(r)$, no $s \geq r$ forces $\varphi(c)$. This says $q \Vdash (\forall x)\neg\varphi(x)$.

Example 2.19: If x is not free in φ, then $\varphi \vee (\forall x)\psi(x) \rightarrow (\forall x)(\varphi \vee \psi(x))$ is intuitionistically valid. To see that any p forces $\varphi \vee (\forall x)\psi(x) \rightarrow (\forall x)(\varphi \vee \psi(x))$ we must show that, for any $q \geq p$, if $q \Vdash \varphi$ or $q \Vdash (\forall x)\psi(x)$ then $q \Vdash (\forall x)(\varphi \vee \psi(x))$. There are two cases. If $q \Vdash \varphi$, then for any $r \geq q$ and any c in $C(r)$, $r \Vdash \varphi \vee \psi(c)$, so $q \Vdash (\forall x)(\varphi \vee \psi(x))$. If $q \Vdash (\forall x)\psi(x)$, then for all $r \geq q$ and all c in $C(r)$, $r \Vdash \psi(c)$, so $r \Vdash \varphi \vee \psi(c)$. This says that $q \Vdash (\forall x)(\varphi \vee \psi(x))$. This example should be compared with Example 2.11.

The frame definition of intuitionistic validity makes it remarkably simple to prove two important properties of intuitionistic logic which embody its constructivity: the disjunction and existence properties. The first says that, if a disjunction is valid, then one of its disjuncts is valid. The second says that, if an existential sentence of $\mathcal{L}$ is valid, then one of its instances via a constant from $\mathcal{L}$ is also valid. When we combine this with the completeness theorem for intuitionistic logic (Theorem 4.10), we will see that this means that if we can prove an existential sentence we can in fact prove some particular instance. Similarly, if we can prove a disjunction then we can prove one of the disjuncts.

Theorem 2.20 (Disjunction Property): *If $(\varphi_1 \vee \varphi_2)$ is intuitionistically valid then one of φ_1, φ_2 is intuitionistically valid.*

Proof: We prove the theorem by establishing its contrapositive. So suppose neither φ_1 nor φ_2 is intuitionistically valid. Thus there are, for $i = 1, 2$, frames $\mathcal{C}_i$ and elements p_i of the associated partial orderings R_i such that φ_1 is not forced by p_1 in $\mathcal{C}_1$ and φ_2 is not forced by p_2 in $\mathcal{C}_2$. By the restriction lemma (2.4), we may assume that p_i is the least element of R_i. Fix a constant c in $\mathcal{L}$. Simply by relabeling the elements of $C_i(p)$ and R_i we may assume that the interpretation of c in both $C_i(p_i)$ is the same, say d, and that the R_i are disjoint. Let R be the union of R_1, R_2, and $\{p_b\}$, with p_b not in either R_i. Make R into a partial order by ordering R_1 and R_2 as before and putting p_b below p_1 and p_2.



We define a frame $\mathcal{C}$ with this ordering on R by setting $C(p)$ equal to $C_i(p)$ for $p \in R_i$ and $C(p_b) = \{d\}$ with $A(p_b) = \emptyset$. In this frame $\mathcal{C}$, p_1 does not force φ_1 by the restriction lemma (2.4). Thus, p_b does not force φ_1 by the monotonicity lemma (2.21). Similarly, p_b does not force φ_2 as p_2 does not. Thus p_b does not force $\varphi_1 \vee \varphi_2$; hence $\varphi_1 \vee \varphi_2$ is not intuitionistically valid: contradiction. □

Theorem 2.21 (Existence Property): *If $(\exists x)\varphi(x)$ is an intuitionistically valid sentence of a language $\mathcal{L}$, then for some constant c in $\mathcal{L}$, $\varphi(c)$ is also intuitionistically valid. (Remember that, by convention, $\mathcal{L}$ has at least one constant.)*

Proof: Suppose that, for each constant a in $\mathcal{L}$, $\varphi(a)$ is not intuitionistically valid. Then, for each such constant, there is an $\mathcal{L}$–frame $\mathcal{C}_a$ with a partially ordered set R_a containing an element p_a which does not force $\varphi(a)$. As in the previous proof, we may, without loss of generality, assume that p_a is the least element of R_a and all the R_a's are pairwise disjoint. We also assume that the interpretation of some fixed constant c of $\mathcal{L}$ is the same element d in every $C(p_a)$. We now form a new partial ordering R by taking the union of all R_a and the union of the partial orders and adding on a new bottom element p_b under all the p_a. We next define an $\mathcal{L}$–frame associated with R, as in the previous proof, by letting $C(p_b) = \{d\}$, $A(p_b) = \emptyset$ and $C(p) = C_a(p)$ for every $p \in R_a$ and every constant a of $\mathcal{L}$. We can now imitate the argument in Theorem 2.20. As we are assuming that $\exists x\varphi(x)$ is intuitionistically valid, we must have $p_b \Vdash_{\mathcal{C}} \exists x\varphi(x)$. Then, by definition, $p_b \Vdash \varphi(a)$ for some constant a in $\mathcal{L}$. Applying first the monotonicity lemma and then the restriction lemma we would have p_a forcing $\varphi(a)$ first in $\mathcal{C}$ and then in $\mathcal{C}_a$; this contradicts our initial hypothesis that p_a and $\mathcal{C}_a$ show that $\varphi(a)$ is not intuitionistically valid. □

Exercises

Sentences (1)–(6) below are classically valid. Verify that they are intuitionistically valid by direct arguments with frames. Remember that $\varphi \leftrightarrow \psi$ is an abbreviation for $(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$.

1. $\neg\varphi \leftrightarrow \neg\neg\neg\varphi$
2. $(\varphi \wedge \neg\psi) \rightarrow \neg(\varphi \rightarrow \psi)$
3. $(\varphi \rightarrow \psi) \rightarrow (\neg\neg\varphi \rightarrow \neg\neg\psi)$
4. $(\neg\neg(\varphi \rightarrow \psi) \leftrightarrow (\neg\neg\varphi \rightarrow \neg\neg\psi)$
5. $\neg\neg(\varphi \wedge \psi) \leftrightarrow (\neg\neg\varphi \wedge \neg\neg\psi)$
6. $\neg\neg(\forall x)\varphi(x) \rightarrow (\forall x)\neg\neg\varphi(x)$
7. Supply the proof for Lemma 2.4.
8. Supply the proofs for the remaining cases of Lemma 2.5.
9. Let K be the set of constants occurring in $(\exists x)\varphi(x)$ and suppose that $(\exists x)\varphi(x)$ is intuitionistically valid. Show that, if K is nonempty, then for some c in K, $\varphi(c)$ is intuitionistically valid. (Hint: Define the restriction of a given frame $\mathcal{C}$ for a language $\mathcal{L}$ to one $\mathcal{C}'$ for a given restriction $\mathcal{L}'$ of $\mathcal{L}$. Now prove that, for any sentence φ of $\mathcal{L}'$ and any element p of the appropriate partial ordering, $p \Vdash_{\mathcal{C}} \varphi$ if and only if $p \Vdash_{\mathcal{C}'} \varphi$.)

10. In case K is empty in the previous exercise, show that $\varphi(c)$ is intuitionistically valid for any constant c. (Hint: For any constants a and c of $\mathcal{L}$ define a map Θ on formulas of $\mathcal{L}$ and on the frames for $\mathcal{L}$ which interchanges a and c. Prove that, for every $\mathcal{C}$ and p, $p \Vdash_{\mathcal{C}} \varphi$ if and only if $p \Vdash_{\Theta(\mathcal{C})} \Theta(\varphi)$.)

3. Intuitionistic Tableaux

We will describe a proof procedure for intuitionistic logic based on a tableau style system like that used for classical logic in II.6. In classical logic, the idea of a tableau proof is to systematically search for a structure agreeing with the starting signed sentence. We either get such a structure or see that each possible analysis leads to a contradiction. When we begin with a signed sentence $F\varphi$, we thus either find a structure in which φ fails or decide that we have a proof of φ. For intuitionistic logic we instead begin with a *signed forcing assertion* $Tp \Vdash \varphi$ or $Fp \Vdash \varphi$ (φ is again a sentence) and try to either build a frame agreeing with the assertion or decide that any such attempt leads to a contradiction. If we begin with $Fp \Vdash \varphi$, we either find a frame in which p does not force φ or decide that we have an intuitionistic proof of φ.

There are many possible variants on the tableau method suitable for intuitionistic propositional and predicate logic due to Kripke, Hughes and Cresswell, Fitting, and others. The one we choose is designed to precisely match our definition of frame so that the systematic tableau will represent a systematic search for a frame agreeing with the starting signed forcing assertion. It is a variant of Fitting's [1983, 4.1] prefixed tableau.

The definitions of tableau and tableau proof for intuitionistic logic are formally very much like those of II.6 for classical logic. *Intuitionistic tableaux* and *tableau proofs* are labeled binary trees. The labels (again called the *entries of the tableau*) are now *signed forcing assertions*, i.e., labels of the form $Tp \Vdash \varphi$ or $Fp \Vdash \varphi$ for φ a sentence of any appropriate language. We read $Tp \Vdash \varphi$ as p forces φ and $Fp \Vdash \varphi$ as p does not force φ.

In classical logic, the elements of the structure we built by developing a tableau were the constant symbols appearing on some path of the tableau. We are now attempting to build an entire frame. The p's and q's appearing in the entries of some path P through our intuitionistic tableau will constitute the elements of the partial ordering for the frame. The ordering on them will also be specified as part of the development of the tableau. As in the classical case, we always build a tableau based on a language expanded from the one for the starting signed assertion by adding on new constants $c_0, c_1, \ldots$. The constants appearing in the sentences φ of entries on P of the form $Tq \Vdash \varphi$ or $Fq \Vdash \varphi$ for $q \leq p$ will be the elements of the required domains $C(p)$. (We use the entries with $q \leq p$ so as to ensure the monotonicity required for domains in the definition of a frame.)

With this motivation in mind, we can specify the *atomic intuitionistic tableaux*.

Definition 3.1: We begin by fixing a language $\mathcal{L}$ and an expansion $\mathcal{L}_C$ given by adding new constant symbols c_i for $i \in \mathbb{N}$. We list in Figure 55 the atomic intuitionistic tableaux (for the language $\mathcal{L}$). In the tableaux in this list, φ and ψ, if unquantified, are any sentences in the language $\mathcal{L}_C$. If quantified, they are formulas in which only x is free.

Formally, the precise meaning of "new c" and "new p" will be defined along with the definition of intuitionistic tableau. The intention for the constants is essentially as in the classical case: When we develop $T\forall x\varphi(x)$, we can put in any c for x and add on $T\varphi(c)$ but, when we develop $\exists x\varphi(x)$ by adding $T\varphi(c)$ on to the tableau, we can only use a c for which no previous commitments have been made. One warning is necessary here. When we say "any appropriate c" we mean any c in the appropriate language. In the classical case that meant any c in $\mathcal{L}_C$. Here, in developing $Tp \Vdash \forall x\varphi(x)$ as in (T$\forall$) above, it will mean any c in $\mathcal{L}$ or appearing on the path so far in a forcing assertion about a $q \leq p$. These restrictions correspond to our intention to define $C(p)$ in accordance with the requirement in the definition of frame that $C(q) \subseteq C(p)$ for $q \leq p$. Technically, similar considerations could be applied to the use of a new c as in (T$\exists$) although as a practical matter we can always choose c from among the c_i in $\mathcal{L}_C$ which have not yet appeared anywhere in the tableau. We will in fact incorporate such a choice into our formal definition.

The restrictions on the elements p introduced into the ordering should also be understood in terms of the definition of frames. In (TAt), for example, we follow the requirement in the definition of a frame that $A(p) \subseteq A(p')$ if $p \leq p'$. The reader should also keep in mind that we are determining the elements p of the partial ordering for our frame as well as defining the ordering itself as we develop the tableau. Thus, for example, when developing $Tp \Vdash \neg\varphi$ we can, in accordance with the definition of forcing a negation, add on $Fp' \Vdash \varphi$ for any $p' \geq p$ which appears on the path so far. On the other hand, if we wish to assert that p does not force $\neg\varphi$, i.e., $Fp \Vdash \neg\varphi$, then the definition of forcing tells us that there must be some $p' \geq p$ which does force φ. As with putting in a new constant, we cannot use a p' for which other commitments have already been made. Thus we can develop $Fp \Vdash \neg\varphi$ as in (F$\neg$) by adding on $Tp' \Vdash \varphi$ for a new element p' of the ordering of which we can only say that it is bigger than p. Thus, we require that p' is larger than p (and so by the requirement of transitivity bigger than any $q \leq p$) but that p' is incomparable with all other elements of the ordering introduced so far. (Again, technically, we only need to worry about the relation between p' and the q appearing on the branch so far. It is simpler to just take an entirely new p', i.e., one not yet appearing anywhere in the tableau. It will then automatically be true that $p \leq q$ only if p and q are on the same path through the tableau.)

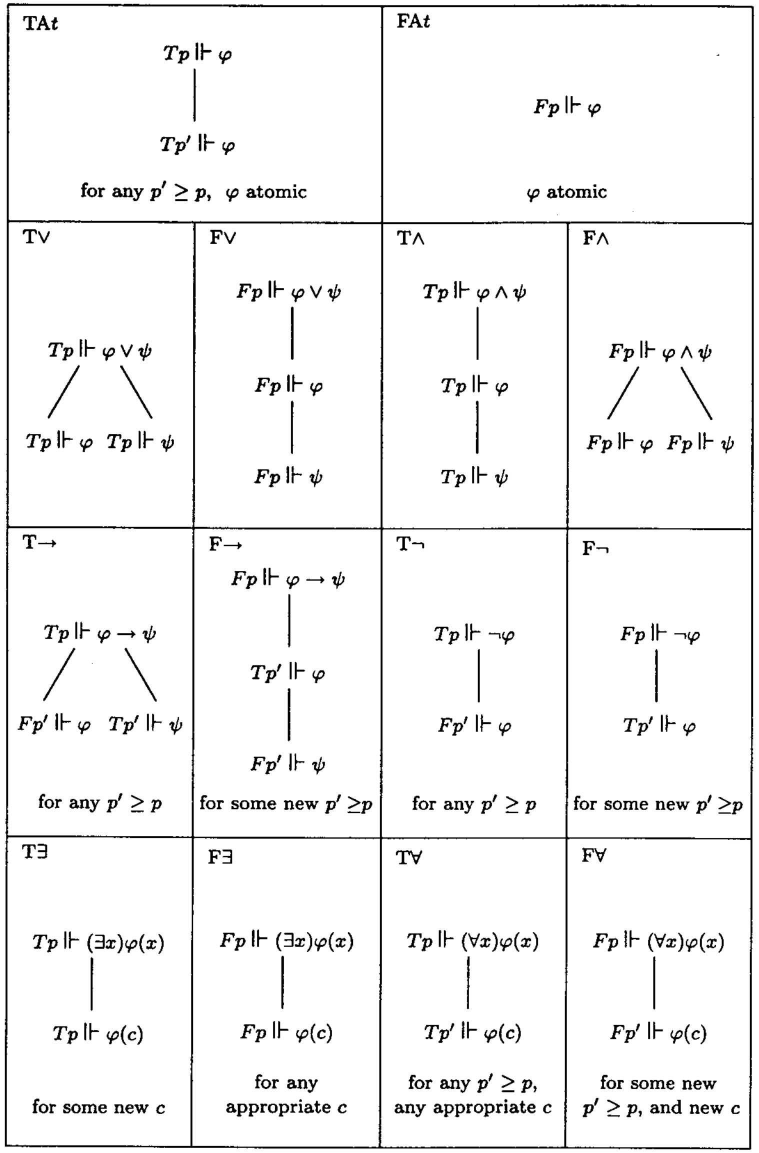


FIGURE 55

The formal definitions of tableaux and tableau proof for intuitionistic logic could perhaps even be left as an exercise. As it would be an exercise with many pitfalls for the unwary we give them in full detail.

Definition 3.2: We continue to use our fixed language $\mathcal{L}$ and extension by constants $\mathcal{L}_C$. We also fix a set $S = \{p_i : i \in \mathbb{N}\}$ of potential candidates for the p's and q's in our forcing assertions. An ***intuitionistic tableau*** (for $\mathcal{L}$) is a binary tree labeled with signed forcing assertions which are called the ***entries*** of the tableau. The class of all intuitionistic tableaux (for $\mathcal{L}$) is defined by induction. We simultaneously define, for each tableau τ, an ordering $\leq_\tau$ the elements of S appearing in τ.

(i) Each atomic tableau τ is a tableau. The requirement that c be new in cases (T∃) and (F∀) here simply means that c is one of the constants c_i added on to $\mathcal{L}$ to get $\mathcal{L}_C$ which does not appear in φ. The phrase "any c" in (F∃) and (T∀) means any constant in $\mathcal{L}$ or in φ. The requirement that p' be new in (F→), (F¬) and (F∀) here means that p' is any of the p_i other than p. We also declare p' to be larger than p in the associated ordering. The phrase "any $p' \geq p$" in (T→), (T¬), (T∀) and (TAt) in this case simply means that p' is p. (Of course we always declare $p \leq p$ for every p in every ordering we define.)

(ii) If τ is a finite tableau, P a path on τ, E an entry of τ occurring on P and τ' is obtained from τ by adjoining an atomic tableau with root entry E to τ at the end of the path P then τ' is also a tableau. The ordering $\leq_{\tau'}$ agrees with $\leq_\tau$ on the p_i appearing in τ. Its behavior on any new element is defined below when we explain the meaning of the restrictions on p' in the atomic tableaux for cases (F→), (F¬) and (F∀).

The requirement that c be new in cases (T∃) and (F∀) here means that it is one of the c_i (and so not in $\mathcal{L}$) which do not appear in any entry on τ. The phrase "any c" in (F∃) and (T∀) here means any c in $\mathcal{L}$ or appearing in an entry on P of the form $Tq \Vdash \psi$ or $Fq \Vdash \psi$ with $q \leq_\tau p$.

In (F→), (F¬) and (F∀) the requirement that $p' \geq p$ be new means that we choose a p_i not appearing in τ as p' and we declare that it is larger than p in $\leq_{\tau'}$. (Of course we insure transitivity by declaring that $q \leq_\tau p'$ for every $q \leq_\tau p$.) The phrase "any $p' \geq p$" in (T→), (T¬), (T∀) and (TAt) means we can choose any p' which appears in an entry on P and has already been declared greater than or equal to p in $\leq_\tau$.

(iii) If $\tau_0, \tau_1, \ldots, \tau_n, \ldots$ is a sequence of finite tableaux such that, for every $n \geq 0$, τ_{n+1} is constructed from τ_n by an application of (ii) then $\tau = \cup\tau_n$ is also a tableau.

As in predicate logic, we insist that the entry E in clause (ii) formally be repeated when the corresponding atomic tableau is added on to P. This is again crucial to the properties corresponding to a classical tableau being finished. In our examples below, however, we will typically omit them purely as a notational convenience.

Note that if we do not declare that either $p \leq p'$ or $p' \leq p$ in our definition of $\leq_\tau$ then p and p' are incomparable in $\leq_\tau$.

We make good on our previous remark about the relation of the ordering $\leq_\tau$ to paths through the tableau τ with the following lemma.

Lemma 3.3: *For any intuitionistic tableau τ with associated ordering $\leq_\tau$, if $p' \leq_\tau p$ then p and p' both appear on some common path through τ.*

Proof: The proof proceeds by an induction on the definition of τ and $\leq_\tau$. We leave it as Exercise 31. $\square$

Definition 3.4 (*Intuitionistic Tableau Proofs*): Let τ be a intuitionistic tableau and P a path in τ.

(i) P is *contradictory* if, for some forcing assertion $p \Vdash \varphi$, both $Tp \Vdash \varphi$ and $Fp \Vdash \varphi$ appear as entries on P.

(ii) τ is *contradictory* if every path through τ is contradictory.

(iii) τ is an *intuitionistic proof of* φ if τ is a finite contradictory intuitionistic tableau with its root node labeled $Fp \Vdash \varphi$ for some $p \in R$. φ is *intuitionistically provable*, $\vdash \varphi$, if there is an intuitionistic proof of φ.

Note that, as in classical logic, if there is any contradictory tableau with root node $Fp \Vdash \varphi$, then there is one which is finite, i.e., a proof of φ : Just terminate each path when it becomes contradictory. As each path is now finite, the whole tree is finite by König's lemma. Thus, the added requirement that proofs be finite (tableaux) has no affect on the existence of proofs for any sentence. Another point of view is that we could have required the path P in clause (ii) of the definition of tableaux be noncontradictory without affecting the existence of proofs. Thus, in practice, when attempting to construct proofs we mark any contradictory path with the symbol $\otimes$ and terminate the development of the tableau along that path.

Before dealing with the soundness and completeness of the tableau method for intuitionistic logic we look at some examples of intuitionistic tableau proofs. Remember that we are abbreviating the tableaux by not repeating the entry which we are developing. We also number the levels of the tableau on the left and indicate on the right the level of the atomic tableau whose development produced the line. In all our examples, the set S from which we choose the domain of our partial order will be the set of finite binary sequences. The declarations of ordering relations are dictated by the atomic tableau added on at each step and so can also be omitted. In fact, we will always choose our p's and q's so as to define our orderings to agree with the usual ordering of inclusion on binary sequences.

Example 3.5: Let φ and ψ be any atomic sentences of $\mathcal{L}$. Figure 56 provides an intuitionistic proof of $\varphi \rightarrow (\psi \rightarrow \varphi)$.

In this proof the first three lines are an instance of (F$\rightarrow$) from the list of atomic tableaux. Lines 4 and 5 are introduced by developing line 3 in accordance with (F$\rightarrow$) again. Line 6, which, together with line 5, provides our contradiction, follows from line 2 by atomic tableau (T*At*).

1	$F\emptyset \Vdash \varphi \rightarrow (\psi \rightarrow \varphi)$	
2	$T0 \Vdash \varphi$	by 1
3	$F0 \Vdash \psi \rightarrow \varphi$	by 1
4	$T00 \Vdash \psi$	by 3
5	$F00 \Vdash \varphi$	by 3
6	$T00 \Vdash \psi$	by 2
	$\otimes$	

FIGURE 56

Example 3.6: Any sentence of $\mathcal{L}$ of the following form is intuitionistically provable:

$$(\exists x)(\varphi(x) \vee \psi(x)) \rightarrow (\exists x)\varphi(x) \vee (\exists x)\psi(x)$$

In this proof (Figure 57) the first three lines are an instance of (F$\rightarrow$). Line 4 follows by applying (T$\exists$) to line 2. Lines 5 and 6 follow by applying (F$\vee$) to line 3. Lines 7 and 8 are applications of (F$\vee$) to lines 5 and 6 respectively. Line 9, which supplies the contradictions to lines 7 or 8 on its two branches, is an application of (T$\vee$) to line 4.

Example 3.7: Consider $(\forall x)(\varphi(x) \wedge \psi(x)) \rightarrow (\forall x)\varphi(x) \wedge (\forall x)\psi(x)$.

Note here (Figure 58) that we develop both sides of the branching at line 4 and write the parallel developments side by side. Also of note is the use of (T$\forall$) applied to line 2 to get line 6. We took advantage of the ability to chose both the constants c and d and the elements of the ordering 00 and 01.

Exercises

Let φ and ψ be any atomic formulas either with no free variables or with only x free as appropriate. For each sentence θ in (1)–(30) below, construct a tableau starting with $F\emptyset \Vdash \theta$ to show that the classically valid θ is also intuitionistically provable. (Remember that $\varphi \leftrightarrow \psi$ is an abbreviation for $(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$).

1	$F\emptyset \Vdash (\exists x)(\varphi(x) \vee \psi(x)) \to (\exists x)\varphi(x) \vee (\exists x)\psi(x)$		
2	$T0 \Vdash (\exists x)(\varphi(x) \vee \psi(x))$		by 1
3	$F0 \Vdash (\exists x)\varphi(x) \vee (\exists x)\psi(x)$		by 1
4	$T0 \Vdash \varphi(c) \vee \psi(c)$		by 2
5	$F0 \Vdash (\exists x)\varphi(x)$		by 3
6	$F0 \Vdash (\exists x)\psi(x)$		by 3
7	$F0 \Vdash \varphi(c)$		by 5
8	$F0 \Vdash \psi(c)$		by 6
9	$T0 \Vdash \varphi(c)$	$T0 \Vdash \psi(c)$	by 4
	$\otimes$	$\otimes$	by 7, 8, 9

FIGURE 57

Distributive Lattice Laws

1. $(\varphi \vee \varphi) \leftrightarrow \varphi$
2. $(\varphi \wedge \varphi) \leftrightarrow \varphi$
3. $(\varphi \wedge \psi) \leftrightarrow (\psi \wedge \varphi)$
4. $(\varphi \vee \psi) \leftrightarrow (\psi \vee \varphi)$
5. $((\varphi \wedge \psi) \wedge \sigma) \leftrightarrow (\varphi \wedge (\psi \wedge \sigma))$
6. $((\varphi \vee \psi) \vee \sigma) \leftrightarrow (\varphi \vee (\psi \vee \sigma))$
7. $(\varphi \vee (\psi \wedge \sigma)) \leftrightarrow ((\varphi \vee \psi) \wedge (\varphi \vee \sigma))$
8. $(\varphi \wedge (\psi \vee \sigma)) \leftrightarrow ((\varphi \wedge \psi) \vee (\varphi \wedge \sigma))$

1	$F\emptyset \Vdash (\forall x)(\varphi(x) \wedge \psi(x)) \to (\forall x)\varphi(x) \wedge (\forall x)\psi(x)$		
2	$T0 \Vdash (\forall x)(\varphi(x) \wedge \psi(x))$		by 1
3	$F0 \Vdash (\forall x)\varphi(x) \wedge (\forall x)\psi(x)$		by 1
4	$F0 \Vdash (\forall x)\varphi(x)$	$F0 \Vdash (\forall x)\psi(x)$	by 3
5	$F00 \Vdash \varphi(c)$	$F01 \Vdash \psi(d)$	by 4
6	$T00 \Vdash \varphi(c) \wedge \psi(c)$	$T01 \Vdash \varphi(d) \wedge \psi(d)$	by 2
7	$T00 \Vdash \varphi(c)$	$T01 \Vdash \varphi(d)$	by 6
8	$T00 \Vdash \psi(c)$	$T01 \Vdash \psi(d)$	by 6
	$\otimes$	$\otimes$	by 5, 7 and 5, 8

FIGURE 58

Pure Implication Laws

9. $\varphi \to \varphi$
10. $\varphi \to (\psi \to \varphi)$
11. $(\varphi \to (\psi \to \sigma)) \to ((\varphi \to \psi) \to (\varphi \to \sigma))$

Introduction and Elimination of $\wedge$

12. $((\varphi \to (\psi \to \sigma)) \to ((\varphi \wedge \psi) \to \sigma)$
13. $((\varphi \wedge \psi) \to \sigma) \to ((\varphi \to (\psi \to \sigma))$
14. $(\neg\varphi \vee \psi) \to (\varphi \to \psi)$

De Morgan's Laws

15. $\neg(\varphi \vee \psi) \leftrightarrow (\neg\varphi \wedge \neg\psi)$
16. $(\neg\varphi \vee \neg\psi) \to \neg(\varphi \wedge \psi)$

Contrapositive

17. $(\varphi \to \psi) \to (\neg\psi \to \neg\varphi)$

Double Negation

18. $\varphi \to \neg\neg\varphi$

Contradiction

19. $\neg(\varphi \wedge \neg\varphi)$

Distributive Laws

20. $(\exists x)(\varphi(x) \vee \psi(x)) \leftrightarrow (\exists x)\varphi(x) \vee (\exists x)\psi(x)$
21. $((\forall x)(\varphi(x) \wedge \psi(x)) \leftrightarrow (\forall x)\varphi(x) \wedge (\forall x)\psi(x)$
22. $(\varphi \vee (\forall x)\psi(x)) \to (\forall x)(\varphi \vee \psi(x))$, x not free in φ
23. $(\varphi \wedge (\exists x)\psi(x)) \to (\exists x)(\varphi \wedge \psi(x))$, x not free in φ
24. $(\exists x)(\varphi \to \psi(x)) \to (\varphi \to (\exists x)\psi(x))$, x not free in φ
25. $(\exists x)(\varphi \wedge \psi(x)) \to (\varphi \wedge (\exists x)\psi(x))$, x not free in φ

De Morgan's Laws

26. $\neg(\exists x)\varphi(x) \to (\forall x)\neg\varphi(x)$
27. $(\forall x)\neg\varphi(x) \to \neg(\exists x)\varphi(x)$
28. $(\exists x)\neg\varphi(x) \to \neg(\forall x)\varphi(x)$
29. $(\exists x)(\varphi(x) \to \psi) \to ((\forall x)\varphi(x) \to \psi)$, x not free in ψ
30. $((\exists x)\varphi(x) \to \psi) \to (\forall x)(\varphi(x) \to \psi)$, x not free in ψ

31. Prove Lemma 3.3.
32. (**Theorem on Constants**) Prove the intuitionistic version of Exercise II.6.13: Let $\varphi(x_1, \ldots, x_n)$ be a formula with all free variables displayed and let $c_1, \ldots, c_n$ be constants not appearing in φ. Prove that there is an intuitionistic tableau proof of $\forall x_1 \ldots \forall x_n \varphi(x_1, \ldots, x_n)$ iff there is one of $\varphi(c_1, \ldots, c_n)$.

4. Soundness and Completeness

Our first goal in this section is to show that in intuitionistic logic (as in classical logic) provability implies validity. As in the classical soundness theorem (II.7.2), we begin by proving that a frame which "agrees" with the root node of a tableau "agrees" with every entry along some path of the tableau. In the classical case (Definition II.7.1) we constructed the path and defined a structure whose domain consists of the constants c occurring in the signed sentences along this path. Now, in addition to interpreting the constants occurring in the assertions along our path P in the appropriate structures $\mathcal{C}(p)$, we must "interpret" the partial ordering elements p occurring in the signed forcing assertions on P in the ordering R of the given frame.

Definition 4.1: Suppose $\mathcal{C} = (R, \leq_R, \mathcal{C}(p))$ is a frame for a language $\mathcal{L}$, τ is a tableau whose root is labeled with a forcing assertion about a sentence φ of $\mathcal{L}$ and P is a path through τ. Let S be the set of p's appearing in forcing assertions on P and let $\leq_S$ be the ordering on S defined in the construction of τ. We say that $\mathcal{C}$ *agrees with* P if there are maps f and g such that

(i) f is an order preserving (but not necessarily one–one) map from S into R.

(ii) g sends each constant c occurring in any sentence ψ of a forcing assertion $Tp \Vdash \psi$ or $Fp \Vdash \psi$ on P to a constant in $\mathcal{L}(f(p))$. Moreover, g is the identity on constants of $\mathcal{L}$. We also extend g to be a map on formulas in the obvious way: To get $g(\psi)$ simply replace every constant c in ψ by $g(c)$.

(iii) If $Tp \Vdash \psi$ is on P, then $f(p)$ forces $g(\psi)$ in $\mathcal{C}$ and if $Fp \Vdash \psi$ is on P, then $f(p)$ does not force $g(\psi)$ in $\mathcal{C}$.

Theorem 4.2: *Suppose $\mathcal{C} = (R, \leq_R, \mathcal{C}(p))$ is a frame for a language $\mathcal{L}$ and τ is a tableau whose root is labeled with a forcing assertion about a sentence φ of $\mathcal{L}$. If either*

(i) *$Fr \Vdash \varphi$ is at the root of τ and $q \in R$ does not force φ in $\mathcal{C}$*

or

(ii) *$Tr \Vdash \varphi$ is at the root of τ and $q \in R$ does force φ in $\mathcal{C}$,*

then there is a path P through τ which agrees with $\mathcal{C}$; moreover, there is a witness function f (as required in Definition 4.1) that sends r to q.

The proof of this theorem proceeds by induction on the construction of the tableau τ. Before providing the details we reformulate the result as the standard version of the soundness theorem.

Theorem 4.3 (Soundness): *If there is an intuitionistic tableau proof of a sentence φ, then φ is intuitionistically valid.*

Proof (of Soundness): An intuitionistic proof of φ is an intuitionistic tableau τ with a root of the form $Fr \Vdash \varphi$ in which every path is contradictory. If φ is not intuitionistically valid, then there is a frame $\mathcal{C} = (R, \leq_R, \mathcal{C}(p))$ and a $q \in R$ such that q does not force φ in $\mathcal{C}$. Now apply Theorem 4.2 to get a path P through τ and functions f and g with the properties listed in Definition 4.1. As τ is contradictory, there is a p and a sentence ψ such that both $Tp \Vdash \psi$ and $Fp \Vdash \psi$ occur on P. Definition 4.1(iii) then provides an immediate contradiction. □

We break the inductive proof of Theorem 4.2 into its component parts. First, there are fourteen base cases corresponding to clause (i) of Definition 3.2 and the fourteen atomic tableaux.

Lemma 4.4: *For each atomic tableau τ which satisfies the hypotheses of Theorem 4.2, there are P, f and g as required in its conclusion.*

There are then fourteen induction cases corresponding to the type of atomic tableau chosen for development in clause (ii) of Definition 3.2. We in fact prove an assertion somewhat stronger than the theorem to facilitate the induction.

Lemma 4.5: *If f and g witness that a path P of a tableau τ agrees with C and τ' is gotten from τ by an application of clause* (ii) *of Definition 3.2, then there are extensions P', f' and g' of P, f and g respectively such that f' and g' witness that the path P' through τ' also agrees with C.*

Theorem 4.2 is an easy consequence of these two lemmas so we present its proof before considering the proofs of the lemmas.

Proof (of Theorem 4.2): Lemma 4.4 establishes the theorem for atomic tableaux. Lemma 4.5 then proves the theorem for all finite tableaux by induction. In fact, it proves it for infinite tableaux as well. Suppose $\tau = \cup\tau_n$ is an infinite tableau as defined by clause (iii) of Definition 3.2. We begin by applying the appropriate case of Lemma 4.4 to τ_0 to get suitable P_0, f_0 and g_0. We then apply Lemma 4.5 to each τ_n in turn to construct P_n, f_n and g_n. The required P, f and g for τ are then simply the unions of the P_n, f_n and g_n respectively. $\square$

Proof (of Lemma 4.4): We begin by defining $f(p) = r$ and g to be the identity on the constants of $\mathcal{L}$. The argument now needed for each atomic tableau is precisely the same as the one for the corresponding case of Lemma 4.5. The point here is that, with this choice of f and g, the root itself agrees with C by the hypothesis of the theorem. The inductive argument applied to the rest of the atomic tableau then provides the required extensions. Thus, we have reduced the proof of this lemma to that of Lemma 4.5. $\square$

Proof (of Lemma 4.5): First, note that, if τ' is gotten by extending τ somewhere other than at the end of P, then the witnesses for τ work for τ' as well. Thus, we may assume that we form τ' by adjoining one of the atomic tableaux at the end of P in τ. We now consider the fourteen cases given by the atomic tableaux of Definition 3.1.

Cases (T$\vee$), (F$\vee$), (T$\wedge$), (F$\wedge$), (T$\rightarrow$), (T$\neg$), (F$\exists$), (T$\forall$) and (FAt) require no extension of f or g. In each of these cases it is obvious from the induction hypothesis and the corresponding case of the definition of forcing (Definition 2.2) that one of the extensions of P to a path through τ' satisfies the requirements of the lemma. Note that (TAt) also requires the monotonicity assumption on $A(p)$.

The arguments for the remaining cases are all illustrated by that for case (F$\forall$). Here the entry of P being developed is $Fp \Vdash (\forall x)\varphi(x)$. The required extension of P to P' is the only possible one. It is determined by adding $Fp' \Vdash \varphi(c)$ to the end of P. By our induction hypothesis, $f(p) \Vdash_C g((\forall x)\varphi(x))$. By the definition of forcing a universal sentence (2.2(iv)), there is a $q' \in R$ and a $c' \in \mathcal{L}(q')$ such that $q' \geq f(p)$ and q' does not force $g(\varphi(c'))$. Fix such q' and c' and extend f and g to f' and g' by setting $f'(p') = q'$and $g'(c) = c'$. It is now obvious that P', f' and g' satisfy the requirements of the lemma, i.e., f' and g' witness that P' agrees with C.

We leave cases (F$\rightarrow$), (F$\neg$) and (T$\exists$) as Exercise 1. $\square$

Our next goal is to prove that the tableau method of proof is complete for intuitionistic logic. We will define a procedure for constructing the appropriate complete systematic tableau starting with a given signed forcing assertion as its root. We will then prove that, for any noncontradictory path P through this tableau, we can build a frame C which agrees with P. Thus if our systematic procedure applied to any forcing assertion $Fp \Vdash \varphi$ fails to produce an intuitionistic tableau proof of φ, then we will have built a frame in which φ is not forced and so demonstrated that φ is not intuitionistically valid.

Rather than trying to first give an abstract definition of when an entry is reduced and a tableau finished, we will directly define the construction procedure and prove it has the properties needed to carry out the completeness proof. (The construction procedure will define a notion of "properly developed" for occurrences of entries in the tableau that will take the place of being "reduced". The properties of a tableau that correspond to its being finished will be listed in Lemma 4.8.) The major simplification that this procedure allows is that, rather than dealing with some abstract partial ordering being constructed with the tableau, we can specify a particular partial ordering from which we will choose all our p's and q's. Any sufficiently rich partial ordering would do. We only have to be able to choose a q from the ordering which extends any given p and is incomparable with any given finite set of elements all of which are incomparable with p. We choose the set S of finite sequences of natural numbers partially ordered by extension, i.e., $p \leq q$ iff q extends p as a sequence. We will take care to declare ordering relations in our tableau so as to agree with this ordering on S. The tableau we construct will use only an initial segment of this ordering. In fact, we will arrange the construction so that, if some sequence p appears on level n of a path P of our tableau, then each initial segment of p appears on P at some level $m < n$ of the tableau.

The procedure for handling the various entries can be motivated considering the atomic tableaux of (T$\forall$) and (F$\forall$). If we develop $Fp \Vdash \forall x\varphi(x)$ by putting down $Fp' \Vdash \varphi(c)$ for some new p' and c, we have exhausted the information contained in the original assertion that p does not force $\forall x\varphi(x)$. On the other hand, developing $Tp \Vdash \forall x\varphi(x)$ by putting down

$Tp' \Vdash \varphi(c)$ for some c and $p' \geq p$ leaves much to be said in terms of the full meaning of the original assertion. $Tp \Vdash \forall x\varphi(x)$ says that for every $p' \geq p$ (in the appropriate partial ordering) and every $c \in \mathcal{L}(p')$, p' forces $\varphi(c)$. Thus, we must arrange to put all of these instances of the forcing assertion on every path containing $Tp \Vdash \forall x\varphi(x)$. Note that we do not need to instantiate φ with every constant, only with those in $\mathcal{L}(p')$. Similarly, we do not have to assert that p' forces $\varphi(c)$ for every $p' \geq p$. As the frame we intend to construct will be built from the information along one noncontradictory path through our finished tableau, we need only consider the constants on the path P which we are extending.

Definition 4.6 (CSIT, *Complete Systematic Intuitionistic Tableaux*): Let φ be a sentence of a language $\mathcal{L}$. Let $d_1, d_2, \ldots, d_n, \ldots$ be a listing of the set D consisting of all the constants of our standard extension $\mathcal{L}_C$ of $\mathcal{L}$ by new constants. For convenience we assume that d_1 is in $\mathcal{L}$. Let $p_1, p_2, \ldots, p_n, \ldots$ be a listing of the set S of all finite sequences of elements of $\mathbb{N}$ which we partially order by extension and let $v_1, v_2, \ldots, v_k, \ldots$ be a listing of the set V of all pairs $\langle d_i, p_j \rangle$ consisting of an element from D and one from S. From now on, when we speak of the least element of D, S or V with some property, we mean the first one in the above lists for the appropriate sets.

We define a sequence τ_n of tableaux and what it means for an occurrence w of an entry E of τ_n to be *properly developed*. The union τ of our sequence of tableaux τ_n will be the *complete systematic intuitionistic tableau* (*the* CSIT) *starting with* φ.

τ_0 is the atomic tableau with root $F\emptyset \Vdash \varphi$. If this tableau requires a partial ordering element p' or a constant c we choose the least elements of S or D which will make it into a tableau according to clause (i) of the Definition 3.2.

Suppose we have constructed τ_n. Let m be the least level of τ_n containing an occurrence of an entry which has not been properly developed and let w be the leftmost such occurrence (say of entry E) on level m of τ_n. We form τ_{n+1} by adding an atomic tableau with root E to the end of every noncontradictory path P through τ_n which contains w. To be precise, we list the noncontradictory paths $P_1, P_2, \ldots, P_k$ of τ_n which contain w. We deal with each P_j in turn by appending an atomic tableau with root E to the end of P_j. Suppose we have reached some P_j on our list. We must now describe the atomic tableau with root E added on to the end of P_j. Cases (T$\vee$), (F$\vee$), (T$\wedge$), (F$\wedge$) and (FAt) of the list of atomic tableaux require no further information to determine the added tableau. Each of the other cases requires fixing some p' and/or some c:

(T$\to$) Let p' be the least q in S which is on P_j, extends p and is such that neither $Fq \Vdash \varphi$ nor $Tq \Vdash \varphi$ occurs on P_j. If there is no such q, let $p' = p$.

(F$\to$) Let $k \in \mathbb{N}$ be least such that $p\hat{}k$ has not occurred in the construction so far and let $p' = p\hat{}k$. (Note that p' is incomparable with everything that has occurred so far except those that are initial segments of p.)

(T$\neg$) Let p' be the least q in S which is on P_j, extends p and is such that $Fq \Vdash \varphi$ does not occur on P_j. If there is no such q, let $p' = p$.

(F$\neg$) Proceed as in case (F$\to$).

(T$\exists$) Let c be the least element of D not occurring in the construction so far.

(F$\exists$) Let c be the least element d of D which is either in $\mathcal{L}$ or else occurs in a forcing assertion $Tq \Vdash \psi$ or $Fq \Vdash \psi$ on P_j for any $q \leq p$ such that $Fp \Vdash \varphi(d)$ does not appear on P_j. If there is no such $d \in D$, let $c = d_1$.

(T$\forall$) Let $\langle p', c \rangle$ be the least $v = \langle r, d \rangle$ in V such that r appears on P_j, d is either in $\mathcal{L}$ or else occurs in a forcing assertion $Tq \Vdash \psi$ or $Fq \Vdash \psi$ on P_j for any $q \leq p$, r extends p and $Tr \Vdash \varphi(d)$ does not appear on P_j. If there is no such pair, we let $p' = p$ and $c = d_1$.

(F$\forall$) Let $k \in \mathbb{N}$ be least such that $p\hat{}k$ has not occurred in the construction so far. We set $p' = p\hat{}k$ and let c be the least element of D not occurring in the construction so far.

(TAt) Let p' be the least q in S on P_j such that $Tq \Vdash \varphi$ does not appear on P_j. If there is no such q, let $p' = p$.

In all of these cases we say that we have *properly developed* the occurrence w of entry E.

Before proceeding with the proofs of the theorems we state some basic properties of CSIT, in particular, the ones that correspond to the classical CST being finished, that allow us to prove the completeness theorem.

Lemma 4.7: *Let* $\tau = \cup\tau_n$ *be a* CSIT *as defined above and* P *a path through* τ.

(i) *If a sequence* $p \in S$ *occurs in an assertion at level* n *of* P, *then every initial segment* q *of* p *occurs on* P *at some level* $m \leq n$ *of* τ.

(ii) τ *is a tableau in accordance with Definition 3.2.*

Proof: (i) We proceed by induction through the construction of τ. The only cases in which we actually introduce a new p on P are (F$\to$) and (F$\forall$). In both cases we introduce some sequence $p\hat{}k$ for a p already on P.

(ii) The only point to verify is that, in the construction of τ_{n+1}, if we add on an atomic tableau with root entry E to the end of some path P_j in τ_n, the p' and c used (if any) satisfy the conditions of Definition 3.2(ii). Otherwise, we obviously are following the prescription for building new tableaux from old ones given in that definition. A simple inspection of the cases shows that we are obeying these restrictions. $\square$

Lemma 4.8: *Let $\tau = \cup\tau_n$ be a* CSIT *as defined above and P a noncontradictory path through τ.*

(T$\vee$) *If $Tp \Vdash \varphi \vee \psi$ appears on P, then either $Tp \Vdash \varphi$ or $Tp \Vdash \psi$ appears on P.*

(F$\vee$) *If $Fp \Vdash \varphi \vee \psi$ appears on P, then both $Fp \Vdash \varphi$ and $Fp \Vdash \psi$ appear on P.*

(T$\wedge$) *If $Tp \Vdash \varphi \wedge \psi$ appears on P, then both $Tp \Vdash \varphi$ and $Tp \Vdash \psi$ appear on P.*

(F$\wedge$) *If $Fp \Vdash \varphi \wedge \psi$ appears on P, then either $Fp \Vdash \varphi$ or $Fp \Vdash \psi$ appears on P.*

(T$\rightarrow$) *If $Tp \Vdash \varphi \rightarrow \psi$ and p' appear on P with $p' \geq p$, then either $Fp' \Vdash \varphi$ or $Tp' \Vdash \psi$ appears on P.*

(F$\rightarrow$) *If $Fp \Vdash \varphi \rightarrow \psi$ appears on P, then for some $p' \geq p$ both $Tp' \Vdash \varphi$ and $Fp' \Vdash \psi$ appear on P.*

(T$\neg$) *If $Tp \Vdash \neg\varphi$ and p' appear on P with $p' \geq p$, then $Fp' \Vdash \varphi$ appears on P.*

(F$\neg$) *If $Fp \Vdash \neg\varphi$ appears on P, then $Tp' \Vdash \varphi$ appears on P for some $p' \geq p$.*

(T$\exists$) *If $Tp \Vdash \exists x\varphi(x)$ appears on P, then $Tp \Vdash \varphi(c)$ appears on P for some c.*

(F$\exists$) *If $Fp \Vdash \exists x\varphi(x)$ appears on P and c is in $\mathcal{L}$ or occurs in a forcing assertion $Tq \Vdash \psi$ or $Fq \Vdash \psi$ on P for any $q \leq p$, then $Fp \Vdash \varphi(c)$ appears on P.*

(T$\forall$) *If $Tp \Vdash \forall x\varphi(x)$ appears on P, c is in $\mathcal{L}$ or occurs in a forcing assertion $Tq \Vdash \psi$ or $Fq \Vdash \psi$ on P for any $q \leq p$ and p' appears on P with $p' \geq p$, then $Tp' \Vdash \varphi(c)$ appears on P.*

(F$\forall$) *If $Fp \Vdash \forall x\varphi(x)$ appears on P, then $Fp' \Vdash \varphi(c)$ appears on P for some c and $p' \geq p$.*

(TAt) *If p and $Tq \Vdash \varphi$ appear on P for any atomic φ and $q \leq p$, then $Tp \Vdash \varphi$ appears on P.*

Proof: First note that every occurrence w in τ of any entry E is properly developed at some stage of the construction. (Consider any w at level n of τ. It is clear that, by the first stage s after all w' at levels $m \leq n$ which are ever properly developed have been so developed, we would have properly developed w.)

Cases (T$\vee$), (F$\vee$), (T$\wedge$), (F$\wedge$), (F$\rightarrow$), (F$\neg$), (T$\exists$) and (F$\forall$) are now almost immediate. Let w be the occurrence of the appropriate signed forcing condition on P. Suppose we properly develop w at stage n of the construction. As w is on P (which is noncontradictory), one of the P_j that we deal with at stage n is an initial segment of P. We add the appropriate atomic tableau to the end of this P_j as part of our construction of τ_n. Thus P must go through one of the branches of this atomic tableau. This immediately gives the desired conclusion.

Next, note that every entry E occurring on P occurs infinitely often on P. The point here is that each occurrence w of E on P is properly developed. When we properly develop w, we add on another occurrence of E to the end of every noncontradictory path in τ_n that goes through w. Thus we make sure that there there is another occurrence of E on P. As every occurrence of each entry E on P is properly developed, the entry itself is properly developed infinitely often. It is now easy to deal with the remaining cases of the lemma. We choose a few examples.

(T$\rightarrow$) Suppose for the sake of a contradiction that p' is the least (in our master listing of S) extension of p that occurs on P such that neither $Fp' \Vdash \varphi$ nor $Tp' \Vdash \psi$ occurs on P. Let Q be the finite set of $q \geq p$ which precede p in our master listing of S. For each $q \in Q$, either $Fq \Vdash \varphi$ or $Tq \Vdash \psi$ occurs on P by our choice of p'. Let m be a stage in the construction of τ by which, for each $q \in Q$, either $Fq \Vdash \varphi$ or $Tq \Vdash \psi$ occurs on the initial segment of P constructed so far. Consider the first stage $n \geq m$ at which we properly develop an occurrence on P of $E = Tp \Vdash \varphi \rightarrow \psi$. (As we properly develop an occurrence of E infinitely often in our construction, there must be such a stage.) The definition of the CSIT then guarantees that we add on the atomic tableau with root E using the given p' to the end of some path P_j which is an initial segment of P. As P_j is an initial segment of P, one of the branches through this atomic tableau must also be an initial segment of P as required.

(T$\forall$) Suppose, for the sake of a contradiction, that $v = \langle p', c \rangle$ is the least pair (in our master listing of V) satisfying the hypotheses of (T$\forall$) but not the conclusion, i.e., $Tp' \Vdash \varphi(c)$ does not occur on P. Let Q be the finite set of pairs $\langle q, d \rangle$ which precede v and satisfy the hypotheses of T$\forall$. Let m be a stage by which, for each $\langle q, d\rangle \in Q$, we already have an occurrence of $Fq \Vdash \varphi(d)$ on the initial segment of P defined so far. Consider the first stage $n \geq m$ at which we properly develop an occurrence on P of $E = Fp \Vdash \forall x\varphi(x)$. The definition of the CSIT then guarantees that we add on the atomic tableau with root E using the given p' and c to the end of some path P_j which is an initial segment of P. As P_j is an initial segment of P, the unique branch through this atomic tableau must also be an initial segment of P as required.

All the remaining cases, (T$\neg$), (F$\exists$) and (TAt), are proved in a similar fashion. We leave them as Exercise 3. $\square$

Theorem 4.9: *Suppose that $\tau = \cup\tau_n$ is a* CSIT *and P is a noncontradictory path in τ. We define a frame $\mathcal{C} = (R, \leq, \mathcal{C}(p))$ associated with P as follows:*

> *R is the set of all sequences in S appearing in forcing assertions on P. The partial ordering on R is the same as that on S: extension.*
>
> *For each $p \in R$, $C(p)$ is the set consisting of the constants of $\mathcal{L}$ and all other constants appearing in forcing assertions $Tq \Vdash \psi$ or $Fq \Vdash \psi$ on P with $q \leq p$.*
>
> *For each $p \in R$, $A(p)$ is the set of all atomic sentences ψ such that $Tq \Vdash \psi$ occurs on P for some $q \leq p$. (Warning: We are using the convention that every $c \in C(p)$ is named by itself in $\mathcal{L}(p)$.)*

If we set f and g to be the identity functions on R and on the set of constants appearing on P, respectively, then they are witnesses that $\mathcal{C}$ agrees with P.

Proof: First, note that the clauses of the definition of $\mathcal{C}$ are designed to guarantee that $\mathcal{C}$ is a frame for $\mathcal{L}$ according to Definition 2.1. Just remember that every constant c in $\mathcal{L}(p)$ names itself.

We now wish to prove that P agrees with $\mathcal{C}$; we use induction on the complexity of sentences φ appearing in forcing assertions on P.

Atomic φ: If $Tp \Vdash \varphi$ appears on P then φ is in $A(p)$ and so forced by p. If $Fp \Vdash \varphi$ appears on P then we must show that $Tq \Vdash \varphi$ does not appear on P for any $q \leq p$. (This is the only way that p could come to force φ in $\mathcal{C}$.) If there were such an occurrence of $Tq \Vdash \varphi$ on P then, by Lemma 4.8 (TAt), $Tp \Vdash \varphi$ would also occur on P contradicting the assumption that P is noncontradictory.

The inductive cases are each handled by the corresponding clauses of Lemma 4.8 and of the definition of forcing (Definition 2.2) together with the induction assumption for the theorem, i.e. the requirements for $\mathcal{C}$ to agree with P are met for sentences of lower complexity.

As a sample we consider $F\forall$: $Fp \Vdash \forall x\varphi(x)$ appears on P. By Lemma 4.8 (F$\forall$), $Fp' \Vdash \varphi(c)$ appears on P for some c and $p' \geq p$. The inductive hypothesis then says that p' does not force $\varphi(c)$ in $\mathcal{C}$. The definition of forcing a universal sentence (2.2(v)) then tells us that p does not force $\forall x\varphi(x)$ in $\mathcal{C}$, as required.

The remaining cases are left as Exercise 4. □

We can now state the standard form of the completeness theorem.

Theorem 4.10: *If φ is intuitionistically valid then it has an intuitionistic tableau proof.*

Proof: Consider the CSIT τ starting with an intuitionistically valid φ. If τ is not an intuitionistic tableau proof of φ, then it has by definition a noncontradictory path P. Theorem 4.9 then provides a frame $\mathcal{C}$ in which $\emptyset$ does not force φ. Thus φ can not be intuitionistically valid for a contradiction. □

Exercises

1. Complete the remaining cases (F$\rightarrow$), (F$\neg$) and (T$\exists$) in the proof of Lemma 4.5.
2. Prove that any CSIT is finite if and only if it is contradictory.
3. Complete the remaining cases (T$\neg$), (F$\exists$) and (TAt) in the proof of Lemma 4.8.
4. Complete the remaining cases in the proof of Theorem 4.9.

5. Decidability and Undecidability

The CSIT gives us a systematic method for searching for either an intuitionistic proof of a given sentence φ or a frame counterexample. As we have noted before, if a proof exists there is a finite proof and the CSIT will give such a proof by the completeness theorem. On the other hand, if φ is not intuitionistically valid, the frame counterexample constructed in the proof of the completeness theorem will usually be infinite. Indeed, there is in general no way of avoiding such a situation. Intuitionistic logic, like classical logic, is undecidable: There is no algorithm which is guaranteed to terminate in a finite time and to tell us if φ is intuitionistically valid (Theorem 5.16). Nonetheless, there are special classes of sentence whose intuitionistic validity can be decided and there are ways of improving our chances of finding both proofs and finite counterexamples in many cases.

On the side of generating proofs more quickly and efficiently, we note that the completeness theorem tells us that we can add on any sound procedure for generating tableaux without changing the class of provable sentences. That is, whatever extension we make to the schemes for generating tableau proofs, if we can still prove a soundness theorem, then the new schemes (as our current ones) generate proofs for exactly the intuitionistically valid sentences. Of course, there is a trade-off. The more rules we add on, the more possibilities there are for developing the tableau. Viewed as a search through all possible tableau proofs, adding on new procedures increases the breadth of the search space for the sake of possibly decreasing the depth of the search, i.e., the size of the proof we construct. We introduce one such rule which considerably shortens many proofs and, in addition, allows us to redress a certain imbalance in the tableau rules we have presented.

To motivate this additional rule, we first reconsider Example 3.5 which proved the intuitionistic validity of $\varphi \to (\psi \to \varphi)$ for atomic φ and ψ:

1	$F\emptyset \Vdash \varphi \to (\psi \to \varphi)$	
2	$T0 \Vdash \varphi$	by 1
3	$F0 \Vdash \psi \to \varphi$	by 1
4	$T00 \Vdash \psi$	by 3
5	$F00 \Vdash \varphi$	by 3
6	$T00 \Vdash \varphi$	by 2
	$\otimes$	by 5, 6

FIGURE 59

Our interest now is in the last step of the proof: Line 6 (which together with line 5 provides our contradiction) follows from line 2 by atomic tableau (T*At*). It is at this point (and only at this point) in the proof that we used the fact that φ and ψ were atomic sentences as atomic tableau TAt applied only to atomic sentences. We chose this rule to correspond to the definition of a frame which demands monotonicity only for atomic tableau. On the other hand, the monotonicity lemma (Lemma 2.12) shows that the rule is sound for any sentence φ: If $p \Vdash \varphi$ and $p' \geq p$, then $p' \Vdash \varphi$ in any frame. Thus we may revise our tableau procedures by strengthening atomic tableau TAt to apply to all sentences φ:

$$\boxed{\begin{array}{l} T\varphi \\ \qquad Tp \Vdash \varphi \\ \qquad\quad | \\ \qquad Tp' \Vdash \varphi \\ \quad \text{for any } p' \geq p \end{array}}$$

MONOTONICITY

The proof of the soundness theorem with this strengthened atomic tableau is exactly as with the original one for atomic φ. Thus we have not changed the class of provable sentences but we have greatly expanded the applicability of our proofs. All the tableau proofs in the examples and exercises in §3 for which φ and ψ were assumed atomic are now seen to prove the validity of the specified expressions for any sentences φ and ψ. Indeed, as (T*At*) was the only way of developing a tableau that was not equally applicable to all sentences, we now see that any tableau proof of a sentence built up from atomic φ, ψ, etc., actually proves all instances of the sentence gotten by replacing φ, ψ, etc., by arbitrary formulas. (Warning: There is one proviso to be observed in such replacements. As our tableau procedures apply only to sentences, we must take care not to introduce any free variables into our formulas.)

Let us now turn to using the tableau proof procedure to try to produce frame counterexamples to sentences that are not intuitionistically valid. Remember, there are are no guaranteed methods here. We can, however, convert the proof of the completeness theorem into a somewhat more general test for having produced a frame counterexample by formulating a notion of a finished tableau based on the assertions in Lemma 4.8.

Definition 5.1: If τ is a tableau and $\leq$ is the ordering defined on the p's and q's appearing in τ then τ is *finished* if every noncontradictory path P through τ has the thirteen properties listed in Lemma 4.8.

Theorem 5.2: *If τ is a finished tableau with root $Fp \Vdash \varphi$ and P is a noncontradictory path through τ, then there is a frame C which agrees with P (and so φ is not intuitionistically valid).*

Proof: We proceed exactly as in Theorem 4.9 except that the ordering on the p's occurring in forcing assertions on P is now defined by τ (rather than being given in advance by extension of sequences). The proof of Theorem 4.9 then makes no use of any properties of the CSIT other than those specified in Lemma 4.8. These properties of P are now guaranteed by the definition of τ being finished. ☐

Thus if we can produce any noncontradictory finished tableau with a root of the form $Fp \Vdash \varphi$, then we know that we have built a frame counterexample to φ and so shown that it is not intuitionistically valid.

Example 5.3: Consider trying to prove the nonvalid sentence

$$\varphi \to (\varphi \to \psi).$$

We begin by developing the two implications to reach line 5 of Figure 60. Now if φ and ψ are atomic sentences, it is easy to see that this tableau is finished. As it is noncontradictory, we have shown that $\varphi \to (\varphi \to \psi)$ is not intuitionistically valid for atomic sentences φ and ψ.

$$\begin{array}{lll}
1 & F\emptyset \Vdash \varphi \to (\varphi \to \psi) & \\
2 & T0 \Vdash \varphi & \text{by } 1 \\
3 & F0 \Vdash (\varphi \to \psi) & \text{by } 1 \\
4 & T00 \Vdash \varphi & \text{by } 3 \\
5 & F00 \Vdash \psi & \text{by } 3
\end{array}$$

FIGURE 60

We can carry the analysis one step further by actually displaying a frame in which $\varphi \to (\varphi \to \psi)$ is not forced; such a frame can be constructed from the tableau in Figure 60 above.

$$\begin{array}{c}
\langle \{c\}, \{\varphi\} \rangle \\
| \\
\langle \{c\}, \{\varphi\} \rangle \\
| \\
\langle \{c\}, \emptyset \rangle
\end{array}$$

Of course, we could just as well eliminate the top line of the frame as it adds nothing to the second one. We also see that this frame provides us with a template to produce counterexamples to $\varphi \to (\varphi \to \psi)$ for some nonatomic φ and ψ. If we can arrange that $\emptyset$ does not force φ while 0 forces φ but not ψ, we will have the desired counterexample.

Example 5.4: Consider $\varphi \vee \neg\varphi$.

$$\begin{array}{lll}
1 & F\emptyset \Vdash \varphi \vee \neg\varphi & \\
2 & F\emptyset \Vdash \varphi & \text{by } 1 \\
3 & F\emptyset \Vdash \neg\varphi & \text{by } 1 \\
4 & T0 \Vdash \varphi & \text{by } 3
\end{array}$$

FIGURE 61

Again, if φ and ψ are atomic sentences, we have a finished tableau and so a proof that $\varphi \vee \neg\varphi$ is not intuitionistically valid. The frame counterexample corresponding to this finished tableau is the same one produced in Example 2.7. The difference here is that it was produced automatically.

Example 5.5: Consider the sentence $(\neg\varphi \to \neg\psi) \to (\psi \to \varphi)$.

$$\begin{array}{llll}
1 & \multicolumn{2}{c}{F\emptyset \Vdash (\neg\varphi \to \neg\psi) \to (\psi \to \varphi)} & \\
2 & \multicolumn{2}{c}{T0 \Vdash \neg\varphi \to \neg\psi} & \text{by } 1 \\
3 & \multicolumn{2}{c}{F0 \Vdash \psi \to \varphi} & \text{by } 1 \\
4 & \multicolumn{2}{c}{T00 \Vdash \psi} & \text{by } 3 \\
5 & \multicolumn{2}{c}{F00 \Vdash \varphi} & \text{by } 3 \\
7 & F00 \Vdash \neg\varphi & T00 \Vdash \neg\psi & \text{by } 2 \\
8 & T000 \Vdash \varphi & F00 \Vdash \psi & \text{by } 7 \\
 & & \otimes & \text{by } 4, 8
\end{array}$$

FIGURE 62

This is not a finished tableau but we can use it to produce a frame counterexample if we are sufficiently clever. The key idea is that no further development will produce any true forcing assertions $Tp \Vdash \theta$ for any new θ. Thus we must have as much information as we need to build the counterexample. In fact, letting $\emptyset$ and 0 force no atomic statements, $00 \Vdash \psi$, $000 \Vdash \psi$ and $000 \Vdash \varphi$ will give the desired frame counterexample. Now, from the viewpoint of forcing, $\emptyset$ and 0 are indistinguishable. Thus we might as well collapse them. So we end up with $\emptyset$, 0 and 00 as the partially ordered set,

$$A(\emptyset) = \emptyset, \quad A(0) = \{\psi\}, \quad A(00) = \{\varphi,\ \psi\}.$$

We leave the verification that this is indeed a frame counterexample as Exercise 13.

Example 5.6: Consider the sentence $(\varphi \to \psi) \vee (\psi \to \varphi)$.

1	$F\emptyset \Vdash (\varphi \to \psi) \vee (\psi \to \varphi)$	
2	$F\emptyset \Vdash \varphi \to \psi$	by 1
3	$F\emptyset \Vdash \psi \to \varphi$	by 1
4	$T0 \Vdash \varphi$	by 2
5	$F0 \Vdash \psi$	by 2
6	$T1 \Vdash \psi$	by 3
7	$F1 \Vdash \varphi$	by 3

FIGURE 63

Observe that this is the first example in which the "new" $p' \geq p$ stipulation of rule $(F{\to})$ (applied here to line 3 to obtain lines 6 and 7) forces our frame to *branch*. Node 1 in line 6 was chosen as the least node greater than $\emptyset$ incomparable with every p on the tree not $\leq \emptyset$ in accordance with rule $(F{\to})$ of Definition 4.6 of a CSIT. In fact, no linear (nonbranching) frame can fail to force $(\varphi \to \psi) \vee (\psi \to \varphi)$. The above tableau, however, is finished and not contradictory, so we see that $(\varphi \to \psi) \vee (\psi \to \varphi)$ is not intuitionistically valid.

We conclude with an example of a sentence which is not intuitionistically valid and for which developing the CSIT for a few steps does not obviously supply a frame counterexample. Indeed, this is an example for which no finite frame can be a counterexample.

Example 5.7: Consider the sentence $\forall x \neg\neg\varphi(x) \to \neg\neg\forall x \varphi(x)$.

The CSIT for this sentence begins, in a somewhat abbreviated form, as in Figure 64.

1	$F\emptyset \Vdash (\forall x)\neg\neg\varphi(x) \to \neg\neg(\forall x)\varphi(x)$	
2	$T0 \Vdash (\forall x)\neg\neg\varphi(x)$	by 1
3	$F0 \Vdash \neg\neg(\forall x)\varphi(x)$	by 1
4	$T0 \Vdash \neg\neg\varphi(c)$	by 2
5	$T00 \Vdash \neg(\forall x)\varphi(x)$	by 3
6	$T00 \Vdash \neg\neg\varphi(c)$	by 2
7	$F0 \Vdash \neg\varphi(c)$	by 4
8	$F00 \Vdash (\forall x)\varphi(x)$	by 5
9	$F00 \Vdash \neg\varphi(c)$	by 6
10	$T01 \Vdash \varphi(c)$	by 7
11	$F000 \Vdash \varphi(c_1)$	by 8

FIGURE 64

It is not easy to see how to construct a frame counterexample from the initial stages of the CSIT. A direct analysis of the semantics shows that we should keep introducing new constants $c_1, c_2, \ldots$ and, while not forcing $\varphi(c_n)$ immediately, guarantee that each $\varphi(c_n)$ is forced in some extension. We can see the beginnings of this phenomena in the cycle generated by line 4 producing lines 7 and 10 on the one hand and in the cycle generated by line 5 producing lines 8 and 11 on the other.

Figure 65 is a simplified frame which gives a counterexample.

$$\vdots$$

$$\langle \{c, c_1, c_2, c_3\}, \{\varphi(c), \varphi(c_1), \varphi(c_2)\} \rangle$$

$$\langle \{c, c_1, c_2\}, \{\varphi(c), \varphi(c_1)\} \rangle$$

$$\langle \{c, c_1\}, \{\varphi(c)\} \rangle$$

$$\langle \{c\}, \emptyset \rangle$$

FIGURE 65

We leave the verification that $\emptyset$ does not force $\forall x \neg\neg\varphi(x) \rightarrow \neg\neg\forall x \varphi(x)$ in the indicated frame as Exercise 15. We will, however, show that no finite frame can be a counterexample.

Proposition 5.8: $\forall x \neg\neg\varphi(x) \rightarrow \neg\neg\forall x \varphi(x)$ *is forced by every node* p *in every frame* $\mathcal{C}$ *with a finite partial ordering* R.

Proof: Let p, $\mathcal{C}$ and R be as in the proposition. To verify that $p \Vdash \forall x \neg\neg\varphi(x) \rightarrow \neg\neg\forall x \varphi(x)$ consider any $q \geq p$ such that $q \Vdash \forall x \neg\neg\varphi(x)$. We must show that $q \Vdash \neg\neg\forall x \varphi(x)$. By Lemma 2.13, this is equivalent to the assertion that, for every $r \geq q$, there is an $s \geq r$ such that $s \Vdash \forall x \varphi(x)$. Fix any $r \geq q$. As R is finite, there is a maximal extension s of r in R. Now by monotonicity, $s \Vdash \forall x \neg\neg\varphi(x)$. Thus, for any $c \in C(s)$, $s \Vdash \neg\neg\varphi(c)$. Applying Lemma 2.13 again, as well as the maximality of s, gives us that $s \Vdash \varphi(c)$. Thus (again by the maximality of s), $s \Vdash \forall x \varphi(x)$ as required. $\square$

It is no accident that the sentence in Example 5.7 which has no finite counterexamples contains quantifiers. It is only for sentences with quantifiers that infinite frames are necessary to get counterexamples. Any sentence without quantifiers is either intuitionistically valid or has a finite frame counterexample. This fact supplies us with a decision procedure (albeit a crude one) for the intuitionistic validity of quantifier-free formulas.

Theorem 5.9 (Finite Model Property): *A quantifier-free sentence is forced in all frames if and only if it is forced in all finite frames.*

Proof: Consider any quantifier-free formula φ. We must show that if it is not forced by some $p \in R$ in a frame $\mathcal{C}$, then there is some finite frame $\mathcal{C}'$ in which it is not forced. Let X be the set of all subformulas of φ. (Recall Definition II.2.6 or Proposition II.3.8 which determine the subformulas of φ.) For p in R, define a class $[p]$ of elements of R which force the same elements of X as p:

$$[p] = \{q \in R \mid (\forall \psi \in X)(p \Vdash \psi \leftrightarrow q \Vdash \psi)\}.$$

Let R' be the set of all such $[p]$ for p in R. Now different classes $[p] \in R'$ correspond to different subsets of X. As X, the set of subformulas of φ, is finite, so is R'. Partially order R' by $[q] \leq [p]$ if every formula in X forced by q is forced by p. (Due to the definition of $[p]$ and $[q]$, this is the same as the requirement that every formula in X forced by some r in $[q]$ is also forced by some s in $[p]$.) Define a finite frame $\mathcal{C}'$ with R' as its partially ordered set by setting $A([p]) = A(p) \cap X$ and $C'([p])$ to be the set of constants appearing in $A([p])$. We claim that for all $p \in R$ and all ψ in X, $[p] \Vdash_{\mathcal{C}'} \psi$ if and only if $p \Vdash_{\mathcal{C}} \psi$. This claim clearly suffices to prove the theorem. We proceed to prove the claim by induction on formulas:

Atomic ψ: $A([p]) = A(p) \cap X$ says that if ψ is an atomic formula in X, then $[p] \Vdash_{\mathcal{C}'} \psi$ if and only if ψ is in $A(p) \cap X$, or equivalently, if and only if $p \Vdash \psi$.

Induction Step: Suppose θ and ψ are in X. Suppose, by induction, that for all $q \in R$, $q \Vdash_{\mathcal{C}} \theta$ if and only if $[q] \Vdash_{\mathcal{C}'} \theta$ and $q \Vdash_{\mathcal{C}} \psi$ if and only if $[q] \Vdash_{\mathcal{C}'} \psi$

(1) Disjunction: $p \Vdash_{\mathcal{C}} \theta \vee \psi \Leftrightarrow p \Vdash_{\mathcal{C}} \theta$ or $p \Vdash_{\mathcal{C}} \psi \Leftrightarrow [p] \Vdash_{\mathcal{C}'} \theta$ or $[p] \Vdash_{\mathcal{C}'} \psi$ (by induction) $\Leftrightarrow [p] \Vdash_{\mathcal{C}'} \theta \vee \psi$.

(2) Conjunction: $p \Vdash_{\mathcal{C}} \theta \wedge \psi \Leftrightarrow p \Vdash_{\mathcal{C}} \theta$ and $p \Vdash_{\mathcal{C}} \psi \Leftrightarrow [p] \Vdash_{\mathcal{C}'} \theta$ and $[p] \Vdash_{\mathcal{C}'} \psi$ (by induction) $\Leftrightarrow [p] \Vdash_{\mathcal{C}'} \theta \wedge \psi$.

(3) Implication: Suppose $[p] \Vdash_{\mathcal{C}'} \theta \rightarrow \psi$. We must show that $p \Vdash_{\mathcal{C}} \theta \rightarrow \psi$. If $q \geq p$ and $q \Vdash_{\mathcal{C}} \theta$, then by induction $[q] \Vdash_{\mathcal{C}'} \theta$, so by our assumption and the fact that $[q] \geq [p]$ follows from $q \geq p$, $[q] \Vdash_{\mathcal{C}'} \psi$. The induction hypothesis then says that $q \Vdash_{\mathcal{C}} \psi$ as required. Conversely, suppose $p \Vdash_{\mathcal{C}} \theta \rightarrow \psi$ and $\theta \rightarrow \psi$ is in X. We must prove that $[p] \Vdash_{\mathcal{C}'} \theta \rightarrow \psi$, that is, if $[q] \geq [p]$ and $[q] \Vdash_{\mathcal{C}'} \theta$, then $[q] \Vdash_{\mathcal{C}'} \psi$. Now as $\theta \rightarrow \psi$ is in X and $p \Vdash_{\mathcal{C}} \theta \rightarrow \psi$ by assumption, $[q] \geq [p]$ implies that $q \Vdash_{\mathcal{C}} \theta \rightarrow \psi$. By our assumption that $[q] \Vdash_{\mathcal{C}'} \theta$ and the induction hypothesis, $q \Vdash_{\mathcal{C}} \theta$. Thus $q \Vdash_{\mathcal{C}} \psi$ and again by induction, $[q] \Vdash_{\mathcal{C}'} \psi$, as required.

(4) Negation is similar to implication and the verification is left as Exercise 16. $\square$

Theorem 5.10: *We can effectively decide the intuitionistic validity of any quantifier-free sentence.*

Proof: We know by the properties of the CSIT expressed in the completeness theorem (Theorems 4.9 and 4.10) that if a given sentence φ is intuitionistically valid then the CSIT will give a (necessarily) finite tableau proof of φ. On the other hand, if φ is not valid then there is, by the finite model property (5.9), a finite frame counterexample. We can thus simultaneously search for a finite frame counterexample to φ and develop the CSIT for φ. We must eventually find either a finite counterexample to φ or an intuitionistic tableau proof that φ is intuitionistically valid. $\Box$

The decision procedure embodied in the proof of Theorem 5.10 is not very satisfactory. It give us little information on how large the proof or counterexample must be nor on how long we must search before the required one shows up. All one can say is that, given a quantifier-free sentence φ, it suffices to consider all frames for the atomic formulas appearing in the given sentence which are of size at most that of the set of all subformulas of φ. With considerably more work it is possible to give a more explicit algorithm for generating a possible tableau proof of a quantifier-free φ with a better bound on the number of steps needed to produce either a proof or finite counterexample (see Nerode [1990, 4.2]). We only remark that the decision procedure is considerably more complicated than for classical logic. As we have noted (Theorem I.4.8), quantifier-free predicate logic is equivalent to propositional logic. The decision problem for satisfiability in classical propositional logic is the archetypal NP complete problem. For propositional intuitionistic logic, however, the decision problem is complete for polynomial space (Statman [1979, 5.3]).

Decidability in intuitionistic logic can be pushed a bit farther. In contrast to classical logic, intuitionistic validity for the class of prenex sentences is decidable. The above proof (or any other) for quantifier-free sentences can be extended (Exercise 17) to prenex sentences by applying the existence property (Theorem 2.21) and an intuitionistic version of the theorem on constants (Exercise 3.32). As one should expect, however, the validity problem for all of intuitionistic logic, like that for classical logic, is undecidable. Given the undecidability of the validity problem for classical logic (Corollary III.7.10), we can deduce it for intuitionistic logic by a validity preserving translation due to Gödel.

Definition 5.11: If A is an atomic formula, then $\neg\neg A$ is a *Gödel formula*. If φ and ψ are Gödel formulas then so are $\neg\varphi$, $\varphi \wedge \psi$ and $\forall x\varphi$.

Recall that, in classical logic, A and $\neg\neg A$ are equivalent and $\{\neg, \wedge, \forall\}$ is adequate. So for every formula φ, there is a Gödel formula φ° which is classically equivalent to φ. The decision problem for classical validity is thus reducible to deciding the validity of just the Gödel formulas. We now wish to show that a Gödel formula ψ is classically valid if and only if it is intuitionistically valid. The "if" direction is simply a special case of Theorem 2.6. We need to prove the converse for Gödel formulas.

Lemma 5.12: *If φ is a Gödel sentence and p is a forcing condition in a frame C, then either $p \Vdash \varphi$ or $(\exists q \geq p)(q \Vdash \neg\varphi)$. In particular, if p does not force φ then one of the following cases holds:*

(1) *If $\varphi = \neg\psi$, then $\exists q \geq p(q \Vdash \psi)$.*

(2) *If $\varphi = \psi \wedge \theta$, then $\exists q \geq p(q \Vdash \neg\psi$ or $q \Vdash \neg\theta)$.*

(3) *If $\varphi = \forall x\psi$, then $(\exists q \geq p)(\exists c \in C(q))(q \Vdash \neg\psi(c))$.*

Proof: We proceed by induction on φ. The base case is that φ is $\neg\neg A$ for some atomic sentence A. In this case, if p does not force φ, there is, by the definition of forcing a negation (Definition 2.2 (iii)), a $q \geq p$ which forces $\neg A$ as required in (1). Note that, in general, if $\varphi = \neg\psi$ and $q \Vdash \psi$ then $q \Vdash \neg\neg\psi$ (i.e., $q \Vdash \neg\varphi$) by the intuitionistic validity of $\psi \to \neg\neg\psi$ (Example 2.15).

If φ is $\neg\psi$ and p does not force $\neg\psi$, then, as in the base case, there is a $q \geq p$ which forces ψ and so $\neg\neg\psi$.

If φ is $\psi \wedge \theta$ and p does not force φ, then either p does not force ψ or p does not force θ. Thus by induction there is a $q \geq p$ which forces $\neg\psi$ or $\neg\theta$. (Again, note that by the basic facts about forcing this implies that $q \Vdash \neg(\psi \wedge \theta)$.)

If φ is $\forall x\psi(x)$ and p does not force φ, then, by the definition of forcing, there is an $r \geq p$ and a $c \in C(r)$ such that r does not force $\psi(c)$. The induction hypothesis then tells us that there is a $q \geq r$ such that $q \Vdash \neg\psi(c)$, as required. Of course any such q forces $\neg\varphi$ as well. $\Box$

Definition 5.13: A sequence $\langle p_i \rangle$ of forcing conditions in a frame C is a *generic sequence extending* p if $p_0 = p$ and the following conditions hold:

(i) For every i, $p_i \leq p_{i+1}$.

(ii) For every atomic sentence ψ there is an i such that p_i forces ψ or p_i forces $\neg\psi$.

(iii) For every Gödel sentence φ, there is an i such that $p_i \Vdash \varphi$ or p_{i+1} is a condition $q \geq p_i$ as required for φ in the appropriate clause of Lemma 5.12.

Lemma 5.14: *For every forcing condition p in a frame C, there is a generic sequence extending p.*

Proof: Let $\{\varphi_i \mid i \in N\}$ be an enumeration of all the Gödel sentences. We define a sequence $\langle p_i \mid i \in \mathbb{N}\rangle$ by induction. Set $p_0 = p$. If $p_i \Vdash \varphi_i$ and φ_i is $\neg\neg\psi$ for some atomic sentence ψ, then, by the definition of forcing, there is a $q \geq p_i$ which forces ψ. Let p_{i+1} be such a q. If $p_i \Vdash \varphi_i$ but φ_i is not of this form, let $p_{i+1} = p_i$. If p_i does not force φ_i, let p_{i+1} be a condition extending p_i as guaranteed in the clause of Lemma 5.12 corresponding to φ_i. (So, in particular, if φ_i is $\neg\neg\psi$ for an atomic ψ and p_i does not force φ_i, then $p_{i+1} \Vdash \neg\psi$.) It is clear that the sequence p_i satisfies the definition of a generic sequence extending p. □

Theorem 5.15: *For every Gödel sentence φ, φ is classically valid if and only if φ is intuitionistically valid.*

Proof: As we remarked, if φ is intuitionistically valid it is classically valid (Theorem 2.6). To prove the converse suppose that φ is not intuitionistically valid, i.e., there is a frame $\mathcal{C}$ and a forcing condition p such that p does not force φ. We must build a classical model $\mathcal{A}$ in which φ is false. By Lemma 5.14 we can choose an enumeration of Gödel sentences in which $\varphi_0 = \varphi$ and a generic sequence $\langle p_i | i \in \mathbb{N} \rangle$ in $\mathcal{C}$ extending p. Note that, by our assumption that p does not force φ and the definition of a generic sequence, $p_1 \Vdash \neg\varphi$. We let the universe A of our required classical model $\mathcal{A}$ be $\cup\{C(p_i) | i \in \mathbb{N}\}$. We define the relations of $\mathcal{A}$ by $\mathcal{A} \vDash R(\vec{c})$ iff $\exists i(p_i \Vdash R(\vec{c}))$.

We claim that, for every Gödel sentence φ, $\mathcal{A} \vDash \varphi \Leftrightarrow \exists i(p_i \Vdash \varphi)$. As $p_1 \Vdash \neg\varphi$ (and $\neg\varphi$ is a Gödel sentence), this gives us the desired classical model of $\neg\varphi$. The proof is by induction on the formation of the sentence φ. (Note that we specify the levels of formation of formulas in the order given in Definition 5.11 so that $\forall x\psi(x)$ follows $\neg\psi(c)$ for each constant c.)

The base case is that φ is $\neg\neg\psi$ for some atomic sentence ψ. As p_i is a generic sequence, there is an i such that p_i forces ψ or p forces $\neg\psi$. In the former case, $\mathcal{A}$ satisfies ψ (and so $\neg\neg\psi$) by definition. In the latter case, no p_j can force ψ. So by the definition of $\mathcal{A}$, ψ is false in $\mathcal{A}$.

If φ is $\psi \wedge \theta$, then $\mathcal{A} \vDash \varphi \Leftrightarrow \mathcal{A} \vDash \psi$ and $\mathcal{A} \vDash \theta$. By induction this condition holds iff there are j and k such that $p_j \Vdash \psi$ and $p_k \Vdash \theta$. As the p_i form an increasing sequence of forcing conditions, this is equivalent to the existence of an i such that $p_i \Vdash \psi \wedge \theta$, i.e., one such that $p_i \Vdash \varphi$.

If φ is $\neg\psi$, then $\mathcal{A} \vDash \varphi \Leftrightarrow \mathcal{A} \nvDash \psi$. By induction this is true if and only if there is no j such that $p_j \Vdash \psi$. By the definition of generic sequence, this last condition is equivalent to the existence of an i such that $p_i \Vdash \neg\psi$, i.e., $p_i \Vdash \varphi$.

If φ is $\forall x\psi(x)$ suppose first that $\mathcal{A} \vDash \varphi$. If there is no p_i forcing φ, then by the definition of a generic sequence, there is an i and a c such that $p_i \Vdash \neg\psi(c)$. Then, by induction, $\mathcal{A} \vDash \neg\psi(c)$ for the desired contradiction. For the converse, suppose that there is an i such that $p_i \Vdash \forall x\psi(x)$. Then, for every $c \in A$, there is a $j \geq i$ such that $c \in C(p_j)$ and $p_j \Vdash \psi(c)$. By induction, we then know that $\mathcal{A} \vDash \psi(c)$ for every $c \in A$, i.e., $\mathcal{A} \vDash \varphi$ as required. □

Theorem 5.16: *The validity problem for intuitionistic logic is undecidable.*

Proof: If we could effectively decide if any given sentence ψ is intuitionistically valid then we could decide if any sentence φ is classically valid by checking if φ° (as defined in 5.11) is intuitionistically valid. This would contradict the undecidability of validity for classical predicate logic (Corollary III.7.10). □

Corollary 5.17: *Not every sentence is intuitionistically equivalent to a sentence in prenex form.*

Proof: If every sentence had an intuitionistically equivalent prenex form, a systematic search for a tableau proof of such an equivalence would find one. The decision procedure for the validity of prenex sentences (Exercise 17) would then supply one for all sentences. □

Exercises

Below is a list (1–12) of classically valid sentences θ which are not intuitionistically valid. For each one, start a tableau with $F\emptyset \Vdash \theta$ and develop it enough to produce a frame in which θ is not forced. In each example assume that φ and ψ are atomic with either no free variables (1–8) or only x free (9–14).

1. $(\varphi \vee \neg\varphi)$
2. $(\neg\neg\varphi \rightarrow \varphi)$
3. $\neg(\varphi \wedge \psi) \rightarrow (\neg\varphi \vee \neg\psi)$
4. $\neg\varphi \vee \neg\neg\varphi$
5. $(\neg\varphi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \varphi)$
6. $(\varphi \rightarrow \psi) \rightarrow (\neg\varphi \vee \psi)$
7. $\neg(\forall x)\varphi(x) \rightarrow (\exists x)\neg\varphi(x)$
8. $(\forall x)\neg\neg\varphi(x) \rightarrow \neg\neg(\forall x)\varphi(x)$
9. $(\forall x)(\varphi \vee \psi(x)) \rightarrow (\varphi \vee (\forall x)\psi(x))$
10. $((\varphi \rightarrow (\exists x)\psi(x)) \rightarrow (\exists x)(\varphi \rightarrow \psi(x))$
11. $(\forall x)\varphi(x) \rightarrow \psi) \rightarrow (\exists x)(\varphi(x) \rightarrow \psi(x))$
12. $((\forall x)(\varphi(x) \vee \neg\varphi(x)) \wedge \neg\neg(\exists x)\varphi(x)) \rightarrow (\exists x)\varphi(x)$
13. Show that $\emptyset$ does not force $(\neg\varphi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \varphi)$ in the frame produced in Example 5.5.

14. Show that $(\varphi \to \psi) \to (\psi \to \varphi)$ is forced by every node in every frame C in which the ordering on R is a linear ordering.

15. Show that $\emptyset$ does not force $\forall x \neg\neg\varphi(x) \to \neg\neg\forall x\varphi(x)$ in the frame produced in Example 5.7.

16. Complete the proof of Theorem 5.9 by verifying the negation case of the induction step.

17. Prove that intuitionistic validity is decidable for the class of prenex sentences. Hint: Successively apply Exercises 2.9 and 3.32 to reduce deciding the validity of a sentence in prenex form to deciding the validity of many quantifier–free sentences.

6. A Comparative Guide

In this section we supply a comparative guide to the similarities and differences between modal and intuitionistic logic. We use the presentation of classical predicate logic in Chapter II as a common starting point.

6.1 Syntax

The syntax for classical logic is presented in II.2.1 – II.2.6. For technical reasons we assume in both modal and intuitionistic logic that our language $\mathcal{L}$ has at least one constant symbol but no function symbols other than constants. We also view $\leftrightarrow$ as a defined symbol

$$\varphi \leftrightarrow \psi \quad \text{means} \quad \varphi \to \psi \ \wedge \ \psi \to \varphi.$$

Of course, modal logic adds two new operators $\Box$ and $\Diamond$ which can be viewed syntactically as unary propositional connectives. Their semantics, however, is really second order.

6.2 Semantics

The semantics for classical logic is presented in II.4 by interpreting the language $\mathcal{L}$ in a structure $\mathcal{A}$ for $\mathcal{L}$. Both modal and intuitionistic logic use systems of structures called *frames* and a new relation, *forcing* ($\Vdash$), to define their semantics. In both cases, a frame consists of a set (W or R) of *possible worlds*, a binary relation S on this set and a function assigning a classical structure $\mathcal{C}(p)$, with domain $C(p)$, to each world p. We let $\mathcal{L}(q)$ be the expansion of $\mathcal{L}$ gotten by adding a constant for each $c \in C(q)$. In modal logic (IV.2.2), the binary relation S can be arbitrary while in intuitionistic logic it is always assumed to be a partial ordering $\leq$. In both cases we assume that the domains of the structures are monotonic: If pSq ($p \leq q$), then $C(p) \subseteq C(q)$. In intuitionistic logic we also require that the sets of true atomic facts be monotonic: If $p \leq q$, φ is an atomic sentence and $\mathcal{C}(p) \vDash \varphi$ then $\mathcal{C}(q) \vDash \varphi$.

The crucial definition (IV.2.2 and V.2.2) is now that of the *forcing relation* between possible worlds p and sentences φ : $p \Vdash \varphi$. For atomic formulas and for the inductive cases corresponding to $\vee$, $\wedge$ and $\exists$, the definitions for both logics are the same "natural" extension of the classical definition of truth (II.4.3):

> if φ is atomic $p \Vdash \varphi \Leftrightarrow \mathcal{C}(p) \vDash \varphi$; $p \Vdash \varphi \vee \psi \Leftrightarrow p \Vdash \varphi$ or $p \Vdash \psi$; $p \Vdash \varphi \wedge \psi \Leftrightarrow p \Vdash \varphi$ and $p \Vdash \psi$; $p \Vdash \exists x\varphi(x) \Leftrightarrow$ there is a c in $\mathcal{L}(p)$ such that $p \Vdash \varphi(c)$. Of course, modal logic also has extra clauses giving the semantics of $\Box$ and $\Diamond$.

The crucial differences between modal and intuitionistic logic, however, are in the treatment of $\neg$, $\to$ and $\forall$. In modal logic, the interpretations continue to follow the "natural" classical style, e.g., $p \Vdash \neg\varphi \Leftrightarrow p$ does not force φ. The situation is quite different in intuitionistic logic. Here $p \Vdash \neg\varphi$ means that no $q \geq p$ forces φ;

$$p \Vdash \varphi \to \psi \quad \Leftrightarrow \quad \text{every } q \geq p \text{ which forces } \varphi \text{ also forces } \psi;$$

$$p \Vdash \forall x\varphi(x) \quad \Leftrightarrow \quad \text{for every } q \geq p \text{ and every } c \in \mathcal{L}(q),\ q \Vdash \varphi(c).$$

Once the forcing relation has been defined, the notions corresponding to classical truth (II.4.3) and validity (II.4.4) are then the same for both modal and intuitionistic logic: φ is *forced in a frame* $\mathcal{C}$ iff $p \Vdash \varphi$ for every world p of $\mathcal{C}$; φ is *valid* if it is forced in every frame.

It is worth pointing out that the "unusual" intuitionistic interpretations for $\neg$, $\to$ and $\forall$ all have a modal flavor. For example, the definitions of $p \Vdash \neg\varphi$ and $p \Vdash \forall x\varphi(x)$ in intuitionistic logic seems much like those of $p \Vdash \Box\neg\varphi$ and $p \Vdash \Box\forall x\varphi(x)$, respectively, in modal logic. Such ideas form the basis of Gödel's validity–preserving translation of intuitionistic logic into the modal logic S4. (See Gödel [1933, 2.3] and the introductory notes in Gödel [1986, 2.3] vol. 1, 269–299.)

6.3 Tableaux

The *atomic tableaux* are designed to mirror the semantics of the various connectives and quantifiers. Thus it is no surprise that the modal tableaux (IV.3.1) for the atomic sentences and the classical connectives are essentially the same as the classical ones of II.6. For example, the classical tableau beginning with $T(\varphi \wedge \psi)$ adjoins both $T\varphi$ and $T\psi$; the modal one beginning $Tp \Vdash \varphi \wedge \psi$ adjoins both $Tp \Vdash \varphi$ and $Tp \Vdash \psi$.

In addition to the classical concern of using only a new constant as a witness for a true existential sentence (or counterexample for a false universal one), the analysis of the quantifiers in modal logic has to reflect the correct relation among the domains. Thus, for example, given an entry $Tp \Vdash \exists x\varphi(x)$ on a path in a tableau, we can introduce a new c and add the assertion that $Tp \Vdash \varphi(c)$ to the path. On the other hand, given an entry $Tp \Vdash \forall x\varphi(x)$, we can expand it by adjoining $Tp \Vdash \varphi(c)$ for any appropriate c, i.e., any c appearing in any forcing assertion about a world q such that p is accessible from q (and so any one in the intended domain $C(p)$). The modal operators introduce the possibility of creating new worlds. For example, given $Tp \Vdash \Diamond\varphi$ we introduce a new world q such that pSq and $q \Vdash \varphi$. We must also interpret $\Box$ to conform to the semantics and the accessibility relation: given $Tp \Vdash \Box\varphi$, we can adjoin $Tq \Vdash \varphi$ for any appropriate q, i.e., any q such that pSq.

In intuitionistic logic, however, we must make changes that correspond to the new semantics for atomic sentences and the connectives $\neg$ and $\rightarrow$ as well as the quantifiers. The monotonicity assumption for atomic facts tells us to adjoin $Tq \Vdash \varphi$ when given $Tp \Vdash \varphi$ and pSq for any atomic φ. The analysis of the "unusual" connectives involves either the creation of new worlds or looking ahead to future worlds that have already been defined, as is required in the modal analysis of $\Box$. Given $Tp \Vdash \neg\varphi$, for example, we can adjoin $Fq \Vdash \varphi$ for any appropriate q, i.e., any q such that $p \leq q$. On the other hand, given $Fp \Vdash \neg\varphi$ we can introduce a new world q such that $p \leq q$ and adjoin the assertion $Tq \Vdash \varphi$. The analysis for intuitionistic implication is similar: If $Tp \Vdash \varphi \rightarrow \psi$, $p \leq q$ and $Tq \Vdash \varphi$ all appear on a path P, we can adjoin $Tq \Vdash \psi$ to P; if $Fp \Vdash \varphi \rightarrow \psi$ appears, we can introduce a new q and adjoin pSq, $Tq \Vdash \varphi$ and $Fq \Vdash \psi$. The tableaux for the quantifiers are constructed as you would expect: Given $Tp \Vdash \exists x\varphi(x)$, we can introduce a new c and adjoin the assertion that $Tp \Vdash \varphi(c)$; given $Tp \Vdash \forall x\varphi(x)$ we can adjoin $Tq \Vdash \varphi(c)$ for any appropriate q and c, i.e., any $q \geq p$ and any c in the world q.

Once the distinct semantics have been built into the atomic tableaux, the definitions of *tableaux* in modal logic (IV.3.2) and intuitionistic logic (V.3.2) are both essentially the same as in classical logic (II.6.1). The only changes are to the notions of "new" and "appropriate" so they fit the intended semantics and notation. We should point out that it is quite possible to put the ordering relations $p \leq q$ directly on the tableaux in the intuitionistic case as we did in the modal case. We did not do that only because the construction of the ordering is so restricted in the intuitionistic case that it can easily be read off from the rest of the tableau. Thus if one is working solely within intuitionistic logic, it is simpler to stick closer to the classical format and not clutter the tableaux with the ordering facts. The generality of the binary relations constructed in the modal tableaux makes it simpler to record them directly.

In any case, the notions of a *tableau proof* (II.6.2, IV.3.3, V.3.3) are essentially the same in all three logics: A tableau beginning with an assertion of the falsity of φ (or of φ's being forced) in which every path contains a contradiction (the assertion of both the truth and falsity of some sentence) is a proof of φ.

6.4 Soundness and Completeness

The soundness (II.7.2, IV.4.3, V.4.3) and completeness (II.7.7, IV.4.13, V.4.10) theorems are formally the same in all three logics: If φ is provable, it is valid; if φ is valid it is provable. Moreover, at least in outline, the proofs of soundness and completeness are the same as well. For the proof of soundness, the crucial lemma is that if a structure or frame agrees with the root of a tableau, then there is a path through the tableau such that the structure or frame agrees with every entry on the path (II.7.1, IV.4.2, V.4.2). Of course the straightforward notion of agreement used in classical logic ($\mathcal{A}$ agrees with $T\varphi$ iff $\mathcal{A} \vDash \varphi$) must be generalized to deal with frames and forcing but the notions are formally identical in modal (IV.4.1) and intuitionistic logics (V.4.1). The proof of the basic lemma in each case is an induction on the construction of the tableau. The modal and intuitionistic arguments again differ only where the atomic tableaux are distinct. Each then follows the appropriate semantics. Given this basic lemma the proof of soundness is straightforward and identical in all three settings: If one has a proof of φ, it begins with an assertion that φ is false and every path on the tableau includes a contradiction. As no structure or frame can agree with a contradiction, none can agree with the root, i.e., φ is valid.

The completeness theorem is proved by defining a systematic way of producing a tableau with a given root that develops every entry: the *Complete Systematic* (*Modal* or *Intuitionistic*) *Tableau* (CST, II.6.9: CSMT, IV.4.8; CSIT, V.4.6). The modal construction also defines the possible worlds and the accessibility relation among them in the construction of the CSMT as these are built into the atomic tableau. In the intuitionistic case, the restriction of the accessibility relation to a partial ordering allows us to specify in advance both the set of possible worlds and the ordering by choosing any sufficiently complex ordering. We choose the set of finite sequences of natural numbers ordered by extension. The idea that every entry has been developed is expressed by the notions of a *reduced entry* and a *finished tableau* (II.6.7, IV.4.6) in the classical and modal developments. Our proof for intuitionistic logic avoids an explicit definition by building the required properties into the construction of the CSIT and the proof of Lemma V.4.8. (The notion of a *properly developed entry* in Definition V.4.6 replaces that of a reduced entry in Definition IV.4.6 and the notion of a finished tableau is replaced by the properties listed in Lemma V.4.8.) It would be just as reasonable, although somewhat longer, to extract the appropriate definitions for intuitionistic logic and present them as for modal logic in IV.4.6. If, as suggested above, the atomic intuitionistic tableaux are also modified to put the ordering relations directly on the tableaux, the definition of the CSIT could be given as for the CSMT with the corresponding adjustments for the quantifiers and the "unusual" connectives.

The completeness theorem is now proved in all three logics by considering the complete systematic tableau beginning with the assertion of the falsity of φ or of φ's being forced at p. If this tableau is not a proof of φ, it has a noncontradictory path P. We now use the entries on P to build a structure or frame $\mathcal{C}$ which agrees with every entry on P (II.7.3, IV.4.11, V.4.9). In the classical case, φ is not true in $\mathcal{C}$ while in the modal and intuitionistic cases, p does not force φ in $\mathcal{C}$. Thus in each case φ is not valid, as required.

For classical logic, we define a single structure $\mathcal{C}$. Its domain C consists of all terms appearing in assertions on P. The structure itself is defined by letting $\mathcal{C} \vDash \varphi$ for each atomic sentence φ iff $T\varphi$ appears on P. In both the modal and intuitionistic constructions, we must build an entire frame $\mathcal{C}$. The set of possible worlds for $\mathcal{C}$ is the set of p appearing in assertions on P. In the modal proof, pSq is true in $\mathcal{C}$ iff it appears on P. The accessibility relation in the intuitionistic case is determined in advance as extension for the sequences p, q on P. However, had we adjusted the intuitionistic atomic tableaux to directly record the ordering facts, we would define the same ordering by taking the facts appearing on P.

The domains $C(p)$ of the structures $\mathcal{C}(p)$ needed to define the frame $\mathcal{C}$ are the same in both cases because our monotonicity assumptions on them are the same: $C(p)$ consists of the constants of $\mathcal{L}$ as well as all those appearing in any forcing assertion involving a q such that p is accessible from (greater than) q. The definitions of the structures $\mathcal{C}(p)$ on these domains $C(p)$ differ because of the monotonicity assumption on atomic facts in intuitionistic logic. In the modal case, in analogy with classical logic, we define an atomic sentence φ to be true in $\mathcal{C}(p)$ iff $Tp \Vdash \varphi$ appears on P. In the intuitionistic construction, $\mathcal{C}(p) \vDash \varphi$ iff $Tq \Vdash \varphi$ appears on P for any $q \leq p$.

For all three logics, the argument that the frame we have defined agrees with every entry on P is a straightforward induction on the complexity of sentences φ appearing on P. This induction concludes the proof of the completeness theorem.

6.5 Logical Consequences

In both classical and modal logic we defined the notion of a sentence φ being a logical consequence of a set Σ of sentences. Classically, $\Sigma \vDash \varphi$ iff φ is true in every structure $\mathcal{C}$ in which every $\psi \in \Sigma$ is true (II.4.4). In modal logic, we said that φ is a logical consequence of Σ iff φ is forced in every frame $\mathcal{C}$ which forces every $\psi \in \Sigma$ (IV.2.8). For each logic, we then introduced a notion of a tableau proof from Σ. The only modification needed to standard tableau proofs was that we could now add on $T\psi$ ($Tp \Vdash \psi$) to any path P on a tableau for any $\psi \in \Sigma$ (and any p occurring on P). (See II.6.1 and IV.3.11 for the classical and modal cases, respectively.) We then defined the complete systematic tableau from premises (II.6.9, IV.4.14) and proved the appropriate soundness and completeness theorems for deductions from premises (II.7.2, IV.4.13; II.7.7, IV.4.15).

Although we have not explicitly made the corresponding generalizations for intuitionistic logic, they can be modeled routinely on the ones for modal logic. We leave this development to the exercises.

6.6 Decidability and Hilbert-Style Systems

In V.5 we proved the decidability of propositional intuitionistic logic (V.5.10) and the undecidability of full predicate intuitionistic logic. A tableau-based proof for the decidability of propositional modal logic can be found in §7 of Chapter 8 of Fitting [1983, 4.4]. As modal predicate logic includes classical logic, i.e., a sentence φ without modal operators is classically valid iff it is modally valid (Exercise IV.1.10), modal logic is *a foritori* undecidable by Church's theorem for predicate logic (III.8.10).

In IV.6 we give a Hilbert-style system of axioms and rules for modal logic. Describing a corresponding system for intuitionistic logic requires some care. As $\neg$ and $\rightarrow$ no longer form an adequate set of connectives, one must explicitly deal with each of the propositional connectives. Similarly, one must explicitly give axioms and rules for $\exists$. Kleene [1952, 2.3] in §19 supplies such a system for intuitionistic logic which has the advantage of being extendable to a proof system for classical logic by simply adding on the axiom scheme $\neg\neg\varphi \rightarrow \varphi$ or $\varphi \vee \neg\varphi$.

Exercises

We outline a treatment of intuitionistic logic more closely modeled on our treatment of modal logic, as suggested above.

1. Reformulate the intuitionistic atomic tableaux so that the ordering relations appear explicitly. As a sample, the atomic tableau ($F\rightarrow$) becomes the following:

$$\begin{array}{c} Fp \Vdash \varphi \rightarrow \psi \\ | \\ p \leq p' \\ | \\ Tp' \Vdash \varphi \\ | \\ Fp' \Vdash \psi. \end{array}$$

2. Describe the corresponding notions of "new" and "appropriate" for this system of intuitionistic tableaux and prove the soundness theorem.

3. Define the appropriate notions of a reduced entry and a finished tableau for this system.

4. Define a corresponding notion of a CSIT and prove the completeness theorem.
5. Define the notion of logical consequence for intuitionistic logic as was done for modal logic.
6. Describe the modifications needed in Exercises 2–4 to develop general intuitionistic tableaux from a set of premises and the corresponding notion of a CSIT from premises.
7. Describe the changes needed in Exercises 2–4 to prove the soundness and completeness theorems for deductions from premises in intuitionistic logic.

Suggestions for Further Reading

The description of a Hilbert–style system of axioms and rules in the style of I.7 and II.8 for intuitionistic logic requires some care. As $\neg$ and $\rightarrow$ no longer form an adequate set of connectives, one must explicitly deal with each of the propositional connectives. Similarly, one must explicitly give axioms and rules for $\exists$. Kleene [1952, 2.3] in §19 supplies such a system for intutionistic logic which has the advantage of being extendible to a proof system for classical logic by simplying adding on the axiom scheme $\neg\neg\varphi \rightarrow \varphi$. Kleene then gives a careful development of logic and recursion theory which tells which theorems have been intuitionistically proved and which not.

In addition to his validity–preserving translation of classical into intuitionistic logic, Gödel [1933, 2.3] also supplied one from intuitionistic propositional logic into an equivalent of the propositional part of the modal logic S4, as described in IV.6. See the introductory note of Gödel [1933, 2.3] in Gödel [1986, 2.3] vol. 1, 296–299, for an explanation of the translation and references for proofs of both the propositional and predicate versions of the translation.

Fitting [1983, 4.1] supplies many variants of the tableau method. For an elementary exposition of intuitionistic logic using natural deduction and algebraic methods, see van Dalen [1983, 3.2]. For a philosophical introduction to intuitionism, see Dummett [1977, 4.1]. For constructive mathematics, see Bishop and Bridges [1985, 4.1], Richman and Bridges [1987, 4.1] and Troelstra and van Dalen [1988, 4.1]. For more advanced topics in the metamathematics of constructive mathematics, see Beeson [1985, 4.1].

For intuitionistic type theory, see Martin–Löf [1984, 4.1]; for its application to a computer system constructive theorem prover, see Constable [1986, 5.6]. For a combination of modal and intuitionistic logics relevant to computer science concerns, see Nerode and Wijesekera [1992, 5.6].

Finally, the "classic" basic text on intuitionism is Heyting [1971, 4.2].

Appendix A
An Historical Overview

We begin with an analogy between the history of calculus and the history of mathematical logic.

1. Calculus

In Athens in the Golden Age of Classical Greece, Plato (c. 428–348 B.C.E.) made geometry a prerequisite for entrance to his philosophical academy, for a master of geometry was a master of correct and exact reasoning (see Thomas [1939, 1.1]). Euclid (c. 300 B.C.E.) emphasized the importance of the axiomatic method, which proceeds by deduction from axioms. (From this point of view, logic is the study of deduction.) Euclid and Archimedes (287–212 B.C.E.) and their predecessors showed how to use synthetic geometry to calculate areas and volumes of many simple figures and solids. They also showed how to solve, using geometry, many simple mechanics, hydrostatics and geometrical optics problems.

In the twenty centuries separating Euclid and Archimedes from Leibniz (1640–1710) and Newton (1640–1722), increasingly difficult problems of calculating areas and volumes and of mechanics and hydrostatics were solved one by one by special methods from Euclidean and Archimedian geometry. Each physical or mathematical advance made by the use of this geometric method required the extraordinary mathematical talent of a Galileo (1564–1642) or a Huygens (1629–1695). Things changed radically after Descartes's discovery, published as the appendix to his *Discours de la Methode* [1637, 2.3], that geometric problems could be translated into equivalent algebraic problems. Geometric methods were replaced by algebraic computations.

There were already strong hints of symbolic-algebraic methods of integration and differentiation in the work of Fermat (1601-1665) and in the works of Newton's teacher Barrow (1630–1677) and Leibniz's predecessor Cavalieri (1598–1647). The symbolic methods of differentiation and integration discovered by Newton and Leibniz made it possible for later generations to use the ordinary calculus to develop science and engineering without being mathematical geniuses. These methods are still the basis for understanding, modeling, simulating, designing, and developing physical and engineering systems. Both Leibniz and Newton were aware of the

Turing, A. M., "Checking a large routine", in *Report of a Conference on High Speed Automatic Calculating Machines*, University Mathematical Library, Cambridge, England, 1949.

Wos, L., Overbeek, R., Lusk, E. and Boyle, J., *Automated Reasoning*, Prentice-Hall, Englewood Cliffs, N. J., 1984.

Index of Symbols

Index of Terms