Technical Report

UCAM-CL-TR-257
ISSN 1476-2986

Number 257

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

Cut-free sequent and tableau systems for propositional normal modal logics

Rajeev Prakhakar Goré

May 1992

15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom
phone +44 1223 763500

http://www.cl.cam.ac.uk/

Technical reports published by the University of Cambridge
Computer Laboratory are freely available via the Internet:

http://www.cl.cam.ac.uk/techreports/

ISSN 1476-2986

Abstract

We present a unified treatment of tableau, sequent and axiomatic formulations for many propositional normal modal logics, thus unifying and extending the work of Hanson, Segerberg, Zeman, Mints, Fitting, Rautenberg and Shvarts. The primary emphasis is on tableau systems as the completeness proofs are easier in this setting. Each tableau system has a natural sequent analogue defining a finitary provability relation $\vdash_L$ for each axiomatically formulated logic **L**. Consequently, any tableau proof can be converted into a sequent proof which can be read downwards to obtain an axiomatic proof. In particular, we present cut-free sequent systems for the logics **S4.3**, **S4.3.1** and **S4.14**. These three logics have important temporal interpretations and the sequent systems appear to be new.

All systems are sound and (weakly) complete with respect to their known *finite* frame Kripke semantics. By concentrating almost exclusively on finite tree frames we obtain finer characterisation results, particularly for the logics with natural temporal interpretations. In particular, all proofs of tableau completeness are constructive and yield the finite model property and decidability for each logic.

Most of these systems are cut-free giving a Gentzen cut-elimination theorem for the logic in question. But even when the cut rule is required, all uses of it remain analytic. Some systems do not possess the subformula property. But in all such cases, the class of "superformulae" remains bounded, giving an analytic superformula property. Thus, all systems remain totally amenable to computer implementation and immediately serve as nondeterministic decision procedures for the logics they formulate. Furthermore, the constructive completeness proofs yield deterministic decision procedures for all the logics concerned.

In obtaining these systems, we demonstrate that the subformula property can be broken in a systematic and analytic way while still retaining decidability. This should not be surprising since it is known that modal logic is a form of second order logic and that the subformula property does not hold for higher order logics.

Contents

Erratum to Cambridge Computer Laboratory technical report number 257: Cut-free sequent and tableau systems for propositional normal modal logics by Rajeev Goré.

In this technical report I claim that Fitting's and Rautenberg's systems are incomplete because they omit contraction. The claim is *wrong* because contraction is implicit in their set notation.

The systems of Fitting and Rautenberg are complete because they allow contraction on any formula whereas the systems in this technical report explicitly build contraction into certain rules.

Please accept my apologies for any confusion this causes you.

RAJEEV GORÉ

23/7/92

Chapter 1

Introduction

Logic: The art of thinking and reasoning in strict accordance with the limitations and incapacities of human misunderstanding.

Ambrose Bierce: The Devil's Dictionary.

Automating deduction in classical first-order logic has been the prime aim of the automatic theorem proving community for the past 25 years [AMCP84]. Although proof procedures for classical logic had been known for many years, the advent of the resolution principle and particularly of the unification procedure in 1963 gave a "machine-oriented logic" which was easy to automate using computers [DP60, Gil60, Pra60, PHV60, Rob65, Rob79]. The most glaring disadvantage of the resolution method is that the initial formula under investigation has to be put into a clausal normal form — a procedure that destroys the structure of this formula and often causes an exponential blowup in the size of the resulting formula. As a response, "non-clausal" theorem proving methods have been developed that require no normal forming and have been claimed to be superior to resolution methods [And81, Bib81]. The results of Eder [Ede84, Ede88] and D'Agostino [D'A90] indicate that the issue is not settled, but the fact remains that automated deduction in classical logic is now well understood and the sights are set on higher-order and non-classical logics [AMCP84] [TMM88].

At almost the same time as the advent of the resolution principle, the advent of the possible-world semantics of Kripke [Kri59, Kri63a, Kri63b, Kri65] and Hintikka [Hin63] was revolutionising the field of modal logic. Modal logic had been suggested as an improvement of classical logic because of philosophical dissatisfaction with the classical interpretation of the conditional statement "if A then B". The classical truth functional interpretation of this conditional as "A is false or B is true" corresponds to the (material) implication $A \Rightarrow B$, making the implication true when A is false. According to Diodorus of Iasus [Den81], and more recently to Lewis [Lew20], there is an element of impossibility involved in the statement "if A then B" so that its correct meaning should be something like "it is impossible for A to be true and B to be false" [Zem73, page 77]. Lewis used $\Diamond$ to denote possibility and his interpretation becomes $\neg\Diamond(A \wedge \neg B)$. Later authors used $\Box$ to denote necessity, the dual of possibility, and it has become customary to use $\Box$ as a primitive and define $\Diamond$ as $\neg\Box\neg$. Under this convention, Lewis' "strict implication" $\neg\Diamond(A \wedge \neg B)$ becomes $\Box(A \Rightarrow B)$ using the equivalences between $(A \wedge \neg B)$ and $\neg(A \Rightarrow B)$. Strict implication is also often written as $A \prec B$ and L and M are also used instead of $\Box$ and $\Diamond$ respectively.

A vast amount of subsequent research had been done on these modal notions of necessity and possibility, but until 1959, the traditional (axiomatic) formulations of modal logic had lacked an intuitive semantics, although algebraic semantics were known [BS84]. Kripke semantics provided an intuitive way to think about modal logics and led to the blossoming of the field as we know it today. The term "modal", incidentally, originates from the fact that in mediaeval times, necessity ($\Box$), possibility ($\neg\Box\neg$), contingency ($\neg\Box$) and impossibility ($\Box\neg$) were thought of as the *modes* in which a proposition could be true or false [HC68, page 23].

This is not the place to justify the use of modal logics. Indeed, many eminent mathematicians and philosophers have vilified modal logic as a waste of time [Qui61, Qui76]. Nevertheless, modal logics have proved useful in artificial intelligence research because certain modal logics can be used in a very natural way to model epistemic notions like belief and knowledge. To model belief, $\Box A$ is read as "A is believed", and to model knowledge as "A is known" [HM85, McA88]. By interpreting $\Box A$ as "A is always true", propositional modal logics have also proved useful in modelling the behaviour of programs and of digital circuits where time is an important concept, leading to a plethora of dynamic logics [Pra79, Gol87] and temporal logics [Pnu77, Wol87, Gol87]. Recently, modal logics have been used as "non-monotonic logics" — logics where conclusions can be drawn tentatively, and then retracted in the light of further (damning) evidence [McD82, Moo85, Shv90]. As a consequence, automated deduction in modal logics has now become an important and even fashionable research area in computer science.

The most obvious way to proceed is to extend resolution to handle modal logics, and various authors do exactly this using different modal clausal normal forms [Far85] [Ven85] [Cha86, Cha87] [CF86, EF89] [GK86] [FH88] [Orc89] [Min90a, Min90b]. In parallel, non-clausal methods for modal deduction have been investigated [Fit83, Fit88, Wal87] [Wol83, Gou84, EH86] [AM90] [Fit90], along with various methods where modal logics are translated into classical first order logic to utilise the wealth of experience in theorem proving in classical first order logic [Mor76] [JR87, JR88] [AE89] [Ohl90] [Bür90, FS91] [Gen91].

But in most of these methods there is a limit to the generality of the solution. Of the resolution methods, Chan's recursive resolution method and the resolution method of Farinas Del Cerro and colleagues have been applied to only a few logics. Of the non-clausal methods, the method of Abadi and Manna is not mechanisable due to its inherent redundancy [AM90] while Wolper's tableau method is unlikely to generalise to handle first order modal logics [AM90]. Even the very recent work of Wallen is limited to those logics that have the subformula property [Wal87, Wal89]. The equational method of Auffray and Enjalbert is not extendible to handle certain modal logics with temporal interpretations [AE89, page 444]. And by translating into first order logic, the translational methods immediately surrender the decidability of the propositional modal logic they translate.

As we shall see, the axiomatic and semantic properties of modal logics are very well understood and a uniform treatment of them is not only simple but is also very elegant [Seg71, Gol87]. In the light of this uniformity, the methods mentioned above seem ad hoc since there is no uniform way to obtain similar theorem proving methods for the vast number of modal logics that exist.

On the other hand, sequent systems and tableau systems for propositional modal logics have been studied in the philosophy literature for almost 50 years because they provide deep insights into important aspects of the metamathematics of these logics via theorems on interpolation, cut-elimination and compactness [Han66b, OM57b, Kan57, Fit83, Rau83]. Except for the work of Abadi and Manna, the non-clausal methods listed above all trace their origins to the tableau methods of Kripke [Kri63b, Kri59] and Beth [Bet55, Bet53], or to the sequent methods of Kanger [Kan57] and Gentzen [Gen35, Sza69], or to the natural deduction methods of Fitch [Fit66]. Recently, Mints has shown a close relationship between modal resolution, Maslov's inverse method and Gentzen's sequent systems [Min90a, Min90b, Lif89]. Fitting has also shown close links between tableau methods, the Davis-Putnam procedure and modal resolution [Fit90, DP60]. Thus, it appears as if sequent and tableau systems are not only fundamental to the study of modal logics *per se*, but are also fundamental to the resolution and non-clausal methods used for automated deduction in modal logics.

We present a uniform semantic treatment of tableau and sequent systems for a large number of propositional normal modal logics, building upon, unifying, and extending the work of Hanson [Han66a], Segerberg [Seg71], Fitting [Fit83], Rautenberg [Rau83] and Shvarts [Shv89]. In most cases, the tableau and sequent systems are cut-free, but even when the cut rule is not eliminable, all uses of it can be made analytic in a tractable and predictable way so that all our tableau and sequent systems give nondeterministic decision procedures for the logics they formulate. Furthermore, all our proofs of tableau completeness are constructive, immediately yielding deterministic decision procedures for each logic.

The crucial step is to abandon the subformula property which has been a tenet of sequent and tableau systems because it guarantees that the system is analytic. This loss is already apparent in certain axiomatic completeness proofs that require using filtrations through a set of "superformulae" [Gol87]. Thus it should not be surprising that a similar tradeoff emerges in tableau and sequent completeness proofs. Fortunately though, the "superformulae" involved are always analytic, thus preserving decidability, and furthermore, they can be obtained in a systematic way. Propositional modal logics appear to slot into that middle ground between classical logic and higher order logic where the loss of the subformula property does not lead to disaster.

1.1 Formulating Logics

This dissertation is rife with mathematical terminology but I have tried to define every term I use and keep the dissertation self contained. In this and the following few sections I have tried to describe what this dissertation is about without using jargon. The description is necessarily inexact and in some places simplistic, so the expert reader is asked to bear with me until the later, more rigorous, chapters.

When dealing with logics we first chosen some (usually countably infinite) alphabet of symbols and deem certain sequences of these symbols to be (well formed) formulae. A logic **L** is then simply some (usually infinite) subset of the set of all formulae. Theorem proving or automated deduction in its most basic form is the task of determining whether some given arbitrary formula is or is not a member of a particular logic **L**. The easiest way to formalise a logic **L** is to simply associate the logic with the list of its member formulae, but this is clearly unsatisfactory. What we seek instead is a finite collection of rules that formalise how to deduce the members of **L**.

There are many different ways to formalise (modal) logics but we concentrate on only three. The first, and oldest, is the axiomatic method dating from Aristotle and its importance stems from this long tradition. The second is the sequent method invented by Gerhard Gentzen in 1935 and its importance stems from its applicability in metamathematics and in theorem proving. The third is the semantic method for modal logics, essentially due to Kripke and Hintikka, and its importance stems from its intuitive appeal. The first two methods are *syntactic* since deduction involves the manipulation of formulae alone, and because of this, they are potential candidates for automation on computers. The third method involves intuitions about objects other than formulae and provides a sense of meaning to the modal operators $\Box$ and $\Diamond$. When these three formalisms coincide, our intuitions about modal deductions correspond to the purely syntactic manipulations performed by a computer.

1.1.1 Axiomatic (Hilbert) Systems

In axiomatic or Hilbert systems, a (usually finite) set of formulae are taken as axioms in that they are included in the logic by definition. These axioms are supplemented with transformation or inference rules that are purely syntactic conditional statements allowing us to add other formulae to the logic when the syntactic conditions are met. The axioms provide a basis from which to start this incremental process and different axioms and different inference rules typically, but not always, give different logics.

In axiomatic systems, a formula is a theorem if it can be obtained by a finite sequence of applications of the inference rules, starting with the axioms. Such a sequence is called a proof and the logic **L** formulated by an axiomatic system consists of the set of theorems of the axiomatic system. Thus, axiomatic systems are "forward" or "bottom up" systems in that deductions always take the form "from the fact that $A_1, A_2, \cdots, A_n$ are theorems, infer that B is a theorem". The biggest disadvantage with "forward" systems is that they give no insight on how to prove that some arbitrary B is, or is not, a theorem since there is no way to direct us to B. Deduction in axiomatic systems requires experience, although the final proofs can be relatively short.

1.1.2 Gentzen Sequent Systems

In sequent or Gentzen systems, there is only one axiom, but many inference rules. The inference rules are again conditional statements just as in axiomatic systems, but the axiom, with no conditions, is accepted as an unquestionable member of every logic. In sequent systems, a formula is a theorem if we can reduce it to an instance of this basic axiom using a finite number of sequent inference rules in a *backward* manner. That is, sequent systems are "backward" or "goal directed" systems in that deductions always take the form "B is a theorem if $A_1, A_2, \cdots, A_n$ are theorems". The inference rules then give a recipe for reducing a goal B into its subgoals $A_1, A_2, \cdots, A_n$ and each subgoal to its own subgoals, and so on until each subgoal is reduced to an instance of the basic axiom. Such a reduction sequence usually forms a tree and is called a proof. Again, the logic **L** formulated by a sequent system consists of the set of all formulae that are theorems of the sequent system.

1.1.3 Kripke Semantics

The third and most recent way to formulate (modal) logics is the semantic method. As the name suggests, semantics are a way of assigning a (hopefully intuitive) meaning to the formulated logic. The usual semantics for classical propositional logic, for example, assigns a truth value of "true" or "false" to each formula by assigning "true" or "false" to each of its primitive components. There are usually many different ways of assigning coherent truth values and a formula is a tautology if and only if it evaluates to "true" under *all* truth assignments. As we shall see, Kripke semantics are a generalisation of this concept using a network of nodes called possible worlds, with each node having its own truth assignment. Such a network is called a Kripke model and different conditions on the allowed interconnections give different classes of Kripke models. The modal analogue of tautology is validity and a formula is said to be valid in a particular model if and only if it evaluates to true in every possible world of that model. The logic **L** formulated by a particular possible world formulation consists of all formulae that are valid in *all* models in that particular class of models.

The interpretations of $\Diamond$ as possibility and $\Box$ as necessity are obtained as follows. There is one world (node) which is deemed to be the real world and all the worlds connected to this world are "possible" alternative worlds. The arcs between worlds are directed and if there is an arc from w_1 to w_2 then w_2 is accessible from or reachable from w_1. If w_2 assigns "true" to A and w_2 is reachable from w_1 then w_1 assigns "true" to $\Diamond A$ (possibly A) since there is a possible alternative world to w_1 which makes A true.

Similarly, if A is assigned "true" by every world reachable from w_1 then w_1 assigns "true" to $\Box A$ (necessarily A) since A is true in all the possible alternatives to w_1 and must therefore be necessary. Different conditions about reachability give different logics. For example, if there is an arc from w_1 to w_2 and one from w_2 to w_3 then should we consider w_3 to be reachable from w_1 ? If so then accessibility is said to be transitive.

1.1.4 Relating Syntactic and Semantic Formulations

The axiomatic and sequent syntactic formulations give no intuitions about the logic they formulate since they involve purely syntactic manipulations of formulae without recourse to the meanings of the formulae. On the other hand, a semantically formulated logic immediately possesses some intuitive appeal since we can visualise the network of nodes and truth assignments associated with our intuitions about the meanings of possibility, necessity, belief, knowledge or time. The desired connections between the syntactic and semantic formulations are achieved via the notions of soundness and completeness.

We say that a syntactic (axiomatic or sequent) formulation is sound with respect to a semantic formulation if every theorem of the syntactic formulation is valid in the semantic formulation. We say that a syntactic (axiomatic or sequent) formulation is complete with respect to a semantic formulation if every valid formula of the semantic formulation is a theorem of the syntactic formulation. Soundness guarantees that if we can prove A syntactically, then A is valid semantically. Completeness guarantees that if A is valid semantically, then there is a syntactic proof of A. We thus establish that validity corresponds to theoremhood, giving an intuitive meaning to the syntactic concept of theoremhood. But note that the correspondence is either between an axiomatic formulation and the semantic formulation or between a sequent formulation and the semantic formulation. It is only when each syntactic formulation corresponds to the same semantics *independently* that we have a correspondence between the two syntactic formulations.

Axiomatic modal formulations have been studied extensively in the literature on modal logic. There are many results relating axiomatic systems and Kripke semantics and we make extensive use of these known results. That is, we show that our sequent systems are sound and complete with respect to Kripke semantics. We then refer to the known results showing that certain axiomatic systems are also sound and complete with respect to the same semantics. We thus obtain the correspondence between these axiomatic and sequent formulations using Kripke semantics as a bridge.

1.2 Sequent and Tableau Systems

Sequent systems were developed by Gentzen [Gen35] as a syntactic formulation of classical logic. Tableau systems were first developed by Beth [Bet53, Bet55]. Kripke semantics are equally applicable to both as the two are really notational variants of each other. There are various properties of sequent (and hence tableau) systems that are of vital importance in automated theorem proving and in the study of logical systems. These play a central role in this dissertation.

1.2.1 Sequent Systems As Decision Procedures

An extremely important property for automated deduction is whether a syntactically formulated logic is decidable or undecidable. That is, are there effective procedures to decide whether an arbitrary formula is or is not a theorem. The beauty of propositional sequent systems is that if they formulate a decidable propositional logic **L**, they implicitly give a procedure to determine whether an arbitrary formula A is or is not a theorem of **L** when used in a backward manner. Of course, the procedure is not always an efficient one, but it can often be made so by using various engineering techniques [Bib81, And81, Wal87]. These comments apply equally to tableau systems.

1.2.2 Sequent and Tableau Systems As Refutation Procedures

When used in a backward manner, sequent and tableau systems are disproof or refutation procedures. Refutation procedures begin by assuming that the formula A which we wish to test for theoremhood is not a theorem. Then a sequence of steps (backward sequent rules in this case) are applied to verify this assumption. If all such steps lead to contradictions then the disproof procedure has failed — and hence A is actually a theorem.

Sequent systems are important because if no such sequence leads to a contradiction, then we usually have enough information to construct a network of possible worlds where one of the possible worlds falsifies A — thus demonstrating that A is not valid. That is, even though sequent systems are purely syntactic in nature, they can be related to the semantic concept of validity. Furthermore, if the axiomatic and sequent formulations correspond to the *same* semantics then a failed attempt to disprove A actually gives an *axiomatic* proof of A as a by-product. That is, the sequent system is a semantic disproof procedure as well as an axiomatic proof procedure in this case.

1.2.3 Analytic Sequent Systems, Subformula Property and Cut

Sequent rules are conditional inference rules consisting of a premiss (or condition) and a conclusion. Given a formula A, a given sequent system induces a search space consisting of many different attempted disproofs because more than one rule is usually applicable at any given stage of the disproof procedure. The disproof procedure may even cycle when a particular state reappears. But a decision procedure must terminate with an answer in a finite number of steps for any finite formula A. For sequent systems this means two things: first, that in any particular disproof attempt, each sequence of rule applications must terminate; and second, that the number of disproofs (in the search space) must be finite.

The (inference) rules of the sequent system for propositional classical logic all have a particularly simple form in that at least one formula in the premiss is always *strictly* simpler than some formulae in the conclusion. That is, each sequence of backward rule applications for classical propositional logic terminates because each (backward) rule application strictly simplifies the job at hand. Hence, cycles cannot arise, and eventually we are left with only atomic components of the initial formula A, to which no rules are applicable. Furthermore, there are only a finite number of disproofs because the formulae that may appear in a disproof are all subformulae of A, and this set is finite.

This "subformula property" is *not* the main reason why sequent systems give decision procedures. The critical point is that it is practical to use a sequent rule in a backwards manner only when the formulae in the premiss are syntactically obtainable from the formulae in the conclusion in some *predetermined* or *analytic* way. The subformula property immediately guarantees this, but it is possible to give up the subformula property and still retain this analytic notion, and hence retain a bounded search space.

What we find is that for many modal logics, we can use certain "superformulae" and still retain decidability. Of course we may get cycles but we can monitor these by keeping track of all previous states. But if a formula A' in the premiss is obtained from a formula A in the conclusion by building up A, rather than by breaking it down, then A' must be forbidden from spawning an A'' in a similar building up process. For otherwise, we would get an infinite chain of formulae, each built up from the previous one. That is, for any particular formulae, the building up procedure must be "once off" so that any particular sequence of rule applications either terminates via a cycle, or terminates because no further rules are applicable. If this restriction is met then we can start a disproof procedure for some fixed formula A and know that the formulae involved will always come from some finite set X_A depending only on A itself and the (predetermined and fixed) building up rules we allow. We can then determine X_A before embarking on the disproof procedure for A, putting a bound on the formulae that can appear in such disproofs, and hence guarantee that there are only a finite number of disproofs that we need to check.

A very powerful (inference) rule of Gentzen's original sequent system, called the cut rule, does not obey this analytic principle since its premiss contains a formula that is totally arbitrary — a guess as it were – having no relation to the formulae in the conclusion. Gentzen's seminal result that the cut rule is *redundant* in his system is at the heart of this dissertation. That is, we seek sequent systems for modal logics that are also *cut-free* since then we have a handle on the formula that the disproof procedure must consider, once we are given the formula A we wish to disprove. As we shall see, not all of our systems obey the subformula principle. But even when they do not, the superformulae that need to be considered are always analytic. In fact, we will even find that some of our systems require cut! But fortunately the uses of the cut rule can be made analytic, and this is the important aspect.

1.3 Consistency, Deducibility, Weak Completeness and Strong Completeness

In most applications, we require that the underlying logic be free from contradictions, although note the recent work on para-consistent logics [dA81, Sub89]. For our purposes, a syntactically formulated logic **L** is consistent if there is no formula A such that both A and $\neg A$ are theorems of **L**. In dealing with logics, the term "complete" is used in different ways and the terms "strongly complete" and "weakly complete" are often used to disambiguate these meanings [Fit83].

A syntactic formulation is "weakly complete" if every valid formula is syntactically derivable as a theorem (either backwards or forwards depending on which syntactic formulation we work with). Weak completeness guarantees that validity in a semantic formulation corresponds to theoremhood in a syntactic formulation.

A syntactic formulation is sometimes said to be "strongly complete" or Post-complete if it is consistent but adding any non-theorem A to it as an axiom makes it inconsistent [HC68, HC84]. Most of the modal logics we consider are not strongly complete in this sense, so this meaning is not the one of most importance [HC68, page 20]. In any case, this notion of "strong completeness" involves no semantics.

There is a stronger semantic notion than validity called logical consequence. In classical propositional logic we say that A is a logical consequence of a set X if any truth assignment that assigns "true" to every formula in X also assigns "true" to A. Correspondingly, there is a stronger syntactic notion than theoremhood called deducibility. In propositional classical logic, we say that A is deducible from a set X if taking the members of X as hypothetical theorems allows us to obtain a syntactic proof of A. These stronger notions can also be made to correspond to each other via the notions of "strong soundness" and "strong completeness". That is, we say that a syntactic classical system is "strongly sound" and "strongly complete" with respect to the semantics if deducibility corresponds to logical consequence.

In a modal semantic setting, the concept of "logical consequence" becomes more complicated for there are two extreme ways to interpret the statement "A is a logical consequence of X". One is that if some particular world of a network assigns "true" to all members of X then it must assign "true" to A. The other is that if all worlds of a network each assign "true" to X, then they must each assign "true" to A. One is like "local logical consequence" while the other is like "global logical consequence". Fitting [Fit83] actually explores both extremes, and even mixtures of the two notions, and extends his tableau systems by assuming some formulae to be "global" assumptions and others to be "local" assumptions with respect to the semantic network of possible worlds. He therefore defines a compound modal notion of "strong completeness".

There are two reasons why we ignore these stronger notions. First, most of the known axiomatic completeness results for modal logics are weak completeness results and we use these to obtain the desired correspondence between axiomatic and sequent systems. Second, the strong completeness notion is easier to handle in tableau systems and axiomatic systems than in sequent systems. We prefer to avoid the complications this causes when trying to relate sequent, axiomatic and semantic systems to these tableau systems. Because of these different nuances and complications with "strong completeness" we deal only with "weak completeness" where our only goal is to show that validity corresponds to theoremhood. So from now on, *completeness means weak completeness.*

1.4 Proving Soundness and Completeness

As the last two sections have shown, the crux of the matter is proving the correspondences between the two syntactic formulations and the semantic formulation via the soundness and (weak) completeness results. For axiomatic systems, soundness and completeness are usually easy although for some logics the proofs can get involved [Gol87]. For sequent systems, soundness is usually not a problem, but for most sequent systems, proving completeness without resorting to the cut rule can be quite hard, and sometimes, impossible. As Dana Scott observes "Gentzen's Elimination Theorem holds only for very special relations" [Fit83]. And we have already seen that cut is an unacceptable rule for automated deduction.

Our tasks neatly split into two parts. We have to prove the completeness of axiomatic systems with respect to Kripke semantics and for this the method of Henkin is the most common [Hen49]. We also have to prove the completeness of sequent systems with respect to Kripke semantics and for this the method of Hintikka is the most common [Hin55]. We begin with classical propositional logic and introduce the modal complications afterwards.

1.4.1 Tableau and Sequent Completeness

A set X is said to be **downward saturated** if it meets the following conditions [Hin55, BS84]:

1. if $\neg A \in X$ then $A \notin X$;
2. if $A \wedge B \in X$ then $A \in X$ and $B \in X$;
3. if $A \vee B \in X$ then $A \in X$ or $B \in X$;
4. if $A \Rightarrow B \in X$ then $\neg A \in X$ or $B \in X$;
5. if $\neg\neg A \in X$ then $A \in X$;
6. if $\neg(A \wedge B) \in X$ then $\neg A \in X$ or $\neg B \in X$;
7. if $\neg(A \vee B) \in X$ then $\neg A \in X$ and $\neg B \in X$;
8. if $\neg(A \Rightarrow B) \in X$ then $A \in X$ and $\neg B \in X$.

If we start with a finite set Y then we can form a downward saturated set Y^* by first including Y in Y^* and then repeatedly adding to Y^* the formulae necessary to ensure that it satisfies conditions 2 to 8. If, for example, $A \wedge B \in Y^*$ then we must ensure that both A and B are in Y^*, adding them if they are not. But note that the first condition must also be satisfied for a set to be downward saturated and it is the only condition that requires that some formula is *not* in the set. It is a *consistency* condition. That is, these conditions give a recipe for checking the consistency of a finite set because the downward saturation process will terminate if the initial set Y is finite, the consistency condition (1) can be checked easily since Y^* then contains only a finite number of formula, and most importantly, the downward saturation procedure preserves consistency [BS84]. This is essentially the soundness of these rules with respect to classical truth valuations.

Recall that a classical propositional truth assignment assigns truth values to each formula of our language. The beauty of downward saturated sets is that if Y^* is a downward saturation of a finite set Y then the truth assignment:

$$A \text{ is "true" if } A \in Y^* \text{ and } A \text{ is "false" if } A \notin Y^*$$

is *guaranteed* to be a coherent and consistent one. That is, a purely *syntactic* procedure, as embodied by conditions 1-8, can be used to obtain a *semantic* truth assignment for a consistent set Y. (Actually we only need to assign truth values to the atomic components.)

Now suppose we take a finite set $Y \cup \{\neg A\}$ and attempt this downward saturation procedure. If we succeed then we know that there is at least one truth assignment that makes all the members of Y "true" and makes A "false". Thus, A is not a logical consequence of Y. On the other hand, if *all* attempts to obtain a (consistent) downward saturation of $Y \cup \{\neg A\}$ fail, because they all break condition 1, then we know that *any* truth assignment that tries to assign "true" to all the members of Y *must* assign "true" to A. That is, A is a logical consequence of Y. In general, if we want to know whether A is a logical consequence of Y then we have to test $Y \cup \{\neg A\}$ for consistency and we do this by attempting to form a downward saturation of $Y \cup \{\neg A\}$, highlighting that downward saturation is a refutation procedure. We obtain the weaker notion of tautology simply by taking Y to be the empty set and testing the singleton $\{\neg A\}$ for consistency when we want to test whether A is a classical tautology.

This is essentially the strong and weak completeness of the tableau and sequent systems for classical propositional logic because the tableau and sequent systems are essentially nothing but downward saturation procedures. That is, a formula A is a theorem of a tableau formulated logic if $\{\neg A\}$ is inconsistent. Hence, if $\{\neg A\}$ is consistent, then the above truth assignment gives a valuation that falsifies A, proving that A is not "true" in all truth assignments and hence is not a tautology.

Notice that the definition of downward saturation obeys a subformula principle since the conclusions of the conditions 2-8 involve formulae that are strictly simpler than the formula in the antecedent, and furthermore, they are always obtained from the formulae in the condition in some fixed analytic way. Condition 6, for example, instructs us to add one of $\neg A$ or $\neg B$ to X and both $\neg A$ and $\neg B$ are strictly simpler than $\neg(A \wedge B)$.

Now for modal logics, we have to handle formulae like $\Box A$ and $\Diamond A$. Recall that Kripke semantics involve a network of nodes called possible worlds, each with its own truth assignment. We therefore work with a network of saturated sets instead of just one saturated set and a possible world assigns "true" to all the formulae in the set associated with it. We can then use the associated set as a name for the possible world. Hintikka [Hin63] shows that the new rules to handle modal formulae take the following forms:

$$\text{if } \Box A \in Y \text{ then } A \in Z \text{ for all nodes } Z \text{ reachable from } Y$$

$$\text{if } \Diamond A \in Y \text{ then } A \in Z \text{ for some node } Z \text{ reachable from } Y.$$

Notice that these conditions also have the property that the formula A in the conclusion is strictly simpler than the formulae ($\Box A$ or $\Diamond A$) in the condition, and are obtained from the condition in some fixed analytic way.

By using these extra modal conditions (and others) we are able to check the modal consistency of a given set X. If X happens to be consistent then this method yields a network of saturated sets, one of which makes all the formulae in X "true". Again, the modal tableau methods we use are nothing but a systematised way of performing this task and hence the completeness result follows.

For some of our modal systems we find conditions like:

$$\text{if } \Diamond A \in X \text{ then } \Box\Diamond A \in X$$

where the conclusion involves a superformula ($\Box\Diamond A$) of the condition ($\Diamond A$). Now if there were another saturation condition that allowed us to build up $\Box A$ into $\Diamond\Box A$ then we would be in trouble. For then these pair of rules would add formulae to X indefinitely (given that $\Diamond A \in X$) via the chain $\Diamond A, \Box\Diamond A, \Diamond\Box\Diamond A, \Box\Diamond\Box\Diamond A, \cdots$ This is what was meant before by the restriction that all building up rules must be "once off". And we find that, fortunately, for a large number of modal logics, we are able to use "building up" rules that are "once off" or analytic, in this way. Another example of a bad rule would be:

$$\text{if } A \in X \text{ then } \neg\Box A \in X$$

since this rule feeds upon itself to give an infinite chain of additions $A, \neg\Box A$, $\neg\Box\neg\Box A$, $\neg\Box\neg\Box\neg\Box A, \cdots$ given that $A \in X$.

The crux of this dissertation is to show that we can find analytic "once off" rules for many modal logics although some of these rules are quite bizarre. The modal tableau method we use is based on networks of finite saturated sets following Hintikka [Hin55] and Rautenberg [Rau83] and yields decision procedures for our logics in the process precisely because all our rules are analytic. Due to the intimate relationship between tableau and sequent systems it is easy to obtain an analogous sequent system from each tableau system. Then we can use these sequent systems to determine (syntactic) theoremhood. But notice that our systems are effective only if we deal with *finite* sets since the tableau procedures are guaranteed to terminate only for finite sets. This is clearly a reasonable restriction for automated theorem proving.

1.4.2 Axiomatic Completeness

We say that a set X is **upward saturated** if it satisfies the converses of conditions 1-8 where the converse of "if A then B" is "if B then A". Note that the converse of condition 1 allows us to add $\neg A$ to X if $A \notin X$. This means that the upward saturation procedure is not *effective,* in that it runs for ever, since there will be an infinite number of formulae outside X, for any X.

We say that a set X is **maximal consistent** if it is saturated both upwards and downwards. Maximal means that for any formula B, we have $B \in X$ or $\neg B \in X$. Consistent means that $B \in X$ iff $\neg B \notin X$. Thus, X is maximal consistent if for *every* formula B, either $B \in X$ or $\neg B \in X$ but not both. Hence, maximal consistent sets are infinite.

To prove axiomatic completeness we again associate sets with worlds. But this time we associate maximal consistent sets instead of downward saturated sets. We already know that for any formula B, either B or $\neg B$ is in any maximal consistent set. The relevant truth assignment is:

$$A \text{ is "true" if and only if } A \in X.$$

The usual method to prove semantic completeness of axiomatic systems is via the method of canonical models and filtrations, using maximal consistent sets [LS77, HC84]. But since we make no use of canonical models or filtrations, we omit the details. Many axiomatic completeness results have been obtained using canonical models and filtrations [Seg71] but unfortunately, this method does not yield a decision procedure for the logic (when the logic is decidable). Hughes and Cresswell [HC84] give an introduction to these concepts.

1.4.3 Abstract Consistency Properties

Fitting [Fit83] tries to get the best of both methods by using the "maximal consistent" set method to prove the strong completeness (in his sense) of his modal tableau systems. His definition of "maximal consistent", however, is slightly different although his technique of abstract consistency properties is more general. In fact, our methods seem extremely low level and technical in comparison. But we can obtain finer grained proofs of completeness and decidability specifically because our methods are so low level. The essential ideas, once worked out, can be lifted to give abstract consistency properties if desired.

1.4.4 Deducibility Relations

Actually, it is possible to prove the correspondence between the three formulations *simultaneously* using deducibility relations [Rau83]. However, this requires us to prove soundness in a purely syntactic way rather than in a semantic way. For the better known logics like **S4** and **S5** this is not a problem. But for the more unusual logics like **G** and **Grz** and especially for **S4.3**, **S4.3.1** and **S4.14** these proofs are not at all trivial (c.f. Zeman's proof for **S4.3** [Zem73, page 236]). Consequently, we use semantic proofs of tableau (and hence sequent) soundness and completeness, and use the known axiomatic completeness results of Segerberg [Seg71] to finalise our correspondence between the axiomatic and sequent formulations.

1.5 Decidability

Soundness and completeness results are all well and good, but for automated deduction, the prime issue is to find a decision procedure allowing us to determine theoremhood. Our tableau and sequent systems provide nondeterministic decision procedures for the logics they formulate because the completeness proofs yield an essential property of a logic called the finite model property. The finite model property means that if A is not a theorem of the logic, then there exists a finite network of possible worlds that contains a world falsifying A. We have already seen that the tableau procedure is an attempt to construct a network of possible worlds where some world falsifies A. When such a network is demonstrably finite, for any A, we immediately obtain the finite model property.

Another advantage of our method is that all our completeness proofs are constructive. Hence each proof gives a *deterministic* decision procedure for the logic in question. That is, instead of using the tableau systems as nondeterministic decision procedures, we can use the steps employed in the completeness proofs to obtain a deterministic decision procedure.

Nevertheless, the nondeterministic sequent systems may be applicable as *efficient* decision procedures directly using a parallel model of computation of Clocksin where nondeterminism is "a good thing" [Clo87, Kle90].

1.6 Syntactic Cut Elimination

Note that the "proper" way to prove cut-elimination is to use direct *syntactic* arguments as was originally done by Gentzen, but this is usually quite difficult [Gen35, Sza69, Val83]. We therefore use the bridge between sequent systems and axiomatic systems provided by Kripke semantics and "eliminate" cut simply by leaving it out from the start (where possible). The semantic completeness proofs then give us semantic cut-elimination theorems for the logics with cut-free sequent systems and pave the way for syntactic cut-elimination proofs since we know that the syntactic proofs exist!

1.7 Summary

The three well known formulations of modal logic we deal with are the axiomatic formulation, the sequent formulation and the Kripke formulation. The first two are syntactic while the third is semantic. The important syntactic concepts are theoremhood and deducibility. The important semantic concepts are validity and logical consequence. Weak soundness and completeness results allow us to equate validity and theoremhood. Strong soundness and completeness allow us to equate logical consequence and deducibility. We prefer to deal with the weaker issues because of various complications involved in using the stronger notions in modal sequent formulations.

Axiomatic completeness with respect to Kripke semantics and sequent completeness with respect to (the same) Kripke semantics must be proved separately in order to obtain a correspondence between the axiomatic and sequent formulations. Fortunately, half of this work has been done for us since axiomatic soundness and completeness results are known for most of the logics we deal with, so we have to worry only about the sequent soundness and completeness results.

Tableau systems are direct analogues of sequent systems but are more amenable to completeness proofs using downward saturated sets. We therefore base our proofs on tableau systems.

The sequent formulation is amenable to automation on computers because of a crucial property of analycity which usually manifests itself via the subformula property and the eliminability of the cut rule. For some of our modal logics we find that we have to give up the subformula property. Fortunately though, the superformulae we introduce obey an analytic principle. Some of our tableau systems even require cut, but the applications of the cut rule can always be made analytic. The constructive completeness proofs yield two types of decision procedures for our logics via the finite model property. One is the tableau system itself and is highly nondeterministic. The other is obtained from the completeness proof and is deterministic.

1.8 Contributions

We present a uniform treatment of sequent and tableau systems for modal logics, showing (the well known fact) that the two are essentially the same. The methods we use are originally due to Hintikka [Hin55] and Beth [Bet55] but are lifted from Rautenberg [Rau83]. This work is in no way original but is included to make the dissertation self contained and also to pinpoint a slight omission of several authors in omitting a structural rule known as contraction. Recent work of Girard [Gir87] has shown the importance of such structural rules. As a result, we unify and extend the work of Hanson [Han66a], Segerberg [Seg71], Mints [Min70], Fitting [Fit83], Rautenberg [Rau83] and Shvarts [Shv89], obtaining sound and complete sequent and tableau systems for the propositional normal modal logics **K**, **T**, **D**, **K4**, **D4**, **S4**, **K45**, **K45D**, **B**, **S5**, **G**, **Grz**, and obtaining several different tableau systems for some of these logics. In the process, we present a new and shorter semantic proof of an embedding of **S5** into **S4**, and also give an embedding of **S5** into **K45** due to Shvarts.

Most of these systems are cut-free but even those that are not require only an analytic form of the cut rule which remains totally amenable for computer implementation.

We then give sound, complete and cut-free sequent and tableau systems for the propositional modal logics **S4.3**, **S4.3.1** and **S4.14**. These logics are of particular importance when the modality □ is given a temporal interpretation because they respectively model time as a linear but dense sequence of points, a linear but discrete sequence of points and as a branching but discrete tree of points.

I had thought that all this work was original until I found that my system for **S4.3** was discovered by Zeman [Zem73] almost 20 years ago in tableau form. Zeman, however, is unable to give the corresponding cut-free sequent system. I know of no other cut-free sequent or tableau systems for the logics **S4.3.1** and **S4.14** although Bull [Bul85] erroneously credits Zeman [Zem73] with the discovery of a tableau system for **S4.3.1** (which Bull, following Prior, calls **D**). Once I found Rautenberg's paper [Rau83], however, it was obvious to me that my formulations were but mere extensions of his method. In fact, Rautenberg [Rau83, page 414] clearly states that his method can be extended to handle other well known modal logics with the finite model property. He even refers to "a simple tableau" system for **S4.3** but does not give details since his main interest is in proving interpolation, and **S4.3** lacks interpolation. In subsequent personal communications I have been unable to ascertain the **S4.3** system to which Rautenberg refers [Rau90].

In the further work sections we sketch incomplete work and suggest further work of direct relevance to automated deduction in modal logics.

(a) we present incomplete work aimed at obtaining tableau systems for the following logics: **KB**, **S4Dum**, **K4DLZ**, **KGL**, **K4L**, $\mathbf{G_o}$, **S4M**, **S4.4**, **S4Zem**, **KLGrz**;

(b) we briefly mention incomplete work on extending the tableau method to propositional tense logics, to propositional discrete linear temporal logic **DX**, to propositional discrete branching temporal logic **CTL**, and to first order modal logics;

(c) we suggest how to obtain syntactic cut-elimination proofs for the modal logics we consider.

1.9 Dissertation Outline

Chapter 2 is a brief introduction to propositional normal modal logics. We cover the syntax, axiomatic formulations and Kripke semantics for these logics and borrow heavily from the lecture notes of Goldblatt [Gol87] and the excellent introductory text by Hughes and Cresswell [HC84]. We sketch how to relate the axiomatic formulations with Kripke semantics via soundness and completeness theorems but do not present detailed proofs. Instead, we catalogue many well known characterisation results, usually due to Segerberg [Seg71], and give pointers to the literature where proofs can be found.

Chapter 3 is an introduction to sequent systems for modal logics. We begin with the syntax of sequent systems, discuss the structural rules and highlight the importance of the contraction rule in modal sequent systems. We introduce Kripke semantics for modal sequent systems, define soundness and completeness, but do not prove specific results.

Chapter 4 is an introduction to tableau systems for modal logics. We show the correspondence between sequent and tableau systems via the Kripke semantics for sequent systems introduced in Chapter 3. We prove soundness and completeness results for the modal tableau system for **S4** as a generic example of the method of Hintikka [Hin55, Hin63] and Rautenberg [Rau83].

Chapter 5 is a uniform treatment of known tableau systems for the propositional normal modal logics **K**, **T**, **D**, **K4**, **D4**, **S4**, **K45**, **K45D**, **B**, **S5**, **G** and **Grz**. Alternative systems for **D**, **S4**, **K45**, **K45D** and **S5** are also given there. Most of these systems are cut-free but even when cut is required, it is always an analytic cut rule which remains amenable for computer implementations.

Chapter 6 is a detailed description of cut-free tableau systems for the propositional normal modal logics **S4.3**, **S4.3.1**, **S4.14** and brief descriptions of various extensions of **S4** having intuitive temporal interpretations.

Chapter 7 is a brief (historical) review of related work in sequent and tableau systems for modal logics, supplementing the specific bibliographic remarks made at various points in the dissertation.

Chapter 8 is a brief description of further work and incomplete work. Some of this work is not related to the main theme of the dissertation, but (hopefully) is of independent interest.

Chapter 9 contains the conclusions.

Chapter 2

Propositional Normal Modal Logics

The first part of this chapter is an introduction to the syntax, axiomatic bases and Kripke semantics for various propositional normal modal logics. The syntactic or proof-theoretic notions of axiom, axiomatic basis, theorem and proof are all defined, as are the semantic or model-theoretic notions of frames, models, satisfiability and validity. These syntactic and semantic concepts are then related via the notions of soundness and completeness. This material is based on the excellent introductory texts by Hughes and Cresswell [HC84, van86] and Goldblatt [Gol87].

The latter parts of this chapter are devoted to various properties of propositional normal modal logics which are of primary importance for automated deduction in these logics. The concepts of decidability, semi-decidability, finite axiomatisation, finite model property and finite frame property are all defined. It is known that these concepts are intimately related for propositional normal modal logics, so the relationships are stated formally but not proved.

The most common method to prove axiomatic completeness results is to use a technique from classical logic due to Henkin [Hen49] involving infinite maximal consistent sets of formulae [Gol87, HC84]. But to establish decidability, one must work with finite models and the most common way to obtain finite models from these infinite, maximal consistent sets is the filtration method [LS77, Seg71]. However, as we make no use of filtrations, or of maximal consistent sets, we omit the necessary definitions.

Instead, the chapter ends with a catalogue of known characterisation results for many axiomatically formulated propositional normal modal logics, concentrating on finite frames. These results, which are taken from various sources, were invariably obtained by using maximal consistent sets and filtrations. But we require the results only as our main interest is to show that our sequent and tableau systems characterise these same logics.

The brevity is justified as we are primarily interested in proof procedures for proving that a formula either is or is not a theorem of some modal logic and as we have seen, axiomatic systems are not good proof procedures.

2.1 Syntax: Propositions, Formulae and Subformulae

The sentences of modal logics are built from

(i) a denumerable non-empty set of primitive propositions $\mathcal{P} = \{p_1, p_2, \cdots\}$;

(ii) the classical binary connective $\wedge$ and the classical unary connective $\neg$;

(iii) the unary modal connective $\Box$;

(iv) the punctuation marks) and (.

A well-formed formula, hereafter simply called a **formula**, is any sequence of symbols obtained from the following rules:

(v) any $p_i \in \mathcal{P}$ is a formula and is usually called an **atomic formula**;

(vi) if A and B are formulae, so are $\neg A$, $\neg B$, $A \wedge B$, $\Box A$ and $\Box B$.

Then the other usual connectives are defined as abbreviations

$$(A \vee B) = (\neg(\neg A \wedge \neg B));$$

$$(A \Rightarrow B) = (\neg(A \wedge \neg B));$$

$$(\Diamond A) = (\neg\Box\neg A)$$

where the $=$ sign is merely a meta-linguistic notation.

Lower case letters like p and q denote members of $\mathcal{P}$. Upper case letters from the beginning of the alphabet like A and B together with P and Q (all possibly annotated) denote formulae. Upper case letters from the end of the alphabet like X, Y, Z (possibly annotated) denote *finite* (possibly empty) sets of formulae. The set of all formulae is denoted by **Fml**.

The symbols $\neg, \wedge, \vee$ and $\Rightarrow$ respectively stand for logical negation, logical conjunction, logical disjunction and logical (material) implication. The connectives $\neg, \Box, \Diamond$ are of equal binding strength but bind tighter than $\wedge$ which binds tighter than $\vee$ which binds tighter than $\Rightarrow$. So $\neg A \vee B \wedge C \Rightarrow D$ should be read as $((\neg A) \vee (B \wedge C)) \Rightarrow D$. The symbols $\Box$ and $\Diamond$ can take various meanings but traditionally stand for "necessity" and "possibility". In the context of temporal logic, they stand for "always" and "eventually" so that $\Box A$ is read as "A is always true" and $\Diamond A$ is read as "A is eventually true".

There are many ways of defining the syntax of modal logics depending on what one takes as primitives and what as abbreviations. Fitting [Fit83], for example, prefers to take both $\Box$ and $\Diamond$ as primitives whilst Goldblatt [Gol87] takes $\Box$ and $\Rightarrow$ as primitives and introduces a special symbol $\bot$, denoting a constant false proposition, to then define negation. In most cases, the particular choice is not important although note that the lattice of logics that each choice induces can be different [Mak73].

The set of all **subformulae** of a formula, or of a set of formulae, is used extensively. For a formula A, the *finite* set of all subformulae $Sf(A)$ is defined inductively as [Gol87]:

$Sf(p) = \{p\}$ where $p \in \mathcal{P}$ is an atomic formula;

$Sf(\neg A) = \{\neg A\} \cup Sf(A)$;

$Sf(A \wedge B) = \{A \wedge B\} \cup Sf(A) \cup Sf(B)$;

$Sf(\Box A) = \{\Box A\} \cup Sf(A)$.

For a finite set of formulae X, the set of all subformulae $Sf(\mathrm{X})$ consists of all subformulae of all members of X; that is, $Sf(X) = \bigcup_{A \in X} Sf(A)$. Note that under this definition, a formula must be in primitive notation to obtain its subformulae; for example, $A \vee B$ must be written as $\neg(\neg A \wedge \neg B)$ to obtain its subformulae.

The set of **strict subformulae** of A is $Sf(A) \setminus \{A\}$.

2.2 Normal Modal Logic: An Axiomatic View

The traditional way to formulate a logic is the axiomatic method described below. Because of the long tradition of axiomatic systems, we assume that all our logics are axiomatically formulated.

An **axiomatic system** is a finite set of formulae called axioms, together with a finite set of rules called inference rules. Each axiomatic system gives rise to a set of formulae $\mathbf{L}$ as we describe below and the axiomatic system is said to be a system for $\mathbf{L}$.

Suppose p is an atomic formula and that p appears (possibly more than once) in a formula A. If we uniformly replace every occurrence of p in A by another formula B giving a formula A' then A' is said to be a **substitutional instance** of A.

An axiomatically formulated logic $\mathbf{L}$ is simply some subset of **Fml** that obeys (is closed under) certain (**inference**) rules governing membership of $\mathbf{L}$. In an axiomatic normal modal setting, the usual rules are known as the rule of uniform substitution **US**, the rule of detachment or modus ponens **MP**, and the rule of necessitation **RN**, as shown below:

US: if $A \in \mathbf{L}$ then $A' \in \mathbf{L}$ where A' is any substitutional instance of A;

MP: if $A \in \mathbf{L}$ and $(A \Rightarrow B) \in \mathbf{L}$ then $B \in \mathbf{L}$;

RN: if $A \in \mathbf{L}$ then $\Box A \in \mathbf{L}$.

These inference rules are all conditional statements and so we require some initial base to which we can apply the inference rules to obtain **L**. The **axiomatic basis** or **axiomatisation** of a logic **L** is the (usually finite) set of formulae that are deemed to be in **L** by definition. The members of such a basis are called the **axioms** of **L**. Different axiomatic bases can give the same logic and so we are usually interested in a minimal set of axioms to form our axiomatic bases.

By the rule of uniform substitution, any axiom of **L** automatically brings all its substitutional instances into **L**. But for historical reasons, where the rule of uniform substitution is sometimes omitted, it has become customary to use formulae like $\Box A \Rightarrow A$ as axiom schemas instead of formulae like $\Box p \Rightarrow p$. By substituting any formula for A, we obtain an **instance** of the axiom schema $\Box A \Rightarrow A$.

An **axiomatic derivation** is a finite sequence of formulae where each formula is either an instance of an axiom, or results from formulae earlier in the sequence by an application of one of the inference rules. If the last formula in an axiomatic derivation is A, then the derivation is an **(axiomatic) proof of** A. So each instance of an axiom constitutes a one line proof of itself, by definition. A formula A is a **theorem** if and only if there is a proof of A and it should be clear that a logic **L** is exactly the set of theorems derivable from its axiomatisation via the rules of inference. We write $\vdash_{\mathbf{L}} A$ to mean "A is a theorem of logic **L**". Formally,

$$\vdash_{\mathbf{L}} A \text{ if and only if } A \in \mathbf{L}.$$

Although different normal modal logics have different axiomatisations, propositional normal modal logics are all extensions of propositional classical logic **PC**, so a minimum requirement is that propositional classical logic be included. The easiest way to do so is to define each logic to contain **PC** since **PC** is just a subset of **Fml** [Gol87]. The alternative, which we take, is to include some minimal set of axioms from which **PC** can be obtained by the three rules of inference. Thus we take the following axiomatisation of **PC** from Goldblatt [Gol87, page 17] and assume that all our axiomatic bases contain these formulae as axioms:

$A \Rightarrow (B \Rightarrow A)$;

$(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$;

$\neg\neg A \Rightarrow A$.

The final step is to introduce the axiom of "normality", named K, in honour of Saul Kripke. The modal logic **K** has the following lone axiom

$$K : \Box(A \Rightarrow B) \Rightarrow (\Box A \Rightarrow \Box B)$$

in its axiomatic basis (as well as the **PC**-axioms of course). That is, **K** is the smallest subset of **Fml** that includes **PC**, includes K and is closed under the rules **RN**, **US** and **MP**. The logic **K** is known as the minimal **normal** modal logic and any logic obtained by extending this set of axioms, and retaining the three rules of inference, is said to be a **normal modal logic.** Note that the choice of inference rules is crucial in modal logics as the omission of **RN** can lead to non-normal (or quasi-normal) extensions of **K** [Seg71, page 171]. In this dissertation, only normal modal logics are treated; see Segerberg [Seg71], Fitting [Fit83] or Chellas [Che80] for non-normal logics.

2.2.1 Axiom Names and Logics

The logics we study and the axioms defining them have been extensively studied in the literature and have acquired weird and wonderful names. The most sensible naming convention is due to Lemmon and Scott [LS77] where each axiom is given a name, and the name of a logic is formed by concatenating the names of its axioms to the prefix K to highlight the fact that the logic is normal. Thus, if $A_1, A_1, \cdots, A_k$ are the axioms in the axiomatic basis for a logic, then the name of the logic is $\mathbf{KA_1A_1 \cdots A_k}$. Figure 2.1 (page 23) is a catalogue of our names for the axioms we require and their common names in the literature. Figure 2.2 (page 24) is a catalogue of well known logics with axiomatic bases drawn from this set of axioms. Note that some logics have more than one axiomatic basis; **S5**, for example, can also be axiomatised as any one of **KT4B**, **KT45**, **KDB4**, **KDB5** [Gol87, page 26] [LS77] [HC84].

Our naming convention is a mixture of tradition and common sense in that we use the traditional names for the most well known logics but use the Lemmon notation for the more obscure logics. Most traditional names are harmless as they do not conflict with the Lemmon notation. However, the traditional names **K4.3**, **D4.3**, **K4.3W** which appear in various texts (last column of Figure 2.2) are downright misleading in the light of the Lemmon notation since these logics are respectively axiomatised as **K4L**, **K4DL**, **K4GL** and L and 3 are *different* in the absence of T. Similarly, the logic named **S4.1** is actually axiomatised as **KT4M**. We therefore avoid these misleading traditional names and use the stated Lemmon names instead.

2.3 Kripke Semantics for Modal Logics

2.3.1 Possible Worlds, Accessibility, Models and Frames

The well known semantics for classical propositional logic is based on interpretations that assign true or false to each formula by assigning true or false to each atomic formula and simultaneously respecting the semantics of the classical connectives. The corresponding notion in modal logic is to imagine a set of "possible worlds" interconnected in some way; imagine a graph of nodes (worlds) with arcs denoting the interconnections. Each world is like a classical propositional interpretation, assigning true or false to each non-modal formula, but the truth values of modal formulae are determined by higher level conditions involving the interconnectivity of the graph of possible worlds. This is the

Axiom Name	Defining Formula	Alternative Names
K	$\Box(A \Rightarrow B) \Rightarrow (\Box A \Rightarrow \Box B)$	
T	$\Box A \Rightarrow A$	M [Rau83]
D	$\Box A \Rightarrow \Diamond A$	
4	$\Box A \Rightarrow \Box\Box A$	
B	$A \Rightarrow \Box\Diamond A$	
5	$\Diamond A \Rightarrow \Box\Diamond A$	E [Seg71]
M	$\Box\Diamond A \Rightarrow \Diamond\Box A$	G [Seg71]
L	$\Box((A \wedge \Box A) \Rightarrow B) \vee \Box((B \wedge \Box B) \Rightarrow A)$	H [Che80], Lem_0 [Seg71]
3	$\Box(\Box A \Rightarrow B) \vee \Box(\Box B \Rightarrow A)$	H [BS84], H_0^+ [LS77],Lem [Seg71]
2	$\Diamond\Box A \Rightarrow \Box\Diamond A$	G [BS84], G [LS77]
G	$\Box(\Box A \Rightarrow A) \Rightarrow \Box A$	W [Seg71]
Grz	$\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \Rightarrow A$	J1 [HC84, page 111]
Go	$\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \Rightarrow \Box A$	Grz_1 [Seg71]
Z	$\Box(\Box A \Rightarrow A) \Rightarrow (\Diamond\Box A \Rightarrow \Box A)$	
Dum	$\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \Rightarrow (\Diamond\Box A \Rightarrow \Box A)$	Dum_1 [Seg71],$M1$ [HC68], $M14$ [Zem73]
Zbr	$\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \Rightarrow (\Box\Diamond\Box A \Rightarrow \Box A)$	
Zem	$\Box\Diamond\Box A \Rightarrow (A \Rightarrow \Box A)$	
R	$\Diamond\Box A \Rightarrow (A \Rightarrow \Box A)$	

Figure 2.1: Axiom names and alternative names.

basis of possible world semantics as proposed by Saul Kripke [Kri59, Kri63b, Kri65]; but see Bull and Segerberg [BS84], Lemmon and Scott [LS77] and Zeman [Zem73] for a historical account of the development of possible world semantics for modal logics.

A **frame** is an ordered pair $\langle W, R\rangle$ where W is a denumerable non-empty set (of possible worlds) and R is a binary relation over W. That is, R is simply a set of ordered binary tuples like (w, w') where $w \in W$ and $w' \in W$. If $(w, w') \in R$, we write wRw' and say that w' is **accessible to** or **reachable from** w. We also write $w\not{R}w'$ to mean that $(w, w') \notin R$.

A **model** is an ordered triple $\langle W, R, V\rangle$ where $\langle W, R\rangle$ is a frame and V is a mapping which assigns a subset $V(p)$ of W to each atomic formula $p \in \mathcal{P}$. Informally, $V(p)$ is the set of worlds at which p is true. Formally, $V : \mathcal{P} \mapsto 2^W$, where 2^W is the set of all subsets of W.

We often use annotated names like w_1 and w_2 to denote possible worlds. Unless stated explicitly, there is no reason why w_1 and w_2 cannot name the same world.

Name of Logic	Lemmon Name (Axiomatic Basis)	Alternative Names and Known Equivalences
K	**K**	
T	**KT**	**M** [Rau83]
D	**KD**	Deontic **T** [BS84]
K4	**K4**	
D4	**KD4**	Deontic **S4** [BS84]
S4	**KT4**	
K45	**K45**	
K45D	**K45D**	Deontic **S5**
B	**KTB**	
S5	**KT5**	
G	**KG**	**KW** [Gol87], **GL** [Val83]
Grz	**KGrz**	
S4.3	**KT43**	**KT4L** [Gol87]
S4.3.1	**KT43Dum**	**D** [HC68], [Pri57], [Zem73]
S4.14	**KT4Zbr**	
S4Grz **S4MDum**	**KT4Grz** **KT4MDum**	known to be the logic **Grz**
---	---	---
Go	**K4Go**	
Grz.3 **S4Grz.3**	**KGrz3** **KT4Grz.3**	same logic
S4M	**KT4M**	**S4.1** [MT48] [BS84]
S4Dum	**KT4Dum**	
S4Zem	**KT4Zem**	
K4L	**K4L**	**K4.3** [Seg71]
K4DL	**K4DL**	**D4.3** [Seg71]
K4DLZ	**K4DLZ**	**K4.3Z** [Seg71]
K4G	**K4G**	equal to **G** [Rau83]
KGL **K4GL**	**KGL** **K4GL**	**G.3** [Rau83], $\mathbf{GL}_{\mathrm{lin}}$ [Val86] same logic **K4.3W** [Seg71]

Figure 2.2: Traditional names of logics, axiomatic bases, alternative names and known equivalences. We give explicit tableau systems for all logics above the break in the table and discuss tableau rules for most of the others.

2.3.2 Truth, Satisfiability and Validity

The classical semantic notions of satisfiability and validity have modal analogues in Kripke semantics. A possible world w is said to satisfy an atomic formula p if and only if $w \in V(p)$, indicating that p is true in world w. We write this as $w \models p$ and write $w \not\models p$ to mean "not $w \models p$". We should really write something like $w \models_V p$ to show that the satisfaction relation $\models$ depends on V, but usually V is clear by context. The semantics of the other connectives and modal operators then follow naturally according to the following where p is an atomic formula and A and B are formulae:

$w \models p$ iff $w \in V(p)$;

$w \models \neg A$ iff $w \not\models A$;

$w \models A \wedge B$ iff $w \models A$ and $w \models B$;

$w \models A \vee B$ iff $w \models A$ or $w \models B$;

$w \models A \Rightarrow B$ iff $w \not\models A$ or $w \models B$;

$w \models \Box A$ iff $\forall w' \in W, wRw'$ implies $w' \models A$;

$w \models \Diamond A$ iff $\exists w' \in W$ such that wRw' and $w' \models A$.

In any model $\langle W, R, V \rangle$, a formula A is said to be **true in a world** $w \in W$ if $w \models A$.

A formula A is said to be **valid in a model** $\mathcal{M} = \langle W, R, V \rangle$, written as $\mathcal{M} \models A$, if it is true in all worlds in that model; that is, if $\forall w \in W, w \models A$.

A frame naturally generalises the notion of a model since augmenting any frame with a valuation V gives a model. A formula A is said to be **valid in a frame** $\mathcal{F} = \langle W, R \rangle$, written as $\mathcal{F} \models A$, if A is valid in all models based on $\mathcal{F}$; that is, if $\forall V, \langle W, R, V \rangle \models A$.

Suppose $\mathcal{C}$ is a class of models, or of frames. A formula A is said to be **valid in a class** $\mathcal{C}$, written as $\mathcal{C} \models A$, if it is valid in every member of $\mathcal{C}$.

An axiom is said to be valid in a model (valid in a frame) if all instances of that axiom have that property. If we have a set of formulae $X \subseteq$ **Fml** then $\mathcal{M} \models X$ ($\mathcal{F} \models X$) denotes that all members of X are valid in $\mathcal{M}$ ($\mathcal{F}$).

Axiom Name	Property Name	Property of R
T	Reflexive	$\forall w \in W, (wRw)$
B	Symmetric	$\forall w, w' \in W, (wRw' \Rightarrow w'Rw)$
D	Serial	$\forall w \in W \exists w' \in W$ such that (wRw')
4	Transitive	$\forall s, t, u \in W, (sRt \wedge tRu \Rightarrow sRu)$
5	Euclidean	$\forall s, t, u \in W, ((sRt \wedge sRu) \Rightarrow tRu)$
L	Weakly-connected	$\forall s, t, u \in W, (sRt \wedge sRu \Rightarrow (tRu \vee t = u \vee uRt))$

Figure 2.3: Names of axioms corresponding to conditions on R.

2.3.3 Restrictions on R

Kripke [Kri59, Kri63b, Kri65] showed that three well known normal modal logics could be given an intuitive semantics based on such models where each axiom of the modal logic corresponded directly to a certain restriction on the form of R. That is,

1. if an axiom A is valid in a frame $\langle W, R\rangle$ then R has a certain property; and
2. if the reachability relation R of a frame $\langle W, R\rangle$ has that property then the axiom A is guaranteed to be valid in $\langle W, R\rangle$.

Thus the axiom A characterises frames with a certain property. The logic **K** is known as the minimal normal modal logic because the axiom K puts no restriction on R whatsoever.

There are many formulae that correspond to restrictions on the reachability relation R. Most restrictions have names and Figure 2.3, which is based on similar tables from Goldblatt [Gol87], shows the correspondences between certain axioms and certain restrictions on R. Theorems 1 and 2 below formalise these correspondences.

Theorem 1 *Let $\mathcal{F} = \langle W, R\rangle$ be a frame. Then, for each of the properties listed in Figure 2.3, if R satisfies the property, then the corresponding axiom is valid in the frame $\mathcal{F}$ [Gol87], [LS77].*

Proof for reflexivity and T: We have to show that if we have a frame $\mathcal{F} = \langle W, R\rangle$ where R is reflexive, then $\mathcal{F} \models \Box A \Rightarrow A$.

Suppose R is reflexive and that $\mathcal{M}$ is any model based on $\mathcal{F}$. Consider any world $w_0 \in W$. We have to show that $w_0 \models \Box A \Rightarrow A$. Suppose to the contrary that $w_0 \not\models \Box A \Rightarrow A$. This means that $w_0 \models \Box A$ and $w_0 \models \neg A$. By definition of $\models$ we know that the former means: $\forall w \in W, w_0 R w$ implies $w \models A$. Since R is reflexive, $w_0 R w_0$, hence we must have $w_0 \models A$. But this contradicts our second requirement that $w_0 \models \neg A$, hence $w_0 \models \Box A \Rightarrow A$. As w_0 was an arbitrary world from W this must be true for all worlds in W. That is, $\mathcal{M} \models \Box A \Rightarrow A$. Since $\mathcal{M}$ was an arbitrary model based on $\mathcal{F}$ we must have $\mathcal{F} \models \Box A \Rightarrow A$. ●

Theorem 2 *If a frame $\langle W, R\rangle$ validates any one of the axioms in Figure 2.3, then R satisfies the corresponding property [Gol87], [LS77].*

Proof for reflexivity and T: We have to show that if $\Box A \Rightarrow A$ is valid in a frame $\langle W, R\rangle$, then R is reflexive.

Let $\mathcal{F} = \langle W, R\rangle$ be a frame and suppose that $\mathcal{F} \models \Box A \Rightarrow A$. We have to show that R is reflexive. Suppose to the contrary that R is not reflexive; that is, $\neg(\forall w \in W, (wRw))$. Since W is non-empty, this means that there is some $w_0 \in W$ such that $w_0 \not R w_0$. Define a valuation V on this frame as:

$$V(p) = \{w \in W | w_0 R w\}$$

giving a model $\mathcal{M} = \langle W, R, V\rangle$.

Now $w_0 \notin V(p)$ for any $p \in \mathcal{P}$ since we know $w_0 \not R w_0$. Thus $w_0 \not\models p$. Also, $w_0 \models \Box p$ for any $p \in \mathcal{P}$ by definition of V; that is, $\forall w \in W, w_0 R w$ implies $w \models p$. But then $w_0 \not\models \Box p \Rightarrow p$ which is an instance of $\Box A \Rightarrow A$, contradicting our assumption that $\mathcal{F} \models \Box A \Rightarrow A$. Thus R must be reflexive. •

Note that Theorems 1 and 2 relate validity of *one* axiom with a property of R. When combining these properties, some of the axioms can be replaced by simpler ones. For example, if T is an axiom then axiom L can be replaced by axiom 3 without affecting the set of theorems.

Also, note that "R is not reflexive" is *not* the same as "R is irreflexive" since there can be a middle ground where R is neither reflexive nor irreflexive. Similar warnings apply for transitivity and symmetry.

2.3.4 Properties Of R Not Corresponding To Any Axioms

As Goldblatt [Gol87, page 13] states, Theorems 1 and 2 explain why Kripke semantics have been so successful for characterising modal logics. There are, however, some desirable properties of R that do not correspond to the validity of *any* modal axiom. There are, for example, no propositional modal axioms whose validity guarantees that R is:

irreflexive:	$\forall w \in W, (w \not R w)$;	[Gol87, p. 14] [HC84, p. 47];
antisymmetric:	$\forall s, t \in W, ((sRt \wedge tRs) \Rightarrow s = t)$	[Gol87, p. 14] [HC84, p. 50] ;
asymmetric:	$\forall s, t \in W, (sRt \Rightarrow (s \not R t))$	[Gol87, p. 14] [HC84, p. 50] ;
intransitive:	$\forall s, t, u \in W, ((sRt \wedge tRu) \Rightarrow (s \not R u))$	[HC84, p. 50] ;
connected:	$\forall s, t \in W, (sRt \vee t = s \vee tRs)$	[Gol87, p. 29] .

2.3.5 First Order Definability

Each of the axioms from Figure 2.3 corresponds directly to some property of R that can be written as a formula of classical first order logic in terms of R and variables like w. Goldblatt [Gol87] shows that there exist properties of R which are characterised by an axiom but which cannot be written as first order formulae. The investigation of this notion of "first-order definability" led to the discovery that propositional modal logic is a form of second order logic [Gol87].

2.4 Soundness and Completeness

Up till now, the syntactic notion of proof and the semantic notion of validity have been kept separate. The ideal is to show that our semantic intuitions about models for a particular logic correspond to the syntactic notion of proof in that logic (but note that this is not always possible). Syntactically, we manipulate formulae, while semantically, we speak in terms of models. The two notions can be related by associating a set of formulae with each world, with the understanding that the formulae in the set are the only formulae that are "true" in that world. These concepts are formalised below.

Let $\mathcal{C}$ be either a collection of models, or of frames. Then logic $\mathbf{L}$ is **sound with respect to** $\mathcal{C}$ if every theorem of $\mathbf{L}$ is valid in each member of $\mathcal{C}$. Formally, $\mathbf{L}$ is **sound** with respect to $\mathcal{C}$ if for every formula A we have that $\vdash_{\mathbf{L}} A$ implies $\mathcal{C} \models A$ [Gol87].

Logic $\mathbf{L}$ is **complete with respect to** $\mathcal{C}$ if every formula that is valid in each member of $\mathcal{C}$ is a theorem of $\mathbf{L}$. Formally, logic $\mathbf{L}$ is **complete** with respect to $\mathcal{C}$ if for every formula A we have that $\mathcal{C} \models A$ implies $\vdash_{\mathbf{L}} A$ [Gol87].

A logic $\mathbf{L}$ is **determined** or **characterised** by a class $\mathcal{C}$ if it is both sound and complete with respect to $\mathcal{C}$; that is, when $\mathcal{C} \models A$ iff $\vdash_{\mathbf{L}} A$.

Lemma 1 *If* $\mathbf{L_1}$ *and* $\mathbf{L_2}$ *are logics determined by the same class of frames* $\mathcal{C}$*, then* $\mathbf{L_1} = \mathbf{L_2}$.

Proof: Let $\mathbf{L_1}$ and $\mathbf{L_2}$ be logics determined by some class of frames $\mathcal{C}$. Suppose to the contrary that $\mathbf{L_1} \neq \mathbf{L_2}$.

Since $\mathbf{L_1}$ and $\mathbf{L_2}$ are just sets of formulae this means that there is some formula A such that $A \in \mathbf{L_1}$ and $A \notin \mathbf{L_2}$ or such that $A \in \mathbf{L_2}$ and $A \notin \mathbf{L_1}$. We consider only the first case since the second can be proved in a similar way. That is, we have $\vdash_{\mathbf{L_1}} A$ and $\nvdash_{\mathbf{L_2}} A$.

Since A is not a theorem of $\mathbf{L_2}$, and $\mathbf{L_2}$ is determined by $\mathcal{C}$, there is some frame $\mathcal{F} = \langle W, R \rangle$ with $\mathcal{F} \in \mathcal{C}$ and $\mathcal{F} \not\models A$. That is, there is some model $\mathcal{M} = \langle W, R, V \rangle$ based on $\mathcal{F}$ with $w_0 \in W$ and $w_0 \not\models A$.

On the other hand, A is a theorem of $\mathbf{L_1}$, and $\mathbf{L_1}$ is determined by $\mathcal{C}$. Hence any frame in $\mathcal{C}$ must validate A. In particular, $\mathcal{F} \models A$. But this means that for all $w \in W$ in the model $\mathcal{M}$ chosen above we must have $w \models A$. In particular, $w_0 \models A$. Contradiction, hence $\vdash_{\mathbf{L_1}} A$ implies $\vdash_{\mathbf{L_2}} A$. The other direction is proved similarly giving $\mathbf{L_1} = \mathbf{L_2}$. •

We shall use this lemma in Section 4.6 to show that the logics defined by our sequent and tableau systems are indeed the axiomatically formulated logics we claim them to be. That is, we shall show that in each case, the axiomatic logic $\mathbf{L_1}$ and the sequent/tableau logic $\mathbf{L_2}$ are determined by the same class of frames, and hence, $\mathbf{L_1} = \mathbf{L_2}$.

Hughes and Cresswell [HC84, page 55] point out that a logic can be proved sound with respect to a certain class of frames by showing that every frame in that class validates each of the axioms of the logic. The reason is that validity in frames, unlike validity in a model, is preserved by each of the three inference rules **US**, **MP**and **RN**.

A logic is said to be **complete** iff it is characterised by some class of frames [HC84, page 55]. Note that there exist non-trivial normal modal logics that are **incomplete** in that they are not characterised by any class of frames [Gol87, page 45].

2.4.1 Finite Models, Finite Frames and Decidability

If $\mathbf{L}$ is a normal modal logic, then $\mathbf{L}$ has the **finite model property** if and only if for every formula A which is not a theorem of $\mathbf{L}$, there is a model $\langle W, R, V \rangle$ in which W is finite and

(a) there is some $w \in W$ such that $w \not\models A$;

(b) if B is a theorem of $\mathbf{L}$, then for all $w \in W$, $w \models B$ [HC84, p.136].

Part (a) simply says that if A is not a theorem of $\mathbf{L}$ then there is some world that falsifies A in some model $\langle W, R, V \rangle$. Part (b) says that this model $\langle W, R, V \rangle$ must validate all theorems of $\mathbf{L}$.

If $\mathbf{L}$ is a normal modal logic, then $\mathbf{L}$ has the **finite frame property** if and only if for every formula A which is not a theorem of $\mathbf{L}$, there is a frame $\langle W, R \rangle$ in which W is finite and

(c) there is some valuation V such that $\langle W, R, V \rangle \not\models A$;

(d) if B is a theorem of $\mathbf{L}$, then for all valuations V, $\langle W, R, V \rangle \models B$ [HC84, p.150].

Part (c) says that if A is not a theorem of $\mathbf{L}$ then there must exist a model $\langle W, R, V \rangle$ and a world $w \in W$ such that $w \not\models A$. Part (d) says that the frame $\langle W, R \rangle$ must validate all theorems of $\mathbf{L}$.

Theorem 3 *A normal modal logic has the finite frame property iff it has the finite model property [Seg71, page 33], [HC84, page 152], [Gol87, page 34].*

A logic **L** is **decidable** iff there is an effective procedure to determine, in a finite number of steps, whether any given formula A is or is not a theorem of **L** [HC84]. Note that for decidability we must be able to distinguish A as *either a theorem or a non-theorem* so that we must be able to detect both theoremhood and non-theoremhood. Usually, it is the latter that poses problems. A logic is **semi-decidable** if there is an effective procedure to determine, in a finite number of steps, whether any given formula A is a theorem. So for a semi-decidable logic, the effective procedure is permitted to run forever when A happens to be a non-theorem, but must return an affirmative answer in a finite number of steps when A happens to be a theorem.

A logic **L** is **finitely axiomatisable** iff there is a finite set of axioms which together with **US**, **MP** and **RN** yield exactly **L** [HC84].

The finite model property, and hence the finite frame property, are important because of the following theorem.

Theorem 4 *If* **L** *is a finitely axiomatisable normal modal logic which has the finite model property, then* **L** *is decidable [HC84, page 153].*

Hence, by proving the finite model property for a finitely axiomatisable logic **L**, we immediately obtain a proof of decidability. We shall see in later chapters that if the finite "axiomatisation" is via a set of tableau or sequent rules, we also obtain a decision procedure for **L**.

Hughes and Cresswell [HC84, page 154] note the following facts as a warning about what Theorem 4 does *not* say. There exist normal modal logics with the finite model property that are undecidable [Urq81]. There are finitely axiomatisable and decidable normal logics that lack the finite model property [van80]. And finally, there are decidable normal logics with the finite model property that are not finitely axiomatisable [Cre79].

2.4.2 Different Types Of Frames

In the rest of this dissertation we shall work almost exclusively with frames that are finite trees or finite linear sequences. We do not give a formal definition of a tree but stipulate that a **tree** has a unique root node, that all the other nodes of a tree (if any) are descendants of this root with no converging arcs, and that each node except the root has a unique parent node. The successor relation of the tree need not be transitive, nor reflexive. The nodes of a tree may either be worlds or collections of worlds called clusters, a term we define below. The notion of clusters makes sense only for transitive frames. So from now on, if a frame involves clusters, then it *must* be transitive.

Suppose that $\langle W, R\rangle$ is a transitive frame. Following Goldblatt [Gol87, page 52] define an equivalence relation $\approx$ over W as:

$$w_1 \approx w_2 \text{ iff } w_1 = w_2 \text{ or } (w_1 R w_2 \text{ and } w_2 R w_1).$$

Then the **R-cluster** of a world $w \in W$ is:

$$C_w = \{w' | w \approx w'\}.$$

Putting

$$C_{w_1} \trianglelefteq C_{w_2} \text{ iff } w_1 R w_2$$

gives a well-defined, transitive and antisymmetric relation between clusters. Putting

$$\begin{aligned} C_{w_1} \lhd C_{w_2} \quad & \text{iff} \quad C_{w_1} \trianglelefteq C_{w_2} \text{ and } C_{w_1} \neq C_{w_2} \\ & \text{iff} \quad w_1 R w_2 \text{ and } w_2 \not R w_1 \end{aligned}$$

defines $\lhd$ to be transitive and irreflexive, and therefore asymmetric; that is, $C_{w_1} \lhd C_{w_2}$ implies $C_{w_2} \not\lhd C_{w_1}$. Thus it makes intuitive sense to speak of "before" and "after" with reference to clusters as well as worlds in a transitive frame. The following definitions formalise these notions and are from Segerberg [Seg71, pages 72-73].

A world w **precedes**, or **occurs before**, another world w' if wRw' and $w' \not R w$. A world w **succeeds**, or **occurs after**, another world w' if $w'Rw$ and $w \not R w'$.

A cluster C **precedes**, or **occurs before**, another cluster C' if $C \lhd C'$. A cluster C **succeeds**, or **occurs after**, another cluster C' if $C' \lhd C$.

An obvious notion of immediate successor and immediate predecessor also holds when the clusters can be mapped onto the natural numbers.

A cluster C is an **initial** cluster if no cluster occurs before it; that is, if $\neg(\exists C' : C' \lhd C)$.

A cluster C is a **final** cluster if no cluster occurs after it; that is, if $\neg(\exists C' : C \lhd C')$.

A cluster C is a **first** cluster if it precedes all other clusters; that is, if $\forall C' : C \lhd C'$.

A cluster C is a **last** cluster if it succeeds all other clusters; that is, if $\forall C' : C' \lhd C$.

A first or last cluster must be unique but there may be more than one initial cluster and more than one final cluster. For example, a branching (transitive) finite tree of clusters has a final cluster on each branch but has a unique first cluster at the root.

A cluster is **simple** iff it consists of just one reflexive world. A cluster is **proper** iff it consists of at least two worlds. If C is a proper cluster then $\forall w, w' \in C$ we have wRw' and $w'Rw$, and R is said to be **universal** over C. Thus, in a proper cluster, R must be reflexive, transitivity and *symmetric*. A cluster is **degenerate** iff it consists of just one irreflexive world. Hence a **nondegenerate** cluster must be either simple or proper. Thus, if all clusters in a frame are degenerate, then R is irreflexive (and transitive). And if all clusters in a frame are nondegenerate, then R is reflexive (and transitive).

A (transitive) frame is called a **sharp tack** if it is either a lone cluster C_1, or a degenerate cluster C_1 and a nondegenerate cluster C_2 with $C_1 \lhd C_2$. A (transitive) frame is called a **blunt tack** if it is either a lone nondegenerate cluster C_1, or a degenerate cluster C_1 and a nondegenerate cluster C_2 with $C_1 \lhd C_2$. The intuition is that C_1 is the sharp (degenerate) or blunt (simple) point and C_2 is the (simple or proper) head of an upside down tack.[1]

A frame is **connected** if it satisfies $\forall s, t(sRt \vee s = t \vee tRs)$ but as we have seen, there is no single axiom that characterises connected frames (although we know that L characterises weakly-connected frames) [Gol87, page 29]. A frame is a **weak linear order** or a **sequence of nondegenerate clusters** if it is reflexive, transitive and connected. A frame is a **strict linear order** or a **sequence of degenerate clusters** if it is irreflexive, transitive and connected (and hence asymmetric). A frame is a **linear order** or a **sequence of simple clusters** if it is reflexive, transitive, connected and antisymmetric.

2.4.3 L-frames, L-models and L-satisfiability

Hughes and Cresswell refer to a model $\mathcal{M}$ as a "model for **L**" iff every theorem of **L** is valid on $\mathcal{M}$ [HC84, page 49], and refer to a frame $\mathcal{F}$ as a "frame for **L**" iff every theorem of **L** is valid on every model based on $\mathcal{F}$ [HC84, page 54]. They point out that care is needed when using this terminology because a "model for **T**" does not have to be reflexive [HC84, page 90]. Hence one *cannot* say "suppose $\mathcal{M} = \langle W, R, V\rangle$ is a model for **T**, then R is reflexive". But by Theorem 2 (page 27) we *can* say that every "frame for **T**" is reflexive since any "frame for **T**" must validate T.

Unfortunately we require the term "model for X" in a different sense to be explained shortly. Also, we are not interested in proving characterisation results per se since these are well known for the logics we study. We therefore follow Fitting [Fit83] and *define* certain types of frames as **L**-frames; one for each logic **L**. We then base **L**-models on **L**-frames. For example, we *define* all **T**-frames to be reflexive. We can do so because of known characterisation results that relate each axiomatically formulated logic **L** to the particular frames we choose as **L**-frames.

A frame $\langle W, R\rangle$ is an **L-frame** if it satisfies the conditions in Figure 2.4 according to the value of **L**.

A model $\mathcal{M} = \langle W, R, V\rangle$ is an **L-model** if $\langle W, R\rangle$ is an **L**-frame. Also, $\mathcal{M}$ is an **L-model for** (a finite set of formulae) X if there exists $w \in W$ such that $w \models X$. Recall that $w \models X$ means that $w \models A$ for all $A \in X$.

A formula A is **L**-valid iff A is valid in all **L**-models, and hence in all **L**-frames.

A finite set X is **L-satisfiable** iff there exists an **L**-model for X. So, X is **L-unsatisfiable** iff there are no **L**-models for X.

[1] I now realise that this terminology is bad as the only difference between the two tacks is in the form of the lone cluster C_1.

L	L-frame
K	finite, irreflexive, intransitive tree of worlds
T	finite, reflexive, intransitive tree of worlds
D	finite, intransitive tree with irreflexive non-final worlds and reflexive final worlds
K4	finite, (transitive) tree of clusters
D4	finite, (transitive) tree of clusters with nondegenerate final clusters
S4	finite, (transitive) tree of nondegenerate clusters
B	finite, reflexive, symmetric tree of worlds
K45	finite, (transitive) sharp tack
K45D	finite, (transitive,) blunt tack
S5	finite, reflexive, transitive, symmetric maximally connected graph of worlds
G **K4G**	finite, (irreflexive, transitive) tree of degenerate clusters
Grz **S4Grz** **S4MDum**	finite, (reflexive, transitive) tree of simple clusters
S4.3	finite, (reflexive, transitive) sequence of nondegenerate clusters
S4.3.1	finite, (reflexive, transitive) sequence of nondegenerate clusters with no proper non-final clusters
S4.14	finite, (reflexive, transitive) tree of nondegenerate clusters with no proper non-final clusters
Go	finite, (transitive) tree with degenerate non-final clusters and simple final clusters
S4M	finite, (reflexive, transitive) tree of nondegenerate clusters with simple final clusters
S4Dum	?
K4L	finite, (transitive) sequence of clusters
K4DL	finite, (transitive) sequence of clusters with a nondegenerate last cluster
K4DLZ	finite, (transitive) sequence of degenerate clusters with a final simple cluster
K4GL **KGL**	finite, (irreflexive, transitive) sequence of degenerate clusters
KB	?
KB4	a finite (transitive) cluster
S4R **S4.4** **S4.3Zem**	if $w_1 \neq w_2$ and $w_1 R w_2$ then $w_1 R w_3$ implies $w_2 R w_3$ [Seg71, page 160] (that is, a blunt tack except that if $C_1 \lhd C_2$ then C_1 must be simple)
S4Zem	finite, reflexive (transitive) frames of rank ≤ 2 such that initial cluster is simple [Seg71, page 153]
S4.3Grz **KLGrz** **KGrz.3** **S4.3Go**	finite, (reflexive, transitive) sequence of simple clusters

Figure 2.4: Definition of **L**-frames. The break in the table is the boundary between completed and uncompleted work as far as tableau systems are concerned.

Theorem 5 *In Figure 2.4, each logic* **L** *is characterised by the corresponding* **L***-frames. (Note that the word finite is used in an absolute sense so that every cluster is itself finite.)*

Proof for K, T, D: [Rau83, Rau79];

Proof for K4, D4, S4: Segerberg [Seg71, page 77] proves that these logics are respectively characterised by finite transitive frames; finite, transitive frames where no final cluster is degenerate; and finite transitive frames where no cluster is degenerate. But this is not in terms of trees. Rautenberg [Rau83, Rau79] provides the characterisation results in terms of trees.

Proof for B: [Rau83, Rau79];

Proof for K45, K45D, S5 : Segerberg [Seg71, page 77-78].

Proof for G: Proved indirectly as follows. Segerberg [Seg71, page 88] proves that **K4G**, which he calls **K4W**, is characterised by the class of finite, irreflexive, transitive trees; our **G**-frames. Goldblatt [Gol87, pages 46 and 56] states that **G** = **K4G**, although he calls them **KW** and **K4W**, citing Boolos [Boo79, pages 30 and 82], where it is shown that 4 is a theorem of **G**.

Proof for Grz: The result is obtained indirectly as follows. Segerberg [Seg71, page 103] proves that **S4Grz** is characterised by the class of finite, reflexive, transitive trees of simple clusters; the class we call **Grz**-frames. Hughes and Cresswell [HC84, page 111] state that **S4Grz** = **Grz** since van Benthem and Blok [vB78] prove that 4 and T are theorems of **Grz**. Segerberg [Seg71, page 107] also proves that **S4MDum** = **S4Grz**; a result mentioned in Bull and Segerberg [BS84, page 50].

Proof for S4.3: Segerberg [Seg71, page 77] and Hughes and Cresswell [HC84, page 149].

Proof for S4.3.1: Segerberg [Seg71, page 106], Goldblatt [Gol87, page 59].

Proof for S4.14: Zeman [Zem73, page 249] gives an intuitive argument but (apparently) does not realise that finiteness is essential.

Proof for Go: Rautenberg [Rau83] mentions a characterisation result in terms of no infinite ascending chains of pairwise distinct points but we conjecture that **Go** is characterised by the class of finite transitive trees with degenerate (irreflexive) non-final clusters and simple (reflexive) final clusters. This class differs from **G**-frames since **G**-frames must be irreflexive. It also differs from **S4.14**-frames since **S4.14**-frames must be reflexive.

Proof for S4M: Bull and Segerberg [BS84, page 49].

Proof for S4Dum: Segerberg and Bull [BS84, page 51] claim that the logic **S4Dum** is characterised by our **S4.14**-frames, but as we shall see later, this is incorrect. Segerberg [Seg71, page 106] gives a characterisation result in terms of reflexive kites and all finite reflexive trees but I don't understand his definition of kite on (his) page 89. Further work is to characterise **S4Dum** in terms of finite trees.

Proof for K4L: Note that **K4L** is often called **K4.3** even though T is missing [Seg71, page 50]. Segerberg [Seg71, page 77] proves that **K4L** is characterised by the class of finite frames of clusters where of every two distinct clusters, one precedes the other. That is, frames that are finite, (transitive) sequence of clusters. Further work is to obtain the obvious tableau system. Goldblatt [Gol87, page 26] claims that **K4L** is characterised by transitive weakly-connected frames where finiteness is not stipulated.

Proof for K4DL: Note that **K4DL** is often called **D4.3** even though T is missing [Seg71, page 50]. Segerberg [Seg71, page 77] proves that **K4DL** is characterised by the class of finite transitive frames where of every two distinct clusters, one precedes the other, and the last cluster of the frame is nondegenerate; the frames we call **K4DL**-frames. Further work is to obtain the obvious tableau system.

Proof for K4DLZ: Goldblatt [Gol87, page 55].

Proof for K4GL: Note that **K4GL** is called **K4.3W** in the literature even though T is missing [Seg71, page 89]. Segerberg [Seg71, page 89] proves that **K4GL** is characterised by the class of strict linear orderings; our **K4GL**-frames. Hughes and Cresswell [HC84, page 107] point out that **K4GL** = **KGL** using the result of Boolos that 4 is a theorem of **KGL** [Gol87, page 26], [Boo79, page 30].

Proof for KB: I have been unable to find a characterisation result for this logic in terms of finite frames.

Proof for KB4: Segerberg [Seg71, page 161] proves that **KB4** is characterised by the finite transitive frames containing only one cluster. Further work is to obtain the not so obvious tableau system. But it is known that **KB4** = **KB5** [Gol87, page 26].

Proof for S4.4: Zeman [Zem73, page 256] gives an intuitive argument for this characterisation result and Segerberg [Seg71, page 160] gives a proof although he calls this logic **S4R**. Segerberg [Seg71, page 161] also proves that **S4.4** = **S4R** = **S4.3Zem**. Using **S4.3**-frames as a starting point, it is not hard to see that Zem forces there to be at most two nondegenerate clusters and also forces the first cluster to be simple when there are exactly two nondegenerate clusters.

Proof for S4Zem: Segerberg [Seg71, page 153] where it is noted that **S4Zem** is also known as **S4.04**.

Proof for KGrz.3: Again proved indirectly. Segerberg [Seg71, page 103] proves that **S4.3Grz** is characterised by the class of all finite linear orderings; our **KGrz.3**-frames. Then the arguments mentioned in the proof for **Grz** apply. Segerberg [Seg71, page 107] also proves that Grz and Go are equivalent when added to **S4** giving **S4.3Go** = **S4.3Grz**. Thus we also have **KLGrz** = **K4Grz.3** = **KGrz.3** ●

2.5 Known Miscellaneous Results of Interest

Some other interesting results which we refer to in later sections are:

S4.3 is characterised by the single frame $\langle\, I,\ \leq\, \rangle$ where I is $\mathcal{R}$ or $\mathcal{Q}$, the set of real and rational numbers respectively [Gol87, page 57];

S4.3.1 is characterised by the single frame $\langle\, \omega,\ \leq\, \rangle$ [Gol87, page 57];

K4DLZ is characterised by the single frame $\langle\, \omega,\ <\, \rangle$ [Gol87, page 54];

K4DLX is characterised by the single frame $\langle\, I,\ <\, \rangle$ where X is the axiom of density $\Box\Box A \Rightarrow \Box A$ [Gol87, page 57];

K4GL is characterised by the single frame $\langle\, \omega,\ >\, \rangle$ [Seg71, page 89].

Another bewildering aspect of the literature is that the same name is often given to slightly different axioms. The axiom *Dum*, named after Michael Dummett, appears as:

Source	Axiom	Name
[Gol87]	$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow \Box P)$	Dum
[Seg71]	$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow P)$	Dum
[HC84]	$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow P)$	N1
[HC68]	$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow \Box P)$	M1
[Zem73]	$\Box\{\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow P)\}$	M13
[Zem73]	$\Box\{\Box(\Box(P \Rightarrow \Box P) \Rightarrow \Box P) \Rightarrow (\Diamond\Box P \Rightarrow \Box P)\}$	M14
[BS84]	$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow \Box P)$	Dum

Segerberg [Seg71, page 107] shows that

$$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow P)$$

$$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow \Box P)$$

$$\Box(\Box(P \Rightarrow \Box P) \Rightarrow \Box P) \Rightarrow (\Diamond\Box P \Rightarrow P)$$

$$\Box(\Box(P \Rightarrow \Box P) \Rightarrow \Box P) \Rightarrow (\Diamond\Box P \Rightarrow \Box P)$$

are all equivalent if K, T and 4 are also present, settling the issue.

A syntactic proof that $S4 + M13$ equals $S4 + M14$ is given in Zeman [Zem73, pages 246-248] and is attributed to Kit Fine.

Segerberg [Seg71, page 89] also considers the logics **K4Z**, **D4Z** and **K4LZ**, and shows that the axiom $\Box\Diamond\top \Rightarrow \Box\bot$ puts an end to time since it is determined by the class of all strict linear orderings that have a last element.

Chapter 3

Sequent Systems for Modal Logics

As was mentioned in the introduction, the major disadvantage of axiomatic systems is that it is difficult to find proofs using them since there is no systematic method for doing so. In 1935, Gerhard Gentzen [Gen35, Sza69] introduced a system of deduction for classical first order logic using many rules of inference and few axioms, instead of many axioms and few rules of inference as in axiomatic systems. Gentzen calculi, or sequent systems, as they are called, are significant in automated deduction because they can be used in a goal directed or "backward" manner. That is, an attempt to prove a particular formula can start with the formula itself and proceed by a sequence of operations that reduce the formula into other formulae. Furthermore, this backward process can be applied systematically so that if a proof exists, it will eventually be found. This is in marked contrast to axiomatic systems where a proof often starts with some apparently arbitrary instance of an axiom whose importance only becomes clear at some later point. Although sequent systems are purely *syntactic* proof procedures, for many logics, if a systematic attempt to find a proof for A fails, then we have enough information to construct a finite (semantic) counter-model for A. So for many logics, sequent systems give decision procedures for testing theoremhood.

There is one sequent rule, called the cut rule, that destroys the nice properties mentioned above. The crux of Gentzen's paper is the Hauptsatz, or cut-elimination theorem, which states that the cut rule is redundant for classical first order logic. That is, if there is a sequent proof containing uses of the cut rule then there is a sequent proof devoid of uses of the cut rule.

Over the years, sequent systems have been found for many nonclassical logics since sequent systems closely mimic the semantics of the logic in question. But for many sequent systems the cut rule is indispensable because some theorems of the associated logic are not provable in a system devoid of the cut rule.

In this chapter, we introduce sequent systems for propositional modal logics.

3.1 Syntax of Sequents

In what follows, capital Latin letters like A and B stand for formulae and capital Greek letters like Γ and Δ stand for finite (possibly empty) sets of formulae. Formally, a **sequent** is an ordered pair $\langle \Gamma, \Delta \rangle$ and an intuitive semantic reading of it for classical propositional logic is "if *all* the formulae in Γ are true then *some* formula in Δ is true". Sequents are usually written as $\Gamma \longrightarrow \Delta$ using a new meta-logical symbol, $\longrightarrow$, to highlight this intuition. Note that the sequent arrow, $\longrightarrow$, is not material implication, $\Rightarrow$, although it is closely related. Consequently, $\Gamma \longrightarrow \Delta$ is *not* a formula of any of our logics.

The set of formula on the left hand side of the sequent arrow constitute the **antecedent** (of the sequent) whilst the set of formula on the right hand side constitute the **succedent** (of the sequent). To draw attention to particular formulae A and B in the antecedent or succedent, we write $\Gamma, A \longrightarrow \Delta, B$ for the sequent $\langle \Gamma \cup \{A\}, \Delta \cup \{B\} \rangle$ where $A \notin \Gamma$ and $B \notin \Delta$. As usual, we write $\longrightarrow \Delta$ instead of $\emptyset \longrightarrow \Delta$ and $\Gamma \longrightarrow$ instead of $\Gamma \longrightarrow \emptyset$ where $\emptyset$ is the empty set. Intuitively, a comma in the antecedent acts like $\land$, and a comma in the succedent acts like $\lor$. In particular, $\longrightarrow A$ intuitively states that "A is true" and $A \longrightarrow$ states that "A is false".

3.2 Sequent Rules, Derivations and Proofs

As we said, sequent systems have many inference rules and few axioms. A sequent rule is just like an axiomatic inference rule except that the basic components of the rule are sequents instead of formulae and that sequent rules are written in a vertical fashion with the "if" part above a horizontal line and the "then" part below the line. A **sequent rule** has one sequent below the horizontal line and a (possibly empty) list of sequents above the horizontal line. The sequent below the line is called the **conclusion** (of the sequent rule) and the sequents above the line are called the **premisses** (of the sequent rule). The particular formulae shown explicitly in the conclusion are called the **principal formulae** (of the sequent rule). The particular formulae shown explicitly in the premisses are called the **side formulae** (of the sequent rule). All other formulae are called **parametric formulae.**

Each sequent rule has a unique name which is either a simple alpha-numeric string or is constructed from the connective of its single principal formula and a small arrow. The arrow precedes the connective if the principal formula appears in the antecedent of the conclusion and follows the connective if the principal formula appears in the succedent of the conclusion.

For example,

$$\frac{\Gamma, A \longrightarrow \Delta \quad \Gamma, B \longrightarrow \Delta}{\Gamma, A \vee B \longrightarrow \Delta} \ (\vee \rightarrow)$$

is a sequent rule. The sequent $\Gamma, A \vee B \longrightarrow \Delta$ is the conclusion and the sequents $\Gamma, A \longrightarrow \Delta$ and $\Gamma, B \longrightarrow \Delta$ are the premisses. The formula $A \vee B$ in the conclusion is the principal formula whilst the formulae A and B in the premisses are the side formulae. The formulae in Γ and Δ are the parametric formulae and the name of this rule is $(\vee \rightarrow)$.

A **sequent calculus** or **sequent system** $\mathcal{S}$ is a collection of sequent (inference) rules. We use these terms interchangeably and identify $\mathcal{S}$ with the set of its rule names. A sequent system $\mathcal{SPC} = \{(Axiom), (\Rightarrow\rightarrow), (\rightarrow\Rightarrow), (\wedge \rightarrow), (\rightarrow \wedge), (\vee \rightarrow), (\rightarrow \vee), (\neg \rightarrow), (\rightarrow \neg)\}$ for propositional logic **PC** is shown in Figure 3.1 (page 40).

In axiomatic systems, the notion of proof is explicitly tied to the *modus ponens* rule since it is the main rule of inference. In sequent systems, the only axiomatic rule is (some variation of) the sequent $\Gamma, A \longrightarrow A, \Delta$ since it has no premisses. It is provable by definition since it intuitively states that "if all formulae in $\Gamma \cup \{A\}$ are true then some formula in $\Delta \cup \{A\}$ is true"; namely A. All other sequent rules can be read downwards as "if all the premisses are provable then so is the conclusion". Thus sequent rules are really meta-level statements about provability.

This downward reading is synonymous with "the conclusion is provable if all the premisses are provable" giving an upward (usually referred to as backward) reading of a sequent rule as a recipe for reducing a conclusion to a list of premisses. If we start with some sequent $\Gamma \longrightarrow \Delta$ and repeat this process for each premiss, a branching (right way up) tree structure results. If all the leaves of this structure are instances of the basic sequent then the tree represents an axiomatic proof of $\Gamma \longrightarrow \Delta$ when read downwards from the leaves to the root. This is formalised as follows.

A **branch** is a *finite* list of sequents, one above the other, separated by horizontal lines. A **derivation** is a *finite* (right way up) tree of branches where each sequent except the root is obtained from the one below it by an application of one of the sequent rules. That is, in every branch of a derivation, the sequent below the line is an instance of the conclusion of a sequent rule and the sequent above the line is an appropriate instance of a premiss of the same sequent rule. When constructing a branch of a derivation (tree), a sequent is a **leaf** (with no successors) if it is a basic sequent, or if no sequent rule is applicable to it, or if it appears in the branch previously. A branch is **closed** if its leaf is an instance of the basic sequent. A derivation is a **proof** if all its branches are closed.

The root sequent of a derivation is called the **endsequent** and a proof with endsequent $\Gamma \longrightarrow \Delta$ is a **proof of** $\Gamma \longrightarrow \Delta$. A sequent is $\mathcal{S}$**-provable** if there is a proof of it using only rules from $\mathcal{S}$. A formula A is an $\mathcal{S}$**-theorem** if the sequent $\longrightarrow A$ is $\mathcal{S}$-provable.

Under these definitions, we need to keep track of previous sequents of a branch to detect cycles. Sequent systems are often formulated without the finiteness restriction on branches and derivations so that a cycle leads to an infinite branch rather than a finite branch [Fit83, Wal87]. We prefer our definitions as they allow us to dispense with König's

$$\Gamma, A \longrightarrow A, \Delta \quad (Axiom)$$

$$\frac{\Gamma \longrightarrow A, \Delta \quad \Gamma, B \longrightarrow \Delta}{\Gamma, A \Rightarrow B \longrightarrow \Delta} (\Rightarrow\longrightarrow) \qquad \frac{\Gamma, A \longrightarrow B, \Delta}{\Gamma \longrightarrow A \Rightarrow B, \Delta} (\longrightarrow\Rightarrow)$$

$$\frac{\Gamma, A, B \longrightarrow \Delta}{\Gamma, A \wedge B \longrightarrow \Delta} (\wedge \longrightarrow) \qquad \frac{\Gamma \longrightarrow A, \Delta \quad \Gamma \longrightarrow B, \Delta}{\Gamma \longrightarrow A \wedge B, \Delta} (\longrightarrow \wedge)$$

$$\frac{\Gamma, A \longrightarrow \Delta \quad \Gamma, B \longrightarrow \Delta}{\Gamma, A \vee B \longrightarrow \Delta} (\vee \longrightarrow) \qquad \frac{\Gamma \longrightarrow A, B, \Delta}{\Gamma \longrightarrow A \vee B, \Delta} (\longrightarrow \vee)$$

$$\frac{\Gamma \longrightarrow A, \Delta}{\Gamma, \neg A \longrightarrow \Delta} (\neg \longrightarrow) \qquad \frac{\Gamma, A \longrightarrow \Delta}{\Gamma \longrightarrow \neg A, \Delta} (\longrightarrow \neg)$$

Figure 3.1: Sequent system $\mathcal{SPC}$.

Lemma in our completeness proofs and lead to constructive proofs of completeness (where "constructive" has no intuitionistic connotation).

Example 1 The following is a proof of the endsequent $A \wedge (A \Rightarrow B) \longrightarrow B$ in the sequent system $\mathcal{SPC}$. At each (backward) step the name of the rule that is used to reduce the conclusion to the premisses is annotated to the right of the horizontal line separating them:

$$\frac{\dfrac{A \longrightarrow A, B \quad A, B \longrightarrow B}{A, A \Rightarrow B \longrightarrow B} (\Rightarrow\longrightarrow)}{A \wedge (A \Rightarrow B) \longrightarrow B} (\wedge \longrightarrow)$$

3.3 The Rules Explained

The rules of sequent systems fall into three categories: basic sequent(s), logical rules and structural rules.

The basic sequent is $\Gamma, A \longrightarrow A, \Delta$. As stated previously, it is the only rule that is axiomatic in any sense.

The logical rules are the next four pairs of rules, where each conclusion contains a connective in its principal formula. There is one pair for each connective, depending on whether the principal formula appears in the antecedent or the succedent. Each logical rule shown in Figure 3.1 (page 40) reduces the principal formula into its constituent formulae although this is not always the case for some of the modal logics we shall encounter.

Traditionally, the logical rules are said to "introduce" the connective of their principal formula into their conclusion in accordance with their downward reading from premisses to conclusion. For example, the $(\wedge \rightarrow)$ rule introduces the connective $\wedge$ into its conclusion. Since we prefer to view sequent rules as reductions from their conclusion to their premisses, we avoid this terminology.

There are no structural rules in $\mathcal{SPC}$.

3.4 Subformulae Property, Analycity, Invertibility

The logical rules of $\mathcal{SPC}$ specify how to reduce some particular formula, $A \wedge B$ for example. If a sequent contains $A \wedge B$ and $C \wedge D$ we have a choice of which to reduce first giving a nondeterministic choice in applying this particular rule. If the sequent also contains $C \vee D$ then there is a nondeterministic choice of which rule to apply first. Thus, for a given endsequent, there are many different derivations corresponding to the different choices. What is the best way to proceed ?

A sequent rule has the **subformula property** if the side formulae are subformulae of the principal formula. A sequent system has the **subformula property** if each rule has it. A sequent rule has the **strict subformula property** if the side formulae are strict subformulae of the principal formula. A sequent system has the **strict subformula property** if each rule has it.

$\mathcal{SPC}$ has the strict subformula property (and hence the subformula property). That is, not only are the side formula of each rule built out of subformulae of the principal formula, but these side formulae are *strictly* simpler than the principal formula. Consequently, *any* sequence of rule applications is guaranteed to terminate with no chance of a repeated sequent and we can blindly apply the rules without worrying about cycles. Note that the strict subformula property guarantees termination only because our sequents cannot contain duplicate formulae. If it were permitted for a formula to appear more than once in the antecedent or the succedent, we would lose the termination property. We return to this issue later.

A sequent rule is **invertible** if whenever the conclusion is provable, so are each of the premisses. Contrapositively, if any of the premisses of an invertible rule are not provable, then neither is the conclusion. This is very different from the top down or bottom up reading of sequents as mentioned above because a sequent rule is invertible exactly when we lose no information in reducing its conclusion to its premisses [D'A90]. For an informal example, the $(\wedge \rightarrow)$ rule premiss retains all the information contained in its conclusion since the premiss also intuitively states that both A and B are true. Kleene [Kle52] formally proves that each rule in $\mathcal{SPC}$ is invertible.

So suppose we have some derivation of an endsequent. If all the branches are closed then we have a proof and the endsequent is $\mathcal{SPC}$-provable. But what if some branch is not closed ? Since $\mathcal{SPC}$ derivations contain no cycles, this sequent cannot be a duplicate. That is, it is a leaf because no $\mathcal{SPC}$ rule is applicable to it, and so it is not $\mathcal{SPC}$-provable.

$$\frac{\Gamma, B, A \longrightarrow \Delta}{\Gamma, A, B \longrightarrow \Delta}\ (Interchange \rightarrow) \qquad \frac{\Gamma \longrightarrow B, A, \Delta}{\Gamma \longrightarrow A, B, \Delta}\ (\rightarrow Interchange)$$

$$\frac{\Gamma, A, A \longrightarrow \Delta}{\Gamma, A \longrightarrow \Delta}\ (Contraction \rightarrow) \qquad \frac{\Gamma \longrightarrow A, A, \Delta}{\Gamma \longrightarrow A, \Delta}\ (\rightarrow Contraction)$$

$$\frac{\Gamma \longrightarrow \Delta}{\Gamma, \Phi \longrightarrow \Psi, \Delta}\ (Weakening) \qquad \frac{\Gamma, A \longrightarrow \Delta \quad \Gamma \longrightarrow A, \Delta}{\Gamma \longrightarrow \Delta}\ (Cut)$$

Figure 3.2: Common structural rules.

But each rule of $\mathcal{S}PC$ is invertible so the conclusion instance of this rule application is also not $\mathcal{S}PC$-provable. We can apply this reasoning all the way to the endsequent and conclude that the endsequent is not $\mathcal{S}PC$-provable.

The importance of invertibility is that it allows us to conclude that a sequent $\Gamma \longrightarrow \Delta$ is *not* $\mathcal{S}PC$-provable as soon as we find *any* derivation of $\Gamma \longrightarrow \Delta$ that cannot be extended to a proof. The strict subformula property guarantees that *any* sequence of rule applications eventually terminates. So together they imply that the order of rule applications is immaterial in $\mathcal{S}PC$ and that each formula need only be reduced once. This is a very desirable feature of a sequent system but unfortunately it does not appear to hold for the modal logics we consider.

3.5 Structural Rules

Gentzen's original sequent systems were (and many modern versions still are) formulated in terms of *sequences* or *lists* of formulae rather than sets. Under Gentzen's reading, the sequent $\Gamma, A, B \longrightarrow \Delta$ is different from the sequent $\Gamma, B, A \longrightarrow \Delta$. Similarly, $\Gamma, A, A \longrightarrow \Delta$ is different from $\Gamma, A \longrightarrow \Delta$. Thus certain "structural rules" were required to manipulate the order of formulae in a sequence, to handle duplication of formulae in a sequence and to remove unnecessary formulae altogether. These structural rules are shown in Figure 3.2 (page 42) along with another common structural rule called cut.

Rules of Interchange: The rules of interchange allow us to change the order of the formulae in a sequent, but the order of the formulae has no bearing on the intuitive reading of a sequent. So, even when formulated in terms of lists or sequences, most modern sequent systems omit the rules of interchange and replace them by the proviso that two sequents differing only in their ordering of formulae are equivalent [Zem73]. In our setting, where sequents are built from sets, the rules of interchange are not necessary since the order of the formulae in a set is irrelevant.

Rules of Weakening: The weakening (or thinning) rule allows us to throw away formulae in reducing a conclusion to a premiss. It is necessary in any sequent system using $A \longrightarrow A$ as the basic sequent. If instead, the sequent $\Gamma, A \longrightarrow A, \Delta$ is accepted as a basic sequent then weakening is also eliminable, as in $\mathcal{SPC}$; see [Wal87, Gal87].

The weakening rule does not have the strict subformula property but it does have the subformula property, so it cannot introduce cycles (as long as we make the reasonable assumption that $\Phi \cup \Psi$ is non-empty). However, it is not invertible since we lose the information content of Φ and Ψ, and this is the main reason for excluding it from $\mathcal{SPC}$ and replacing it with our more general basic sequent.

The weakening (or thinning) rule provides us with yet another nondeterministic aspect since it allows us to discard *any* subset Φ of the antecedent and *any* subset Ψ of the succedent from the conclusion — the choice is left to us. We retain weakening because this property proves useful for some of the modal logics we consider. Ways of dispensing with weakening are discussed in Section 4.5.

Example 2 The following is also a proof of $A \wedge (A \Rightarrow B) \longrightarrow B$ (see Example 1 page 40) but the basic sequent is assumed to be $A \longrightarrow A$ instead of $\Gamma, A \longrightarrow A, \Delta$. Notice how this small change in the basic sequent forces us to introduce the weakening rule:

$$\cfrac{\cfrac{\cfrac{A \longrightarrow A}{A \longrightarrow A, B}\,(Weakening) \qquad \cfrac{B \longrightarrow B}{A, B \longrightarrow B}\,(Weakening)}{A, A \Rightarrow B \longrightarrow B}\,(\Rightarrow\longrightarrow)}{A \wedge (A \Rightarrow B) \longrightarrow B}\,(\wedge \longrightarrow)$$

Cut Rule: The cut rule is unique in that a formula A in its premisses does not appear in its conclusion; not even as a subformula. A reduction from the conclusion to the premiss thus involves a formula A that is unknown, and hence, the cut rule does not give us a recipe for reducing the conclusion to the premisses as we have to guess the correct A. The cut rule corresponds to a generalisation of *modus ponens* in axiomatic systems since putting $\Gamma = \emptyset$ and $\Delta = B$ effectively gives an instance of modus ponens when read downwards:

$$\frac{A \longrightarrow B \qquad \longrightarrow A}{\longrightarrow B}$$

It is well known that a sequent system $\mathcal{S}$ without the cut rule can be exponentially worse at proving certain sequents than a sequent system $\mathcal{S}'$ containing the cut rule [D'A90, Boo84]. This is the price we have to pay for avoiding (the nondeterminism of) guesswork.

Contraction: Finally, there is the rule of contraction. The contraction rule allows us to reduce any formula more than once. That is, since $A \wedge A$ is the same as A for the logics we consider, we simply duplicate A whenever we need it by "reducing" the conclusion $\Gamma, A \longrightarrow \Delta$ to the premiss $\Gamma, A, A \longrightarrow \Delta$. The contraction rule makes sense only when we use sequences of formulae rather than sets of formulae, hence this form of the contraction rule does not make sense in our setting.

Contraction is clearly a form of duplication but note that the $(\vee \longrightarrow)$, $(\wedge \longrightarrow)$ and $(\Rightarrow\longrightarrow)$ rules have a form of duplication inherent in them since copies of the parametric formulae Γ and Δ appear in both premisses. We do not consider this sort of duplication to be contraction although both Girard and Wallen do [Gir87, Wal87]. For us, contraction means the reuse of the *principal* formulae on the same branch.

The contraction rule does not have the strict subformula property although it has the subformula property. That is, the side formulae still consist of subformulae of the principal formula, but not strict subformulae. Our formulation of sequents as sets does not permit the contraction rule as it stands in Figure 3.2 because a formula may not appear more than once in our antecedents (or succedents). It is possible to build the effect of the contraction rule into each rule by explicitly adding a copy of the principal formula to each premiss. For example, the $(\wedge \longrightarrow)$ rule would become:

$$\frac{\Gamma, A \wedge B, A, B \longrightarrow \Delta}{\Gamma, A \wedge B \longrightarrow \Delta}\ (\wedge \longrightarrow)$$

As an immediate consequence we admit the possibility of cycles since we can apply this rule repeatedly. There is little point in doing so, however, as we have already seen that all forms of contraction are redundant in $\mathcal{SPC}$.

An alternate method to obtain the effect of contraction is to abuse the fact that our sequents are made up of sets. We mention it only because it has caused some confusion in the literature due to the fact that in this method, contraction is hidden. In this "method", all the rules of Figure 3.1 remain the same and contraction comes for free because $\Gamma, A, A \longrightarrow \Delta$ is the same sequent as $\Gamma, A \longrightarrow \Delta$ since $\Gamma \cup \{A\} \cup \{A\} = \Gamma \cup \{A\}$. But note that this is forbidden by the definitions of proof since a proof of $\Gamma, A \longrightarrow \Delta$ must have endsequent $\Gamma, A \longrightarrow \Delta$. In particular, $\Gamma, A, A \longrightarrow \Delta$ will not suffice. It could be argued that a proof of $\Gamma, A, A \longrightarrow \Delta$ *is* a proof of $\Gamma, A \longrightarrow \Delta$ but strictly this is not so. Note that the sequent systems of Fitting [Fit83] and Rautenberg [Rau83] are incomplete precisely because they omit contraction. The error is corrected in Fitting [Fit88] where a strict tableau system is introduced.

When contraction is necessary, we follow the first method and build contraction into our sequent rules explicitly since it forces us to identify the types of formulae that may require duplication.

Smullyan [Smu68b] discusses systems for classical first order logic where contraction is built into the rules, giving a system with no structural rules at all.

3.6 Modal Sequent Systems

Up till now, we have ignored rules to handle modalities like $\Box$ and $\Diamond$. In general, there are two rules for each modality, one where the principal formula containing that modality appears in the antecedent of the conclusion and one where it appears in the succedent of the conclusion. Since all our modal logics are extensions of classical propositional logic, $\mathcal{SPC}$ should be a subset of all the modal sequent systems we encounter.

$$A \longrightarrow A \qquad (Ax)$$

$$\frac{\Gamma \longrightarrow A, \Delta \quad \Gamma, B \longrightarrow \Delta}{\Gamma, A \Rightarrow B \longrightarrow \Delta} (\Rightarrow\longrightarrow) \qquad \frac{\Gamma, A \longrightarrow B, \Delta}{\Gamma \longrightarrow A \Rightarrow B, \Delta} (\longrightarrow\Rightarrow)$$

$$\frac{\Gamma, A, B \longrightarrow \Delta}{\Gamma, A \wedge B \longrightarrow \Delta} (\wedge \longrightarrow) \qquad \frac{\Gamma \longrightarrow A, \Delta \quad \Gamma \longrightarrow B, \Delta}{\Gamma \longrightarrow A \wedge B, \Delta} (\longrightarrow \wedge)$$

$$\frac{\Gamma, A \longrightarrow \Delta \quad \Gamma, B \longrightarrow \Delta}{\Gamma, A \vee B \longrightarrow \Delta} (\vee \longrightarrow) \qquad \frac{\Gamma \longrightarrow A, B, \Delta}{\Gamma \longrightarrow A \vee B, \Delta} (\longrightarrow \vee)$$

$$\frac{\Gamma \longrightarrow A, \Delta}{\Gamma, \neg A \longrightarrow \Delta} (\neg \longrightarrow) \qquad \frac{\Gamma, A \longrightarrow \Delta}{\Gamma \longrightarrow \neg A, \Delta} (\longrightarrow \neg)$$

$$\frac{\Box\Gamma, A \longrightarrow \Diamond\Delta}{\Box\Gamma, \Diamond A \longrightarrow \Diamond\Delta} (\Diamond \longrightarrow S4) \qquad \frac{\Gamma \longrightarrow A, \Diamond A, \Delta}{\Gamma \longrightarrow \Diamond A, \Delta} (\longrightarrow \Diamond S4)$$

$$\frac{\Gamma, A, \Box A \longrightarrow \Delta}{\Gamma, \Box A \longrightarrow \Delta} (\Box \longrightarrow S4) \qquad \frac{\Box\Gamma \longrightarrow A, \Diamond\Delta}{\Box\Gamma \longrightarrow \Box A, \Diamond\Delta} (\longrightarrow \Box S4)$$

$$\frac{\Gamma \longrightarrow \Delta}{\Gamma, \Phi \longrightarrow \Psi, \Delta} (Weakening)$$

Figure 3.3: Sequent system $\mathcal{S}S4$.

For modal logics the interchange rules are redundant for the same reasons as for $\mathcal{S}PC$. But for most modal logics we will encounter, the contraction rule is not always eliminable since some formulae have to be used more than once on the same branch to obtain a proof. For one of the modal logics we shall consider, we find that weakening is also useful although it is eliminable at a cost. To explain these complications we introduce a very common cut-free sequent system for **S4**.

Figure 3.3 shows a very common sequent system $\mathcal{S}S4$ for modal logic **S4**. At the moment we do not have the machinery to prove this claim but a proof is given in Section 4.3. In Figure 3.3, $\Box\Gamma$ denotes the set $\{\Box A \mid A \in \Gamma\}$ and similarly, $\Diamond\Delta$ denotes the set $\{\Diamond A \mid A \in \Delta\}$. Note that the basic sequent is now $A \longrightarrow A$ instead of $\Gamma, A \longrightarrow A, \Delta$. But weakening is also present to make this change feasible (and in doing so, deprive $\mathcal{S}S4$ of invertibility). Thus $\mathcal{S}PC$ is included in $\mathcal{S}S4$ in an indirect way.

The $(\Diamond \rightarrow S4)$ rule conclusion has principal formula $\Diamond A$. But the parametric formulae $\Box\Gamma$ in the antecedent must have $\Box$ as their outermost connective and the parametric formulae $\Diamond\Delta$ in the succecedent must have $\Diamond$ as their outermost connective. If an arbitrary sequent $\Gamma_1, \Diamond A \longrightarrow \Delta_1$ does not meet these restrictions, $(\Diamond \rightarrow S4)$ is not directly applicable to it. We must first partition Γ_1 into $\Box\Gamma \cup \Phi$ and partition Δ_1 into $\Diamond\Delta \cup \Psi$ giving a conclusion $\Box\Gamma, \Phi, \Diamond A \longrightarrow \Psi, \Diamond\Delta$. Then weakening allows us to discard Φ and Ψ giving a premiss $\Box\Gamma, \Diamond A \longrightarrow \Diamond\Delta$ where in the extreme case, both $\Box\Gamma$ and $\Diamond\Delta$ may be empty. Now $(\Diamond \rightarrow S4)$ is applicable. Similar restrictions apply to the $(\rightarrow \Box S4)$ rule and this is the reason for including weakening. It is possible to achieve the same effect by building weakening into the $(\Diamond \rightarrow S4)$ and $(\rightarrow \Box S4)$ rules; see Fitting [Fit83] and Wallen [Wal87].

The $(\Diamond \rightarrow S4)$ and $(\rightarrow \Box S4)$ rules are invertible, but since this invertible formulation requires weakening, the system as a whole loses invertibility. Thus we are already doomed to search *all* derivations of a given endsequent in order to conclude that it is not $\mathcal{S}S4$-provable. Either way, for most modal logics, we find that we *must* throw away information in reducing a conclusion to a premiss in the $(\Diamond \rightarrow S4)$ and $(\rightarrow \Box S4)$ rules, a property that Fitting terms "destructive" [Fit90].

The $(\Diamond \rightarrow S4)$ and $(\rightarrow \Box S4)$ rules have the strict subformula property and they contain no form of contraction. So there is a chance that derivations in $\mathcal{S}S4$ may be cycle free and hence the termination property may hold. Alas, the $(\Box \rightarrow S4)$ and $(\rightarrow \Diamond S4)$ rules obey only the subformula principle since the principal formula $\Box A$ or $\Diamond A$ appears in the respective premisses. Thus $(\Box \rightarrow S4)$ and $(\rightarrow \Diamond S4)$ have a form of contraction built into them and cycles are possible in constructing $\mathcal{S}S4$ derivations. Therefore, we must keep track of previous sequents and this is why we have formulated our sequent systems in terms of finite trees with possibly cyclic branches. The form of the contraction, however, is very limited since we are permitted to duplicate a formula in $(\Box \rightarrow S4)$ only if its outermost connective is $\Box$ and we are permitted to duplicate a formula in $(\rightarrow \Diamond S4)$ only if its outermost connective is $\Diamond$.

A syntactic study of this system is carried out by Zeman [Zem73]. Zeman starts by augmenting Gentzen's original system for **PC**, which includes cut, contraction and weakening, along with the rules:

$$\frac{\Gamma, A \longrightarrow \Delta}{\Gamma, \Box A \longrightarrow \Delta}\ (\Box \rightarrow) \qquad\qquad \frac{\Box\Gamma \longrightarrow A, \Diamond\Delta}{\Box\Gamma \longrightarrow \Box A, \Diamond\Delta}\ (\rightarrow \Box S4)$$

and their duals for $\Diamond$. He then proves syntactic cut-elimination and syntactic contraction-elimination using arguments similar to those of Gentzen [Gen35, Sza69]. But in order to carry through the proof of the latter he is forced to alter the $(\Box \rightarrow)$ rule to $(\Box \rightarrow S4)$ [Zem73, pages 126-128]. Such limited contraction is necessary in $\mathcal{S}S4$ since an antecedent formulae with outermost connective $\Box$, or a succedent formula with outermost connective $\Diamond$, may have to be used more than once *on the same branch,* as illustrated by the following example.

Example 3 The formula $\Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A)$ is a theorem of **S4**, see [Fit83, page 223]. But a sequent proof requires that this formula appear twice in the proof as shown below.

$$
\begin{array}{cl}
\Diamond A \longrightarrow \Diamond A, \Box\Diamond A, \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) \quad (\text{Axiom}) & \\
\hline
\longrightarrow \Diamond A, \Diamond A \Rightarrow \Box\Diamond A, \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) & (\longrightarrow\Rightarrow) \\
\hline
\longrightarrow \Diamond A, \Box(\Diamond A \Rightarrow \Box\Diamond A), \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) & (\longrightarrow \Box S4) \\
\hline
\longrightarrow \Diamond A, \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) & (\longrightarrow \Diamond S4) \\
\hline
\Diamond A \longrightarrow \Box\Diamond A, \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) & (\longrightarrow \Box S4) \\
\hline
\longrightarrow \Diamond A \Rightarrow \Box\Diamond A, \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) & (\longrightarrow\Rightarrow) \\
\hline
\longrightarrow \Box(\Diamond A \Rightarrow \Box\Diamond A), \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) & (\longrightarrow \Box S4) \\
\hline
\longrightarrow \Diamond\Box(\Diamond A \Rightarrow \Box\Diamond A) & (\longrightarrow \Diamond S4)
\end{array}
$$

3.7 Cut-elimination

Note the importance of the subformula property and hence of cut-elimination. Given some (finite) endsequent, each sequent in the derivation is finite, hence there are only a finite number of sequents that can be built out of the subformulae of the endsequent. Hence there are a finite number of derivations of the endsequent using the rules of a given sequent system $\mathcal{S}$. If we search through all of them and find that *all* of them are not proofs then we can declare that the endsequent is not $\mathcal{S}$-provable, and hence, not an $\mathcal{S}$-theorem.

For most modal logics, invertibility is lost due to the $(\Diamond \longrightarrow L)$ rule and the strict subformula property is lost due to the presence of a limited form of contraction. However, $\mathcal{S}S4$ is still cut free. How can we be sure that cut is redundant ?

Cut-elimination can be proved in two ways:

(a) one can give syntactic transformations to eliminate all uses of the cut rule, as was done by Gentzen; or

(b) one can start with a calculus omitting the cut rule and show that the calculus is nevertheless sound and complete with respect to the intended semantics.

Syntactic cut-elimination is usually very difficult, so we follow the much easier semantic cut-elimination route via a method which traces its origins to the work of Beth and Hintikka [Bet53, Bet55, Hin55]. We therefore require a semantics for modal sequent systems.

3.8 Semantics for Sequent Systems

Up till now we have mentioned "intuitions" of sequents and sequent systems informally. In this section, these "intuitions" are formalised so that theorems *about* sequent systems, rather than theorems *using* sequent systems, can be proved. Since our sequent systems are for modal logics, we simply extend the satisfaction relation $\models$ to handle sequents instead of formulae and also extend the corresponding semantic notions of validity, satisfiability and unsatisfiability.

If $\mathcal{M} = \langle W, R, V \rangle$ is an **L**-model and $w \in W$ then

$$w \models \Gamma \longrightarrow \Delta \quad \text{iff} \quad w \not\models A \text{ for some } A \in \Gamma \text{ or } w \models B \text{ for some } B \in \Delta.$$

Note that $w \models \ \longrightarrow A$ iff $w \models A$. Also, $w \models A \longrightarrow \ $ iff $w \not\models A$.

A sequent $\Gamma \longrightarrow \Delta$ is **true** in an **L**-model $\mathcal{M} = \langle W, R, V \rangle$ if for all $w \in W$, $w \models \Gamma \longrightarrow \Delta$.

A sequent $\Gamma \longrightarrow \Delta$ is **L-valid** if it is true in all **L**-models. A sequent $\Gamma \longrightarrow \Delta$ is **L-satisfiable** if there is some **L**-model $\mathcal{M} = \langle W, R, V \rangle$ with some $w \in W$ such that $w \models \Gamma \longrightarrow \Delta$. A sequent $\Gamma \longrightarrow \Delta$ is **L-unsatisfiable** if it is not **L**-satisfiable.

3.9 Soundness and Completeness

The sequent calculus $\mathcal{S}S4$ is a purely syntactic proof procedure, but what is the logic it defines ? Recall that logics are simply subsets of the set of all formulae **Fml**. Thus if we define

$$\mathbf{L}\mathcal{S} = \{A \in \mathbf{Fml} : \ \longrightarrow A \text{ is } \mathcal{S}L\text{-provable}\}$$

then $\mathbf{L}\mathcal{S}$ is the logic defined by the sequent system $\mathcal{S}L$.

In general we use, $\mathcal{S}L$ for the sequent system and $\mathbf{L}\mathcal{S}$ for the logic defined by this system. These definitions are cyclic because at the moment we have no way to show that $\mathcal{S}L$ is actually the sequent system for axiomatic logic **L**. For this we require the notions of soundness and completeness.

The semantics of Section 2.3 (page 22) are still appropriate and so our first task is to show that $\mathcal{S}L$-provability coincides with **L**-validity. This is done in two steps.

Soundness: If A is an $\mathcal{S}$-theorem then A is **L**-valid. That is, if $\longrightarrow A$ is $\mathcal{S}$-provable then A is **L**-valid.

Completeness: If A is **L**-valid then A is an $\mathcal{S}L$-theorem. That is, if A is **L**-valid then $\longrightarrow A$ is $\mathcal{S}L$-provable.

Finite Model Property (fmp): If A is not an $\mathcal{S}L$-theorem then there is a *finite* **L**-model $\mathcal{M}$ which satisfies $\neg A$. That is, $\mathcal{M} = \langle W, R, V \rangle$ is an **L**-model, but for some $w \in W$, $w \not\models A$.

$$A \longrightarrow A \qquad (Ax)$$

$$\frac{\Gamma, A, B \longrightarrow \Delta}{\Gamma, A \wedge B \longrightarrow \Delta} (\wedge \rightarrow) \qquad \frac{\Gamma \longrightarrow A, \Delta \quad \Gamma \longrightarrow B, \Delta}{\Gamma \longrightarrow A \wedge B, \Delta} (\rightarrow \wedge)$$

$$\frac{\Gamma \longrightarrow A, \Delta}{\Gamma, \neg A \longrightarrow \Delta} (\neg \rightarrow) \qquad \frac{\Gamma, A \longrightarrow \Delta}{\Gamma \longrightarrow \neg A, \Delta} (\rightarrow \neg)$$

$$\frac{\Gamma, A, \Box A \longrightarrow \Delta}{\Gamma, \Box A \longrightarrow \Delta} (\Box \rightarrow S4) \qquad \frac{\Box\Gamma \longrightarrow A}{\Box\Gamma \longrightarrow \Box A} (\rightarrow \Box S4)$$

$$\frac{\Gamma \longrightarrow \Delta}{\Gamma, \Phi \longrightarrow \Psi, \Delta} (Weakening)$$

Figure 3.4: Sequent system $\mathcal{S}S4$ in primitive notation.

3.10 Primitive Notation

Since the only primitive connectives are $\neg$, $\wedge$ and $\Box$, I should really show that the rules for $\Rightarrow$, $\vee$ and $\Diamond$ are obtainable from the rules for these primitive connectives. For example, the ($\Rightarrow\rightarrow$) rule can be derived by rewriting $A \Rightarrow B$ into its equivalent primitive notation form $\neg(A \wedge \neg B)$ viz:

$$\cfrac{\cfrac{\cfrac{\Gamma \longrightarrow A, \Delta \qquad \cfrac{\Gamma, B \longrightarrow \Delta}{\Gamma \longrightarrow \neg B, \Delta} (\rightarrow \neg)}{\Gamma \longrightarrow A \wedge \neg B, \Delta} (\rightarrow \wedge)}{\Gamma, \neg(A \wedge \neg B) \longrightarrow \Delta} (\neg \rightarrow)}{\Gamma, A \Rightarrow B \longrightarrow \Delta} \text{(rewriting)}$$

Figure 3.4 shows the rules of $\mathcal{S}S4$ in primitive notation. All occurrences of $\Diamond A$ are assumed to have been translated into $\neg\Box\neg A$ and moved to the appropriate side to remove the outermost $\neg$ sign. All the rules of the old $\mathcal{S}S4$ system can be derived in a manner similar to the derivation of the ($\Rightarrow\rightarrow$) rule shown above.

3.11 From Sequent Systems To Tableau Systems

Sequent systems are essentially refutation procedures. That is, a derivation for $\Gamma \longrightarrow \Delta$ corresponds to a search for a model that satisfies all formulae in Γ and falsifies all formulae in Δ. If no such model can be found then $\Gamma \longrightarrow \Delta$ is valid.

$$(0)\ \frac{P; \neg P}{0}$$

$$(\wedge)\ \frac{X; P \wedge Q}{X; P; Q} \qquad (\vee)\ \frac{X; \neg(P \wedge Q)}{X; \neg P \mid X; \neg Q}$$

$$(\neg)\ \frac{X; \neg\neg P}{X; P}$$

$$(\theta)\ \frac{X; Y}{X}$$

Figure 3.5: Tableau rules for $\mathcal{CPC}$

The **associated set** of a sequent $\Gamma \longrightarrow \Delta$ is the set $\Gamma \cup \neg\Delta = \Gamma \cup \{\neg B : B \in \Delta\}$. Note that different sequents may have the same associated set; for example, the sequents $\neg A \longrightarrow B$ and $\longrightarrow A, B$ both have associated set $\{\neg A, \neg B\}$. The importance of this transformation from sequents to sets is that

Proposition 1 *A sequent* $\Gamma \longrightarrow \Delta$ *is* **L**-*valid iff the associated set* $\Gamma \cup \neg\Delta$ *is* **L**-*unsatisfiable.*

Proof ($\Rightarrow$): Suppose $\Gamma \longrightarrow \Delta$ is **L**-valid and $\mathcal{M} = \langle W, R, V \rangle$ is an arbitrary **L**-model. Then for every $w \in W$ we have $w \models \Gamma \longrightarrow \Delta$. That is, $w \not\models A$ for some $A \in \Gamma$ or $w \models B$ for some $B \in \Delta$. In either case, $w \not\models \Gamma \cup \neg\Delta$. Since w was an arbitrary world in $\mathcal{M}$ we know that $\mathcal{M}$ is not an **L**-model for $\Gamma \cup \neg\Delta$. But $\mathcal{M}$ was an arbitrary **L**-model, hence there are no **L**-models for $\Gamma \cup \neg\Delta$. That is, $\Gamma \cup \neg\Delta$ is **L**-unsatisfiable.

Proof ($\Leftarrow$): Suppose $\Gamma \cup \neg\Delta$ is **L**-unsatisfiable and $\mathcal{M}' = \langle W', R', V' \rangle$ is an arbitrary **L**-model. Then for any $w' \in W'$ we have $w' \not\models \Gamma \cup \neg\Delta$. That is, $w' \not\models A$ for some $A \in \Gamma$ or $w' \not\models \neg B$ for some $\neg B \in \neg\Delta$. In other words, $w' \not\models A$ for some $A \in \Gamma$ or $w' \models B$ for some $B \in \Delta$. Hence $w' \models \Gamma \longrightarrow \Delta$. But w' was an arbitrary world in an arbitrary **L**-model, hence $\Gamma \longrightarrow \Delta$ is **L**-valid. •

This means that each of the sequent rules can be viewed in terms of sets rather than sequents. Since our completeness proof involves the semantic notion of models made up of sets of formulae, it is much easier to work with sets rather than with sequents *per se*.

The rules of Figure 3.5 are the nonmodal sequent rules of Figure 3.4 written in a different form, using sets. The rules of Figure 3.5 are upside down counterparts of the sequent rules so that the associated set of the conclusion of each sequent rule appears above the line as the numerator of the corresponding set rule. The associated sets of the premisses of each sequent rule appear below the line as the denominators of the corresponding set rule and furthermore, are separated by vertical bars. There is no analogue of the $(\neg \rightarrow)$ rule because the premiss and conclusion are associated with the same set $\Gamma \cup \neg\Delta \cup \{\neg A\}$.

Such rules are usually called tableau rules.

Chapter 4

Tableau Systems for Modal Logics

In the last chapter we showed that sequents could be viewed in terms of their associated sets. Tableau systems which originate from the work of Beth [Bet53, Bet55] and Hintikka [Hin55] are the analogues of sequent systems when formula sets are used instead of sequents. Smullyan [Smu68a] uses a different formulation and Fitting [Fit83] uses Smullyan-tableaux rather than Beth-tableaux. We use Beth-tableaux because the direct correspondence between sequent systems and tableau systems is easier to see using Beth's formulation. Smullyan-tableaux are discussed later.

Since our tableau systems work with *finite* sets of formulae, we use the following notational conventions:

> p, q denote primitive (atomic) propositions from $\mathcal{P}$;
>
> P, Q, Q_i and P_i denote (well formed) formulae;
>
> X, Y, Z denote finite sets of (well formed) formulae;
>
> $X; Y$ stands for $X \cup Y$ and $X; P$ stands for $X \cup \{P\}$;
>
> $\Box X$ stands for $\{\Box P \mid P \in X\}$;
>
> $\neg\Box X$ stands for $\{\neg\Box P \mid P \in X\}$.

The following measures on formulae and formulae sets are also useful for later arguments. For a formula P, the degree $deg(P)$ is the number of nested connectives in P according to:

> $deg(p) = 0$ when $p \in \mathcal{P}$ is atomic;
>
> $deg(\neg P) = 1 + deg(P)$;
>
> $deg(P \land Q) = 1 + \max(deg(P), deg(Q))$;
>
> $deg(\Box P) = 1 + deg(P)$.

For a formula P, the modal degree $mdeg(P)$ is the number of nested $\Box$ connectives in P according to:

$mdeg(p) = 0$ when $p \in \mathcal{P}$ is atomic;

$mdeg(\neg P) = mdeg(P)$;

$mdeg(P \wedge Q) = \max(mdeg(P), mdeg(Q))$;

$mdeg(\Box P) = 1 + mdeg(P)$.

For a finite set X,

$$deg(X) = \sum_{P \in X} deg(P);$$

$$mdeg(X) = \sum_{P \in X} mdeg(P); \text{ and}$$

$$mdegmax(X) = max\{mdeg(P) \mid P \in X\}.$$

As usual, the cardinality $card(X)$ of X is the number of elements in X.

To minimise the number of rules, we work with primitive notation. Each of our tableau rules has a dual rule which can be easily obtained by using the definition of $\Diamond$ as $\neg\Box\neg$.

4.1 Syntax of Tableau Systems

Tableau systems consist of a collection of tableau (inference) rules. A tableau rule consists of a **numerator** above the line and a list of **denominators** (below the line). The denominators are separated by vertical bars. The numerator is a finite set of formulae and so is each denominator. We use the terms numerator and denominator rather than premiss and conclusion to avoid confusion with the sequent terminology.

Figure 4.1 (page 54) shows a common tableau system for **S4**.

Note that the tableau rules given in Figure 4.1 correspond directly to the sequent rules given in Figure 3.4 (page 49) when sequents are replaced by their associated sets and the whole rule is turned upside down. Each tableau rule is read downwards as "if the numerator is **L**-satisfiable, then so is one of the denominators". A tableau calculus $\mathcal{C}L$ is a finite collection of tableau rules identified with the set of its rule names. Thus the tableau calculi $\mathcal{C}S4 = \{(0), (\neg), (\theta), (\wedge), (\vee), (T), (S4)\}$ and $\mathcal{C}PC = \{(0), (\neg), (\theta), (\wedge), (\vee)\}$.

$$(0)\ \frac{P; \neg P}{0}$$

$$(\wedge)\ \frac{X; P \wedge Q}{X; P; Q} \qquad\qquad (\vee)\ \frac{X; \neg(P \wedge Q)}{X; \neg P \ \mid\ X; \neg Q}$$

$$(\neg)\ \frac{X; \neg\neg P}{X; P}$$

$$(T)\ \frac{X; \Box P}{X; \Box P; P} \qquad\qquad (S4)\ \frac{\Box X; \neg\Box P}{\Box X; \neg P}$$

$$(\theta)\ \frac{X; Y}{X}$$

Figure 4.1: Tableau rules for $\mathcal{C}S4$

Following Rautenberg [Rau83], a $\mathcal{CL}$**-tableau** for X is a finite tree $\mathcal{T}$ with root X whose nodes carry finite formula sets stepwise constructed by the rules of $\mathcal{CL}$ according to:

- if a rule with n denominators is applied to a node then that node has n successors with the proviso that
- if a node E carries a set Y and Y has already appeared on the branch from the root to E then E is an end node of $\mathcal{T}$.

A tableau is **closed** if all its end nodes carry $\{0\}$. A set X is $\mathcal{CL}$**-consistent** if no closed $\mathcal{CL}$-tableau for X exists. We write $\mathcal{CL}(X)$ if X is $\mathcal{CL}$-consistent and $\not{\mathcal{C}}L(X)$ otherwise. Note that both consistency as defined on page 9 and $\mathcal{CL}$-consistency are purely syntactic concepts. The aim is to show that they coincide.

The subformula property for tableau systems in primitive notation is slightly different than that for sequent systems. In a sequent $\Gamma \longrightarrow \Delta$, the left side and right side of the sequent arrow respectively act as signs representing "true" and "false". In fact, Fitting makes these signs explicit in his signed tableau [Fit83]. In our tableau systems, the formulae from the right side of the sequent arrow appear with an extra negation sign in the tableau node carrying $\Gamma \cup \neg\Delta$. Hence the "subformulae" we need to consider in our tableaux must contain the negated versions of the sequent subformulae. The following definitions cater for this change.

For any finite set X :

- let *Sf(X)* denote the set of all subformulae of all formulae in X, see page 20;
- let $\neg$*Sf(X)* denote $\{\neg P \mid P \in Sf(X)\}$;
- let $\widetilde{X}$ denote the set $Sf(X) \cup \neg Sf(X) \cup \{0\}$
- let $X^*_{S4} = \widetilde{X}$.

Thus, a tableau system $\mathcal{C}L$ has the subformula property if $X^*_L = \widetilde{X}$. As we shall see, some of our tableau systems do not have the subformula property. But in all of them, the set X^*_L is always bounded, so that the "superformulae" that may appear in a tableau node are bounded. We call this an **analytical superformula** property and formalise this with the following lemma.

Lemma 2 *If there is a closed $\mathcal{C}S4$ tableau for X then there is a closed $\mathcal{C}S4$ tableau for X with all nodes in the finite set X^*_{S4}.*

Proof: Obvious from the fact that all rules for $\mathcal{C}S4$ operate with subsets of X^*_{S4} only. •

A set X is **closed with respect to a tableau rule** if, whenever (an instantiation of) the numerator of the rule is in X, so is (a corresponding instantiation of) at least one of the denominators of the rule. If $\mathcal{C}$ is a finite collection of tableau rules then a set X is **closed with respect to $\mathcal{C}$** if it is closed with respect to each rule in $\mathcal{C}$.

In general, for each tableau calculus $\mathcal{C}L$, we shall define some $\mathcal{C} \subseteq \mathcal{C}L$ with $\{(0), (\neg), (\wedge), (\vee)\} \subseteq \mathcal{C}$ and say that a set X is $\mathcal{C}L$-saturated if it is $\mathcal{C}L$-consistent and closed with respect to $\mathcal{C}$. For instance, a set X is $\mathcal{C}S4$-saturated if it is $\mathcal{C}S4$-consistent and closed with respect to $\mathcal{C} = \{(0), (\neg), (\wedge), (\vee), (T)\}$.

Let $\mathcal{C} = \{\rho_1, \rho_2, \cdots, \rho_k\}$ be a collection of tableau rules and let X_0 be some finite set of formulae. The set X_1 is a **$\mathcal{C}$-reduction** of X_0 if X_1 results from a single application of some $\rho \in \mathcal{C}$ to X_0 and $X_1 \neq X_0$. Note that the principal formula of ρ must be in X_0 but it is possible that this principal formula is not in X_1. This is the main difference between strict-saturation and the usual definition of saturation [Fit83, Rau83]. The set X_n is a **strict-$\mathcal{C}$-saturation** of X_0 if there is a finite chain $X_0, X_1, \cdots, X_n$ of finite sets such that each X_{i+1} is a $\mathcal{C}$-reduction of X_i and no rule from $\mathcal{C}$ is applicable to X_n. That is, either X_n consists of atomic formulae and negated atomic formulae only, or all attempted $\mathcal{C}$-reductions of X_n give X_n back.

Lemma 3 *For each $\mathcal{C}S4$-consistent X there is an effective procedure to construct some finite $\mathcal{C}S4$-saturated X^* with $X^* \subseteq X^*_{S4}$.*

Proof: By definition, since X is $\mathcal{C}S4$-consistent, no $\mathcal{C}S4$-tableau for X closes. By definition, a set is $\mathcal{C}S4$-saturated if it is closed with respect to $\mathcal{C} = \{(0), (\neg), (\wedge), (\vee), (T)\}$.

Hence we can apply any number of $\mathcal{C}$-reductions knowing that at each rule application, at least one denominator is $\mathcal{C}S4$-consistent. This would give a sequence of $\mathcal{C}S4$-consistent sets $X = X_0, X_1, \cdots$. If this procedure terminates with X_n then X_n is $\mathcal{C}S4$-consistent as well and putting $X^* = X_0 \cup X_1 \cup X_2 \cup \cdots \cup X_n$ would ensure that X^* is $\mathcal{C}S4$-saturated. For example, if $X_i = Y_i; \neg(P \wedge Q)$, we first decide whether $\mathcal{C}S4(Y_i; \neg P)$ or $\mathcal{C}S4(Y_i; \neg Q)$ and then add $\neg P$ or $\neg Q$ to Y_i correspondingly to obtain X_{i+1}. That is, we saturate X step by step. Thus the important aspect is to show that this procedure terminates with some X_n where n is finite. Since the tableau rules carry a subset of X^*_{S4} to another subset of X^*_{S4}, there are only a finite number of possible values for the X_i so the only way this sequence may not terminate is if it goes into a cycle.

So suppose that $X = X_0, X_1, X_2, \cdots, X_i, X_{i+1}, \cdots, X_{n-1}, X_n, X_i$ is a cyclic sequence of $\mathcal{C}$-reductions and consider $card(X_i)$. First of all, since each member of the sequence is $\mathcal{C}S4$-consistent, the (0) rule could not have been used in obtaining this cycle. Second, the rules $(\wedge)$ and (T) each increase the cardinality in a $\mathcal{C}$-reduction and none of the rules in $\mathcal{C}$ decreases the cardinality, so neither of these two rules could have been used in obtaining this cycle. That leaves only $(\vee)$ and $(\neg)$, but note that each of these decreases the degree and so neither of these could have been used to obtain this cycle. Thus, it is not possible to obtain a cycle by repeated $\mathcal{C}$-reductions for $\mathcal{C} = \{(0), (\neg), (\wedge), (\vee), (T)\}$ and so the procedure must terminate.

Also, since each rule carries subsets of X^*_{S4} to subsets of X^*_{S4} and we start with $X \subseteq X^*_{S4}$ we have $X \subseteq X^* \subseteq X^*_{S4}$. ●

Such $\mathcal{C}S4$-saturated sets (and in general $\mathcal{C}L$-saturated sets) are important because they provide a direct connection between the syntactic and semantic aspects of tableau systems. This is the subject of the next section.

4.2 Soundness and Completeness of Modal Tableau Systems

We want to show that the purely syntactic (and proof theoretic) notion of $\mathcal{C}L$-consistency corresponds to the purely semantic notion of **L**-satisfiability. That is, we want to show that: X is **L**-satisfiable if and only if X is $\mathcal{C}L$-consistent. The following definitions are central for this aim.

Soundness of $\mathcal{C}L$: if X is **L**-satisfiable then X is $\mathcal{C}L$-consistent. Contrapositively, if X is $\mathcal{C}L$-inconsistent then X is **L**-unsatisfiable, which in symbols is, if $\not{\mathcal{C}}L(X)$ then X is **L**-unsatisfiable. In words, if there is a closed $\mathcal{C}L$-tableau for X then X has no **L**-models (i.e. no **L**-model is a model for X).

Proof Outline: To prove this claim we assume that we have a closed $\mathcal{C}L$-tableau for X: that is, $\not{\mathcal{C}}L(X)$. Then we use induction on the structure of this tableau to show that X is **L**-unsatisfiable.

The base case is when the tableau consists of just one application of the (0) rule. In this case, the set X must contain some P and also $\neg P$ and is clearly **L**-unsatisfiable.

Now suppose that the (closed) $\mathcal{C}L$-tableau is some finite but arbitrary tree. We know that all leaves of this tableau end in $\{0\}$. So all we have to show is that for each $\mathcal{C}$L-tableau rule: if all the denominators are **L**-unsatisfiable, then the numerator is **L**-unsatisfiable. This would allow us to conclude that the root X is **L**-unsatisfiable since we know that the leaves are **L**-unsatisfiable. Instead we show the contrapositive; that is, for each $\mathcal{C}L$-tableau rule we show that if the numerator is **L**-satisfiable then at least one of the denominators is **L**-satisfiable. •

Completeness of $\mathcal{C}L$: if X is $\mathcal{C}L$-consistent then X is **L**-satisfiable. In words, if there is no closed $\mathcal{C}L$-tableau for X then X has an **L**-model (i.e. there is an **L**-model which is an **L**-model for X).

Proof Outline: To prove this claim we assume that no $\mathcal{C}L$-tableau for X closes. Then we construct an **L**-model $\mathcal{M}$ for X from a *particular* sequence of $\mathcal{C}L$-tableau rule applications, knowing that at each rule application, this sequence does not produce a closed $\mathcal{C}L$-tableau. The basic idea is due to Hintikka [Hin55]. •

Finite Model Property: In the above procedure, if $\mathcal{M}$ can be chosen finite (for finite X) then the logic $\mathbf{L}\mathcal{C} = \{P \mid \not{\mathcal{C}}L\,(\neg P)\}$ defined by $\mathcal{C}L$ has the finite model property (fmp).

The following definition from Rautenberg [Rau83] is central for the model construction mentioned above. Note the similarities with the informal modal rules of Hintikka mentioned on page 12. A **model graph** for some finite fixed set of formulae X is a finite **L**-frame $\langle W_0, R\rangle$ such that all $w \in W_0$ are $\mathcal{C}L$-saturated sets with $w \subseteq X_L^*$ and

(i) $X \subseteq w_0$ for some $w_0 \in W_0$;

(ii) if $\neg\Box P \in w$ then there exists some $w' \in W_0$ with wRw' and $\neg P \in w'$;

(iii) if wRw' and $\Box P \in w$ then $P \in w'$.

Lemma 4 *If $\langle W_0, R\rangle$ is a model graph for X then there exists an* **L**-*model $\langle W_0, R, \vartheta\rangle$ for X. [Rau83].*

Proof: An implicit assumption is that $\mathcal{C}L$-saturated sets are closed with respect to each of $(0), (\neg), (\wedge)$, and $(\vee)$ but we have already mentioned that $\mathcal{C}PC$ is a subset of all our tableau systems. Take the valuation ϑ from atomic propositions to subsets of W_0, where $\vartheta : p \mapsto \{w \in W_0 \mid p \in w\}$. Using simultaneous induction on the degree of $P \in w_0$ we show that

(a) $P \in w$ implies $w \models P$ and

(b) $\neg P \in w$ implies $w \not\models P$.

Base case: Suppose $p \in w$ where $p \in \mathcal{P}$. By definition of ϑ, $w \in \vartheta(p)$ and so by definition of $\models$, $w \models p$.

Base case: Suppose $\neg p \in w$ where $p \in \mathcal{P}$. Since w is $\mathcal{CL}$-consistent, $p \notin w$. Hence $w \notin \vartheta(p)$ and so $w \not\models p$. By definition of $\models$, $w \models \neg p$.

Induction Hypothesis: Assume for all P with $deg(P) \leq k$, and all $w \in W_0$ that $P \in w$ implies $w \models P$ and that $\neg P \in w$ implies $w \not\models P$, where k is some fixed $k \geq 1$.

Induction Step: Suppose $Q \in w$ with $deg(Q) = k + 1$ and consider the structure of Q. We have to show that $w \models Q$ and that if $Q = \neg P$ then $w \not\models P$.

$(\neg)$: if $Q = \neg P$ then since w is $\mathcal{CL}$-consistent we have $P \notin w$ and $deg(P) = k$. Since $\neg P \in w$, the induction hypothesis gives $w \not\models P$. By definition of $\models$, $w \models \neg P$ which means that $w \models Q$ and we are done.

$(\wedge)$: if $Q = P_1 \wedge P_2$ then since w is $\mathcal{CL}$-saturated we have $P_1 \in w$ and $P_2 \in w$ by $(\wedge)$ and also that $deg(P_1) \leq k$ and $deg(P_2) \leq k$. By the induction hypothesis this implies that $w \models P_1$ and $w \models P_2$. By definition of $\models$, $w \models P_1 \wedge P_2$. That is, $w \models Q$.

$(\vee)$: if $Q = \neg(P_1 \wedge P_2)$ then since w is $\mathcal{CL}$-saturated we have $\neg P_1 \in w$ or $\neg P_2 \in w$ by $(\vee)$ and also that $deg(\neg P_1) \leq k$ and $deg(\neg P_2) \leq k$. By the induction hypothesis this implies that $w \models \neg P_1$ or $w \models \neg P_2$ and hence that $w \not\models P_1$ or $w \not\models P_2$. By definition of $\models$, $w \not\models (P_1 \wedge P_2)$ i.e. $w \models Q$.

$(\Box)$: if $Q = \Box P$ then $deg(P) = k$. Suppose $w' \in W_0$ and wRw', then by (iii), $P \in w'$ and then by the induction hypothesis, $w' \models P$. But w' was any arbitrary world with wRw' so for all $w' \in W_0$, wRw' implies $w' \models P$. By definition of $\models$, $w \models \Box P$ i.e. $w \models Q$.

$(\neg\Box)$: if $Q = \neg\Box P$ then by (ii) there exists $w' \in W_0$ such that wRw' and $\neg P \in w'$. But $deg(P) = k - 1$ and hence $deg(\neg P) = k$. By the induction hypothesis, $w' \models \neg P$ and $w' \not\models P$. By definition of $\models$, $w \models \neg\Box P$.

By (a), $w_0 \models X$ hence $\langle W_0, R, \vartheta \rangle$ is an **L**-model for X. •

This model graph construction is similar in spirit to the subordinate frames construction of Hughes and Cresswell [HC84] except that Hughes and Cresswell use maximal consistent sets and do not consider cycles, giving infinite models rather than finite models.

4.3 Soundness and Completeness of $\mathcal{C}S4$

4.3.1 Soundness of $\mathcal{C}S4$

Theorem 6 *The $\mathcal{C}S4$ rules are sound with respect to* **S4**-*frames.*

For each rule in $\mathcal{C}S4 = \{(0), (\neg), (\wedge), (\vee), (\theta), (T), (S4)\}$ we have to show that if the numerator of the rule is **S4**-satisfiable then so is at least one of the denominators.

Proof for (0)**:** The numerator $P; \neg P$ is never **S4**-satisfiable and neither is the denominator 0.

Proof for (θ)**:** We have to show that if $X; Y$ is **S4**-satisfiable, then so is X. Suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **S4**-model and $w_0 \in W$ is such that $w_0 \models X; Y$. Since $X; Y$ is just $X \cup Y$, $w_0 \models X$ and $w_0 \models Y$ and we are done.

Proof for ($\neg$)**:** We have to show that if $X; \neg\neg P$ is **S4**-satisfiable, then so is $X; P$. Suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **S4**-model and $w_0 \in W$ is such that $w_0 \models X; \neg\neg P$. By definition of $\models$, $w_0 \not\models \neg P$ meaning that $w_0 \models X; P$.

Proof for ($\wedge$)**:** We have to show that if $X; P \wedge Q$ is **S4**-satisfiable, then so is $X; P; Q$. Suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **S4**-model and $w_0 \in W$ is such that $w_0 \models X; P \wedge Q$. By definition of $\models$ this means that $w_0 \models P$ and $w_0 \models Q$ so $w_0 \models X; P; Q$.

Proof for ($\vee$)**:** We have to show that if $X; \neg(P \wedge Q)$ is **S4**-satisfiable, then so is at least one of $X; \neg P$ and $X; \neg Q$. Suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **S4**-model and $w_0 \in W$ is such that $w_0 \models X; \neg(P \wedge Q)$. By definition, $w_0 \not\models P \wedge Q$, that is, not $w_0 \models P \wedge Q$. Thus not $w_0 \models P$ or not $w_0 \models Q$ which means that $w_0 \not\models P$ or $w_0 \not\models Q$ which means that $w_0 \models \neg P$ or $w_0 \models \neg Q$. Hence $w_0 \models X; \neg P$ or $w_0 \models X; \neg Q$.

Proof for (T)**:** We have to show that if $X; \Box P$ is **S4**-satisfiable, so is $X; \Box P; P$. Suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **S4**-model and $w_0 \in W$ is such that $w_0 \models X; \Box P$. Since $\mathcal{M}$ is an **S4**-model, R is reflexive, hence $w_0 R w_0$. Then by definition of $\models$, $w_0 \models \Box P$ implies that $w_0 \models P$, hence $w \models X; \Box P; P$.

Proof for ($S4$)**:** Since there is only one denominator, we have to show that if $\Box X; \neg\Box P$ is **S4**-satisfiable, then $\Box X; \neg P$ is **S4**-satisfiable. Finiteness of the model is not essential. Assume that $\Box X; \neg\Box P$ is **S4**-satisfiable. That is, there is an **S4**-model $\mathcal{M} = \langle W, R, V \rangle$ with $w_0 \in W$ such that $w_0 \models \Box X; \neg\Box P$. Then by the definition of $\models$ there exists $w \in W$ such that $w_0 R w$ and $w \models \neg P$. By the transitivity of R, $w_0 \models \Box X$ implies $w \models \Box X$. By the reflexivity of R, $w \models \Box X$ implies $w \models X$. We have just shown that $w \models X; \Box X; \neg P$ which means that $\mathcal{M}$ satisfies $X; \Box X; \neg P$. •

Semantic Intuitions: In summary, the soundness of the ($S4$) and (T) rules follows from the semantics for $\neg\Box P$ as "eventually there is a world where P is false"; from the guaranteed seriality of R for **S4**-models by the reflexivity of R; and from the transitivity of R for **S4**-models. The ($S4$) rule can be seen as a "jump" to the world where $\neg P$ eventually becomes true [Fit83]. That is, we can identify the numerator with w_0 and the denominator with w.

4.3.2 Completeness of $\mathcal{C}S4$

Theorem 7 *If X is a finite set of formulae and X is $\mathcal{C}S4$-consistent then there is an* **S4**-*model for X on a finite frame $\langle W_0, R\rangle$ which is a finite (reflexive and transitive) tree of nondegenerate clusters (and hence is an* **S4**-*frame) [Rau83].*

Proof : The construction of the model graph is due to Rautenberg [Rau83] where $\prec$ denotes the immediate successor relation. By Lemma 3 (page 55) we can construct some $\mathcal{C}S4$-saturated $X^* = w_0$ with $X \subseteq w_0 \subseteq X^*_{S4}$. If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph since it is an **S4**-frame and (i)-(iii) are satisfied. Otherwise, let $Q_1, Q_2, \cdots Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$ and $\neg Q_i \notin w_0$.

For any set Y, let $Y^\Box = \{P | \Box P \in Y\}$ and let $\Box Y = \{\Box P | P \in Y\}$.

Put $w' = w_0^\Box$. Since $\Box w' \subseteq w_0$, $\Box w' \cup \{\neg\Box Q_i\}$ is $\mathcal{C}S4$-consistent by (θ); hence so is each $X_i = \Box w' \cup \{\neg Q_i\},\ i = 1, \cdots, m$ by $(S4)$.

Clearly for each X_i we can find some $\mathcal{C}S4$-saturated $v_i \supseteq X_i$, with $v_i \subseteq X^*_{S4}$. Put $w_0 \prec v_i$, $i = 1, \cdots, m$ and call v_i the Q_i-successor of w_0. These are the immediate successors of w_0. Now repeat the construction with each v_i thus obtaining the nodes of level 2 and so on.

In general, the above construction of $\langle W_0, \prec\rangle$ runs ad infinitum. However, since $w \in W_0$ implies $w \subseteq X^*$, a sequence $w_0 \prec w_1 \prec \cdots$ in $\langle W_0, \prec\rangle$ either terminates, or a node repeats. If in the latter case $n > m$ are minimal with $w_n = w_m$ we stop the construction and identify w_n and w_m in $\langle W_0, \prec\rangle$ thus obtaining a circle instead of an infinite path. One readily confirms that $\langle W_0, R\rangle$ is a model graph for X where R is the reflexive and transitive closure of $\prec$. It is obvious that clusters in $\langle W_0, R\rangle$ form a tree.

By Lemma 4 (page 57), $\langle W_0, R\rangle$ is an **S4**-model graph for X where $\vartheta : p \mapsto \{w \in W_0 \mid p \in w\}$ and hence $\langle W_0, R, \vartheta\rangle$ is an **S4**-model for X. •

4.4 Decision Procedures for S4

The tableau systems give nondeterministic and deterministic decision procedures as described below.

4.4.1 A Nondeterministic Decision Procedure for S4

The **S4**-model we constructed in the completeness proof is finite, hence **S4** has the finite model property; see page 57. We already know that **S4** is finitely axiomatisable as **KT4** therefore **S4** is decidable by Theorem 4, page 30. We now show that $\mathcal{C}S4$ is a decision procedure for **S4**.

The soundness of $CS4$ guarantees that if there is a closed $CS4$-tableau for X then X has no **S4**-models. In particular, if we put $X = \{\neg A\}$ then this means that if there is a closed $CS4$-tableau for $\{\neg A\}$ then $\{\neg A\}$ has no **S4**-models.

The completeness of $CS4$ guarantees that if there is no closed $CS4$-tableau for X then X has an **S4**-model. In particular, if we put $X = \{\neg A\}$ then this means that if there is no closed $CS4$-tableau for $\{\neg A\}$ then $\{\neg A\}$ has an **S4**-model.

To test whether a formula A is **S4**-valid, we simply have to run a $CS4$-tableau construction for $X = \{\neg A\}$. Since X^*_{S4} is finite, there are only a finite number of such $CS4$-tableaux. If one of them is closed then, by soundness, $\{\neg A\}$ has no **S4**-models, and hence A is **S4**-valid. If none of these $CS4$-tableaux closes then, by completeness, we can construct a finite model graph $\langle W_0, R\rangle$ which satisfies $\neg A$, hence A is not **S4**-valid.

We already know that the axiomatically formulated logic **S4** is characterised by **S4**-frames. That is, we know that a formula A is **S4**-valid iff it is an **S4**-theorem. Therefore, $CS4$ is a highly nondeterministic decision procedure for **S4**.

4.4.2 A Deterministic Decision Procedure for S4

There is, however, a completely different deterministic decision procedure for **S4** in the $CS4$ completeness proof.

The $CS4$ completeness proof is a satisfiability test; that is, it is a (deterministic) procedure which uses $CS4$-saturated sets to construct a finite **S4**-model for some finite set X. To construct a $CS4$-saturated set from some given X we can use the procedure given by the proof of Lemma 3 (page 55). Therefore we can attempt to construct an **S4**-model for X by attempting to form a tree of clusters of $CS4$-saturated sets according to the recipe given by the completeness proof for $CS4$.

In the completeness proof we know that each Q-successor we create is $CS4$-consistent because our initial assumption is that X has no closed $CS4$-tableau and hence that X is $CS4$-consistent. We can no longer make this assumption since this is exactly what we have to determine. Instead, we attempt to create a $CS4$-saturated X^* from X. If successful, we attempt to create all the $CS4$-saturated Q_i-successors of X^*. If we find that any one of them is inconsistent because it contains some P and $\neg P$, then we know that X is **S4**-unsatisfiable. If all of these Q_i-successors of level 1 are consistent then we attempt to construct the nodes of level 2 and so on. We know that this procedure must terminate since there are only a finite number of nodes in X^*_{S4}. If at any time we find that a Q_i-successor duplicates a previous node then we simply form a cycle as in the completeness proof.

If this procedure terminates with all nodes being consistent then the resulting $\langle W_0, R\rangle$ is an **S4**-model for X and hence X is **S4**-satisfiable. On the other hand, if this procedure terminates having found some inconsistent node, then X is not **S4**-satisfiable. In particular putting $X = \{\neg A\}$ allows us to determine whether $\{\neg A\}$ is **S4**-satisfiable or **S4**-unsatisfiable. But if $\{\neg A\}$ is **S4**-satisfiable then A is not **S4**-valid, and if $\{\neg A\}$ is

S4-unsatisfiable then A is **S4**-valid. That is, this allows us to test whether or not A is **S4**-valid and hence whether or not A is an **S4**-theorem.

The deterministic decision procedure described above is the basis of most decision procedures for temporal logics as exemplified by those of Wolper [Wol83]. In fact, the "tableau" methods of Kripke and the semantic diagrams of Hughes and Cresswell [HC68], which we shall discuss in Chapter 7, are just such decision procedures.

There is a duality between these deterministic decision procedures and the tableau systems we present. That is, it is possible to obtain each from the other. But as we shall see, there is a systematic way to obtain tableau rules for modal logics and the resulting completeness proofs (if constructive) then give the deterministic decision procedures we seek. In the process, we can usually also show that cut is eliminable. But this duality is only possible when the tableau completeness proofs are *constructive*. That is, we must actually give a recipe for constructing a model graph $\langle W_0, R \rangle$ rather than just show that some such model graph must exist.

Obtaining tableau systems from the deterministic decision procedures is usually not obvious. For example, Hughes and Cresswell [HC68] mention a deterministic decision procedure for **S4.3**, but it is not obvious how to obtain a tableau system for **S4.3** from this method.

4.5 Eliminating Thinning

In all of our modal tableau systems, the only structural rule is (θ). As stated on page 43, the thinning (or weakening) rule introduces a form of nondeterminism where we have to guess which formula to throw away. That is, we have to guess which formulae are really essential to the proof.

The (θ) rule can be eliminated by building the effects of (θ) into the transitional rule $(S4)$ by changing it from

$$(S4)\ \frac{\Box X; \neg\Box P}{\Box X; \neg P} \qquad \text{to} \qquad \frac{Y; \neg\Box P}{Y'; \neg P}$$

where $Y' = \{\Box Q \mid \Box Q \in Y\}$ and simultaneously changing the basic axiomatic tableau rule from (0) to $(0')$ as shown below

$$(0)\ \frac{P; \neg P}{0} \qquad\qquad (0')\ \frac{X; P; \neg P}{0}$$

This technique is used by Fitting [Fit83].

4.6 Relating Sequents Systems and Tableau Systems

The tableau calculus presented in the last section is a standard method of testing a set of formulae for **L**-satisfiability. As there is an obvious translation between the tableau and sequent rules, any closed $\mathcal{C}L$-tableau for some set $\Gamma \cup \neg\Delta$ can be used to derive an $\mathcal{S}L$-proof of $\Gamma \longrightarrow \Delta$. As long as $\mathcal{C}L$ is sound and complete with respect to **L**-frames we can formalise this translation via the following proposition.

Proposition 2

(a) $\Gamma \longrightarrow \Delta$ *is* **L**-*valid iff*

(b) $\Gamma \cup \neg\Delta$ *is* **L**-*unsatisfiable iff*

(c) $\Gamma \cup \neg\Delta$ *is* $\mathcal{C}L$-*inconsistent iff*

(d) $\not{\mathcal{C}}L(\Gamma \cup \neg\Delta)$ *iff*

(e) $\Gamma \cup \neg\Delta$ *has a closed* $\mathcal{C}L$-*tableau iff*

(f) $\Gamma \longrightarrow \Delta$ *is* $\mathcal{S}L$-*provable.*

Proof: By Proposition 1 (page 50) (a) iff (b). By the soundness and completeness of $\mathcal{C}L$ with respect to **L**-frames, (b) iff (c). By definition, (c) iff (d) and (d) iff (e). Each tableau rule corresponds to a sequent rule by the fact that $\Gamma \longrightarrow \Delta$ has associated set $\Gamma \cup \neg\Delta$. Thus the closed $\mathcal{C}L$-tableau can be turned into an $\mathcal{S}L$-proof of $\Gamma \longrightarrow \Delta$. The proof of (e) iff (f) proceeds by induction on the length of the closed $\mathcal{C}L$-tableau; see Fitting [Fit83]. •

In particular, for a single formula, we can test for theoremhood via the following proposition.

Proposition 3 *If* $\mathcal{C}L$ *is sound and complete with respect to* **L**-*frames then* A *is a theorem of axiomatic logic* **L** *iff* $\longrightarrow A$ *is* $\mathcal{S}L$-*provable iff* $\not{\mathcal{C}}L\{\neg A\}$.

Proposition 4 *If* $\mathcal{C}L$ *is sound and complete with respect to* **L**-*frames then* **L**$\mathcal{C}$= **L**$\mathcal{S}$= **L**.

Proof: If $\mathcal{C}L$ is sound and complete with respect to **L**-frames then **L**$\mathcal{C}$ is determined by the class of **L**-frames. We know that $A \in$ **L**$\mathcal{C}$ iff $\not{\mathcal{C}}L\{\neg A\}$ which by Proposition 2 occurs iff $\longrightarrow A$ is $\mathcal{S}$**L**-provable iff $A \in$ **L**$\mathcal{S}$. Hence **L**$\mathcal{C}$= **L**$\mathcal{S}$. But we have deliberately defined **L**-frames to be the class of (finite) frames known to characterise the axiomatically formulated logic **L** (Theorem 5 page 34). Hence **L**$\mathcal{C}$= **L**$\mathcal{S}$= **L**. •

So the two calculi $\mathcal{S}L$ and $\mathcal{C}L$ are essentially the same. Both provide a decision procedure for **L** because X^*_L is finite for finite X so that there are only a finite number of $\mathcal{C}L$-tableaux for X. If we try all such $\mathcal{C}L$ tableaux and find that none is closed then we can declare that X is $\mathcal{C}L$-consistent, and hence, **L**-satisfiable. If at any time in this procedure we find that one of the $\mathcal{C}L$-tableaux closes then we can declare X is $\mathcal{C}L$-inconsistent and hence **L**-unsatisfiable.

4.6.1 Smullyan's Tableau Formulation

Tableau calculi originated in the work of Beth [Bet53, Bet55] and since then they have been refined by various authors into the form we have presented them. One of the most popular alternative formulations of tableau is that of Smullyan [Smu68a] and the subsequent extensions of it to modal logics by Fitting [Fit83].

In Smullyan-tableaux, the underlying construction is also a tree. But each node carries only one formula instead of a set of formulae. If we wish to construct a Smullyan-tableau for a set $Y \cup \{\neg A\}$ then we write each member of Y in a separate node, one beneath the other. Then we add one more node containing $\neg A$ to give a linear tree. Each tableau rule allows us to add its denominators to the end of a branch if the branch already contains its numerator. For example, a disjunctive formula like $A \vee B$ leads to a fork or branching node in the tree. That is, if $A \vee B$ is on the path to some leaf then that leaf is allowed to have two children, one containing A and the other containing B. Smullyan-tableaux implicitly allow contraction since a formula is not deleted from a branch once it is used as a numerator of a rule application. Fitting [Fit88] introduces *strict* tableau by marking a formula as used once a rule is applied to it and restricting a rule application to unmarked formulae.

Smullyan-tableaux were designed for their ease of use with pencil and paper. The idea is to save the duplication of formulae that are common to the numerator and the denominator of our Beth-tableau rules. Thus each branch of a Smullyan-tableau corresponds to one of the sets carried by one of the nodes of our Beth-tableau. For modal Smullyan-tableaux, however, some sort of thinning rule is required since some rules involve a "jump" to another world. Fitting [Fit83] handles this by simply crossing out the formulae that do not survive the "jump". But then he faces a problem, for although one branch may involve a "jump", its sister branch might not. The formulae above their point of departure, which are common to them both, must not be deleted from the branch that does not involve a "jump". Fitting adds a "repetition rule" to copy these formulae down onto the branch that requires them [Fit83, page 37-38]. Consequently, much crossing out and copying of formulae is needed, and the efficacy of minimising the duplication of formulae in the numerator and denominator as in Beth-tableau is lost.

4.7 Discussion

Much is made of the subformula property for a logic. As we saw in $\mathcal{SPC}$ the strict subformula property guaranteed that $\mathcal{SPC}$ was deterministic since we did not have to worry about cycles. But it was invertibility that won the day, since it meant any non-proof gave a counter-model.

For most modal logics we lose the strict subformula property due to the need for contraction (either explicit contraction as a rule or built in as in the $(S4)$ rule). Thus we are already doomed to worry about cycles. Furthermore, in most cases, there is a rule like $(S4)$ or (θ) that loses information and so we lose invertibility as well. Thus we are doomed to search the whole search space in order to declare a formula a non-theorem. This is the basic reason why the decision problem for most propositional modal logics is PSPACE-complete whereas the decision problem for **PC** is (only!) NP-complete [SC85].

The point of this discussion then is that the subformula property is not essential for decidability and is not as crucial as it has been made out to be from a theoretical aspect. Of course, structure sharing techniques like those of Wallen [Wal87] require the subformula property – so it is important from a practical aspect.

For most modal logics we require some form of nondeterminism, for example in choosing the principal formula of the $(S4)$ rule when the numerator contains more than one formula of the form $\neg\Box P$. One way of handling nondeterminism is to use parallel implementations of our decision procedures. And one of the simplest ways to do so is to use the *replication* of computation over a distributed network of processors working in parallel [Clo87, Kle90], rather than structure sharing. Thus rather than striving for special tricks or embeddings to regain the subformula property, it may be better to accept both nondeterminism and the analytical superformula property, and to attack these with parallelism.

4.8 Deducibility Relations

We require known characterisation results in order to be sure that $\mathbf{L}\mathcal{C} = \mathbf{L}$ (and hence that $\mathbf{L}\mathcal{S} = \mathbf{L}$). That is, we need to rely on proofs that **L** is indeed characterised by the frames we call **L**-frames. This is because our soundness and completeness proofs are totally semantic. One way around this is to work with deducibility relations as is done by Rautenberg [Rau83]. Instead of defining a logic to be the set of theorems of an axiomatic system, Rautenberg defines a finitary deducibility relation $\vdash$ between finite formula sets. The deducibility relation $\vdash$ is **normal** if it is closed under:

$$(\Box) \quad \frac{X \vdash P}{\Box X \vdash \Box P} \quad \text{(normality rule)} .$$

The deducibility relation is complete for **PC** if

$(k\wedge)$ $\quad \{P, Q\} \vdash P \wedge Q \vdash \{P, Q\}$

$(k\neg)$ $\quad X \wedge \neg P \vdash 0 \quad$ *iff* $\quad X \vdash P,$

$(k0)$ $\quad 0 \vdash P.$

The latter imply the usual deduction theorem

(Dt) $\quad \{X, P\} \vdash Q$ if and only if $X \vdash P \Rightarrow Q$.

The familiar normality conditions for **L** derive easily from $(\Box)$ and (Dt). Rautenberg writes $\vdash^L$ for $\vdash$ since **L** uniquely determines $\vdash$, by (Dt). The symbol $\vdash^L P$ stands for $\emptyset \vdash^L P$ (i.e. $P \in \mathbf{L}$). A set X is said to be **L**-consistent, if $X \not\vdash^L 0$. By $(k\neg)$, **L**-consistency involves a characterisation of the consequence relation $\vdash^L$, in particular,

$\vdash^L P$ if and only if $\{\neg P\} \vdash^L 0$.

Bull and Segerberg [BS84] consider infinitary deducibility relations and show that if $\vdash$ is compact then $\vdash$ and the usual axiomatic formulations define the same logic. We have not defined the compactness property of a logic but Rautenberg's deducibility relation is compact since it deals only with finite sets.

When working with deducibility relations, soundness has to be proved *syntactically*, that is *formally*, by showing that for every rule of the deducibility relation, the denominator is provable whenever the numerator is provable. As stated previously, such syntactic proofs of soundness are easy for logics like **S4** but not at all obvious for others. On the other hand, most of the soundness proofs of the next chapter are based on semantic insights and are not too difficult.

Chapter 5

Various Tableau (and Sequent) Systems

In this chapter we present a unified treatment of tableau systems for many propositional normal modal logics based on the work of Rautenberg [Rau83], Fitting [Fit83], Shvarts [Shv89] and Hanson [Han66a]. Most of the systems are cut-free but even those that are not use only an analytical cut rule. Each tableau system immediately gives an analogous (cut-free) sequent system. The presentation is based on the basis laid down in the last chapter and is therefore rather repetitive. The procedure for each tableau system $\mathcal{C}L$ is:

1. define the tableau rules for $\mathcal{C}L$;
2. define X^*_L for a given fixed X and given $\mathbf{L}$;
3. define $\mathcal{C}L$-saturation ;
4. prove that each $\mathcal{C}L$-consistent X can be extended (effectively) to a $\mathcal{C}L$-saturated X^* with $X \subseteq X^* \subseteq X^*_L$;
5. prove that the $\mathcal{C}L$ rules are sound with respect to $\mathbf{L}$-frames;
6. prove that the $\mathcal{C}L$ rules are complete with respect to $\mathbf{L}$-frames by giving a procedure to construct a *finite* $\mathbf{L}$-model for any finite $\mathcal{C}L$-consistent X and hence prove that $\mathbf{L}$ has the finite model property, that $\mathbf{L}$ is decidable and that $\mathcal{C}L$ is a decision procedure for $\mathbf{L}$.

The first section contains my own soundness proofs but contains (corrected and expanded) completeness proofs surveyed by Rautenberg [Rau83]. Corrected because some of Rautenberg's rules contain a minor technical flaw that renders some of his tableau systems incomplete. Rautenberg proves soundness directly with respect to a finitary axiomatic consequence relation as mentioned in the last section, and since I do not use that approach, I have had to prove soundness semantically.

The second section relates the work of Fitting to Rautenberg's work, demonstrating a tradeoff between rules that require superformulae and rules that carry more semantic information.

The third section relates the work of Shvarts to that of Fitting and Rautenberg.

The forth section highlights the utility of the analytic cut rules (sfc) and $(sfcT)$ via the work of Hanson.

The fifth section presents various alternative tableau systems to those of Rautenberg, Fitting, Shvarts and Hanson. This work is mildly original.

5.1 Rautenberg's Tableau Systems

All the tableau calculi contain the rules of CPC and one or more logical rules from Figure 5.1 on page 70. The sequent analogues of the tableau rules are shown in Figure 5.2 on page 71. The tableau systems are shown in Figure 5.3 on page 72 and the only structural rule is (θ). The logical rules are categorised into two sorts, static rules or transitional rules, as follows:

Static Rules	Transitional Rules
(T), (D), (B), (5), (sfc), $(sfcT)$	(K), $(K4)$, $(S4)$, (G), (Grz)

We have seen that the tableau method is a search for a counter model. The intuition behind this sorting is that in the static rules, the numerator and denominator represent the same world, whereas in the transitional rules, the numerator and denominator represent different worlds.

The semantic and sometimes axiomatic intuitions behind these rules are as follows.

Intuitions for (K) **:** if the numerator represents a world w where $\Box X$ and $\neg\Box P$ are true, then there must be a world w' representing the denominator with wRw' such that w' makes P false and makes all the formulae in X true;

Intuitions for (T) **:** if the numerator represents a world w where X and $\Box P$ are true, then by reflexivity of R the world w itself must also make P true.

Intuitions for (D) **:** if the numerator represents a world w where X and $\Box P$ are true, then by seriality of R there must exist some w' such that wRw' and P must be true at w'. Hence $\neg\Box\neg P$, that is $\Diamond P$, must be true at w.

Intuitions for $(K4)$ **:** if the numerator represents a world w where $\Box X$ and $\neg\Box P$ are true, then by transitivity of R there must be a world w' representing the denominator, with wRw', such that w' makes X and $\Box X$ true and makes P false.

Intuitions for $(S4)$ **:** if the numerator represents a world w where $\Box X$ and $\neg\Box P$ are true, then by transitivity of R there must be a world w' representing the denominator, with wRw', such that w' makes $\Box X$ true and makes P false.

Intuitions for (B) **:** if the numerator represents a world w where X and $\neg\Box P$ are true, we know that this world either makes P true or makes P false. If w makes P true then we have the left denominator. If w makes P false, then we have the right denominator which also contains $\Box\neg\Box P$ since $A \Rightarrow \Box\Diamond A$ is a theorem of **B**.

Intuitions for (5) **:** Suppose the numerator represents a world w where X and $\neg\Box P$ are true. Then we immediately have that w also makes $\Box\neg\Box P$ true since $\neg\Box A \Rightarrow \Box\neg\Box A$ is just another way of writing the **S5** axiom 5.

Intuitions for (sfc) **:** if the numerator represents a world w where $\neg(P \land Q)$ is true, then we know that w either makes both P and Q false; or makes P false and Q true; or makes P true and Q false. The other cases use similar intuitions.

Intuitions for $(sfcT)$ **:** as for the (sfc) rule except that by reflexivity we cannot have both $\Box P$ and $\neg P$ true at w so one of the cases cannot occur.

Intuitions for (G) **:** The axiom G is

$$\Box(\Box A \Rightarrow A) \Rightarrow \Box A.$$

The contrapositive is

$$\neg\Box A \Rightarrow \neg(\Box(\Box A \Rightarrow A))$$

which is the same as

$$\neg\Box A \Rightarrow \Diamond(\Box A \land \neg A).$$

Thus, if the numerator represents a world where $\neg\Box P$ is true, then there exists another world where $\Box P$ is true and P is false. The denominator represents this world.

Intuitions for (Grz) **:** The axiom Grz is

$$\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \Rightarrow A.$$

It is known that 4 and T are theorems of **Grz** [HC84, page 111], hence **S4** $\subseteq$ **Grz**. Segerberg [Seg71, page 107] shows that in the field of **S4**, the axiom Grz is equivalent to the axiom

$$\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \Rightarrow \Box A$$

which gives the following as theorems of **Grz**:

$$\begin{aligned} \neg\Box A &\Rightarrow \neg\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \\ \neg\Box A &\Rightarrow \Diamond(\Box(A \Rightarrow \Box A) \land \neg A). \end{aligned}$$

Thus, if $\neg\Box P$ is true at the numerator, then there exists some world where $\Box(P \Rightarrow \Box P) \land \neg P$ eventually becomes true. The denominator of (Grz) represents this world.

$$(K)\ \frac{\Box X; \neg\Box P}{X; \neg P}$$

$$(T)\ \frac{X; \Box P}{X; \Box P; P} \qquad (D)\ \frac{X; \Box P}{X; \Box P; \neg\Box\neg P}$$

$$(K4)\ \frac{\Box X; \neg\Box P}{X; \Box X; \neg P} \qquad (S4)\ \frac{\Box X; \neg\Box P}{\Box X; \neg P}$$

$$(B)\ \frac{X; \neg\Box P}{X; \neg\Box P; P \mid X; \neg\Box P; \neg P; \Box\neg\Box P} \qquad (5)\ \frac{X; \neg\Box P}{X; \neg\Box P; \Box\neg\Box P}$$

$$(sfc)\quad \frac{X; \neg(P \wedge Q)}{X; \neg P; \neg Q \mid X; \neg P; Q \mid X; P; \neg Q} \qquad \frac{X; \neg\Box P}{X; \neg\Box P; P \mid X; \neg\Box P; \neg P} \qquad \frac{X; \Box P}{X; \Box P; P \mid X; \Box P; \neg P}$$

$$(sfcT)\quad \frac{X; \neg(P \wedge Q)}{X; \neg P; \neg Q \mid X; \neg P; Q \mid X; P; \neg Q} \qquad \frac{X; \neg\Box P}{X; \neg\Box P; P \mid X; \neg\Box P; \neg P}$$

$$(G)\ \frac{\Box X; \neg\Box P}{X; \Box X; \neg P; \Box P} \qquad (Grz)\ \frac{\Box X; \neg\Box P}{X; \Box X; \neg P; \Box(P \Rightarrow \Box P)}$$

Figure 5.1: Rautenberg's Tableau Rules

$$\frac{\Gamma \longrightarrow A}{\Box\Gamma \longrightarrow \Box A}\ (\rightarrow \Box : K)$$

$$\frac{\Gamma, A, \Box A \longrightarrow \Delta}{\Gamma, \Box A \longrightarrow \Delta}\ (\Box \rightarrow: T) \qquad \frac{\Gamma, \Box A, \Diamond A \longrightarrow \Delta}{\Gamma, \Box A \longrightarrow \Delta}\ (\Box \rightarrow: D)$$

$$\frac{\Gamma, \Box\Gamma \longrightarrow A}{\Box\Gamma \longrightarrow \Box A}\ (\rightarrow \Box : K4) \qquad \frac{\Box\Gamma \longrightarrow A}{\Box\Gamma \longrightarrow \Box A}\ (\rightarrow \Box : S4)$$

$$\frac{\Gamma, A \longrightarrow \Box A, \Delta \quad \Gamma \longrightarrow \Diamond\Box A, A, \Box A, \Delta}{\Gamma \longrightarrow \Box A, \Delta}\ (\rightarrow \Box : B) \qquad \frac{\Gamma \longrightarrow \Diamond\Box A, \Box A, \Delta}{\Gamma \longrightarrow \Box A, \Delta}\ (\rightarrow \Box : 5)$$

$$\frac{\Gamma \longrightarrow A, B, \Delta \quad \Gamma, B \longrightarrow A, \Delta \quad \Gamma, A \longrightarrow B, \Delta}{\Gamma \longrightarrow A \wedge B, \Delta}$$

$$\frac{\Gamma, A \longrightarrow \Box A, \Delta \quad \Gamma \longrightarrow A, \Box A, \Delta}{\Gamma \longrightarrow \Box A, \Delta} \qquad (\rightarrow: sfc)$$

$$\frac{\Gamma, A, \Box A \longrightarrow \Delta \quad \Gamma, \Box A \longrightarrow A, \Delta}{\Gamma, \Box A \longrightarrow \Delta}$$

$$\frac{\Gamma \longrightarrow A, B, \Delta \quad \Gamma, B \longrightarrow A, \Delta \quad \Gamma, A \longrightarrow B, \Delta}{\Gamma \longrightarrow A \wedge B, \Delta}$$

$$\frac{\Gamma, A \longrightarrow \Box A, \Delta \quad \Gamma \longrightarrow A, \Box A, \Delta}{\Gamma \longrightarrow \Box A, \Delta} \qquad (\rightarrow: sfcT)$$

$$\frac{\Gamma, \Box\Gamma, \Box A \longrightarrow A}{\Box\Gamma \longrightarrow \Box A}\ (\rightarrow \Box : G) \qquad \frac{\Gamma, \Box\Gamma, \Box(A \Rightarrow \Box A) \longrightarrow A}{\Box\Gamma \longrightarrow \Box A}(\rightarrow \Box : Grz)$$

Figure 5.2: Rautenberg's Sequent Rules

$\mathcal{C}L$	Static Rules	Transitional Rules	Structural Rules
$\mathcal{C}PC$	$(0), (\neg), (\wedge), (\vee)$	–	(θ)
$\mathcal{C}K$	$(0), (\neg), (\wedge), (\vee)$	(K)	(θ)
$\mathcal{C}T$	$(0), (\neg), (\wedge), (\vee), (T)$	(K)	(θ)
$\mathcal{C}D$	$(0), (\neg), (\wedge), (\vee), (D)$	(K)	(θ)
$\mathcal{C}K4$	$(0), (\neg), (\wedge), (\vee)$	$(K4)$	(θ)
$\mathcal{C}D4$	$(0), (\neg), (\wedge), (\vee), (D)$	$(K4)$	(θ)
$\mathcal{C}S4$	$(0), (\neg), (\wedge), (\vee), (T)$	$(S4)$	(θ)
$\mathcal{C}B$	$(0), (\neg), (\wedge), (\vee), (T), (B), (sfcT)$	(K)	(θ)
$\mathcal{C}S5$	$(0), (\neg), (\wedge), (\vee), (T), (5), (sfcT)$	$(S4)$	(θ)
$\mathcal{C}G$	$(0), (\neg), (\wedge), (\vee)$	(G)	(θ)
$\mathcal{C}Grz$	$(0), (\neg), (\wedge), (\vee), (T)$	(Grz)	(θ)

Figure 5.3: Rautenberg's Tableau Calculi

A set X is $\mathcal{C}L$-**saturated** if it is $\mathcal{C}L$-consistent and closed with respect to the static rules of $\mathcal{C}L$. For example, a set X is $\mathcal{C}S5$-saturated if it is $\mathcal{C}S5$-consistent and saturated with respect to the rules $(0), (\neg), (\wedge), (\vee), (T), (5)$ and $(sfcT)$.

Let $X_K^* = X_T^* = X_{K4}^* = X_{S4}^* = X_G^* = \widetilde{X} = Sf \neg Sf\, X; \emptyset$.

Let $X_D^* = X_B^* = X_{S5}^* = Sf \neg Sf \,\Box \widetilde{X}$.

Let $X_{Grz}^* = Sf\, \Box(\widetilde{X} \Rightarrow \Box\widetilde{X})$ where $\Box(\widetilde{X} \Rightarrow \Box\widetilde{X})$ is $\{\Box(P \Rightarrow \Box P) \mid P \in \widetilde{X}\}$.

Lemma 5 *If there is a closed $\mathcal{C}L$-tableau for X then there is a closed $\mathcal{C}L$-tableau for X with all nodes in the finite set X_L^*.*

Proof: Obvious from the fact that all rules for $\mathcal{C}L$ operate with subsets of X_L^* only. •

Lemma 6 *For each $\mathcal{C}L$-consistent X there is an effective procedure to construct some finite $\mathcal{C}L$-saturated X^* with $X \subseteq X^* \subseteq X_L^*$, where* **L** $\in$ { **K, T, D, K4, D4, S4, B, S5, G, Grz** }.

Proof Outline: Since X is $\mathcal{C}L$-consistent, no $\mathcal{C}L$-tableau for X closes. Thus we can obtain a sequence of finite sets $X = X_0, X_1, \cdots$ where each X_i is obtained from X_{i-1} by an application of one of the static rules of $\mathcal{C}L$. Each X_i is $\mathcal{C}L$-consistent and if this sequence terminates with some X_n then we can form $X^* = X_0 \cup X_1 \cup \cdots \cup X_n$. Thus the crux of the proof is to show that this sequence terminates. Since the tableau rules carry a subset of X_L^* to another subset of X_L^*, there are only a finite number of possible values for the X_i so the only way this sequence may not terminate is if it goes into a cycle. As for $\mathcal{C}S4$ we prove that the static rules of $\mathcal{C}L$ cannot lead to a cycle. Recall that the proof of this for $\mathcal{C}S4$ used the fact that each rule either increases the number of formulae or reduces the maximum degree and that (0) cannot be used at any time since each X_i is $\mathcal{C}L$-consistent by supposition.

Proof for $\mathcal{C}K$: The argument is as for $\mathcal{C}S4$ but omitting the argument for (T).

Proof for $\mathcal{C}T$: The (T) rule cannot lead to a cycle since it increases the number of formula and no static rule of $\mathcal{C}T$ decreases the number of formulae.

Proof for $\mathcal{C}D$: The (D) rule cannot lead to a cycle since it increases the number of formula and no static rule of $\mathcal{C}D$ decreases the number of formulae.

Proof for $\mathcal{C}K4$: The static rules for $\mathcal{C}K4$ are identical to those for $\mathcal{C}K$ so the proof for $\mathcal{C}K$ suffices.

Proof for $\mathcal{C}D4$: The static rules for $\mathcal{C}D4$ are identical to those for $\mathcal{C}D$ so the proof for $\mathcal{C}D$ suffices.

Proof for $\mathcal{C}S4$: See page 55.

Proof for $\mathcal{C}B$: The (B) and $(sfcT)$ rules cannot lead to a cycle since they increase the number of formulae and no static rule in $\mathcal{C}B$ decreases the number of formulae.

Proof for $\mathcal{C}S5$: The (5) and $(sfcT)$ rules cannot lead to a cycle since they increase the number of formulae and no static rule in $\mathcal{C}S5$ decreases the number of formulae.

Proof for $\mathcal{C}G$: As for $\mathcal{C}S4$, except omit the argument for (T).

Proof for $\mathcal{C}Grz$: As for $\mathcal{C}S4$. •

5.1.1 Soundness of $\mathcal{C}L$

Theorem 8 *The $\mathcal{C}L$ rules are sound with respect to* **L***-frames.*

Proof Outline : For each rule in $\mathcal{C}L$ we have to show that if the numerator of the rule is **L**-satisfiable then so is at least one of the denominators.

The proofs for $(0), (\neg), (\theta), (\wedge)$ and $(\vee)$ are as for $\mathcal{C}S4$ since the proofs do not involve R. The proofs for the rules (sfc) and $(sfcT)$ are similar to those for $(\vee)$ by the fact that a world in any model either satisfies P or satisfies $\neg P$ for any formula P.

Some of the proofs below use general properties of the respective **L**-frames like reflexivity or transitivity. Others rely on the structural aspects of the respective **L**-frames like lack of proper clusters or finiteness. Most of the proofs are really just technical versions of the intuitions given on page 68 but for some proofs, finiteness is essential.

We often use annotated names like w_1 and w' to denote possible worlds. Unless stated explicitly, there is no reason why w_1 and w' cannot name the same world.

Proof for (K)**:** We show that (K) is sound with respect to all our **L**-frames. Suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **L**-model, $w_0 \in W$ and $w_0 \models \Box X; \neg\Box P$. This implies that there exists a $w_1 \in W$ with $w_0 R w_1$ and $w_1 \models \neg P$. Since $w_0 \models \Box X$ and $w_0 R w_1$, the definition of $\models$ implies that $w_1 \models X$, hence $w_1 \models X; \neg P$. That is, the (K) rule is sound in all of our systems even though it is not necessary for some of them.

Proof for (T)**:** In each $\mathcal{CL}$ that contains the (T) rule, the corresponding **L**-frames are reflexive. Thus if $\mathcal{M} = \langle W, R, V \rangle$ is an **L**-model where R is reflexive and $w_0 \in W$ and $w_0 \models \Box X; \Box P$, then $w_0 \models \Box X; \Box P; P$ by definition of $\models$ and the reflexivity of R.

Proof for (D)**:** In each $\mathcal{CL}$ that contains the (D) rule, the corresponding **L**-frames are serial. So suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **L**-model where R is serial. That is, $\forall w \in W, \exists w' \in W : wRw'$. Suppose $w_0 \in W$ and $w_0 \models X; \Box P$. We show that $w_0 \models \neg\Box\neg P$. Assume to the contrary that $w_0 \not\models \neg\Box\neg P$, then $w_0 \models \Box\neg P$. By seriality there exists some $w_1 \in W$ with $w_0 R w_1$. But $w_0 \models \Box P$ and $w_0 \models \Box\neg P$ which together imply that $w_1 \models P$ and $w_1 \models \neg P$ giving a contradiction, so $w_0 \models \neg\Box\neg P$. Hence $w_0 \models X; \Box P; \neg\Box\neg P$ and we are done.

Proof for $(K4)$**:** In each $\mathcal{CL}$ that contains the $(K4)$ rule, the corresponding **L**-frames are transitive. So suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **L**-model where R is transitive. Suppose $w_0 \in W$ and $w_0 \models \Box X; \neg\Box P$. Thus there exists $w_1 \in W$ with $w_0 R w_1$ and $w_1 \models \neg P$. Since R is transitive the definition of $\models$ gives $w_1 \models X; \Box X; \neg P$ and we are done.

Proof for $(S4)$**:** See page 58.

Proof for (B)**:** Suppose $\mathcal{M} = \langle W, R, V \rangle$ is a **B**-model, $w_0 \in W$ and $w_0 \models \Box X; \neg\Box P$. We show that $w_0 \models P$ or $w_0 \models \neg P; \Box\neg\Box P$. If $w_0 \models P$ then $w_0 \models \Box X; \neg\Box P; P$ and we are done. Otherwise $w_0 \models \neg P$. In this latter case, suppose $w_0 \not\models \Box\neg\Box P$. Then $w_0 \models \neg\Box\neg\Box P$, that is $w_0 \models \Diamond\Box P$, so there exists some $w_1 \in W$ with $w_0 R w_1$ and $w_1 \models \Box P$. Since R is symmetric, $w_0 R w_1$ implies $w_1 R w_0$ which together with $w_1 \models \Box P$ gives $w_0 \models P$. But this contradicts the supposition that $w_0 \models \neg P$. Hence $w_0 \models \Box X; \neg\Box P; P$ or $w_0 \models \Box X; \neg\Box P; \neg P; \Box\neg\Box P$ and we are done.

Proof for (5)**:** Suppose $\mathcal{M} = \langle W, R, V \rangle$ is an **S5**-model, then R is reflexive, transitive and symmetric. Suppose $w_0 \in W$ and $w_0 \models X; \neg\Box P$. We have to show that $w_0 \models \Box\neg\Box P$. Assume for a contradiction that $w_0 \not\models \Box\neg\Box P$. Then $w_0 \models \neg\Box\neg\Box P$ and hence $w_0 \models \Diamond\Box P$. Thus there exists some $w_1 \in W$ with $w_0 R w_1$ and $w_1 \models \Box P$. But R is symmetric, so $w_0 R w_1$ implies $w_1 R w_0$. Since R is also transitive, we must have $w_0 \models \Box P$ contradicting the supposition that $w_0 \models \neg\Box P$. Hence $w_0 \models \Box X; \neg\Box P; \Box\neg\Box P$ and we are done.

Proof for (G): Suppose $\mathcal{M} = \langle W, R, V \rangle$ is a **G**-model, $w_0 \in W$ and $w_0 \models \Box X; \neg\Box P$. Thus there exists some $w_1 \in W$ with $w_0 R w_1$ and $w_1 \models X; \Box X; \neg P$ by the transitivity of R. Since R is irreflexive, $w_0 \neq w_1$. Suppose $w_1 \not\models \Box P$. Then $w_1 \models \neg\Box P$ and there exists some $w_2 \in W$ with $w_1 R w_2$ and $w_2 \models X; \Box X; \neg P$ by transitivity of R. Since R is irreflexive, $w_1 \neq w_2$. Since R is transitive, $w_0 = w_2$ would give $w_1 R w_0 R w_1$ implying $w_1 R w_1$ and contradicting the irreflexivity of R, hence $w_0 \neq w_2$. Suppose $w_2 \not\models \Box P$ then ... Continuing in this way, it is possible to obtain an infinite path of distinct worlds in $\mathcal{M}$ contradicting the finiteness of $\mathcal{M}$. Thus there must exist some $w_i \in W$ with $w_0 R w_i$ and $w_i \models X; \Box X; \neg P; \Box P$ and we are done.

Proof for (Grz) : Suppose $\mathcal{M} = \langle W, R, V \rangle$ is a **Grz**-model, then R is reflexive and transitive. Suppose $w_0 \in W$ is such that $w_0 \models \Box X; \neg\Box P$. We have to show that there exists some $w_n \in W$ with $w_0 R w_n$ and $w_n \models X; \Box X; \neg P; \Box(P \Rightarrow \Box P)$. Since R is reflexive and transitive, $w_0 \models \Box X$ means that $\forall w \in W, w_0 R w$ implies $w \models X; \Box X$. Thus our task is reduced to showing that there exists some $w_n \in W$ such that $w_0 R w_n$ and $w_n \models \neg P; \Box(P \Rightarrow \Box P)$. Suppose for a contradiction that no such world exists in W. That is,

$$(a) \quad \forall w \in W, w_0 R w \text{ implies } w \not\models \neg P; \Box(P \Rightarrow \Box P).$$

Since $w_0 \models \neg\Box P$, there exists some $w_1 \in W$ with $w_0 R w_1$ and $w_1 \models \neg P$. By (a), $w_1 \not\models \Box(P \Rightarrow \Box P)$ and hence $w_1 \models \neg\Box(P \Rightarrow \Box P)$. Thus there exists some $w_2 \in W$ with $w_1 R w_2$ and $w_2 \models \neg(P \Rightarrow \Box P)$, that is, $w_2 \models P \wedge \neg\Box P$. Since $w_1 \models \neg P$, $w_1 \neq w_2$ and since **Grz**-models cannot contain proper clusters, $w_0 \neq w_2$. Since $w_2 \models \neg\Box P$ there exists some $w_3 \in W$ with $w_3 \models \neg P$. By (a), ... Continuing in this way, we obtain an infinite path, contradicting the finiteness of $\mathcal{M}$, or a cycle, contradicting the absence of proper clusters in **Grz**-frames. Hence (a) cannot hold and $\exists w \in W, w_0 R w$ and $w \models \neg P; \Box(P \Rightarrow \Box P)$. That is, the desired w_n exists. •

5.1.2 Completeness of $\mathcal{CL}$

Some of the completeness proofs make extensive use of the following property. A set X is **subformula-complete** if $P \in Sf(X)$ implies either $P \in X$ or $\neg P \in X$.

Lemma 7 *If X is closed with respect to* $(0), (\neg), (\wedge), (\vee)$ *and* (sfc) *then X is subformula-complete.*

Proof: Obvious. •

The $(sfcT)$ rule is just a special case of (sfc) and always appears with (T). Thus, the lemma also holds if we have both $(sfcT)$ and (T) instead of (sfc).

Theorem 9 *If X is a finite set of formulae and X is $\mathcal{C}L$-consistent then there is an* $\mathbf{L}$*-model for X on a finite* $\mathbf{L}$*-frame where* $\mathbf{L} \in \{\mathbf{K}, \mathbf{T}, \mathbf{D}, \mathbf{K4}, \mathbf{D4}, \mathbf{S4}, \mathbf{B}, \mathbf{S5}, \mathbf{G}, \mathbf{Grz}\}$ *[Rau83].*

Proof Outline: For each $\mathcal{C}L$ we give a way to construct a finite model graph $\langle W_0, R\rangle$ for X. Recall that a model graph for some finite fixed set of formulae X is a finite $\mathbf{L}$-frame $\langle W_0, R\rangle$ such that all $w \in W_0$ are $\mathcal{C}L$-saturated sets with $w \subseteq X_L^*$ and

(i) $X \subseteq w_0$ for some $w_0 \in W_0$;

(ii) if $\neg\Box P \in w$ then there exists some $w' \in W_0$ with wRw' and $\neg P \in w'$;

(iii) if wRw' and $\Box P \in w$ then $P \in w'$.

By Lemma 4 (page 57), $w_0 \models X$ under the truth valuation $\vartheta : p \mapsto \{w \in W_0 \mid p \in w\}$.

The first step is to create a $\mathcal{C}L$-saturated w_0 with $X \subseteq w_0 \subseteq X_L^*$. This is possible via Lemma 6 (page 72). So w_0, and in general w (possibly annotated) stands for a finite $\mathcal{C}L$-saturated set of formulae (that corresponds to a world of W_0). Since X is $\mathcal{C}L$-consistent, we know that *no* $\mathcal{C}L$-tableau for X closes. We use this fact to construct a graph of $\mathcal{C}L$-saturated worlds, always bearing in mind that the resulting model graph must be based on an $\mathbf{L}$-frame. We use a successor relation $\prec$ while building this graph and then form R from $\prec$.

A formula $\neg\Box P$ is called an **eventuality** since it entails that eventually $\neg P$ must hold. A set w is said to fulfill an eventuality $\neg\Box P$ when $\neg P \in w$. A sequence $w_1 \prec w_2 \prec \cdots \prec w_m$ of sets is said to fulfill an eventuality $\neg\Box P$ when $\neg P \in w_i$ for some w_i in the sequence.

Proof for $\mathbf{L} = \mathbf{K}$: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \emptyset\rangle$ is the desired model graph since this is a $\mathbf{K}$-frame and (i)-(iii) are satisfied. Otherwise, let $Q_1, Q_2, \cdots Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$. Since w_0 is $\mathcal{C}K$-consistent, so are each of $w_0^{\Box} \cup \neg Q_i$ for $i = 1, \cdots, m$ by (θ) and (K). Create a $\mathcal{C}K$-saturated $v_i \subseteq X_K^*$ from each $w_0^{\Box} \cup \neg Q_i$ for $i = 1, \cdots m$, giving the nodes of level 1. Continue to create the nodes of further levels using (θ) and (K). Note that the $\mathcal{C}K$-saturation process does not increase the maximum modal degree. Hence a path $w_0 \prec w_1 \prec w_2 \cdots$ must terminate (without cycles) because the Q_i-successor created by (K) has a maximum modal degree strictly lower than that of the parent node. Let R be $\prec$ and let W_0 consist of all the nodes created in this process, then $\langle W_0, R\rangle$ is a finite, irreflexive and intransitive tree and a model graph for X. Hence there is a $\mathbf{K}$-model for X at w_0.

Proof for $\mathbf{L} = \mathbf{T}$: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph since (i)-(iii) are satisfied. Otherwise, let $Q_1, Q_2, \cdots Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$ and $\neg Q_i \notin w_0$. Proceed as for $\mathbf{L} = \mathbf{K}$, except for this minor change of ignoring $\neg\Box Q \in w$ if $\neg Q \in w$, and let R be the reflexive closure of $\prec$.

Proof for L = D: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph since (i)-(iii) are satisfied. Otherwise, proceed as for **L** = **K** and let W_{end} be the nodes of W_0 which have no successors. For each $w, w' \in W_0$, put wRw' if $w \prec w'$ and put wRw if $w \in W_{end}$. We have to show that (i)-(iii) are satisfied by this R. The only interesting case is to show that $\Box P \in w$ implies $P \in w$ for $w \in W_{end}$. This is true since $w \in W_{end}$ implies that w contains no $\Box P$, as otherwise, w would contain $\neg\Box\neg P$ by (D) and hence would have a successor node by (K), contradicting that $w \in W_{end}$.

Proof for L = G: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \emptyset\rangle$ is the desired model graph as (i)-(iii) are satisfied. Otherwise, let $Q_1, Q_2, \cdots Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$. Create a $\mathcal{C}G$-saturated Q_i-successor for each Q_i using (θ) and (G) giving the nodes v_i of level one, and so on for other levels. Consider any sequence $w_i \prec w_{i+1} \prec w_{i+2} \cdots$. Since w_i has a successor, there is some $\neg\Box Q \in w_i$ and $\Box Q \in w_{i+j}$ for all $j \geq 1$ by (G). Thus $w_i \neq w_{i+j}$ for any $j \geq 1$ and each such sequence must terminate since X_G^* is finite. Let R be the transitive closure of $\prec$ to obtain a model graph $\langle W_0, R\rangle$ for X which is also a **G**-frame.

Proof for L = Grz: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph as (i)-(iii) are satisfied. Otherwise, let $Q_1, Q_2, \cdots Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$ and $\neg Q_i \notin w_0$. Create a $\mathcal{C}Grz$-saturated Q_i-successor for each Q_i using (θ) and (Grz) giving the nodes v_i of level one, and so on for other levels. Consider any sequence $w_i \prec w_{i+1} \prec w_{i+2} \cdots$. Since w_i has a successor, there is some $\neg\Box Q \in w_i$, $\neg Q \notin w_i$, and by (Grz), $\Box(Q \Rightarrow \Box Q) \in w_{i+j}$ for all $j \geq 1$. Suppose $w_{i+j} = w_i$, then $\Box(Q \Rightarrow \Box Q) \in w_i$ and hence $Q \Rightarrow \Box Q \in w_i$ by (T). Since $Q \Rightarrow \Box Q$ is just abbreviation for $\neg(Q \wedge \neg\Box Q)$, we know that $\neg Q \in w_i$ or $\neg\neg\Box Q \in w_i$. We created a successor w_{i+1} for w_i precisely because $\neg Q \notin w_i$ and so the first case is impossible. And if $\neg\neg\Box Q \in w_i$ then $\Box Q \in w_i$ by $(\neg)$, contradicting the **Grz**-consistency of w_i since $\neg\Box Q \in w_i$ by supposition. Thus each such sequence must terminate (without cycles). Let R be the reflexive and transitive closure of $\prec$ to obtain a model graph $\langle W_0, R\rangle$ for X which is also a **Grz**-frame.

Proof for L = K4: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \emptyset\rangle$ is the desired model graph. Otherwise proceed as for $\mathcal{C}S4$ except create a successor for *every* eventuality $\neg\Box P \in w$ and use $(K4)$ to create successors instead of $(S4)$. That is, let $Q_1, \cdots, Q_m$ be all the formulae such that $\neg\Box Q_i \in w$. In $\mathcal{C}S4$, we do not count $\neg\Box P$ if $\neg P \in w$. Let R be the transitive closure of $\prec$ only (instead of the reflexive and transitive closure of $\prec$).

Proof for L = D4: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph. Otherwise, proceed as for **L** = **K4**. A sequence either terminates or cycles since X_{D4}^* is finite. Put $w \prec w$ for all $w \in W_{end}$ and let R be the transitive closure of $\prec$. Property (iii) is satisfied by $w \in W_{end}$ just as in the proof for **L** = **D**.

Proof for L = S4: See page 60.

Proof for L = B: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph as (i)-(iii) are satisfied. Otherwise, let $Q_1, Q_2, \cdots Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$ and $\neg Q_i \notin w_0$. Since w_0 is $\mathcal{C}B$-saturated, w_0 is subformula-complete, hence $Q_i \in w_0$ for each Q_i. Create a Q_i-successor for each Q_i using (θ) and (K) giving the nodes of level one. Repeat this procedure to give the nodes of level two and so on. For any node w in this construction let $s(w)$ be the number of formulae with $P \in w$ and $\neg\Box P \in w$. Let $t(w) = s(w) + mdegmax(w)$. To quote Rautenberg *"It is easily seen that* $w \prec v \Rightarrow t(v) < t(w)$*, so that* W_0 *is finite."*, but as we shall see this is not quite correct. We accept Rautenberg's claim for the moment and return to this issue at the end of this section.

By construction, $\langle W_0, R\rangle$ is a **B**-frame. We have to show that (i)-(iii) hold. The only difficulty is to show symmetry: that is, $\Box P \in w_{i+1}$ and $w_i \prec w_{i+1}$ implies $P \in w_i$. So suppose that $w_i \prec w_{i+1}$ and $\Box P \in w_{i+1}$. We have to show that $P \in w_i$. There are two cases: $\Box P \in Sf(w_i)$ or $\Box P \notin Sf(w_i)$.

Case 1: If $\Box P \in Sf(w_i)$, then $\Box P \in w_i$ or $\neg\Box P \in w_i$ since w_i is subformula-complete. If $\Box P \in w_i$ then $P \in w_i$ by (T) and we are done. Otherwise, if $\neg\Box P \in w_i$ and $P \notin w_i$ then $\neg P \in w_i$ since w_i is subformula-complete. But $\neg P \in w_i$ and $\neg\Box P \in w_i$ implies $\Box\neg\Box P \in w_i$ by (B) and so $\neg\Box P \in w_{i+1}$ contradicting the consistency of w_{i+1} since $\Box P \in w_{i+1}$ by supposition. Hence $\neg\Box P \in w_i$ also implies that $P \in w_i$.

Case 2: If $\Box P \notin Sf(w_i)$ then $\Box P = \Box\neg\Box Q$ for some $\neg\Box Q \in w_{i+1}$ and $\neg Q \in w_{i+1}$. Hence $\neg\Box Q \in Sf(w_i)$ and so $\Box Q \in w_i$ or $\neg\Box Q \in w_i$. If $\Box Q \in w_i$, then $Q \in w_{i+1}$ contradicting the $\mathcal{C}B$-consistency of w_{i+1} since $\neg Q \in w_{i+1}$. Hence $\neg\Box Q \in w_i$. But then $P \in w_i$ since P is $\neg\Box Q$ and we are done.

Proof for L = S5: If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph as (i)-(iii) are satisfied. Otherwise, let $Q_1, Q_2, \cdots Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$. Since w_0 is $\mathcal{C}S5$-saturated, $\Box\neg\Box Q_i \in w_0$ for each Q_i by (5). Create a Q_i-successor for each Q_i using (θ) and $(S4)$ giving the nodes v_i of level one, put $w_0 \prec v_i$, for each $i = 1, 2, \cdots, m$ and stop! Let R be the reflexive, transitive and symmetric closure of $\prec$. By construction, $\langle W_0, R\rangle$ is an **S5**-frame. We have to show that (i)-(iii) hold.

For any k, with $1 \leq k \leq m$, and $w_0 \prec v_k$, we show that :

(a) $\neg\Box P \in v_k$ implies $\neg\Box P \in w_0$

and

(b) $\Box P \in v_k$ implies $\Box P \in w_0$

from which (i)-(iii) follow.

(a) Suppose $w_0 \prec v_k$, $\neg\Box P \in v_k$ and $\neg\Box P \notin w_0$. Since $\neg\Box P \in Sf(w_0)$, and w_0 is subformula-complete, we have $\Box P \in w_0$. But then, by $(S4)$, $\Box P \in v_k$, contradicting the $\mathcal{C}S5$-consistency of v_k. Hence $\neg\Box P \in w_0$.

(b) Suppose $w_0 \prec v_k$ and $\Box P \in v_k$, then $\Box P \in Sf(w_0)$ or $\Box P \notin Sf(w_0)$.

(b1) If $\Box P \in Sf(w_0)$ and $\Box P \notin w_0$, then $\neg\Box P \in w_0$ since w_0 is subformula-complete. Then $\Box\neg\Box P \in w_0$ by (5) and $\neg\Box P \in v_k$ by $(S4)$, contradicting the $\mathcal{C}S5$-consistency of v_k. Hence, if $\Box P \in v_k$ and $\Box P \in Sf(w_0)$ then $\Box P \in w_0$.

(b2) If $\Box P \notin Sf(w_0)$ then $\Box P = \Box\neg\Box Q$ for some $\neg\Box Q \in v_k$ since this is the only way that formulae from outside $Sf(w_0)$ can appear in v_k. By (a), $\neg\Box Q \in v_k$ implies $\neg\Box Q \in w_0$ which by (5) implies $\Box\neg\Box Q \in w_0$. Since $\Box\neg\Box Q$ *is* $\Box P$, we have $\Box P \in w_0$. But this is absurd since it implies that $\Box P \in Sf(w_0)$ and our supposition was that $\Box P \notin Sf(w_0)$. Hence the subcase (b2) cannot occur. •

Aside on Rautenberg's proof for B: Rautenberg's completeness proof for $\mathcal{C}B$ requires that a sequence $w_0 \prec w_1 \prec \cdots$ in the model graph construction must terminate. That is, we seek a tree of worlds where each world is reflexive and each arc is symmetric. Rautenberg [Rau83, page 413] claims the following:

> *Let $s(w)$ be the number of formulas $P, \neg\Box P \in w$, $t(w) = s(w) + deg\ w$. It is easily seen that $w \prec v \Rightarrow t(v) < t(w)$, so that W_0 is finite.*

Rautenberg's definition of *deg w* is our definition of maximal modal degree $mdegmax(w) = max\{mdeg(P) \mid P \in w\}$. But the claim is not correct as the following counter-example shows that his metrics do not guarantee that $t(v) < t(w)$ whenever $w \prec v$.

Let

$$w = \{p, \neg\Box p, \neg q, \neg\Box q, \Box\neg\Box q, \Box(p \vee q), p \vee q, \Box\neg\Box\neg p, \neg\Box\neg p, \neg\neg p\};$$

$$v = \{\neg p, \neg\Box q, p \vee q, q, \neg\Box\neg p\}$$

where v is a p-successor to w by (K). Then the set w is $\mathcal{C}B$-consistent as v gives two children containing $\{\neg\neg p, p\}$ and $\{\neg q\}$, respectively, to give a **B**-model for w. But we now have

$$s(w) = 1 \text{ since } \{p, \neg\Box p\} \subseteq w;$$

$$mdegmax(w) = 2\ ;$$

$$t(w) = s(w) + mdegmax(w) = 3\ ;$$

$$s(v) = 2 \text{ since } \{\neg p, \neg\Box\neg p, q, \neg\Box q\} \subseteq v;$$

$$mdegmax(v) = 1\ ;$$

$$t(v) = s(v) + mdegmax(v) = 3\ ;$$

Hence $t(w) = t(v)$ and the metric t is not sufficient.

I have been unable to find a suitable metric but I feel that the result still holds because Hughes and Cresswell [HC84, page 122] note that "**B** *is characterised by the class of the reflexive symmetrical extensions of all tree frames*". Their definition of a tree (frame) is the same as ours [HC84, page 118].

5.2 Fitting's Systems

Fitting's tableau systems for **K**,**T**, **K4**, and **S4** are identical to those presented by Rautenberg since these systems are well known. Rautenberg's system for **G** is due to Fitting [Fit73]. The differences between the two authors' work become apparent in the way they treat **D** and **S5**.

5.2.1 Fitting's System for D

An alternative cut-free sequent system for **D** is possible using a transitional rule to encode the seriality of **D**-frames as is done by Shvarts [Shv89] and originally by Fitting [Fit83]. Let $\mathcal{C}D'$ be the tableau calculus $\mathcal{C}D$ minus the *static* (D) rule but plus the additional *transitional* rule:

$$(D')\ \frac{\Box X}{X}$$

That is,

$$\mathcal{C}D' = \{(0), (\neg), (\wedge), (\vee)\} \cup \{(K), (D')\} \cup \{(\theta)\}$$

where the three sets respectively represent the static, transitional and structural rules of $\mathcal{C}D'$. The (D') rule is just really the (K) rule where the eventuality $\neg\Box P$ is absent. But by seriality of R we know that a formula like $\Box P$ entails the existence of a world where P become true. That is, $\Box P$ cannot be true vacuously in a serial R.

Let $X^*_{D'} = \widetilde{X}$. Thus $\mathcal{C}D'$ has the subformula property whereas $\mathcal{C}D$ does not.

A finite set X is $\mathcal{C}D'$-saturated if it is $\mathcal{C}D'$-consistent and saturated with respect to (0), $(\wedge)$, $(\vee)$ and $(\neg)$ as these are the static rules for $\mathcal{C}D'$. That is, as (D') is a transitional rule, it does not contribute to $\mathcal{C}D'$-saturatedness. It is easy to show that (D') is sound with respect to **D**-frames using the seriality of R in **D**-frames and the fact that $\Box P \Rightarrow \Diamond P$ is an axiom of **D**.

For completeness, suppose X is $\mathcal{C}D'$-consistent. Since the static rules of $\mathcal{C}D$ are those of $\mathcal{C}PC$, it is possible to create a $\mathcal{C}D'$-saturated w_0 with $X \subseteq w_0 \subseteq X^*_{D'}$. Do so and proceed as follows to obtain a finite intransitive tree with irreflexive non-leaf nodes and reflexive leaf nodes.

If no $\neg\Box Q$ occurs in w_0 and no $\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph. Otherwise, let $Q_1, Q_2, \cdots, Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$ and let $Y = \{P_1, P_2, \cdots, P_n\}$ be all the formulae such that $\Box P_j \in w_0$. Since w_0 is $\mathcal{C}D'$-consistent each $\neg Q_i; Y$ is $\mathcal{C}D'$-consistent, for $i = 1, 2, \cdots, m$ by (θ) and (K). Also, Y itself is $\mathcal{C}D'$-consistent by (θ) and (D'). If $m \neq 0$ then create a Q_i-successor v_i using (K) containing $\neg Q_i; Y$ for each Q_i. But if $m = 0$ then create a single P-successor y using (D') containing Y. Put $w_0 \prec v_i$ for each $i = 1 \cdots m$ or $w_0 \prec y$ as the case may be, giving the node(s) of level one. Continuing in this way obtain the node(s) of level two etc. A sequence $w_0 \prec w_1 \prec w_2 \cdots$ must terminate since both (K) and (D') reduce the maximum modal degree and $\mathcal{C}D'$-saturation does not increase it. As in the first proof for

$\mathcal{C}D$ put wRw if $w \in W_{end}$ and put wRw' if $w \prec w'$. Property (iii) holds for $w \in W_{end}$ as end nodes do not contain any $\Box P$, as otherwise, w would have a successor by (D'), contradicting that $w \in W_{end}$. •

The advantage of $\mathcal{C}D'$ is that this tableau system and the corresponding sequent system is not only cut-free, but it also has the subformula property. As usual, we lose invertibility so that we must keep track of cycles. The task of the static (D) rule in $\mathcal{C}D$ is to ensure seriality by adding an eventuality $\Diamond P$ for every formula of the form $\Box P$. The transitional (K) rule then fulfills that eventuality. In $\mathcal{C}D'$ seriality is guaranteed by the transitional (D') rule.

Instead of adding an explicit rule like (D'), we could make do with a version of (K) called (KD'), say, where the numerator of (KD') need not contain $\neg\Box P$, and when this happens, the denominator of (KD') does not contain $\neg P$.

5.2.2 Fitting's Systems for S5

Fitting [Fit83] presents a system for **S5** which requires a "semi-analytic" cut rule for completeness. The system is essentially $\mathcal{C}S4 + (S5) + (cut)$ where the $(S5)$ rule is:

$$(S5)\ \frac{\Box X; \neg\Box Y; \neg\Box P}{\Box X; \neg\Box Y; \neg P}$$

and the (cut) rule is only allowed to cut on subformulae and superformulae of formulae that are in the numerator. Since the superformulae are not bounded, as they are in Rautenberg's systems, the semi-analytic (cut) rule cannot give a decision procedure. We therefore defer discussing this system in detail until Chapter 7.

Fitting [Fit83, page 226] replaces the semi-analytic (cut) rule with a building up rule of the form

$$(\pi)\ \frac{X; P}{X; \Diamond P; P}$$

and proves that his system $\mathcal{C}S5\pi = \mathcal{C}S4 + (S5) + (\pi)$ is sound and weakly complete with respect to **S5**-frames. But notice that the (π) rule is not "once off" since it can lead to an infinite chain $A \in w, \Diamond A \in w, \Diamond\Diamond A \in w, \cdots$ so this system cannot give a decision procedure for **S5** either. That is, we have merely traded one non-analytic rule for another.

Fitting then proves the curious fact that a formula A is an **S5**-theorem if and only if a $\mathcal{C}S5\pi$-tableau for $\{\neg A\}$ closes, and furthermore, that the (π) rule needs to be used only *once* at the beginning of the $\mathcal{C}S5\pi$-tableau [Fit83, page 229]. That is, the system $\mathcal{C}S5\pi$ *without* the (π) rule is (weakly) complete for **S5** in the sense that A is an **S5**-theorem if and only if a $\mathcal{C}S5\pi$-tableau for $\{\neg\Box A\}$ closes. We shall return to this point later when discussing Shvarts' embedding of **S5** into **K45**.

5.2.3 An Embedding of S5 into S4

Fitting [Fit83, pages 223-225] presents a proof of an embedding of **S5** into **S4** originally proved by Matsumoto [Mat55]. Here we use the characterisation results with respect to *finite* frames to obtain a simplified semantic proof of the same result. A syntactic proof of the following theorem is given by Zeman [Zem73, page 254].

Theorem 10 *A formula A is an* **S5***-theorem if and only if the formula $\Diamond\Box A$ is an* **S4***-theorem.*

Proof Outline: The result follows from the observation that each leaf of an **S4**-frame is a nondegenerate cluster, and hence an **S5**-frame in its own right. Thus, if $\Diamond\Box A$ is an **S4**-theorem then $\Box A$ will be valid over each leaf of each **S4**-frame. That is, over each **S5**-frame.

Proof: We make use of the following points:

1. **S5** is characterised by finite, reflexive, transitive and symmetric graphs;
2. **S4** is characterised by finite, reflexive and transitive trees of nondegenerate clusters;
3. **S4** $\subseteq$ **S5**; that is, every **S4**-theorem is an **S5**-theorem;
4. we write $\vdash_L A$ to mean that A is an **L**-theorem and write $\not\vdash_L A$ to mean that A is not an **L**-theorem.

Proof $\Leftarrow$: We have to show that $\vdash_{S4} \Diamond\Box A$ implies $\vdash_{S5} A$. Suppose $\vdash_{S4} \Diamond\Box A$. Then $\vdash_{S5} \Diamond\Box A$ by point 3 above. We already know that $\vdash_{S5} \Diamond\Box A \Rightarrow \Box A$ since this is just the contrapositive of axiom 5. We also know that $\vdash_{S5} \Box A \Rightarrow A$, hence $\vdash_{S5} A$ by modus ponens.

Proof $\Rightarrow$: We have to show that $\vdash_{S5} A$ implies $\vdash_{S4} \Diamond\Box A$. Suppose to the contrary that $\vdash_{S5} A$ but $\not\vdash_{S4} \Diamond\Box A$.

This means that there is some finite **S4**-model $\mathcal{M} = \langle W, R, V\rangle$ such that $\mathcal{F} = \langle W, R\rangle$ is a finite, reflexive, transitive tree of nondegenerate clusters and that there exists $w \in W$ with $w \not\models_V \Diamond\Box A$. That is, $w \models_V \Box\Diamond\neg A$.

Consider the subtree of w in $\mathcal{F}$ (there may be none since w may be a leaf). Consider any leaf cluster C of this subtree ($C = w$ in the case that w is a leaf). The leaf C is either a simple cluster or a proper cluster.

Case 1: If C is a simple cluster then $C = w'$ and we know that $w' \models_V \Diamond\neg A$. By reflexivity and the fact that w' is a leaf we have $w' \models_V \neg A$. But $\mathcal{M}' = \langle\{w'\}, \{(w', w')\}, V_{w'}\rangle$ where $V_{w'}$ is the restriction of V to w is then an **S5**-model falsifying A, contradicting our supposition that $\vdash_{S5} A$. Hence C cannot be a simple cluster.

Case 2: If C is a proper cluster then $C = c_1 R c_2 R \cdots R c_n R c_1$ where each c_i is a reflexive world and each $c_i \models_V \Diamond\neg A$. Since C is a leaf, one of these worlds c_j must fulfill this eventuality; that is, there exists a c_j such that $c_j \models_V \neg A$. But $\mathcal{M}'' = \langle W_C, R_C, V_C \rangle$, where W_C, R_C and V_C are the restrictions of W, R and V to C, is then an **S5**-model falsifying A, again contradicting our supposition that $\vdash_{S5} A$. Hence C cannot be a proper cluster either.

But this is impossible since one of these cases has to hold. Hence the theorem follows. •

5.3 Shvarts' Systems for K45, K45D and S5

The cut-free tableau system $CK45$ for **K45** and $CK45D$ for **K45D** presented below are essentially the tableau versions of Shvarts' sequent systems for **K45** and **K45D** [Shv89]. Shvarts considers the first order versions of these logics but does not consider other modal logics.

5.3.1 Shvarts' Systems for K45 and K45D

The logic **K45** is often used as a basis for logics of belief where the formula $\Box A$ is read as "A is believed" and the reflexivity axiom, $\Box A \Rightarrow A$, is deliberately omitted on the grounds that believing A should not imply that A is true. The logic **K45D** is another candidate for such logics of belief because its extra axiom, $\Box A \Rightarrow \Diamond A$, which can be written as $\Box A \Rightarrow \neg\Box\neg A$, encodes the intuition that "if A is believed then $\neg A$ is not believed".

The logic **K45** is characterised by finite transitive frames which we call sharp tacks [Seg71, pages 77-78], where a sharp tack is either:

1. a single cluster (degenerate or nondegenerate); or
2. a degenerate cluster followed by a nondegenerate cluster.

The logic **K45D** is characterised by finite transitive frames which we call blunt tacks [Seg71, pages 77-78], where a blunt tack is either:

3. a single nondegenerate cluster; or
4. a degenerate cluster followed by a nondegenerate cluster.

The difference from **K45**-frames is that a single degenerate point is no longer allowed and the intuition behind both names is that frames of type 2 and type 4 look like an inverted tack if the second nondegenerate cluster is visualised as the head of the tack.

Let

$$CK45 = \{(0), (\wedge), (\vee), (\neg)\} \cup \{(45)\} \cup \{(\theta)\}$$

where (45) is:

$$(45)\ \frac{\Box X; \neg\Box Y; \neg\Box P}{X; \Box X; \neg\Box Y; \neg\Box P; \neg P}$$

and the three sets are the static, transitional and structural rules of $CK45$. Note that instead of (5) we build in the effect of (5) by carrying $\neg\Box P$ and $\neg\Box Y$ from the numerator into the denominator.

As for **D** and **D4** we can extend $CK45$ to $CK45D$ in two ways. The first is given below as it is the form used by Shvarts. The second is given at the end of this section.

The first way to extend the system for **K45** to handle **K45D** is to use an obvious transitive version of Fitting's (D') rule called $(45D)$:

$$(45D)\ \frac{\Box X; \neg\Box Y; \neg\Box P}{X; \Box X; \neg\Box Y; \neg\Box P; \neg P}$$

where $\neg\Box Y; \neg\Box P$ in the numerator may be empty, and when this happens, $\neg\Box Y; \neg\Box P; \neg P$ in the denominator is also empty.

Let

$$CK45D = \{(0), (\wedge), (\vee), (\neg)\} \cup \{(45D)\} \cup \{(\theta)\}$$

where the three sets are the static, transitional and structural rules of $CK45D$.

A set is $CK45$-saturated and $CK45D$-saturated if it is closed with respect to $(0), (\wedge), (\vee)$ and $(\neg)$.

Let $X^*_{K45} = X^*_{K45D} = \widetilde{X}$.

It is easy to show that we can obtain a $CK45$-saturated X^* for a given X with $X \subseteq X^* \subseteq X^*_{K45}$ since the static rules of $CK45$ are just the CPC rules. The same proof shows that we can obtain a $CK45D$-saturated X^* for a given X with $X \subseteq X^* \subseteq X^*_{K45D}$.

Theorem 11 *The $CK45$ and $CK45D$ rules are sound with respect to* **K45**-*frames and* **K45D**-*frames respectively.*

Proof : It is easy to show that the rules (0), $(\neg)$, $(\wedge)$, (θ), and $(\vee)$ are sound with respect to **K45**-frames and **K45D**-frames.

Proof for (45) for K45-frames: Let $\mathcal{M} = \langle W, R, V\rangle$ be a **K45**-model and suppose that $w_0 \in W$ and $w_0 \models \Box X; \neg\Box Y; \neg\Box P$. We have to show that there exists a $w' \in W$ such that $w' \models X; \Box X; \neg\Box Y; \neg\Box P; \neg P$.

Clearly, the $X; \Box X$ part will follow from the transitivity of R so we need only prove that there exists a $w' \in W$ such that $w' \models \neg\Box Y; \neg\Box P; \neg P$.

If $\langle W, R\rangle$ is of type 1 then it cannot be a single degenerate cluster since $w_0 \models \neg\Box P$. But it may be a single, proper or simple, (nondegenerate) cluster. So if $\langle W, R\rangle$ is of type 1 then there must be some $w' \in W$ such that $w_0 R w'$ and $w' \models \neg P$. Also, w' must be reflexive since $\langle W, R\rangle$ is a nondegenerate cluster, hence $w' \models \neg\Box P; \neg P$. In a nondegenerate (reflexive, transitive) cluster, R must be symmetric as well, so $w' R w_0$. But in a reflexive, transitive and symmetric cluster, $w_0 \models \neg\Box Y$ implies $w_0 \models \Box\neg\Box Y$, hence $w' \models \neg\Box P; \neg P; \neg\Box Y$ and we are done.

If $\langle W, R\rangle$ is of type 2 then, regardless of whether w_0 is in the first or last cluster, there must be some w' in the last nondegenerate cluster such that $w' \models \neg P$ since $w_0 \models \neg\Box P$. Similarly, if $Y = \{Q_1, Q_2, \cdots, Q_m\}$, then there must exist (not necessarily distinct) worlds $w_1, w_2, \cdots, w_m$ in the last nondegenerate cluster such that $w_i \models \neg Q_i$ for each $Q_i \in Y$. Since R is reflexive, transitive and symmetric over a nondegenerate cluster, this means that $w' \models \neg P; \neg\Box P; \neg\Box Y$.

Proof for (45D) **for K45D-frames:** Let $\mathcal{M} = \langle W, R, V\rangle$ be a **K45D**-model and suppose that $w_0 \in W$ and $w_0 \models \Box X; \neg\Box Y; \neg\Box P$. We have to show that there exists a $w' \in W$ such that $w' \models X; \Box X; \neg\Box Y; \neg\Box P; \neg P$ allowing for the case where the $\neg\Box Y; \neg\Box P$ part is missing. Every **K45D**-frame is a **K45**-frame hence the proof above applies when the $\neg\Box Y; \neg\Box P$ part is present. If there are no eventualities in w_0 then the seriality and transitivity of R guarantees that there is some world w' with wRw' such that $w' \models X; \Box X$ and we are done. ●

Theorem 12 (Shvarts) *If X is a finite set of formulae and X is $\mathcal{C}L$-consistent then there is an* **L**-*model for X on a finite* **L**-*frame where* $\mathbf{L} \in \{\mathbf{K45}, \mathbf{K45D}\}$ *[Shv89].*

Proof for L = K45: Suppose X is $\mathcal{C}K45$-consistent and create a $\mathcal{C}K45$-saturated superset $w_0 \subseteq X^*_{K45}$ of X as usual. If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \emptyset\rangle$ is the desired model graph since (i)-(iii) are satisfied.

Otherwise let $Q_i, Q_2, \cdots, Q_k$ be all the formulae such that $\neg\Box Q_i \in w_0$ and create a Q_i-successor for each Q_i using the (45) rule. Continue construction of one such sequence $S = w_0 \prec w_1 \prec \cdots$ always choosing the successor that fulfills an eventuality *not* fulfilled by the current sequence. Since X^*_{K45} is finite, we must sooner or later come to a node w_m such that the sequence $S = w_0 \prec w_1 \prec \cdots \prec w_m$ already contains *all* the successors of w_m. That is, it is not possible to choose a new successor. Choose the successor w_x of w_m that appears earliest in S and put $w_m \prec w_x$ giving $S = w_0 \prec w_1 \prec \cdots \prec w_x \prec \cdots \prec w_m \prec w_x$. There are two cases to consider depending on whether $w_x = w_0$ or $w_x \neq w_0$.

Case 1: If $w_x = w_0$, put R as the reflexive, transitive and symmetric closure of $\prec$ over $W_0 = \{w_0, w_1, \cdots, w_m\}$. This gives a frame $\langle W_0, R\rangle$ which is a nondegenerate cluster of type 1.

Case 2: If $w_x \neq w_0$, put $W_0 = \{w_0, w_x, w_{x+1}, \cdots, w_m\}$, discarding $w_1, w_2, \cdots, w_{x-1}$, and let R' be the reflexive, transitive and *symmetric* closure of $\prec$ over $W_0 \setminus \{w_0\}$. That is, $R' = \{(w_i, w_j) | w_i \in W_0, w_j \in W_0, i \geq x, j \geq x\}$. Now put $R'' = R' \cup \{(w_0, w_x)\}$ and let R be the transitive closure of R''. The frame $\langle W_0, R\rangle$ is now of type 2.

Property (i) is satisfied by $\langle W_0, R\rangle$ by construction. We show that (ii) and (iii) are satisfied as follows.

Proof of (ii): The (45) rule also carries *all* eventualities from the numerator to the denominator, including the one it fulfills. Therefore, for all $w_i \in W_0$ we have

$$\neg\Box P \in w_i \text{ implies } \neg\Box P \in w_m.$$

But we stopped the construction at w_m because no new Q_i-successors for w_m could be found. Hence there is a Q_i-successor for each eventuality of w_m. Since we have a cycle, and eventualities cannot disappear, these are all the eventualities that appear in the cycle. Furthermore, we chose w_x to be the successor of w_m that was earliest in the sequence S. Hence all of the eventualities of w_m are fulfilled by the sequence $w_x R \cdots R w_m$. All the eventualities of w_0 are also in w_m, hence (ii) holds.

Proof of (iii): The (45) rule carries all formulae of the form $\Box P$ from its numerator to its denominator. Hence $\Box P \in w$ and $w \prec v$ implies that $P \in v$ and $\Box P \in v$. But we know that $w_x \prec \cdots \prec w_m \prec w_x$ forms a cycle, hence (iii) holds as well.

Proof for L = K45D: Based on the previous proof. If the $(45D)$ rule is ever used with no eventualities present then this can only happen when w_0 contains no eventualities. For if w_0 contained an eventuality then so would all successors.

So if w_0 contains no eventualities and no formulae of the form $\Box P$ then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph. This gives a frame of type 3.

Otherwise, let $Q_1, \cdots, Q_k$ be all the formulae such that $\neg\Box Q_i \in w_0$ and let $P_1, \cdots, P_m$ be all the formulae such that $\Box P_j \in w_0$. Create a successor w_1 for w_0 using $(45D)$ for some Q_i or P_j and continue creating successors using $(45D)$, always choosing a successor new to the sequence until no new successors are possible. Choose w_x as the successor nearest to w_0 giving a cycle $w_0 \prec \cdots \prec w_x \prec \cdots \prec w_m \prec w_x$ and discard $w_1, w_2, \cdots w_{x-1}$ as in the previous proof.

As in the previous proof, $w_x = w_0$ gives a frame of type 3 and $w_x \neq w_0$ gives a frame of type 4.

Properties (i)-(iii) can be proved in a similar manner •

Note that the requirement to continually choose a new successor is tantamount to following an infinite path in Shvarts' formulation. That is, the inevitable cycle that we encounter constitutes an infinite branch if it is unfolded out.

5.3.2 An Alternative Tableau System for K45D

Let

$$\mathcal{C}K45D' = \mathcal{C}K45 + (D) = \{(0), (\wedge), (\vee), (\neg), (D)\} \cup \{(45)\} \cup \{(\theta)\}$$

where the three sets are the static, transitional and structural rules of $\mathcal{C}K45D'$.

Let $X^*_{K45D'} = Sf \neg Sf \Box \widetilde{X}$.

A set is $\mathcal{C}K45D'$-saturated if it is closed with respect to $(0), (\wedge), (\vee), (\neg)$ and (D).

It is easy to show that we can obtain a $\mathcal{C}K45D'$-saturated X^* for a given X with $X \subseteq X^* \subseteq X^*_{K45D'}$.

The (D) rule is also sound with respect to **K45D**-frames since R is serial in these frames.

The completeness of $\mathcal{C}K45D'$ can be shown using an argument like that for $\mathcal{C}K45$ with the knowledge that the (D) rule adds an eventuality $\neg\Box\neg P$ for every formula $\Box P$. We omit details.

5.3.3 An Embedding of S5 into K45

A syntactic proof of the following theorem is given by Shvarts [Shv89].

Theorem 13 *A formula A is an* **S5***-theorem if and only if $\Box A$ is a* **K45***-theorem.*

Proof ($\Rightarrow$): Suppose $\Box A$ is a **K45**-theorem. Then by the fact that **K45** is characterised by the class of finite sharp tacks (type 1 and 2) we have that A is valid on all frames of type 1 and type 2. But type 2 frames have a finite nondegenerate cluster as their second cluster. Thus A is valid on all finite nondegenerate clusters, that is, on all **S5**-frames. Since **S5** is characterised by **S5**-frames, A is an **S5**-theorem.

Proof ($\Leftarrow$): We have to show that if A is an **S5**-theorem then $\Box A$ is a **K45**-theorem. Contrapositively, we have to show that if $\Box A$ is not a **K45**-theorem then A is not an **S5**-theorem. Suppose $\Box A$ is not a **K45**-theorem. Thus, there is some **K45**-model $\mathcal{M} = \langle W, R, V \rangle$ with $w_0 \in W$ and $w_0 \not\models \Box A$.

Case 1: Suppose $\mathcal{M}$ is of type 1. Then $\langle W, R \rangle$ cannot be a degenerate cluster since $w_0 \models \Diamond\neg A$; that is, there must be a some $w' \in W_0$ such that $w_0 R w'$. Thus $\langle W, R \rangle$ is either a simple or proper cluster with some $w' \in W$ such that $w' \models \neg A$. But then $\langle W, R \rangle$ is an **S5**-frame and $\mathcal{M}$ is also an **S5**-model falsifying A.

Case 2: If $\mathcal{M}$ is of type 2 then either w_0 is in the first degenerate cluster C_1 or in the last nondegenerate cluster C_2. Regardless of where w_0 appears, there is a w' in the second cluster such that $w' \models \neg A$ because $w_0 \not\models \Box A$. Take $\mathcal{M}' = \langle W', R', V' \rangle$ where $\mathcal{M}'$ is $\mathcal{M}$ restricted to $W' = W \setminus C_1$. The model $\mathcal{M}'$ is an **S5**-model falsifying A. •

5.3.4 Connections Between S5 and K45D

If all atomic components of a formula A are within the scope of a $\Box$ connective then A is called a **modalised** formula [Min70, Shv89].

Theorem 14 (Shvarts) *A modalised formula A is an* **S5***-theorem iff it is a* **K45D***-theorem [Shv89].*

Thus, **S5** and **K45D** have the same modalised formulae as theorems and $CK45D$ allows us to test whether a modalised formula A is a theorem of **S5**.

5.4 Hanson's Rules for S4 and S5

Hanson [Han66a] gives Kripke-like tableau systems for **S4** and **S5** using a form of $(sfcT)$ as early as 1966. The tableau systems presented below are not exactly Hanson's systems but the ideas are his. See page 129 for details of Hanson's rules.

Let

$$CS4' = CS4 + (sfcT) = \{(0), (\wedge), (\vee), (\neg), (sfcT), (T)\} \cup \{(S4)\} \cup \{(\theta)\}.$$

Let $X^*_{S4'} = \widetilde{X}$.

A set is $CS4'$-saturated if it is $CS4'$-consistent and closed with respect to $(0), (\neg), (\wedge), (\vee)$, (T) and $(sfcT)$.

The saturation termination lemma goes through since $(sfcT)$ increases the number of formulae in a reduction and no static rule of $CS4'$ decreases the number of formulae. Hence every $CS4'$-consistent X can be extended to give a $CS4'$-saturated X^* such that $X \subseteq X^* \subseteq X^*_{S4'}$. It is easy to see that $(sfcT)$ is sound with respect to **S4**-frames.

For completeness, suppose that X is $CS4'$-consistent. We have to give a model graph for X which is based on an **S4**-frame.

Let w_0 be some $CS4'$-saturation of X so that $X \subseteq w_0 \subseteq X^*_{S4'}$. Create a tree of $CS4'$-consistent and $CS4'$-saturated subsets of $X^*_{S4'}$ as follows.

If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph. Otherwise, let $Q_1, Q_2, \cdots, Q_k$ be all the formulae such that $\neg\Box Q_i \in w_0$ and $\neg Q_i \notin w_0$. Create a Q_i-successor v_i of level 1 for each Q_i using the (θ) and $(S4)$ rules and continue in this way to obtain the nodes of level 2 and so on with the following termination condition:

(*) if $w_0 \prec w_1 \prec \cdots \prec w_{i-1} \prec w_i$ is a path in this construction and $i \geq 1$ is the least index such that $\Box P \in w_i$ implies $\Box P \in w_{i-1}$, then put $w_i \prec w_{i-1}$ giving a cycle on this path and stop!

First of all, this termination condition is satisfactory since $(S4)$ ensures that $\Box P \in w_j$ implies $\Box P \in w_{j+1}$ so that $\Box$-formulae accumulate and we eventually run out of new $\Box$-formulae since $X^*_{S4'}$ is finite.

Let R be the reflexive and transitive closure of $\prec$. It is obvious that clusters of R form a tree. To prove that $\langle W_0, R\rangle$ is a model graph for X we have to prove (i)-(iii).

Clearly (i) holds so we have to prove that (ii) and (iii) hold.

(ii) Suppose $\neg\Box P \in w_j$ where w_j is some arbitrary world of W_0. If the termination condition was not applied to w_j, then either $\neg P \in w_j$ or w_j has a P-successor fulfilling $\neg\Box P$ by $(S4)$ and so (ii) is satisfied. That is (ii) holds for any world to which the termination condition was not applied.

If the termination condition was applied to w_j, then it could not have been applied to w_{j-1}. Hence (ii) holds for w_{j-1}. So all we have to show is that $\neg\Box P \in w_{j-1}$ because, in this case, (ii) would then hold for w_j from the fact that $w_j R w_{j-1}$.

Suppose to the contrary that $\neg\Box P \notin w_{j-1}$. Then $\Box P \in w_{j-1}$ by $(sfcT)$, and $\Box P \in w_j$ by $(S4)$ contradicting the $\mathcal{C}S4'$-consistency of w_j since $\neg\Box P \in w_j$. Hence (ii) also holds.

(iii) Suppose $\Box P \in w_j$. If (*) was not applied to w_j then (iii) holds as for $\mathcal{C}S4$ by (T) since $(S4)$ preserves $\Box$-formulae. If (*) was applied to w_j then (iii) would follow from $\Box P \in w_{j-1}$ by $(S4)$ and (T). But this is exactly what (*) guarantees. Hence (iii) holds as well. •

The advantage of adding $(sfcT)$ is that the termination condition is much easier to check than the one for $\mathcal{C}S4$ where we have to look through all predecessors of w_j. That is, if we obtain a decision procedure from the completeness proofs, then the one from $\mathcal{C}S4'$ is easier to implement than the one from $\mathcal{C}S4$ because the $\mathcal{C}S4'$ version requires us to compare the current node with its parent only. For the $\mathcal{C}S4$ version, we have to compare it with all previous nodes. Furthermore, for the $\mathcal{C}S4'$ version, we only have to check that the formulae of the form $\Box P$ are common to both nodes whereas for the $\mathcal{C}S4$ version we have to check that all formulae in the nodes are the same. And all we have done is to add one rule.

To see this advantage suppose $w_0 \prec w_1 \prec \cdots \prec w_k \prec w_{k+1} \prec \cdots \prec w_m \prec w_k$ is a cycle obtained in the $\mathcal{C}S4$ completeness proof. Since all successors are due to $(S4)$ only, $w_m \prec w_k$ implies that all nodes from w_k onwards have the same $\Box$-formulae. Thus the $\mathcal{C}S4'$ procedure would have stopped at w_{k+1} giving $w_0 \prec w_1 \prec \cdots \prec w_k \prec w_{k+1} \prec w_k$.

So in general, the $\mathcal{C}S4'$ procedure gives smaller counter models than the $\mathcal{C}S4$ procedure. This is due to the fact that $(sfcT)$ packs more information into each node by forcing each node to be subformula-complete. That is, a value of true or false is forced onto each subformula of the formulae in the nodes of a $\mathcal{C}S4'$ model construction, whereas the absence of $(sfcT)$ in $\mathcal{C}S4$ allows some subformulae to be "undefined" in the $\mathcal{C}S4$ model construction. This packing of more information will of course take more time per node, so although we save space, we may not save much time. But the time to compare nodes is the most significant cost and this is reduced in the $\mathcal{C}S4'$ version since we compare with the immediate ancestor and only compare formula of the form $\Box P$.

Hanson also suggests a tableau system for **S5** along these lines, but in it he uses a rule which explicitly adds a formula to the parent node to obtain symmetry. This is forbidden for our tableau since we cannot return to previous nodes. The details are discussed in Chapter 7 (page 129).

5.5 Alternatives for Various Systems

5.5.1 A Tableau System for S5 With the Subformula Property

The tableau calculus $\mathcal{C}S5$ does not possess the subformula property because it contains (5). Here we present a tableau system for **S5** that possesses the subformula property. It is an obvious amalgamation of ideas from Hanson [Han66a], Fitting [Fit83] and Rautenberg [Rau83].

Let

$$\mathcal{C}S5' = \{(0), (\wedge), (\vee), (\neg), (T), (sfcT)\} \cup \{(S5)\} \cup \{(\theta)\}$$

where $(S5)$ is:

$$(S5)\ \frac{\Box X; \neg\Box Y; \neg\Box P}{\Box X; \neg\Box Y; \neg P}$$

and the three sets correspond to the static, transitional and structural rules of $\mathcal{C}S5'$.

Let $X^*_{S5'} = \widetilde{X}$.

It is easy to show that $(S5)$ is sound with respect to **S5**-frames using the facts that such frames are reflexive, transitive and symmetric. And a set is $\mathcal{C}S5'$-saturated if it is $\mathcal{C}S5'$-consistent and closed with respect to $(0), (\wedge), (\vee), (\neg), (T)$ and $(sfcT)$. It is easy to show that we can obtain a $\mathcal{C}S5$-saturated X^* for a given X with $X \subseteq X^* \subseteq X^*_{S5'}$.

For completeness suppose X is $\mathcal{C}S5'$-consistent and create a $\mathcal{C}S5'$-saturated superset w_0 with $X \subseteq w_0 \subseteq X^*_{S5'}$ as usual.

If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph. Otherwise, let $Q_1, Q_2, \cdots, Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$ and $\neg Q_i \notin w_0$. Create a Q_i-successor v_i of level 1 for each Q_i using the $(S5)$ rule and stop!

Let $W_0 = \{w_0, v_1, v_2, \cdots, v_m\}$. Consider any two nodes v_i and v_j of level 1 so that $w_0 \prec v_i$ and $w_0 \prec v_j$ with $i \neq j$. We claim that:

(a) $\Box P \in v_i$ implies $\Box P \in w_0$ implies $\Box P \in v_j$; and

(b) $\neg\Box P \in v_i$ implies $\neg\Box P \in w_0$ implies there exists a $w \in W_0$ with $\neg P \in w$.

Putting R equal to the reflexive, symmetric and transitive closure of $\prec$ gives an **S5**-model graph since (i)-(iii) follow from (a) and (b).

Proof of (a): Suppose $\Box P \in v_i$, then $P \in v_i$ by (T). Also, $\Box P \in Sf(w_0)$ as there are no building up rules, hence $\Box P \in w_0$ or $\neg\Box P \in w_0$ by $(sfcT)$. If $\neg\Box P \in w_0$ then either $\neg P \in v_i$ or $\neg\Box P \in v_i$ by $(S5)$. The first contradicts the $\mathcal{C}S5'$-consistency of v_i since $P \in v_i$ and so does the second since $\Box P \in v_i$. Hence $\Box P \in w_0$. And then $\Box P \in v_j$ by $(S5)$ and $P \in v_j$ by (T).

Proof of (b): Suppose $\neg\Box P \in v_i$. Then as there are no building up rules, $\neg\Box P \in Sf(w_0)$. Hence $\Box P \in w_0$ or $\neg\Box P \in w_0$ since w_0 is subformula-complete. If $\Box P \in w_0$ then $\Box P \in v_i$ by $(S5)$, contradicting the $\mathcal{C}S5'$-consistency of v_i since $\neg\Box P \in v_i$ by supposition. Hence $\neg\Box P \in w_0$. And then either $\neg P \in w_0$, or there is some v_k such that $\neg P \in v_k$ by $(S5)$. That is, the w we seek is either w_0 itself, or one of the nodes of level 1. •

The system $\mathcal{C}S5'$ is effectively Fitting's system for **S5** with the $(sfcT)$ rule replacing his semi-analytic cut rule. But note that it is trivial to implement $\mathcal{C}S5'$ and $\mathcal{C}S5$ whereas this is not so of Fitting's systems as the semi-analytic rule does not bound the superformula in the cut-class.

5.5.2 Tableau Systems For K45 and K45D Containing Analytic Cut

Another way to obtain a tableau calculus for **K45** is to add analytic cut by adding (sfc) to $\mathcal{C}K45$. Note that we add (sfc), not $(sfcT)$ since **K45**-frames are not reflexive. The advantage is that the completeness proof, and hence the deterministic decision procedure based upon it, is much simpler.

Let

$$\mathcal{C}K45' = \{(0), (\neg), (\wedge), (\vee), (sfc)\} \cup \{(45)\} \cup \{(\theta)\}.$$

Let $X^*_{K45'} = \widetilde{X}$.

A set is $\mathcal{C}K45'$-saturated if it is $\mathcal{C}K45'$-consistent and closed with respect to $(0), (\neg), (\wedge)$, $(\vee)$ and (sfc). It is easy to show that (sfc) is sound with respect to **K45**-frames since it is essentially a classical propositional (static) rule.

For completeness suppose X is $\mathcal{C}K45'$-consistent and create a $\mathcal{C}K45'$-saturated superset w_0 with $X \subseteq w_0 \subseteq X^*_{K45'}$ as usual. If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \emptyset\rangle$ is the desired model graph since (i)-(iii) are satisfied.

Otherwise, let $Q_i, Q_2, \cdots, Q_m$ be all the formulae such that $\neg\Box Q_i \in w_0$ and create a Q_i-successor v_i for each Q_i using the (45) rule. This gives all the nodes of level 1, so put $w_0 \prec v_i$, for each $i = 1 \cdots m$, and stop!

Consider any two nodes v_i and v_j with $i \neq j$. Using the facts that each node is subformula-complete and there are no building up rules, we show that

(a) $\Box P \in v_i$ implies $\Box P \in w_0$ implies $P \in v_j$ and $\Box P \in v_j$;

(b) $\neg\Box P \in v_i$ implies $\neg\Box P \in w_0$ implies there exists a v_k such that $\neg P \in v_k$.

Proof of (a): Suppose $\Box P \in v_i$. Then $\Box P \in Sf(w_0)$ and so $\Box P \in w_0$ or $\neg\Box P \in w_0$ since w_0 is subformula-complete. If $\neg\Box P \in w_0$ then $\neg\Box P \in v_i$ by (45), contradicting the $\mathcal{C}K45'$-consistency of v_i. Hence $\Box P \in w_0$. Note that this holds only because the (45) rule carries $\neg\Box P$ into its denominator along with $\neg\Box Y$.

Proof of (b): As for (a) except uniformly replace $\neg\Box P$ by $\Box P$ and vice-versa. The crux of the proof is that (45) preserves all formulae of the form $\Box P$ and $\neg\Box P$. •

Hence we can put $v_i R v_j R v_i$ for all v_i and v_j giving a reflexive, transitive and symmetric nondegenerate cluster. If we also put $w_0 R v_i$ for all $i = 1 \cdots m$, then we obtain a **K45**-frame of type 2. If some $v_k = w_0$ then we obtain a nondegenerate **K45**-frame of type 1.

In either case, (i)-(iii) are satisfied giving a model graph and hence a **K45**-model for X. •

Note the difference from $\mathcal{C}S5$ where we could guarantee that $\Box P \in v_i$ implies $P \in v_i$ by (T). The (45) rule carries $\neg\Box P$ into the denominator specifically to make up for the absence of (T) and allow the proof of (a) to go through.

For **K45D** let

$$\mathcal{C}K45D'' = \{(0), (\neg), (\wedge), (\vee), (sfc)\} \cup \{(45D)\} \cup \{(\theta)\}.$$

Let $X^*_{K45D''} = \widetilde{X}$.

We omit details of soundness and completeness.

5.6 Symmetry, Subformula Property and Analytic Cut

For the logics with a symmetric R we seem to need analytic cut, either as (sfc) or as $(sfcT)$. The subformula property can be regained for some logics by changing the transitional rules to carry more information from the numerator to the denominator. But note that a building up rule seems essential for $\mathcal{C}B$, so not all the systems are amenable to this trick.

Notice that the effects of $(sfcT)$ on w_0 when R is to be transitive and there are no building up rules like (5) is to flush out all the eventualities that could possibly appear in any successor. That is, if $\neg\Box P$ is going to appear in a successor, it must be in $Sf(w_0)$. But then it must be in w_0 since otherwise by $(sfcT)$, we would have $\Box P \in w_0$ contradicting the appearance of $\neg\Box P$ in any consistent successor. Hence the number of eventualities never increases as all the eventualities that will ever appear are already in w_0.

The idea behind (sfc) and $(sfcT)$ is to put extra information into a node before leaving it for good. That is, once we leave a node in our tableau procedure, we can never return to it. Also, the transitional rules usually lose information in the transition from the numerator to the denominator. The (sfc) and $(sfcT)$ rules are used to make up for this "destructive" aspect of our transitional rules.

5.7 Eliminating Thinning

Thinning is eliminable from all the systems of this chapter just as it was in $\mathcal{C}S4$ (page 62). For example, the effects of (θ) can be built into the $(K4)$ rule by changing it from

$$(K4)\ \frac{\Box X; \neg\Box P}{X; \Box X; \neg P} \qquad \text{to} \qquad \frac{Y; \neg\Box P}{Y'; \Box Y'; \neg P}$$

where $Y' = \{Q \mid \Box Q \in Y\}$ and simultaneously changing the basic axiomatic rule from (0) to (0′).

5.8 Bibliographic Remarks

The proofs and rules presented in this section are all from Rautenberg [Rau83] except for $\mathcal{C}D'$, $\mathcal{C}D4$, $\mathcal{C}S4'$, $\mathcal{C}K45$, $\mathcal{C}K45D$, $\mathcal{C}K45'$, $\mathcal{C}K45D'$, $\mathcal{C}K45D''$ and $\mathcal{C}S5'$. The system $\mathcal{C}D4$ is an obvious extension of Rautenberg's system $\mathcal{C}D$, and $\mathcal{C}D'$ is lifted straight from Fitting [Fit83]. The $\mathcal{C}S4'$ system is based on ideas of Hanson [Han66a]. The tableau systems $\mathcal{C}K45$, $\mathcal{C}K45D$, $\mathcal{C}K45D'$ are based on the work of Shvarts [Shv89]. The tableau systems $\mathcal{C}S5'$, $\mathcal{C}K45'$, and $\mathcal{C}K45D''$ are an amalgamation of ideas of Rautenberg, Fitting and Shvarts.

There is one minor flaw in Rautenberg's paper [Rau83] because he does not distinguish between transitional and static modal rules. Hence his rules for (T), (D), (B), (sfc) and $(sfcT)$ do not carry all the numerator formulae into their denominators. For example, Rautenberg's (T) rule is

$$\frac{X; \Box P}{X; P}$$

whereas ours is

$$(T)\ \frac{X; \Box P}{X; \Box P; P}$$

Thus there is no proviso for contraction in his systems and as we saw in Example 3 (page 47), contraction is necessary for some modal systems. Consequently, Rautenberg's systems are not complete as claimed. For example, his system for $\mathcal{C}S5$ is unable to prove the **S5**-theorem $\Diamond\Box P \Rightarrow P$ because the proof requires the $(sfcT)$ rule to carry its principal formula into its denominator, and his $(sfcT)$ rule does not do so.

The flaw in Rautenberg's tableau rules is fundamental since we cannot just add a rule of contraction like:

$$\frac{X; P}{X; P; P}$$

because then we no longer work with sets but with multisets. Using multisets, the notion of beginning a tableau for X with X makes no sense. Also, termination depends on the fact that we start with only one copy of each element in X. If we allow more than one copy then we cannot be sure that adding just one more copy will not give a closed tableau.

Rautenberg's rules are complete for Smullyan-tableau where contraction on any formula is implicit, but Rautenberg does not work with Smullyan-tableau since he explicitly associates sets with the denominator and numerators of each rule.

In order to fix the flaw, we have explicitly carried *all* numerator formulae into the denominators for the *static* modal rules. Then, a complication arises when saturating a set as we are allowed to use any rule only once on any formula. The notion of strict-$\mathcal{C}L$-saturation becomes necessary and we have to take the union of the intermediate one-step saturation sets to obtain a $\mathcal{C}L$-saturated set.

Fitting's [Fit83, page 81] sequent systems suffer from an identical flaw since there is no sequent equivalent of his "repetition rule" [Fit83, page 37-38] and this is the rule that allows contraction.

As we note later on page 132, the embedding of **S5** into **K45** explicitly proved by Shvarts is implicit in the $\mathcal{C}S5\pi$ tableau system of Fitting.

Chapter 6

The Temporal Systems

In this chapter we first present cut-free tableau systems for the logics **S4.3**, **S4.3.1** and **S4.14**. These logics are important when the modalities $\Box$ and $\Diamond$ are given the temporal interpretation "always" and "eventuality". This work was done before I found Rautenberg's paper and seems to be new since I have been unable to find any similar work for the last two logics; see Chapter 7. Zeman [Zem73] gives a cut-free tableau system for **S4.3**, but is unable to extract the analogous cut-free sequent system. With hindsight though, these tableau systems are clearly just extensions of Rautenberg's method and are presented in this light.

We then discuss cut-free tableau systems for various extensions of **S4**. Although these extensions are not usually associated with temporal interpretations, the *finite* frames that characterise these logics have interesting temporal interpretations. We also show that some well-known axiomatic logics are different axiomatisations of the same logic since they are all characterised by the same class of (finite) frames. This work is not particularly original but it unifies disparate notes made by various authors into a coherent form.

The methods for proving soundness and completeness are as in the last chapter except that we treat each logic individually. The main reason is that the tableau rules are somewhat bizarre and the corresponding soundness and completeness proofs are more intricate.

6.1 A Cut-free Tableau System for S4.3

The logic **S4.3** is characterised by the class of (frames which are) finite linear sequences of nondegenerate clusters; see page 33. It can be shown that **S4.3** is also characterised by either of the single frames $\langle \mathcal{Q}, \leq \rangle$ or $\langle \mathcal{R}, \leq \rangle$ where $\mathcal{R}$ is the set of real numbers and $\mathcal{Q}$ is the set of rational numbers [Gol87, page 57]. Goldblatt [Gol87, page 57] shows that any **S4.3**-frame is a p-morphic image of $\langle \mathcal{I}, \leq \rangle$, where $\mathcal{I}$ is either $\mathcal{R}$ or $\mathcal{Q}$, so that

$$\langle \mathcal{I}, \leq \rangle \models A \text{ iff } \vdash_{S4.3} A.$$

Consequently, between any two worlds there is always a third and **S4.3**-frames are said to be dense. So, although we always work with finite sequences of clusters, **S4.3** is the logic where time is modelled as the positive real or positive rational number line. That is, there is an initial point corresponding to "now" and then a continuum of points to the right of "now" corresponding to the real or rational number line. Since **S4.3**-frames are finite sequences of nondegenerate clusters, this may seem odd. The linear continuum can be obtained by "bulldozing" the proper clusters of an **S4.3**-frame into an infinite linear sequence [Seg71, page 80] [HC84, page 82-88]. That is, proper clusters, when bulldozed, give rise to a continuum.

6.1.1 Tableau Rules for $\mathcal{C}S4.3$

The following notation is used in this section:

$\neg\Box\{P_1, \cdots, P_k\}$ stands for $\{\neg\Box P_1, \neg\Box P_2, \cdots, \neg\Box P_k\}$;

if $Y = \{P_1, \cdots, P_k\}$ then $\overline{Y_i}$ stands for $Y \setminus \{P_i\}$.

The tableau system $\mathcal{C}S4.3$ is:

$$\mathcal{C}S4.3 = \{(0), (\neg), (\wedge), (\vee), (T)\} \cup \{(S4.3)\} \cup \{(\theta)\}$$

where $(S4.3)$ is:

$$(S4.3) \quad \frac{\Box X; \neg\Box\{P_1, \cdots, P_k\}}{\Box X; \neg\Box\overline{Y_1}; \neg P_1 \quad | \quad \cdots \quad | \quad \Box X; \neg\Box\overline{Y_k}; \neg P_k}$$

where $Y = \{P_1, \cdots, P_k\}$ and $\overline{Y_i} = Y \setminus \{P_i\}$

and the three sets are respectively the static, transitional and structural rules of $\mathcal{C}S4.3$.

Semantic Intuitions: The $(S4.3)$ rule is based on a consequence of the characteristic **S4.3** axiom 3. Adding 3 to **S4** gives a weakly-connected R for **S4.3** so that each of the following is a theorem of **S4.3** [Zem73, page 232-233]:

$$\begin{aligned}
\Box(\Box P \vee Q) \wedge \Box(\Box Q \vee P) &\Rightarrow \Box P \vee \Box Q \\
\neg(\Box P \vee \Box Q) &\Rightarrow \neg(\Box(\Box P \vee Q) \wedge \Box(\Box Q \vee P)) \\
\neg\Box P \wedge \neg\Box Q &\Rightarrow \neg\Box(\Box P \vee Q) \vee \neg\Box(\Box Q \vee P) \\
\neg\Box P \wedge \neg\Box Q &\Rightarrow \Diamond(\neg\Box P \wedge \neg Q) \vee \Diamond(\neg\Box Q \wedge \neg P).
\end{aligned}$$

So when there are only two eventualities, say $\neg\Box P$ and $\neg\Box Q$, the $(S4.3)$ rule can be seen as a disjunctive choice between the two possible orderings of these eventualities together with an appropriate "jump" to the corresponding worlds. The soundness of the $(S4.3)$

rule follows from a generalised version of this last **S4.3**-theorem containing k formulae of the form $\neg\Box P_1 \cdots \neg\Box P_k$ [Zem73, pages 236-238].

The sequent analogue of the $(S4.3)$ rule is:

$$\frac{S_1 \qquad S_2 \qquad \cdots \qquad S_k}{\Box\Gamma \longrightarrow \Box A_1, \cdots, \Box A_k} \quad (\rightarrow \Box : S4.3)$$

where for $1 \leq i \leq k$

$$Y = \{A_1, \cdots, A_k\}$$

$$\overline{Y_i} = Y \setminus \{A_i\}$$

$$S_i = \Box\Gamma \longrightarrow A_i,\ \Box\overline{Y_i}$$

Let $X^*_{S4.3} = \widetilde{X}$.

Lemma 8 *If there is a closed $CS4.3$ tableau for X then there is a closed $CS4.3$ tableau for X with all nodes in the* finite set $X^*_{S4.3}$.

Proof: Obvious from the fact that all rules for $CS4.3$ operate with subsets of $X^*_{S4.3}$ only. ●

A set X is $CS4.3$-saturated if it is $CS4.3$-consistent and closed with respect to the static rules of $CS4.3$.

Lemma 9 *For each finite $CS4.3$-consistent X there is an effective procedure to construct some finite $CS4.3$-saturated X^* with $X \subseteq X^* \subseteq X^*_{S4.3}$.*

Proof: As for $CS4$ since the static rules for $CS4.3$ and $CS4$ are identical. ●

6.1.2 Soundness of $\mathcal{C}S4.3$

Theorem 15 *The tableau system $\mathcal{C}S4.3$ is sound with respect to* **S4.3**-*frames.*

The static and structural rules of $\mathcal{C}S4.3$ are those of $\mathcal{C}S4$ and **S4.3**-frames are also reflexive and transitive. Thus all the proofs of soundness for these rules are as for $\mathcal{C}S4$ since they depend only on the reflexivity and transitivity of R. The soundness of the remaining rule is proved below.

Proof for $(S4.3)$ **:** We have to show that if the numerator of the $(S4.3)$ rule is **S4.3**-satisfiable, then so is at least one of the denominators. That is, we have to show that if

$$\Box X; \neg\Box\{P_1, P_2, \cdots, P_k\}$$

is **S4.3**-satisfiable, with $Y = \{P_1, P_2, \cdots, P_k\}$ and finite $k > 0$, then there is some i with $1 \leq i \leq k$ such that

$$\Box X; \neg P_i; \neg\Box\overline{Y_i}$$

is **S4.3**-satisfiable. Finiteness is not essential for soundness.

As before assume that $\Box X; \neg\Box\{P_1, P_2, \cdots, P_k\}$ is **S4.3**-satisfiable. That is, there is an **S4.3**-model $\mathcal{M} = \langle W, R, V\rangle$ with $w_0 \in W$ such that $w_0 \models \Box X; \neg\Box\{P_1, P_2, \cdots, P_k\}$.

Then, by the definition of $\models$ there exist $w_1, w_2, \cdots, w_k$ not necessarily distinct worlds in W such that $w_0 R w_j$ and $w_j \models \neg P_j$, for $1 \leq j \leq k$.

Now, we know that an **S4.3**-model is a (finite) linear sequence of nondegenerate clusters and that $w_0 \in W$. If $\mathcal{M}$ is just one nondegenerate (proper or simple) cluster then R is reflexive, transitive and symmetric, and we immediately have that some world satisfies $\Box X; \neg P_j; \neg\Box\overline{Y_j}$ since $w_j \models \neg P_j$ implies $w_j \models \neg\Box P_j$ implies $w_j \models \Box\neg\Box P_j$ in such a cluster. Thus the only interesting case is when $\mathcal{M}$ is not a single cluster. The $\lhd$ ordering defined previously on page 31 is then appropriate. Label the cluster containing w_0 as C_0.

Let C_f be the first cluster equal to or after C_0 that contains a world w_f that fulfills any of the eventualities in $E = \{\neg\Box P_1, \neg\Box P_2, \cdots, \neg\Box P_k\}$. That is, $w_f \in C_f$ such that $w_f \models \neg P_m$ for some $1 \leq m \leq k$. The world w_f may satisfy other eventualities from E and it is even possible that C_f is C_0, but we do not care. Our aim is to prove that $w_f \models \neg\Box\overline{Y_m}$ where $\overline{Y_m} = \{P_1, P_2, \cdots, P_k\} \setminus \{P_m\}$.

Suppose to the contrary that $w_f \not\models \neg\Box\overline{Y_m}$. Then $w_f \models \Box P_l$, for some $1 \leq l \neq m \leq k$. If $w_f \in C_0$ then $w_f R w_0$ implying that $w_0 \models \Box P_l$, but we already know that $w_0 \models \neg\Box P_l$ by supposition. Hence, $w_f \notin C_0$ and $C_0 \lhd C_f$. But we also know that there must be some world $w_g \in W$ such that $w_0 R w_g$ and $w_g \models \neg P_l$. Label the cluster containing w_g as C_g. Since $w_f \models \Box P_l$, all worlds w_x with $w_f R w_x$ are such that $w_x \models P_l$. Hence $w_f \not R w_g$, that is, $C_g \lhd C_f$. But this contradicts the choice of C_f as the *first* cluster with respect to C_0 that fulfills any of the eventualities in E. So it must be that $w_f \models \neg\Box\overline{Y_m}$. Remember we have already disposed of the case where $\mathcal{M}$ is a single cluster.

Since $w_0 R w_f$ and $w_f R w_f$ and $w_0 \models \Box X$, we have $w_f \models \Box X; \neg P_m$ by transitivity of R. Hence $w_f \models \Box X; \neg P_m; \neg\Box\overline{Y_m}$ and we are done. ●

6.1.3 Completeness of $CS4.3$

Theorem 16 *If X is a finite set of formulae and X is $CS4.3$-consistent then there is an* **S4.3**-*model for X on a finite frame $\langle W_0, R\rangle$ which is a finite, reflexive and transitive sequence of nondegenerate clusters (and hence an* **S4.3**-*frame).*

Proof sketch: The completeness proof of $CS4.3$ is similar to the completeness proof for $CS4$. The differences are that only *one* branch is constructed, and that in doing so, the $(S4.3)$ rule is used instead of the $(S4)$ rule. The basic idea is to follow one branch until it cycles and then to use thinning to sprout a continuation of the branch, thus escaping out of the cycle. The cycles give rise to clusters in the resulting **S4.3**-model. Thinning seems essential.

Proof: Let $w_0 \supseteq X$ be $CS4.3$-saturated, $w_0 \subseteq X^*_{S4.3}$. If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph since (i)-(iii) are satisfied. Otherwise do Step 1 below.

Step1: Let $Y = \{Q \mid \neg\Box Q \in w_0 \text{ and } \neg Q \notin w_0\}$ so that $\neg\Box Y$ contains all the eventualities in w_0 that are unfulfilled by w_0. If Y is empty, all eventualities are fulfilled, so go to Step 4. Otherwise $\neg\Box Y = \neg\Box\{Q_1, \cdots Q_m\}$, with $m \geq 1$. Put $w' = w_0^{\Box}$. Now $\Box w' \subseteq w_0$, so $\Box w' \cup \neg\Box Y$ is $CS4.3$-consistent by (θ); hence so is at least one of

$$X_j = \Box w' \cup \{\neg Q_j\} \cup \neg\Box\overline{Y_j}, \text{ for } j = 1, \cdots, m$$

by $(S4.3)$.

Choose one[1] such $CS4.3$-consistent set X_i and create a **S4.3**-saturated Q_i-successor w_1 from it as usual, and put $w_0 \prec w_1$. There may be a choice of Q_i-successor since more than one of the X_j may be $CS4.3$-consistent but the choice is not important at this stage. Repeating this step for w_1 will give a $CS4.3$-consistent set w_2 and so on giving a sequence $w_0 \prec w_1 \prec w_2 \cdots$ that will either terminate or will sooner or later repeat some set w_m since $X^*_{S4.3}$ is finite. If the sequence terminates, with no cycles, then we are done since all the eventualities in w_0 are fulfilled, so go to Step 4. Otherwise, in the cyclic case, this will give $S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec w_m \prec w_{m+1} \prec \cdots \prec w_{n-1} \prec w_m$, containing a cycle $C = w_m \prec w_{m+1} \prec \cdots \prec w_{n-1} \prec w_m$ which we write pictorially as

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec \overline{w_m \prec w_{m+1} \cdots \prec w_{n-1}}.$$

The cycle C fulfills at least one of the eventualities in w_{n-1}, namely the $\neg\Box Q$ that gave the duplicated Q-successor w_m of w_{n-1}. But C may not fulfill *all* the eventualities in w_{n-1}. To check this, do Step 2.

Step 2: Let $Y = \{P \mid \neg\Box P \in w_{n-1} \text{ and } \neg P \notin w_j,\ m \leq j \leq n-1\}$, so that $\neg\Box Y$ is the set of eventualities in w_{n-1} that remain unfulfilled by C. If Y is empty, then go to Step 4 since S is now almost an **S4.3**-frame, otherwise go on with Step 3 below.

[1]Note that we choose only one successor, not all successors as in the proof for $CS4$. This is because, for $CS4.3$, we want to construct a linear sequence, not a tree. Also, we can guarantee only that at least one of the X_j is $CS4.3$-consistent whereas for $CS4$ each successor was independently $CS4$-consistent.

Step 3: So $\neg\Box Y = \neg\Box\{P_1, \cdots, P_k\}$, where $k \geq 1$. Put $w' = w_{n-1}^{\Box}$. Since $\Box w' \cup \neg\Box Y \subseteq w_{n-1}$ and w_{n-1} is $\mathcal{C}S4.3$-consistent so is at least *one* of

$$X_j = \Box w' \cup \{\neg P_j\} \cup \neg\Box\overline{Y_j}, \text{ for } j = 1, \cdots, k$$

by (θ) and $(S4.3)$. As before, choose *one* such $\mathcal{C}S4.3$-consistent X_i and create one **S4.3**-saturated P_i-successor for w_{n-1}. This successor is either new to S or not new to S.

(3a): If the P_i-successor is new to S then put $w_{n-1} \prec w_n$ giving

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec \overline{w_m \prec w_{m+1} \cdots w_{n-1}} \prec w_n$$

where w_n is this new P_i-successor and the overlining indicates the link $w_{n-1} \prec w_m$ marking the extent of C. Now proceed to Step 1 using w_n instead of w_0 to extend the sequence S.

(3b): Otherwise, the P_i-successor must duplicate some member $w_{m'}$ of S with $\neg P_i \in w_{m'}$. By the definition of Y, this successor $w_{m'}$ cannot be in C, otherwise C would fulfill the eventuality $\neg\Box P_i$. So $w_{m'}$ must precede w_m in the sequence. Replace the link $w_{n-1} \prec w_m$ by $w_{n-1} \prec w_{m'}$ thus extending C towards w_0 so that

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec \overline{w_{m'} \prec w_{m'+1} \prec \cdots \prec w_m \prec w_{m+1} \prec \cdots \prec w_{n-1}}.$$

Now go to Step 2 using the new C to calculate Y. That is, put $Y = \{P \mid \neg\Box P \in w_{n-1} \text{ and } \neg P \notin w_j,\ m' \leq j \leq n-1\}$.

Step 3 either extends the cycle C to include sets earlier in the sequence S by finding smaller values of m' etc.; or leaves C intact and extends the sequence S itself to include a new set w_n after w_{n-1}. In the first case, the extended C fulfills an extra eventuality in w_{n-1}, thus reducing the size of Y, but w_{n-1} remains the last set in S. In the latter case, the new set w_n fulfills an eventuality of w_{n-1} that C does not fulfill, and w_n becomes the last set in S. Attention turns to the fulfillment of the eventualities of w_n but note that the eventualities in w_{n-1} that are unfulfilled by C and w_n, are now in w_n by virtue of the fact that $\neg\Box\overline{Y_j} = \neg\Box Y \setminus \{\neg\Box P_j\}$ is a subset of w_n.

A loop involving Steps 2 and 3 extends C towards w_0. On the other hand, Step 1 applied to w_n extends S further or creates another cycle C' in S originating from w_n.

This process must terminate because $X^*_{S4.3}$ is finite. That is, eventually, in Step 3, only the second alternative (3b) will remain open because no new successors will be possible. The resulting loop involving Steps 2 and 3 will extend the corresponding C'' towards w_0 until C'' fulfills all the eventualities in the last set of S.

Either way, the resulting sequence S either contains no proper clusters, or is a finite sequence of finite, possibly overlapping, proper clusters (allowing S to be a single cluster). The overlapping clusters merge in Step 4 below.

Step 4: Let R be the reflexive and transitive closure of $\prec$ so that the overlapping clusters of $\prec$ become maximal disjoint clusters of R. Then $\langle W_0, R\rangle$ is a finite sequence of nondegenerate clusters that satisfies properties (i)-(iii) and hence is an **S4.3**-model graph for X. ●

Step 3(b′): Instead of Step 3(b), sometimes an **alternative argument** also works. Suppose w_l is the last set in S. Choose a $\mathcal{C}S4.3$-saturated P_i-successor for *each* unfulfilled eventuality $\neg\Box P_i$ of w_l since each $\Box w_l' \cup \{\neg\Box P_i\}$ is **S4.3**-consistent, as always, where $w_l' = w_l^{\Box}$. If *each* of these P_i-successors already appears in S, simply choose the P_i-successor w_x that appears earliest in S and put $w_l \prec w_x$, creating a cycle C'' which *must* fulfill all the eventualities in w_l by the choice of w_x.

6.1.4 Peculiarities of $\mathcal{C}S4.3$

For convenience we repeat the $(S4.3)$ rule:

$$(S4.3) \quad \frac{\Box X; \neg\Box\{P_1, \cdots, P_k\}}{\Box X; \neg\Box\overline{Y_1}; \neg P_1 \quad | \quad \cdots \quad | \quad \Box X; \neg\Box\overline{Y_k}; \neg P_k}$$

$$\text{where } Y = \{P_1, \cdots, P_k\} \text{ and } \overline{Y_i} = Y \setminus \{P_i\}$$

The $(S4.3)$ rule together with (θ) can simulate the $(S4)$ rule by thinning the current formula set to match a numerator where $k = 1$ and thereby making Y a singleton. Hence $\mathcal{C}S4.3$ is able to prove all **S4**-theorems which is just as well since **S4** $\subset$ **S4.3**.

The $(S4.3)$ rule is the only rule we have encountered where more than one eventuality plays an active role in any one rule application. That is, if $\neg\Box Z = \{\neg\Box P_1, \cdots, \neg\Box P_m\}$ are all the eventualities in the current tableau proof node, then by appropriate uses of (θ) we may choose Y to be *any* non-empty subset of Z. Consequently, $(S4.3)$ has no dual rule. In all the previous tableau systems we have seen, (θ) is used to leave only a single eventuality in the numerator so that the appropriate transitional rule is applicable. This is a direct consequence of the fact that, up till now, we have dealt with frames which are trees. For **S4.3**, this is no longer possible since we require a linear frame.

The $(S4.3)$ rule can be seen as a collection of rules, one for each value of $k = 1, 2, \cdots$ As a consequence, there is no easy way to deal with infinite sets in $\mathcal{C}S4.3$. In fact, **S4.3** is not compact, and the natural generalisation of the $(S4.3)$ rule is the (ω) rule which has an infinite number of denominators. See Sundholm [Sun77] for a description of the (ω) rule.

It appears that (θ) is essential for completeness of $\mathcal{C}S4.3$ since Step 2 requires us to ignore the eventualities of w_{n-1} that are fulfilled by the cycle C. And this is only possible if we know that a $\mathcal{C}S4.3$-tableau has been tried where these eventualities are ignored, and that it has failed to close. That is, in Step 2, we must be able to throw away some of the eventualities in w_{n-1} using (θ) and this means that (θ) is now an essential rule of $\mathcal{C}S4.3$.

In Section 4.5 (page 62) and Section 5.7 (page 93) we saw that (θ) could be eliminated by building the effects of (θ) into the transitional rules and by changing the basic axiomatic tableau rule from (0) to (0′). It may be possible to eliminate thinning by explicitly taking subsets as explained below.

There is another way to handle weak-connected frames as espoused by the cut-free sequent system for **KGL** due to Valentini [Val86] (although he uses the name $\mathbf{GL_{lin}}$ for **KGL** and uses the name **GL** for **G**). The following ($S4.3'$) rule can be obtained as an analogue of Valentini's rule.

$$(S4.3') \quad \frac{\Box X; \neg\Box Y}{\Box X; \neg Y_1; \neg\Box Y_1' \quad | \quad \cdots \quad | \quad \Box X; \neg Y_m; \neg\Box Y_k'}$$

where

$Y = \{P_1, \cdots, P_k\}$

$k \geq 1$

$Y_1, \cdots, Y_m$ is an enumeration of the non-empty subsets of Y

$m = 2^k - 1$ and

$Y_i' = Y - Y_i$

This rule makes the use of all subsets explicit, whereas we achieve the same effect via (θ). Rautenberg also mentions a similar rule but not in the context of **S4.3**. It would be interesting to analyze the relative complexities of these two rules as this is the form of the linearity rule that is required when reflexivity is missing. That is, when L is an axiom but 3 is not. It seems reasonable to conjecture that a rule similar to this one is the one required to give cut-free tableau and sequent systems for the logic **K4DLZ** characterised by finite linear sequences of degenerate clusters [Gol87]. I simply have not had time to formalise these systems and their soundness and completeness proofs.

Incidentally, the fact that **S4.3** is a subset of discrete linear temporal logics may explain why Wolper's [Wol83] method also requires an explicit check to ensure that all eventualities are fulfilled.

6.2 A Cut-free Tableau System for S4.3.1

The logic **S4.3.1** is characterised by the class of (frames which are) finite linear sequences of nondegenerate clusters with no proper non-final clusters; see page 33. It can be shown that **S4.3.1** is also characterised by the single frame $\langle \omega, \leq \rangle$ where ω is the set of natural numbers [Gol87, page 59]. Goldblatt [Gol87] shows that every **S4.3.1**-frame is a p-morphic image of $\langle \omega, \leq \rangle$ so that

$$\langle \omega, \leq \rangle \models A \text{ iff } \vdash_{S4.3.1} A.$$

Hence, between any two worlds there is always a finite number (possibly none) of other worlds and **S4.3.1**-frames are said to be discrete. So, although we always work with finite sequences of clusters, **S4.3.1** is the logic where time is modelled as the natural number line.

The history of **S4.3.1** goes back via Dummett and Lemmon [DL59] to Prior [Pri57]; see [HC68]. Prior had sought to model time as a linear sequence and conjectured that **S4** was this logic although it was soon proved that this was incorrect. Bull [Bul65] proved the completeness result for **S4.3** with respect to dense rather than discrete linear frames and also proved the completeness result for **S4.3.1** with respect to discrete frames although he calls this logic "**D**" for "the Diodorian logic". This use of "**D**" has nothing to do with the logic **KD** which we refer to as **D**.

The logic **S4.3.1** is also known as **S4.3Dum**, in honour of Michael Dummett [Seg71], but the credit for axiomatising it appears to be Kripke's [Bul65]. In fact, Bull [Bul65] claims that Kripke used semantic tableau for **S4.3.1**, in 1963, but he gives no reference and subsequent texts that use semantic tableau do not mention this work [Zem73]. Presumably, Kripke used semantic tableau based on cyclic graphs like his tableau systems for **S5** rather than tree tableau as we do; see Chapter 7.

Bull [Bul85] states that

> Zeman's **Modal Logic** (XLII 581), gives tableau systems for **S4.3** and **D** in its Chapter 15, ...

The **D** mentioned by Bull is **S4.3.1** but Zeman [Zem73, page 245] merely shows that his tableau procedure for **S4.3** goes into unavoidable cycles when attempting to prove *Dum*. Zeman does not investigate remedies and consequently does *not* give a tableau system for **S4.3.1** as claimed by Bull.

6.2.1 Tableau Rules for $\mathcal{C}S4.3.1$

The tableau system $\mathcal{C}S4.3.1$ is :

$$\mathcal{C}S4.3.1 = \{(0), (\neg), (\wedge), (\vee), (T)\} \cup \{(S4.3), (S4.3.1)\} \cup \{(\theta)\}$$

where $(S4.3.1)$ is

$$(S4.3.1) \quad \frac{U; \Box X; \neg\Box\{Q_1, \cdots, Q_k\}}{S_1 \quad | S_2 | \quad \cdots \quad | \quad S_k \quad | S_{k+1} | \quad S_{k+2} \quad | \quad \cdots \quad | \quad S_{2k}}$$

where

$$Y = \{Q_1, \cdots, Q_k\};$$

$$\overline{Y_j} = Y \setminus \{Q_j\};$$

$$S_j = U; \Box X; \neg\Box\overline{Y_j}; \Box\neg\Box Q_j$$

$$S_{k+j} = \Box X; \neg Q_j; \Box(Q_j \Rightarrow \Box Q_j); \neg\Box\overline{Y_j}$$

for $1 \leq j \leq k$

and the three sets represent the static, transitional and structural rules of $\mathcal{C}S4.3.1$.

Note that the denominators $S_1, \cdots, S_k$ are the same as the numerator except that $\neg\Box Q_j$ has been lifted to $\Box\neg\Box Q_j$ similar to the (5) rule in $\mathcal{C}S5$. The denominators $S_{k+1}, \cdots, S_{2k}$, however, do not carry U since they involve a "jump" to the first world that fulfills $\neg\Box Q_j$.

Semantic Intuitions: Since *Dum*, that is,

$$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow (\Diamond\Box P \Rightarrow \Box P)$$

is an **S4.3.1** theorem, so are each of:

$$\begin{aligned}
\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \wedge \Diamond\Box P &\Rightarrow \Box P \\
\neg\Box P &\Rightarrow \neg\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \vee \neg\Diamond\Box P \\
\neg\Box P &\Rightarrow \Diamond(\Box(P \Rightarrow \Box P) \wedge \neg P) \vee \Box\neg\Box P.
\end{aligned}$$

Suppose w satisfies the two formulae $\neg\Box P$ and $\neg\Box Q$ and consider the instance of ($S4.3.1$) where Y contains only two formulae P and Q:

$$\frac{U; \Box X; \neg\Box P; \neg\Box Q}{S_1 \quad | \; S_2 \quad | \; S_{2+1} \quad | \; S_4}$$

where

$$\begin{aligned} S_1 &= U; \Box X; \Box\neg\Box P; \neg\Box Q \\ S_2 &= U; \Box X; \neg\Box P; \Box\neg\Box Q \\ S_{2+1} &= \Box X; \neg\Box Q; \Box(P \Rightarrow \Box P); \neg P \\ S_{2+2} &= \Box X; \neg\Box P; \Box(Q \Rightarrow \Box Q); \neg Q. \end{aligned}$$

There are four ways of pairing the disjuncts of the two implications

$$\neg\Box P \Rightarrow \Diamond(\Box(P \Rightarrow \Box P) \land \neg P) \lor \Box\neg\Box P$$

and

$$\neg\Box Q \Rightarrow \Diamond(\Box(Q \Rightarrow \Box Q) \land \neg Q) \lor \Box\neg\Box Q$$

namely;

(a) $\Diamond(\Box(P \Rightarrow \Box P) \land \neg P) \land \Box\neg\Box Q$

(b) $\Diamond(\Box(P \Rightarrow \Box P) \land \neg P) \land \Diamond(\Box(Q \Rightarrow \Box Q) \land \neg Q)$

(c) $\Box\neg\Box P \land \Diamond(\Box(Q \Rightarrow \Box Q) \land \neg Q)$

(d) $\Box\neg\Box P \land \Box\neg\Box Q$.

The children S_1 and S_2 of the corresponding ($S4.3.1$) rule handle the three separate cases where at least one of $\Box\neg\Box P$ or $\Box\neg\Box Q$ are involved; that is, cases (a), (c) and (d). These cases involve no "jump".

The children S_{2+1} and S_{2+2} handle the final case (b) involving both $\Box(P \Rightarrow \Box P) \land \neg P$ and $\Box(Q \Rightarrow \Box Q) \land \neg Q$. These events may happen in any order, so the two children S_{2+1} and S_{2+2} cater to these two orderings in the same way as the ($S4.3$) rule.

The fact that S_{2+1} and S_{2+2} are not couched in terms of *both* $\Box(P \Rightarrow \Box P) \land \neg P$ *and* $\Box(Q \Rightarrow \Box Q) \land \neg Q$ can be explained by an argument based on which of them occurs *first* since this notion is well-defined for **S4.3.1**. That is, if $\Box(P \Rightarrow \Box P) \land \neg P$ happens first then we know that $\neg\Box Q$ is true at the world where this happens. If $\Box(Q \Rightarrow \Box Q) \land \neg Q$ happens first then we know that $\neg\Box P$ is true at the world where this happens. If they happen simultaneously then we are still safe because $\neg Q$ implies $\neg\Box Q$ and $\neg P$ implies $\neg\Box P$ due to reflexivity so that both S_{2+1} and S_{2+2} suffice.

The $(S4.3.1)$ rule has a sequent analogue as shown below:

$$\frac{S_1 \quad S_2 \quad \cdots \quad S_k \quad S_{k+1} \quad S_{k+2} \quad \cdots \quad S_{2k}}{\Sigma,\ \Box\Gamma \longrightarrow \Box A_1, \cdots, \Box A_k,\ \Delta}\ (\rightarrow \Box : S4.3.1)$$

where for $1 \leq i \leq k$

$$Y = \{A_1, \cdots, A_k\}$$

$$\overline{Y_i} = Y \setminus \{A_i\}$$

$$S_i = \Sigma,\ \Box\Gamma \longrightarrow \Diamond\Box A_i,\ \Box\overline{Y_i},\ \Delta$$

$$S_{k+i} = \Box\Gamma,\ \Box(A_i \Rightarrow \Box A_i) \longrightarrow A_i,\ \Box\overline{Y_i}$$

For a finite set X let $X^*_{S4.3.1} = Sf\, \Box(\widetilde{X} \Rightarrow \Box\widetilde{X})$ where $\Box(\widetilde{X} \Rightarrow \Box\widetilde{X}) = \{\Box(P \Rightarrow \Box P) \mid P \in \widetilde{X}\}$.

Lemma 10 *If there is a closed $CS4.3.1$ tableau for X then there is a closed $CS4.3.1$ tableau for X with all nodes in the finite set $X^*_{S4.3.1}$.*

Proof: Obvious from the fact that all rules for $CS4.3.1$ operate with subsets of $X^*_{S4.3.1}$ only. •

A set X is $CS4.3.1$-**saturated** if it is $CS4.3.1$-consistent and closed with respect to the static rules of $CS4.3.1$.

Lemma 11 *For each $CS4.3.1$-consistent X there is an effective procedure to construct some finite $CS4.3.1$-saturated X^* with $X \subseteq X^* \subseteq X^*_{S4.3.1}$.*

Proof: As for $CS4$ since the static rules for $CS4.3.1$ and $CS4$ are identical. •

6.2.2 Soundness of $CS4.3.1$

Theorem 17 *The tableau system $CS4.3.1$ is sound with respect to the class of finite sequences of nondegenerate clusters with no proper non-final clusters, that is, with respect to* **S4.3.1**-*frames.*

The static and structural rules of $CS4.3.1$ are those of $CS4$ and **S4.3.1**-frames are also reflexive and transitive. Thus all the proofs of soundness for these rules are as for $CS4$ since they depend only on the reflexivity and transitivity of R. Every **S4.3.1**-frame is an **S4.3**-frame, hence the $(S4.3)$ rule is also sound with respect to **S4.3.1**-frames. The soundness of the remaining rule is proved below.

Proof for $(S4.3.1)$: We have to show that if the numerator of the $(S4.3.1)$ rule is **S4.3.1**-satisfiable, then so is at least one of the denominators. That is, we have to show that if

$$U; \Box X; \neg\Box\{Q_1, \cdots, Q_k\}$$

is **S4.3.1**-satisfiable and $Y = \{Q_1, \cdots, Q_k\}$ for finite $k > 0$, then there is at least one i with $1 \leq i \leq k$ such that

$$U; \Box X; \Box\neg\Box Q_i; \neg\Box\overline{Y_i} \quad or \quad \Box X; \neg\Box\overline{Y_i}; \neg Q_i; \Box(Q_i \Rightarrow \Box Q_i)$$

is **S4.3.1**-satisfiable. Although finiteness is used in the proof it is not essential as the proofs can also be carried through without assuming finiteness.

As before assume that $U; \Box X; \neg\Box\{Q_1, \cdots, Q_k\}$ is **S4.3.1**-satisfiable. That is, there is an **S4.3.1**-model $\mathcal{M} = \langle W, R, V\rangle$ with $w_0 \in W$ such that

$$w_0 \models U; \Box X; \neg\Box\{Q_1, \cdots, Q_k\}.$$

We know that $\mathcal{M}$ is a finite linear sequence of simple clusters with one final, possibly proper, cluster. Let C_0 denote the cluster containing w_0. We can write $\mathcal{M}$ as

$$\lhd \cdots \lhd C_0 \lhd C_1 \lhd \cdots \lhd C_n$$

There are two cases depending on whether C_0 is a final cluster or not.

Case 1: C_0 is the final cluster, that is, $n = 0$. We know $w_0 \in C_0$ and $w_0 \models U; \Box X; \neg\Box\{Q_1, \cdots, Q_k\}$. Therefore, there exist $w_1, w_2, \cdots, w_k$ not necessarily distinct worlds in W such that for each world w_j we have $w_0 R w_j$ and $w_j \models \neg Q_j$. Since C_0 is the final cluster, these worlds are all in C_0 and are mutually accessible by R which is reflexive, transitive and symmetric over C_0. That is, we have $w_j \models \neg Q_j; \neg\Box\overline{Y_j}$. But since R is reflexive, $w_j \models \neg Q_j$ implies $w_j \models \neg\Box Q_j$ and in a (proper or simple) *final* cluster, $w_j \models \neg\Box Q_j$ implies $w_j \models \Box\neg\Box Q_j$. In a nondegenerate cluster, we also have both $w_0 R w_j$ and $w_j R w_0$ giving $w_0 \models U; \Box X; \neg\Box\overline{Y_j}; \Box\neg\Box Q_j$ and we are done.

Case 2: C_0 is not the final cluster, hence it is simple. Suppose further that

$$(**) \qquad \text{for all } i,\ 1 \leq i \leq k, w_0 \not\models U; \Box X; \Box\neg\Box Q_i; \neg\Box\overline{Y_i};$$

as otherwise, we would be done.

Since R is reflexive and $w_0 \models U; \Box X; \neg\Box\{Q_1, \cdots, Q_k\}$ and we are assuming that $(**)$ holds, it must be that $w_0 \not\models \Box\neg\Box Q_i$ for all $1 \leq i \leq k$. But then $w_0 \models \neg\Box\neg\Box Q_i$, that is, $w_0 \models \Diamond\Box Q_i$ for all $1 \leq i \leq k$. Thus there exist $w_1, \cdots, w_k$ not necessarily distinct worlds in W such that $w_0 R w_i$ and $w_i \models \Box Q_i$ for $1 \leq i \leq k$. Since $w_0 \models \neg\Box\{Q_1, \cdots, Q_k\}$ we must have $w_0 \lhd w_i$ so that the w_i occur *strictly* after w_0 (which is fine since w_0 is a member of a non-final cluster).

Let w_f be the first of these worlds that fulfills any of the eventualities $E = \{\Diamond\Box Q_1, \cdots, \Diamond\Box Q_k\}$. That is, $w_f \models \Box Q_m$ for some $1 \leq m \leq k$. If $w_f \in C_n$ then consider c_{n-1}, the unique and simple immediate predecessor of C_n. By definition of w_f as "first", c_{n-1} cannot fulfill any of the eventualities in E hence $c_{n-1} \models \neg\Box\overline{Y_m}$. It also cannot satisfy Q_m as then c_{n-1} would be "first" instead. Hence, $c_{n-1} \models \neg\Box\overline{Y_m}; \neg Q_m; \Box(Q_m \Rightarrow \Box Q_m)$ and we are done. If $w_f \notin C_n$, then w_f is a simple non-final cluster and has a unique immediate predecessor w_{f-1} such that $w_{f-1} \models \Box X; \neg\Box\overline{Y_m}; \neg Q_m; \Box(Q_m \Rightarrow \Box Q_m)$ and again we are done. Note that we are guaranteed that w_{f-1} exists because we know that the w_i occur strictly after w_0. •

6.2.3 Completeness of $\mathcal{C}S4.3.1$

Theorem 18 *If X is a finite set of formulae and X is $\mathcal{C}S4.3.1$-consistent then there is an* **S4.3.1**-*model for X on a finite frame $\langle W_0, R\rangle$ which is a finite (reflexive and transitive) sequence of nondegenerate clusters with no non-final proper clusters (and hence an* **S4.3.1**-*frame).*

Proof Outline: The completeness proof of $\mathcal{C}S4.3.1$ is similar to the completeness proof for $\mathcal{C}S4.3$ and $\mathcal{C}S4$. The difference from $\mathcal{C}S4$ is that only *one* branch is constructed, and that in doing so, the $(S4.3)$ rule is used instead of the $(S4)$ rule. But in $\mathcal{C}S4.3$ we are allowed to have cycles, whereas in $\mathcal{C}S4.3.1$ we are allowed only *one* final cycle (if any). The basic idea is again to follow one branch until it cycles and then to use thinning to sprout a continuation of the branch, thus escaping out of the cycle. But this time we use the $(S4.3.1)$ rule to avoid the cycle. At all times, we ignore the eventualities that are invariant; that is, we ignore $\neg\Box P$ if $\Box\neg\Box P$ is also present since the $(S4.3)$ and $(S4.3.1)$ rules will carry such eventualities into the successor they spawn. In this way, invariants accumulate until the final, possibly proper, cluster. We then fulfill them in this final cluster. The complications arise because although we have categorised $(S4.3.1)$ as a transitional rule, only the S_{k+j} branches involve a "jump". The S_k branches are really static since all they do is to lift the $\neg\Box P$ to $\Box\neg\Box P$ as in the (5) rule. Thinning seems essential.

Proof: Let $w_0 \supseteq X$ be $\mathcal{C}S4.3.1$-saturated, $w_0 \subseteq X^*_{S4.3.1}$. If no $\neg\Box P$ occurs in w_0 then $\langle\{w_0\}, \{(w_0, w_0)\}\rangle$ is the desired model graph since (i)-(iii) are satisfied. Otherwise do Step 1 below.

Step 1: Let $Y = \{Q \mid \neg\Box Q \in w_0 \text{ and } \neg Q \notin w_0 \text{ and } \Box\neg\Box Q \notin w_0\}$.

If $Y = \emptyset$ go to Step 3.

Otherwise, $\neg\Box Y = \neg\Box\{Q_1, Q_2, \cdots, Q_m\}, m \geq 1$. Put $w' = w_0^{\Box}$. Now $\Box w' \cup \neg\Box Y \subseteq w_0$, so $\Box w' \cup \neg\Box Y$ is $\mathcal{C}S4.3.1$-consistent by (θ); hence so is at least one of

$$X_j = \Box w' \cup \{\neg Q_j\} \cup \neg\Box\overline{Y_j}, \text{ for } j = 1, \cdots, m$$

by $(S4.3)$.

Choose one such $\mathcal{C}S4.3.1$-consistent set X_i and create a $\mathcal{C}S4.3.1$-consistent Q_i-successor w_1 from it as usual, and put $w_0 \prec w_1$ as long as w_1 is new to the sequence. We may have a choice of Q_i-successor since more than one of the X_j may be $\mathcal{C}S4.3.1$-consistent but the choice is not important as long as the chosen successor is new to the sequence. If possible, always choose the Q_i-successor that is new to S.

Repeating this step for w_1 will give a $\mathcal{C}S4.3.1$-consistent set w_2 and so on giving a sequence $S = w_0 \prec w_1 \prec w_2 \cdots$ which will either terminate or will sooner or later repeat some set since $X^*_{S4.3.1}$ is finite. Continue extending S with new sets until this is no longer possible either because all choices of Q_i-successor would lead to a cycle in S or because the corresponding Y becomes empty.

If $Y = \emptyset$ go to Step 3, otherwise go to Step 2.

Step 2: Thus, in general,

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec w_{n-1}$$

where all potential $(S4.3)$ (yes $(S4.3)$) Q-successors of w_{n-1} are not new to the sequence and

$$Y = \{Q \mid \neg\Box Q \in w_{n-1} \text{ and } \neg Q \notin w_{n-1} \text{ and } \Box\neg\Box Q \notin w_{n-1}\} = \{Q_1, \cdots, Q_k\}$$

is not empty.

(2a) Now at this point it is possible that the alternate argument of the **S4.3** proof will work. That is, put $w'_{n-1} = w_{n-1}^{\Box}$ and note that $\Box w'_{n-1} \cup \{\neg\Box P\}$ is $\mathcal{C}S4.3.1$-consistent by (θ) where $\neg\Box P$ is *any* unfulfilled eventuality of w_{n-1}. Hence so is *each* $\Box w'_{n-1} \cup \{\neg P\}$ by $(S4.3)$ (not $(S4.3.1)$). So we can create a P-successor for each of these sets and if each of these successors already appears in S then simply chose the one closest to w_0 in the $\prec$ ordering, thus creating a final cycle C that fulfills all the eventualities of w_{n-1}, and go to End. Otherwise do Step (2b).

(2b) Note that $\neg\Box Q \in \neg\Box Y$ implies $(\forall w_i \in S, \Box\neg\Box Q \notin w_i)$. To see this let $w'_i = w_i^{\Box}$ and note that by construction $\Box w'_0 \subseteq \Box w'_1 \subseteq \Box w'_2 \subseteq \cdots \subseteq \Box w'_{n-2} \subseteq \Box w'_{n-1}$ and $\Box\neg\Box Q \notin \Box w'_{n-1}$ by definition of Y.

Since $\neg\Box Y \cup \Box w'_{n-1} \subseteq w_{n-1}$ and w_{n-1} is $\mathcal{C}S4.3.1$-consistent, so is one of

$$X_j = \{\Box\neg\Box Q_j\} \cup w_{n-1} \setminus \{\neg\Box Q_j\}$$

or one of

$$Z_j = \{\Box(Q_j \Rightarrow \Box Q_j), \neg Q_j\} \cup \Box w'_{n-1} \cup \neg\Box\overline{Y_j}$$

where $i \leq j \leq k$ by (θ) and ($S4.3.1$).

If some Z_i is $\mathcal{C}S4.3.1$-consistent then create a Q_i-successor w_n from Z_i. Now if w_n duplicates some member of S, where

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec w_{n-2} \prec w_{n-1}$$

then w_{n-1} contains $\Box(Q_i \Rightarrow \Box Q_i)$ and by (T) also contains $Q_i \Rightarrow \Box Q_i$. Therefore, $\neg(Q_i \wedge \neg\Box Q_i) \in w_{n-1}$ since $(Q_i \Rightarrow \Box Q_i) = \neg(Q_i \wedge \neg\Box Q_i)$. By ($\vee$), $\neg Q_i \in w_{n-1}$ or $\neg\neg\Box Q_i \in w_{n-1}$. By definition of Y, $\neg Q_i \notin w_{n-1}$ hence $\neg\neg\Box Q_i \in w_{n-1}$. But then w_{n-1} contains both $\neg\neg\Box Q_i$ and $\neg\Box Q_i$ contradicting its $\mathcal{C}S4.3.1$-consistency. Hence, w_n must be new to the sequence S, so put $w_{n-1} \prec w_n$ giving

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec w_{n-1} \prec w_n.$$

Now go to Step 1 using w_n instead of w_0 to calculate Y.

Else, some X_i must be $\mathcal{C}S4.3.1$-consistent, so construct the corresponding $\mathcal{C}S4.3.1$-saturated Q_i-successor v_{n-1} and replace the link $w_{n-2} \prec w_{n-1}$ with $w_{n-2} \prec v_{n-1}$, thus discarding w_{n-1} altogether and giving

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec w_{n-2} \prec v_{n-1}.$$

Condition (2b) guarantees that v_{n-1} does not duplicate any set in S since $\Box\neg\Box Q_i \in v_{n-1}$ for the $\neg\Box Q_i \in \neg\Box Y$ that gives rise to v_{n-1}. Also note that by (T), $\Box\neg\Box Q_i \in v_{n-1}$ implies $\neg\Box Q_i \in v_{n-1}$ so that $w_{n-1} \subseteq v_{n-1}$ and hence all the eventualities in w_{n-1} are also in v_{n-1}. That is, all we have done in replacing w_{n-1} with v_{n-1} is to lift some $\neg\Box Q$ to $\Box\neg\Box Q$. Now go to Step 1 using v_{n-1} instead of w_0 to calculate Y.

Step 3: If $Y = \{Q \mid \neg\Box Q \in w_{n-1}$ and $\neg Q \notin w_{n-1}$ and $\Box\neg\Box Q_i \notin w_{n-1}\}$ is empty, then for each $\neg\Box Q \in w_{n-1}$
(3a) $\neg Q \in w_{n-1}$ or
(3b) $\Box\neg\Box Q \in w_{n-1}$.

If (3a) is true for all eventualities in w_{n-1} or if w_{n-1} contains no eventualities then w_{n-1} fulfills all its eventualities; so go to End.

Otherwise (3b) *alone* must hold for at least one eventuality in w_{n-1}. Let

$$Z = \{P \mid \neg\Box P \in w_{n-1} \text{ and } \neg P \notin w_{n-1} \text{ and } \Box\neg\Box P \in w_{n-1}\}$$

and suppose $\neg\Box Z = \neg\Box\{P_1, \cdots, P_m\}$. These are *all* the unfulfilled eventualities of w_{n-1} since Y above is empty. We know that $m \geq 1$ since Z is non-empty. Put $w' = w^{\Box}_{n-1}$.

Since $\{\neg\Box P_i\} \cup \Box w' \subseteq w_{n-1}$ for *each* $1 \leq i \leq m$ and w_{n-1} is $\mathcal{C}S4.3.1$-consistent, so is *each* of

$$X_i = \{\neg P_i\} \cup \Box w' \text{ for } i = 1, \cdots, m$$

by (θ) and $(S4.3)$, (not $(S4.3.1)$) as always.[2] Furthermore, by (T), the P_i-successor obtained from X_i contains all the other unfulfilled eventualities $\neg\Box Z$ since $\Box\neg\Box Z \subseteq \Box w' \subseteq X_i$.

If *every* P_i-successor, $i = 1 \cdots m$, already occurs in S then choose the P_i-successor w_x that appears earliest in the sequence S and put $w_{n-1} \prec w_x$ giving a final *lone* cycle $C = w_x \prec w_{x+1} \prec \cdots \prec w_{n-1} \prec w_x$ in

$$S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec \overline{w_x \prec w_{x+1} \prec \cdots \prec w_{n-1}}.$$

By the choice of w_x, the cycle C fulfills all the eventualities of w_{n-1}.

Otherwise, it is possible to choose a P_i-successor w_n that is new to S, so do so, giving $S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec w_{n-1} \prec w_n$ and go to Step 1 using w_n instead of w_0 to calculate Y. Note that all the unfulfilled eventualities of w_{n-1} appear in w_n because $\neg\Box Z \subseteq w_n$, but one of them, namely $\neg\Box P_i$, does not appear in the $\neg\Box Y$ calculated from w_n in Step 1 because w_n fulfills it; that is, $\neg P_i \in w_n$.

The general control flow of the method is to repeat Step 1 until a cycle is imminent. Then do either Step 2 or Step 3, both of which avoid the cycle, when necessary, and return to Step 1. In both Step 2 and Step 3 there is a check to see if a lone final cycle suffices. If so then both steps terminate the process. Otherwise, each step, in its own way, generates a set w_n that is *new* to S and which fulfills an eventuality of w_{n-1} giving $S = w_0 \prec w_1 \prec w_2 \prec \cdots \prec w_{n-1} \prec w_n$. Sometimes, though, Step 2 does not lead to a new successor, and we are forced to assume that some eventuality $\neg\Box P$ is actually an invariant by lifting it to $\Box\neg\Box P$ via v_{n-1}. But the subsequent $\neg\Box Y$ will now exclude $\neg\Box P$ because $\Box\neg\Box P \in v_{n-1}$.

Since $X^*_{S4.3.1}$ is finite, this process must terminate since there are a finite number of different successors, there are a finite number of eventualities in each node and no rule introduces new eventualities.

End: Let R be the reflexive and transitive closure of $\prec$.

By construction, $\langle W_0, R\rangle$ is an **S4.3.1**-frame and properties (i)-(iii) hold. Hence, it is a model graph for X. ●

Note on $\mathcal{C}S4.3.1$ proof: It may seem as if the $(S4.3)$ rule may be eliminable from the $\mathcal{C}S4.3.1$, however this is not the case. The $(S4.3)$ rule is needed to handle the case where $\neg\Box P \in w_{n-1}$ implies $\Box\neg\Box P \in w_{n-1}$. This case is immune to the $(S4.3.1)$ rule since one of the S_i denominators of the $(S4.3.1)$ rule is already satisfied by w_{n-1}. Thus, in this case, the $(S4.3.1)$ rule achieves nothing.

---

[2] By this I mean that this is also the case in Steps 1 and 2, but it cannot be used there since the resulting P_i-successor is not guaranteed to contain all the other unfulfilled eventualities of w_{n-1}. Furthermore, blind use of this fine-grained approach leads to branching in S which is unwanted.

6.2.4 Peculiarities of $\mathcal{C}S4.3.1$

Since **S4.3** ⊂ **S4.3.1** it makes sense that $\mathcal{C}S4.3$ is a subset of $\mathcal{C}S4.3.1$.

The proof of soundness of the ($S4.3.1$) rule relies on the cut-like property that for any given eventuality $\neg\Box P$ and any given world w, either $w \models \Box\neg\Box P$ or $w \models \neg\Box\neg\Box P$. The former says that the eventuality $\neg\Box P$ is an invariant of the sequence of worlds in this model since it reappears incessantly. The latter is the same as $w \models \Diamond\Box P$ and says that eventually there is a point where the value of P settles to "true" forever. Thus, there are two eventualities that play a role in the ($S4.3.1$) rule. One is $\neg\Box P$ and is explicit since it appears in the numerator. The other is $\neg\Box\neg\Box P$ and is implicit.

But note that in the completeness proof, we never actually attempt to fulfill this second eventuality as our only requirement is to fulfill the explicit eventuality $\neg\Box P$. That is, the S_i denominators "assume" that $\Box\neg\Box P$ is true by lifting some $\neg\Box P$ to $\Box\neg\Box P$. The S_{k+i} denominators make the opposite assumption that $\Box\neg\Box P$ is false; that is that $\Diamond\Box P$ is true. But we cannot simply "lift" $\neg\Box P$ to $\neg\Box\neg\Box P$ for then the ($S4.3.1$) rule would no longer be "once off" as the eventuality $\neg\Box P$ would spawn the eventuality $\neg\Box\neg\Box P$ which could then spawn another eventuality $\neg\Box\neg\Box\neg\Box P$ and so on. That is, ($S4.3.1$) would no longer be analytic because the set of superformulae would fall outside $X^*_{S4.3.1}$. The S_{k+i} denominators therefore "jump" to the world immediately preceding the world where $\Box P$ become true. At this world, we know that $\neg P$ holds, and we know that $\Box(P \Rightarrow \Box P)$ also holds.

Since ($S4.3$) is a part of $\mathcal{C}S4.3.1$, the peculiarities of $\mathcal{C}S4.3$ are also present in $\mathcal{C}S4.3.1$ and thinning seems essential.

Finally, note that the completeness proof depends critically on determining whether some complex formula is already in a set w. That is, if $\neg\Box P \in w$ then we have to determine whether $\Box\neg\Box P \in w$, and this is only possible because we use primitive notation. So the deterministic decision procedure based on the completeness proof may require that all formulae be in negated normal form.

6.3 A Cut-free Tableau System for S4.14

In the last two sections we have seen the temporal logics corresponding to linear dense and linear discrete frames. The bulldozing technique of Segerberg [Seg71] [HC84] can be used to show that **S4** is the logic of branching dense frames. So what is the logic of branching discrete frames ?

The axiomatic system **S4.14** (axiomatised as **KT4Zbr**) is proposed by Zeman [Zem73, page 249] as the temporal logic for branching integer time. That is, imagine the points of time to have a tree structure starting from some root node corresponding to "now" and branching out into the "future". There is no past "before" the root and each branch is a linear sequence of points isomorphic to the natural numbers 1, 2, 3, $\cdots$ This means that the logic **S4.14** is characterised by the class of (frames which are) finite trees where every branch is a finite linear sequence of nondegenerate clusters with no proper non-final clusters although Zeman does not seem to realise that finiteness is essential; see page 33. **S4.14**-frames are also discrete and by bulldozing the final cluster each branch gives rise to an infinite sequence of points analogous to the natural number line. The name **S4.14** is due to Zeman.

6.3.1 Tableau Rules for $\mathcal{C}S4.14$

The tableau system $\mathcal{C}S4.14$ is:

$$\mathcal{C}S4.14 = \{(0), (\neg), (\wedge), (\vee), (T)\} \cup \{(S4), (S4.14)\} \cup \{(\theta)\}$$

where $(S4.14)$ is:

$$(S4.14)\ \frac{\Box X; \neg\Box P}{\Box X; \Box\neg\Box P \quad | \quad \Box X; \neg P; \Box(P \Rightarrow \Box P)}$$

and the three sets are respectively the static, transitional and structural rules for $\mathcal{C}S4.14$.

Semantic Intuitions: The intuitions behind the $(S4.14)$ rule are based on the axiom Zbr as follows. Since Zbr, that is,

$$\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \Rightarrow (\Box\Diamond\Box A \Rightarrow \Box A)$$

is a theorem, so are:

$$\begin{aligned} \Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \wedge \Box\Diamond\Box A &\Rightarrow \Box A \\ \neg\Box A &\Rightarrow \neg\Box(\Box(A \Rightarrow \Box A) \Rightarrow A) \vee \neg\Box\Diamond\Box A \\ \neg\Box A &\Rightarrow \Diamond(\Box(A \Rightarrow \Box A) \wedge \neg A) \vee \Diamond\Box\neg\Box A. \end{aligned}$$

The left fork of the $(S4.14)$ rule is a jump to the world where $\Box\neg\Box A$ eventually becomes true and the right fork is a jump to the world where $\Box(A \Rightarrow \Box A) \wedge \neg A$ eventually becomes true.

The $(S4.14)$ rule has a sequent analogue as shown below:

$$\frac{\Box\Gamma \longrightarrow \Diamond\Box A \qquad \Box\Gamma,\ \Box(A \Rightarrow \Box A) \longrightarrow A}{\Box\Gamma \longrightarrow \Box A}\ (\rightarrow \Box : S4.14)$$

For a finite set X let $\Box(\widetilde{X} \Rightarrow \Box\widetilde{X})$ denote the set $\{\Box(P \Rightarrow \Box P) \mid P \in \widetilde{X}\}$ and let $X^*_{S4.14} = Sf\,\Box(\widetilde{X} \Rightarrow \Box\widetilde{X})$.

Lemma 12 *If there is a closed $\mathcal{C}S4.14$ tableau for X then there is a closed $\mathcal{C}S4.14$ tableau for X with all nodes in the* finite set $X^*_{S4.14}$.

Proof: Obvious from the fact that all rules for $\mathcal{C}S4.14$ operate with subsets of $X^*_{S4.14}$ only. $\bullet$

A set X is $\mathcal{C}S4.14$**-saturated** if it is $\mathcal{C}S4.14$-consistent and closed with respect to the static rules of $\mathcal{C}S4.14$.

Lemma 13 *For each $\mathcal{C}S4.14$-consistent X there is an effective procedure to construct some finite $\mathcal{C}S4.14$-saturated X^* with $X \subseteq X^* \subseteq X^*_{S4.14}$.*

Proof: As for $\mathcal{C}S4$ since the static rules for $\mathcal{C}S4.14$ and $\mathcal{C}S4$ are identical. $\bullet$

6.3.2 Soundness of $\mathcal{C}S4.14$

Theorem 19 *The system $\mathcal{C}S4.14$ is sound with respect to* **S4.14**-*frames.*

The rules of $\mathcal{C}S4.14$ are those of $\mathcal{C}S4$ plus $(S4.14)$ and **S4.14**-frames are also reflexive and transitive. Thus all the proofs of soundness for the $\mathcal{C}S4$ rules are as for $\mathcal{C}S4$ since they depend only on the reflexivity and transitivity of R. The soundness of the remaining rule, $(S4.14)$, is proved below.

Proof for $(S4.14)$**:** We have to show that if the numerator of the $(S4.14)$ rule is **S4.14**-satisfiable, then so is at least one of the denominators. That is, we have to show that if $\Box X; \neg\Box P$ is **S4.14**-satisfiable then one of $\Box X; \Box\neg\Box P$ and $\Box X; \Box(P \Rightarrow \Box P); \neg P$ is **S4.14**-satisfiable.

As before assume that $\Box X; \neg\Box P$ is **S4.14**-satisfiable. That is, there is an **S4.14**-model $\mathcal{M} = \langle W, R, V\rangle$ with $w_0 \in W$ such that $w_0 \models \Box X; \neg\Box P$. Also, W is finite and this is essential.

Suppose that there is some world $w \in W$ such that $w_0 R w$ and $w \models \Box\neg\Box P$. Then we are done since transitivity of R implies that $w \models \Box X$ giving $w \models \Box X; \Box\neg\Box P$.

Otherwise, suppose that there is no such world $w \in W$. That is, suppose that

$(*)$ $\forall w \in W, w_0 R w$ implies $w \not\models \Box\neg\Box P$.

This is just the same as: $\forall w \in W, w_0 R w$ implies $w \models \Diamond\Box P$, which by definition of $\models$ implies that

$$w_0 \models \Box\Diamond\Box P.$$

We know that an **S4.14**-model is a finite tree of nondegenerate clusters with no non-final proper clusters, and that $w_0 \in W$. So any particular branch is a linear sequence of simple (nondegenerate) clusters except for the final, possibly proper, cluster. Label the cluster containing w_0 as C_0. There are two cases depending on whether or not C_0 is a final cluster.

Case 1: Suppose C_0 is a final (and hence possibly proper) cluster and $w_0 \in C_0$. Since $w_0 \models \Box X; \neg\Box P$, there exists a world $w_1 \in C_0$ such that $w_0 R w_1$ and $w_1 \models \neg P$. Since $w_0 R w_1$ we know $w_1 \models \Box X$ as well. But in a (proper or simple reflexive, transitive and symmetric) *final* cluster, $w_1 \models \neg P$ implies $w_1 \models \neg\Box P$ implies $w_1 \models \Box\neg\Box P$. This contradicts (∗) and so Case 1 cannot occur.

Case 2: Then $w_0 \in C_0$ and C_0 is a non-final, and hence simple, cluster. Also, $w_0 \models \Box\Diamond\Box P$ by (∗). So choose any branch emanating from w_0 and let C_f be the first cluster on this branch that is after C_0 and that contains a world w_f such that $w_f \models \Box P$. Since $w_0 \models \neg\Box P$, we know that $C_f \neq C_0$, hence the "after" is justified. Consider the (unique) parent cluster C_{f-1} of C_f where C_{f-1} may be C_0 itself. Since C_{f-1} is non-final, it is simple and is just one reflexive world c_{f-1}.

Since $w_0 R c_{f-1}$, we know that $c_{f-1} \models X; \Box X; \Diamond\Box P; \Box\Diamond\Box P$ by reflexivity and transitivity of R. By choice of C_f as "first to satisfy $\Box P$" we know that $c_{f-1} \models \neg\Box P$. We do not know whether $c_{f-1} \models P$ or $c_{f-1} \models \neg P$ but this is not important yet. There are two subcases depending on whether c_{f-1} has other children (apart from C_f).

Case 2(a): If c_{f-1} has no other children (apart from C_f) then $c_{f-1} \models \Box(P \Rightarrow \Box P)$. In this subcase, by choice of C_f as "first to satisfy $\Box P$", we also know that $c_{f-1} \models \neg P$. But then $c_{f-1} \models \Box X; \Box(P \Rightarrow \Box P); \neg P$ and we are done.

Case 2(b): If c_{f-1} does have other children (apart from C_f) then we can choose any other branch emanating from c_{f-1} and let C'_f be the first cluster on this branch that is after c_{f-1} and that contains a world w'_f such that $w'_f \models \Box P$. Since $c_{f-1} \models \neg\Box P$, we know that $C'_f \neq c_{f-1}$, hence the "after" is justified. Consider the (unique and simple) parent cluster C'_{f-1} of C'_f where C'_{f-1} may be c_{f-1} itself. Since C'_{f-1} is non-final, it is simple, and so is just one reflexive world c'_{f-1}. But we have seen this situation before and by repeating the argument in the previous two paragraphs, we either end up at a cluster satisfying Case 2(a) or we go on for ever.

The infinite process either gives an infinite branch or gives a world with an infinite number of immediate children, both of which are impossible since W is finite. Hence sooner or later Case 2(a) must hold. •

6.3.3 Completeness of $\mathcal{C}S4.14$

Theorem 20 *If X is a finite set of formulae and X is $\mathcal{C}S4.14$-consistent then there is an* **S4.14***-model for X on a finite* **S4.14***-frame, where an* **S4.14***-frame is a finite tree of nondegenerate clusters with no non-final proper clusters.*

Recall that a formula $\neg\Box P$ is called an **eventuality** since it entails that eventually $\neg P$ must hold. A set w is said to fulfill an eventuality $\neg\Box P$ when $\neg P \in w$. A sequence $w_1 \prec w_2 \prec \cdots \prec w_m$ of sets is said to fulfill an eventuality $\neg\Box P$ when $\neg P \in w_i$ for some w_i in the sequence.

Proof sketch: For **S4.14** which is the basic discrete branching time logic, the model construction has two stages. In the first stage, a finite branching tree of nondegenerate clusters is created along the lines of the $\mathcal{C}S4$ model construction. In the second phase, each non-final proper cluster is flattened into an arbitrary sequence of its constituent worlds. Note that this has nothing to do with the bulldozing technique of Segerberg which is also used to flatten out proper clusters [Seg71, pages 80-81], [HC84, pages 82-88]. It may be an analogue of the "virtually last" argument of Segerberg [Seg71, pages 96] but I am not sure of this at the moment because an *arbitrary* sequence suffices for **S4.14**, so that no particular world has to be shown to be "virtually last".

Proof: Let $w_0 \supseteq X$ be $\mathcal{C}S4.14$-saturated, $w_0 \subseteq X^*_{S4.14}$. Construct a model graph from w_0 using the method for $\mathcal{C}S4$ except for one additional step. In general, when a Q_i-successor is created for $\neg\Box Q_i \in w$ based on the $(S4)$ rule, where $w' = w^{\Box}$, the additional rule $(S4.14)$ means that

(a) $\Box w' \cup \{\Box\neg\Box Q_i\}$ is $\mathcal{C}S4.14$-consistent or

(b) $\Box w' \cup \{\Box(Q_i \Rightarrow \Box Q_i), \neg Q_i\}$ is $\mathcal{C}S4.14$-consistent.

So each node can have a Q_i-successor due to $(S4)$ and at least one Q_i-successor due to $(S4.14)$. Note that the $(S4.14)$ rule denominators are not mutually exclusive so they can both be **S4.14**-consistent at the same time.

The construction still gives a preorder over $\prec$ as for $\mathcal{C}S4$ and each branch either terminates, or gives a cycle due to the finiteness of $X^*_{S4.14}$ by choosing the minimum i and j such that $w_i = w_j$, $i < j$ and putting $w_{j-1} \prec w_i$.

As for $\mathcal{C}S4$ let R be the reflexive and transitive closure of $\prec$ giving a finite model graph $\mathcal{F} = \langle W_0, R\rangle$ whose clusters form a tree. The graph may not be an **S4.14**-frame because **S4.14**-frames must not contain non-final proper clusters and this is not guaranteed of the graph $\langle W_0, R\rangle$.

We claim that all non-final proper clusters can be eliminated from $\mathcal{F}$ whilst still preserving properties (i)-(iii) giving a model graph $\mathcal{F}'$.

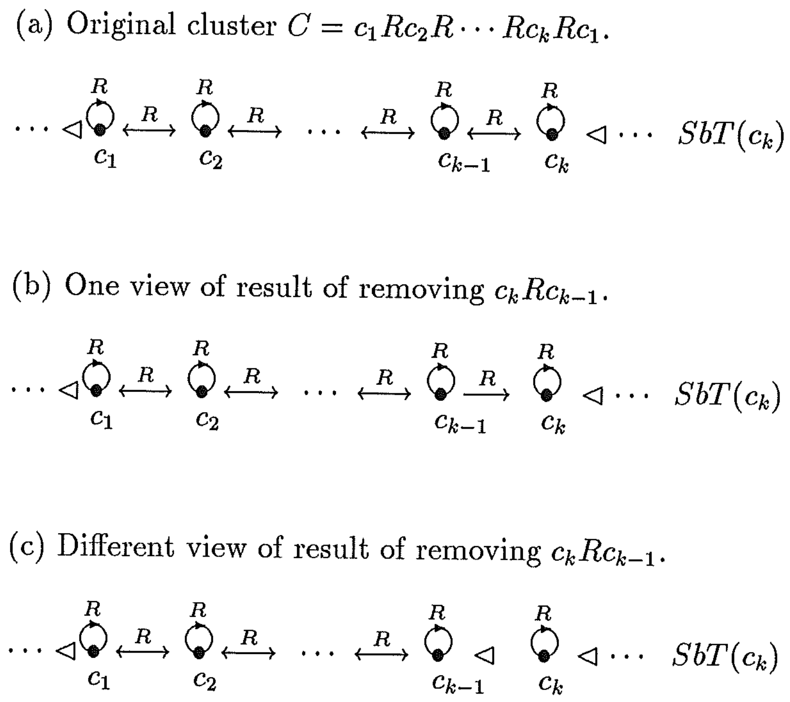


Figure 6.1: Flattening non-final clusters in **S4.14**.

To see this first note that $\mathcal{F}$ is a finite tree where each branch is a sequence of nondegenerate clusters of R. Figure 6.1(a) shows some non-final proper cluster, deliberately expanded horizontally to show its constituent worlds $c_1, c_2, \cdots, c_k$. The double-headed arrows represent the fact that c_iRc_{i+1} and $c_{i+1}Rc_i$ so that any c_i and c_j are mutually accessible via R. That is, these worlds form a nondegenerate proper cluster whose extent is delimited by $\lhd$. We claim that the link c_kRc_{k-1} (the arrow $c_k \longleftarrow c_{k-1}$ in the figure) can be broken without disturbing properties (i)-(iii), giving Figure 6.1(b) and hence Figure 6.1(c).

Suppose $C = c_1Rc_2R\cdots c_kRc_1$ is some non-final cluster in $\mathcal{F}$ where each c_i is also reflexive and where $\mathcal{F}$ is constructed as above. The parts of the tree immediately below c_k (to the right of c_k in the figures) are the clusters C_x that satisfy $C \lhd C_x$. Call this part of the tree the subtree of cluster C and denote it as $SbT(C)$. Note that $SbT(C)$ is non-empty since C is a non-final cluster by supposition. This is represented in Figure 6.1(a).

The tree structure of $\mathcal{F}$ would be left undisturbed if the arc c_kRc_{k-1} were eliminated since no world is eliminated by this step; see Figure 6.1(b). In fact, the only property that may be disturbed is that c_kRc_i would no longer hold for each c_i, $1 \leq i \leq k-1$. Thus the only property that may be altered is property (ii); that is, c_k may contain some eventuality $\neg\Box P$ which is fulfilled by at least one of $c_1, \cdots, c_{k-1}$ because $\neg P \in c_i$ but which is not fulfilled by any other worlds reachable from c_k. Eliminating the said arc may destroy property (ii) since c_kRc_i would no longer hold.

We show that either:

(A) c_k itself fulfills $\neg\Box P$; that is, $\neg P \in c_k$ or

(B) there exists a $w \in SbT(C)$ that fulfills $\neg\Box P$; that is, $\neg P \in w \in SbT(C)$.

Since c_k has at least one successor, it must contain at least one eventuality $\neg\Box P$ as this is the only way that successors can arise (even when R is the reflexive transitive closure of $\prec$). Since c_k is **S4.14**-consistent it must have a successor d due to $\neg\Box P$ and $(S4.14)$ such that $c_k R d$. Let $c_k' = c_k^{\Box} = \{Q | \Box Q \in c_k\}$. Depending on which denominator of $(S4.14)$ gives rise to d :

(a) $\Box c_k' \cup \{\Box\neg\Box P\} \subseteq d$ or

(b) $\Box c_k' \cup \{\Box(P \Rightarrow \Box P), \neg P\} \subseteq d.$

Suppose there is some $w \in SbT(C)$ such that $\neg P \in w$. Then we are done since (B) holds. Otherwise, we have that (B) does not hold, implying that

$(B1)$ $\quad \forall w \in SbT(C), \neg\Box P \notin w$

because $w \in SbT(C)$ and $\neg\Box P \in w$ would imply the existence of some w' with wRw' and $\neg P \in w'$. This w' would have to be in $SbT(C)$ which would contradict our supposition that (B) does not hold.

Given that (B) does not hold and that $(B1)$ does hold, consider the cases (a) and (b).

Case (a): We have $\Box\neg\Box P \in d$. By (T), $\neg\Box P \in d$. But then $d \notin SbT(C)$ by $(B1)$. If d is one of the c_i including c_k itself then $\Box\neg\Box P \in d$ gives $\neg\Box P \in w$ for all $w \in SbT(C)$, which again contradicts $(B1)$. And we know that $SbT(C)$ is non-empty since C was a non-final cluster. Hence when (B) does not hold, (a) cannot hold and (b) must hold.

Case (b): This means that there exists a d, $c_k R d$ such that $\{\Box(P \Rightarrow \Box P), \neg P\} \subseteq d$. Since $\neg P \in d$, our assumption that (B) does not hold implies $d \notin SbT(C)$. Therefore, $d = c_i$ for some $1 \le i \le k$. If $i = k$ then we have shown that c_k fulfills $\neg P$ and we are done since (A) then holds. If $i < k$ then $\Box(P \Rightarrow \Box P) \in c_i = d$, and then dRc_k implies $(P \Rightarrow \Box P) \in c_k$ by (iii). Since $(P \Rightarrow \Box P)$ is just abbreviation for $\neg(P \wedge \neg\Box P)$ this means that $\Box\neg(P \wedge \neg\Box P) \in d$ and $\neg(P \wedge \neg\Box P) \in c_k$. Then by $(\vee)$, $\neg P \in c_k$ or $\neg\neg\Box P \in c_k$. But we already know that $\neg\Box P \in c_k$ and that c_k is $CS4.14$-consistent, so $\neg\neg\Box P \in c_k$ is impossible and it must be that $\neg P \in c_k$. We have just shown that (A) holds whenever (B) does not hold.

Thus we can liberate c_k from the cluster C and still maintain properties (i)-(iii). The cluster C can be replaced by two clusters C_1 and c_k with $C_1 = C \setminus \{c_k\}$, $C_1 \lhd c_k$, $c_k R c_k$ and $c_k \lhd SbT(C)$ as shown in Figure 6.1(c). Repeating this process gives the linear order $\lhd\, c_1 \lhd c_2 \lhd c_3 \cdots \lhd c_k \lhd$ giving $\mathcal{F}_1$ where each cluster c_i is reflexive and $\mathcal{F}_1$ is still a model graph. As C was any non-final proper cluster, this can be done for all non-final proper clusters giving some final $\mathcal{F}' = \langle W_0, R' \rangle$ that is also a model graph where R' is the altered reachability relation. But $\mathcal{F}'$ is now an **S4.14**-frame since it contains no non-final proper clusters.

Property (i) still holds because we have not removed any elements of W_0 hence $X \subseteq w_0 \subseteq W_0$. Property (iii) holds because we have not added any extra tuples to R, only removed some. So if it held before the pruning process, it must hold after it. And property (ii) holds because of the argument above. Since properties (i)-(iii) still hold, $\mathcal{F}'$ is also a model graph and hence an **S4.14**-model for X.

Note that the proof does not stipulate any particular ordering for $C = c_1 R \cdots R c_k$. That is, C can be flattened into an arbitrary sequence of its constituent worlds and consequently, the proof is constructive. ●

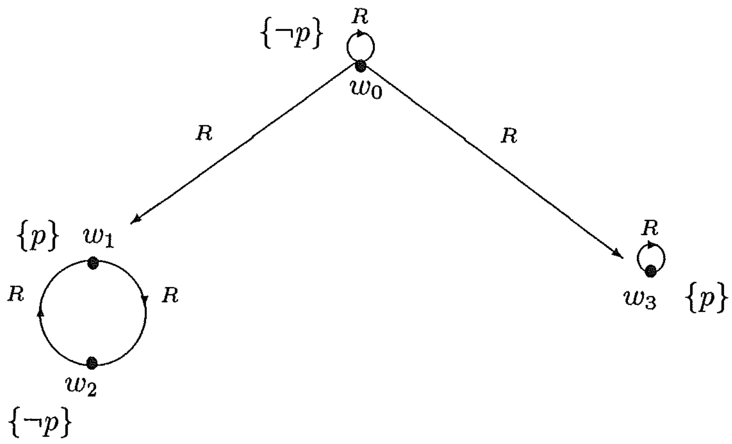
6.3.4 A Note on S4.14

In the following passage from a chapter on modal logic by Segerberg and Bull [BS84, page 51], Bull refers to **S4M** as **S4.1**, refers to **S4MDum** as **S4.1Dum** and cites Segerberg's PhD. thesis [Seg71]:

> With what natural axiom can **S4.1** be extended to **S4Grz**? Clearly we need a formula A such that **S4A** is characterised by the finite reflexive-and-transitive frames in which all but the final clusters are simple. Segerberg [1971], chapter II, section 3, shows that
>
> $$\textbf{Dum.} \quad \Diamond\Box P \Rightarrow (\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow P)$$
>
> (i.e., $\Diamond\Box P \Rightarrow Grz$) has this property, so that **S4Grz** is **S4.1Dum**.

Bull's **Dum** is not identical to our *Dum* but see page 36 where we noted that the differences are immaterial in the field of **S4**.

There are two claims in the above passage. One is that "**S4Grz** is **S4.1Dum**" which we do not dispute. The other is that the logic **S4Dum** "is characterised by the finite reflexive-and-transitive frames in which all but the final clusters are simple". We show that this second claim is not correct by giving a finite reflexive-and-transitive model in which all but the final clusters are simple, but in which neither *Dum* nor **Dum** is valid. The model is pictured in Figure 6.2.



Dum can be written as: $\Box(\neg p \Rightarrow \Diamond(p \wedge \Diamond\neg p)) \wedge \Diamond\Box p \Rightarrow \Box p$;

$w_0 \models \Diamond\Box p$ because $w_3 \models \Box p$;

$w_0 \models \neg p \Rightarrow \Diamond(p \wedge \Diamond\neg p)$ because of w_1 and w_2;

$w_0 \models \Box(\neg p \Rightarrow \Diamond(p \wedge \Diamond\neg p))$

but $w_0 \not\models p$ and hence $w_0 \not\models \Box p$

Figure 6.2: A finite reflexive-and-transitive model in which all but the final clusters are simple in which *Dum* and **Dum** are false at w_0.

The explanation rests on the fact that

$$\Box(\Box(P \Rightarrow \Box P) \Rightarrow P)$$

can be written as

$$\Box(\neg P \Rightarrow \Diamond(P \wedge \Diamond\neg P)).$$

Thus *Dum* can be written as:

$$\Box(\neg P \Rightarrow \Diamond(P \wedge \Diamond\neg P)) \wedge \Diamond\Box P \Rightarrow \Box P.$$

This is just as well because we have just shown that **S4.14** characterises this class and *Dum* and *Zbr* are different. But note that the extra $\Box$ modality in *Zbr* is exactly what is needed since, in the counter-example of Figure 6.2, $w_0 \not\models \Box\Diamond\Box p$. That is, the counter-example does not falsify *Zbr* because the extra modality handles the branching inherent in **S4.14**-models which is absent in **S4.3.1**-models.

Segerberg [Seg71, page 106] shows that "**S4Dum** *is determined by the class of all reflexive kites and all finite reflexive trees*". The correct reading of this statement is "**S4Dum** is determined by the class that consists of all reflexive kites and all finite reflexive trees".

6.4 Equivalence of Grz, KTGrz, S4Grz and S4MDum

An axiom that Segerberg [Seg71, page 169] attributes to Sobociński [Sob64] but which he names Grz after Grzegorczyk [Grz67] is:

$$Grz : \Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow P.$$

It actually comes in three other flavours:

Grz_1: $\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \Rightarrow \Box P$;

Grz_2: $\Box(\Box(P \Rightarrow \Box P) \Rightarrow \Box P) \Rightarrow P$;

Grz_3: $\Box(\Box(P \Rightarrow \Box P) \Rightarrow \Box P) \Rightarrow \Box P$;

but Segerberg [Seg71, page 107] shows that in the field of **S4** these three are all equivalent. Note that Grz_1 is just G_o.

In [Seg71, page 103] it is stated that **S4Grz** is determined by the class of all finite reflexive (transitive) trees. But we now know (page 33) that the logic **KGrz**, which we call **Grz**, is characterised by this same class of frames. That is **Grz** = **KGrz** = **S4Grz**. It is known that **KGrz** = **KTGrz** [HC84]. Segerberg [Seg71, page 107] also shows that **S4Grz** = **S4MDum**. Hence **Grz** = **KGrz** = **KTGrz** = **S4Grz** = **S4MDum**.

That is, $\mathcal{C}Grz$ is a decision procedure for **S4MDum** = **S4Grz** = **KTGrz** = **KGrz** = **Grz**.

The above equivalences imply the following implications:

(a) $Grz \Rightarrow 4$;

(b) $Grz \Rightarrow Dum$;

(c) $Grz \Rightarrow T$;

(d) $Grz \Rightarrow M$.

Property (a) was proved by van Benthem and Blok [vB78]. Property (b) is obvious since axiom Dum (page 23) is a specialisation of axiom Grz_1; that is, the proof is an instance of: if $A \Rightarrow C$ then $A \Rightarrow (B \Rightarrow C)$, where $B = \Diamond\Box P$. Property (c) has been proved by van Benthem and Blok [vB78]. Property (d) has been proved by Sobociński [Sob64]; see [BS84, page 32].

6.5 Systems For Logics Of Linear Finite Time Intervals

Consider the logic **S4.3MDum** axiomatised as **KT43MDum** where M is $\Box\Diamond P \Rightarrow \Diamond\Box P$. Zeman notes that $\Box\Diamond P \Rightarrow \Diamond\Box P$ guarantees the existence of end points; see [Zem73, page 270] where the observation is attributed to Prior.

The final cluster in an **S4.3.1**-frame is allowed to be a proper cluster precisely because that is the only place in an **S4.3.1**-frame where formulae like $\Box\Diamond P$ and $\Box\Diamond\neg P$ can be fulfilled infinitely often without falling into inconsistency. But the addition of M now forbids $\Box\Diamond P$ and $\Box\Diamond\neg P$ from co-existing in harmony. That is, M forces the value of P to eventually settle to a fixed value forever.

We already know that **S4MDum** = **Grz**. Therefore it seems reasonable to conjecture that **S4.3MDum** = **Grz.3** and that **Grz.3** is characterised by the class of all *finite* sequences of simple clusters. This would then make **Grz.3** an appropriate logic for finite (reflexive and transitive) discrete linear time intervals.

It seems reasonable to believe that a cut-free tableau system for **Grz.3** is possible using our knowledge about the effects of 3 in obtaining $\mathcal{C}S4.3$.

Alternatively, since Dum is a theorem of **S4.3MDum**, so are the following:

$$\begin{aligned} \neg\Box P &\Rightarrow \neg\Box(\Box(P \Rightarrow \Box P) \Rightarrow P) \vee \neg\Diamond\Box P \\ \neg\Box P &\Rightarrow \Diamond(\Box(P \Rightarrow \Box P) \wedge \neg P) \vee \Box\Diamond\neg P. \end{aligned}$$

But M gives $\Box\Diamond\neg P \Rightarrow \Diamond\Box\neg P$, hence in **S4.3MDum** we have

$$\neg\Box P \Rightarrow \Diamond(\Box(P \Rightarrow \Box P) \wedge \neg P) \vee \Diamond\Box\neg P$$

as a theorem.

This latter intuition gives a rule like ($S4.3.1$) except that the left denominator would also involve a "jump" to the world where $\Box\neg P$ eventually becomes true. Recall that in the ($S4.3.1$) the left denominator merely "lifts" the eventuality $\neg\Box P$ to $\Box\neg\Box P$.

We therefore seem to have two alternative ways to attack **Grz.3** which we leave as further work.

6.6 Systems For Logics Of Branching Finite Time Intervals

The logic **Grz** is characterised by finite trees of simple clusters, hence it is the logic for a branching model where each branch is a finite time interval.

The axiom G_o is a variant of Grz; see page 23. Rautenberg [Rau83, page 414] notes that $\mathbf{G_o}$ is characterised by transitive frames with no infinite ascending chain of pairwise distinct points. We conjecture that it is also characterised by the class of finite transitive trees with (irreflexive) degenerate non-final clusters and (reflexive) simple final clusters.

The semantic intuitions of the (Grz) rule involved the fact that G_o is a theorem of **Grz**, hence $\mathcal{C}G_o$ is just $\mathcal{C}Grz$ minus the rule (T). But in the completeness proof for $\mathcal{C}Grz$, we relied on the fact that we created a successor for $\neg\Box P \in w$ only when $\neg P \notin w$, and this was crucial to show that a sequence of successors terminates. We cannot use this fact for $\mathcal{C}G_o$ since reflexivity is missing and the completeness proof must allow cycles. It may be possible to flatten non-final clusters as in the completeness proof for $\mathcal{C}S4.14$ but this requires further work.

Chapter 7

Related Work

Tableau and sequent systems for modal logics have been the focus of research in philosophy, mathematics and lately, in computer science. The main aim in philosophy has been to obtain completeness results with respect to classes of frames, and for this, tableaux have been used more than sequent systems [BS84]. The main aim in mathematics has been to obtain syntactic cut-elimination proofs and thereby relate modal logics to logics of provability and consistency of Peano arithmetic [Boo79, SV80]. The main aim in computer science has been to obtain decision procedures for theorem proving in modal logics [Fit88]. The three broadest classifications are tableau systems, sequent systems and normal forming methods, although modal resolution methods are also of importance in computer science.

7.1 Tableau Systems

There are two basic tableau methods for modal logics and as both are refutation procedures, both methods use sets of formulae to represent different possible worlds. The differences lie in the way that the reachability relation is modelled. In the first method, the desired semantic reachability relation is imposed onto the tableau construction explicitly as a set of external constraints, above and beyond the tableau rules. In the other, the effect of the reachability relation is achieved (implicitly) by the tableau rules themselves, using the basic successor relation S of trees. The methods are not equivalent and to distinguish the two, we refer to the former as **constrained tableau systems** and the latter as **tree tableau systems.**

In older work, each possible world is called a tableau and tableaux are interconnected by an auxiliary relation representing the semantic reachability relation. We retain this terminology *for this chapter only* and refer to each world as a tableau and say that one tableau is auxiliary to another if the latter is reachable from the former. It should be clear whether we mean tableau as worlds or tableau systems. Also note that we use R to mean the tableau auxiliary relation and not the semantic reachability relation as we have done previously. Since the auxiliary relation R always mimics the semantic reachability relation, this is not a problem.

In constrained tableau systems, the semantics of the logic are explicitly forced onto the auxiliary relation by stipulating that it be reflexive, transitive or whatever. In tree tableau systems, a tree construction is used giving a successor relation relating parent tableau to their immediate offspring. But this relation is neither reflexive nor transitive nor symmetric and the "effect" of these properties is obtained via the the tableau rules themselves. Thus constrained tableau systems work on a global level; we can work on any node of the construction at will. On the other hand, tree tableau work at a local level; we can work on the current node, or on its immediate children, or we can create a new node.

Apart from this there is one distinction between older tableau systems and their newer counterparts. In the older works on tableau systems, the non-determinism inherent in the various rules is made explicit by exploring "alternative" sets of tableau where necessary. The idea is that only one of these alternative sets need close to obtain a proof. That is, the whole search space of attempted tableau proofs is maintained explicitly, and a notion of "alternative" sets of tableaux becomes necessary. In newer tableau methods this non-determinism is implicit and only one tableau construction is kept at any point of the procedure. We ignore this aspect of the older work as it is not central to the understanding of the methods, and may have even contributed to some confusion [Kap66].

7.1.1 Constrained Tableau Systems

The most celebrated work is of course that of Kripke [Kri59] where possible worlds related by an accessibility relation are first proposed as a semantics for modal logics. Bull and Segerberg [BS84] give an account of the genesis of the possible worlds approach and suggest that credit is also due to Hintikka and Kanger. Zeman [Zem73] even credits C. S. Pierce with the idea of "a book of possible worlds" as far back as 1911!

Kripke follows Beth [Bet55] and divides each tableau into a left hand side and a right hand side where the left side is for formulae that must be assigned "true" and the right side is for formula that must be assigned "false". Thus it is clear that this is a refutation procedure and we are attempting to obtain a falsifying model of possible worlds for the given formula. To handle the added complexities of modal formulae like $\Box A$ and $\neg\Box A$, Kripke uses auxiliary tableau, where a new tableau is used for each possible world and these auxiliary tableaux are interrelated by an auxiliary reachability relation R. Auxiliary tableaux may have tableaux auxiliary to them and so on, obtaining a complex web of tableaux.

Kripke uses two basic rules to handle modal formulae: one to handle $\Box A$ on the left of a tableau and one to handle $\Box A$ on the right of a tableau. They are,

Yl: If $\Box A$ appears on the left of a tableau t, then for every tableau t' such that tRt', put A on the left of t';

Yr: If $\Box A$ appears on the right of a tableau t, then start out a new tableau t', with A on the right, and such that tRt'.

Different constraints on this auxiliary relation give different tableau systems. That is, the definition of the auxiliary relation R changes with each logic, so that the auxiliary relation directly mimics the required accessibility relation. For example, the auxiliary relation R for **S4** is defined to be reflexive and transitive, so for any tableau t we have *tRt by definition.* These constraints form an extra theory about R that must be taken into account at each rule application.

Note also that the application of the Yl rule can have delayed consequences. For example, if a new auxiliary tableau t'' is created and it happens to be auxiliary to the tableau t in which the Yl rule has already been applied, then we have to keep track of this previous application of Yl and add A to the left of t''. Thus, the meaning of "every tableau t' such that tRt'" includes tableaux that may come into existence via the Yr rule at any later point of the construction. The rules are therefore like constraints that may be activated at a later time and this is why we call these systems "constrained tableaux".

This is essentially a way to keep track of all worlds in the counter model being sought. When a new world comes into existence, it is immediately linked into this counter-model according to the constraints on R. That is, Kripke's method is a refutation procedure where extra modal information is kept in the auxiliary relation between tableau. The construction is on a global level in that we can return to previous nodes of the tableau construction at will. In our tableau systems CL we cannot return to nodes higher up in the tree.

The semantic diagrams of Hughes and Cresswell [HC68] and the tableau systems of Zeman [Zem73] use essentially the same ideas except that Hughes and Cresswell use annotations of ones and zeros instead of using a left and right side. Slaght [Sla77] goes one step further than usual and adds rules for quantifiers and also incorporates a form of negated normal form by translating $\neg\Box P$ into $\Diamond\neg P$, $\neg\Diamond P$ into $\Box\neg P$, $\neg\exists x(\cdots)$ into $\forall x\neg(\cdots)$ and $\neg\Box x(\cdots)$ into $\exists x\neg(\cdots)$.

Kanger's spotted formulae [Kan57], which precede Kripke's work, are slightly more sophisticated versions of these ideas where the extra information is kept via prefixes of strings annotating each formula. Fitting's prefixed tableaux are direct applications of Kanger's idea to handle many different modal logics [Fit83, chapter 8]. In this method, each formula is prefixed with a string to retain its modal context and an extra level of reasoning about these prefixes is built into the rules of the prefixed tableau as restrictions on their applicability. Refinements of this idea have been investigated by Morgan [Mor76], Wallen [Wal87], Ohlbach [Ohl90], Auffray and Enjalbert [AE89] and Frisch and

Scherl [FS91]. In fact, these methods have become known in the computer science literature as "translational methods" because they either explicitly or implicitly translate the modal logic into some extension of classical first order logic. The most recent and comprehensive treatment of this idea to my knowledge is the work of Gent [Gen91] who seems to have identified the limits of this method.

Gent works with what he calls "logics of restricted quantification". Gent uses a standard first order classical logic except that its quantifiers carry restrictions from a separate "meta-theory". A typical restricted quantifier is one like $\forall w_{:C}$ where C is some condition from the meta-theory. Gent's "theory tableaux" work with two theories, where the restrictions from the meta-theory appear as side conditions of the tableau rules. The object theory is quite general but the meta-theory has certain syntactic restrictions. The meta-theory effectively acts as a constraint theory determining when an object level derivation is a proof in the overall theory of restricted quantification. To capture modal logics, Gent pushes the modal aspects into the meta-theory by making it a theory involving explicit constraints about R (where R is now the semantic reachability relation). That is, the modality $\Box$ becomes a restricted universal quantifier $\forall x_{:wRx}$ of the object theory and $\Diamond$ becomes a restricted existential quantifier $\exists x_{:wRx}$ of the object theory with different meta-theoretic restrictions corresponding to different modal logics.

In all these translational methods, the modal logics **K**, **T**, **K4**, **S4** and **S5** are easily handled and Gent has also obtained systems for **B** and **S4.3**. The most striking feature of Gent's work is that he is unable to give a system for **S4.3.1** and this is essentially due to the fact that the reachability relation R for **S4.3.1**-frames is not first order definable. It is known that a formula of second order logic is required to express the reachability relation for **S4.3.1** [vB83]. This deficiency of translational methods is also mentioned by Auffray and Enjalbert [AE89]. Frisch and Scherl [FS91] find that **K45** and **K45D** also prove problematic for exactly the same reasons.

In all fairness, it must be mentioned that the translational methods seem to be much better for automated deduction in first order modal logics where various domain restrictions can complicate matters for the first order versions of our tableau systems $\mathcal{CL}$; see [Ohl90]. At the first order level, all modal logics are only semi-decidable since they all include classical first order logic. Then, decidability is no longer an important issue.

7.1.2 Tree Constructions

As we have noted, the constrained tableau method is really a way to use external constraints to mimic the desired reachability relation. Kripke tries to avoid this by using tree tableau. In tree tableau, a single tree of tableau nodes is kept where a parent tableau t and a child t' are related by the successor relation S of the tree, giving tSt'. Kripke [Kri63b, page 80] shows that **T**, **B**, **S4** and **S5** are also complete in terms of trees of possible worlds. Instead of using an explicit auxiliary relation R to mimic the reachability relation, Kripke shows that the basic successor relation S of a tree suffices. This idea is the basis of the completeness proofs of Chapter 5 where we use $\prec$ instead of S. That is, the tree tableau described below are essentially the deterministic decision procedures we can extract from our completeness proofs for $\mathcal{CL}$. Thus it should be possible to extract our tableau systems from the procedures described below. However, it is much easier to go from $\mathcal{CL}$ to these procedures than vice-versa.

A tableau t' is auxiliary to a tableau t if tSt', that is, if t' is an immediate child of t. New tableau are created by the rule Yr which remains the same, except that we use S instead of R. A priori, the relation S is not reflexive, not transitive and not symmetric but these properties are obtained by changing the tableau updating rules to instill the effect of the desired property.

For example, reflexivity is obtained by replacing R by S and changing the Yl rule for handling $\Box A$ on the left to:

Yl: Let $\Box A$ appear on the left of t_1. Then put A on the left of t_1 and of any tableaux t_2 such that $t_1 S t_2$.

Thus the responsibility of putting A on the left of t is part of the rule Yl itself rather than a consequence of the explicit constraint $t_1 R t_1$ as in the first version of the Yl rule.

Kripke [Kri63b, page 81] notes that this basic relation S is not automatically transitive but that "the *effect* of transitivity" and reflexivity can be obtained by changing the above rule to:

Yl: if $\Box A$ appears on the left of a tableau t_1, put A on the left of t_1 and put $\Box A$ on the left of any tableau t_2 such that $t_1 S t_2$.

Now if a further tableau t_3 appears such that $t_2 S t_3$, we do not have to remember that S is supposed to be transitive from t_1 to t_3 and put A on the left of t_3. That is, the Yl rule, when eventually applied to the $\Box A$ in t_2 will give us "the effect of transitivity" by putting A on the left of t_2 and putting $\Box A$ on the left of t_3.

In this way, each of the rules Yl and Yr become local rules, each ensuring that enough information is included in the tableau nodes to achieve *the effect* of reflexivity and transitivity. But notice that this requires us to apply Yr before Yl so that rule order becomes significant. The usual assumption is that all possible rules are applied to a tableau before moving onto its children. That is, each tableau is saturated before moving onto its children as in the deterministic decision procedures we can extract from our completeness proofs.

For reflexivity and symmetry together, Kripke [Kri63b, page 81] changes Yl to:

Yl: Let $\Box A$ appear on the left of t_1. Then: (1) put A on the left of t_1; (2) put A on the left of every tableau t_2 such that $t_1 S t_2$; (3) put A on the left of the (unique) tableau t_3 such that $t_3 S t_1$, if such a tableau exists.

Here is the major difference between our completeness proofs and Kripke's tree tableau. For now, the Yl rule is explicitly allowed to add information to a *parent* node. That is, symmetry is imposed explicitly rather than achieved as an "effect". In our completeness proofs, the effect of symmetry was achieved by the tableau rules $(sfcT), (5), (B)$ and $(S5)$ depending on the logic in question.

Kripke [Kri63b, page 81, footnote 1] then asks whether a similar local trick can be used to get the effect of symmetry, and notes a trick due to Hintikka viz: if $\Box A$ appears on the right of t and tSt', then put $\Box A$ on the right of t'. But Kripke notes that this trick is not enough as the **S5**-theorem $A \Rightarrow \Box\Diamond A$ has no proof in this modified system. Kripke conjectures that this incompleteness is evidence for the non-existence of a cut-free tableau system for **S5** and relates this to the inability to prove the Gentzen Hauptsatz in various sequent systems for **S5**. This trick is used again in the preservation of eventualities as in Fitting's system for **S5** and in Shvarts' systems for **K45** and **K45D**; see Section 5.3 (page 67).

The crucial trick that Kripke seeks is, of course, the combined effects of $(sfcT)$, and (5). What Rautenberg realises is that under certain cases, the effect of symmetry of S can be built into the rules via $(T), (sfcT)$ and (5) answering Kripke's question in the affirmative for **S5** and **B**. In Section 5.5.1 (page 90) we showed that for **S5**, the $(sfcT)$ rule alone is sufficient as long as we use Hintikka's trick in the $(S5)$ rule. That is, we can replace the building up rule (5) by $(S5)$ and regain the subformula property. But note that Rautenberg's method works only because of the support of other rules like (T) and $(sfcT)$ which are both present in $\mathcal{C}S5$ and $\mathcal{C}B$. Rautenberg does not give a tableau system for **KB** and this is what Kripke is really asking about.

We have already noted that (sfc) is really just Smullyan's "analytic cut" in disguise. But the utility of $(sfcT)$ (alone) appears to have been first noted by Hanson [Han66a] in 1966. In both constrained tableau systems and tree tableau systems, we must keep track of all previous nodes of a tableau construction to detect termination. This is analogous to the cycles we sought in the completeness proofs of Chapter 5.

Hanson shows that the test for termination via cycles can be greatly simplified by adding $(sfcT)$ to Kripke's tree tableau. Specifically Hanson proposes to replace the rules for $A \wedge B$ on the right and $\Box A$ on the right by:

$\wedge r^*$: *if $A \wedge B$ appears on the right of a tableau t, then there are three alternatives:*

1. *put A on the left and B on the right of t;*
2. *put B on the left and A on the right of t;*
3. *put both A and B on the right of t.*

At the next stage of the construction the ordered set of tableaux of which t is a member is replaced by three alternative sets, each of which embodies a different one of the three alternatives.

Yr^*: If $\Box A$ appears on the right of a tableau t, then there are two alternatives:

1. *put A on the right of t;*
2. *put A on the left of t, start a new tableau t' such that tSt', and put A on the right of t'.*

At the next stage of the construction the ordered set of tableaux of which t is a member is replaced by two alternative sets, each of which embodies a different one of the two alternatives.

Hanson defines the level $L(t)$ of a tableau t as:

$L(t) = 0$ if t is the main tableau;

If there is a tableau t' such that $t'St$, and $L(t') = k$ then $L(t) = k + 1$.

His termination rules are:

S5 : *In an* **S5***-construction apply no rule to any wff in a tableau t if $L(t) > 1$;*

S4 : *In an* **S4***-construction apply no rule to any wff in a tableau t if $L(t) > 1$ and each wff of the form $\Box C$ that appears on the left of t' also appears on the left of t'', where t' and t'' are tableaux such that $t''St'$ and $t'St$.*

In the **S4** case, if the **S4** condition is satisfied then we can construct a (counter) model by discarding t, putting $t'St''$ and taking the reflexive and transitive closure of S. This gives a cycle in S as in $CS4'$. Thus Hanson's rules are the deterministic analogues of $CS4'$; see page 88.

In the **S5** case, the (counter) model is contained in the tableaux of levels 0 and 1 since discarding all tableau of level greater than two and forcing symmetry and transitivity on S gives a tree of depth two. But note that Hanson's rules for **S5** are not the exact analogues of $CS5'$ for he uses the symmetric Yl rule described on page 128 that explicitly puts A into the left side of the parent node. In $CS5'$ we use Hintikka's trick to achieve this effect by forcing $\neg\Box Y$ into its denominator thus allowing the proof of property (a) to go through on page 91.

Zeman's [Zem73] tableau system for **T** is identical to Kripke's constrained tableau system for **T**. But for **S4**, Zeman uses tree tableau to achieve the effect of transitivity and further stipulates that the auxiliary relation be transitive; the latter appears to be unnecessary. So at first Zeman does not appear to distinguish between constrained tableau and tree tableau. But Zeman then considers two tableau systems for **S5**. One is called $MS5$ and is the same as Kripke's since the Yl rule is allowed to explicitly place formulae into the parent node. The other, called $MS5'$, is an attempt to achieve the effect of symmetry and Zeman finds that $MS5'$ is not complete unless a tableau cut rule is added, confirming Kripke's comments about the Gentzen Hauptsatz. Zeman uses $MS5'$ to show that each

$MS5'$ tableau proof corresponds directly to a sequent system proof for **S5** where his sequent system $LS5$ is allowed to contain cut. He does not analyze the type of the cut rule and so does not narrow it down to either analytical or semi-analytical cut.

As we stated previously, Zeman [Zem73] appears to be the first to give a tableau system for **S4.3** but he uses alternative tableau, which as we have seen, are unnecessary. The explicit use of alternative tableau appears to have blinded people to the strong semantic links between tableau systems and sequent systems. For example, Zeman is unable to translate his cut-free tableau system for **S4.3** into an analogous cut-free sequent system for **S4.3** [Zem73, page 232] although he does offer a sequent system for **S4.3** requiring cut. This deficiency with alternative tableau is noted in David Kaplan's review of Kripke [Kap66].

Zeman [Zem73, page 235] also appears to realise that thinning is essential for **S4.3** because his tableau rule for **S4.3** in our notation has the form *"when* $\Box P_1, \Box P_2, \cdots, \Box P_k$ *are present on the right of an* **S4.3** *tableau then ...* ". That is, Zeman does *not* stipulate that these are *all* the eventualities, only that they are some (finite subset) of all the eventualities of a **S4.3** tableau node.

The book by Fitting [Fit83] is probably the most comprehensive treatment of proof methods for modal logics. As stated previously, Fitting systematises Kanger's use of prefixes to obtain cut-free constrained tableau systems for many normal, non-normal and even first order modal logics. But he also gives tree tableau systems for **K**, **T**, **D**, **K4**, **D4**, **S4**, **KB**, **DB**, **B**, **S5** and **G**.

Fitting divides his tree tableau systems into the **analytical** ones like **K**, **T**, **D**, **K4**, **D4**, and **S4** that have the subformula property and are cut-free, and **non-analytical** ones like **KB**, **DB**, **B**, **S5** and **G** that either require some form of cut rule or that do not possess the subformula property. Fitting's analytical (tree) tableau systems and our systems CK, CT, CD, $CK4$, $CD4$, $CS4$ are almost identical; the only difference is that Fitting builds in an explicit contraction rule since he uses Smullyan-tableau. We have already noted that Fitting's sequent systems [Fit83, pages 81 onwards] for these logics are incomplete for exactly the same reasons as are Rautenberg's; see page 93.

For the non-analytic logics **KB**, **DB**, **B** and **S5**, Fitting uses a "semi-analytic" cut rule proposed by Osanu Sonobe based on Smullyan's idea of "analytic cut" for classical first order logic [Smu68a]. The "analytic cut" idea in terms of Beth-tableau is to use the cut rule:

$$(cut)\ \frac{X; A}{X; A; P \mid X; A; \neg P} \quad \text{where } P \in Sf\,(A)$$

on any subformula P of any formula that has appeared previously on the current branch. In this way, the uses of the cut rule are kept "analytic" since this set of subformulae is finite for any given initial set of formulae. But analytical cut is not enough for modal tableau involving symmetry alone and so Fitting uses a "semi-analytical" cut rule where *"the cut rule is restricted to subformulas of formulas already on the branch, and to formulas built up from them by prefixing modal operators"* [Fit83, page 193]. And later, on the same page, Fitting comments that semi-analytic cut cannot be used for proving decidability because *"it can introduce formulas of arbitrary high degree into a proof"*. That is, the "cut-class" of superformulae is not bounded, so that semi-analytic cut still involves guessing the right cut formula. But note that Fitting obtains "strong completeness" rather than weak completeness.

(Rautenberg shows that for **B** and **S5** this cut-class of superformulae *can* be bounded since this cut-class is exactly X_L^*, and that analytic cut, that is (sfc) or $(sfcT)$, is sufficient as long as some extra building up rules like (5) and (B) are present that add selected superformulae from this cut-class to the tableau. For **S5** we have shown that analytic cut and the subformula property are sufficient; see $\mathcal{C}S5'$ (page 90). This however, only gives us weak completeness).

Fitting [Fit83, page 225] also gives another tree tableau system for proving single formulae of **S5** which is cut-free and weakly complete as long as an extra building up rule is added. The Beth-tableau counterpart for this rule is:

$$(\pi)\ \frac{X; P}{X; P; \neg\Box\neg P}$$

The rule is certainly sound since by reflexivity we have $P \Rightarrow \Diamond P$ as a theorem in **S5**. As a refinement, Fitting shows that this extra rule need be applied only once at the very beginning of the tableau construction [Fit83, page 229]. This is effectively the result of Shvarts [Shv89] that A is a theorem of **S5** iff $\Box A$ is a theorem of **K45**. That is, to test A for theoremhood in **S5**, we would construct a $\mathcal{C}S5\pi$-tableau for $\{\neg A\}$. If the (π) rule is applied only once right at the start then we are really constructing a $\mathcal{C}S5\pi$-tableau for $\{\neg\Box\neg\neg A\}$, which is the same as testing $\Box A$ for theoremhood. Shvarts, however, does not mention Fitting's work and Fitting does not seem to realise that his system, minus the (π) rule, is complete for **K45**.

The only other non-analytic system that Fitting considers is for **G**, and his system is the Smullyan-tableau system analogous to $\mathcal{C}G$ where there is no cut rule, but where the rule (G) breaks the subformula property. Boolos [Boo79] also gives a semantic cut-elimination proof for **G**. Since then, syntactic cut-elimination has been proved by Bellin [Bel85a] and an improved proof has been given by Valentini [Val83]. Boman gives a recent survey of provability interpretations of **G** [Bom90].

Fitting suspects that the loss of the subformula property could be a general principle to obtain cut-elimination for other modal logics. For example, Fitting [Fit83, page 226] states *"Second, the effects of dropping cut but adding explicit "building up" rules to the logics KB, DB and B, is not known"* and goes on to ask whether adding the following building up rules and their duals would give strongly complete (in his sense) tableau calculi for these systems: build $A \wedge B$ from A, B; build $A \vee B$ from A; build $A \vee B$ from B; and build $\Diamond A$ from A.

The answer for **B** is given by Rautenberg [Rau83] and, ironically, his paper was published in the same year as Fitting's book [Fit83]. Rautenberg appears to be the first to realise that symmetry can be handled using an analytic cut rule provided the subformula property is forsaken, but his work is one of the least cited work I have encountered, possibly due to its publication in a philosophical journal rather than in a logic journal. I know of only one reference to Rautenberg's work! Although an erratum was published for Rautenberg's paper [Rau85], this erratum does not correct the incompleteness of his systems due to the absence of any form of contraction.

Wallen [Wal87] bases his connection matrix methods on the work of Andrews [And81], Bibel [Bib81] and Fitting [Fit83] although Binkley and Clark [BC67] appear to have given a "connection method" for classical first order logic as early as 1967. The connection matrix method utilises an explicit form of the contraction rule in the multiplicities associated with each formula. Thus, the fine grained form of contraction present in our ($S4$) rule, for example, is lost. As a consequence, Wallen is forced to abandon decidability even at the propositional level. In any case, Gent shows that his "theory tableaux" systems generalise Wallen's method and we have already mentioned the limits of Gents' method.

7.2 Sequent Systems

As Bull and Segerberg [BS84] note, sequent systems for modal logics are rare. Most of the work is in tableau systems following the seminal semantic completeness results of Kripke [Kri63b]. According to Kripke [Kri63b], Ohnishi and Matsumoto [OM57b] and Kanger [Kan57], the earliest work appears to be that of Curry [Cur52, Cur50] where a cut-free sequent system with the subformula property is given for **S4**.

Ohnishi and Matsumoto [OM57b, OM57a, OM59] give cut-free sequent systems with the subformula property for **K**, **T** and **S4** but are unable to eliminate cut from their sequent system for **S5**. They retain the contraction rule and the rules are essentially (inverted) sequent versions of the tableau rules of $\mathcal{C}K$, $\mathcal{C}T$ and $\mathcal{C}S4$ except that their rules do not duplicate the numerator in the denominator (since contraction is freely available). Kripke [Kri63b, page 91, footnote 1] points out a slight omission from the work of Ohnishi and Matsumoto in that the **K**-theorem $\neg\Box\neg A \Rightarrow \Diamond A$ is not provable in their systems since they do not cater for the obvious inter-definability of $\Box$ and $\Diamond$ via $\Diamond P = \neg\Box\neg P$. Kripke gives an obvious fix.

Kanger [Kan57] also gives cut-free sequent systems with the subformula property for **T** and **S4** but is able to give a (prefixed) cut-free sequent system with the subformula property for **S5** as well where additional information is kept via spotted formulae. This additional information is precisely the information required to keep modal contexts as in the prefixed tableaux of Fitting and the translational methods in general.

Shvarts [Shv89] gives cut-free sequent systems with the subformula property for **K45** and **K45D** as shown in Section 5.3 (page 83) and gives embeddings of **S5** into each so that his sequent systems for **K45** and **K45D** individually suffice for **S5** as well. As stated previously, Fitting precedes Shvarts in this aspect although Fitting does not point out the relationship to **K45** and **K45D** of his result.

Mints [Min70, Min] (sometimes referred to as Minc) appears to have been the first to realise that a cut-free sequent system for **S5** is possible without using prefixes if the subformula property is forsaken. However Mints does not use the concept to obtain sequent systems for other modal logics although he does give a sequent system for an intuitionistic version of **S5**.

Zeman [Zem73] not only gives cut-free tableau systems for many modal logics but also gives sequent systems for most of them. Zeman's cut-elimination proofs for his sequent systems are all syntactic, making them quite complicated. The most curious aspect of Zeman's work is that he is unable to give cut-free sequent systems for some logics despite giving cut-free tableau systems for the same logics. A striking example is the sequent system for **S4.3**.

Instead of extracting a sequent rule for **S4.3** from his tableau system for **S4.3**, Zeman adds an extra axiomatic sequent rule (with no premisses):

$$\Diamond P, \Diamond Q \longrightarrow \Diamond(P \wedge \Diamond Q), \Diamond(Q \wedge \Diamond P)$$

and then shows that this rule together with the cut rule can simulate the **S4.3** tableau rule. Zeman concludes that "we are unable to prove cut elimination for this system" [Zem73, page 232]. But it should be clear that we can extract a sound, complete and cut-free sequent system for **S4.3** from our tableau system $\mathcal{C}S4.3$ using the relationship between sequents and associated sets from Section 3.11 on page 49.

Wallen [Wal87] is probably the most recent work on sequent systems for modal logics but the actual systems are taken directly from the prefixed tableau calculi of Fitting [Fit83] as Wallen's interest is not in the sequent systems themselves but with techniques to allow efficient implementation of the connection matrix method for modal logics.

Syntactic cut-elimination proofs for various logics with provability interpretations have been given by Bellin [Bel85a], Sambin and Valentini [SV80, SV82], Valentini [Val86], Valentini and Solitro [VS83], Borga [Bor83], Borga and Genitlini [BG86] and Avron [Avr84].

Borga [BG86] gives a syntactic proof of cut-elimination for **Grz** and credits Avron [Avr84] with the semantic proof. Rautenberg's work predates this work of Avron. Bellin [Bel85a], Borga [Bor83] and Valentini [Val83] give syntactic proofs of cut-elimination for **G** although they call this logic **GL**. Boman [Bom90] gives a recent survey of provability interpretation of **G**. Valentini [Val86] gives a syntactic proof of cut-elimination for **KGL** although he calls this logic $\mathbf{GL_{lin}}$.

7.3 Discussion

As we have seen, tableau and sequent systems for modal logics have been an active area of research for over twenty five years. The early tableau constructions tended to become confused due to the use of alternative sets of tableaux to explicitly maintain the search space. The early methods were also global methods where we can return to previous tableau nodes at will. In this respect they are essentially the deterministic decision procedures we obtain from our completeness proofs. The modern tableau methods, as espoused by Rautenberg [Rau83], are highly nondeterministic and only manipulate one tableau construction, using backtracking to explore the search space. Consequently, we cannot return to previous nodes at will and have to build the information required by descendants into a node at the time it is processed. For some logics, this requires building up rules.

Rautenberg shows two things. First, that analytic cut is enough as long as we allow some sort of building up rule. But he also shows that for many logics, the building up rule can remain analytic and hence give decidability. We have independently discovered this property for the logics **S4.3.1** and **S4.14** but clearly our work is not as general as Rautenberg's.

Chapter 8

Further Work

In this chapter we present incomplete work and suggest avenues for further work.

8.1 Tableau Systems For Other Propositional Modal Logics

In the following sections we consider ways to extend the tableau method to other propositional normal modal logics, to propositional tense logics and to propositional temporal logics. We refer to a logic as a tense logic if it involves both future and past tense modalities. We refer to a logic as a temporal logic if it involves only future (or only past) modalities. This nomenclature is purely for convenience.

8.1.1 Logics of Convergent Frames

One of the most glaring omissions in this dissertation is the absence of logics involving the axiom 2:

$$\Diamond\Box A \Rightarrow \Box\Diamond A.$$

It is well known [Gol87] that this axiom characterises weakly-directed frames where a frame is weakly-directed if it satisfies:

$$\forall s, t, u(sRt \wedge sRu \Rightarrow \exists v(tRv \wedge uRv)).$$

Such frames are also called convergent frames and the most famous such logic is the logic **S4.2**, axiomatised as **KT42**. Rautenberg [Rau83] gives a cut-free system for **S4.2** but we have avoided discussing it because the rule is not strictly "once off". Further work is required to understand the nuances of **S4.2** and related logics of convergent frames.

8.1.2 Logics of Strict Linear Frames

The axiom that imposes linearity on a frame is

$$L : \Box((A \wedge \Box A) \Rightarrow B) \vee \Box((B \wedge \Box B) \Rightarrow A)$$

since any frame that validates L must be weakly-connected by Theorem 2 (page 27).

We know that **S4.3**-frames and **S4.3.1**-frames are linear, but the clusters are always nondegenerate. Using tableau rules analogous to those used by Valentini [Val86] for **KGL** (see page 102) we believe that it is possible to obtain cut-free tableau and sequent systems for the logics **K4L**, **K4DL** and **K4DLZ**. These logics are all characterised by frames involving degenerate clusters rather than nondegenerate clusters because the reflexivity axiom T is missing; see page 33.

8.1.3 Logics Close to S5

We believe that the logic **S4.4** axiomatised as **KT4R** (see page 24) is also amenable to our tableau methods. The characteristic axiom R is [Zem73]:

$$\Diamond\Box A \Rightarrow (A \Rightarrow \Box A).$$

This means that the following are also theorems of **S4.4**:

$$\begin{array}{rcl}
\Diamond\Box A \wedge A & \Rightarrow & \Box A \\
\neg\Box A & \Rightarrow & \neg\Diamond\Box A \vee \neg A \\
\neg\Box A & \Rightarrow & \neg\neg\Box\neg\Box A \vee \neg A \\
\neg\Box A & \Rightarrow & \Box\neg\Box A \vee \neg A
\end{array}$$

which suggests the tableau rule:

$$(S4R)\ \frac{X; \neg\Box P}{X; \Box\neg\Box P \mid X; \neg P; \neg\Box P}$$

We believe that a cut-free tableau system for **S4.4** is possible using this rule as a basis. The logic **S4.4** is characterised by finite frames that consist of a sequence of at most two nondegenerate clusters. Zeman [Zem73] gives a tableau system for **S4.4** but we have not made detailed comparisons.

8.1.4 Logics of Finite Linear Sequences

The logic **Grz.3** is axiomatised as **KGrz.3** and is equal to **S4.3Grz**. We conjecture that it is characterised by finite linear sequences of simple clusters. Borga and Gentilini [BG86] give a syntactic cut-elimination proof for **Grz**. Using our experience from **S4.3** we would like to obtain cut-free tableau systems for this logic as well; see Section 6.5 (page 122).

8.1.5 Propositional Linear Temporal Logics

The logic **S4.3.1** is characterised by the single frame $\langle \omega, \leq \rangle$ and there is a natural notion of immediate successor in **S4.3.1**, although the modalities $\Box$ and $\Diamond$ cannot be used to express this notion of "next" [Kam68]. One of the most frequently used linear temporal logics is based on **S4.3.1** by adding an extra modal operator $\bigcirc$ to represent this notion of "next".

The logic **DX** is the most basic such logic and it can be axiomatised in various ways [GPSS80, Wol83, Gol87]. The following axiomatisation is taken from Goldblatt [Gol87]:

K $\Box(A \Rightarrow B) \Rightarrow (\Box A \Rightarrow \Box B)$;

$K_\circ$ $\bigcirc(A \Rightarrow B) \Rightarrow (\bigcirc A \Rightarrow \bigcirc B)$;

Fun $\bigcirc \neg A \iff \neg \bigcirc A$;

Mix $\Box A \Rightarrow A \wedge \Box A$;

Ind $\Box(A \Rightarrow \bigcirc A) \Rightarrow (A \Rightarrow \Box A)$.

The semantics of **DX** is based on finite frames of the form $\langle W, \sigma \rangle$ where W is a finite (non-empty) set of worlds and $\sigma : W \mapsto \omega$ is a mapping from W onto the natural numbers [Gol87]. That is, we can imagine W to be an ordered *infinite* set $W = \{w_0, w_1, \cdots\}$ and put $w_i R_\circ w_{i+1}$ for all natural numbers i where $R_\circ$ is an *intransitive* and *irreflexive* reachability relation. The semantics of $\bigcirc A$ is given by:

$$w_i \models \bigcirc A \text{ iff } w_{i+1} \models A.$$

Since $\Box$ and $\Diamond$ are also present in **DX**, we have to work with $R_\circ$ and use the reflexive and transitive closure $R_\circ^*$ of $R_\circ$ to obtain these modalities [Gol87]. The semantics of $\Box$ and $\Diamond$ become:

$$\begin{aligned} w_i \models \Box A & \quad \text{iff} \quad \forall j \geq i, w_j \models A; \\ w_i \models \Diamond A & \quad \text{iff} \quad \exists j \geq i, w_j \models A. \end{aligned}$$

Using the experience gained from our systems for **S4.3.1**, it seems that the **DX** axiom *Ind* which can be written as

$$\neg\Box A \Rightarrow \neg\Box(A \Rightarrow \bigcirc A) \vee \neg A$$

would give a tableau rule like

$$(Ind)\ \frac{U; \Box X; \neg\Box P}{\Box X; P; \neg\bigcirc P \mid U; \Box X; \neg P; \neg\Box P}$$

The left branch is an arbitrary "jump" into the future to the world immediately preceding the world that satisfies $\neg P$. The right branch is a static assumption that $\neg\Box P$ is fulfilled in the same world as the numerator. Bellin [Bel85b] uses a similar rule in sequent form, shown below,

$$\frac{\Box\Gamma \longrightarrow \bigcirc P}{P, \Box\Gamma \longrightarrow \Box P}$$

and shows that his sequent system for **DX** is not complete unless the cut rule is added.

There are three problems with both these rules. The first is that they ignore eventualities of the form $\bigcirc A$. The second is that the jump is arbitrary. To create a **DX**-model, we have to build the model one state at a time, always jumping only one state forward using $R_\circ$ [GPSS80, Gol87]. The arbitrary jump involved in the (Ind) rule means that we cannot be sure that all eventualities are fulfilled by the model we construct. Wolper handles this with a two stage procedure that explicitly checks that all eventualities are fulfilled [Wol83]. The third is that $\neg\bigcirc P$ is also an eventuality so that one eventuality has spawned another. There is no guarantee that this process will not continue ad infinitum.

Another rule we have explored can be derived as follows. Since *Ind* is an axiom, the formula

$$\Box(P \Rightarrow \bigcirc P) \Rightarrow (\bigcirc P \Rightarrow \bigcirc\Box P)$$

is a theorem of **DX**. Then so are each of

$$\begin{aligned}
\bigcirc P &\Rightarrow (\Box(P \Rightarrow \bigcirc P) \Rightarrow \bigcirc\Box P)\\
\bigcirc P &\Rightarrow \neg\Box(P \Rightarrow \bigcirc P) \vee \bigcirc\Box P\\
\bigcirc P &\Rightarrow \Diamond(P \wedge \neg\bigcirc P) \vee \bigcirc\Box P\\
\bigcirc P &\Rightarrow \Diamond(P \wedge \bigcirc\neg P) \vee \bigcirc\Box P
\end{aligned}$$

giving a rule:

$$(DX)\ \frac{U; \Box X; \bigcirc P}{U; \Box X; \bigcirc\Box P \mid \Box X; P; \bigcirc\neg P}$$

But again, this rule ignores other eventualities, the transition involved is an arbitrary one and the eventuality $\bigcirc P$ spawns two eventualities $\bigcirc\Box P$ and $\bigcirc\neg P$.

Goldblatt [Gol87, page 73] shows that both *Dum* and L are theorems of **DX**, which is not surprising since they together with T and 4 characterise discrete linear frames. Goldblatt then gives an axiomatic completeness proof for **DX** using filtrations through a superset of $Sf(A)$ where A is any non-theorem of **DX**. His proofs make critical use of the fact that both Dum and L are theorems of **DX**.

We have already seen the effects of Dum and L and therefore believe that the tableau system, $\mathcal{C}DX$ for **DX** will require some version of the $(S4.3)$ and $(S4.3.1)$ rules. That is, any cut-free tableau system for **DX** must involve rules like (Ind) and (DX), but they must cater for the interaction of eventualities as in the $(S4.3)$ and $(S4.3.1)$ rules. These considerations would remove objections one and two mentioned above. But I have no idea how to ensure that one eventuality does not spawn another one. A possible solution is that the (Ind) rule causes an eventuality of the form $\neg\Box P$ to spawn an eventuality $\neg\bigcirc P$. As long as $\neg\bigcirc P$ does not in turn spawn an eventuality of the form $\neg\Box Q$ then we are safe. That is, this amounts to a gradual reduction in the number of eventualities and may remain analytic.

On the other hand, it may turn out that the explicit induction axiom destroys cut-elimination for **DX**. That is, we either have to resort to an (ω) rule or to a cut rule. The (ω) rule could be made semi-analytic since we know that for a finite set, we only need a finite model. That is, we could guess the width of the (ω) rule. On the other hand, we have seen that the cut rule often remains analytic in the guise of (sfc) and $(sfcT)$. We intend to pursue these matters as further work.

8.1.6 Propositional Temporal Logics

Just as **S4.3.1** is the modal basis for **DX**, the logic **S4.14** is the modal basis for the branching temporal logics of the **CTL** family [EH85]. The insights obtained via the work on **DX** may shed light on the cut-elimination theorem for **CTL**. Tableau-like decision procedures for **CTL** have been given by Emerson and Halpern [EH85] but they do not give analogous sequent systems.

8.1.7 First Order Normal Modal Logics

Fitting [Fit83] and Rautenberg [Rau83] show that König's lemma can be used to obtain soundness and completeness results for certain first order normal modal logics by simply adding the two rules:

$$(\forall)\frac{X; \forall x P(x)}{X; \forall x P(x); P(a)} \text{ where } a \text{ is any constant}$$

and

$$(\exists)\frac{X; \neg\forall x P(x)}{X; P(c)} \text{ where } c \text{ is any new constant}$$

as long as we work with increasing domains only. That is, if wRw' then $D(w) \subseteq D(w')$.

Fitting notes a result of Fine [Fin79] regarding the interpolation theorem and Beth's definability lemma for first order **S5** which indicates that "there can be no "reasonable" cut-free tableau systems for such logics ..." [Fit83, page 383]. But it seems plausible that a reasonable system may be possible where all uses of the cut rule are *analytic*. That is, it may be possible to make some headway using our knowledge about the utility of (sfc) and $(sfcT)$.

8.1.8 First Order Linear Temporal Logics

Recent results of Szalas [Sza86, Sza87] show that the first order linear temporal logic based on **DX** with function symbols and equality cannot have a complete finitary axiomatisation. Thus there is likely to be no cut-free and complete tableau system for this logic. Valentini [Val83] refers to the fact that the first order version of **G** also cannot have a cut-free and complete sequent system [Avr84]. Došen [Doš85] also shows that cut is not eliminable from certain systems between **S4** and **S5**. Thus it seems possible to *prove* that a logic can have no cut-free sequent system. Is it possible to apply these methods to propositional **DX** ?

There is a surprising jump in complexity associated with the $\bigcirc$ operator. This is the price we have to pay for the extra expressiveness it brings, but in general, the $\bigcirc$ operator seems to be too expressive. However, the $\bigcirc$ operator is indispensable for certain aspects of timing in hardware verification.

8.2 Syntactic Cut Elimination

As stated previously, syntactic cut-elimination is much harder than semantic cut-elimination. Now that we have cut-elimination for **S4.3**, **S4.3.1** and **S4.14**, we intend to seek syntactic cut-elimination proofs of these results. We believe that the essential idea is to introduce a third parameter, called the width, in addition to the rank and weight of a cut as is already done by Valentini [Val86]. The name is in itself suggestive of the branching inherent in the $(S4.3)$ rule.

8.3 Normal Forming Techniques

Kit Fine [Fin75] proves the finite model property for **T** and **K4** using normal forms. Hughes and Cresswell [HC84, page 162] claim that Fine's normal forming methods may be applied to all the systems in that section of their book. Hence it may be possible to obtain decision procedures for some of our logics by generalising the work of Fine although a reduction to a normal form is essential for this to work.

8.4 Filtration Proofs of Completeness

The standard method for proving axiomatic completeness and the finite model property is to use canonical models and filtrations [Gol87]. Given some formula A which is not an **L**-theorem, a filtration through the set $Sf(A)$ is usually sufficient because of the direct correspondence between characteristic axioms and properties of R. For some logics, like **S4.3.1**, a finite superset of $Sf(A)$ is required and the filtration proofs are not so simple [Gol87, pages 58-59]. Thus, superformulae also appear in the filtration proofs.

Is there a connection between the filtration set for **L** and X_L^* ? If so, then it may be possible to obtain simpler filtration proofs using a filtration through X_L^* rather than a filtration through $Sf(A)$.

8.5 Interpolation and Compactness

The logic **S4.3** and **S4.3.1** are the only logics where there is any interaction between eventualities and this is clearly because R is connected in these logics. However there may be a deeper reason.

The interpolation theorem for a logic **L** says that if $A \Rightarrow C$ is **L**-valid, then there exists a formula B such that both $A \Rightarrow B$ and $B \Rightarrow C$ are **L**-valid, and such that all the propositional variables of B appear in A and also appear in C. We write $vars(B) = vars(A) \cap vars(C)$ to indicate the condition on propositional variables.

It is known that the interpolation theorem fails to hold for **S4.3**. That is, there is at least one formula $A \Rightarrow C$ such that $A \Rightarrow C$ is **S4.3**-valid, but for which there is no B such that both $A \Rightarrow B$ and $B \Rightarrow C$ are **S4.3**-valid and $vars(B) = vars(A) \cap vars(C)$.

This could explain why the tableau rules for **S4.3** are so different from those for other systems. In fact, it may be that linearity is responsible for the failure of the interpolation theorem. For example, it is also known that **KGL** [Rau83] fails to have interpolation, although **G** itself has interpolation.

Rautenberg [Rau83] gives a very general class of tableau rules from which he proves interpolation for most of the logics we have dealt with. A detailed analysis of why interpolation fails for **S4.3** may explain the bizarre nature of the $(S4.3)$ rule.

The one term I have used without explanation until now is the notion of compactness and for this we need the notion of **L**-consistent sets.

A formula A is **L**-consistent iff its negation $\neg A$ is not an **L**-theorem [HC84, page 17]. That is, A is **L**-consistent iff there is no closed CL-tableau for $\{A\}$, and hence iff A is **L**-satisfiable. When X is a finite set, X is **L**-consistent iff X is **L**-satisfiable, that is, iff there is no closed CL-tableau for X. But when X is an infinite set, we must use a different notion. A (possibly infinite) set X is **L**-consistent iff all finite subsets of X are

L-satisfiable [HC84, page 17]. Now it can happen that an infinite set X is **L**-consistent but it is not **L**-satisfiable. That is, although every finite subset of X is **L**-satisfiable, X itself may not be **L**-satisfiable.

A logic **L** is said to be **compact** if every **L**-consistent set is **L**-satisfiable.

It is known that **S4.3.1** is not compact [HC84, page 109]. The non-compactness follows directly from the fact that in **S4.3.1**, the infinite set

$$\{\Diamond\Box P, P, \Diamond\neg P, \Diamond(P \wedge \Diamond\neg P), \Diamond(\neg P \wedge \Diamond(P \wedge \Diamond\neg P)), \Diamond(P \wedge \Diamond(\neg P \wedge \Diamond(P \wedge \Diamond\neg P))), \cdots\}$$

is not **S4.3.1**-satisfiable, and yet any finite subset of it is.

There appears to be some connection between compactness, interpolation and the fact that (θ) is not eliminable from $\mathcal{C}S4.3$ and $\mathcal{C}S4.3.1$.

Chapter 9

Conclusions

We have presented a unified treatment of tableau, sequent and axiomatic formulations for many propositional normal modal logics by concentrating on finite frame characterisation results, thereby unifying the work of Hanson, Segerberg, Zeman, Mints, Fitting, Rautenberg and Shvarts.

Independently, we have found cut-free tableau systems for **S4.3**, **S4.3.1** and **S4.14**, of which the last two appear to be new, and shown that Gentzen's cut-elimination theorem holds for these logics. We have also made some progress towards a cut-free tableau system for **DX**, the extension of **S4.3.1** with an explicit next-time operator. Our tableau systems are clearly just extensions of Rautenberg's method and have been presented in this light.

All our tableau systems are sound and weakly complete with respect to their known Kripke semantics and each tableau and sequent system serves as a nondeterministic decision procedure for the logic it formulates. The sequent analogues of our tableau systems give a finitary syntactic deducibility relation $\vdash_L$ so that any sequent proof can be read downwards to give an axiomatic proof of the endsequent. Furthermore, the proofs of tableau completeness are all constructive and yield deterministic decision procedures for each logic.

All our tableau rules are based on some sort of semantic insight into the consequences of $\neg\Box P$ being true at the numerator. The characteristic axioms of the logic seem to contain all the essential information for obtaining these tableau rules although complications arise when linearity or symmetry are present.

Two of the strongest tenets of classical logic are the subformula property and cut-elimination. We have seen that there is a systematic way to break the subformula property and replace it with an *analytic* superformula property so that the resulting tableau systems remain tractable for computer implementation. The tableau systems $\mathcal{C}S4.3.1$ and $\mathcal{C}S4.14$ also involve a form of implicit cut on the superformula $\Box\neg\Box P$, although its dual $\neg\Box\neg\Box P$ never appears in any tableau node (for then it would be an explicit cut). Thus we see a gradual introduction of some sort of cut rule as we traverse the series of logics between **S4** and **S5**.

An explicit cut rule appears to be essential for modal tableau and sequent systems for logics with a symmetric reachability relation although for some logics, it can be replaced by an *analytic* cut rule. For some logics, like **B**, we require both the analytic cut rule and the analytic superformula property although we have shown that the subformula property can be regained for **S5**. Pure symmetry, that is **KB**, has still proved elusive. Induction, that is **DX**, also poses problems although we have made some progress in this regard.

Adding the analytic cut rule to tableau systems where it is not essential also helps us as shown by the systems of Hanson and the alternatives for **K45** and **K45D** based on them.

We therefore have the following theorems.

Theorem 21 *Each* **L** $\in$ { **K**, **T**, **D**, **K4**, **D4**, **S4**, **K45**, **K45D**, **B**, **S5**, **G**, **Grz**, **S4.3**, **S4.3.1**, **S4.14** } *is decidable and each* $\mathcal{C}L$ *is a decision procedure for* **L**.

Proof: By the soundness and completeness theorems $\mathcal{C}L$ characterises the logic **L**. If A is not an **L**-theorem then the completeness proof yields a finite **L**-model for $\neg A$ proving that each **L** has the finite model property. Since each **L** is finitely axiomatisable, Theorem 4, page 30, implies that **L** must be decidable. ●

Theorem 22 *Gentzen's cut-elimination theorem holds for each* **L** $\in$ { **K**, **T**, **D**, **K4**, **D4**, **S4**, **K45**, **K45D**, **G**, **Grz**, **S4.3**, **S4.3.1**, **S4.14** }.

For some of our tableau systems, thinning seems essential. We believe that this is related to the failure of the interpolation theorem. What does it mean for a logic not to have interpolation ?

Bibliography

[AE89] Y. Auffray and P. Enjalbert. Modal theorem proving: An equational viewpoint. In *11th International Joint Conference on Artificial Intelligence*, pages 441–445, 1989.

[AM90] M. Abadi and Z. Manna. Nonclausal deduction in first-order temporal logic. *JACM*, 37(2):279–317, 1990.

[AMCP84] P. B. Andrews, D. A. Miller, E. Cohen, and F. Pfenning. Automating higher order logic. In W. W. Bledsoe and D. E. Loveland, editors, *25 Years of Theorem Proving,*. American Mathematical Society, Contemporary Mathematics Series, Vol. 29, 1984.

[And81] P. B. Andrews. Theorem proving via general matings. *JACM*, 28(2):193–214, 1981.

[Avr84] Arnon Avron. On modal systems having arithmetical interpretations. *Journal of Symbolic Logic*, 49:935–942, 1984.

[BC67] Robert Binkley and Romane Clark. A cancellation algorithm for elementary logic. *Theoria*, 33:79–97, 1967.

[Bel85a] G. Bellin. A system of natural deduction for GL. *Theoria*, 51:89–114, 1985.

[Bel85b] G. Bellin. Unpublished notes on sequent systems for temporal logics, March 1985.

[Bet53] E. W. Beth. On Padoa's method in the theory of definition. *Indag. Math.*, 15:330–339, 1953.

[Bet55] E. W. Beth. Semantic entailment and formal derivability. *Mededelingen der Koninklijke Nederlandse Akademie van Wetenschappen, Afd. Letterkunde*, 18:309–342, 1955.

[BG86] M. Borga and P. Gentilini. On the proof theory of the modal logic Grz. *ZML*, 32:145–148, 1986.

[Bib81] W. Bibel. On matrices with connections. *JACM*, 28(4):633–645, 1981.

[Bom90] Magnus Boman. A survey of provability logic and a note on its relevance to nonmonotonic federated information systems. Technical report, The Royal Institute of Technology and Stockholm University, Sweden, 1990.

[Boo79] G. Boolos. *The Unprovability of Consistency.* Cambridge University Press, 1979.

[Boo84] George Boolos. Don't eliminate cut. *Journal of Philosophical Logic*, 13:373–378, 1984.

[Bor83] M. Borga. On some proof theoretical properties of the modal logic GL. *Studia Logica*, 42:453–459, 1983.

[BS84] R. A. Bull and K. Segerberg. Basic modal logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic, Volume II: Extensions of Classical Logic*, pages 1–88. D. Reidel, 1984.

[Bul65] R. A. Bull. An algebraic study of Diodorean modal systems. *Journal of Symbolic Logic*, 30(1):58–64, 1965.

[Bul85] R. A. Bull. Review of 'Melvin Fitting, Proof Methods for Modal and Intuitionistic Logics, Synthese Library, Vol. 169, Reidel, 1983'. *JSL*, 50:855–856, 1985.

[Bür90] H-J. Bürckert. A resolution principle for clauses with constraints. In *Proc. 10th International Conference on Automated Deduction, LNCS 449*, pages 178–192. Springer-Verlag, 1990.

[CF86] Marta Cialdea and Luis Fariñas Del Cerro. A modal Herbrand's property. *ZML*, 32:523–530, 1986.

[Cha86] Man-Chung Chan. Reasoning in a logic of linear time using the recursive resolution priciple. Technical Report 2/86, La Trobe University, Melbourne, Australia, 1986.

[Cha87] Man-Chung Chan. The recursive resolution method for modal logic. *New Generation Computing*, 5:155–183, 1987.

[Che80] B. F. Chellas. *Modal Logic: An Introduction.* Cambridge University Press, 1980.

[Clo87] W. F. Clocksin. Principles of the DelPhi parallel inference machine. *The Computer Journal*, 30(5):386–392, 1987.

[Cre79] M. J. Cresswell. BSeg has the finite model property. *Bulletin of the section of logic, Polish Academy of Sciences*, 8:154–160, 1979.

[Cur50] H. B. Curry. A theory of formal deducibility. Technical Report Number 6, University of Notre Dame, USA, 1950.

[Cur52] H. B. Curry. The elimination theorem when modality is present. *Journal of Symbolic Logic*, 17:249–265, 1952.

[dA81] Newton C. A. da Costa and E. H. Alves. Relations between paraconsistent logic and many-valued logic. *Bulletin of the Section on Logic, Polish Academy of Sciences*, 10:185–191, 1981.

[D'A90] Marcello D'Agostino. *Investigations into the complexity of some propositional calculi.* PhD thesis, Oxford University Computing Laboratory, 1990.

[Den81] N. Denyer. Time and modality in Diodorus Cronus. *Theoria*, 47:31–53, 1981.

[DL59] M. Dummett and E. J. Lemmon. Modal logics between S4 and S5. *ZML*, 5:250–264, 1959.

[Doš85] Kosta Došen. Sequent-systems for modal logic. *Journal of Symbolic Logic*, 50(1):149–169, 1985.

[DP60] Martin Davis and Hilary Putnam. A computing procedure for quantification theory. *JACM*, 7:201–215, 1960.

[Ede84] Elmar Eder. An implementation of a theorem prover based on the connection method. In *Proc. Artifical Intelligence, Methodology, Systems and Applications (AIMSA'84), Bulgaria, 1984*, pages 121–128. North-Holland, 1984.

[Ede88] Elmar Eder. A comparison of the resolution calculus and the connection method, and a new calculus generalizing both methods. In *Proc. 2nd Workshop on Computer Science Logics, LNCS 385*, pages 80–98. Springer-Verlag, 1988.

[EF89] P. Enjalbert and L. Fariñas Del Cerro. Modal resolution in clausal form. *Theoretical Computer Science*, 65:1–33, 1989.

[EH85] E. A. Emerson and J. Y. Halpern. Decision procedures and expressiveness in the temporal logic of branching time. *Journal of Computer and System Sciences*, 30:1–24, 1985.

[EH86] E. A. Emerson and J. Halpern. 'sometime' and 'not never' revisted: On branching versus linear time temporal logic. *JACM*, pages 151–178, 1986.

[Far85] L. Fariñas Del Cerro. Resolution modal logics. In K. R. Apt, editor, *Logics and Models of Concurrent Systems, NATO ASI Series, Vol. F13*, pages 123–144. 1985.

[FH88] L. Fariñas Del Cerro and A. Hertzig. Linear modal deductions. In E. Lusk and R. Overbeek, editors, *Proceedings, Conference on Automated Deduction.* Springer-Verlag, 1988. LNCS 310.

[Fin75] Kit Fine. Normal forms in modal logic. *Notre Dame Journal of Formal Logic*, 16(2):229–237, 1975.

[Fin79] K. Fine. Failures of the interpolation lemma in quantified modal logic. *Journal of Symbolic Logic*, 44(2):201–206, June 1979.

[Fit66] F. B. Fitch. Natural deduction rules for obligation. *American Philosophical Quarterly*, 3:27–38, 1966.

[Fit73] Melvin Fitting. Model existance theorems for modal and intuitionistic logics. *Journal of Symbolic Logic*, 38:613–627, 1973.

[Fit83] M. Fitting. *Proof Methods for Modal and Intuitionistic Logics*, volume 169 of *Synthese Library*. D. Reidel, Dordrecht, Holland, 1983.

[Fit88] M. Fitting. First order modal tableaux. *Journal of Automated Reasoning*, 4:191–213, 1988.

[Fit90] M. Fitting. Destructive modal resolution. *Journal of Logic and Computation*, 1(1):83–97, 1990.

[FS91] Alan M. Frisch and Richard B. Scherl. A general framework for modal deduction. In J. Allen, R. Fikes, and E. Sandewall, editors, *Proc. 2nd Conference on Principles of Knowledge Representation and Reasoning.* Morgan-Kaufmann, 1991.

[Gal87] J. H. Gallier. *Logic for Computer Science: Foundations of Automatic Theorem Proving.* John Wiley and Sons, 1987.

[Gen35] G. Gentzen. Unterschungen über das logische schliessen. *Mathematische Zeitschrift*, 39:176–210 and 405–431, 1935. English translation: Investigations into logical deduction, in The Collected Papers of Gerhard Gentzen, edited by M. E. Szabo, pp 68-131, North-Holland, 1969.

[Gen91] Ian Gent. *Analytic Proof Systems for Classical and Modal Logics of Restricted Quantification.* PhD thesis, Dept. of Computer Science, University of Warwick, Coventry, England, 1991.

[Gil60] Paul C. Gilmore. A proof method for quantification theory. *I.B.M. Journal of Research and Development*, 4:28–35, 1960.

[Gir87] J-Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[GK86] C. Geissler and K. Konolige. A resolution method for quantified modal logics of knowledge and belief. In J. Halpern, editor, *Proceedings Theoretical Aspects of Reasoning about Knowledge*, 1986.

[Gol87] R. I. Goldblatt. *Logics of Time and Computation.* CSLI Lecture Notes Number 7, CSLI Stanford, 1987.

[Gou84] G. Gough. Decision procedures for temporal logics. Master's thesis, Dept. of Computer Science, University of Manchester, England, 1984.

[GPSS80] D. Gabbay, A. Pnueli, S. Shelah, and J. Stavi. On the temporal analysis of fairness. In *Proc. 7th Conference on Principles of Programming Languages*, pages 165–173, 1980.

[Grz67] Andrzej Grzegorczyk. Some relational systems and the associated topological spaces. *Fundamenta Mathematicae*, 60:223–231, 1967.

[Han66a] William Hanson. Termination conditions for modal decision procedures (abstract only). *Jornal of Symbolic Logic*, 31:687–688, 1966.

[Han66b] William H. Hanson. On some alleged decision procedures for S4. *Journal of Symbolic Logic*, 31:641–643, 1966.

[HC68] G. E. Hughes and M. J. Cresswell. *Introduction to Modal Logic*. Methuen, London, 1968.

[HC84] G. E. Hughes and M. J. Cresswell. *A Companion to Modal Logic*. Methuen, London, 1984.

[Hen49] Leon Henkin. The completeness of the first-order functional calculus. *Journal of Symbolic Logic*, 14:159–156, 1949.

[Hin55] K. J. J. Hintikka. Form and content in quantification theory. *Acta Philosophica Fennica*, 8:3–55, 1955.

[Hin63] K. J. J. Hintikka. The modes of modality. *Acta Philosophica Fennica, Proceedings of a colloquium on modal and many-valued logics 1962*, 16:65–81, 1963.

[HM85] J. Y. Halpern and Y. Moses. A guide to the modal logics of knowledge and belief: Preliminary draft. In *Proc. IJCAI*, pages 480–490, 1985.

[JR87] P. Jackson and H. Reichgelt. A general proof method for first-order modal logic. In *9th International Joint Conference on Artificial Intelligence*, pages 942–944, 1987.

[JR88] P. Jackson and H. Reichgelt. A general proof method for modal predicate logic without the Barcan formula. In *Proc. AAAI*, pages 177–181, 1988.

[Kam68] Johan Anthony Willem Kamp. *Tense Logic and the Theory of Linear Order*. PhD thesis, Dept. of Philosophy, University of California, USA, 1968.

[Kan57] S. Kanger. *Provability in Logic*. Stockholm Studies in Philosophy, University of Stockholm, Almqvist and Wiksell, Sweden, 1957.

[Kap66] D. Kaplan. Review of S. A. Kripke. Semantical analysis of modal logic I. Normal modal propositional calculi. Zeitschrift für mathematische Logik und Grundlagen der Mathematik, vol 9, (1963), pp-67-96. *Journal of Symbolic Logic*, 31:120–122, 1966.

[Kle52] S. C. Kleene. Permutability of inferences in Gentzen's calculi LK and LJ. *Memoirs of the American Mathematical Society*, 10:1–26, 1952.

[Kle90] C. S. Klein. *Exploiting OR-Parallelism in Prolog Using Multiple Sequential Machines*. PhD thesis, Computer Laboratory, University of Cambridge, England, 1990.

[Kri59] Saul Kripke. A completeness theorem in modal logic. *Journal of Symbolic Logic*, 24(1):1–14, March 1959.

[Kri63a] S. Kripke. Semantical considerations on modal logic. *Acta Philosophica Fennica, Proceedings of a colloquium on modal and many-valued logics 1962*, 16:83–94, 1963.

[Kri63b] Saul Kripke. Semantical analysis of modal logic I: Normal modal propositional calculi. *Zeitschrift für Mathematik Logik und Grundlagen der Mathematische*, 9:67–96, 1963.

[Kri65] S. Kripke. Semantical analysis of modal logic II: Non-normal modal propositional calculi. In *Symposium on the Theory of Models*, pages 206–220. North-Holland, Amsterdam, 1965.

[Lew20] C. I. Lewis. Strict implication: An emendation. *The Journal of Philosophy*, 17:300–302, 1920.

[Lif89] Vladimir Lifschitz. What is the inverse method. *Journal of Automated Reasoning*, 5:1–23, 1989.

[LS77] E. J. Lemmon and D. Scott. *An Introduction To Modal Logic.* American Philosophical Quarterly, Monograph Series, Basil Blackwell, Oxford, 1977.

[Mak73] David Makinson. A warning about the choice of primitive operators in modal logic. *Jornal of Philosophical Logic*, 2:193–196, 1973.

[Mat55] K. Matsumoto. Reduction theorem in Lewis's sentential calculi. *Mathematica Japonica*, 3:133–135, 1955.

[McA88] G. L. McArthur. Reasoning about knowledge and belief: a survey. *Computational Intelligence*, 4(3):223–242, 1988.

[McD82] D. McDermott. Nonmonotonic logic II: Nonmonotonic modal theories. *JACM*, 29:33–57, 1982.

[Min] G. E. Minc. See G. E. Mints.

[Min70] G. E. Mints. Cut-free calculi of the S5 type. In *Studies in constructive mathematics and mathematical logic, Part II, Seminars in Mathematics*, pages 115–120. Steklov Institute, USSR, 1970. English translation from the American Mathematical Society.

[Min90a] G. E. Mints. Gentzen-type systems and resolution rules part I: Propositional logic. *Proc. COLOG 88*, 417:198–231, 1990.

[Min90b] G. E. Mints. Gentzen-type systems and resolution rules part II: Predicate logic. *Proc. Logic Colloquim*, 1990.

[Moo85] R. C. Moore. Semantical considerations on nonmonotonic logic. *Artificial Intelligence*, 25:272–279, 1985.

[Mor76] C. G. Morgan. Methods for automated theorem proving in nonclassical logics. *IEEE Transactions on Computers*, C-25(8):852–862, 1976.

[MT48] J. C. C. McKinsey and Alfred Tarski. Some theorems about the sentential calculi of Lewis and Heyting. *Journal of Symbolic Logic*, 13:1–15, 1948.

[Ohl90] H-J. Ohlbach. Semantics based translation methods for modal logics. Technical Report SEKI Report SR-90-11, Universität Kaiserslautern, Postfach, 3049, D-6750, Kaiserslautern, Germany, 1990.

[OM57a] M. Ohnishi and K. Matsumoto. Corrections to our paper 'Gentzen method in modal calculi I'. *Osaka Mathematical Journal*, 10:147, 1957.

[OM57b] M. Ohnishi and K. Matsumoto. Gentzen method in modal calculi I. *Osaka Mathematical Journal*, 9:113–130, 1957.

[OM59] M. Ohnishi and K. Matsumoto. Gentzen method in modal calculi II. *Osaka Mathematical Journal*, 11:115–120, 1959.

[Orc89] Terttu Orci. Clause graph theorem proving in modal temporal logic Q. Technical Report UMINF-169.89, University of Umea, S-901 87 Umea, Sweden, 1989.

[PHV60] Dag Prawitz, D. Hakan, and Neri Vogera. A mechanical proof procedure and its realization in an electronic computer. *JACM*, 7:102–128, 1960.

[Pnu77] A. Pnueli. The temporal logic of programs. In *Proceedings of 18th IEEE Annual Symposium on the Foundations of Computer Science*, 1977.

[Pra60] Dag Prawitz. An improved proof procedure. *Theoria*, 26:109–139, 1960.

[Pra79] V. R. Pratt. Process logic: Preliminary report. In *Proc. 6th Annual ACM Symposium on Principle Of Programming Languages*, pages 93–100, January 1979.

[Pri57] A. Prior. *Time and Modality.* Oxford University Press, 1957.

[Qui61] W. V. Quine. *From a logical point of view.* Cambridge, Massachussetts, 1961.

[Qui76] W. V. Quine. *The Ways of Paradox.* Cambridge, Massachussetts, 1976.

[Rau79] W. Rautenberg. *Klassicshe und Nichtklassicshe Aussagenlogik.* Vieweg, Wiesbaden, 1979.

[Rau83] W. Rautenberg. Modal tableau calculi and interpolation. *JPL*, 12:403–423, 1983.

[Rau85] W. Rautenberg. Corrections for modal tableau calculi and interpolation by W. Rautenberg, JPL 12 (1983). *Journal of Philosophical Logic*, 14:229, 1985.

[Rau90] W. Rautenberg. Personal communication, December 5th, 1990.

[Rob65] A. Robinson. A machine oriented logic based on the resolution principle. *Journal of the ACM*, 12:23–41, 1965.

[Rob79] J. A. Robinson. *Logic: form and function (the mechanization of deductive reasoning).* Edinburgh University Press, 1979.

[SC85] A. P. Sistla and E. M. Clarke. The complexity of propositional linear temporal logics. *JACM*, 32(3):733–749, 1985.

[Seg71] Krister Segerberg. An essay in classical modal logic (3 vols.). Technical Report Filosofiska Studier, nr 13, Uppsala Universitet, Uppsala, 1971.

[Shv89] Grigori F. Shvarts. Gentzen style systems for K45 and K45D. In A. R. Meyer and M. A. Taitslin, editors, *Logic at Botik '89, Symposium on Logical Foundations of Computer Science, LNCS 363*, pages 245–256. Springer-Verlag, 1989.

[Shv90] Grigori F. Shvarts. Autoepistemic modal logics. In Rohit Parikh, editor, *Theoretical Aspects About Reasoning About Knowledge*, pages 97–109, 1990.

[Sla77] Ralph L. Slaght. Modal tree constructions. *Notre Dame Journal of Formal Logic*, 18(4):517–526, 1977.

[Smu68a] R. Smullyan. *First order Logic.* Springer-Verlag, 1968.

[Smu68b] Raymond Smullyan. Uniform Gentzen systems. *Journal of Symbolic Logic*, 33:549–559, 1968.

[Sob64] Boleslaw Sobociński. Family K of the non-Lewis modal systems. *Notre Dame Journal of Formal Logic*, 5:313–318, 1964.

[Sub89] V. S. Subrahmanian. Algebraic properties of the space of multivalued and paraconsistent logic programs. In *9th Conf. Foundations of Software Technology and Theoretical Computer Science, LNCS 405*, pages 56–67. Springer-Verlag, 1989.

[Sun77] G. Sundholm. A completeness proof for an infinitary tense-logic. *Theoria*, 43:47–51, 1977.

[SV80] G. Sambin and S. Valentini. A modal sequent calculus for a fragment of arithmetic. *Studia Logica*, 34:245–256, 1980.

[SV82] G. Sambin and S. Valentini. The modal logic of provability: the sequential approach. *Journal of Philosophical Logic*, 11:311–342, 1982.

[Sza69] M. E. Szabo, editor. *The collected papers of Gerhard Gentzen*, pages 68–131. North-Holland, Amsterdam, 1969.

[Sza86] A. Szalas. Concerning the semantic consequence relation in first-order temporal logic. *Theoretical Computer Science*, 47:329–334, 1986.

[Sza87] A. Szalas. A complete axiomatic characterization of first-order temporal logic of linear time. *Theoretical Computer Science*, 54:199–214, 1987.

[TMM88] P. B. Thistlewaite, M. A. McRobbie, and B. K. Meyer. *Automated Theorem Proving in Non Classical Logics.* Pitman, 1988.

[Urq81] A. Urquhart. Decidability and the finite model property. *Journal of Philosophical Logic*, 10:367–370, 1981.

[Val83] S. Valentini. The modal logic of provability: Cut-elimination. *Journal of Philosophical Logic*, 12:471–476, 1983.

[Val86] S. Valentini. A syntactic proof of cut elimination for GL_{lin}. *ZML*, 32:137–144, 1986.

[van80] J. F. A. K. van Benthem. Some kinds of modal incompleteness. *Studia Logica*, 34:125–141, 1980.

[van86] J. F. A. K. van Benthem. Review of 'A Companion To Modal Logic'. *Journal of Symbolic Logic*, 51:824–826, 1986.

[vB78] J. F. A. K. van Benthem and W. Blok. Transitivity follows from Dummett's axiom. *Theoria*, 44:117–118, 1978.

[vB83] J. F. A. K. van Benthem. *The Logic of Time: a model-theoretic investigation into the varieties of temporal ontology and temporal discourse.* Synthese library; vol. 156, Dordrecht: Reidel, 1983.

[Ven85] G. Venkatesh. A decision method for temporal logic based on resolution. In *5th Conf. Foundations of Software Technology and Theoretical Computer Science, LNCS 206*, pages 272–289. Springer-Verlag, 1985.

[VS83] S. Valentini and U. Solitro. The modal logic of consistency assertions of Peano arithmetic. *ZML*, 29:25–32, 1983.

[Wal87] L. A. Wallen. *Automated Proof Search in Non-Classical Logics: Efficient Proof Methods for Modal and Intuitionistic Logics.* PhD thesis, University of Edinburgh, 1987.

[Wal89] L. A. Wallen. *Automated Deduction in Nonclassical Logics: Efficient Matrix Proof Methods for Modal and Intuitionistic Logics.* MIT Press, 1989.

[Wol83] P. Wolper. Temporal logic can be more expressive. *Information and Control*, 56:72–99, 1983.

[Wol87] Pierre Wolper. On the relation of programs and computations to models of temporal logic. In *Proc. Workshop on Temporal Logic in Specification, LNCS 398*, pages 75–122, 1987.

[Zem73] J. J. Zeman. *Modal Logic: The Lewis-Modal Systems.* Oxford University Press, 1973.

Index