General Modular Tableau Methods for Modal Logic

Souymodip

Abstract

This report presents a detailed description of the implementation of a general modular Tableau method for Modal logic. Logic Systems such as **K**, **KT**, **KB**, **K4**, **K5**, **K45** are studied in details. Though other system such as **S4** and **S5** can easily be interpolated from the studied systems. They are thus chosen to reveal the various problems that occurs in implementing tableau methods. The report shows the evolution of the ideas. First the simplest of the logical system i,e K logic is studied, then we move on to KT. We then take on KB logical system which needs an entirely radical deviation. We extend the ideas in K4 and K5. (This is not a final report, some proofs may be incomplete.)

Contents

Chapter 1

Modal Logic

1.1 Introduction

The first question that comes to mind is *What is modal logic?*. In the most succinct terms modal logic is the logic of various modalities to quantify truth. For example, consider the proposition P as *Roses are red*. In some worlds it can be true in other worlds false. We can qualify our judgment by rephrasing the proposition to a modal expression by saying *It is possible that Roses are red.*, or *It is necessary that Roses are red.* Thus a modal logic is a very general logic system that is able to capture an array of logics which includes logic for beliefs, for knowledge, for tense, for temporal expressions, for deontic (moral) expression like *it it Obligatory that* and *it is permitted that* and many others. Though modal logic can be polymodal in nature, where one can study the interrelation between different modal constructs , here we will only concern ourself with monomodal logic.

1.2 Kripke Insight and Modal language

The central concept of modal logic is that of a *labeled transition structure.* These are relational structures used to support the standard semantics of modal languages. In their monomodal form they are known as *Kripke structures or frames.* A Kripke frame is a collection of worlds and accessibility relation.

$$\langle W, R \rangle$$

where W is the set of worlds and R is the accessibility relation on the set W. A Kripke model is a Kripke frame with an additional construct valuation.

$$\langle W, R, \nu \rangle$$

A valuation is a function from a set of literals to the power set of worlds. It shows the set of worlds where a literal is satisfiable. If we extend the above example, we can have world where *Roses are red* and we can also have places where *Roses are not red.*.

$$\nu : P \longrightarrow \mathbf{2}^W$$

Consider this, in the modality of believes (Doxatic) of an individual (say Mr. S), if Mr. S is an inhabitant of certain world w, then all the worlds which he can imagine are the all the accessible worlds from w. We say Mr. S *believes* "Roses are red' says that in all the accessible worlds "Roses are red".

The propositional modal language is an extension of the pure propositional language formed by adding a new 1-ary connectives. In monomodal logic this denoted by $\Box$.[1]

The language is defined as,

- The set of all proposition.
- The element P of a fixed countable set of variables

[1] In multimodal logic each [i] is a separate operator.

- The propositional connectives

$$\top, \bot, \neg, \rightarrow, \wedge, \vee$$

- The box connectives

$$\Box$$

The box connective has a dual $\neg\Box$. Since it would be used a lot, it has been given a special symbol,

$$\neg\Box\neg P \equiv \Diamond P$$

1.3 Formal Systems and Satisfaction

The notion of *Formal System* is central. A formal system is defined by acceptable axioms $\mathcal{S}$. These set of axioms define a class of models where formulas of the axiom set are *Valid*. Before we get into details of proof system lets first go through the notion of validity and satisfiability.

The basic satisfaction Relation Let $\mathcal{A}$ be a frame. $(\mathcal{A}, \nu)$ be a given valued structure (a model) the relation

$$w \Vdash \phi$$

For any world w of $(\mathcal{A}, \nu)$ is defined as follows

- (Const) For the constants

$$w \Vdash \top \text{ , } not[a \Vdash \bot]$$

- (Var) For each variable P

$$w \Vdash P \Leftrightarrow w \in \nu(P)$$

- ($\neg$) For each formula ϕ

$$w \Vdash \neg\phi \Leftrightarrow not[w \Vdash \phi]$$

- ($\wedge, \vee, \rightarrow$) For all formulas θ, ψ
  - $w \Vdash (\theta \wedge \psi) \Leftrightarrow w \Vdash \theta$ and $w \Vdash \psi$
  - $w \Vdash (\theta \vee \psi) \Leftrightarrow w \Vdash \theta$ or $w \Vdash \psi$
  - $w \Vdash (\theta \rightarrow \psi) \Leftrightarrow w \Vdash \psi$ whenever $w \Vdash \theta$
- ($\Box$) For box in monomodal logic

$$w \Vdash \Box\phi \Leftrightarrow (\forall w_i \prec w)[w_i \Vdash \phi]$$

A formula ϕ is **satisfiable** if there exist $w \in (\mathcal{A}, \nu)$ with $w \Vdash \phi$. In modal logic Satisfaction comes in three different flavors .

- Pointed valued satisfaction

$$\Vdash^p$$

  The general notion of satisfiability where a formula ϕ is satisfied at a particular world.

- Satisfaction in the valued structure

$$\Vdash^v$$

Defined as

$$(\mathcal{A}, \nu) \Vdash^v \phi \Leftrightarrow (\forall w)[(\mathcal{A}, \nu, w) \Vdash \phi]$$

- Unadorned Structures

$$\Vdash^u$$

Defined as

$$\mathcal{A} \Vdash^u \phi \Leftrightarrow (\forall \nu)[(\mathcal{A}, \nu) \Vdash \phi]$$

Armed with this knowledge we can define validity. A formula ϕ is **Valid in a model** $\langle W, R, \nu \rangle$ if it is satisfiable for all worlds of W. (in other words valued satisfiability). A formula ϕ is **Valid in a frame** $\langle W, R \rangle$ if it is valid for all models of the frame. A class of structures **M** is completely matched by a formal system $\mathcal{S}$ if for each formula ϕ we have

$$\vdash_S \phi \Leftrightarrow \mathbf{M} \Vdash \phi$$

Definition 1.3.1

Sub Model . A model $\boldsymbol{G} = \langle W', R', \nu' \rangle$ is called a Sub Model of a Model $\boldsymbol{M} = \langle W, R, \nu \rangle$ iff,

- $W' \subseteq W$
- $R' \subseteq R$
- $\forall P \in \mathcal{P}. \nu'(P) \subseteq \nu(P)$

1.4 Theorem Proving using Tableau

The objective is to prove that certain formulas can be derived deform some axiom sets. Thus,

$$\Phi \vdash \phi \Leftrightarrow \Phi \models \phi$$

We concentrate on use of modal tableau systems for performing deduction. We will get to know in details how it is done for different formal systems (like $K, T, B, S4$) later. This section presents a brief introduction to a general tableau method.

Deduction using tableau method is a refutation procedure, where we try to find the satisfiability of the negation of the formula (say ϕ) that we are trying to validate. We start with the negation of the formula and call it a numerator.[2] Each step we check whether we have contradiction (inconsistency) or not. A formula set is **Inconsistent** if there exist a complementary pair of formulas in it. We take any of the formula ψ from the formula set and apply a tableau rule to yield new tableau. We call it the **Principal Formula** and apply a suitable tableau rule(ρ).

$$(\rho) \quad \frac{\mathcal{N}; X}{\mathcal{D}_1 | \mathcal{D}_1 | \ldots | \mathcal{D}_m}$$

Each of the D_i is separate branch of the tableau. We say the formula is satisfiable or unsatisfiable from the fate of each of these branches. Here (ρ) is the tableau rule, X is the principal formula and $\mathcal{N}$ is the reset of the formula set. For a logic system **S**, we will denote the sequence of rules by $\mathcal{C}\mathbf{S}$.

1.5 Closure of a Tableau

We say a formula is valid if its negation is unsatisfiable. We say a formula is unsatisfiable if every branch of its tableau closes. Thus the notion of closed tableau is crucial to the tableau method. Here we look into the general definition of closure of a branch. A branch B of a tableau is closed if it contains either a complementary pair of formulas or false. A tableau is said to be closed if all its branches closes. Later we will revise this definition. We will see that this idea of closure leads to incompleteness of tableau proof technique.

[2]The formula can be a collection of formula and thus, the numerator is set of those formulas

Chapter 2

Tableau methods for K - logic System

2.1 Introduction

This article walks through various implementation decision for building a $\mathcal{C}$K tableau. Before we get into the details, lets familiarize ourselves with various symbols which will be used. I first present the sketch of the implementation. Then I will elaborate the problem that will creep up if naïvely implemented. Then I follow by the actual implementation. Lastly but not the least I will analyses its time complexity. Time complexity is the worst case which will be exponential in nature.

2.2 Definitions

1. $\mathcal{C}K^{\star} \equiv \mathcal{C}K$ - $\{(K), (\theta)\} \cup \{(\theta K)\}$
2. $\mathcal{C}P$ rules $\equiv \{(\wedge), (\vee), (\bot), (\neg)\}$
3. Box Modal Variant of $\phi \equiv \Box^i\phi$ with i $\geq$ 0
4. $\mathcal{N}$ is the numerator of the rules
5. $\mathcal{D}_i$ is the i^{th} denominator.
6. $\Box$Y = $\{ \Box\phi \mid \phi$ is a modal formula$\}$
7. ρ is a sequence of $\mathcal{C}P$ rules
8. (K) rule

$$\frac{\Box Y; \Diamond P}{Y; P}$$

2.3 The Steps

If we envisage the tableau method as a black box ,what we have as input are a Formal System by its axiom set $\mathcal{S}$ and a formula set F. The decision problem is whether the formula set is a theorem of $\mathcal{S}$. Ideally we want a deduction steps starting from the set $\mathcal{S}$ and culminating in each of the formulas of the set F. Or we can say, for every Models M belonging to the class of models $\mathcal{M}_s$ defined by $\mathcal{S}$, the formulas of F are valid. In order to prove that, we check whether the negation of the formula set is satisfiable in some model of $\mathcal{C}_s$. If we are successful then we have a counter example that the formula set is not valid, if we fail then theorem F is valid.

We start with negation of all the formula that we want to validate. Lets call it F. At each step of the tableau method we work on this set F and transform it using some predefined rules or methods. At each step we select a formula from our pool of formula set F, and call it the *principal formula*. An applicable rule transforms this formula and we have a modified set of formulas to work with.

The rules are defined below. Each of the numerator is the set F.[1]

1. *Remove all box modal variant of* $\top$.
   These formula do not contribute to the tableau methods.They are always true any way.

2. *The* (α) *rule.* Same as the α rule of the proportional tableau, it is defined as follows.

$$(\alpha) \quad \frac{X; P \wedge Q}{X; P; Q}$$

   If $P \wedge Q$ is satisfiable then P and Q are also satisfiable. It can be shown that the rule is not only sound but complete to. The proof is quite similar to work on tableau methods by R.Smullyan[1] for propositional logic. I will defer from going through it for now.

3. *The* (β) *rule* It works on disjuncts.

$$(\beta) \quad \frac{X; P \vee Q}{X; P | X; Q}$$

   It is interpreted as if $P \vee Q$ is satisfiable then either P is satisfiable or Q is satisfiable.

4. *The* $(\neg)$ *rule*

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

5. *The* $(\bot)$ *rule*

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

   We can never have a situation where both P and ¬ P is satisfiable simultaneously and hence we have a contradiction.

   We call all the above rules $\mathcal{C}$P rules. These are all propositional rules which are sound and complete[1].

6. *The* (θ) *rule*

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

   This is one of the strangest rule. This rule is sound ,i.e, if we know that X and Y are satisfiable then one can deduce that X alone is satisfiable, but this rule is not *invertible*[2]. It is used only to guide our search for contradiction in some direction. We will get to know more of its malicious effects later.

7. *The* (K) *rule*

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

   The soundness and completeness of this rule will be taken up in details later. But, it important to clarify few details first $\Box$X includes formulas of type $\Box Q$ and $\neg\Diamond Q$ and $\Diamond P$ can be also be $\neg\Box P$.

[1] ";" is used to partition the set
[2] we will come to it very soon

8. *The (θK) rule*

$$(\theta K)\frac{S;\Box Y;\Diamond P}{Y;P}$$

Where S is set of purely propositional rules, Y = $\{P|\Box P \in F\} \cup \{\neg P|\neg\Diamond P \in F\}$ This rule is derived from the above two rules. The soundness and completeness of this rule is of great importance as our system will be partially based on it. It first elements all the pure propositional formulas denoted by Y are eliminated by the (θ) rule, finally (K) rule is applied.

9. *Termination.*
At each step of the procedure we check whether the termination condition has been reached or not. Suppose we have reached a stage when we are left with X.

If,

$$\mathrm{X} = \mathrm{S} \cup \Box\mathrm{Y}$$

Where S is a set of atomic proposition. X is $\mathcal{CP}$ saturated, i.e, no $\mathcal{CP}$ rules are applicable.

If ($\perp$) rule is not applicable to the X , then a termination has been reached. The branch of the tableau under consideration is open.

Lemma 2.3.1 *If $X = S \cup \Box Y$ (has of the form 9 describe above) and $\perp$ rule is not applicable, then X is satisfiable.*

Proof We can create a K-model which satisfies X. Consider a graph with one node w, with accessibility relation being empty. For all atomic proposition P $\in$ X, the valuation ν is defined as

$$w \in \nu(P)$$

The Model M is

$$\langle\ w\ ,\ \emptyset\ ,\nu\rangle \models_K \mathrm{X}$$

If X contains a formula pair on which ($\perp$) rule is applicable ,i.e, say we have no such pair say p_i and p_j which are compliments of each other, then the tableau has reached a termination in which case the branch is open. The formula is satisfiable. The model M is the counter example. ■

Now that we are familiar with few of the rules lets get on to few details that are required to build a deterministic algorithm for tableau methods.

The order of application of rules

The tableau method is non-deterministic in general. The idea is to translate it to a deterministic method that is sound and complete and is space and time efficient.

In tableau methods we have a choice on the order on the application of rules. If we reach a contradiction (by the application of ($\perp$)) at every branch we say the tableau is closed and the formula set is unsatisfiable. The reason import of tableau methods to modal logic is not straightforward cause some sequence of rules will lead to closed tableau while other will not. Thus, closeness of the tableau is defined as if there exist a sequence in which the tableau is closed then the formula is unsatisfiable[3].Thus, the order of rules is crucial here.

We cannot apply (θK) to X without first making it $\mathcal{CP}$ saturated.
We will see why such unrestricted application of θK is erroneous. For example let,

$$\mathrm{X} = \{\ \Diamond\neg P \wedge \mathrm{Q}\ ;\ \Box\mathrm{P}\ ;\ \Diamond\mathrm{R}\ \}$$

[3]Or the negation of it is valid

Let Y = $\{\Diamond\neg P \wedge$ Q$\}$ contains no box modal variant, thus,

$$(\theta K) \quad \frac{X}{P; R},$$

The denominator is open. Thus X should be satisfiable. It can be easily be seen that X cannot be satisfiable by any K-model. Since if

$$w \Vdash \text{X}$$

then,

$$w \Vdash \Diamond\neg\text{P and w} \Vdash \Box\text{P}$$

Then there exists a w' such that $w' \prec w$ (w' precedes w)[4] and

$$w' \Vdash \neg\text{P and } w' \Vdash P$$

Which is clearly impossible.

Thus (θK) is only applied when we have exhausted all $\mathcal{C}$P rules.

Multiple Occurrence of $\Diamond$ Modal variant.

So long, we have not addressed the question of how (θK) rule will work when we have a very general X of the form,

$$\text{X} = \text{S} \cup \Box\text{Y} \cup \Diamond\text{Z}$$

where, S is pure propositional formula set, it contains all the literals. Y = $\{ P_i \mid \Box P_i \in$ X , 0 $\leq$i$\leq$n$\}$ $\cup$ $\{\neg P_j | \neg\Diamond P \in X0 \leq j \leq n\}$. and Z = $\{ Q_i \mid \Diamond Q_i \in$ X , 1 $\leq$i$\leq$ m$\}$ $\cup$ $\{\neg Q_j | \neg\Box Q_j \in X, 1 \leq j \leq m\}$. *(Here forth this will be general meaning of X,S,Y and Z unless stated otherwise)*
There are two ways of handling this,

1. The Denominator contains $\mathcal{D}$ = Y;Z
2. There are m $\mathcal{D}_i$s each of them contain $\mathcal{D}_i$ = $\{$ Y ; $Q_i\}$ for all i $\leq$ m

In the first case will lead to false closure, which will make us conclude X as unsatisfiable. For example consider

$$\text{X} = \{ \Diamond\text{P} ; \Diamond\neg\text{P} \}$$

If we follow the first method we have $\{$P;$\neg$P$\}$ as denominator. So, we have a closed tableau. But clearly we can imagine a K-model $\langle\{w, w_1, w_2\}, \{w_1 \prec w, w_2 \prec w\}, \nu$ in which

$$\text{w} \Vdash \{\Diamond\text{P},\Diamond\neg\text{P}\} \text{ and } w_1 \Vdash \text{P and } w_2 \Vdash \neg\text{P}$$

This problem can be avoided by resorting to the second method. The crux of this method is
The purpose is not to seek a closed tableau (non-the-less we welcome it), but we look for satisfiability of a given formula set. If we fail to find a satisfiability even by our best effort, we claim the formula is not satisfiable. Or the negation of the formula is valid.

It is evident that each denominator after the application of (θK) represent a different world, accessible from the world that satisfies the numerator. If we don't redefine the closure of a branch, then we are in trouble. Following example will clearly show the problem,

Definition 2.3.2

$$(\theta K) \quad \frac{\Diamond(Q \wedge \neg P); \Box P; \Diamond R)}{(Q \wedge \neg P); P | P; R)}$$

$$(\wedge) \quad \frac{(Q \wedge \neg P); P | P; R)}{Q; \neg P; P | P; R}$$

It can be seen that there can't exist any model for which satisfies the formula set $\{\Diamond(Q \wedge \neg P); \Box P; \Diamond R)\}$.

[4]sometimes $w' \prec w$ is used instead of $w \rightarrow w'$, its used to avoid confusion with propositional Implication operator

Closure

We need to ask ourselves the following question. *What does it mean if one of the branches is closed?* Suppose, $w \Vdash X$ and (θK) is applicable on X. We are no longer in the world we started with. If one branch closes then it means the existence of the world corresponding to the branch is impossibility. If we follow blindly the tableau schema of closed tableau, then we are actually ignoring that branch and moving to other branch i,e. to other worlds, which is erroneous. Since the closed branch dictates us that such world w' is not possible and the (θK) rule made it mandatory that such a world exist which is accessible $(w' \prec w)$ from the world on which the rule was initially applied (w). Thus, we have actually hit a contradiction. Which renders the initial formula unsatisfiable , as we are unable to build a model.

The above situation compels us to believe if any branch closes after the application of (θK) then the tableau closes. This statement is also false. Consider the following example,

Definition 2.3.3

$$(\theta K) \quad \frac{\Diamond(\neg P \vee Q); \Box P; \Diamond R}{(\neg P \vee Q); P | P; R}$$

$$(\vee) \quad \frac{(\neg P \vee Q); P | P; R}{\neg P; P | P; Q | P : R}$$

Here we see that one of the branch is closed (after the application of (θK)) but we can very easily find a model that satisfies the initial formula.

2.4 The Way Out

If we look a little more closely we can see the difference between the two type of inconsistent branch that has been described so far. In the first case, a construction of a necessary world was impossible, in the second case, a world could be constructed in many ways, one of the way was not possible. Thus, the formula that transformed by the tableau rules are closely related to the world they are satisfied in. The only way out of this predicament is by placing the worlds along with the formulas[2].

If all the possible ways of building any particular world gets closed then the initial formula set is unsatisfiable.

Let's see how the above mentioned problem disappear with the new Tableau methods,

Definition 2.4.1

$w \Vdash \Diamond(Q \wedge \neg P; \Box P; \Diamond R)$

$$(\theta K) \quad \frac{w \Vdash \Diamond(Q \wedge \neg P; \Box P; \Diamond R)}{w' \Vdash (Q \wedge \neg P); P | w'' \Vdash P; R)}$$

$$(\wedge) \quad \frac{w' \Vdash (Q \wedge \neg P); P | w'' \Vdash P; R)}{w' \Vdash Q; \neg P; P | w'' \Vdash P; R}$$

w' is inconsistent, and thus the formula set is unsatisfiable

The second example,

Definition 2.4.2

$w \Vdash \Diamond(\neg P \vee Q); \Box P; \Diamond R$

$$(\theta K) \quad \frac{w \Vdash \Diamond(\neg P \vee Q); \Box P; \Diamond R}{w' \Vdash (\neg P \vee Q); P | w'' \Vdash P; R}$$

$$(\vee) \quad \frac{w' \Vdash (\neg P \vee Q); P | w'' \Vdash P; R}{w' \Vdash \neg P; P | w' \Vdash P; Q | w'' \Vdash P; R}$$

One of the branch of w' is consistent. We can build the model from the other consistent branch.

- *The set of worlds $W = \{w, w', w''\}$*
- *The accessibility relation $R = \{(w, w'), (w, w'')\}$*
- $\nu(P) = \{w', w''\}$, $\nu(Q) = \{w'\}$ *and* $\nu(R) = \{w''\}$

2.5 The Implementation

We start by assuming the given formula set X is satisfiable in some world w. We apply $\mathcal{C}P$ rules to X till we x becomes $\mathcal{C}P$ *saturation*[5]. If the Formula X is $\mathcal{C}P$ inconsistent then we close the Tableau , since it is impossible to construct a model for X. We have reached a stage where

X has the form , $X = S \cup \Box Y \cup \Diamond Z$

Where S is purely propositional and Y = $\{P_i | \Box P_i \in X, i \leq n\}$ and Z = $\{Q_j | \Diamond Q_j \in X, j \leq m\}$.

If m = 0 i.e, there are no $\Diamond$ modal variant, the tableau is open and we have indeed found a model for X
Else if, m > 0 then the world w has descendant $\{w_j | 1 \leq j \leq m\}$ where we take each

$$w_j \Vdash Y \cup \{Q_j\}$$

We will call this step as $(\theta K)^\star$. Now, in addition to a collection of Formula Sets for each world, we have a collection of reachable worlds. We define a Tableau for each of the worlds. let the tableau T_j corresponding to world w_j and it tries to make the formula set $\mathcal{C}P$-saturated. If the tableau T_j of the world closes then the world is inconsistent and hence the initial Formula was inconsistent. On the other hand if open tableau then we have to check all other worlds thus far created. If all the world are consistent, then the initial formula is consistent.

The above examples show the necessity of revising the notion of closed and open of our new Tableau method. The sequence of tableau rule defines a tableau method. We will form now on denote a tableau method by the collection of all the sequence of rules. For example the tableau method for propositional rule will be denoted by $\mathcal{C}P$. Similarly the above mentioned tableau will be denoted by $\mathcal{C}K^\star$.

$\mathcal{C}K^\star$ is same as $\mathcal{C}K$ except that ,

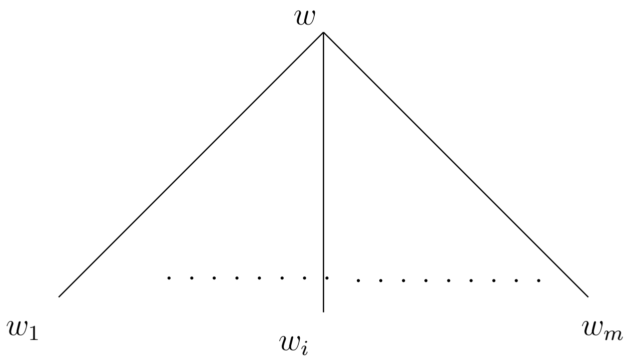
1. We carry the worlds along with the formula set.
2. When applying (θK) rule we create new set of reachable worlds as

$$(\theta K) \quad \frac{w \Vdash S; \Box Y; \neg \Box Z}{w_1 \Vdash Y; \neg Q_1 \S \ldots \S w_m \Vdash Y; \neg Q_m}$$

S,Y,Z has their usual definition

We introduce a new separator § along with |. While the later means alternative satisfiable set of formula for a world w. More importantly it is an *or* separator. That means any if any one of the formula set is satisfiable then it is satisfiable. § separates different worlds. It also means all the worlds need to be simultaneously satisfiable, hence its an *and* separator.

[5]A formula is called $\mathcal{C}P$ saturated if and only if no more $\mathcal{C}P$ rule is applicable.



2.6 Correctness

Theorem 2.6.1 *$\mathcal{C}K^{\star}$ is a Sound and complete for K System*

Proof It can be shown that $\mathcal{C}K$ is sound and complete[4]. The proof technique described here is deterministic. The $\mathcal{C}K$ is nondeterministic in nature because of the θ rule. Let's look at the rules of $\mathcal{C}K$ which contains $\mathcal{C}P$ which same as in $\mathcal{C}K^{\star}$. Now any application of (θ) followed immediately by (K) is equivalent to (θK). If $\mathcal{C}K\star$ is a sequence of rules which gives a closed tableau for a formula set Y, then we can conclude that the same sequence we can achieve a closed tableau for $\mathcal{C}K$. Thus,

$$\vdash_{\mathcal{C}K\star} \phi \quad \Longrightarrow \quad \vdash_{\mathcal{C}K} \phi$$

To show every $\mathcal{C}K$ sequence be equivalently translated to $\mathcal{C}K^{\star}$ is difficult prospect.

The other way is proving correctness of $\mathcal{C}K^{\star}$ separately. We can directly try to prove $\mathcal{C}K^{\star}$ is sound and complete in K-models. We will take up this now.

Sound

Soundness of a proof system S defined by

$$\vdash_{\mathcal{C}S} \Phi \quad \Longrightarrow \quad \models_K \Phi$$

Or, a formula set Φ whose negation has a closed tableau is a valid. Which can be restated as if the Denominator is closed then the numerator is closed too. Or, the contrapositive of the mentioned implication gives us, if the numerator is satisfiable then *at least one of* the denominators are also satisfiable.

If $\mathcal{N}$ is satisfiable in K then its denominators are to be satisfiable in K. $\mathcal{C}P$ is sound[1]. Only thing left is (θK). Let world w satisfies the formula set X.

Which has the form , $X = S \cup \Box Y \cup \neg\Box Z$. where S contains purely propositional formulas Y = $\{P_i | \Box P_i \in X\}$ and Z = $\{Q_i | \neg\Box Q_i \in X\}$.

$$(\theta K) \quad \frac{w \Vdash S; \Box Y; \neg\Box Z}{w_1 \Vdash Y; \neg Q_1 \S \ldots \S w_m \Vdash Y; \neg Q_m}$$

w satisfies X , then there are worlds w_i which satisfies Y and $\neg Q_i$, which are accessible from w. This comes directly from the definition of modal operators[3].

Completeness

$$\models_K \Phi \Longrightarrow \vdash_{K^\star} \Phi$$

That is if Φ is valid in K logic then the tableau method must prove Φ in $\mathcal{C}K^{\star}$, or $\neg\phi$ is closed.

$$not[\vdash_{K^\star} \Phi] \Longrightarrow not[\models_K \Phi]$$

Or, tableau is open for $\neg\Phi$ implies $\neg\Phi$ is satisfiable ,i.e, there exist a model which satisfies $\neg\Phi$. The tableau steps can be used to build such a model.

$\neg\Phi$ be defined by a set of formulas X, with numerator $\mathcal{N} \equiv w \Vdash X$

We start with formula set X and the numerator has the form

$$\mathcal{N} \equiv w \Vdash X$$

After the application of $\mathcal{C}P$ rule

$$(\mathcal{C}P)\mathcal{N} \equiv w \Vdash S \cup \Box Y \cup \Diamond Z$$

If $Z = \emptyset$ then the required model is

$$\langle \ \{\text{w}\}, \emptyset \ , \nu \ \rangle$$

where $w \in \nu(P)$ for all $P \in S$

Else,

$$(\theta K) \quad \frac{w \Vdash S; \Box Y; \neg\Box Z}{w_1 \Vdash Y; \neg Q_1 \S \ldots \S w_m \Vdash Y; \neg Q_m}$$

Since, the tableau is open none of the tableau $\mathcal{T}_i$ for w_i closes. Every (θK) application adds to R a new edge of the frame

$$(w, w_i)$$

We have now $\mathcal{T}_i$ for each $w_i \Vdash X_i$, or $w_i \Vdash S' \cup \Box Y' \cup \neg\Box Z'$.(making X_i saturated) $w_i \in \nu(P')$ where $P' \in S'$ .At each step we work with only a subset of $\mathcal{SF}(X)$[6] thus the number of worlds thus created is finite. The resultant graph along with the valuation is

$$\langle \ W \ , R \ , \nu \ \rangle$$

The claim is that the resultant model satisfies X. It can be verified by induction on the degree[7] of a formula. Without loss of generality we can assume all formulas are in negation normal form. The tableau starts with $w_0 \Vdash X$. Here we introduce the concept of **Modal Graph**. It is going to be prove a very important tool to prove completeness.

Model Graph

A Model Graph G(V,R) as V as set of worlds and R are directed edges defining accessibility relationship. Each world has a set of formula associated with. Each Model Graph is associated with a formula set X that is given to the tableau. These formulas are true for this world. G has the following property.

1. There exist a world $w_0 \Vdash \mathcal{C}P(X)$
2. If $w \Vdash P \wedge Q$ then $w \Vdash P$ and $w \Vdash Q$. Similarly if $w \Vdash P \vee Q$ then $w \Vdash P$ or $w \Vdash Q$.
3. If $w \Vdash P$ then $w \nVdash \neg P$.
4. If $w \Vdash \Diamond P$ then there exist a world w' such that wRw' and $w' \Vdash P$.
5. If $w \Vdash \Box P$ then there exit a world w' such that for all w' with wRw' we have $w' \Vdash P$.

Theorem 2.6.2 *If the graph defines a frame in **L** logic then X is satisfiable in **L**.*

Proof This can be very easily be shown by induction on the degree of the formula $f \in X$. If X is an atomic predicate then $w \Vdash X$. If X has the form of $P \wedge Q$, since $w \Vdash \mathcal{C}P(X)$ then P and Q are satisfied in w. similarly for other propositional operators. If X has the for of $\Box P$ then P is true in all the descendants of w, similar condition holds for $\Diamond P$. ■

[6]Set of sub formulas of X

[7]An atomic proposition has degree 0, P$\circ$Q where $\circ$ is propositional operator has degree = max{degree(P),degree(Q)}; degree($\neg P$) = 1+degree(P) ; degree($\Box P$) = degree(P) + 1 ; degree($\Diamond P$) = degree(P) + 1

We construct the model graph the following way. If P is purely propositional and P $\in$ X, then by construction $w_0 \in \nu(P)$ or $w_0 \Vdash P$. Let all formula of degree less than equal to k is satisfied by the model. We can show all formula of the type $f_1 \circ f_2$ are satisfiable, where $\circ$ are binary propositional operators, from the completeness proof of $\mathcal{C}P$.

Now, we prove $\Box P \in$ X is satisfiable. We start with $w_0 \Vdash X$. w_0 satisfies $\Box P$ if all the worlds accessible from w_0 satisfies P. The tableau method makes $w_i \Vdash P$ (θK rule.) P has a degree less than $\Box P$, and by induction hypothesis $P \in \mathcal{SF}(X)$ is satisfied by $w_i s$.

Similar argument holds for $\Diamond P$.

What the above theorem says that the Model graph satisfies X. The following corollary will lay the foundation for all future completeness proofs.

Corollary 2.6.3 *A model graph of a formula set Φ is a sub model of every **L**-Model which satisfies Φ*

Proof An Important observation is that there can be many more than one Model graph for a given satisfiable formula. This is justified by the fact we don't care about the logic system **L** in the definition of the Model graph. Let $\{G_i\}$ be the set of all possible model graph of Φ. We intent to show that there exist a Homomorphic function $f : G_i \longrightarrow M$, where M belongs to $\mathcal{M}_{\mathbf{L}}$.

Let us assume that $M \models \Phi$. Then there exist a world w of the model that satisfies Φ. We will construct a sub model of M and show that it is a model graph. This means we are trying to construct the image of G_i under f.

We start with a sub model S,

- S(W) := $\{w\}$
- S(R) := $\emptyset$
- $\forall P \in \mathcal{P}.S(\nu)(P) := \emptyset$

We know that $w \Vdash \Phi$. That means for any P of $\mathcal{P}$ and $P \in \Phi$, we make $w \in \nu(P)$. This will make $w \Vdash P$ (by definition).

If $P \wedge Q \in \Phi$ then both $w \Vdash P$ and $w \vdash Q$. If $\neg P \in \Phi$ then $w \nVdash P$ and $w \Vdash \neg P$. This takes care of all the logical operators.

Let D_P be the set of worlds $\{w_i | w_i \prec w$ and $w_i \Vdash P\}$. D_P is never $\emptyset$, as $w \Vdash \Diamond P$. Choose a $w' \in D_P$ and

$$S(W) := S(W) \cup \{w'\}.$$

Similarly, if $\Box P \in \Phi$ we know from the definition of $\Box$ that all the accessible worlds of w satisfy P. We build S by doing similar computation for all the worlds of S.

Let $R(U) \subseteq R$, with $U \subseteq W$, be the accessibility relation defined over the subset of W. The accessibility relationship of S is defined as

$$\mathrm{S(R) := R(S(W))}.$$

It can be seen that S conforms to every model graph property. Thus, S is a image of some homomorphic (structure preserving) f from G_i to S. ■

The above mentioned theorem and its corollary tell us we can create a **L**-Model from a model graph of Φ that is satisfiable in **L**. We will come across these particular set of **L**-Models (that are isomorphic to some model graph) quite often. We will call these models as **Strict L Models**.

2.7 Complexity Analysis

Lets analyses the process for the purpose of calculating running time. We ask ourselves the following Question

- *How many steps or iteration it will need?*
  At each step we apply rules on one of the member of the formula set. This causes it to break into sub formulas. Thus , the maximum number of iteration cannot exceed the number of subset of the sub formula of X . $|\mathcal{SF}(X)|$ is the size of the sub formula, then for an extreme case the size of the tableau is $2^{|sf(X)|}$.

- *How much time was spent in each step?*
  At each step we check whether we have reached a termination or not. We take every pair of formula from X and check whether ($\perp$) is applicable. We have a choice whether to execute negative normal form and then check for conjugate or check for conjugate without it.

  How do we check whether one formula X_i is complement of the other X_j?
  We can naïvely check at the top most level, that is, $X_i = \neg X_j$. This seems like it can be done in constant time, but equality checks for congruency in the structure of the two formulas. Thus, performing negative normalization does not incur excessive additional cost , it is actually proportional to checking equality. If we consider each formula as a tree , and while comparing them give charge to each node of the tree, then the total charge is proportionate to the size of the tree. Thus, amortized charge over all the nodes is thus proportionate to the number of node pair. Which is equal to $|sf(X)|^2$.

  This gives the total time complexity of ,

  $$O(|s\mathcal{F}(X)|^2 2^{|s\mathcal{F}(X)|})$$

  Which as one can see is exponential in the size of the sub formula.

Chapter 3

KT - Tableau Method

3.1 Introduction

The K formal system is one of the simplest modal system. In this chapter we apply the knowledge gained thus far in creating tableau method for T system. All KT - models have a reflexive accessibility relation. $\mathcal{A} = \langle W, R, \nu \rangle$ be a reflexive model (*correspondence theory*)[3]. By reflexive we mean,

$$\text{For every } w \in W,\ \{w, w\} \in R.$$

This, gives us two axioms which every world of these models must satisfy. Namely, the K axiom and the T axiom.

K: $\Box(P \to Q) \longrightarrow (\Box P \to \Box Q)$

T: $\Box P \longrightarrow P$

We, have already seen how to handle $(\theta K^{\star})$ rule. We must now see what dose (T) rule means,

$$\frac{X; \Box P}{X; P}$$

The Natural question that come to mind at this stage is how to incorporate the the new rule into the existing mechanism.

The closing of Tableau formulas are dependent on the order of application of rules.
It is sufficient to provide an example. Consider the following closed tableau.

$$(T) \quad \frac{w \Vdash \neg P; \Box P; \Diamond R}{w \Vdash P; \neg P; \Box P; \Diamond R}$$

Which is closed, hence the formula Set is unsatisfiable. This can be visually verified. When the same formula is churned in a Tableau where (θK) rule is applied first the following mishap happens.

$$(\theta K) \frac{w \Vdash \neg P; \Box P; \Diamond R}{w' \Vdash P; R}$$

Which is open.

The predicament is which one to apply first, the (T) rule or the (θK) rule? Before we answer this question, two new terms need to be defined.

- *Static* rules. A rule (ρ) is static if both numerator and denominators represent the same world.
- *Transitional rule.* A rule (ρ) is transitional if its not static i.e, the numerator and the denominators define different worlds.

Now we are in a position to answer the above question.

As (T) rule is a static rule it makes intuitive sense to first to check for unsatisfiability in the world under consideration before moving on to different accessible worlds.

If we assume the contrary and assume the existence of a world w which satisfies formula set X, then application of $\mathcal{C}P$ rules will lead to either an inconsistent collection of formulas, in which case we are done and the world w does not exists. Else if that is not the case let $(\mathcal{C}P)$X=X' is downward saturated ($\mathcal{C}P$ saturated, i,e no more $\mathcal{C}P$ rule is applicable) then X' has the all too familiar structure.

$$X' = \mathrm{S} \cup \Box\mathrm{Y} \cup \Diamond\mathrm{Z}$$

The purely proposition set S defines the decoration of the world. That is the set of all propositions that are true in that world. It might be the case that w cannot be constructed or any of its descendant cannot be constructed. Now if w is inconsistent that is there exist a formula P such that both P and ¬ P is true in w, then we need to enumerate all the propositions true in the world.But, we might not have yet discovered all such formulas satisfiable in w. Being a reflexive world $w \prec w$, w also satisfies Y. All the pure propositions of Y are true in w as well. This leads to fact that the application of (T) rule should precedence that of (θK) rule. Thus, if w cannot be created its imperative that (T) rule is applied before (θK) rule.

The Way

As the order does have an effect on the status of the final tableau , I choose to execute all (T) rules before moving on to the (θK) rules. What it semantically means that, when we are checking for satisfiability of a formula Set X at a particular world we enumerate all the sub formulas that are satisfiable in it before moving on to the other accessible worlds.

3.2 The Method

Suppose we are in the world w which satisfies X. X is $\mathcal{C}P$ saturated and has the form $S \cup \Box Y \cup \Diamond Z$. Since the accessibility relation is reflexive, if w satisfies formulas of the form $\Box P$ then w also satisfies formula P. Thus, every application of the (T) rule will discover new formulas satisfied by w. The following steps implements the idea

1. The starting numerator is defined as $w \Vdash X$. X is the negation of the formulas that we set out to validate.
2. Make X $\mathcal{C}P$ saturated.
3. For $w \Vdash X = S \cup \Box Y \cup \Diamond Z$, apply (T) rule. X now becomes X $\cup$ Y.
4. Check for termination. Terminating condition is same as in the $\mathcal{C}K$ tableau system.
5. If no new formula is added to X then we move on and apply (θK) rule.
6. Else we go to step 2, (we are making X (T) and $(\mathcal{C}P)$ saturated).
7. Go to step 2 to each of the descendants of w generated during θK.

The algorithm terminates if either we get a contradiction or we can no longer apply (θK).

3.3 Soundness and Completeness

Only think left is to show the correctness of the KT-Tableau method.

Theorem 3.3.1 *The KT tableau is sound and complete with respect to KT logic*

Sound means if whatever the proof system proves is consistent with the logic system. Completeness says all theorems of the logic system is provable by the proof system.

Soundness

We will use the same proving mechanism as for other rules. We will show if the Numerator is satisfiable then formulas given to each of the worlds in the denominator is also satisfiable.

$$(T) \quad \frac{w \Vdash S; \Box Y; \Diamond Z}{w \Vdash S; Y; \Box Y; \Diamond Z}$$

Thus, if there exist a world w which satisfies $\Box P$ then by the (T) axioms (which must be satisfied by every world of the models of KT-class) w also satisfies P. Hence the denominator Y is also satisfiable by w

Completeness

We prove completeness by showing that if the tableau method gives us the satisfiability of the formula then there exist a model in KT system which satisfies the formula. The central to the proof is the concept of *Model Graph*[4].

The theorem 2.6.2 makes life quite easy. The tableau method gives us a graph just like the K-tableau. The worlds and accessibility relation are defined as follows. With the addition that for every $w \in V$ we add an edge $\{w, w\}$ to R. Now the graph is reflexive and hence satisfies (T) axiom.[1]. These addition of new edges doesn't harm the property of Model Graph G. As the (T) rule ensures every $w \Vdash \Box P$ is $w \Vdash P$. Thus the graph G satisfies every model graph property. Further more, the accessibility relation defined by the graph is reflexive, that makes the corresponding model M member of KT logic system.

■

3.4 Termination and time Complexity

One can speed up the above process by taking every $\Box$ modal variant say $\Box P$ and adding P to it. After all the T axiom says $\Box P \rightarrow P$. What about termination? If it does terminate what is its worst case time complexity. It terns out the method does indeed terminate. A simple observation;*as no formula is introduced in the tableau that does not belong to the sub formula $\mathcal{SF}(X)$ and no formula set is repeated.* As the formula set of the sub formula is finite thus, the number of possible steps are also finite. The worst case time complexity takes the similar shape as in the case of K tableau methods.

$$O(|s\mathcal{F}(X)|^2 2^{|s\mathcal{F}(X)|})$$

[1]Correspondence theory, defined in the next chapter.

Chapter 4

KB - The Symmetric class of models

4.1 Introduction to the problem

The tableau construction is a refutation procedure.So if we want to check for the validity of a formula set X,we would try to build a model which satisfies formula set $\neg$X[1]. We must also make sure that model construction should also follow some specific rules. That is, we want the model to belong to some class of models. These classes are defined by a set of modal axioms. Our tableau rules generally mimic these rules (Implicit).For example the KT-tableau. But, sometimes direct application of these rules might lead to explosion in the size of the formulas. For example the B(P) : P $\rightarrow \Box\Diamond$P.

Definition 4.1.1

Correspondence Theory [3] Correspondence tells us about interrelationship between modal formulas and property of the accessibility relation of the class of models belonging to the formal system defined by the formula. We have thus far met two formal system K and KT. K system was defined by K axiom. It corresponding property on the relation is unrestricted. T axiom defines KT systems. Its corresponding property is that the accessibility relation $\mathcal{R}$ is reflexive.[2]

- $T(P) : \Box P \rightarrow P \equiv R$ *is reflexive*
- $B(P) : P \rightarrow \Box\Diamond P \equiv R$ *is Symmetric*
- $D(P) : \Box P \rightarrow \Diamond P \equiv R$ *is serial*
- $4(P) : \Box P \rightarrow \Box\Box P \equiv R$ *is transitive*
- ....

Implicit Tableau Systems These tableau system are purely syntactical in nature. The property of the accessibility relation are not directly built into the system. These properties are build into the rules of the tableau system. For example the (T) rule of the KT tableau represented the reflexivity of the accessibility relation.

Explicit Tableau System These systems are semantic in nature and hence accessibility relation play a pivotal role. The accessibility relation is represented directly and we reason around these property while proving validity of a formula.

4.2 Explicit Construction of models

The correspondence theory defines the property of the accessibility relation on the basis of the axiom set. In this section the Construction of KB logic system is undertaken. All models in this Class satisfy B axioms and have symmetric accessibility relation. The tableau method seen thus far had the attribute that once we move from a world w to its descendants we never revisit the world w again. In symmetric

[1]If A is set of formula then $\neg$A = $\{\neg P | P \in$ A$\}$
[2]Detail treatment on this subject can be found in Sally Popkorn :First step in modal logic.

relation we can no longer afford this luxury. Thus, our tableau method involves creating a graph where each node represent a world and accessibility relation is defined by the edges of the graph.

The idea is to maintain adjacency list of worlds. As the relation is symmetric, a world w' accessible from world w, makes w is also accessible from w'. There is no separate rule for B(P) axiom (and hence the explicit nature). As we transition from one world w to the next accessible world w' we add w' to the adjacency list of w and add w to the adjacency list of w'.

4.3 Algorithm

From now on we will call the adjacency list of w by $\mathcal{A}(w)$ and the set of formula satisfied at a particular world w as $\mathcal{S}(w)$. Initial numerator is $\mathcal{N}$=$w \Vdash$ X

1. Make X $\mathcal{CP}$-saturated. The numerator has the form $w \Vdash$S $\cup$ $\Box$Y $\cup$ $\Diamond$Z.
2. Let at some point in iteration we are at a world w which satisfies X. As always X is completely saturated and has the usual structure of $X = S \cup \Box Y \cup \Diamond Z$. The θK rule is modified in the following way. We maintain a graph where each node represent a world that has been discovered so far. Add Y to all the formula set of world which is adjacent to w. Each world contains a collection of formula which it satisfies.
3. Creates fresh worlds nw_i for each Q_i. Since the accessibility relation is symmetric, we make w also adjacent to nw_i. For each new nw_i, $\mathcal{S}(nw_i) = Y \cup Q_i$ and for the existing worlds $w_i \in \mathcal{A}(w)$, $\mathcal{S}(w_i) := \mathcal{S}(w_i) \cup$ Y . We take one of the worlds adjacent to $w_i \in \mathcal{A}(w)$.
4. Take one of the worlds w_i of Adj(w) whose $\mathcal{S}(w_i)$ has changed. Let F = $\mathcal{S}(w_i)$. F is $\mathcal{CP}$ saturated and has the form F = $S \cup \Box Y \cup \Diamond Z$. If the newly added formulas from w (i,e form the previous step Y) was the form $X' = S' \cup \Box Y' \cup \Diamond Z'$. Add fresh adjacent worlds for each member of $\Diamond Z'$ and add $Y' \cup Y$ to all the members of Adj(w_i). Follow step 3 till termination is achieved
5. Termination can be achieved in one of the following way.
   - One of the world become inconsistent. That the $\mathcal{S}(w)$ contains P and $\neg P$. Thus, the world is inconsistent and we stop immediately. The starting formula set was hence inconsistent and the negative of the formula set is valid.
   - For a world w the k rule has no longer has any effect on the $\mathcal{A}(w)$ and $\mathcal{S}(w_i)$ for each $w_i \in \mathcal{A}(w)$.

4.4 Little more details on termination

In this section we do a detail study of each of the steps that was brushed over in the previous section. The algorithm tries to build a Model Graph with symmetric accessibility relation (bidirectional edges.) which satisfies formula set X whose negation we want to validate. Lets start by familiarizing with with a new construct. For each world w there is $\mathcal{T}(w)$. It is set of formulas that w obtained from its parent in the previous iteration. It represent the set of formula that is tentatively satisfied by the world. The formulas in $\mathcal{S}(w)$ are really satisfied at w. We will consider $\mathcal{N}w$ as creation of a new world w.

We want to check for the satisfiability of the formula set X. The first step is that we consider a world w and assume X is satisfiable in it. Some tableau rule is applied in some order on the formula set X. Let us define the initial condition more formally.

$$\mathcal{T}(w) := X \text{ and } \mathcal{S}(w) := \emptyset \text{ and } \mathcal{A}(w) := \emptyset$$

For any world w the algorithm does the following transformation.

Definition 4.4.1

$\mathcal{CP}$: 1. *Check $\mathcal{S}(w) \cup \mathcal{T}(w)$ is consistent or not. (Check for the existence of complementary pairs)*

2. *$\mathcal{T}(w) := \mathcal{CP}(\mathcal{T}(w))$ Makes the formulas in $\mathcal{T}(w)$ $\mathcal{CP}$ saturated.*

θKB: This is carried out in the following steps

1. $\forall \Diamond P_i \in \mathcal{T}(w)[\ (\mathcal{N} w_i \prec w) \wedge (\mathcal{T}(w_i) := P_i) \wedge (\mathcal{T}(w) := \mathcal{T}(w)/\Diamond P_i) \wedge (\mathcal{A}(w_i) := \{w\})\ ]$[3]
2. *If* $\Box Y \subseteq \mathcal{T}(w) \cup \mathcal{S}(w)$ *then* $\forall w_i \in \mathcal{A}(w)\ [(\mathcal{T}(w_i) := \mathcal{T}(w_i) \cup (Y - \mathcal{S}(w_i)))]$[4]
3. $\mathcal{S}(w) := \mathcal{S}(w) \cup \mathcal{T}(w)$

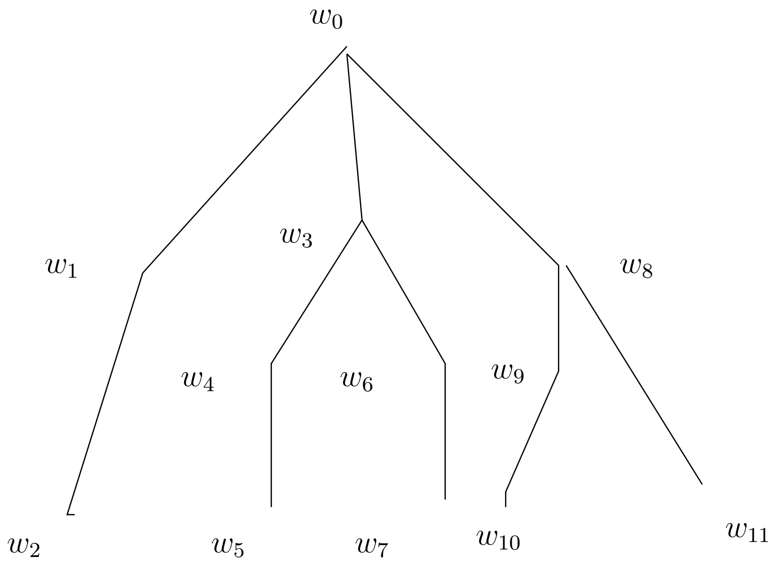
This finishes the transformation of the state w. The next decision is the choice of next world and termination. Let's take termination with inconsistency first. If during the application of the $\mathcal{CP}$ rule if the $\perp$ rule is applicable, we immediately close the process and declare the formula set X to be unsatisfiable. The correctness of this step has been studied in the context of K-Tableau and KT-tableau and the argument is quite similar. It comes from the soundness property of the tableau rules.

Lemma 4.4.2 *At any instance,if there is no formulas in the tentative set of all the worlds then all the formulas in $\mathcal{S}(w)$ for all worlds are satisfiable in the L-logic.*

Proof We can observe that $\mathcal{S}(w)$ contains only pure proposition and $\Box$modal variant formulas. Let's try to find out when does a proposition resides in $\mathcal{S}(w)$ of a world w. An atomic proposition P will come to $\mathcal{S}(w)$ from $\mathcal{T}(w)$ only after $\mathcal{T}(w)$has been $\mathcal{C}P$-saturated and no complimentary formula of P exist in $\mathcal{S}(w) \cup \mathcal{T}(w)$. . Thus , for some valuation ν, $w \in \nu(w)$. This makes P true in w.

Similarly, any formula of the type $\Box P$ is only in $\mathcal{S}(w)$ when the (θK) rule has been applied on the w. Thus all the descendants w_i of w have been given P, i.e, $\mathcal{T}(w_i) := \mathcal{T}(w_i) \cup P$. But we are at a state when there is no formula in the tentative set of any world, thus P is satisfiable in w_i. This makes $w \Vdash \Box P$.

Let us see how the model evolves. The proof method creates a tree like model where edges are bidirectional.



The leaf worlds w_l satisfies the formulas in $\mathcal{S}(w_l)$ then using induction we can prove the lemma.

■

The lemma 4.4.2 give us the condition for the termination of the algorithm. When the tentative set of every world is empty

$$\forall w \in W[\mathcal{T}(w) = \emptyset]$$

[3]For ever $\Diamond P$ in the tentative satisfiable set create a new world and add P to its tentative set and add a back edge. Remove $\Diamond P$ from the tentative set of the world considered

[4]If the formula is a atomic proposition add to the satisfiability set, else add all the $\Box$ modal variant of both the sets to the descendants of the world considered.

Now, the biggest question whether the termination conditions will ever be reached? Whether B-tableau is decidable. Let us define modal depth

Definition 4.4.3

Modal Depth : Modal depth md of a formula is recursively defined as follows.

Definition 4.4.4

$md(P) = 0$ *where* P *is literal*

$md(P \wedge Q) = min\{md(P), md(Q)\}$

$md(\neg P) = md(P)$

$md(P \vee Q) = min\{md(P), md(Q)\}$

$md(\Box P) = 1 + md(P)$

$md(\Diamond P) = 1 + md(P)$

Lemma 4.4.5 *The number of steps of the algorithm is bounded*

Proof We do an amortized analysis on the bound of the maximum number of possible steps. For that we chose a formula from the initial formula set and calculate the maximum number of world whose the tentative set it can reside in. Let the formula be f_k where k is the modal depth of the formula (finite). At a world $w \Vdash f_k$ with $k \geq 1$ is going to get divided into all the descendant $w_i \Vdash f_k$ (has the form $f_k = \Box f_{k-1}$. If f_k has any other form the distribution will be even less. Let l be the maximum spread[5] among all the node of the Model Graph.

At each step the degree reduces by one, the graph is finite [6] . Thus the maximum number of step associated with the formula is bounded by

$$O(kl)$$

which is finite. Thus, if number of steps a formula can break is finite then the total number of steps is also finite. ■

4.5 soundness and completeness

Now we prove correctness of the KB - tableau method.

Theorem 4.5.1 *The KB- Tableau method is sound and complete.*

Proof The soundness of the proof system comes from the soundness of each of the steps. $\mathcal{CP}$ rules is same as the others. Now the θKB rule is superimposition of θK rule on a symmetric frames. If P is purely propositional, then $w \Vdash P$ remains unchanged. If $w \Vdash \Box P$ then all the descendants w_i of w, have $w_i \Vdash P$ which is valid by the definition of $\Box$. Similarly if $w \Vdash \Diamond P$ then one of the descendants w' of w has $w' \Vdash P$. The steps are we haven't account for is the removal of $\Diamond P$ from the Satisfiable set. It is not difficult to see that the $\Diamond P$ plays no part in the Model Graph after the information P has been passed to one of the descendants.

Completeness is self evident, the algorithm gives a Model Graph which is symmetric in its accessibility relation. The lemma 4.4.2 gives a valuation scheme under which the initial formula set X is satisfiable ■

[5] out degree

[6] since each branch is created by some $\Diamond$ modal variant and there are finite number of them

4.6 *Alternative Universe**

One details which we have never assiduously looked up the case of $\vee$ propositional operator. We have always assumed intuitively the $\mathcal{C}P$ rule takes care of everything. Let's see would does $\vee$ do to the satisfiable set of a world w. If

$$w \Vdash P \vee Q$$

then either

$$w \Vdash P \text{ Or } w \Vdash Q$$

How does it fit into our above model of tableau system? We can see that each of the Ors gives a different view of the model that we want to create , that is either w is world in a model where P is true at w or, w is in some other model satisfying Q. This observation has lead to the idea of *alternate universes.* Every time the $\mathcal{C}P$rule splits the formulas set into various possibilities of models, we consider each possibility as alternate view of the model that we are trying to create.

Thus, at w we assume w satisfies P and try to build the model from there on. If we are successful then all is well. We have model which satisfies our initial formula X. If the model becomes inconsistent we try our luck with w satisfying Q. This spiting of views can occur at any step of our model construction At each split we say we have an alternate possibility of our model.

Chapter 5

K4 Logic

5.1 4 Axiom

The K4 formal system is characteristic by the formula 4(P).

$$4(P) \quad : \quad \Box P \quad \longrightarrow \quad \Box\Box P$$

The correspondence property[3] tells us if this formula is valid for every valuation of a frame $\langle W, R\rangle$ then the accessibility relation R is transitive. So, K4 formal system contains the usual K axiom along with 4. Let's see how the axiom 4 is realized as a tableau rule.

It can be seen from the structure of the axiom 4 that if a world ,

$$w \Vdash \Box P$$

then

$$w \Vdash \Box\Box P.$$

The degree of the formula increases by one. Thus, the axiom 4 can't be translated to tableau rule directly. We take advantage of the fact that R is transitive. We move to the next adjacent world to the current world w and apply the following rule.

$$(K4) \qquad \frac{X; \Box Y; \Diamond P}{Y; \Box Y; P}[4]$$

The rule is aptly named (K4) as it is an amalgamation of both 4 and K axiom . The proper way of expressing this rule along with worlds in which the numerator and denominator is satisfied is ,

$$(K4) \qquad \frac{w \Vdash S; \Box Y; \Diamond Z}{w_1 \Vdash Y; \Box Y; Q_1 \S \ldots \S w_m \Vdash Y; \Box Y; Q_m}$$

Where the worlds w_i with $1 \le i \le m$ are accessible from the world w.

5.2 The Algorithm

Before we start proving the soundness and completeness of the (K4) rule , there is a minor glitch that need to be disposed off. The degree of the formula doesn't increase but it stays the same. We can come up with a pathologic example in which the tableau implemented on the basis of (K4) rule will fall in an infinite loop. Consider the following set of formulas,

$$\{P; \Box\Diamond P; \Diamond P\}$$

Now, if we apply the (K4) rule to the above set ,

$$(K4) \qquad \frac{w \Vdash P; \Box\Diamond P; \Diamond P}{w' \Vdash \Diamond P; \Box\Diamond P; P}$$

This is same as the denominator. Thus, whatever algorithm we may devise it must tackle the termination condition more carefully.

We will take the cue from the KB Tableau method. Though the tableau is *Implicit* in nature but we keep *explicit* information of the Model Graph as the tableau method progresses. The above example showed the imperative of saving the information of *seen* worlds. The **(K4)** rule uses the information stored in previously seen worlds to decide whether to create a new world or add an edge to a world already seen.

The Steps are as follows

1. We start with some world w with, $\mathcal{T}(w) = \neg X$ and $\mathcal{S}(w) = \emptyset$

2. $\mathcal{T}(w) := \mathcal{C}P(\mathcal{T}(w))$

3. **if** *the set* $\mathcal{T}(w)$ *is inconsistent* **then**
   stop
   Declare the formula set to be unsatisfiable.
   end

4. **if** *the set* $\mathcal{T}(w)$ *is not inconsistent* **then**
   normalize the $\mathcal{T}(w)$ so that $w \Vdash S \cup \Box Y \cup \Diamond Z$ [1]
   Continue
   end

5. **foreach** *worlds* $w_i \in \mathcal{A}(w)$ **do**
   $\mathcal{T}(w_i) = Y \cup \Box Y \cup Q_i$.
   if $\mathcal{T}(w_i) \subseteq \mathcal{T}(w')$, *where* w' *has already been seen* **then**
   make w' accessible to w instead of creating of the new world w_i.
   end
   end

6. **if** *The exist worlds* $w_i \in \mathcal{A}(w)$ **then**
   foreach *worlds* $w_i \in \mathcal{A}(w)$ *that has not yet been explored* **do**
   apply step 2 to it.
   end
   end

The deviation from the usual tableau rule has great consequences. It has the potentiality of creating a loop. A loop Model Graph G in a transitive relation leads to the formation of maximally connected modal sub directed graph or a *directed Clique*. Lets denote such sub graph by H. This compels that,if any world w[2] in the sub graph satisfies $\Box P$ then all other worlds in H must satisfy P. It is now evident that this algorithm avoids the pitfall of infinite recursion of the above example.

The following Property must be satisfied keep Model Graph property untampered.

$$\forall w \in H \ , \forall \Box X \ [w \Vdash \Box X \longrightarrow \forall w' \in H[\ w \Vdash X \ ]]$$

5.3 Soundness and Completeness

In this section we establish the credibility of the algorithm. The pivotal point of the algorithm is the **(K4)** rule. We have already seen the correctness of the $\mathcal{C}P$ rule, we will devote this section to establishing correctness of the **(K4)** rule.

[2]This is an abuse of notation as we are using the nodes of Model Graph and the corresponding world synonymously

Soundness

Consider any model $\mathrm{M} \in \mathcal{M}_{K4}$[3]. As mentioned earlier accessibility relation R in M is transitive, by the correspondence property. To prove soundness we show that if the numerator $\mathcal{N}$ is satisfiable then the denominator $\mathcal{D}$ is also satisfiable.

Lemma 5.3.1 *The (K4) tableau rule is sound with respect to **K4** logic*

Proof Let w be a world of $\mathrm{M} = \langle W, R, \nu \rangle$ with

$$\mathcal{N} : w \Vdash S \cup \Box Y \cup \Diamond Z$$

where the symbols have their usual meaning. There exist a world $w_i \in W$ with wRw_i and $w_i \Vdash Y \cup Q_i$. Since R is transitive, all successors of w_i are accessible from w as well. Thus, they would also satisfy Y. Therefore, w_i also satisfy $\Box Y$. OR,

$$\mathcal{D} : w_i \Vdash Y \cup \Box Y \cup Q_i$$

■

Completeness

The notion of Model Graph is central to proof of completeness. If we ignore the loops and backward edges, the **(K4)** rule behaves similar to that of (K) rule. It makes sure that there exist a $w_i \Vdash Q_i$ for each $w \Vdash \Diamond Q_i$ and if $w \Vdash \Box P$ then for every $w' \prec w$, $w' \Vdash P$. Thus, all the properties are satisfied. The problem comes when **(K4)** execute that step 5 of the algorithm. Two cases can ensue from this point. Case I the formation of loop, which in a transitive graph crests a cliques. Case II loop is not formed. The following theorem shows that the creation of loops or not the rule conforms to the Model Graph properties.

Theorem 5.3.2 *If a loop is created then every World of the directed clique still satisfy the Model Graph property.*

Proof Let's assume we start at some world w_0 which satisfies $\{S; \Box Y; \Diamond Z\}$. [4] Consider a sequence of steps that results in the following sequence of worlds which results in a loop.

$$w_0 \to w_1 \to w_2 \ldots \to w_x.$$

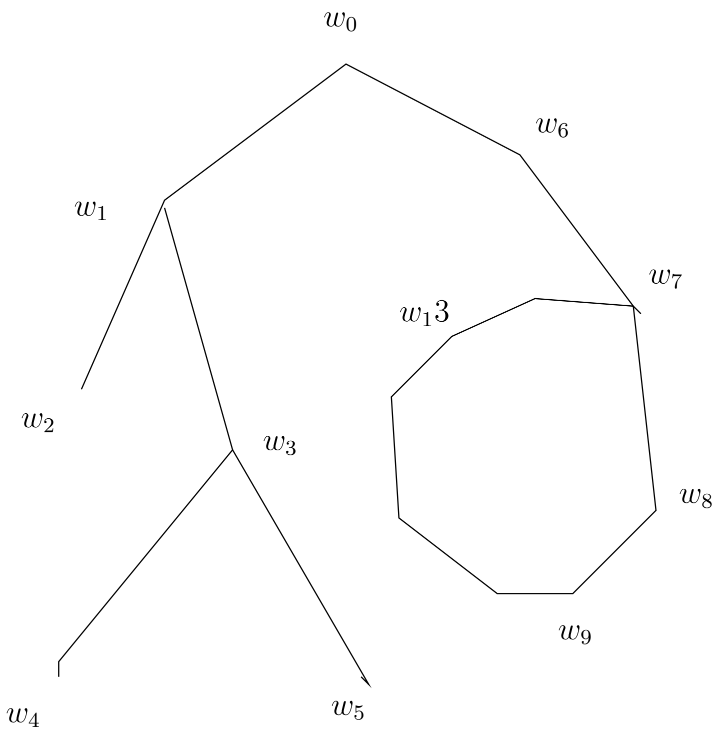
We take $\mathcal{S}(w_x) \subseteq \mathcal{S}(w_0)$. It is self evident that the satisfiability set of each world only decrease or remain the same in size. The algorithm says that world w_x is same as w_0. The transitive closure of a loop leads to a clique. Thus every w_i is connected to any other w_j with j $\neq$ i and with $0 \leq i \leq x-1$, $0 \leq j \leq x-1$. Lets see if the model graph properties are preserved or not.

If $w_i \Vdash \Box P$ then every j > i $w_j \Vdash P$((**K4**) rule ensures that much). Now for worlds from w_0 to w_{i-1} if we could prove that w_0 also satisfies $\Box P$ and P then we are done. This is easy as $\mathcal{S}(w_x) \subseteq \mathcal{S}(w_0)$. $w_{x-1} \Vdash \Box P$ and P then w_x and hence will also satisfy $\Box P$ and P.

■

[3] $\mathcal{M}_{K4}$ are set of finite models where axiom 4 is valid

[4] Symbols have their usual meaning



When $\mathcal{S}(w_x)$ is a subset of some $\mathcal{S}(w')$ which is not the ancestor of w_x, we don't get a loop. This make our job easier since we don't have to worry about cliques. Modal graph property remains preserved, since for all the worlds w that can access w_x now can access w'. The following theorem establishes this fact

Theorem 5.3.3 *The Model Graph property is not violated even when loops are not created.*

Proof Let $\mathcal{S}(w_x) \subseteq \mathcal{S}(w')$. Let w' is not the ancestor of w_x. We will denote $\mathcal{A}(w)$ as the set of ancestor of w and $\mathcal{L}(w)$ as set of worlds accessible from w.

According to the algorithm if we make w_x as w' then the following changes occurs.

- Every world $w \in \mathcal{A}(w_x)$ can access w'. For every $\Box P$ satisfiable w, the **(K4)** rule make sure that w_x satisfies P. As $\mathcal{S}(w_x) \subseteq \mathcal{S}(w')$, w' also satisfies P.
- Every world $w \in \mathcal{A}(w_x)$ can access every world $u \in \mathcal{L}(w')$. Arguing the same way as before, if $w \Vdash \Box P$ then $w_x \Vdash P$ and $\Box P$ (due to **(K4)**), thus $w' \Vdash \Box P$. If w' satisfies $\Box P$ then every u satisfies P.

■

Next we must consider the scenario when more than one world contains the formulas satisfied at w_x. There can be two possible cases. Firstly, if the worlds belong to the loop then, then make a back edge to the world which was discovered the earliest. The transitive closure of the loop take cares of the rest of the worlds. The second condition the worlds do not to the same chain. Ideally we would like to connect w_x to all the worlds, but it is not necessary. If we make w_x same as any one of them, we still preserve the Model Graph property.

Chapter 6

K5 Logic

6.1 The Axiom

All models M of **K5** logic are euclidean. The accessibility relation of M $\langle W, R, \nu \rangle$ satisfy

$$\textbf{Euclidean :} \ \forall abc \in W[b \prec a \wedge c \prec a \longrightarrow b \prec c]$$

The corresponding modal formula is

$$\textbf{5 :} \ \Diamond P \longrightarrow \Box \Diamond P$$

The Euclidean property dictates that, if a world 'a' can access worlds b and c then b can also access c. It's necessary to study the structure of an euclidean model in order to come up with correct tableau rules. Every euclidean model M can be divided into two parts. It is not necessary that both of them will always be present. The first part is the genesis. Here a single world (say a) acts as the seed for the model. The seed is not a mandate for an Euclidean Models.(and there can be more than one root) The second part or complete mesh (Or directed clique), comes into play when a has one or more than one descendants $\{w_i\}$. Every world (along with $\{w_i\}$) created from here on belong to the second part. These worlds are completely connected with one another. If W' be the set of these worlds, then the accessibility relation over W' is defines as $R' = W' \times W'$. Evidently the accessibility relation R' is *reflexive, transitive and Symmetric.*

6.2 Algorithm

The first part of every euclidean model can be generated by variation of $K5$ rule.

$$(K5) \quad \frac{w \Vdash S; \Box Y; \Diamond Z}{w_1 \Vdash \{Y; Q_i; \Diamond Z\} \S \ldots \S w_m \Vdash \{Y; Q_m; \Diamond Z\}}$$

The next part is constructed by repeated application of $K45$ rule.

$$(K45) \quad \frac{w \Vdash S; \Box Y; \Diamond Z}{w_1 \Vdash \{Y; \Box Y; Q_1; \Diamond Z\} \S \ldots \S w_r \Vdash \{Y; \Box Y; Q_r; \Diamond Z\}}.$$

As usual we try to construct a model which satisfy the negation of the formula to be proven and start with a world w whose temporary set contains the negation of the formula (say X). We will call it the root. Lets first go through the algorithm,

1. $\mathcal{T}(w) := X$

2. $\mathcal{T}(w) := \mathcal{C}P(\mathcal{T}(w))$ (* make it $\mathcal{C}(P)saturate$*)

3. **if** $\mathcal{T}(w)$ *is inconsistent* **then**
   Stop. Declare X to be unsatisfiable

   end

4. Apply $(K5)$.
   It creates a number of descendants of w, $W_1 = \{w_1, \ldots, w_m\}$.

5. **foreach** $w_i \in W_1$ **do**
   if $\mathcal{T}(w_i)$ *is inconsistent* **then**
   Stop.Declare X to be unsatisfiable

   end

   end

6. Chose any world $w' \in W_1$
   (*here we will introduce a trick. Since the accessibility relation over the worlds yet to be discovered is reflexive, transitive and symmetric, we will not explicitly explore each member of W_1. Any one of the descendants would suffice. We have chosen w'.*)

7. $\mathcal{T}(w') := \mathcal{C}P(\mathcal{T}(w'))$

8. **if** $\mathcal{T}(w')$ *is inconsistent* **then**
   Stop. Declare X to be unsatisfiable

   end

9. $\mathcal{S}(w') := \mathcal{T}(w')$
   $\mathcal{T}(w') := \emptyset$

10. Add w' to W_s.

11. Explore world w'. Apply (K45).

$$(K45) \quad \frac{w' \Vdash S'; \Box Y'; \Diamond Z'}{w_1' \Vdash \{Y'; \Box Y'; Q_1'; \Diamond Z'\} \S \ldots \S w_r' \Vdash \{Y'; \Box Y'; Q_r'; \Diamond Z'\}}.$$

12. **foreach** w_i' **do**
    if $\exists u \in W_s$ *[* $\mathcal{T}(w_1') \subseteq \mathcal{S}(w)$*]* **then**
    make u accessible from w'

    end

    else
    $\mathcal{T}(w_i') := \mathcal{C}P(\mathcal{T}(w_i'))$
    foreach $\Box P \in \mathcal{T}(w_1')$ **do**
    if $P \notin \mathcal{S}(w_1')$ **then**
    $\mathcal{S}(w') := \mathcal{S}(w') \cup \{P\}$
    if $\mathcal{S}(w')$ *is inconsistent* **then**
    Stop
    Declare X unsatisfiable
    end

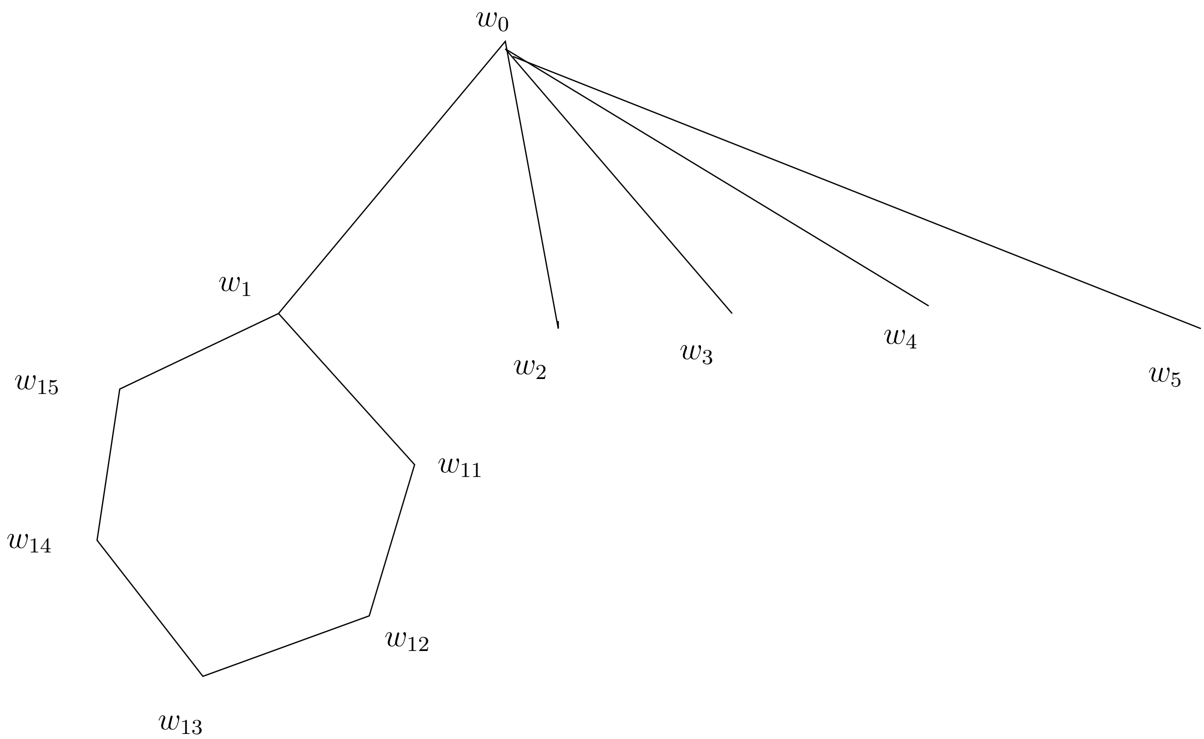
    end

    end

    Repeat step 5 for w_i'

    end

    (* That is apply 5 to branch headed by w_1' replace w' by w_1'*)
    end



The above figure gives a snap shot of what sort of model graph will be created.

6.3 Correctness

The above stated algorithm is quite different compare to other tableau methods that we have seen thus far. In this section we will try to justified the algorithm and argue on the soundness and completeness of this proof method.

Soundness

The soundness of the proof method implies that proof steps are logically valid. Soundness can also be stated as, whenever the numerator is satisfiable then the denominators are also satisfiable. Since, the notion of numerator and denominator gets vague, we will restate soundness as, if the formula was satisfied before the application of the step then it must be satisfiable after the execution of the step. We will take each step one by one.

The first two steps are always sound. We have already seen the soundness proof for the $\mathcal{C}P$ rules. The following lemma shows the nest step is also valid

Lemma 6.3.1 *The tableau rule* $(K5)$ *is sound in euclidean logic.*

Proof Suppose there exists a world w which satisfies $S \cup \Box Y \cup \Diamond Z$. There exists a world w', which is accessible from w, that satisfies $Y \cup \{Q_i\}$. There are other worlds accessible from w which satisfy Q_j with $0 \leq \text{j} \leq |Z|$. Being euclidean w' can also access these worlds. Thus, $w' \Vdash \Diamond Q_j$ for all j. Or, $w' \Vdash \Diamond Z$. ■

Definition 6.3.2

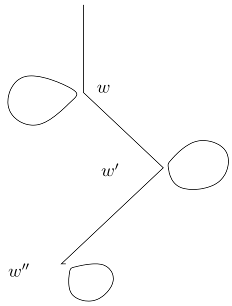
We will denote $R(U)$ *as a sub set of* R *and* U *is sub set of* W. $(x, y) \in R(U)$ *if* $x \in U$ *and* $y \in U$.

Let $M = \langle W, R, \nu \rangle$ be an euclidean model. Let a be the root of the model. W' is defined as $W - \{a\}$. Let us define the Euclidean property satisfied by three worlds as

$$\mathcal{E}(w_1, w_2, w_3) \equiv (w_2 \prec w_1 \wedge w_3 \prec w_1) \rightarrow (w_2 \prec w_3)$$

Lemma 6.3.3 *The* $R(W')$ *is transitive.*

Proof Take any element of $R(W')$ $w' \prec w$. By $\mathcal{E}(w, w', w')$ we get $w' \prec w'$. This property holds even for the worlds accessible from 'a'. We also have $\mathcal{E}(w, w', w)$, i.e, $w \prec w' \in R(W')$. For any $w'' \prec w'$ we can say $\mathcal{E}(w', w, w'')$. This will give us $w'' \prec w$.



■

The above lemma shows that the application of **(K45)** rule is valid. If w' satisfy $S \cup \Box Y \cup \Diamond Z$, then there exits a world w_r which satisfy $\Box Y$ as the relation is transitive and it also satisfy $\Diamond Z$ as the relation is Euclidean. Since, the size of the formulas don't reduce always, we do the checking whether world has already been discovered.

The rationale behind step 12 will not be clear till we take on completeness. The step is none the less valid. The proof of transitivity showed that every forward edge is accompanied by a back edge. If we assume the formula assigned to each worlds discovered thus far are satisfiable, then with the creation of the new world w_r (froward edge), which is an element of $W - \{a\}$, can access every other world of the set (Due to the back edges). Or $w' \prec w_r$. Thus, if $w_r \Vdash \Box P$ then $w' \Vdash P$.

Completeness

To show the completeness of the proof method we take the usual road of building a counter example of the formula to be proved. If the proof method can't find an inconsistency, we should be able to build a model of the logic system L which satisfies the negation of the formula. The notion of Model Graph is pivotal to the model creation. Recall that model graph for some finite fixed formula X forms a $L-$frame has the following property.

1. X $\subseteq \mathcal{S}(w_0)$ for some world $w_0 \in W$.
2. if $w \Vdash \Diamond P$ then there exist a world $w' \in W$ such that $w' \prec w$ and $w' \Vdash P$.
3. if $w' \prec w$ and $\Box P \in w$ then $w' \Vdash P$.

Also recall the notion of **Strict L Models**. Here we consider a *strict Euclidean Model* M $= \langle W, R, \nu,$ with $a \Vdash X$.

Lemma 6.3.4 *Let $W' = W - \{a\}$. The reflexive, transitive and symmetric ($W' \times W'$) closure of the accessibility relationship that we build up using the proof method will give a Euclidean accessibility relationship over W' (. Where a is the root of the model).*

Proof Let W_1 be the set of worlds accessible from the root $'a'$. It can be shown that $R(W_1)$ is $W_1 \times W_1$. Consider a world $w' \in W_1$. We know that $w' \prec a$. Thus, by $\mathcal{E}(a, w', w')$ we get $w' \prec w'$. Take any other world $w'' \in W_1$. By $\mathcal{E}(a, w', w'')$ we get $w'' \prec w'$ and by $\mathcal{E}(a, w'', w')$ we get $w' \prec w''$.

Now, consider any world $w_i \prec w'$. By $\mathcal{E}(w', w_i, w_i)$ we get $w_i \prec w_i$. We can show that w_i is connected to every world of $w \in W_1$. For all $w \in W_1$, we have $w \prec w'$. By $\mathcal{E}(w', w_i, w)$ and $\mathcal{E}(w', w, w_i)$ we get $w \prec w_i$ and $w_i \prec w$ respectively. The relation $w_i \prec w'$ is also symmetric. Take any other world $w \in W_1$, with $w \neq w'$, we have $w' \prec w$. By $\mathcal{E}(w, w_i, w')$ we get $w' \prec w_i$. Its transitivity is trivial, as worlds are totally connected.

Lets constructively prove the rest. Add w_i to W_1 to make W_2. Similar argument for any other $w \in W_s$, and add w to W_2. We can continue till we can no longer add new world $w \in W_s$ such there exists a world $u \in W_x$ with $w \prec u$. (* what about the worlds $W_s - W_i$? *) ■

The proof method can be used to build the appropriate Model Graph. The formula belonging to the $\mathcal{S}(w)$ are satisfied at w. We assume that the proof method has terminated without contradiction. This means that all branch of the method has loops. We will first build the graph and then show that it satisfies the Model Graph properties.

The application of the $(K5)$ rule has led to creation the first level of descendants of the root w_0. Let us denote this set of worlds by $W_1 = \{w_1, w_2, \ldots, w_m\}$ where m is the cardinality of the set Z^1. We choose one of the world $w_s \in W_1$ and apply $(K45)$. consider the chain of worlds starting from w_s. $S = w_s \succ w_{s1} \succ \ldots \succ w_{sn} \succ w_{sx}$. The chain cannot run infinitely as no new formula are created and the set of the subformulas is finite. Suppose w_x is same as $w_y \in S$. Let W_s be defined as $\{w_s, w_y, w_{y+1}, w_{y+2}, \ldots, w_{sn}\}$. Let R_s be the reflexive, transitive and symmetric closure of $\succ$ over W_s.

Lemma 6.3.5 *Modal graph property 2 and 3 is preserved in W_s.*

Proof Each world carries all the $\Diamond$ variant formulas((K45) rule). Therefore, for all $\Diamond P$ satisfied by $w \in W_s$ makes $w_{sn} \Vdash \Diamond P$. AS the sequence is a finite, this means every descendant of $w_s n$ has already been discovered. Thus, there exists a world w_{sj} which satisfies P for every $\Diamond P$ satisfied by w_{sn}. This shows all P satisfiable worlds are part of the loop. This takes care of the property 2. Since R_s is transitive all $\Box$ variant formula satisfiable at a world $w \in W_s - \{w_s\}$ satisfy the property3. The argument is same as the one given while proving the completeness of $(K4)$. Only problem is that w_s is also an element of W_s. Thus for every $\Box$ variant formula $\Box P$ satisfied at any other world of W_s, w_s must satisfy P. The step 12 takes care of this, as each newly unearthed $\Box$ variant formula $\Box P$ satisfiable at any $w \in W_s - \{w_s\}$, we make sure that $w_s \Vdash P$. This way property 3 remains untampered. ■

Consider the worlds of W_1. Let $R_1 = \{(w_i, w_j) |$ For every $w_i,\, w_j \in W_1\}$. To preserve property3, do the following augmentation, for every $\Box P$ satisfiable by $w \in W_1$ make $w' \in W_1$ satisfy P.

Lemma 6.3.6 *"For every $\Box P$ satisfiable by $w \in W_1$ make $w' \in W_1$ satisfy P " doesn't lead to contradiction.*

Proof There can be two possible source of inconsistency. One being $\Box P \in Y$ and $w_i \Vdash \neg P$. If $\neg P$ belonged to the set Y, then $\Box P$ and $\neg P \in \mathcal{S}(w_s)$. This is not possible, because eventually the contradiction would have been caught by the repeated application of $(K45)$. Alternatively let $\neg P \in \mathcal{SF}_\wedge(Q_i)$.[2] $\Diamond Q_i \in \mathcal{S}(w_s)$. The application of $(K45)$ would have led to a world with both P and $\neg P$. Another possible source of $\Box P$ is Q_i itself. The same argument with w_s containing $\Diamond\Box P$ holds. ■

We are almost to the end of our model creation. Let

$$W_0 = W_1 \cup W_s.\ R_0 = \{(w, W') |\text{ for every } w, w' \in W_0\ \} \cup R_1.$$

Lemma 6.3.7 *The frame $\langle W_0, R_0\rangle$ satisfy Model Graph property 2 and 3.*

Proof For every $\Box P$ satisfied by $w \in W_0$, every other world w' is satisfies P. This property existed among all the worlds of W_s and the augmentation preserves the property even with the addition of W_1. The loop contains at least one world which satisfies Q_i for every $\Diamond Q_i$. Thus our property 2 remains unscathed. ■

The final $K5-$ Frame is given by $\langle W, R\rangle$

$$W = W_0 \cup \{w_0\} \text{ and } R = R_0 \cup \{(w_0, w) |\text{ For every } w \in W_1\ \}.$$

Theorem 6.3.8 *The model $\langle W, R, \nu\rangle$, with $w \in \nu(P)$ for every $P \in \mathcal{P}$ and $P \in \mathcal{S}(w)$, satisfies X.*

Proof The creation of the Model Graph has trivialized the proof. We already have $w_0 \Vdash X$. w_0 is connected to every world of W_1 ((K5) rule). The lemma 6.3.5, 6.3.6 and 6.3.7 make sure that property2 and property3 are satisfied. Modal graph theorem gives the required result. ■

[1] Symbols carry the usual meaning
[2] $\mathcal{SF}_\wedge(A \wedge B)$ contains $\mathcal{SF}_\wedge(A)$ and $\mathcal{SF}_\wedge(B)$

Chapter 7

K45 & S5 Logic

7.1 The Axioms of K45

This system is built upon the axioms **K,4** and **5**. The accessibility relation **R** is thus, transitive and Euclidean. We have already seen how a transitive logic behaves (K4 logic). We have also developed tableau methods for K5 (with accessibility relation having only Euclidean property). The added property namely Euclidean, has some interesting effect on the models $\langle W, R, \nu \rangle$ of this system .

$$\textbf{Euclidean :} \ \forall abc \in W[b \prec a \wedge c \prec a \longrightarrow b \prec c]$$

The corresponding modal formula is

$$\textbf{5 :} \ \Diamond P \longrightarrow \Box \Diamond P$$

Let us study the structure of K45 models. Consider any model M of the system. Just as K5 models M can have zero or more roots. Where as in K5 the root could only access a set of worlds, in K45 models the transitive property makes every other world accessible from the root(s). The rest of the worlds (as is *Strict Euclidean Models*) is totally connected. Thus, a K45 model has zero or more roots, which can access all the rest of the worlds (except other roots). Let us call the set of worlds accept the roots W_1. Every world $w \in W_1$ can access every other world $w' \in W_1$.

To prove whether ϕ is a theorems or not in K45 logic, we will try to build a *Strict K45 model* that satisfies the negation of the formula ,i.e, $\neg\phi$. The strict K45 model will have a single root, which is connected to rest of the worlds (of W_1).

7.2 The Tableau Rule

We already have covered tableau rule for K45. We will use the same tableau rule described in K5 tableau method.

$$(K45) \quad \frac{w \Vdash S; \Box Y; \Diamond Z}{w_1 \Vdash \{Y; \Box Y; Q_1; \Diamond Z\} \S \ldots \S w_r \Vdash \{Y; \Box Y; Q_r; \Diamond Z\}}.$$

In K5 tableau methods the strict K5 model was divided into two parts one contained the root, for which (k5) rule was applicable and the rule (K45) was applicable for the rest. (K45) was proven to be both sound and complete for the rest of the worlds (which where totally connected). We bring the same concepts here.

The algorithm is straight forward enough. We are trying to build a K45 model which satisfies the formula set X. We start an arbitrary world w_0 and repeatedly apply (K45) rule to the Numerator.

Algorithm

1. $\mathcal{T}(w_0) := X$.

2. $W := \{w_0\}, R := \emptyset$

3. $\mathcal{T}(w_0) := \mathcal{C}P(\mathcal{T}(w_0))$.

4. **if** $\mathcal{T}(w)$ *is inconsistent* **then**
   Stop. Declare X to be unsatisfiable

   end

5. Apply $(K45)$ to the numerator. (* $u \Vdash X$ *)
   It creates a number of descendants of u, $W_1 = \{w_1, \ldots, w_m\}$.
   (* Here we have an opportunity for some optimization. Instead of creating each and every descendant it suffices to explore only one of them. Since every world is connected to every other we will eventually discover all the worlds. in a depth first manner. Thus, K45 rule reduces to

$$(K45) \quad \frac{u \Vdash S; \Box Y; \Diamond Z}{w_r \Vdash \{Y; \Box Y; Q_r; \Diamond Z\}}$$

   *)

6. **foreach** $w_i \in W_1$ **do**
   $\mathcal{T}(w_i) := \mathcal{C}(P)(\mathcal{T}(w_i))$.
   if $\mathcal{T}(w_i)$ *is inconsistent* **then**
     Stop. Declare X to be unsatisfiable
   end
   else
     if $\exists w' \in W.\mathcal{T}(w_i) \subseteq \mathcal{S}(w')$ **then**
       $R := R \cup \{(u, w')\}$
     end
     else
       $W := W \cup \{w_i\}$.
       $R := R \cup \{(u, w_i)\}$.
       Exit the for loop.(* The optimization *)
     end
   end

   end

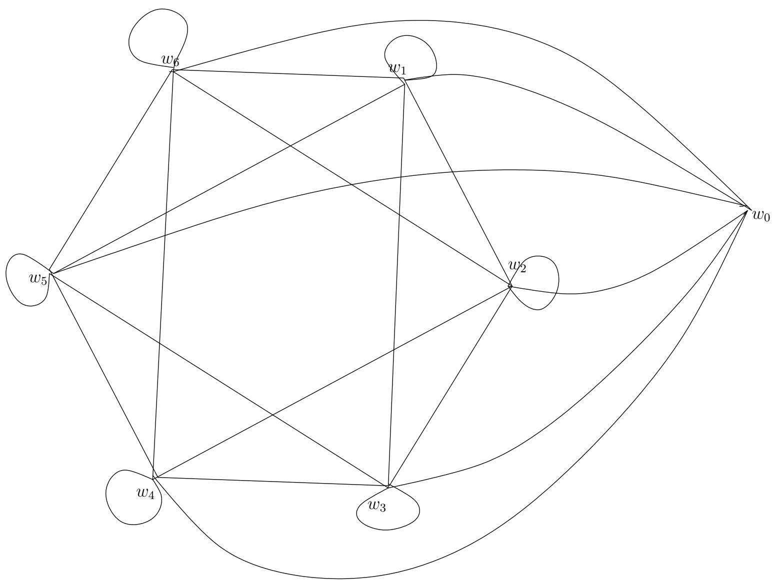
7. **if** *No worlds where added to* W **then**
   Stop.

   end

   else
   Apply 5 to the Numerator

   end

7.3 Correctness



In the above figure w_0 is the root, and $W_1 = \{w_1, w_2, w_3, w_4, w_5, w_6\}$

Much of the correctness prove was already considered in the correctness of the (K5) tableau Methods. Here we will see that algorithm is indeed both sound and complete with respect to K45 Logic system.

Soundness
To show soundness of every step we show that if the numerator was satisfiable before the step then it is still satisfiable after the application.

The $\mathcal{CP}$ rules have been proved sound for any logic system. What we are interested is the soundness of the (K45) rule.

Lemma 7.3.1 *The tableau rule (K45) is sound for **K45** logic.*

Proof Let M $= \langle W, R, \nu \rangle$ be a model of the **K45** system. This means that R is both transitive and Euclidean. Consider the numerator

$$\mathcal{N} := w_0 \Vdash X = S \cup \Box Y \cup \Diamond Z$$

Where S is the set of purely propositional sentences. $\Box Y = \{P | \Box P \in X\} \cup \{\neg P | \neg \Diamond P \in X\}$. Similarly Z is also defined. For each $Q_r inZ$ there exists one world w_r which as accessible from w_0. Thus,

$$w_r \Vdash \{Q_r\} \cup Y$$

Since, R is transitive, we also have from lemma 5.3.1, that

$$w_r \Vdash \Box Y$$

Similarly, being an Euclidean too, we can conclude the following from lemma 6.3.1

$$w_r \Vdash \Diamond Z$$

■

It can be seen that neither the degree nor the modal depth of the formulas is guaranteed to reduce. Thus, we have to keep a look out for the loops. The rest of the steps does exactly that. When an new branch lead to an already explored world we create a back edge and proceed along other branches.

Completeness

Completeness prove (as like so many before) tries to build a model whose accessibility relation is both Euclidean and transitive. We will rely on Model graph theorem 2.6.2 and and its corollary 2.6.3.

Lemma 7.3.2 *The tableau proof method is complete for **K45** logic system.*

Proof Suppose no contradiction was found after the execution of the proof method. We will build a strict K45 model using the information obtained from the tableau method. Let X be the formula set that we started with (after $\mathcal{CP}$ saturation).

If $Z = \emptyset$, then the desired model is $\langle \{w_0\}, \emptyset, \nu \rangle$. It can be observed that all the model graph property are trivially satisfied.

By the end of the execution of the algorithm we have a set of world W and an accessibility relation R. Let $Z = \{Q_1, Q_2, \ldots, Q_m\}$. We have discovered a new world w_r which each application of the (K45) rule. Let the following sequence show the worlds discovered. $W = \{w_0, w_1, \ldots\}$. Since the $\mathcal{S}(w_i)$ are unique and subset of the subformulas ($\mathcal{SF}(X)$) of X, W is finite. Let w_x be the last world to be discovered. This shows that all the successors of w_x has already explored. Let w_y be the oldest (earliest discovered) which has an back edge from w_x.

Build $W_0 = \{w_0, w_y, w_{y+1}, \ldots, w_X\}$. Let R_0 be the reflexive, transitive and symmetric closure of $R(W - \{w_0\})$. The $w \in \nu(P)$ if $P \in \mathcal{S}(w)$. We know that the reflexive transitive and symmetric closure of a loop gives us clique. The lemma 6.3.4 shows that strict K45 model does has clique. The lemma 6.3.5 shows that the set $W_0 - \{w_0\}$ contains at least one world which satisfies P for each $\Diamond P \in \mathcal{SF}(X)$. Thus, model graph properties 2 and 3 are satisfied. ■

The final *strict K45 model* is given as

$$M = \langle W_0, R', \nu \rangle$$

where $R' = R_0 \cup \{(w_0, w_i) | \forall w_i \in W_0 - \{w_0\}\}$

Bibliography

[1] Richard M Smullyan *First Order Logic*

[2] Melvin Fitting *Tableau Method for Modal Logic*

[3] Sally Popkorn *First steps in Modal Logic*

[4] Rajeev Göre *Technical report TR-ARP-15-95*