Lecture 9

Propositional Modal Logic

February 5, 1998
Lecturer: Richard Ladner
Notes: Todd Millstein

9.1. Syntax and Semantics

We will be discussing the modal logic K, which is named for Saul Kripke, who analyzed it in 1963. The logic is propositional logic augmented with a new unary operator, denoted $\Box$. We can informally understand $\Box$ to mean several different things: "it is necessary that ..., ' "I know that ...," "I believe that ...," "it is true in all next states that ..."

Syntax: If α is a well-formed formula in K, then so is $\Box\alpha$.

We introduce a K-*model*, which is essentially a special kind of directed graph. Each node represents a "world" or " state" in which every boolean variable has an assignment of either true or false. An edge from state s_1 to state s_2 means that we can get from s_1 to s_2.

More formally, a K-model M is a triple (S, π, R) where

- S is a (possibly infinite) set of world states.
- π is a function mapping each world state in S to a truth assignment for all the boolean variables.
- $R \subseteq S \times S$ is a relation on S.

Semantics: Given a K-model $M = (S, \pi, R)$ and a state $s \in S$, we define the semantics of a modal logic formula inductively as follows (x denotes a boolean variable, and α and β denote arbitrary modal logic formulas):

- $M, s \models x$ if $\pi(s)(x) = \mathrm{T}$
- $M, s \models \neg\alpha$ if $M, s \not\models \alpha$
- $M, s \models \alpha \vee \beta$ if $M, s \models \alpha$ or $M, s \models \beta$
- $M, s \models \Box\alpha$ if $\forall t \in S, (s,t) \in R \Rightarrow M, t \models \alpha$

The other propositional connectives ($\wedge, \Rightarrow, \equiv$) can be defined in terms of the above connectives in the usual way. In addition, we can define a new operator, $\Diamond$, defined as follows: $\Diamond\alpha \equiv \neg\Box\neg\alpha$. The intuitive meaning is that we can get to a state where α is true from the current state.

As an example, consider the K-model $M = (S, \pi, R)$ defined as follows:

- $S = \{s_1, s_2, s_3\}$
- $\pi = \{(s_1, \{(x,T),(y,T)\}), (s_2, \{(x,F),(y,T)\}), (s_3, \{(x,T),(y,T)\})\}$
- $R = \{(s_1, s_2), (s_1, s_3)\}$

In this example, $M, s_1 \models y \wedge \Box y$, but $M, s_1 \not\models x \wedge \Box x$. Also, $M, s_1 \models \Diamond x$.

9.2. K-validity and K-satisfiability

As in propositional logic, we can define notions of validity and satisfiability. K-validity is defined as follows:

$$\models_K \alpha \text{ if for all } K\text{-models } M = (S, \pi, R) \text{ and states } s \in S,\ M, s \models \alpha$$

Correspondingly, we say that a formula α is K-satisfiable if there exist a K-model $M = (S, \pi, R)$ and a state $s \in S$ such that $M, s \models \alpha$. Finally, K-model checking is the problem of deciding, given a K-model $M = (S, \pi, R)$, a state $s \in S$, and a formula α, whether $M, s \models \alpha$.

An example of K-validity is $\models_K (\Box\alpha \wedge \Box(\alpha \Rightarrow \beta)) \Rightarrow \Box\beta$. A non-example of K-validity is $\not\models_K (\Diamond\alpha \wedge \Diamond(\alpha \Rightarrow \beta)) \Rightarrow \Diamond B$.

Finally, a meta-rule relating the K-validity of two formulas is the "Rule of Necessity": if $\models_K \alpha$ then $\models_K \Box\alpha$.

Now we prove a complexity result about K-satisfiability.

Theorem 9.1: K-satisfiability is PSPACE-complete for log-space reductions.

Proof: We don't show here that K-satisfiability $\in$ PSPACE because it is a fairly difficult proof. Therefore, we only prove that any language $L \in$ PSPACE can be reduced to K-satisfiability in log-space. In particular, we'll show that B_ω, the set of true Quantified Boolean Formulas, can be log-space reduced to K-satisfiability. The result then follows from the fact that B_ω is PSPACE-complete.

Assume we are given a quantified boolean formula $F_{B_\omega} = Q_1 x_1 \ldots Q_k x_k A$, where each $Q_i \in \{\exists, \forall\}$. Further assume the formula is in prenex form, so A contains no quantifiers. We will create a modal logic formula F_K which is K-satisfiable if and only if F_{B_ω} is true.

The informal idea of the construction is as follows: The modal logic formula which we build will force a certain structure on any K-model and state s which K-satisfies it. Starting from s, the K-model's branching must mimic the order of quantifiers in F_{B_ω}. In particular, the construction will ensure that, when a universal quantifier is encountered, the K-model will be forced to branch from the current state, setting the associated quantified variable to true and false respectively on each branch. Similarly, for every existential quantifier encountered, the K-model will be forced to choose an appropriate value for the associated variable at that point in the K-model.

In this way, the construction requires k levels of states above s, one level per quantifier of F_{B_ω}. The levels are indexed 0 to k, with s at level 0. At level k, we simply need to ensure that the boolean formula A holds. To keep track of the properties that must hold at each level, we introduce a set of k *level variables* $y_0, y_1, \ldots, y_k$. The construction enforces that y_i is true iff we are currently at level i in the K-model.

A convenient definition for the construction of F_K is the following:

$$\Box^{(m)}\alpha = \alpha \wedge \Box\alpha \wedge \Box\Box\alpha \ldots \wedge \Box^m \alpha$$

Now we formally describe the construction. F_K is defined to be the conjunction of the following modal logic formulas (some of these are actually modal logic schemas, to be instantiated for all i such that $0 \leq i < k$):

1. $y_0 \wedge \neg y_1 \wedge \ldots \wedge \neg y_k$ This ensures that the initial state s is considered to be in level 0.
2. $\Box^{(k)}(y_i \Rightarrow \Box(\neg y_0 \wedge \ldots \wedge \neg y_i \wedge y_{i+1} \wedge \neg y_{i+2} \wedge \ldots \wedge \neg y_k))$ This ensures that y_i and only y_i is true at level i.
3. $\Box^{(k)}(y_i \Rightarrow ((x_i \Rightarrow \Box^{(k)} x_i) \wedge (\neg x_i \Rightarrow \Box^{(k)} \neg x_i)))$ This ensures that the truth value of x_i at some state in level i remains the truth value of x_i for all transitively reachable states in subsequent levels.
4. $\Box^{(k)}(y_i \Rightarrow (\Diamond(y_{i+1} \wedge x_{i+1}) \vee \Diamond(y_{i+1} \wedge \neg x_{i+1})))$, for all i such that $Q_{i+1} = \exists$. This ensures that each state at level i branches to at least one state in the $(i+1)$st level. This thereby mimics the behavior of the existentially quantified variable x_{i+1} of F_{B_ω}.
5. $\Box^{(k)}(y_i \Rightarrow (\Diamond(y_{i+1} \wedge x_{i+1}) \wedge \Diamond(y_{i+1} \wedge \neg x_{i+1})))$, for all i such that $Q_{i+1} = \forall$. This ensures that each state at level i branches to at least two states in the $(i+1)$st level: one which sets x_{i+1} to true and another which sets it to false. This thereby mimics the behavior of the universally quantified variable x_{i+1} of F_{B_ω}.
6. $\Box^{(k)}(y_k \Rightarrow A)$ This ensures that A holds at the last level.

This completes the construction. By the comments next to each subformula above, the correctness of the construction should be intuitively clear. We can easily see that each of the subformulas above are polynomial size with respect to the length of F_{B_ω}. Further, these formulas can be generated by a machine using only space logarithmic in the length of F_{B_ω}, so the reduction is log-space. $\Box$

Exercise 9.2 (Thought Problem): Build a modal logic formula F such that F is K-satisfiable and all K-models which K-satisfy F have a number of states exponential in the length of F.

Solution: We can use ideas from the construction in the above theorem. In particular, the simulation of a universal quantifier requires the K-satisfying K-model to double in size because it must branch to at least two states: one in which the quantified variable is true and one in which it is false. Intuitively, a succession of such branches will give us the exponential size.

In particular, define F to be the conjunction of the following modal logic formulas:

1. $y_0 \wedge \neg y_1 \wedge \ldots \wedge \neg y_k$ This ensures that the state s which K-satisfies F is considered to be in level 0.

2. $\Box^{(k)}(y_i \Rightarrow \Box(\neg y_0 \wedge \ldots \wedge \neg y_i \wedge y_{i+1} \wedge \neg y_{i+2} \wedge \ldots \wedge \neg y_k))$ This ensures that all states that are i edges away from s are considered to be in level i.

3. $\Box^{(k)}(y_i \Rightarrow (\Diamond x_{i+1} \wedge \Diamond \neg x_{i+1}))$ This ensures that a K-satisfying K-model must do a "universal branch" from each state at level i. That is, from each state at level i, there is an edge to a state in which x_{i+1} is true and an edge to a state in which x_{i+1} is false.

4. $\Box^{(k)}(y_i \Rightarrow ((x_i \Rightarrow \Box^{(k)} x_i) \wedge (\neg x_i \Rightarrow \Box^{(k)} \neg x_i)))$ This ensures that the truth value of variable x_i at level i remains the truth value of x_i for all subsequent levels.

This completes the construction. To K-satisfy this formula, a K-model has to have a state s from which we can transitively reach one state in level k per possible truth assignment of the variables $x_1, \ldots, x_k$. Since there are 2^k such assignments, this gives us the exponential lower bound.

One caveat about this construction is that, although a K-satisfying K-model will be exponential in the number of variables of F, F is not linear in the number of variables. For example, the formulas in the above construction that start with $\Box^{(k)}$ have size at least $\Omega(k^2)$. Therefore, a challenge problem is to find a construction such that the K-model is still exponential, but F is linear in the number of variables.