Why is modal logic decidable

Petros Potikas

NTUA

9/5/2017
Outline

1 Introduction

2 Syntax

3 Semantics

4 Modal logic vs. First-Order Logic
About modal logic

What is modal logic?
About modal logic

What is modal logic? A modal is anything that qualifies the truth of a sentence.
What is modal logic? A *modal* is anything that qualifies the truth of a sentence.

\(\square p, \Diamond p\)
About modal logic

What is modal logic? A *modal* is anything that qualifies the truth of a sentence.

$\Box p$, $\Diamond p$

Historically it begins from Aristotle goes to Leibniz.
About modal logic

What is modal logic? A modal is anything that qualifies the truth of a sentence. □p, ◊p

Historically it begins from Aristotle goes to Leibniz. Continues in 1912 with C.I. Lewis and Kripke in the 60’s.
About modal logic

What is modal logic? A modal is anything that qualifies the truth of a sentence. □p, ◇p

Historically it begins from Aristotle goes to Leibniz. Continues in 1912 with C.I. Lewis and Kripke in the 60’s.

Applications of ML: artificial intelligence (knowledge representation), program verification, hardware verification, and distributed computing
About modal logic

What is modal logic? A *modal* is anything that qualifies the truth of a sentence.

\(\square p, \Diamond p\)

Historically it begins from Aristotle goes to Leibniz. Continues in 1912 with C.I. Lewis and Kripke in the 60’s.

Applications of ML: artificial intelligence (knowledge representation), program verification, hardware verification, and distributed computing

Reason: good balance between expressive power and computational complexity
Computational problems

Two computational problems:

1. **Model-checking** problem: is a given formula true at a given state at a given Kripke structure

2. **Validity** problem: is a given formula true in all states of all Kripke structures
Computational problems

- Both problems are decidable.
Computational problems

- Both problems are decidable.
- Model-checking can be solved in linear time, while validity is PSPACE-complete.
Computational problems

- Both problems are decidable.
- Model-checking can be solved in linear time, while validity is PSPACE-complete.
- However, ML is a fragment of first order logic (FO).
Computational problems

- Both problems are decidable.
- Model-checking can be solved in linear time, while validity is PSPACE-complete.
- However, ML is a fragment of first order logic (FO).
- In first order logic, the above problems are computationally hard.
Computational problems

- Both problems are decidable.
- Model-checking can be solved in linear time, while validity is PSPACE-complete.
- However, ML is a fragment of first order logic (FO).
- In first order logic, the above problems are computationally hard.
- Only very restricted fragments of FO are decidable, typically defined in terms of bounded quantifier alternation.
Computational problems

- Both problems are decidable.
- Model-checking can be solved in linear time, while validity is PSPACE-complete.
- However, ML is a fragment of first order logic (FO).
- In first order logic, the above problems are computationally hard.
- Only very restricted fragments of FO are decidable, typically defined in terms of bounded quantifier alternation.
- But in ML we have arbitrary nesting of modalities.
Computational problems

- Both problems are decidable.
- Model-checking can be solved in linear time, while validity is PSPACE-complete.
- However, ML is a fragment of first order logic (FO).
- In first order logic, the above problems are computationally hard.
- Only very restricted fragments of FO are decidable, typically defined in terms of bounded quantifier alternation.
- But in ML we have arbitrary nesting of modalities.
- So, this cannot be captured by bounded quantifier alternation.
Modal logic and first-order logic with two variables

- Taking a closer look at ML, we see that it is a fragment of 2-variable first-order logic $\text{FO}^2$. 

- $\text{FO}^2$ is more tractable than full first-order logic. However, this is not enough, as extensions of ML, such as computation-tree logic (CTL), are not captured by $\text{FO}^2$. CTL can be viewed as a fragment of 2-variable fixpoint logic ($\text{FP}^2$).

- $\text{FP}^2$ does not enjoy the nice computational properties of $\text{FO}^2$.

- Decidability of CTL can be explained by the tree model property, which is enjoyed by CTL, but not by $\text{FP}^2$.

- Finally, the tree model property leads to automata-based decision procedures.
Taking a closer look at ML, we see that it is a fragment of 2-variable first-order logic $\text{FO}^2$. $\text{FO}^2$ is more tractable than full first-order logic.
Taking a closer look at ML, we see that it is a fragment of 2-variable first-order logic $\text{FO}^2$.

$\text{FO}^2$ is more tractable than full first-order logic.

However, this is not enough, as extensions of ML, as computation-tree logic (CTL) while not captured by $\text{FO}^2$. 

$\text{CTL}$ can be viewed as a fragment of 2-variable fixpoint logic ($\text{FP}^2$).

$\text{FP}^2$ does not enjoy the nice computational properties of $\text{FO}^2$.

Decidability of $\text{CTL}$ can be explained by tree-model property, which is enjoyed by $\text{CTL}$, but not by $\text{FP}^2$.

Finally, the tree model property leads to automata-based decision procedures.
Modal logic and first-order logic with two variables

- Taking a closer look at ML, we see that it is a fragment of 2-variable first-order logic FO^2.
- FO^2 is more tractable than full first-order logic.
- However, this is not enough, as extensions of ML, as computation-tree logic (CTL) while not captured by FO^2.
- CTL can be viewed as a fragment of 2-variable fixpoint logic (FP^2).
Taking a closer look at ML, we see that it is a fragments of 2-variable first-order logic $\text{FO}^2$.

$\text{FO}^2$ is more tractable than full first-order logic.

However, this is not enough, as extensions of ML, as computation-tree logic (CTL) while not captured by $\text{FO}^2$.

CTL can be viewed as a fragment of 2-variable fixpoint logic ($\text{FP}^2$).

$\text{FP}^2$ does not enjoy the nice computational properties of $\text{FO}^2$. 
Modal logic and first-order logic with two variables

- Taking a closer look at ML, we see that it is a fragment of 2-variable first-order logic $\text{FO}^2$.
- $\text{FO}^2$ is more tractable than full first-order logic.
- However, this is not enough, as extensions of ML, as computation-tree logic (CTL) while not captured by $\text{FO}^2$.
- CTL can be viewed as a fragment of 2-variable fixpoint logic ($\text{FP}^2$).
- $\text{FP}^2$ does not enjoy the nice computational properties of $\text{FO}^2$.
- Decidability of CTL can be explained by tree-model property, which is enjoyed by CTL, but not by $\text{FP}^2$. 
Taking a closer look at ML, we see that it is a fragments of 2-variable first-order logic $\text{FO}^2$.

$\text{FO}^2$ is more tractable than full first-order logic.

However, this is not enough, as extensions of ML, as computation-tree logic (CTL) while not captured by $\text{FO}^2$

CTL can be viewed as a fragment of 2-variable fixpoint logic ($\text{FP}^2$)

$\text{FP}^2$ does not enjoy the nice computational properties of $\text{FO}^2$.

Decidability of CTL can be explained by *tree-model property*, which is enjoyed by CTL, but not by $\text{FP}^2$.

Finally, the tree model property leads to automata-based decision procedures.
Syntax

Definition

(The Basic Modal Language) Let \( \mathcal{P} = \{\mathcal{P}_0, \mathcal{P}_1, \mathcal{P}_2, \ldots\} \) be a set of sentence letters, or atomic propositions. We also include two special propositions \( \top \) and \( \bot \) meaning ‘true’ and ‘false’ respectively. The set of well-formed formulas of modal logic is the smallest set generated by the following grammar:

\[
\mathcal{P}_0, \mathcal{P}_1, \mathcal{P}_2, \ldots \mid \top \mid \bot \mid \neg A \mid A \lor B \mid A \land B \mid A \rightarrow B \mid \Box A \mid \Diamond A
\]

Examples

Modal formulas include: \( \Box \bot, \mathcal{P}_0 \rightarrow \Diamond (\mathcal{P}_1 \land \mathcal{P}_2) \).
Truth

- A *Kripke structure* $M$ is a tuple $(S, \pi, R)$, where $S$ is set of states (or *possible worlds*), $\pi : \mathbb{P} \to 2^S$, and $R$ a binary relation on $S$. 

Truth conditions:

1. $(M, s) | = P$ iff $s \in \pi(P)$
2. $(M, s) | = \top$
3. $(M, s) | \neq \bot$
4. $(M, s) | = \neg A$ iff not $(M, s) | = A$
5. $(M, s) | = A \lor B$ iff either $(M, s) | = A$ or $(M, s) | = B$, or both
6. $(M, s) | = \Box A$ iff for every $t$, s.t. $R(s, t)$, $(M, t) | = A$
Truth

- A Kripke structure $M$ is a tuple $(S, \pi, R)$, where $S$ is set of states (or possible worlds), $\pi : \mathbb{P} \to 2^S$, and $R$ a binary relation on $S$.
- $(M, s) \models A$, sentence $A$ is true at $s$ in $M$
Truth

- A Kripke structure $M$ is a tuple $(S, \pi, R)$, where $S$ is set of states (or possible worlds), $\pi : P \rightarrow 2^S$, and $R$ a binary relation on $S$.
- $(M, s) \models A$, sentence $A$ is true at $s$ in $M$

Truth conditions:

1. $(M, s) \models P_i$ iff $s \in \pi(P_i)$
2. $(M, s) \models \top$
3. $(M, s) \models \bot$
4. $(M, s) \models \neg A$ iff not $(M, s) \models A$
5. $(M, s) \models A \lor B$ iff either $(M, s) \models A$ or, $(M, s) \models B$, or both
6. $(M, s) \models \Box A$ iff for every $t$, s.t. $R(s, t), (M, t) \models A$
Truth

- A *Kripke structure* $M$ is a tuple $(S, \pi, R)$, where $S$ is set of states (or *possible worlds*), $\pi : P \to 2^S$, and $R$ a binary relation on $S$.
- $(M, s) \models A$, sentence $A$ is true at $s$ in $M$

Truth conditions:

1. $(M, s) \models P_i$ iff $s \in \pi(P_i)$
2. $(M, s) \models \top$
3. $(M, s) \not\models \bot$
4. $(M, s) \models \neg A$ iff not $(M, s) \models A$
5. $(M, s) \models A \lor B$ iff either $(M, s) \models A$ or, $(M, s) \models B$, or both
6. $(M, s) \models \Box A$ iff for every $t$, s.t. $R(s, t), (M, t) \models A$

- A sentence true at every possible world in every model is said to be *valid*, written $\models A$
Model-checking problem

Theorem

There is an algorithm that, given a finite Kripke structure $M$, a state $s$ of $M$ and a modal formula $\phi$, determines whether $(M, s) \models \phi$ in time $O(||M|| \times |\phi|)$. 

Proof. Let $\phi_1, \ldots, \phi_m$ be the subformulas of $\phi$ listed in order of length. Thus $\phi_m = \phi$, and if $\phi_i$ is a subformula of $\phi_j$, then $i < j$. There are at most $|\phi|$ subformulas, so $m \leq |\phi|$. By induction on $k$, we can show that we can label each state $s$ with $\phi_j$ or $\neg \phi_j$, for $j = 1, \ldots, k$, depending on whether or not $\phi_j$ is true in $s$ in time $O(||M|| \times |\phi|)$. Only interesting case is $\phi_{k+1} = \square \phi_j$, $j < k + 1$. By induction hypothesis, we have that each state has already been labeled with $\phi_j$ or $\neg \phi_j$, so we know if node $s$ can be labeled with $\phi_{k+1}$ or not in time $O(||M|| \times |\phi|)$. 
Theorem

There is an algorithm that, given a finite Kripke structure $M$, a state $s$ of $M$ and a modal formula $\phi$, determines whether $(M, s) \models \phi$ in time $O(||M|| \times |\phi|)$.

$||M||$: number of states in $S$, and number of pairs in $R$
Theorem

There is an algorithm that, given a finite Kripke structure \( M \), a state \( s \) of \( M \) and a modal formula \( \phi \), determines whether \( (M, s) \models \phi \) in time \( O(||M|| \times |\phi|) \).

\( ||M|| \): number of states in \( S \), and number of pairs in \( R \)

\( |\phi| \): length of \( \phi \), number of symbols is \( \phi \)
Model-checking problem

**Theorem**

There is an algorithm that, given a finite Kripke structure $M$, a state $s$ of $M$ and a modal formula $\phi$, determines whether $(M, s) \models \phi$ in time $O(||M|| \times |\phi|)$.

$||M||$: number of states in $S$, and number of pairs in $R$

$|\phi|$: length of $\phi$, number of symbols is $\phi$

**Proof.**

Let $\phi_1, ..., \phi_m$ be the subformulas of $\phi$ listed in order of length. Thus $\phi_m = \phi$, and if $\phi_i$ is a subformulas of $\phi_j$, then $i < j$. There are at most $|\phi|$ subformulas, so $m \leq |\phi|$. By induction on $k$, we can show that we can label each state $s$ with $\phi_j$ or $\neg \phi_j$, for $j = 1, ..., k$, depending on whether or not $\phi_j$ is true in $s$ in time $O(k||M||)$. Only interesting case is $\phi_{k+1} = \Box \phi_j$, $j < k + 1$. By induction hypothesis, we have that each state has already been labeled with $\phi_j$ or $\neg \phi_j$, so we know if node $s$ can be labeled with $\phi_{k+1}$ or not in time $O(||M||)$. 

□
Characterizing the properties of necessity

Set of valid formulas can be viewed as a characterization of the properties of necessity
Characterizing the properties of necessity

Set of valid formulas can be viewed as a characterization of the properties of necessity.

Two approaches:

1. *Proof-theoretic*: all properties of necessity can be formally derived from a short list of basic properties.

2. *Algorithmic*: we study algorithms that recognize properties of necessity and consider their computational complexity.
Properties of necessity

Some basic properties of necessity:

**Theorem**

For all formulas $\phi$, $\psi$, and Kripke structures $M$:

1. if $\phi$ is an instance of a propositional tautology, then $M \models \phi$
2. if $M \models \phi$ and $M \models \phi \to \psi$, then $M \models \psi$
3. $M \models (\Box \phi \land \Box (\phi \to \psi)) \to \Box \psi$
4. if $M \models \phi$, then $M \models \Box \phi$
Consider the following axiom system $\mathcal{K}$:

- (A1) All tautologies of propositional calculus
- (A2) $(\Box \phi \land \Box (\phi \rightarrow \psi)) \rightarrow \Box \psi$ (Distribution axiom)
- (R1) From $\phi$ and $\phi \rightarrow \psi$ infer $\psi$ (Modus ponens)
- (R2) From $\phi$ infer $\Box \phi$ (Generalization)

Theorem (Kripke '63)

$\mathcal{K}$ is a sound and complete axiom system.
Characterizing the properties of necessity: Proof-theoretic

Consider the following axiom system $\mathcal{K}$:

- (A1) All tautologies of propositional calculus
- (A2) $(\Box \phi \land \Box (\phi \rightarrow \psi)) \rightarrow \Box \psi$ (Distribution axiom)
- (R1) From $\phi$ and $\phi \rightarrow \psi$ infer $\psi$ (Modus ponens)
- (R2) From $\phi$ infer $\Box \phi$ (Generalization)

**Theorem (Kripke ’63)**

$\mathcal{K}$ is a sound and complete axiom system.
Characterizing the properties of necessity: algorithmically

- The above characterization of the properties of necessity is not constructive.

Theorem (Fischer, Ladner '79)
If a modal formula \( \phi \) is satisfiable, then \( \phi \) is satisfiable in a Kripke structure with at most \( 2^{\|\phi\|} \) states.
The above characterization of the properties of necessity is not constructive.

An algorithm that recognizes valid formulas is another characterization.
Characterizing the properties of necessity: algorithmically

- The above characterization of the properties of necessity is not constructive.
- An algorithm that recognizes valid formulas is another characterization.
- First step, if a formula is satisfiable, it is also satisfiable in a finite structure of bounded size (bounded-model property).
Characterizing the properties of necessity: algorithmically

- The above characterization of the properties of necessity is not constructive.
- An algorithm that recognizes valid formulas is another characterization.
- First step, if a formula is satisfiable, it is also satisfiable in a finite structure of bounded size (bounded-model property).
- Stronger than the finite-model property, which asserts that if a formula is satisfiable, then it is satisfiable in a finite structure.
Characterizing the properties of necessity: algorithmically

- The above characterization of the properties of necessity is not constructive.
- An algorithm that recognizes valid formulas is another characterization.
- First step, if a formula is satisfiable, it is also satisfiable in a finite structure of bounded size (bounded-model property).
- Stronger than the finite-model property, which asserts that if a formula is satisfiable, then it is satisfiable in a finite structure.
- This implies that formula $\phi$ is valid in all Kripke structures iff $\phi$ is valid in all finite Kripke structures.
The above characterization of the properties of necessity is not constructive.

An algorithm that recognizes valid formulas is another characterization.

First step, if a formula is satisfiable, it is also satisfiable in a finite structure of bounded size (bounded-model property).

Stronger than the finite-model property, which asserts that if a formula is satisfiable, then it is satisfiable in a finite structure.

This implies that formula $\phi$ is valid in all Kripke structures iff $\phi$ is valid in all finite Kripke structures.

**Theorem (Fischer,Ladner ’79)**

*If a modal formula $\phi$ is satisfiable, then $\phi$ is satisfiable in a Kripke structure with at most $2^{|\phi|}$ states.*
Characterizing the properties of necessity: algorithmically

From the above Theorem we can get an algorithm (not efficient) for testing validity of a formula $\phi$: construct all Kripke structures with at most $2^{\mid\phi\mid}$ states and check if the formula is true in every state of each of these structures.
Characterizing the properties of necessity: algorithmically

- From the above Theorem we can get an algorithm (not efficient) for testing validity of a formula $\phi$: construct all Kripke structures with at most $2^{\left|\phi\right|}$ states and check if the formula is true in every state of each of these structures.

- The “inherent difficulty” of the problem is given by the next theorem:

**Theorem (Ladner ’77)**

The validity problem for modal logic is PSPACE-complete.
Modal logic vs. First-Order Logic

- Modal logic can be viewed as a fragment of first-order logic.
Modal logic vs. First-Order Logic

- Modal logic can be viewed as a fragment of first-order logic.
- The states in a Kripke structure correspond to domain elements in a relational structure and modalities correspond to quantifiers.
Modal logic vs. First-Order Logic

- Modal logic can be viewed as a fragment of first-order logic.
- The states in a Kripke structure correspond to domain elements in a relational structure and modalities correspond to quantifiers.
- Given a set $\mathbb{P}$ of propositional constants, let the vocabulary $\mathbb{P}^*$ consist of unary predicate $q$ corresponding to each propositional constant $q$ in $\mathbb{P}$, as well as binary predicate $R$. 

Petros Potikas (NTUA)
Modal logic decidability
9/5/2017  15 / 26
Modal logic vs. First-Order Logic

- Modal logic can be viewed as a fragment of first-order logic.
- The states in a Kripke structure correspond to domain elements in a relational structure and modalities correspond to quantifiers.
- Given a set $P$ of propositional constants, let the vocabulary $P^*$ consist of unary predicate $q$ corresponding to each propositional constant $q$ in $P$, as well as binary predicate $R$.
- Every Kripke structure $M$ can be viewed as a relational structure $M^*$ over the vocabulary $P^*$.
Modal logic vs. First-Order Logic

- Modal logic can be viewed as a fragment of first-order logic.
- The states in a Kripke structure correspond to domain elements in a relational structure and modalities correspond to quantifiers.
- Given a set \( \mathbb{P} \) of propositional constants, let the vocabulary \( \mathbb{P}^* \) consist of unary predicate \( q \) corresponding to each propositional constant \( q \) in \( \mathbb{P} \), as well as binary predicate \( R \).
- Every Kripke structure \( M \) can be viewed as a relational structure \( M^* \) over the vocabulary \( \mathbb{P}^* \).
- Formally, a mapping from a Kripke structure \( M = (S, \pi, R) \) to a relational structure \( M^* \) over the vocabulary \( \mathbb{P}^* \) has:
  1. domain of \( M^* \) is \( S \).
  2. for each propositional constant \( q \in \mathbb{P} \), the interpretation of \( q \) in \( M^* \) is the set \( \pi(q) \).
  3. the interpretation of the binary predicate \( R \), is the binary relation \( R \).
Translation of Modal logic to First-Order Logic

A translation from modal formulas into first-order formulas over the vocabulary $\mathbb{P}^*$, so that for every modal formula $\phi$ there is corresponding first-order formula $\phi^*$ with one free variable (ranging over $S$):

1. $q^* = q(x)$ for a propositional constant $q$
2. $(\neg \phi)^* = \neg (\phi^*)$
3. $(\phi \land \psi)^* = (\phi^* \land \psi^*)$
4. $(\Box \phi)^* = (\forall y (R(x, y) \rightarrow \phi^*(x/y))),$ where $y$ is a new variable not appearing in $\phi^*$ and $\phi^*(x/y)$ is the result of replacing all free occurrences of $x$ in $\phi^*$ by $y$
Translation of Modal logic to First-Order Logic

A translation from modal formulas into first-order formulas over the vocabulary $\mathbb{P}^*$, so that for every modal formula $\phi$ there is corresponding first-order formula $\phi^*$ with one free variable (ranging over $S$):

1. $q^* = q(x)$ for a propositional constant $q$
2. $(\neg \phi)^* = \neg(\phi^*)$
3. $(\phi \land \psi)^* = (\phi^* \land \psi^*)$
4. $(\Box \phi)^* = (\forall y (R(x, y) \rightarrow \phi^*(x/y))))$, where $y$ is a new variable not appearing in $\phi^*$ and $\phi^*(x/y)$ is the result of replacing all free occurrences of $x$ in $\phi^*$ by $y$

Example

$(\Box \Diamond q)^* = \forall y (R(x, y) \rightarrow \exists z (R(y, z) \land q(z))))$
Theorem (vBenthem ’74,’85)

1. \((M, s) \models \phi \iff (M^*, V) \models \phi^*(x)\), for each assignment \(V\) s.t. \(V(x) = s\).

2. \(\phi\) is a valid modal formula iff \(\phi^*\) is a valid first-order formula.

\(\phi^*\) is true of exactly the domain elements corresponding to states \(s\) for which \((M, s) \models \phi\).
Translation of Modal logic to First-Order Logic

Is there a paradox?

Modal logic is essentially a first-order logic. Model-checking in first-order logic is PSPACE-complete while in modal logic in linear time. Validity is robustly undecidable in first-order logic (decidable only by bounding the alternation of quantifiers), while in modal logic is PSPACE-complete.

Carefully examining propositional modal logic, reveals that it is a fragment of \( \mathsf{FO}^2 \), e.g., \( \forall x \forall y (R(x, y) \rightarrow R(y, x)) \) is in \( \mathsf{FO}^2 \), while \( \forall x \forall y \forall z (R(x, y) \land R(y, z) \rightarrow R(x, z)) \) is not in \( \mathsf{FO}^2 \).

Two variables suffice to express modal logic formulas, see the above definition, where new variables are introduced only in the last clause:

Example \((\Box\Box q)^* = \forall y (R(x, y) \rightarrow \forall z (R(y, z) \rightarrow q(z)))\).
Translation of Modal logic to First-Order Logic

Is there a paradox?

- Modal logic is essentially a first-order logic.
Translation of Modal logic to First-Order Logic

Is there a paradox?

- Modal logic is essentially a first-order logic.
- Model-checking in first-order logic is PSPACE-complete while in modal logic in linear time.

Example: \( \square \square q \) is translated to:

\[ \forall y \left( R(x, y) \rightarrow \forall z \left( R(y, z) \rightarrow q(z) \right) \right) \]
Translation of Modal logic to First-Order Logic

Is there a paradox?

- Modal logic is essentially a first-order logic.
- Model-checking in first-order logic is PSPACE-complete while in modal logic in linear time.
- Validity is robustly undecidable in first-order logic (decidable only by bounding the alternation of quantifiers), while in modal logic is PSPACE-complete.
Translation of Modal logic to First-Order Logic

Is there a paradox?

- Modal logic is essentially a first-order logic.
- Model-checking in first-order logic is PSPACE-complete while in modal logic in linear time.
- Validity is robustly undecidable in first-order logic (decidable only by bounding the alternation of quantifiers), while in modal logic is PSPACE-complete.
- Carefully examining propositional modal logic, reveals that it is a fragment of 2-variable first-order logic (FO^2), e.g. \( \forall x \forall y (R(x, y) \rightarrow R(y, x)) \) is in FO^2, while \( \forall x \forall y \forall z (R(x, y) \land R(y, z) \rightarrow R(x, z)) \) is not in FO^2.
Translation of Modal logic to First-Order Logic

Is there a paradox?

- Modal logic is essentially a first-order logic.
- Model-checking in first-order logic is PSPACE-complete while in modal logic in linear time.
- Validity is robustly undecidable in first-order logic (decidable only by bounding the alternation of quantifiers), while in modal logic is PSPACE-complete.
- Carefully examining propositional modal logic, reveals that it is a fragment of 2-variable first-order logic ($\text{FO}^2$), e.g. $\forall x \forall y (R(x, y) \rightarrow R(y, x))$ is in $\text{FO}^2$, while $\forall x \forall y \forall z (R(x, y) \land R(y, z) \rightarrow R(x, z))$ is not in $\text{FO}^2$.
- Two variables suffice to express modal logic formulas, see the above definition, where new variables are introduced only in the last clause:
Translation of Modal logic to First-Order Logic

Is there a paradox?

- Modal logic is essentially a first-order logic.
- Model-checking in first-order logic is PSPACE-complete while in modal logic in linear time.
- Validity is robustly undecidable in first-order logic (decidable only by bounding the alternation of quantifiers), while in modal logic is PSPACE-complete.
- Carefully examining propositional modal logic, reveals that it is a fragment of 2-variable first-order logic (FO²), e.g. \( \forall x \forall y (R(x, y) \rightarrow R(y, x)) \) is in FO², while \( \forall x \forall y \forall z (R(x, y) \land R(y, z) \rightarrow R(x, z)) \) is not in FO².
- Two variables suffice to express modal logic formulas, see the above definition, where new variables are introduced only in the last clause:

Example

\[
(\square \square q)^* = \forall y (R(x, y) \rightarrow \forall z (R(y, z) \rightarrow q(z))).
\]
Translation of Modal logic to First-Order Logic

But re-using variables we can avoid introducing new variables. Replace the definition of $\phi^*$ by definition $\phi^+$:

\begin{align*}
q^+ &= q(x) \\
¬\phi^+ &= ¬(\phi^+) \\
(\phi \land \psi)^+ &= (\phi^* \land \psi^+) \\
□\phi^+ &= (\forall y (R(x, y) \rightarrow (\forall x (x = y \rightarrow (\forall y (R(x, y) \rightarrow (∀x (x = y \rightarrow q(x))'))))).
\end{align*}
Translation of Modal logic to First-Order Logic

But re-using variables we can avoid introducing new variables. Replace the definition of $\phi^*$ by definition $\phi^+$:

1. $q^+ = q(x)$ for a propositional constant $q$
2. $(\neg\phi)^+ = \neg(\phi^+)$
3. $(\phi \land \psi)^+ = (\phi^* \land \psi^+)$
4. $(\Box\phi)^+ = (\forall y (R(x, y) \rightarrow \forall x (x = y \rightarrow \phi^+))))$
Translation of Modal logic to First-Order Logic

But re-using variables we can avoid introducing new variables. Replace the definition of $\phi^*$ by definition $\phi^+$:

1. $q^+ = q(x)$ for a propositional constant $q$
2. $(\neg \phi)^+ = \neg (\phi^+)$
3. $(\phi \land \psi)^+ = (\phi^* \land \psi^+)$
4. $(\Box \phi)^+ = (\forall y (R(x, y) \rightarrow \forall x (x = y \rightarrow \phi^+)))$

Example

$((\Box \Box q)^+ = \forall y (R(x, y) \rightarrow \forall x (x = y \rightarrow \forall y (R(x, y) \rightarrow \forall x (x = y \rightarrow q(x)))))$.
Translation of Modal logic to First-Order Logic

Theorem

1. \((M, s) \models \phi \iff (M^*, V) \models \phi^+(x), \text{ for each assignment } V \text{ s.t. } V(x) = s.\)

2. \(\phi\) is a valid modal formula iff \(\phi^+\) is a valid \(FO^2\) formula.
Complexity of $\text{FO}^2$

How hard is to evaluate truth of $\text{FO}^2$ formulas?

Theorem (Immerman '82, Vardi '95)

There is an algorithm that, given a relational structure $M$ over a domain $D$, an $\text{FO}^2$-formula $\phi(x,y)$ and an assignment $V: \{x,y\} \rightarrow D$, determines whether $(M, V) \models \phi$ in time $O(|M|^2 \times |\phi|)$. 
Complexity of $\text{FO}^2$

How hard is to evaluate truth of $\text{FO}^2$ formulas?

**Theorem (Immerman ’82, Vardi ’95)**

There is an algorithm that, given a relational structure $M$ over a domain $D$, an $\text{FO}^2$-formula $\phi(x, y)$ and an assignment $V : \{x, y\} \rightarrow D$, determines whether $(M, V) \models \phi$ in time $O(||M||^2 \times |\phi|)$. 
Complexity of FO$^2$

- Historically, Scott in 1962 showed the first decidability result for FO$^2$, without equality. The full class FO$^2$ was considered by Mortimer in 1975, who proved decidability by showing that it has the finite model property.
Complexity of FO$^2$

- Historically, Scott in 1962 showed the first decidability result for FO$^2$, without equality. The full class FO$^2$ was considered by Mortimer in 1975, who proved decidability by showing that it has the finite model property.

- But Mortimer’s proof shows bounded-model property.

**Theorem**

*If an FO$^2$-formula $\phi$ is satisfiable, then $\phi$ is satisfiable in a relational structure with at most $2^{|\phi|}$ elements.*
Complexity of FO$^2$

- To check the validity of a FO$^2$ formula $\phi$, one has to consider only all structures of exponential size.
Complexity of FO$^2$

- To check the validity of a FO$^2$ formula $\phi$, one has to consider only all structures of exponential size.
- Further, the translation of modal logic to FO$^2$ is linear, so we have Theorem 5.
Complexity of $\text{FO}^2$

- To check the validity of a $\text{FO}^2$ formula $\phi$, one has to consider only all structures of exponential size.
- Further, the translation of modal logic to $\text{FO}^2$ is linear, so we have Theorem 5.
- Note, however, that the validity problem for $\text{FO}^2$ is hard for co-NEXPTIME (F"urer81) and also complete, while from Theorem 6 modal logic is PSPACE-complete.
Complexity of FO$^2$

- To check the validity of a FO$^2$ formula $\phi$, one has to consider only all structures of exponential size.
- Further, the translation of modal logic to FO$^2$ is linear, so we have Theorem 5.
- Note, however, that the validity problem for FO$^2$ is hard for co-NEXPTIME (Förer81) and also complete, while from Theorem 6 modal logic is PSPACE-complete.
- The embedding to FO$^2$ does not give a satisfactory explanation of the tractability of modal logic.
Reflexivity

- In epistemic logic veracity is needed, what is known is true,
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e.
  \( \Box \phi \rightarrow \phi \)
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e. 
  \( \square \phi \rightarrow \phi \)
- Logical properties of necessity are related with the properties of the graph, e.g. veracity is reflexivity
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e. $\Box \phi \rightarrow \phi$
- Logical properties of necessity are related with the properties of the graph, e.g. veracity is reflexivity
- A Kripke structure $M = (S, \pi, R)$ is said to be reflexive if the relation $R$ is reflexive. Let $M_r$ be the class of all reflexive Kripke structures.
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e.
  \( \Box \phi \rightarrow \phi \)
- Logical properties of necessity are related with the properties of the graph, e.g. veracity is reflexivity
- A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive if the relation \( R \) is reflexive. Let \( M_r \) be the class of all reflexive Kripke structures.
- Axiom \( T: \Box p \rightarrow p \)
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e.
  \( \Box \phi \rightarrow \phi \)
- Logical properties of necessity are related with the properties of the graph, e.g. veracity is reflexivity
- A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive if the relation \( R \) is reflexive. Let \( M_r \) be the class of all reflexive Kripke structures.
- Axiom \( T: \Box p \rightarrow p \)

Theorem

\( T \) is sound and complete for \( M_r \).
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e. 
  \(\square \phi \rightarrow \phi\)
- Logical properties of necessity are related with the properties of the graph, e.g. veracity is reflexivity
- A Kripke structure \(M = (S, \pi, R)\) is said to be reflexive if the relation \(R\) is reflexive. Let \(M_r\) be the class of all reflexive Kripke structures.
- Axiom \(\mathcal{T}: \square p \rightarrow p\)

**Theorem**

\(\mathcal{T}\) is sound and complete for \(M_r\).

How hard is validity under the assumption of veracity?
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e. \( \Box \phi \rightarrow \phi \)

- Logical properties of necessity are related with the properties of the graph, e.g. veracity is reflexivity

- A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive if the relation \( R \) is reflexive. Let \( M_r \) be the class of all reflexive Kripke structures.

- Axiom \( \mathcal{T} \): \( \Box p \rightarrow p \)

**Theorem**

\( \mathcal{T} \) is sound and complete for \( M_r \).

How hard is validity under the assumption of veracity?

**Theorem**

The validity problem for modal logic in \( M_r \) is PSPACE-complete.
Reflexivity

- In epistemic logic veracity is needed, what is known is true, i.e.
  \[ \Box \phi \rightarrow \phi \]
- Logical properties of necessity are related with the properties of the graph, e.g. veracity is reflexivity
- A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive if the relation \( R \) is reflexive. Let \( M_r \) be the class of all reflexive Kripke structures.
- Axiom \( T: \Box p \rightarrow p \)

**Theorem**

\( T \) is sound and complete for \( M_r \).

How hard is validity under the assumption of veracity?

**Theorem**

The validity problem for modal logic in \( M_r \) is PSPACE-complete.

**Theorem**

A modal formula \( \phi \) is valid in \( M_r \) iff the \( FO^2 \) \( \forall x(R(x, x) \rightarrow \phi^+) \) is valid.
Axiom system S5

What about other properties of necessity?

1. Positive introspection - "I know what I know": $\Box p \rightarrow \Box \Box p$.

2. Negative introspection - "I know what I don't know": $\neg \Box p \rightarrow \Box \neg \Box p$.

A Kripke structure $M = (S, \pi, R)$ is said to be reflexive, symmetric, transitive if the relation $R$ is reflexive, symmetric, transitive.

Let $M_{rst}$ be the class of all reflexive, symmetric and transitive Kripke structures.

Let $S_5$ be the axiom system obtained from $T$ by adding the two rules of introspection.

Theorem 1: $S_5$ is sound and complete for $M_{rst}$.

The validity problem for $S_5$ is NP-complete.

Symmetry can be expressed by $\text{FO}^2$, $\forall x, y (R(x, y) \rightarrow R(y, x))$, while transitivity cannot $\forall x, y, z (R(x, y) \land R(y, z) \rightarrow R(x, z))$.
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”:
   \[ \Box p \rightarrow \Box \Box p \].

2. Negative introspection - “I know what I don’t know”:
   \[ \neg \Box p \rightarrow \Box \neg \Box p \].

A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive, symmetric, transitive if the relation \( R \) is reflexive, symmetric, transitive.

Let \( M_{rst} \) be the class of all reflexive, symmetric and transitive Kripke structures.

Let \( S_5 \) be the axiom system obtained from \( T \) by adding the two rules of introspection.

Theorem

1. \( S_5 \) is sound and complete for \( M_{rst} \).

2. The validity problem for \( S_5 \) is NP-complete.

Symmetry can be expressed by \( \forall x, y (R(x, y) \rightarrow R(y, x)) \), while transitivity cannot be expressed by \( \forall x, y, z (R(x, y) \land R(y, z) \rightarrow R(x, z)) \).
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”:

\[ \Box p \rightarrow \Box \Box p. \]

2. Negative introspection - “I know what I don’t know”:

\[ \neg \Box p \rightarrow \Box \neg \Box p. \]
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: $\square p \rightarrow \square\square p$.

2. Negative introspection - “I know what I don’t know”: $\neg\square p \rightarrow \square\neg\square p$.

A Kripke structure $M = (S, \pi, R)$ is said to be reflexive, symmetric, transitive if the relation $R$ is reflexive, symmetric, transitive.

Let $M_{rst}$ be the class of all reflexive, symmetric and transitive Kripke structures.

Let $S_5$ be the axiom system obtained from $T$ by adding the two rules of introspection.

Theorem 1

$S_5$ is sound and complete for $M_{rst}$.

The validity problem for $S_5$ is NP-complete.

Symmetry can be expressed by $\text{FO}_2$, $\forall x, y (R(x, y) \rightarrow R(y, x))$, while transitivity cannot $\forall x, y, z (R(x, y) \land R(y, z) \rightarrow R(x, z))$.
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: $\Box p \rightarrow \Box \Box p$.
2. Negative introspection - “I know what I don’t know”:
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: $\square p \rightarrow \square \square p$.
2. Negative introspection - “I know what I don’t know”: $\neg \square p \rightarrow \square \neg \square p$.
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: \( \Box p \rightarrow \Box \Box p \).

2. Negative introspection - “I know what I don’t know”: \( \neg \Box p \rightarrow \Box \neg \Box p \).

A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive, symmetric, transitive if the relation \( R \) is reflexive, symmetric, transitive.
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: $\box p \rightarrow \box\box p$.
2. Negative introspection - “I know what I don’t know”: $\neg\box p \rightarrow \box\neg\box p$.

A Kripke structure $M = (S, \pi, R)$ is said to be reflexive, symmetric, transitive if the relation $R$ is reflexive, symmetric, transitive.

Let $M_{rst}$ be the class of all reflexive, symmetric and transitive Kripke structures.
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: \( \square p \rightarrow \square \square p \).
2. Negative introspection - “I know what I don’t know”: \( \neg \square p \rightarrow \square \neg \square p \).

A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive, symmetric, transitive if the relation \( R \) is reflexive, symmetric, transitive.

Let \( M_{rst} \) be the class of all reflexive, symmetric and transitive Kripke structures.

Let \( S5 \) be the axiom system obtained from \( T \) by adding the two rules of introspection.
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: \( \Box p \to \Box \Box p \).
2. Negative introspection - “I know what I don’t know”: \( \neg \Box p \to \Box \neg \Box p \).

A Kripke structure \( M = (S, \pi, R) \) is said to be reflexive, symmetric, transitive if the relation \( R \) is reflexive, symmetric, transitive.

Let \( M_{rst} \) be the class of all reflexive, symmetric and transitive Kripke structures.

Let \( S5 \) be the axiom system obtained from \( T \) by adding the two rules of introspection.

Theorem

1. \( S5 \) is sound and complete for \( M_{rst} \).
2. The validity problem for \( S5 \) is NP-complete.
Axiom system S5

What about other properties of necessity? Consider introspection:

1. Positive introspection - “I know what I know”: $\square p \rightarrow \square \square p$.
2. Negative introspection - “I know what I don’t know”: $\neg \square p \rightarrow \square \neg \square p$.

- A Kripke structure $M = (S, \pi, R)$ is said to be reflexive, symmetric, transitive if the relation $R$ is reflexive, symmetric, transitive.
- Let $M_{rst}$ be the class of all reflexive, symmetric and transitive Kripke structures.
- Let $S5$ be the axiom system obtained from $T$ by adding the two rules of introspection.

Theorem

1. $S5$ is sound and complete for $M_{rst}$.
2. The validity problem for $S5$ is NP-complete.

Symmetry can be expressed by $\text{FO}^2$, $\forall x, y (R(x, y) \rightarrow R(y, x))$, while transitivity cannot $\forall x, y, z (R(x, y) \land R(y, z) \rightarrow R(x, z))$. 
About decidability of modal logic

- The validity in a modal logic is typically decidable. It is very hard to find a modal logic, where validity is undecidable.
- The translation to $\text{FO}^2$ provides a partial explanation why modal logic is decidable.