An Introduction to Modal Logic IX

Decidability

Marco Cerami

Palacký University in Olomouc
Department of Computer Science
Olomouc, Czech Republic

Olomouc, November 28^{th} 2013

european social fund in the czech republic
EUROPEAN UNION
MINISTRY OF EDUCATION, YOUTH AND SPORTS
OP Education for Competitiveness

INVESTMENTS IN EDUCATION DEVELOPMENT

Introduction

- **Finite model property** is a fundamental property;
- it is strictly related to **decidability**;
- indeed, it implies decidability of several normal modal logics;
- nevertheless, having finite models and decidability **does not mean to have a good computational behavior**;
- **models**, even though they are finite, **can be inherently huge**;
- later on we will address this fact and prove that, in the worst case, the model of a formula **may not be polynomial** on the size of the formula.

Decidability

Syntax vs semantics

- Since a normal modal logic can be defined either semantically of syntactically, also the **reasoning problems** on this logic can be defined either way:
  - **syntactical definition:** a formula can be consistent or a theorem;
  - **semantical definition:** a formula can be satisfiable or valid.
- As we have seen with the weak completeness theorems, the notions of **valid formula** and **theorem** are someway **equivalent** in the sense that a formula is valid if and only if it is a theorem;
- so, from the point of view of computability, **it is the same** deciding whether a formula φ belongs either to the set of valid formulas or to the set of theorems.

Satisfiability vs validity

- A formula φ is **satisfiable** if and only if its negation $\neg\varphi$ is not valid;
- since the negation of a formula is a **recursive reduction**, then we can decide satisfiability **if and only if** we can decide validity.
- In the same way, the same transformation and statement hold for the pair **consistent formulas vs theorems**.
- So, as long as we are concerned with decidability, **it does not matter** whether we are interested either in the validity/theorem problem or in the satisfiability/consistency problem.
- Nevertheless, there can be a difference in the **design of effective algorithm** for the different formulations of the problem.

Proving decidability of the validity problem

- We recall that in order to prove that a set X on a language $\mathcal{L}$ is **recursive**, it is enough to prove that **both X and its complement are recursively enumerable**.
- Hence, if we want to prove that **the set of the theorems** of a normal modal logic Λ is decidable, we have to prove that Λ and its complement are recursively enumerable.
- The first fact is a consequence of the **finite axiomatizability** of normal modal logics.
- The second fact is a consequence of the **finite model property** of normal modal logics.

Recursively enumerable sets

- A set of words X on a language $\mathcal{L}$ is said to be **recursively enumerable** if there is an algorithm that, for every input x, decides whether $x \in X$;
- **equivalently** a set of words X on a language $\mathcal{L}$ is said to be recursively enumerable if there is an algorithm that **effectively enumerates** every $x \in X$;
- recursive enumerability **does not mean decidability**;
- indeed, a recursively enumerable set X may not be decidable if **its complement** is not recursively enumerable too;
- for example, the set of theorems of first order predicate logic is **recursively enumerable but not decidable**;

Recursive enumeration of theorems

- It is a known fact that if a logic is **recursively axiomatizable** and the **language is countable**, then the set of its theorems is **recursively enumerable**.
- We can indeed obtain the set of theorems by **applying the deductive rules** to axioms and previously obtained theorems.
- We can use the enumeration of axioms, logical connectives and propositional variables and the length of the proof to **effectively enumerate** the theorems.
- The fact that deductions are finite object, implies also that we have a **finite procedure** to obtain each theorem.
- This, clearly **does not mean** that we have a procedure to obtain all the formulas that are not theorems.

Recursive enumeration of non-valid formulas

- Consider a **recursively enumerable set M of finite models**.
- Consider an **enumeration of formulas** (we are considering a countable language).
- We can then design a procedure to **recursively enumerate all the non-valid formulas**,
- at every step:
  1. store the model and formula that are next in the enumeration,
  2. check every formula in every model already stored (models are finite),
  3. output the formulas that are not true in some model.
- This procedure gives a **recursive enumeration** of non-valid formulas.

Conditions for decidability

- So, we have proved that the set of theorems of a logic is **decidable**, under the condition that:

  1. the set of **propositional variables** is countable,
  2. the set of **logical symbols** is countable,
  3. the set of **axioms** is recursively enumerable,
  4. the set of **models** is recursively enumerable,
  5. models are **finite**.

- All this condition are **often met** in normal modal logics.

Strengthening the conditions

- Nevertheless, for at least **all the logics we have seen until now**, we can strengthen those conditions:

  1. the set of propositional variables is **still countable**,
  2. the set of logical symbols is **still countable**,
  3. the set of axioms is **finite**,
  4. models are **still finite**.

- Since **a finite set of axioms is recursively axiomatizable**, the proof that the set of theorems is r.e. is the same as before.

- To prove that the set of non-valid formulas is r.e. the proof is also the same, but the detail that **we do not need** a r.e. set of models, since the fact that the set of axioms is finite, makes **the set of models decidable**.

Finite axioms vs r.e. models

- Note that, in order to prove decidability we have required either:

  1. that the set of models is **recursively enumerable**,
  2. or that the set of axioms is **finite**,

- Clearly the **second condition is stronger** than the first.
- Nevertheless, both condition make us sure that every considered model **M is a model of a logic**.
- Indeed, it is not enough that a model is finite to be sure that it is the model of a given logic, if to prove it, an **infinite set of axioms has to be checked**!!.

The strong finite model property

Finite models of a given size

- As we have seen, the Finite Model Property is a **powerful property** when proving decidability
- Nevertheless it does not give any hint on how big are models.
- In many normal modal logic it is indeed possible to someway **take under control the size of models**.
- As we will see, however **there are limitations** on the control that we can have.

The Strong Finite Model Property

Let

- Λ be a normal modal logic complete with respect to
- a set **M** of finite models
- and $f : \mathbb{N} \longrightarrow \mathbb{N}$ a function on natural numbers.

Then

- we say that Λ has the $f(n)$-**size model property** if every Λ-consistent formula φ is satisfiable in a model $\mathfrak{M} \in$ **M whose domain has cardinality at most** $f(|\varphi|)$;
- we say that Λ has the **strong finite model property** if it has the $f(n)$-size model property for a **computable function** f;

Strong Finite Model Property and Decidability

The strong finite model property provides a simpler and **just semantical** way of proving decidability of a normal modal logic Λ.

- Let φ be a modal formula.
- Now the **only condition** required is that the set $\mathbf{M}$ of models be **recursive**.
- By s.f.m.p. we can generate all the models $\mathfrak{M}$ of size at most $f(|\varphi|)$ (**there is a finite number of them**).
- Since $\mathbf{M}$ is recursive we can check whether $\mathfrak{M} \in \mathbf{M}$.
- For every model $\mathfrak{M} \in \mathbf{M}$ of size at most $f(|\varphi|)$ we can check whether it satisfies φ.
- If φ is not satisfiable in any of them, then it is **unsatisfiable** and we are done.

Strong Finite Model Property and Filtrations

- But, **how can we prove** that a normal modal logic Λ has the strong finite model property.
- In general it is not known, but in some case we can use the method of **filtrations**.
- Let φ be a modal formula and $\mathfrak{M}$ a model of Λ.
- we have already proved that are finite and that there is at least one filtration $\mathfrak{M}_{Sub(\varphi)}$ that is a model of Λ.
- By definition, $W_{Sub(\varphi)}$ has at most $2^{|Sub(\varphi)|}$ nodes.
- The only condition needed is that $\mathfrak{M}_{Sub(\varphi)}$ is a model of Λ, but it is decidable in those cases when some particular property (e.g. transitivity) has to be checked.

The polysize model property

Finite models of a small size

- As we have seen, the Strong Finite Model Property is a **powerful property** when taking under control the size of models.
- Nevertheless, models obtained by means of filtrations are **still too big**, indeed, they are exponential on the size of a given formula.
- If we could obtain models that have polynomial size on the size of a given formula it would be a great improvement.
- Unfortunately, this is **in general not possible**.
- As we will now prove, the models of the minimal modal logic K **can be larger** than polynomial on the size of certain formulas.

K and the Polysize Model Property

- We say that a normal modal logic Λ has the **polysize model property** if it has the $f(n)$-size model property for a **polynomial function** f.
- The minimal modal logic K **has not the polysize model property**.
- The proof is **through a counter-example**.

A conter-example

For every $n \in \mathbb{N}$, consider the sets of propositional variables

$$p_0, \dots, p_n \quad \text{and} \quad q_0, \dots, q_n$$

For every i with $0 \leq i \leq n-1$ define the following formulas

$$B_i \;:=\; q_i \to (\Diamond(q_{i+1} \wedge p_{i+1}) \wedge \Diamond(q_{i+1} \wedge \neg p_{i+1}))$$

and

$$S_i \;:=\; (p_i \to \Box p_i) \wedge (\neg p_i \to \neg \Box p_i)$$

and the following abbreviations:

$$\Box^i \varphi := \overbrace{\Box \dots \Box}^{i \text{ times}} \varphi \quad \text{and} \quad \Box^{(m)} \varphi := \varphi \wedge \Box \varphi \wedge \Box^2 \varphi \wedge \dots \wedge \Box^m \varphi$$

Now, for every $n \in \mathbb{N}$, consider the formula $\varphi^{\mathcal{B}}(n)$ as the conjunction of the following formulas:

$$
\begin{array}{lllllll}
\text{(i)} & q_0 & & & & & \\
\text{(ii)} & \Box^{(n)}(q_i \rightarrow (\bigwedge_{i \neq j} \neg q_j)) & & & (0 \leq i \leq n) & & \\
\text{(iii)} & B_0 & \wedge \Box B_1 & \wedge \Box^2 B_2 & \wedge \Box^3 B_3 & \wedge \ldots \wedge \Box^{n-1} B_{n-1} & \\
\text{(iv)} & & \Box S_1 & \wedge \Box^2 S_1 & \wedge \Box^3 S_1 & \wedge \ldots \wedge \Box^{n-1} S_1 & \\
 & & & \wedge \Box^2 S_2 & \wedge \Box^3 S_2 & \wedge \ldots \wedge \Box^{n-1} S_2 & \\
 & & & & \wedge \Box^3 S_3 & \wedge \ldots \wedge \Box^{n-1} S_3 & \\
 & & & & & \vdots & \\
 & & & & & \wedge \Box^{n-1} S_{n-1} &
\end{array}
$$

$\varphi^{\mathcal{B}}(n)$ is a small formula

On the one hand, $\varphi^{\mathcal{B}}(n)$ is a **small formula**. . . in the sense that its growth is few more larger than quadratical. Let us see what happen when we increase from n to $n+1$:

1. item (i) does not change,
2. item (ii) gains one conjunct of the form
$$\Box^{n+1}(q_i \rightarrow (\wedge_{i \neq j} \neg q_j)),\ (0 \leq i \leq n+1)$$
and each conjunct gains an extra variable,
3. item (iii) gains one conjunct of the form
$$q_n \rightarrow (\Diamond(q_{n+1} \wedge p_{n+1}) \wedge \Diamond(q_{n+1} \wedge \neg p_{n+1})),$$
4. each row in item (iv) gains one conjunct of the form
$$(p_n \rightarrow \Box p_n) \wedge (\neg p_n \rightarrow \neg \Box p_n).$$

This means that

- by 3 and 4 we have to add a whole column passing from $\frac{(n-1)^2}{2}$ to $\frac{n^2}{2}$, that is, the growth in $\mathcal{O}(n^2)$,
- by 2, we have to add $n+1$ times a new variable, $n+1$ new formulas and $n+1$ new boxes and this at most multiplies the growth by n.

In total, the growth of $\varphi^{\mathcal{B}}(n)$ is in $\mathcal{O}(n^3)$

Every model of $\varphi^{\mathcal{B}}(n)$ contains a binary tree of depth n

On the other hand, we will see that each model of $\varphi^{\mathcal{B}}(n)$ contains a binary tree of depth n, that is, they are exponential in n:

1. item (i) forces for a root node,
2. item (ii) forces that in in every node only one among $q_0, \ldots, q_n$ is true,
3. item (iii) forces that, at level i, every node has at least two successors, one with p_i and the other with $\neg p_i$,
4. item (iv) forces the propagation of either p_i or $\neg p_i$ to every path that starts from a point where either p_i or $\neg p_i$ are true.

In conclusion, every model of $\varphi^{\mathcal{B}}(n)$ contains as much final leaves as subsets of $\{p_0, \ldots, p_n\}$.