Axiomatizing truth in a finite model

Gilles Dowek, Ying Jiang

Axiomatizing truth in a finite model

Gilles Dowek[1] and Ying Jiang[2]

[1] Inria
23 avenue d'Italie, CS 81321, 75214 Paris Cedex 13, France
gilles.dowek@inria.fr
[2] State Key Laboratory of Computer Science
Institute of Software
Chinese Academy of Sciences
P.O. Box 8718, 100190 Beijing, China
jy@ios.ac.cn

Abstract. Given a finite model, we build an axiomatic theory such that the propositions provable in this theory are those valid in the model. We sketch applications to automated theorem proving.

1 Introduction

This work is a contribution to the project to bring model-checking and proof theory together, by defining an axiomatic theory such that the propositions provable in this theory are exactly those valid in a given model. A focus is put on designing a theory that is adapted to automated theorem proving.

1.1 Truth and the universe of discourse

When a proposition, such as $(\forall x\ (P(x) \wedge Q(x))) \Rightarrow \forall y\ P(y)$, is provable in predicate logic, it is true independently of the universe of discourse, that is independently of the meaning of the symbols that occur in it and of the variation domain of its variables. In some cases, however, we want to define another notion of truth, for a specific universe of discourse.

For instance, assume we have three objects s_1, s_2, and s_3, a binary relation R relating s_1 to s_2, s_2 to s_1 and s_2 to s_3, but no other elements and a unary predicate P verified by s_1 and s_2, but not by s_3, then the proposition

$$\exists x \exists y\ (R(x,y) \wedge P(x) \wedge \neg P(y))$$

is true in this specific universe of discourse, but may be false in others.

To define a notion of truth in a universe of discourse containing a finite number of objects, a common approach is to formalize this universe as a finite model and to define truth as validity in this model. For instance, the universe described above, can be formalized as the model whose domain is the set $\{s_1, s_2, s_3\}$ and where the symbols R and P are interpreted by the sets $\{\langle s_1, s_2\rangle, \langle s_2, s_1\rangle, \langle s_2, s_3\rangle\}$ and $\{s_1, s_2\}$ respectively and the proposition above is valid in this model.

Because the model is finite, this notion of truth is effective: propositions can be model-checked.

1.2 Proofs

When truth is defined as validity in a model, if the proposition

$$\exists x \exists y \ (R(x, y) \wedge P(x) \wedge \neg P(y))$$

has been model-checked and needs to be model-checked again, all the elements of the domain must be tried for x again. A proof is a way to record that the first time the proposition was model-checked, the variable x was interpreted by s_2, in order to avoid enumerating the elements of the domain the second time. If a proof is built the first time the proposition is model-checked, checking this proof costs less than model-checking the proposition again. This has led to definitions of an equivalent notion of truth based on the existence of a proof, rather than on the validity in a model. See, for instance, [3–7].

Defining such a notion of proof also permits to use proof-search algorithms instead of model-checking ones, and, in some cases, proof-search may be more efficient than model-checking. For instance, one can express the fact that the predicate P is verified by s_1 and s_2 but not by s_3 with the axioms $P(s_1)$, $P(s_2)$, and $\neg P(s_3)$ and attempting to prove the proposition $\exists x \ \neg P(x)$ in this theory with the resolution method yields the clauses $P(s_1)$, $P(s_2)$, $\neg P(s_3)$, $P(x)$. As the resolution rule can only be applied to a negative and a positive literal, the only possibility is to apply it to $P(x)$ and $\neg P(s_3)$, yielding $x = s_3$ without enumerating all the possible values for x. Of course, the generality of this example remains to be investigated.

Several works have focused on the definition of deduction rules for specific logics, such as LTL, CTL, CTL*, etc. Our goal, in this paper is slightly different as we try to axiomatize these logics in plain predicate logic. So our focus will be more on axioms than on deduction rules. Also we do not focus on a specific logic, but rather on the axiomatization, for a given model, of the general theory of classes, also known as *(monadic) second order logic*, where several of these logics can be translated.

Thus, we to attempt to define a theory in predicate logic such that the propositions provable in this theory are exactly those that are valid in some model. As a consequence of Gödel's incompleteness theorem, there is no such theory for the standard model of arithmetic, for instance. But there is always one, when the model is finite.

1.3 Classes

Let $\mathcal{M}$ be a finite model. To axiomatize validity in this model, a possibility is to introduce constants $s_1, ..., s_n$ for the elements of the domain of $\mathcal{M}$ and for each predicate symbol Q of arity k and each k-tuple of constants $c_1, ..., c_k$ either the axiom $Q(c_1, ..., c_k)$ if $Q(c_1, ..., c_k)$ is valid in $\mathcal{M}$ or the axiom $\neg Q(c_1, ..., c_k)$ if it is not.

But, we also need to express that there is no other elements than those expressed by the constants $s_1, ..., s_n$. This can be made, by introducing an axiom

scheme E

$$((s_1/x)A \wedge ... \wedge (s_n/x)A) \Rightarrow \forall x\ A$$

This theory is sound and complete with respect to validity in the model $\mathcal{M}$: a simple induction over the structure of A permits to prove that if a closed proposition A is valid in the model $\mathcal{M}$ then it is provable in this theory and a simple induction over proof structure permits to prove the converse.

But the language of predicate logic, with the usual interpretation that the variables vary over the objects of the universe has a limited expressivity. For instance, given a binary relation R and two elements s_1 and s_2 of the domain, it cannot express the existence of a path from s_1 to s_2 in the interpretation of R. But, such a property can be easily expressed if we introduce, besides a sort for objects, another sort for classes of objects and a membership predicate

$$\forall Y\ (s_1 \in Y \Rightarrow (\forall x \forall x'\ (x \in Y \Rightarrow R(x, x') \Rightarrow x' \in Y)) \Rightarrow s_2 \in Y)$$

We prefer to use the word *class*, rather than the *set*, to emphasis that the two-sorted structure of the language prohibits classes to be elements of other classes. We prefer to use the name *class theory*, rather than *(monadic) second order logic*, to emphasis the fact that this formalism is a theory and not a logic.

In the same way, the fact that all paths starting from s_1 eventually reach a point where P holds is expressed by the proposition, written $AF(P)(s_1)$ in CTL

$$\forall Y\ (\forall x\ ((P(x) \vee \forall x'\ (R(x, x') \Rightarrow x' \in Y)) \Rightarrow x \in Y) \Rightarrow s_1 \in Y)$$

Co-inductive properties can also be expressed this way. For instance, the existence of an infinite path starting at state s_1 and such that all the elements of the path verify a property P can be expressed by the proposition, written $EG(P)(s_1)$ in CTL

$$\exists Y\ (s_1 \in Y \wedge \forall x\ (x \in Y \Rightarrow (P(x) \wedge \exists x'\ (R(x, x') \wedge x' \in Y))))$$

As we have introduced class variables, we need to assert the existence of some classes. A usual way is to introduce a *comprehension scheme*

$$\exists Y\ \forall x\ (x \in Y \Leftrightarrow A)$$

But here, as the domain of the model is finite, all the classes are finite as well and we can define them *in extension*, that is by the list of their elements. Thus, we introduce a constant $\varnothing$ for the empty class and a binary function symbol *add* for the operation of adding an element to a class and the axioms

$$\forall x\ \neg x \in \varnothing$$

$$\forall x \forall y \forall Z\ (x \in add(y, Z) \Leftrightarrow (x = y \vee x \in Z))$$

We must also express with axioms the meaning of the introduced equality symbol

$$s_1 = s_1$$

$$\neg s_1 = s_2$$

etc. Like for objects, an axiom scheme should also be added, to express that there are no other classes than those built this way

$$((T_1/Y)A \wedge ... \wedge (T_{2^n}/Y)A) \Rightarrow \forall Y\ A$$

where T_1, ..., T_{2^n} are 2^n terms expressing the 2^n subsets of the domain of $\mathcal{M}$. But using this axiom scheme would lead to 2^n premises and should definitively be avoided.

The first contribution of this paper is to show that we keep a complete axiomatization of finite models, even if we do not include such an axiom scheme for classes. The second it to show that, in some cases, we can also drop the axiom scheme E for objects. We conclude by sketching possible applications to automated theorem proving.

2 Axiomatizing finite models

We consider a two sorted language $\mathcal{L}$ containing

- constants $s_1, ..., s_n$,
- predicate symbols,
- a binary predicate symbol $\in$,
- a constant $\varnothing$,
- a binary function symbol *add*.

We consider a standard model $\mathcal{M}$ of this language: the object domain has n elements $s_1, ..., s_n$, the class domain contains the 2^n subsets of the object domain, the constants denotes themselves, and the symbols $\in$, $\varnothing$, and *add* are interpreted in the standard way.

Definition 1 (The theory $\mathcal{T}$). *The theory $\mathcal{T}$ contains*

- *the axiom scheme E: for each proposition A, the axiom*

$$((s_1/x)A \wedge ... \wedge (s_n/x)A) \Rightarrow \forall x\ A$$

- *for each predicate symbol Q of arity k and each k-tuple of constants $c_1, ..., c_k$ the axiom*

$$Q(c_1, ..., c_k)$$

  if the sequence $\langle c_1, ..., c_k \rangle$ is in the interpretation of Q in $\mathcal{M}$ and the axiom

$$\neg Q(c_1, ..., c_k)$$

  otherwise.
- *the axioms*

$$\forall x\ \neg x \in \varnothing$$

$$\forall x \forall y \forall Z\ (x \in add(y, Z) \Leftrightarrow (x = y \vee x \in Z))$$

– *the axioms*

$$s_1 = s_1$$

$$\neg s_1 = s_2$$

etc.

We call $\mathcal{T}^-$ the theory $\mathcal{T}$ minus the scheme E.

$$\frac{}{A \vdash A}\ \text{axiom}$$

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}\ \text{contr-l} \qquad \frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}\ \text{contr-r}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}\ \text{weak-l} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta}\ \text{weak-r}$$

$$\frac{}{\Gamma \vdash \top, \Delta}\ \top\text{-r}$$

$$\frac{}{\Gamma, \bot \vdash \Delta}\ \bot\text{-l}$$

$$\frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}\ \neg\text{-l} \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta}\ \neg\text{-r}$$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}\ \wedge\text{-l} \qquad \frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta}\ \wedge\text{-r}$$

$$\frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}\ \wedge\text{-l}$$

$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}\ \vee\text{-l} \qquad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta}\ \vee\text{-r}$$

$$\frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \vee B, \Delta}\ \vee\text{-r}$$

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta}\ \Rightarrow\text{-l} \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \Rightarrow B, \Delta}\ \Rightarrow\text{-r}$$

$$\frac{\Gamma, (t/x)A \vdash \Delta}{\Gamma, \forall x\ A \vdash \Delta}\ \forall\text{-l} \qquad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash \forall x\ A, \Delta}\ \forall\text{-r (*)}$$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, \exists x\ A \vdash \Delta}\ \exists\text{-l (*)} \qquad \frac{\Gamma \vdash (t/x)A, \Delta}{\Gamma \vdash \exists x\ A, \Delta}\ \exists\text{-r}$$

$$\frac{\Gamma, (T/Y)A \vdash \Delta}{\Gamma, \forall Y\ A \vdash \Delta}\ \text{c-}\forall\text{-l} \qquad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash \forall Y\ A, \Delta}\ \text{c-}\forall\text{-r (**)}$$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, \exists Y\ A \vdash \Delta}\ \text{c-}\exists\text{-l (**)} \qquad \frac{\Gamma \vdash (T/Y)A, \Delta}{\Gamma \vdash \exists Y\ A, \Delta}\ \text{c-}\exists\text{-r}$$

(*) x not free in $\Gamma\Delta$, (**) Y not free in $\Gamma\Delta$

Fig. 1. Sequent calculus

The notion of a classical proof can be defined, for instance, with the sequent calculus of Figure 1. We shall use the following result, which is a consequence of the admissibility of the cut rule in sequent calculus.

Proposition 1. *If a proposition B is provable both under the hypotheses Γ, A and $\Gamma, \neg A$, then it is provable under the hypotheses Γ.*

3 Soundness and completeness

Theorem 1 (Soundness). *If a proposition A is provable in $\mathcal{T}$ then it is valid in $\mathcal{M}$.*

Proof. All axioms of $\mathcal{T}$ are valid in $\mathcal{M}$, we conclude with a simple induction on proof structure.

We now want to prove the converse. The proof will be by induction over the structure of A and the main difficulty is for propositions of the form $\forall Y\ B$. In this case from $[\![\forall Y\ B]\!]_\phi = 1$, we can deduce that, for all classes C, $[\![B]\!]_{\phi,Y=C} = 1$, then with the substitution lemma, for all terms T, $[\![(T/Y)B]\!]_\phi = 1$ and by induction hypothesis that, for all terms T, the proposition $(T/Y)B$ is provable. But we cannot conclude directly that the proposition $\forall Y\ B$ is provable, because we do not have an axiom

$$((T_1/Y)B \wedge ... \wedge (T_{2^n}/Y)B) \Rightarrow \forall Y\ B$$

Thus, we shall prove that when the proposition $(T/Y)B$ is provable, the proposition B is provable under hypotheses of the form $s_i \in Y$ or $\neg s_j \in Y$ completely defining the class expressed by the term T. Then, we shall eliminate these hypotheses using Proposition 1 and obtain a proof of B, and finally a proof of $\forall Y\ B$.

Proposition 2 (Substitution lemma).

$$[\![(t/x)u]\!]_\phi = [\![u]\!]_{\phi,x=[\![t]\!]_\phi}$$

$$[\![(t/x)A]\!]_\phi = [\![A]\!]_{\phi,x=[\![t]\!]_\phi}$$

Proof. By induction over the structure of u and over the structure of A.

Consider a subset $\{i_1, ..., i_p\}$ of the set $\{1, ..., n\}$, its complement $\{j_1, ..., j_{n-p}\}$, and the term

$$T = add(s_{i_1}, add(s_{i_2}, ...add(s_{i_p}, \varnothing)))$$

Proposition 3. *The propositions $s_{i_1} \in T$, ..., $s_{i_p} \in T$, $\neg s_{j_1} \in T$, ..., $\neg s_{j_{n-p}} \in T$ are provable in the theory $\mathcal{T}^-$.*

Proof. The proposition $s \in T$ is equivalent to

$$s = s_{i_1} \vee s = s_{i_2} \vee ... \vee s = s_{i_p}$$

If s is one of the s_{i_k}, then one of the disjuncts is $\top$ and the proposition is provable. If s is one of the s_{j_k}, then all the disjuncts are $\bot$ and the negation of the proposition is provable.

Proposition 4. *Let s be a constant. The proposition $s \in T \Leftrightarrow s \in Y$ is provable in the theory $\mathcal{T}^-$, $s_{i_1} \in Y$, ..., $s_{i_p} \in Y$, $\neg s_{j_1} \in Y$, ..., $\neg s_{j_{n-p}} \in Y$.*

Let x be a variable. The proposition $x \in T \Leftrightarrow x \in Y$ is provable in the theory $\mathcal{T}$, $s_{i_1} \in Y$, ..., $s_{i_p} \in Y$, $\neg s_{j_1} \in Y$, ..., $\neg s_{j_{n-p}} \in Y$.

Proof. Let s be a constant. If s is one of the s_{i_k} then the proposition $s \in Y \Leftrightarrow s \in T$ is provable because, using Proposition 3 and the hypotheses, both sides are provable. If s is one of the s_{j_k} then the proposition $s \in Y \Leftrightarrow s \in T$ is provable because, using Proposition 3 and the hypotheses, the negations of both sides are provable.

Thus, using the axiom scheme E, the proposition

$$\forall x\ (x \in T \Leftrightarrow x \in Y)$$

is provable. Hence the proposition

$$x \in T \Leftrightarrow x \in Y$$

is provable.

Proposition 5. *The proposition $(T/Y)A \Leftrightarrow A$ is provable in the theory $\mathcal{T}$, $s_{i_1} \in Y$, ..., $s_{i_p} \in Y$, $\neg s_{j_1} \in Y$, ..., $\neg s_{j_{n-p}} \in Y$.*

Proof. By induction over the structure of A.

- If A is atomic and contains the variable Y, then it has the form $u \in Y$ where u is either a constant or a variable and the proposition $(T/Y)A \Leftrightarrow A$ is $u \in T \Leftrightarrow u \in Y$. Using Proposition 4, this proposition is provable in $\mathcal{T}$, $s_{i_1} \in Y$, ..., $s_{i_p} \in Y$, $\neg s_{j_1} \in Y$, ..., $\neg s_{j_{n-p}} \in Y$.
- If A is atomic and does not contain the variable Y, then the proposition $(T/Y)A \Leftrightarrow A$ is $A \Leftrightarrow A$ and it is obviously provable.
- If A is $\top$ or $\bot$, the proof is similar.
- If $A = B \wedge C$, then, by induction hypothesis, the propositions $(T/Y)B \Leftrightarrow B$ and $(T/Y)C \Leftrightarrow C$ are provable. Thus the proposition $(T/Y)(B \wedge C) \Leftrightarrow (B \wedge C)$ is provable.
- If $A = B \vee C$, $A = B \Rightarrow C$, or $A = \neg B$, the proof is similar.
- If $A = \forall x\ B$, then, by induction hypothesis, the proposition $(T/Y)B \Leftrightarrow B$ is provable. Thus the proposition $\forall x\ ((T/Y)B \Leftrightarrow B)$ is provable. Hence the proposition $(\forall x\ (T/Y)B) \Leftrightarrow (\forall x\ B)$, that is the proposition $(T/Y)(\forall x\ B) \Leftrightarrow (\forall x\ B)$ is provable.
- If $A = \forall Z\ B$, the proof is similar.
- If $A = \exists x\ B$, then, by induction hypothesis, the proposition $(T/Y)B \Leftrightarrow B$ is provable. Thus the proposition $\forall x\ ((T/Y)B \Leftrightarrow B)$ is provable. Hence the proposition, $(\exists x\ (T/Y)B) \Leftrightarrow (\exists x\ B)$, that is the proposition $(T/Y)(\exists x\ B) \Leftrightarrow (\exists x\ B)$ is provable.
- If $A = \exists Z\ B$, the proof is similar.

Theorem 2 (Completeness). *Let A be a closed proposition. If the proposition A is valid in $\mathcal{M}$, then it is provable in $\mathcal{T}$.*

Proof. We prove, more generally, that if A is valid in $\mathcal{M}$, then it is provable in $\mathcal{T}$ and if A is not valid in $\mathcal{M}$, then $\neg A$ is provable in $\mathcal{T}$.

By induction over the structure of A.

– if A is $\top$ then it is valid and provable.
– if A is $\bot$ then it is not valid and $\neg\bot$ is provable.
– if $A = \neg B$ is valid then B is not valid and by induction hypothesis $\neg B$ is provable.
  If $A = \neg B$ is not valid then B is valid and by induction hypothesis B is provable, thus $\neg\neg B$ is provable.
– If $A = B \wedge C$ is valid then B and C are both valid and by induction hypothesis B and C are provable, thus $B \wedge C$ is provable.
  If $A = B \wedge C$ is not valid then either B or C is not valid and by induction hypothesis $\neg B$ or $\neg C$ is provable, thus $\neg(B \wedge C)$ is provable.
– If $A = B \vee C$ or $A = B \Rightarrow C$ the proof is similar.
– If $A = \forall x\ B$ is valid then for all i, $[\![B]\!]_{x=s_i} = 1$ thus, using the substitution lemma, $[\![(s_i/x)B]\!] = 1$ that is $(s_i/x)B$ is valid. By the induction hypothesis $(s_i/x)B$ is provable. Thus, using the axiom scheme E, $\forall x\ B$ is provable.
  If $A = \forall x\ B$ is not valid, there exists an i such that $[\![B]\!]_{x=s_i} = 0$ thus, using the substitution lemma, $[\![(s_i/x)B]\!] = 0$ that is $(s_i/x)B$ is not valid. By the induction hypothesis $\neg(s_i/x)B$ is provable. Thus, $\neg\forall x\ B$ is provable.
– If $A = \exists x\ B$ is valid, there exists an i such that $[\![B]\!]_{x=s_i} = 1$ thus, using the substitution lemma, $[\![(s_i/x)B]\!] = 1$ that is $(s_i/x)B$ is valid. By the induction hypothesis $(s_i/x)B$ is provable. Thus $\exists x\ B$ is provable.
  If $A = \exists x\ B$ is not valid then for all i, $[\![B]\!]_{x=s_i} = 0$ thus, using the substitution lemma, $[\![(s_i/x)B]\!] = 0$ that is $(s_i/x)B$ is not valid. By the induction hypothesis $\neg(s_i/x)B$ is provable. Thus, using the axiom scheme E, $\neg\exists x\ B$ is provable.
– If $A = \forall Y\ B$ is valid then for all subsets $\{i_1, ..., i_p\}$ of $\{1, ..., n\}$, $[\![B]\!]_{Y=\{s_{i_1},...,s_{i_p}\}} = 1$ thus, using the substitution lemma, the proposition

$$(add(s_{i_1}, add(s_{i_2}, ..., add(s_{i_p}, \varnothing)))/Y)B$$

  is valid. Using the induction hypothesis, it is provable. Thus using Proposition 5, the proposition B is provable under the hypotheses $s_{i_1} \in Y$, ..., $s_{i_p} \in Y$, $\neg s_{j_1} \in Y$, ..., $\neg s_{j_{n-p}} \in Y$. Using $2^n - 1$ times Proposition 1, the proposition B is provable. Hence, the proposition $\forall Y\ B$ is provable.
  If $A = \forall Y\ B$ is not valid then there exists a subset $\{i_1, ..., i_p\}$ of $\{1, ..., n\}$, such that $[\![B]\!]_{Y=\{s_{i_1},...,s_{i_p}\}} = 0$ thus, using the substitution lemma, the proposition

$$(add(s_{i_1}, add(s_{i_2}, ..., add(s_{i_p}, \varnothing)))/Y)B$$

  is not valid. Thus, using the induction hypothesis, the proposition

$$\neg(add(s_{i_1}, add(s_{i_2}, ..., add(s_{i_p}, \varnothing)))/Y)B$$

  is provable. Thus, the proposition $\neg\forall Y\ B$ is provable.
– If $A = \exists Y\ B$ is valid then there exists a subset $\{i_1, ..., i_p\}$ of $\{1, ..., n\}$, such that $[\![B]\!]_{Y=\{s_{i_1},...,s_{i_p}\}} = 1$ thus, using the substitution lemma, the proposition

$$(add(s_{i_1}, add(s_{i_2}, ..., add(s_{i_p}, \varnothing)))/Y)B$$

is valid. Thus, using the induction hypothesis, it is provable. Thus, the proposition $\exists Y\ B$ is provable.
If $A = \exists Y\ B$ is not valid then for all subsets $\{i_1, ..., i_p\}$ of $\{1, ..., n\}$, $[\![B]\!]_{Y=\{s_{i_1},...,s_{i_p}\}} = 0$ thus, using the substitution lemma, the proposition

$$(add(s_{i_1}, add(s_{i_2}, ..., add(s_{i_p}, \varnothing)))/Y)B$$

is not valid. Using the induction hypothesis, the proposition

$$\neg(add(s_{i_1}, add(s_{i_2}, ..., add(s_{i_p}, \varnothing)))/Y)B$$

is provable. Thus using Proposition 5, the proposition $\neg B$ is provable under the hypotheses $s_{i_1} \in Y$, ..., $s_{i_p} \in Y$, $\neg s_{j_1} \in Y$, ..., $\neg s_{j_{n-p}} \in Y$. Using $2^n - 1$ times Proposition 1, the proposition $\neg B$ is provable. Hence, the proposition $\neg\exists Y\ B$ is provable.

4 Removing the axiom scheme *E*

In this section, we generalize Theorem 2, by showing that we can drop the axiom scheme E, if we restrict to propositions containing no object quantifiers. Object quantifiers can always be removed from propositions, preserving validity, by replacing propositions of the form $\forall x\ A$ by $(s_1/x)A \wedge ... \wedge (s_n/x)A$ and propositions of the form $\exists x\ A$ by $(s_1/x)A \vee ... \vee (s_n/x)A$.

We first prove an analog of Proposition 5.

Proposition 6. *Let A to be a proposition containing no object quantifiers and no free object variables. Then proposition $(T/Y)A \Leftrightarrow A$ is provable in the theory $\mathcal{T}^-$, $s_{i_1} \in Y$, ..., $s_{i_p} \in Y$, $\neg s_{j_1} \in Y$, ..., $\neg s_{j_{n-p}} \in Y$.*

Proof. The proof is analog to that of Proposition 5, except that we do not consider the cases corresponding to object quantifiers and we restrict the case of atomic propositions containing the variable Y to the case where u is a constant.

Theorem 3. *Let A be a closed proposition containing no object quantifiers and no free object variables. If the proposition A is valid in $\mathcal{M}$, then it is provable in $\mathcal{T}^-$.*

Proof. The proof is analog to that of Theorem 2, except that we use Proposition 6 instead of Proposition 5 and we drop the cases corresponding to the object quantifiers.

5 Quantifiers as abbreviations

In this section, we extend Theorem 3 to closed propositions containing no positive occurrences of object quantifiers, that is no positive occurrences of the universal quantifier and no negative occurrences of the existential one, but possibly containing negative ones.

The intuition is that the negative occurrences of $\forall x\ A$ and the positive occurrences of $\exists x\ A$ are treated in sequent calculus like the propositions $(s_1/x)A \wedge ... \wedge (s_n/x)A$ and $(s_1/x)A \vee ... \vee (s_n/x)A$: the $\exists$-r rule applied to a proposition $\exists x\ A$ transforms it into a proposition $(s/x)A$, like the $\vee$-r rule transforms, in several steps, the proposition $(s_1/x)A \vee ... \vee (s_n/x)A$ into a proposition $(s/x)A$.

Proposition 7. *Let $\Gamma \vdash \Delta$ be a sequent containing no positive occurrences of object quantifiers and $\Gamma' \vdash \Delta'$ be the sequent obtained by replacing in $\Gamma \vdash \Delta$ all occurrences of $\forall x\ A$ by $(s_1/x)A \wedge ... \wedge (s_n/x)A$ and all occurrences of $\exists x\ A$ by $(s_1/x)A \vee ... \vee (s_n/x)A$. Then, if the sequent $\Gamma' \vdash \Delta'$ has a proof, so does the sequent $\Gamma \vdash \Delta$.*

Proof. Consider a proof of the sequent $\Gamma' \vdash \Delta'$. Without loss of generality, we can assume that all axioms are on atomic propositions. In this proof, we replace the propositions of the form $(s_1/x)A \wedge ... \wedge (s_n/x)A$ introduced by the translation and all their non atomic descendants by $\forall x\ A$ and the propositions of the form $(s_1/x)A \vee ... \vee (s_n/x)A$ introduced by the translation and all their non atomic descendants by $\exists x\ A$. Removing the silent steps, corresponding, for instance, to a transformation of $(s_i/x)A \wedge ... \wedge (s_n/x)A$ into $(s_{i+1}/x)A \wedge ... \wedge (s_n/x)A$, both being replaced by $\forall x\ A$, yields a proof of $\Gamma \vdash \Delta$.

Theorem 4. *Let A be a closed proposition with no positive occurrences of object quantifiers. If the proposition A is valid in $\mathcal{M}$, then it is provable in $\mathcal{T}^-$.*

Proof. Consider the proposition A' obtained by replacing all occurrences of $\forall x\ B$ by $(s_1/x)B \wedge ... \wedge (s_n/x)B$ and all occurrences of $\exists x\ B$ by $(s_1/x)B \vee ... \vee (s_n/x)B$. If the proposition A is valid in $\mathcal{M}$, then so is A', by Theorem 3, the sequent $\mathcal{T}^- \vdash A'$ has a proof, and, by Proposition 7, so does the sequent $\mathcal{T}^- \vdash A$.

Remark 1. It would also be possible to keep the positive quantifiers and treat the positive occurrences of $\forall x\ A$ and the negative occurrences of $\exists x\ A$ like the propositions $(s_1/x)A \wedge ... \wedge (s_n/x)A$ and $(s_1/x)A \vee ... \vee (s_n/x)A$. But this would require to replace the $\forall$-r and the $\exists$-l rules by the enumeration rules

$$\frac{\Gamma \vdash (s_1/x)A, \Delta \ ...\ \Gamma \vdash (s_n/x)A, \Delta}{\Gamma \vdash \forall x\ A, \Delta}$$

$$\frac{\Gamma, (s_1/x)A \vdash \Delta \ ...\ \Gamma, (s_n/x)A \vdash \Delta}{\Gamma, \exists x\ A \vdash \Delta}$$

similar to the ω-rule, but finitary.

6 Conclusion and future work

The theory $\mathcal{T}^-$ is well suited for automated theorem proving as all its axioms can be transformed into rewrite rules. The axioms of the form $Q(c_1, ..., c_k)$ are transformed into

$$Q(c_1, ..., c_k) \longrightarrow \top$$

Those of the form $\neg Q(c_1, ..., c_k)$ into

$$Q(c_1, ..., c_k) \longrightarrow \bot$$

The axioms

$$\forall x \ \neg x \in \varnothing$$

$$\forall x \forall y \forall Z \ (x \in add(y, Z) \Leftrightarrow (x = y \vee x \in Z))$$

into

$$x \in \varnothing \longrightarrow \bot$$

$$x \in add(y, Z) \longrightarrow x = y \vee x \in Z$$

The axioms $s_1 = s_1$ $\neg s_1 = s_2$, etc. into

$$s_1 = s_1 \longrightarrow \top$$

$$s_1 = s_2 \longrightarrow \bot$$

etc.

This theory is positive, hence it has the cut elimination property [2] and this allows to use *Polarized resolution modulo* [1].

In order to use Theorem 4, we must replace, in the proposition to be proved, all the positive occurrences of a proposition of the form $\forall x \ A$ by $(s_1/x)A \wedge ... \wedge (s_n/x)A$ and all the negative occurrences of a proposition of the form $\exists x \ A$ by $(s_1/x)A \vee ... \vee (s_n/x)A$. This creates an exponential blow up of the proposition to be proved. But this blow up would occur anyway if we model-checked the proposition in a finite model, as we would have to check that all the elements of the model verify the property A.

In this method the positive occurrences of object universal quantifiers and the negative occurrences of object existential quantifiers lead to enumeration. But the positive occurrences of object existential quantifiers and the negative occurrences of object universal quantifiers do not: they yield variables that are instantiated during the proof search. In the same way, the positive occurrences of class existential quantifiers and the negative occurrences of class universal quantifiers yield variables that are instantiated during the proof search. Because we have kept the eigenvariable rules for the class quantifiers, the positive occurrences of class universal quantifiers and the negative occurrences of class existential quantifier just yield an eigenvariable. Proofs are always generic with respect to such variables, and the method remains complete.

The example below illustrates how an existential class quantifiers in a coinductive definition, yields a variable instantiated step by step during the proof-search.

Example 1. Consider a structure defined by three states s_1, s_2 and s_3, transitions from s_1 to s_2, s_2 to s_1 and s_2 to s_3, and a property P that is verified at s_1 and s_2 but not at s_3.

The clausal form of the theory $\mathcal{T}^-$ is formed with the following clauses: $\underline{P(s_1)}$, $\underline{P(s_2)}$, $\underline{\neg P(s_3)}$, $\underline{R(s_1, s_2)}$, $\underline{R(s_2, s_1)}$, $\underline{R(s_2, s_3)}$, $\underline{\neg R(s_1, s_1)}$, $\underline{\neg R(s_1, s_3)}$, $\underline{\neg R(s_2, s_2)}$,

$\underline{\neg R(s_3, s_1)}$, $\underline{\neg R(s_3, s_2)}$, $\underline{\neg R(s_3, s_3)}$, $\underline{\neg(x \in \varnothing)}$, $\underline{\neg x \in add(y, Z)} \vee x = y \vee x \in Z$, $\underline{x \in add(y, Z)} \vee \neg x = y$, $\underline{x \in add(y, Z)} \vee \neg x \in Z$, $\underline{s_1 = s_1}$, $\underline{s_2 = s_2}$, $\underline{s_3 = s_3}$, $\underline{\neg s_1 = s_2}$, $\underline{\neg s_1 = s_3}$, $\underline{\neg s_2 = s_1}$, $\underline{\neg s_2 = s_3}$, $\underline{\neg s_3 = s_1}$, $\underline{\neg s_3 = s_2}$.

Assume we want to prove the proposition

$$\exists Y \ (s_1 \in Y \wedge \forall x \ (x \in Y \Rightarrow (P(x) \wedge \exists x' \ (R(x, x') \wedge x' \in Y))))$$

expressing the existence of an infinite path starting at s_1 and the that all the elements of the path verify the property P.

In this proposition, we must replace the proposition

$$\forall x \ (x \in Y \Rightarrow (P(x) \wedge \exists x' \ (R(x, x') \wedge x' \in Y)))$$

by a conjunction of three instances for s_1, s_2, and s_3. Then, the clausal form of the negation of this proposition contains eight clauses among which

$\neg s_1 \in Y \vee \neg P(s_1) \vee \neg R(s_1, x_1) \vee \neg x_1 \in Y \vee \neg P(s_2) \vee \neg R(s_2, x_2) \vee \neg x_2 \in Y \vee s_3 \in Y$

Applying the Resolution rule four times with the clauses $\underline{P(s_1)}$, $\underline{P(s_2)}$, $\underline{R(s_1, s_2)}$, and $\underline{R(s_2, s_1)}$ yields $x_1 = s_2$, $x_2 = s_1$ and the clause

$$\neg s_1 \in Y \vee \neg s_2 \in Y \vee s_3 \in Y$$

Thus the problem boils down to proving the existence of a class that contains s_1 and s_2 but not s_3. Applying the Resolution rule with the clause $\underline{\neg(x \in \varnothing)}$, $\underline{\neg x \in add(y, Z)} \vee x = y \vee x \in Z$, $\underline{x \in add(y, Z)} \vee \neg x = y$, $\underline{x \in add(y, Z)} \vee \neg x \in Z$, and the equality clauses will eventually lead to substitute the term $add(s_1, add(s_2, \varnothing))$ for Y and to the empty clause.

Acknowledgements

The authors want to thank Guillaume Burel and Kailiang Ji for enlightening discussions.

References

1. G. Dowek. Polarized resolution modulo. *IFIP Theoretical Computer Science*, 2010.
2. G. Dowek and B. Werner. Proof normalization modulo. *The Journal of Symbolic Logic*, 68(4):1289–1316, 2003.
3. E.A. Emerson, J.Y. Halpern. Decision Procedures and Expressiveness in the Temporal Logic of Branching Time. Journal of Computer and Systerm Sciences, 30(1), 1–24, 1985.
4. M.Fisher, C.Dixon, and M.Peim. Clausal temporal resolution. ACM Transactions on Computational Logic 2(1): 12-56, 2001.
5. O. Friedmann. A Proof System for CTL*mu. Diploma Thesis, University of Munich, 2008.
6. D.M. Gabbay and A. Pnueli. A Sound and Complete Deductive System for CTL* Verification. *Logic Journal of the IGPL* 16(6): 499–536, 2008.
7. M. Reynolds. An axiomatization of full Computational Tree Logic. *The Journal of Symbolic Logic* 66(3): 1011–1057, 2001.