Friendly Logics, Fall 2015, Lecture Notes 2

Val Tannen

1 Reduction Classes

When Church and, independently, Turing gave a negative answer to Hilbert's Entscheidungsproblem (Decision Problem) for FOL a number of related results had already been obtained for certain fragments of FOL. In this section we look at some of the negative results and in the next section at some of the positive results.

Since no precise definition of "decidable" was available there were no negative results per se. Instead, these results provided what we call today many-one reductions from the validity (satisfiability) of arbitrary FO sentences to the validity (satisfiability) of certain syntactically defined subsets of sentences.

Warning! Although Hilbert-Ackermann define the decision problem for both validity and satisfiability, subsequently the term became associated implicitly with just satisfiability. This is not an important distinction for classes of sentences that are closed under negation (because, recall, φ is valid iff $\neg\varphi$ is unsatisfiable). Unfortunately, *most* of the classes of sentences for which positive results are established are *not* closed under negation! To be safe we make the next definition about both satisfiability and validity.

Definition 1.1 *A class C of FO sentences is called a* ***validity (satisfiability) reduction class*** *if there is a computable function f that maps arbitrary FO sentences into C-sentences such that φ is valid (satisfiable) iff $f(\varphi)$ is valid (satisfiable). Therefore VALID $\leq_m C \cap$ VALID (FO-SAT $\leq_m C \cap$ FO-SAT) and it follows from the Church/Turing Theorem that the validity (satisfiability) decision problem for C is also undecidable.*

Definition 1.2 A formula of the form $\varphi \stackrel{\text{def}}{=} Q_1 x_1 \cdots Q_n x_n\ \psi$ where each Q_i is either $\forall$ or $\exists$ and ψ is quantifier-free is called a *prenex* formula. If the quantifiers are all $\forall$ (all $\exists$) then φ is called a *universal* formula (an *existential* formula).

In what follows, by **equivalent** sentences we mean two sentences that are logical consequences of each other. This is denoted $\varphi \models\!\!\!\dashv \psi$ and it holds iff $\models \varphi \Leftrightarrow \psi$.

Lemma 1.1 *For each sentence φ we can compute (in PTIME) an equivalent prenex sentence $Prx(\varphi)$.*

Proof "Pull out" the quantifiers by repeatedly using transformations like

$$\neg\exists x\,\varphi \longmapsto \forall x\,\neg\varphi \qquad \varphi \wedge (\forall x\,\psi) \longmapsto \forall x(\varphi \wedge \psi) \qquad (\forall x\,\varphi) \Rightarrow \psi \longmapsto \exists x(\varphi \Rightarrow \psi)$$

etc., while renaming bound variables to avoid unintended scope capture. □

Exercise 1.1 *Analyze the complexity of the* $Prx()$ *algorithm sketched above. You can choose the data structure used to represent sentences.*

Lemma 1.2 (Skolem) *Let* $\mathcal{V}$ *be a vocabulary and let* $\overline{\mathcal{V}}$ *be its extension with countably many fresh function symbols of each arity (including nullary functions i.e. constants). For each prenex sentence* φ *over* $\mathcal{V}$ *we can compute (in PTIME) a prenex sentence* $Sk(\varphi)$ *over* $\overline{\mathcal{V}}$ *such that*

- $Sk(\varphi)$ *is a universal sentence.*
- φ *is true in the* $\mathcal{V}$*-restriction of any model of* $Sk(\varphi)$*.*
- *Any model of* φ *can be extended, keeping the same universe (domain), to a* $\overline{\mathcal{V}}$*-model that satisfies* $Sk(\varphi)$*. Hence,* φ *is satisfiable iff* $Sk(\varphi)$ *is satisfiable.*

Proof Eliminate the existential quantifiers from left to right in the prenex sentence repeating the transformation

$$\forall x_1 \cdots \forall x_n \exists y\, \varphi(y) \longmapsto \forall x_1 \cdots \forall x_n\, \varphi(f(x_1, \ldots, x_n))$$

where f is a fresh functions symbol.

For example:

$$Sk(\exists u\, \forall x\, \exists v\, \forall y\, \exists w\, R(u,v) \wedge f(x,w) = g(v)) \stackrel{\text{def}}{=} \forall x\, \forall y\, R(r, s(x)) \wedge f(x, t(x,y)) = g(s(x))$$

□

$Sk(\varphi)$ is unique up to some renaming so it is called the *Skolem Normal Form* of φ. The transformation $\varphi \longmapsto Sk(\varphi)$ is called *skolemization*. Symbols like r, s and t in the proof example are called *Skolem functions*.

Exercise 1.2 *Analyze the complexity of the* $Sk()$ *algorithm sketched above. Again, you can choose the data structure used to represent sentences.*

Theorem 1.3 *The universal sentences form a satisfiability reduction class. The existential sentences form a validity reduction class.*

Proof The two reductions are given by $f(\varphi) = Sk(Prx(\varphi))$ and $g(\varphi) = Prx(\neg Sk(Prx(\neg\varphi)))$. □

As you saw, skolemization introduces unboundedly many function symbols. A more "economical" satisfiability reduction class has *no* function symbols but the sentences have quantifier prefix $\forall\exists\forall$ (Kahr 1962). Finally, having just one universal quantifier and just two unary functions also suffices ((Gurevich 1976). See the complete classification in the Börger-Grädel-Gurevich monograph.

2 The Finite Model Property

Although finite validity has bad computational properties for the class of *all* FO sentences, the r.e.-ness of finite satisfiability can be exploited for positive results of the decision problem for certain classes of FO sentences.

Definition 2.1 A class of sentences has the **finite model property** if any satisfiable sentence in the class is also finitely satisfiable.

The class of all FO sentences does not have the finite model property. Indeed, it is fairly easy to concoct sentences that are satisfiable but not finitely satisfiable (see Exercise 3.2) . Such sentences are called *infinity axioms*. In fact, if the class of all FO sentences would have the finite model property then the next result would contradict the undecidability results shown earlier!

Proposition 2.1 *Let C be a class of sentences that is decidable (i.e., it is decidable whether an FO sentence φ is in C). If C has the finite model property then satisfiability of C-sentences is decidable.*

Proof Recall that for the class of all FO sentences satisfiability is co-r.e. and finite satisfiability is r.e. Since C is decidable it enjoys the same properties. But for the sentences in C satisfiability and finite satisfiability coincide! Hence they are both r.e. and co-r.e. and thus decidable. □

The decision procedure provided by the previous proof is very inconvenient: it consists of trying, in parallel, to finitely satisfy the sentence and to prove its negation (if the sentence is satisfiable then the first thread succeeds; if not then the second thread does). A better procedure is given by the following.

Definition 2.2 A class of sentences has the **small model property** if there is a total recursive function u such that any satisfiable sentence in the class has a model with less than $u(|\varphi|)$ elements. (Here $|\varphi|$ denotes the **size** of φ.)

The small model property implies the finite model property. However, a decidable class of sentences with the small model property has a decision procedure for satisfiability that is potentially simpler. There is no need for the annoying attempt to prove the negation of a sentence φ; it suffices to check all models with less than $u(|\varphi|)$ elements.

Note that we only said *potentially* simpler: it all depends on how easy to compute u is. Indeed, if C is decidable then the finite model property implies the small model property! [1] Consider the following algorithm for u:

> On input n, generate the (finitely many) sentences in C of size n. For each of them, in parallel, check for finite satisfiabilty and try to prove the negation, and thus compute either the size of a model or 0 (if unsatisfiable). Return as $u(n)$ the largest of these.

[1] I am grateful to Scott Weinstein for sharing this observation.

Using this u the small model property gives a decision procedure that is just as incovenient as the one given by the finite model property. If we consider classes C that are not decidable a general observation is that there exist such classes that have the finite model property but do not have the small model property: simply take the class of all finitely satisfiable sentences. Indeed, suppose there is a total recursive function u such that any satisfiable sentence φ in this class has a model with less than $u(\mid \varphi \mid)$ elements. But all the sentences in this class are satisfiable! Hence φ is finitely satisfiable iff it has a model with less than $u(\mid \varphi \mid)$ elements. This would make finite satisfiability decidable.

Anyway, what happens for concrete classes of sentences is that a better and more specific u is derived which moreover gives a useful complexity upper bound for the decision procedure. Here is an example:

Theorem 2.2 *The existential FO sentences have the small model property. In fact, any satisfiable existential sentence φ has a model with at most $\mid \varphi \mid$ elements.*

Proof Let $\varphi \stackrel{\text{def}}{=} \exists x_1 \cdots \exists x_m \, \psi$ be an existential sentence where ψ is quantifier-free. We replace each non-variable functional term occurring in ψ with a fresh existentially quantified variable and an additional equality atom. We do this bottom-up for all subterms, including constants, so if $t \equiv f(t_1, \ldots, t_k)$ is such a term and $t_1, \ldots, t_k$ are replaced by $y_1, \ldots, y_k$ then t is replaced by a fresh variable y and we add the equality atom $f(y_1, \ldots, y_k) = y$. Therefore, φ is transformed into a sentence $\overline{\varphi}$ of the form

$$\overline{\varphi} \stackrel{\text{def}}{=} \exists x_1 \cdots \exists x_m \, \exists y_1 \cdots \exists y_n \, \overline{\psi} \wedge f_1(\ldots) = y_1 \wedge \cdots \wedge f_n(\ldots) = y_n$$

where $f_1, \ldots, f_n$ are the functional symbol ocurrences in ψ (we treat constants as nullary functions).

For example, $\exists x \, R(x, f(c, x))$ is transformed into $\exists x \, \exists y_1 \exists y_2 \, R(x, y_2) \wedge c = y_1 \wedge f(y_1, x) = y_2$.

It is not hard to see that φ and $\overline{\varphi}$ hold in the same models and that if $\overline{\varphi}$ is satisfiable then it has a model with at most $m + n$ elements. Since $n \leq \mid \psi \mid$ we conclude that if φ is satisfiable then it has a model with at most $\mid \varphi \mid$ elements. $\Box$

Corollary 2.3 *Satisfiability of existential sentences is decidable and NP-complete.*

Proof Decidability follows the previous theorem and proposition. But the details of the small model theorem give us also the membership in NP: let the sentence be $\varphi \stackrel{\text{def}}{=} \exists x_1 \cdots \exists x_m \, \psi(x_1, \ldots, x_m)$. We guess a model $\mathcal{A}$ with less then $\mid \varphi \mid$ elements, we also guess a valuation v that maps each of $x_1, \ldots, x_m$ to some element of $\mathcal{A}$, and then we check $\mathcal{A}, v \models \psi$. Checking the truth of a quantifier-free formula can be done in PTIME in the size of the formula, (see next set of lecture notes).

For NP-hardness we provide a reduction from boolean satisfiability. Given a boolean formula β with propositional variables $p_1, \ldots, p_n$ we construct the existential sentence $\varphi \stackrel{\text{def}}{=} \exists x_1 \cdots \exists x_n \, \overline{\beta}$ over a vocabulary with one unary predicate symbol R where $\overline{\beta}$ is obtained by replacing each p_i with $R(x_i)$. Now consider the "binary" model $\mathcal{B}$ whose universe is $\{0, 1\}$ and where R is interpreted as $\{1\}$. Clearly β is satisfiable iff $\mathcal{B} \models \varphi$. It is also easy to see that if φ is satisfiable then it holds true in $\mathcal{B}$. Therefore, β is satisfiable iff φ is. $\Box$

And this finally gives a class of FOL sentences for validity is decidable:

Corollary 2.4 *Validity of universal sentences is decidable and coNP-complete.*

In the same Börger-Grädel-Gurevich monograph we find the "classical" classes of sentences *without equality and without function symbols* for which satisfiability is decidable:

- The Löwenheim class, just unary relation symbols, all prefixes.
- The Bernays-Schönfinkel class, quantifier prefix $\exists^*\forall^*$.
- Ackermann's class, quantifier prefix $\exists^*\forall\exists^*$.
- The Gödel/Kalmár/Schütte class, quantifier prefix $\exists^*\forall^2\exists^*$.

There are interesting stories related to extending these classes with equality or function symbols.

For certain simple existential sentences satisfiability is trivially decidable: they are in fact all satisfiable!

Exercise 2.1 *An* existential-conjunctive *sentence is a sentence of the form* $\exists x_1 \cdots \exists x_n \, \varphi$ *where* φ *is a conjunction of atomic formulas (equalities are allowed). Prove that any existential-conjunctive sentence is satisfiable in a model with one element.*

Such sentences are related to the *conjunctive queries* studied in databases.