Review

Last time
- Compactness
- Enumerability Theorem
- Definability of Models
- Finite Models
- Size of Models

Outline
- Theories
- Congruence Closure
- Interpretations Between Theories

Sources:
Sections 2.6 through 2.7 of Enderton.

Size of Models
The cardinality \(|L|\) of a language \(L\) is the least infinite cardinal greater than or equal to the number of symbols in the signature of \(L\).

The cardinality \(|M|\) of a model \(M\) is the cardinality of its domain \(\text{dom}(M)\).

Löwenheim-Skolem (LS) Theorem
Let \(\Gamma\) be a satisfiable set of formulas in a language \(L\), then \(\Gamma\) is satisfiable in some model of cardinality \(\kappa \leq |L|\).

Proof
By soundness, \(\Gamma\) is consistent, and is thus satisfiable by the model constructed in the proof of the completeness theorem. But the domain of that model is \(M/E\) which has cardinality \(\leq |M|\), and \(|M| = |L|\).
**Size of Models**

**“Skolem’s paradox”**

Let $A_{ST}$ be your favorite set of axioms for set theory. If they are consistent, they have a model. Because the signature of the language of set theory is finite, there is a countable model. But we can prove, starting with the axioms of set theory, that there are “uncountably” many sets.

How is this possible?

The answer is that in the countable model of set theory, things do not correspond to what we normally think of as the model of set theory. Thus, the model of the “natural numbers” in the countable model cannot be put in one-to-one correspondence with all of the elements of the model. But this does not mean that the size of the model is truly uncountable.

---

**Size of Models**

**LST Theorem**

Let $\Gamma$ be a set of formulas in a language of cardinality $\kappa$, and assume that $\Gamma$ is satisfiable in some infinite model. Then for every cardinal $\lambda \geq \kappa$, there is a model of cardinality $\lambda$ in which $\Gamma$ is satisfiable.

**Proof**

Let $M$ be an infinite model where $\Gamma$ is satisfiable. Expand the language by adding a set $C$ of $\lambda$ new constant symbols. Let $\Delta = \{c_1 \neq c_2 \mid c_1, c_2$ are distinct members of $C\}$. Then, any finite subset $\Gamma_0$ of $\Gamma \cup \Delta$ is satisfiable in $M'$ where $M'$ is $M$ extended to map all constants in $\Gamma_0$ to different elements of $M$. By compactness, $\Gamma \cup \Delta$ is satisfiable. By the LST Theorem, there is a model of cardinality $\leq \lambda$. But any model must have at least cardinality $\lambda$. Thus there is a model of cardinality $\lambda$.

---

**Theories**

Last time, we defined a **theory** as a set of first-order sentences. For this lecture we will refine our definition to be a set of first-order sentences **closed under logical implication**.

Thus, $T$ is a theory if $T$ is a set of sentences and if $T \models \sigma$, then $\sigma \in T$ for every sentence $\sigma$.

**Examples**

- For a given signature, the smallest possible theory consists of exactly the valid sentences over that signature.
- The largest theory for a given signature is the set of all sentences. It is the only unsatisfiable theory. Why?
Theories

For a class $\mathcal{K}$ of models over a given signature $\Sigma$, define the theory of $\mathcal{K}$ as

$$Th\mathcal{K} = \{ \sigma \mid \sigma \text{ is a } \Sigma\text{-sentence which is true in every model in } \mathcal{K}\}.$$ 

Theorem

$Th\mathcal{K}$ is indeed a theory.

Proof

Suppose $Th\mathcal{K} \models \sigma$. We know that $\models_M Th\mathcal{K}$ for each $M$ in $\mathcal{K}$. It follows that $\models_M \sigma$ for each $M$ in $\mathcal{K}$, and thus $\sigma \in Th\mathcal{K}$. □

Suppose $\Gamma$ is a set of sentences.

Define the set $Cn \Gamma$ of consequences of $\Gamma$ to be $\{ \sigma \mid \Gamma \models \sigma \}$.

Then $Cn \Gamma = Th Mod \Gamma$.

Theories

Using the above terminology, we can restate our earlier results as follows:

- An axiomatizable theory (in a reasonable language) is effectively enumerable.
- A complete axiomatizable theory (in a reasonable language) is decidable.

Our results about theories can be summarized in the following diagram.

Decidable $\iff$ if complete $\iff$ Effectively Enumerable $\iff$ Axiomatizable

Los-Vaught Test

For a theory $T$ and a cardinal $\lambda$, say that $T$ is $\lambda$-categorical iff all models of $T$ having cardinality $\lambda$ are isomorphic.

Theorem

Let $T$ be a theory in a countable language such that

- $T$ is $\lambda$-categorical for some infinite cardinal $\lambda$.
- All models of $T$ are infinite.

Then $T$ is complete.

Proof

It suffices to show that for any two models $M$ and $M'$ of $T$, $M \equiv M'$. Since $M$ and $M'$ are infinite, there exist (by LST) elementarily equivalent models of cardinality $\lambda$. But these models must be isomorphic, and by the homomorphism theorem, isomorphic models are elementarily equivalent. □
Validity and Satisfiability Modulo Theories

Given a $\Sigma$-theory $T$, a $\Sigma$-formula $\phi$ is

1. **$T$-valid** if $\models_M \phi[s]$ for all models $M$ of $T$ and all variable assignments $s$.
2. **$T$-satisfiable** if there exists some model $M$ of $T$ and variable assignment $s$ such that $\models_M \phi[s]$.
3. **$T$-unsatisfiable** if $\not\models_M \phi[s]$ for all models $M$ of $T$ and all variable assignments $s$.

The **validity problem** for $T$ is the problem of deciding, for each $\Sigma$-formula $\phi$, whether $\phi$ is $T$-valid.

The **satisfiability problem** for $T$ is the problem of deciding, for each $\Sigma$-formula $\phi$, whether $\phi$ is $T$-satisfiable.

Similarly, one can define the **quantifier-free validity problem** and the **quantifier-free satisfiability problem** for a $\Sigma$-theory $T$ by restricting the formula $\phi$ to be quantifier-free.

---

The Theory $T_E$ of Equality

The theory $T_E$ of equality is the theory $\text{Cn} \ \emptyset$.

Note that the exact set of sentences in $T_E$ depends on the signature in question.

The theory does not restrict the possible values of symbols in any way. For this reason, it is sometimes called the theory of **equality with uninterpreted functions (EUF)**.

The satisfiability problem for $T_E$ is just the satisfiability problem for first order logic, which is undecidable.

The satisfiability problem for conjunctions of literals in $T_E$ is decidable in polynomial time using **congruence closure**.

---

The Theory $T_Z$ of Integers

Let $\Sigma_Z$ be the signature $(0, 1, +, -, \leq)$.

Let $A_Z$ be the standard model of the integers with domain $\mathbb{Z}$.

Then $T_Z$ is defined to be $\text{Th} A_Z$.

As showed by Presburger in 1929, the validity problem for $T_Z$ is decidable, but its complexity is triply-exponential.

The quantifier-free satisfiability problem for $T_Z$ is “only” NP-complete.

Let $\Sigma_Z^\times$ be the same as $\Sigma_Z$ with the addition of the symbol $\times$ for multiplication, and define $A_Z^\times$ and $T_Z^\times$ in the obvious way.

The satisfiability problem for $T_Z^\times$ is undecidable (a consequence of Gödel’s incompleteness theorem).

In fact, even the quantifier-free satisfiability problem for $T_Z^\times$ is undecidable.
The Theory $T_R$ of Reals

Let $\Sigma_R$ be the signature $(0, 1, +, -, \leq)$.

Let $A_R$ be the standard model of the reals with domain $\mathbb{R}$.

Then $T_R$ is defined to be $Th_{A_R}$.

The satisfiability problem for $T_R$ is decidable, but the complexity is doubly-exponential.

The quantifier-free satisfiability problem for conjunctions of literals (atomic formulas or their negations) in $T_R$ is solvable in polynomial time, though exponential methods (like Simplex or Fourier-Motzkin) often perform better in practice.

Let $\Sigma^\times_R$ be the same as $\Sigma_R$ with the addition of the symbol $\times$ for multiplication, and define $A_R^\times$ and $T_R^\times$ in the obvious way.

In contrast to the theory of integers, the satisfiability problem for $T_R^\times$ is decidable.

The Theory $T_A$ of Arrays

Let $\Sigma_A$ be the signature ($\text{read}$, $\text{write}$).

Let $\Lambda_A$ be the following axioms:

\begin{align*}
\forall a \forall i \forall v \ (\text{read} (\text{write} (a, i, v), i) = v) \\
\forall a \forall i \forall j \forall v \ (i \neq j \rightarrow \text{read} (\text{write} (a, i, v), j) = \text{read} (a, j)) \\
\forall a \forall b \ ((\forall i \ (\text{read} (a, i) = \text{read} (b, i))) \rightarrow a = b)
\end{align*}

Then $T_A = Cn \Lambda_A$.

The satisfiability problem for $T_A$ is undecidable, but the quantifier-free satisfiability problem for $T_A$ is decidable (the problem is NP-complete).

Theories of Inductive Data Types

An inductive data type (IDT) defines one or more constructors, and possibly also selectors and testers.

Example: list of int

- Constructors: $\text{cons} : (\text{int}, \text{list}) \rightarrow \text{list}$, $\text{null} : \text{list}$
- Selectors: $\text{car} : \text{list} \rightarrow \text{int}$, $\text{cdr} : \text{list} \rightarrow \text{list}$
- Testers: $\text{is}_\text{cons}$, $\text{is}_\text{null}$

The first order theory of an inductive data type associates a function symbol with each constructor and selector and a predicate symbol with each tester.

Example: $\forall x : \text{list}. \ (x = \text{null} \lor \exists y : \text{int}, z : \text{list}. \ x = \text{cons} (y, z))$

For IDTs with a single constructor, a conjunction of literals is decidable in polynomial time.

For more general IDTs, the problem is NP-complete, but reasonably efficient algorithms exist in practice.

Other Interesting Theories

Some other interesting theories include:

- Theories of bit-vectors
- Fragments of set theory
**Congruence Closure**

Let $G = (V, E)$ be a directed graph such that for each vertex $v$ in $G$, the successors of $v$ are ordered.

Let $C$ be any equivalence relation on $V$.

The **congruence closure** $C^*$ of $C$ is the finest equivalence relation on $V$ that contains $C$ and satisfies the following property for all vertices $v$ and $w$.

- Let $v$ and $w$ have successors $v_1, \ldots, v_k$ and $w_1, \ldots, w_l$ respectively. If $k = l$ and $(v_i, w_i) \in C^*$ for $1 \leq i \leq k$, then $(v, w) \in C^*$.

In other words, if the corresponding successors of $v$ and $w$ are equivalent under $C^*$, then $v$ and $w$ are themselves equivalent under $C^*$.

Often, the vertices are labeled by some labeling function $\lambda$. In this case, the property becomes:

- If $\lambda(v) = \lambda(w)$ and if $k = l$ and $(v_i, w_i) \in C^*$ for $1 \leq i \leq k$, then $(v, w) \in C^*$.

**A Simple Algorithm**

Let $C_0 = C$ and $i = 0$.

1. Number the equivalence classes in $C_i$ consecutively from 1.
2. Let $\alpha$ assign to each vertex $v$ the number $\alpha(v)$ of the equivalence class containing $v$.
3. For each vertex $v$ construct a signature $s(v) = \lambda(v)(\alpha(v_1), \ldots, \alpha(v_k))$, where $v_1, \ldots, v_k$ are the successors of $v$.
4. Group the vertices into classes of vertices having equal signatures.
5. Let $C_{i+1}$ be the finest equivalence relation on $V$ such that two vertices equivalent under $C_i$ or having the same signature are equivalent under $C_{i+1}$.
6. If $C_{i+1} = C_i$, let $C^* = C_i$; otherwise increment $i$ and repeat.

**Congruence Closure and $T_\Sigma$**

Recall that $T_\Sigma$ is the empty theory with equality over some signature $\Sigma$ containing only function symbols.

If $\Gamma$ is a set of ground $\Sigma$-equalities and $\Delta$ is a set of ground $\Sigma$-disequalities, then the satisfiability of $\Gamma \cup \Delta$ can be determined as follows.

- Let $G$ be a graph which corresponds to the abstract syntax trees of terms in $\Gamma \cup \Delta$, and let $v_t$ denote the vertex of $G$ associated with the term $t$.
- Let $C$ be the equivalence relation on the vertices of $G$ induced by $\Gamma$.
- $\Gamma \cup \Delta$ is satisfiable iff for each $s \neq t \in \Delta$, $(v_s, v_t) \notin C^*$.

**An Algorithm for $T_\Sigma$**

`union` and `find` are abstract operations for manipulating equivalence classes.

`union(x, y)` merges the equivalence classes of $x$ and $y$.

`find(x)` returns a unique representative of the equivalence class of $x$.

$CC(\Gamma, \Delta)$

Construct $G(V, E)$ from terms in $\Gamma$ and $\Delta$.

while $\Gamma \neq \emptyset$

- Remove some equality $a = b$ from $\Gamma$;
- $\text{Merge}(a, b)$;
- if `find(a) = find(b)` for some $a \neq b \in \Delta$ then
  return `false`
- return `true`
An Algorithm for $T_E$

$\text{Merge}(a, b)$

\[
\begin{align*}
\text{if } \text{find}(a) &= \text{find}(b) \text{ then return; } \\
\text{Let } A \text{ be the set of all predecessors of all vertices equivalent to } a; \\
\text{Let } B \text{ be the set of all predecessors of all vertices equivalent to } b; \\
\text{union}(a, b); \\
\text{foreach } x \in A \text{ and } y \in B \\
\text{if } \text{signature}(x) = \text{signature}(y) \text{ then Merge}(x, y); \\
\end{align*}
\]

Congruence Closure

DST Algorithm

The Downey-Sethi-Tarjan Congruence Closure algorithm is more efficient. It makes use of some additional data structures and methods.

Additional Helper Methods

- $\text{union}(a, b)$ in this algorithm, the first argument always becomes the new equivalence class representative.
- $\text{list}(e)$ returns the list of vertices with at least one successor in equivalence class $e$.
- $\text{enter}(v)$ stores $(v, \text{signature}(v))$ in a signature table.
- $\text{delete}(v)$ removes $(v, \text{signature}(v))$ from the signature table if it is there. Note that this operation does not remove any other entry, even if it has the same signature as $v$.
- $\text{query}(v)$ if there is an entry $(w, \text{signature}(w))$ in the signature table, and $\text{signature}(w) = \text{signature}(v)$, then return $w$; otherwise, return $\bot$.

DST Algorithm

$\text{Merge}(\text{combine})$

\[
\begin{align*}
\text{pending} &:= \text{set of all vertices;} \\
\text{while } \text{pending} \neq \emptyset \\
\text{foreach } v \in \text{pending} \\
\text{if } \text{query}(v) = \bot \text{ then enter}(v); \\
\text{else add } (v, \text{query}(v)) \text{ to } \text{combine}; \\
\text{pending} &:= \emptyset; \\
\text{foreach } (a, b) \in \text{combine} \\
\text{if } \text{find}(a) \neq \text{find}(b) \text{ then } \\
\text{if } |\text{list}(\text{find}(a))| < |\text{list}(\text{find}(b))| \text{ then swap } a \text{ and } b; \\
\text{foreach } u \in \text{list}(\text{find}(b)) \\
\text{delete}(u); \text{ add } u \text{ to } \text{pending}; \\
\text{union}(\text{find}(a), \text{find}(b)); \\
\text{combine} &:= \emptyset; \\
\end{align*}
\]
Interpretations Between Theories

Given two theories, $T_0$ and $T_1$, sometimes it is possible to show that one of the theories is at least as powerful as the other.

A simple case is when $T_0$ and $T_1$ are in the same language and one is a subset of the other.

We will consider more general cases in which the languages of the two theories differ.

Defining Functions

A common practice in mathematical reasoning is to introduce a new piece of notation, defining it in terms of a formula not containing the new notation.

Formally, suppose $\Sigma$ is a signature and $f$ is a function symbol not in $\Sigma$. Let $\Sigma^+ = \Sigma \cup \{ f \}$. A function definition is a formula of the form:

$$ \forall v_1 \forall v_2 \ (fv_1 = v_2 \leftrightarrow \phi) $$

where $\phi$ is a $\Sigma$-formula in which only $v_1$ and $v_2$ may occur free. Let the above sentence be designated $\delta$.

Theorem

The following are equivalent:

1. The definition is non-creative, i.e. for any $\Sigma$-sentence $\sigma$ if $T \cup \{ \delta \} \models \sigma$ in $\Sigma^+$, then $T \models \sigma$ in $\Sigma$.
2. $f$ is well-defined, i.e. the following sentence (which we designate $\epsilon$) is in the theory $T$:

$$ \forall v_1 \exists! v_2 \phi. $$

Note that $\exists!$ means “there exists uniquely”. Technically it is an abbreviation for a more complicated formula which expresses both existence and uniqueness.

Proof

To show that (1) implies (2), note that $\delta \models \epsilon$. Thus, if we take $\sigma \equiv \epsilon$ in (1), it follows that $T \models \epsilon$.

To show that (2) implies (1), suppose that $T \models \epsilon$. Let $M$ be a $\Sigma$-model of $T$. For $d \in \text{dom}(M)$, let $F(d)$ be the unique $e \in \text{dom}(M)$ such that $M \models \phi[[d, e]]$. The existence of such an $e$ for each $d$ is ensured because $T \models \epsilon$. Now let $(M, F)$ be the $\Sigma^+$ model which agrees with $M$ on all parameters except $F$ and which assigns $F$ to the symbol $f$. It is easy to see that $(M, F)$ is a model of $\delta$.

Thus, if $T \cup \{ \delta \} \models \sigma$, then $(M, F) \models \sigma$. But since $\sigma$ is a $\Sigma$-sentence, it follows that $M \models \sigma$. Since $M$ was chosen arbitrarily, it follows that $T \models \sigma$.

Interpretations

There are more general ways in which one theory can be as strong as another theory in another language.

Example

Consider the theory of $(\mathbb{N}, 0, S)$ (natural numbers with 0 and successor) and on the other hand the theory of $(\mathbb{Z}, +, \times)$.

The second theory is at least as strong as the first. To show this, we make the following observations:

- An integer is nonnegative iff it is the sum of four squares.
- The set $\{ 0 \}$ is defined by $v_1 + v_1 = v_1$.
- The successor relation is defined by

$$ \forall z (z \times z = z \land z + z \neq z \rightarrow v_1 + z = v_2). $$

Thus, for example, the sentence $\forall x.Sx \neq 0$ can be translated as

$$ \forall x \exists y_1 \exists y_2 \exists y_3 \exists y_4 \ x = y_1 \times y_1 + y_2 \times y_2 + y_3 \times y_3 + y_4 \times y_4 \rightarrow \neg \forall u \ (u + u = u \rightarrow \forall v \ (\forall z (z \times z = z \land z + z \neq z \rightarrow x + z = v) \rightarrow v = u))). $$
Interpretations

Suppose $\Sigma_0$ and $\Sigma_1$ are signatures and $T_1$ is a $\Sigma_1$-theory.

An interpretation $\pi$ of $\Sigma_0$ into $T_1$ consists of the following three items.

1. a $\Sigma_1$-formula $\pi \varphi$ in which at most $v_1$ occurs free, such that
   - $T_1 \models \exists v_1. \pi \varphi$.

2. a $\Sigma_1$-formula $\pi \psi$ for each $n$-ary predicate symbol $P \in \Sigma_0$ in which at most
   the variables $v_1, \ldots, v_n$ occur free.

3. a $\Sigma_1$-formula $\pi \lambda$ for each $n$-ary function symbol $f \in \Sigma_0$ in which at most
   $v_1, \ldots, v_n, v_{n+1}$ occur free such that
   - $T_1 \models \forall v_1, \ldots, v_n. (\pi \psi(v_1) \rightarrow \cdots \rightarrow \pi \psi(v_n) \rightarrow \exists x. (\pi \varphi(x) \land \forall v_{n+1}. (\pi f(v_1, \ldots, v_{n+1}) \leftrightarrow v_{n+1} = x))).$

For our previous example, we have

1. $\pi \varphi(x) = \exists y_1 \exists y_2 \exists y_3 \exists y_4. x = y_1 \times y_1 + y_2 \times y_2 + y_3 \times y_3 + y_4 \times y_4$
2. $\pi \psi(y) = x + x = x$
3. $\pi \lambda(x, y) = \forall y (z \times z + z \neq z \rightarrow x + z = y)$

Interpretations

Given a $\Sigma_0$ formula $\varphi$ and an interpretation $\pi$ of $\Sigma_0$ into $T_1$, we can find a

formula $\pi \varphi$ which in some sense corresponds exactly to $\varphi$.

For an atomic formula $\alpha$, we scan the formula from right to left. When a function
symbol $f$ is found, applied to arguments $x_1, \ldots, x_n$, it is replaced by a new

variable $y$ and the atomic formula is prefixed by $\forall y (\pi f(x_1, \ldots, x_n, y) \rightarrow)$. We

continue until there are no more function symbols. Finally, we replace the

predicate symbol $P$ (if it is not equality) by $\pi \psi$.

For example,

$$\pi \varphi(x, y) = \forall y (\pi \varphi(x, y) \rightarrow (P \pi \psi(x, y))$$

$$= \forall y (\pi \varphi(x, y) \rightarrow \forall z (\pi f(y, z) \rightarrow (P \pi \psi(z))))$$

$$= \forall y (\pi \varphi(x, y) \rightarrow \forall z (\pi f(y, z) \rightarrow \pi \lambda z)).$$

The interpretation of non-atomic formulas are defined in the obvious way:

$$(\neg \varphi)^\pi = \neg (\pi \varphi)^\pi,$$

$$(\varphi \rightarrow \psi)^\pi = (\pi \varphi \rightarrow \pi \psi)^\pi,$$

and

$$(\forall \psi \varphi)^\pi = \forall x (\pi \varphi(x) \rightarrow \pi \varphi^\pi).$$

Interpretations

Assume that $\pi$ is an interpretation and let $M$ be a model of $T_1$.

There is a natural way to extract from $M$ a model $\pi M$ for $\Sigma_0$. Let

- $\text{dom}(\pi M) = \{1 \text{ } \text{ } \text{ } | \text{ } M \}$
- $\text{P} \pi M = \text{the relation defined in } M \text{ by } \pi P,$
- $\pi f \pi M(a_1, \ldots, a_n) = \{b \text{ } \text{ } \text{ } | \text{ } M \}$

where $a_1, \ldots, a_n$ are in $\text{dom}(\pi M)$.

By condition (i), $\text{dom}(\pi M) \neq \emptyset$. By condition (ii), the definition of $\pi f \pi M$ makes

sense.

Define the set $\pi^{-1}[T_1]$ of $\Sigma_0$-sentences as

$$\text{Th}(\pi M | M \in \text{Mod} T_1).$$

In other words, $\pi^{-1}[T_1]$ is the set of all $\Sigma_0$-sentences true in every model

obtainable from a model of $T_1$ as shown above.

Interpretations

Lemma

Let $\pi$ be an interpretation of $\Sigma_0$ into $T_1$ and $M$ a model of $T_1$. For any

$\Sigma_0$-formula $\varphi$ and any map $s$ of the variables into $\text{dom}(\pi M)$,

$$\models \pi M \varphi[s] \text{ if } \models \pi \varphi[s].$$

The proof is by induction on $\varphi$ and is omitted.

Corollary

For a $\Sigma_0$-sentence $\sigma, \sigma \in \pi^{-1}[T_1]$ iff $\sigma ^\pi \in T_1$.

An interpretation $\pi$ of a theory $T_0$ into a theory $T_1$ is an interpretation $\pi$ of the

signature of $T_0$ into $T_1$ such that $T_0 \subseteq \pi^{-1}[T_1]$.

If $T_0 = \pi^{-1}[T_1]$, then $\pi$ is said to be a faithful interpretation of $T_0$ into $T_1$.

For a faithful interpretation, we have $\sigma \in T_0$ if $\sigma ^\pi \in T_1$. 