Math Logic, Week#VII

Moon's Day

¿Any Q U E S T I O N S yet on the soundness theorem?

- ¿on the strategy for the soundness theorem?
- ¿on the proof of the validity of any of the axioms?

¿Any other Q U E S T I O N S on last week's material?

A Hint on Exercise 2.2.14: Important point here: we've really seen only one way to show that a relation on a structure is *undefinable* ...

automorphisms.

- **¿What's an automorphism of a structure?**
- Contrast the structure $(\mathbb{N}, <)$. What automorphisms does it have?

  It has only the identity automorphism — the (uninteresting) automorphism that maps each element to itself.

  ¿Why?

- By contrast, $(\mathbb{R}, <)$ has *many* automorphisms?
  - Can you think of any?
  - Can you get an intuitive idea of what they all are?
  - Indeed,

    if unary relation on $(\mathbb{R}, <)$ is *not* moved by any automorphism it is easy to define.
  - Indeed, every unary relation on $(\mathbb{R}, <)$ that is *not* moved by any automorphisms is easy to define.
  - For binary relations — so subsets of $\mathbb{R} \times \mathbb{R}$ — the situation is a bit more complicated, but the approach is the same.

Definitions:

- A contradiction is a formula $(\varphi \wedge \neg\varphi)$, for any formula φ.

- **Aside:** How do we know that if $\Gamma \vdash (\varphi \wedge \neg\varphi)$,
  then for *every* formula ψ of the language, $\Gamma \vdash (\psi \wedge \neg\psi)$?

- A set Γ of formulas is *inconsistent* if $\Gamma \vdash (\varphi \wedge \neg\varphi)$ (for any φ).
  (Otherwise, Γ is *consistent*.)

Completeness of the proof system: *Show that if $\Gamma \models \varphi$, $\Gamma \vdash \varphi$.*

- It's enough to show that if Γ is *unsatisfiable*

  then Γ is *inconsistent*.

- We'll actually prove the contrapositive:

  If Γ is consistent,
  then Γ is satisfiable.

- So we need to build a model of Γ — thus, build a structure.
  ¿What material do we have to build the structure out of?

  ¡Not much!
  All we have are language elements.
  So that's what L. Henkin (and J. Herbrand) did for this proof.
  Add infinitely many additional constant symbols to the language.
  The universe of the model is the set of terms of this extended language.

Completeness Theorem Disclaimer: We'll prove the completness theorem only for axiom systems Γ in *countable* languages.

Here "*countable*" means either finite

or "countably infinite" — where we're able to list the symbols in the language indexed by natural numbers.

If, say, we had one constant for each *real number* r, this proof would not work — and the proof needs (a weak form of) the the *Axiom of Choice*.

Completeness, Step #1: *Expand the language by adding a countably infinite set of new constant symbols.*

I'll call the original language $\mathcal{L}$ and the expanded langauge $\mathcal{K}$.

Theorem: *Let* Γ *be a consistent set of* $\mathcal{L}$*-formulas. Then* Γ *is consistent in* $\mathcal{K}$*.*

Proof Sketch:

- How could Γ be inconsistent in $\mathcal{K}$?
- There'd have to be a $\mathcal{K}$-formula β where $\Gamma \vdash (\beta \wedge \neg\beta)$.
- But then Enderton's theorem on generalization on constants would let us replace all the new constants in β with variables
- And we'd have a proof of a contradiction in $\mathcal{L}$, contradicting the assuption that Γ is consistent.

Completeness Step 1.5: *Enumerate all the $\mathcal{K}$-formulas — φ_1, φ_2, ...*

This is a result from Chapter 0: Since $\mathcal{K}$ is countably infinite, there are countably many finite strings of symbols from

$$\mathcal{K} \cup \{v_1, v_2, v_3, \ldots, \text{“(”}, \text{“)”}, \text{“}\neg\text{”}, \text{“}\rightarrow\text{”}, \text{“}\forall\text{”}, \text{“,”}\}$$

(the comma is for my syntax; Enderton doesn't use it)

and every $\mathcal{K}$-formula is one of those strings.

Completeness Step 2: *For each $\mathcal{K}$-formula φ, add to Γ the formula $\neg\forall x\varphi \rightarrow \neg\varphi^x_c$, where c is one of the new constant symbols.*

(Then c is called a *witness* to $\neg\forall x\varphi$.)

Theorem *We can do this so that Γ is still consistent.*

Proof Skecth:

- Go through the $\mathcal{K}$-formulas in the order φ_1, φ_2, ...
- For each φ_i there are only finitely many of the new constants we've encountered so far (either in a formula φ_j, $j < i$, or as a witness for a φ_j)
- Pick one of the remaining new constants for the witness.

Enderton's theorems on new constants tell us Γ is still consistent.

Completeness Step 3: *Expand Γ to a set Δ of $\mathcal{K}$-formulas such that*

- *Δ is consistent, and*
- *For any $\mathcal{K}$-formula φ, either $\varphi \in \Delta$ or $(\neg\varphi) \in \Delta$.*

This construction is pretty much the same as in propositional logic — exercise 1.7.1.

Completness Step 4: Replace the $=$ symbol in Γ with a new binary relation symbol E.

Build a structure $\mathfrak{A}$ *from* Δ**:**

- $|\mathfrak{A}|$ is the set of all terms of language $\mathcal{K}$ (including terms with variable symbols)
- For a k-place relation P of $\mathcal{L}$ — including E:

$$P^{\mathfrak{A}} \text{ is } \{(t_1, \ldots, t_k) : P(t_1, \ldots, t_k) \in \Delta\}$$

- For each constant symbol c, $c^{\mathfrak{A}}$ is c.
- For each k-place function symbol f and each $t_1, \ldots, t_k \in |\mathfrak{A}|$ — recall that $t_1, \ldots, t_k$ are themselves terms of language $\mathcal{K}$ — $f^{\mathfrak{A}}(t_1, \ldots, t_k)$ is (the term) $f(t_1, \ldots, t_k)$

Completeness, Step 4.5: Let s be the variable interpretation mapping each variable to itself.

Theorem: $\mathfrak{A} \models \Gamma[s]$ — where we have modified Γ by replacing each occurrence of "$=$" with "E".

Proof: By fairly straightforward induction on formulas.

(We did a similar exercise for propositional logic in §1.7.)

Read quickly through the details: induction on formulas is always important.

Completeness Step 5: From $\mathfrak{A}$, construct a structure $\mathfrak{B}$ that handles the symbol $=$ correctly.

- Recall that, in first order logic with the equality symbol,
  we require $=$ to be interpreted by the real equality relation on the universe.

  But that's not true in our structure $\mathfrak{A}$ above.

- Important properties of $E^{\mathfrak{A}}$ — all provable from Enderton's equality axioms:
  1. It's an *equivalence relation*:
     - It's reflexive: for each term t, $(t, t) \in E^{\mathfrak{A}}$.
     - It's symmetric: for all terms t_1, t_2, if $(t_1, t_2) \in E^{\mathfrak{A}}$, then $(t_2, t_1) \in E^{\mathfrak{A}}$ also.
     - It's transitive: for all terme t_1, t_2, t_3, if $(t_1, t_2) \in E^{\mathfrak{A}}$ and $(t_2, t_3) \in E^{\mathfrak{A}}$ then $(t_1, t_3) \in E^{\mathfrak{A}}$.

     So can apply standard theorem from mathematics:
     - We can divide — *partition* — the elements of $|\mathfrak{A}|$ into disjoint subsets,
     - where each $t \in |\mathfrak{A}$ is in exactly 1 subset, called $[t]$, and
     - for any $t_1, t_2 \in |\mathfrak{A}|$,

     $$t_2 \in [t_1] \text{ iff } (t_1, t_2) \in E^{\mathfrak{A}}.$$

     Each set $[t]$ is called the $E^{\mathfrak{A}}$*-equivalence class of* t.

     The universe $|\mathfrak{B}|$ of structure $\mathfrak{B}$ is the set of all these equivalence classes.

2. And, $E^{\mathfrak{A}}$ is a *congruence relation* on $\mathfrak{A}$ (which I won't define here but will implicitly cover below):

   – For c a constant symbol of $\mathcal{K}$, $c^{\mathfrak{B}} = [c^{\mathfrak{A}}]$.

   – For a k-ary relation P of $\mathcal{L}$, we define

   $$([t_1], \ldots, [t_k]) \in P^{\mathfrak{B}} \text{ iff } (t_1, \ldots, t_k) \in P^{\mathfrak{A}}.$$

   But is that ambiguous? What happens if we look at two different elements, $t_1, t_1' \in [t_1]$; do we get a conflict about whether $([t_1], \ldots, [t_k]) \in P^{\mathfrak{B}}$.
   We can prove from the equality axioms that this can't happen.

   – Similarly, for f a k-ary function symbol, and for $t_1, \ldots, t_k \in |\mathfrak{A}|$, we set

   $$f^{\mathfrak{B}}([t_1], \ldots, [t_k]) \text{ to be } [f(t_1, \ldots, t_k)]$$

   and the equality axioms prove that this is unambigouous.

- *Show by a straightforward induction on formulas* φ —
  where φ^E is the result of replacing each occurrence of "=" with "E"
  and s' maps each variable v_i to $[s(v_i)]$
  that

  $$\mathfrak{B} \models \varphi[s] \text{ iff } \mathfrak{A} \models \varphi^E[s'].$$

Step 6: A technicality: change to the desired language.

- We showed above that, for each formula $\varphi \in \Gamma$, $\mathfrak{B} \models \varphi[s']$.
- But technically, we haven't done what was asked for.
- We wanted a structure for the original language $\mathcal{L}$, but we got a structure for the new language $\mathcal{K}$.
- The cure: In $\mathfrak{B}$,
  - keep the same $|\mathfrak{B}|$,
  - interpret each symbol in $\mathcal{L}$ exactly as we do in $\mathfrak{B}$,
  - but just don't interpret the extra symbols in $\mathcal{K}$ — "forget" those interpretations.

  (The result is called the *reduct* of $\mathfrak{B}$ to $\mathcal{L}$.)

Corrolaries of Soundness and Completeness: (For the proof of completness we have done above, these results are shown only for countable languages $\mathcal{L}$.)

Compactness Theorem: *A set Γ of formulas is satisfiable iff every finite subset of Γ is satisfiable.*

Proof: It's obvious that Γ of formulas is consistent iff every finite subset of Γ is consistent

— merely because every proof of a contradiction is finite, and hence involves only finitely many formulas from Γ.

But, by soundness and completeness, satisfiability is the same as consistence above.

Enumerability Theorem:

- *Suppose $\mathcal{L}$ is finite.* (We could allow $\mathcal{L}$ to be infinite but "reasonable.")
  Then the set of valid formulas of $\mathcal{L}$ can be effectively enumerated.

- **Proof:** We can express each proof as a finite string of symbols from $\mathcal{L} \cup \{v_1, v_2, v_3, \ldots, \text{"("}, \text{")"}, \text{"}\neg\text{"}, \text{"}\rightarrow\text{"}, \text{"}\forall\text{"}, \text{","}\}$.

  I leave it to you to sketch how to write a program to generate all such strings. Feel free to add line feeds, line numbers, and some way to encode some way to encode reasons if you wish. (But that's for your programming skills; I won't ask you to claim or present that.)

- Now just generate all these strings, check whether each is a proof from $\Gamma = \emptyset$, and, if it is, output the formula it proves — and then go on.

- It follows that $\{\varphi : \emptyset \models \varphi\}$ is semi-decidable:
  For a formula φ, just start the above process and, if it ever outputs φ, output "yes" and halt both programs.'

- **But** the set of all valid formulas is not decidable.
  (We'll almost get that in Chapter 3.)

Extended Enumerability: *If Γ is finite* (or "reasonable"), *then $\{\varphi : \Gamma \models \varphi\}$ is semi-decidable.*

More homework problems: §2.5 ## 4 & 7.

Comment on finite models:

1. Easy: find a formula that has infinite models but no finite ones.
2. For many CS purposes, we are interested only in finite models
   — *e.g.*, databases with no restrictions on their size
   — *or* computers with no limits on their memory.
3. Suppose we restrict attention to *finite* structures.
   - Enderton's proof system is still sound, but it's incomplete.
     ¿How do we know that?
   - For first order logic, for finite and infinite structures,
     the set of *valid* formulas is semi-decidable but not decidable.
   - For first order logic, for finite structures only,
     the set of *satisfiable* formulas is semi-decidable but not decidable.
   - Accordingly, there is *no sound, complete, and decidable proof system* for first order logic restricted to finite structures.

Theorem 26B: *If a set Σ of sentences has arbitrarily large finite models, it has an infinite model.* (Actually, infinitely many.)

Proof: For natural numbers $k \geq 2$, the following sentence λ_k "says" there are at least k elements in the structure:

$$\exists x_1 \exists x_2 \ldots \exists x_k(\ (\neg x_2 = x_1) \wedge (\neg x_3 = x_1) \wedge (\neg x_3 = x_2) \wedge \ldots \\ \wedge (\neg x_k = x_1) \wedge (\neg x_k = x_2) \wedge \cdots \wedge (\neg x_k = x_{k-1}) \\ ).$$

Because Σ has arbitrarily large finite models, every finite subset of $\Sigma \cup \{\lambda_2, \lambda_3, \ldots, \lambda_k, \ldots\}$ is satisfiable.

So, by compactness, $\Sigma \cup \{\lambda_2, \lambda_3, \ldots, \lambda_k, \ldots\}$, is satisfiable.

Reading Assignment: Read §2.6 and skim §2.7.