Propositional Calculus (Part 4)

CS 4710 Course Notes

January 25, 2018

Completeness of PC

Def. 1 Set S of WFF's is *syntatically consistent* iff there is **no** WFF R such that:

$$S \vdash R \text{ and } S \vdash \neg R$$

Note that satisfaction is a semantic concept (defined around possible valuations), whereas consistency is a syntactic concept (defined around derivability by MP).

We'll rely on two facts to prove the completeness of PC. To be fully thorough, we would have to prove these facts as well. (In order to produce a good proof of Fact 2, we would have to spend more time on PC and sacrifice other topics. Since we're moving on soon, we'll simply accept these facts as proven and rely on them for our completeness proof.)

Fact 1. $S \vDash P$ iff $S \cup \{\neg P\}$ is *unsatisfiable.*

Fact 2. $S \vdash P$ iff $S \cup \{\neg P\}$ is *inconsistent.*

Completeness of PC

Def. 2 A set of WFF's S is *negation-complete* iff, for all statements R, **either** $R \in S$ **or** $\neg R \in S$.

We can rewrite soundness and completeness using our definitions for syntactic consistency and satisfiability. We'll use the rewritten version of completeness in our proof. Verifying that these restatements are valid is left as an exercise.

Soundness (rewritten): every satisfiable set of WFF's is syntactically consistent

Completeness (rewritten): every syntactically consistent set of WFF's is satisfiable

We will prove the rewritten version of completeness, as it's a bit easier to approach than the original form.

Proof. Let S be a syntactically consistent set of WFF. We will extend S to a negation-complete and consistent set of WFF's, Σ using a recursive definition.

Let $P_1, P_2, \ldots$ be an enumeration of all WFF (as an exercise, consider why we are justified asserting the existence of such an enumeration). Now we will define Σ as follows:

$$\Sigma_1 = S$$

$$\Sigma_{n+1} = \begin{cases} \Sigma_n \cup \{P_n\} & \text{if } \Sigma_n \cup \{P_n\} \text{ is consistent} \\ \Sigma_n \cup \{\neg P_n\} & \text{otherwise} \end{cases}$$

$$\Sigma = \bigcup_n \Sigma_n$$

Note that $\bigcup_n$ works the same as summation, except for set theoretic union – it acts as an iterator, performing a union operation on each index:

$$\bigcup_n S_n = S_1 \cup S_2 \cup \ldots \cup S_n$$

We can demonstrate that Σ is both consistent (by construction) and negation-complete by definition. We can use Fact 2 to help demonstrate this. If $\Sigma \not\vdash R$, by the contrapositive of Fact 2, $\Sigma \cup \{\neg R\}$ is consistent. (A note on contrapositives: if X iff Y is true, then $\neg X$ iff $\neg Y$ is also true.)

Since $\neg R$ is in our set of WFF's, there's some k such that $P_k = \neg R$. Therefore, $\Sigma_k \cup \{\neg R\}$ is consistent and, therefore, $\neg R \in \Sigma_k \subseteq \Sigma$. If $\Sigma \not\vdash \neg R$, instead, the argument runs the same way (check this for yourself).

Let V be:

$$[\![P_i]\!]^V = \begin{cases} \top & \Sigma \vdash P_i \\ \bot & \Sigma \vdash \neg P_i \end{cases}$$

Now we seek to prove: for each WFF R, $[\![R]\!]^V = \top$ iff $\Sigma \vdash R$. We'll do this by induction on the construction of WFF's, using only the operators $\{\neg, \wedge\}$ (since they are logically complete operators – as an exercise, verify this).

Base case. Let $R = P_i$. Then $[\![P_i]\!]^V = \top$ iff $\Sigma \vdash P_i$ (by definition). $\square$

Step. Let P, Q be WFF's such that $[\![P]\!]^V = \top$ iff $\Sigma \vdash P$ and $[\![Q]\!]^V = \top$ iff $\Sigma \vdash Q$. We will have to prove a case for each of our operators. Note we have provided ourselves with two statements, as $\wedge$ takes two statements.

Case 1: Let $R = \neg P$. By definition of consistency and negation-completeness:

$$\Sigma \vdash R \text{ iff } \Sigma \not\vdash P \text{ iff } [\![P]\!]^V = \bot \text{ iff } [\![R]\!]^V = \top$$

$\blacksquare$

Case 2: Let $R = P \wedge Q$. Since $R \rightarrow P$ and $R \rightarrow Q$ are tautologies, if $\Sigma \vdash R$ then $\Sigma \vdash P$ and $\Sigma \vdash Q$.

Conversely, if $\Sigma \vdash P$ and $\Sigma \vdash Q$, then $\Sigma \vdash R$. This is due to $P \rightarrow (Q \rightarrow (P \wedge Q))$ being a tautology (verify this yourself).

$$\Sigma \vdash R \text{ iff } (\Sigma \vdash P \text{ and } \Sigma \vdash Q) \text{ iff } ([\![P]\!]^V = \top \text{ and } [\![Q]\!]^V = \top) \text{ iff } [\![R]\!]^V = \top$$

For each $P \in S$, $S \vdash P$. Since $S \subseteq \Sigma$, $\Sigma \vdash P$. Therefore, $[\![P]\!]^V = \top$ and V satisfies Σ. $\blacksquare$

Completeness and soundness together mean that we can build an algorithm capable of deriving any true statement of PC, of arbitrary size.