Notes on propositional calculus and Hilbert systems

CS105L: Discrete Structures
I semester, 2006-07

Amitabha Bagchi*

August 16, 2006

1 Propositional formulas

The language of propositional calculus is a set of strings referred to as *propositional formulas* or simply *formulas.* These strings are generated by a *context-free grammar.* Let us first see what we mean by a context-free grammar.

1.1 Context-free grammars

A context-free grammar is a set of rules for generating a string. Every string in the language is generated from one distinguished symbol. To generate the strings we have two kinds of rules. The first kind of rule is of the form:

$$A ::= A_1 \ A_2 \ \cdots \ A_n \tag{1}$$

This means that an occurrence of the symbol A can be replaced with the string of symbols $A_1 \ A_2 \ \cdots \ A_n$.

The other kind of rule is of the form:

$$A ::= A_1 \mid A_2 \mid \cdots \mid A_n \tag{2}$$

This means that an occurrence of the symbol A can be replaced with either of A_1 or A_2 or so forth till A_n.

*These notes are based on the book **Mathematical Logic for Computer Science** by M. Ben-Ari (Prentice-Hall International (UK), 1993.)

The symbols which appear on the left side of a rule are known as *non-terminals* and the symbols which never occur on the left side (i.e. occur only on the right side) are known as *terminals*.

To generate a string we always start with a particular non-terminal and keep applying the rules till we get a string with no non-terminals left in it.

1.2 Grammars for propositional formulas

Definition 1.1 *Given a set of arbitrary symbols $\mathcal{P}$ called the set of* atomic propositions *or* atoms, *a* formula *of the propositional calculus is a string generated from the non-terminal $formula$ by the following grammar:*

$$\begin{array}{rcl} formula & ::= & p \quad \text{for any } p \in \mathcal{P} \\ formula & ::= & \neg\, formula \\ formula & ::= & formula\ op\ formula \\ op & ::= & \vee \mid \wedge \mid \Rightarrow \mid \equiv \mid \Leftarrow \end{array}$$

Each formula in the propositional calculus has a derivation from this grammar. In other words given a propositional formula A, there is a sequence of steps to get to it from $formula$:

$$(formula =)A_0 \to A_1 \to A_2 \to \cdots A_{n-1} \to A_n (= A)$$

Each step $A_i \to A_{i+1}$ is the expansion of an instance of either $formula$ or *op* according to one of the rules of the grammar. Let us call this sequence of steps the *derivation* of A.

Definition 1.2 *A formula B is said to be a* subformula *of A if there is a intermediate step A_i in the derivation of A with a single occurrence of $formula$ from which B is derived. Let us call this occurrence of $formula$ the* root symbol *for B. B is said to be a* proper subformula *of A if B is not the same as A.*

2 Boolean interpretations

In order to interpret propositional formulas in terms of boolean logic,we extend Definition 1.1 to include the following rule:

$$formula ::= \mathrm{T} \mid \mathrm{F}$$

A	$v(A_1)$	$v(A_2)$	$v(A)$
T			T
F			F
$\neg A_1$	T		F
$\neg A_2$	F		T
$A_1 \vee A_2$	F	F	F
$A_1 \vee A_2$	o.w.[1]	o.w.	T
$A_1 \wedge A_2$	T	T	T
$A_1 \wedge A_2$	o.w.	o.w.	F
$A_1 \Rightarrow A_2$	T	F	F
$A_1 \Rightarrow A_2$	o.w.	o.w.	T
$A_1 \Leftarrow A_2$	F	T	F
$A_1 \Leftarrow A_2$	o.w.	o.w.	T
$A_1 \equiv A_2$	T	T	T
$A_1 \equiv A_2$	F	F	T
$A_1 \equiv A_2$	o.w.	o.w.	F

Figure 1: Truth value assignment to formulas.

Definition 2.1 *Let A be a propositional formula and let $\{p_1, p_2, \ldots, p_n\}$ be the set of atoms appearing in A. An interpretation for A is a function $v : \{p_1, p_2, \ldots, p_n\} \to \{T, F\}$, that is v assigns* one of *the truth values T or F to each atom. Further, v assigns T or F to A according to the inductive definition in Figure 1.*

2.1 Logical equivalence and substitution

Definition 2.2 *Given two formulas A_1, A_2, if $v(A_1) = v(A_2)$ for all interpretations, then A_1 is* (logically) equivalent *to A_2, denoted $A_1 \leftrightarrow A_2$.*

Note that $\leftrightarrow$ is not a symbol of the propositional calculus and should never appear in any string which claims to be a propositional formula. It is just our shorthand for denoting logical equivalence. This logical equivalence is not the same as the equivalence denoted by $\equiv$ in the propositional. However, they are closely related:

Theorem 2.3 *$A_1 \leftrightarrow A_2$ if and only if $A_1 \equiv A_2$ evaluates to T in every interpretation.*

[1] otherwise

Now let us turn to substitution. Before proceeding recall Definition 1.2 where we talked about subformulas.

Definition 2.4 *If A is a subformula of B and A' is any formula, then B',the* substitution *of A' for A in B, denoted $B\{A \leftarrow A'\}$, is the formula obtained by deriving A' from the root symbol of A in the derivation of B.*

The following theorem holds for substitutions

Theorem 2.5 *Let A be a subformula of B and let A' be a formula such that $A \leftrightarrow A'$. Then $B \leftrightarrow B\{A \leftarrow A'\}$.*

The proof is by induction and is left as an exercise.

2.2 Satisfiability, validity and consequence

Definition 2.6 *A propositional formula A is* satisfiable *if its value is T in* some *interpretation. A satisfying interpretation is called a* model *for A. A is* valid *if its value is T in* all *interpretations. This is denoted $\models A$.*

Definition 2.7 *A propositional formula is* unsatisfiable *or* contradictory *if its value is F in* all *interpretations. A formula is* falsifiable *if it is not valid, i.e. its value is F in* some *interpretation.*

From these definitions we can conclude that all valid formulas are satisfiable although there may be satisfiable formulas which are not valid. And we have that:

Theorem 2.8 *A is valid if and only if $\neg A$ is unsatisfiable. A is satisfiable if and only if $\neg A$ is falsifiable.*

We extend the definition of satisfiability to sets of formulas as follows:

Definition 2.9 *A set of formulas $U = \{A_1, A_2, \ldots, A_n\}$ is* (mutually) satisfiable *if there exists an interpretation v (for all the atoms in U) such that $v(A_1) = v(A_2) = \cdots = v(A_n) = T$. The satisfying interpretation is called a* model *of U. U is unsatisfiable if for every interpretation there exists an i such that $v(A_i) = F$.*

Definition 2.10 *Let U be a set of formulas and A a formula. If A evaluates to T in every model of U, then A is a* logical consequence *of U, or A is logically implied by U, denoted $U \models A$.*

Note that A need not always be true. It only needs to be true in the interpretations which model U. Also, it is not necessary that every model of A is a model of U. As with $\leftrightarrow$, the symbol $\models$ is also not a symbol of the propositional calculus, but just of the meta-language we use to talk about it. It is however related to the symbol "$\Rightarrow$" by the following theorem:

Theorem 2.11 *If $U = \{A_1, \ldots, A_n\}$ the $U \models A$ if and only if $\models A_1 \wedge \cdots \wedge A_n \Rightarrow A$.*

Further we can show that:

Theorem 2.12 *If $U \models A$ then $U \cup \{B\} \models A$.*

Theorem 2.13 *If $U \models A$ and B is valid then $U \setminus \{B\} \models A$.*

The proof of these theorems is left as an exercise.

The notion of logical consequence is a central concept in the foundations of mathematics. Mathematics proceeds by making certain assumptions and then determining what are the logical consequences of those assumptions. The formal definition of a mathematical theory is:

Definition 2.14 *Let $\mathcal{T}(U) = \{A \mid U \models A\}$. $\mathcal{T}(U)$ is called the* theory *of U and the elements of $\mathcal{T}(U)$ are called the* theorems *of U. The elements of U are called the* axioms *of $\mathcal{T}(U)$.*

3 Deductive proofs: Hilbert systems

Instead of working with *semantic* concepts like satisfiability and validity, we present a purely syntactical method of arriving at new formulas. We start with a set of axioms and a set of rules for deducing new formulas from the axioms. The new formulas are called *theorems* and the description of the deduction is called the *proof.* A system such as this one is referred to as a *deductive proof system.* We will discuss a Hilbert-style proof system here which has many axioms and one rule.

3.1 Definition

Now let us define a Hilbert system $\mathcal{H}$ for the propositional calculus.

Definition 3.1 *For any formulas A, B, C, the following formulas are axioms in $\mathcal{H}$:*

$$\begin{array}{lr} \vdash (A \Rightarrow (B \Rightarrow A)) & \textit{Axiom 1} \\ \vdash (A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C)) & \textit{Axiom 2} \\ \vdash (\neg B \Rightarrow \neg A) \Rightarrow (A \Rightarrow B) & \textit{Axiom3} \end{array}$$

The notation $\vdash$ means that the formula is provable. Any axiom is provable (this can be verified explicitly.) Note that each of the axioms listed above are schematic i.e. each one of them gives rise to an infinite set of axioms by (uniformly) substituting some other formulas for A, B and C. For example

$$(p \vee q) \Rightarrow ((p \Rightarrow p) \Rightarrow (p \vee q))$$

is an instance of Axiom 1 where A is replaced by $p \vee q$ and B is replaced by $p \Rightarrow p$.

The axioms in themselves have limited use, so we add a rule of inference:

Definition 3.2 *For any formulas A, B, the rule of inference in $\mathcal{H}$ is*

$$\frac{\vdash A \qquad \vdash A \Rightarrow B}{\vdash B}$$

This rule is called *modus ponens* (Latin for "mode that affirms"), *MP* for short.

We now demonstrate the method of proof in the Hilbert system by proving that for any formula A, $\vdash A \Rightarrow A$.

Theorem 3.3 $\vdash A \Rightarrow A$.

Proof.

$$\begin{array}{rll} 1. & \vdash A \Rightarrow ((A \Rightarrow A) \Rightarrow A) \Rightarrow & \\ & ((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A)) & \text{Axiom 2} \\ 2. & \vdash A \Rightarrow ((A \Rightarrow A) \Rightarrow A) & \text{Axiom 1} \\ 3. & \vdash ((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A)) & \text{MP 1, 2} \\ 4. & \vdash A \Rightarrow (A \Rightarrow A) & \text{Axiom 1} \\ 5. & \vdash A \Rightarrow A & \text{MP 3, 4} \end{array}$$

□

Proofs like this which use only the axioms and modus ponens can sometimes be lengthy and tricky so we augment our armoury with a set of derived rules. For each derived rule we have to show that it is *sound* i.e. that a proof using this rule can be replaced by a proof using only the axioms and modus ponens. This prevents us from augmenting $\mathcal{H}$ with unprovable formulas.

We now state some derived rules. The proofs of the soundness of these are left as exercises.

Deduction rule

$$\frac{U \cup \{A\} \vdash B}{U \vdash A \Rightarrow B}$$

Contrapositive rule

$$\frac{\vdash \neg B \Rightarrow \neg A}{\vdash A \Rightarrow B}$$

Transitivity rule

$$\frac{U \vdash A \Rightarrow B \quad U \vdash B \Rightarrow C}{\vdash A \Rightarrow C}$$

Exchange of antecedent rule

$$\frac{U \vdash A \Rightarrow (B \Rightarrow C)}{U \vdash B \Rightarrow (A \Rightarrow C)}$$

Double negation rule

$$\frac{\vdash \neg\neg A}{A}$$

We conclude these notes by remarking that the Hilbert system is sound and complete. We state these facts formally:

Theorem 3.4 (Soundness) *If A is provable in the Hilbert system i.e. if $\vdash A$ then A is satisfiable i.e. $\models A$.*

Theorem 3.5 (Completeness) *If A is satisfiable i.e. if $\models A$ then A is provable in the Hilbert system i.e. $\vdash A$.*