Propositional Logic: Semantics and Bottom-Up Proofs

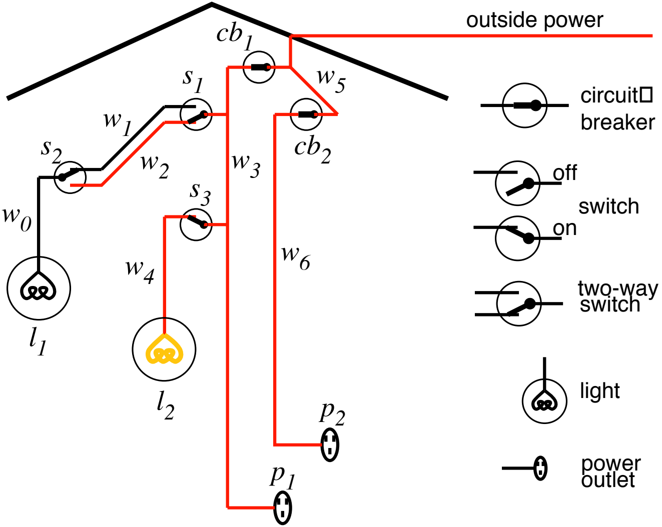
CPSC 322 Lecture 19

February 24, 2006
Textbook §4.2

Propositional Definite Clauses: Syntax

- An atom is a symbol starting with a lower case letter
- A body is an atom or is of the form $b_1 \land b_2$ where b_1 and b_2 are bodies.
- A definite clause is an atom or is a rule of the form $h \leftarrow b$ where h is an atom and b is a body.
  - read this as "h if b"
- A knowledge base is a set of definite clauses

Electrical Environment



Representing the Electrical Environment

$light\_l_1.$

$light\_l_2.$

$down\_s_1.$

$up\_s_2.$

$up\_s_3.$

$ok\_l_1.$

$ok\_l_2.$

$ok\_cb_1.$

$ok\_cb_2.$

$live\_outside.$

$live\_l_1 \leftarrow live\_w_0$

$live\_w_0 \leftarrow live\_w_1 \wedge up\_s_2.$

$live\_w_0 \leftarrow live\_w_2 \wedge down\_s_2.$

$live\_w_1, \leftarrow live\_w_3 \wedge up\_s_1.$

$live\_w_2 \leftarrow live\_w_3 \wedge down\_s_1.$

$live\_l_2 \leftarrow live\_w_4.$

$live\_w_4 \leftarrow live\_w_3 \wedge up\_s_3.$

$live\_p_1 \leftarrow live\_w_3.$

$live\_w_3 \leftarrow live\_w_5 \wedge ok\_cb_1.$

$live\_p_2 \leftarrow live\_w_6.$

$live\_w_6 \leftarrow live\_w_5 \wedge ok\_cb_2.$

$live\_w_5 \leftarrow live\_outside.$

Propositional Definite Clauses: Semantics

Semantics allows you to relate the symbols in the logic to the domain you're trying to model.

- An interpretation I assigns a truth value to each atom.

We can use the interpretation to determine the truth value of clauses and knowledge bases:

- A body $b_1 \wedge b_2$ is true in I if b_1 is true in I and b_2 is true in I.
- A rule $h \leftarrow b$ is false in I if b is true in I and h is false in I. The rule is true otherwise.
- A knowledge base KB is true in I if and only if every clause in KB is true in I.

Models and Logical Consequence

- A model of a set of clauses is an interpretation in which all the clauses are *true*.
- If KB is a set of clauses and g is a conjunction of atoms, g is a logical consequence of KB, written $KB \models g$, if g is *true* in every model of KB.
  - we also say that g logically follows from KB, or that KB entails g.
- In other words, $KB \models g$ if there is no interpretation in which KB is *true* and g is *false*.

Example: Models

$$KB = \begin{cases} p \leftarrow q. \\ q. \\ r \leftarrow s. \end{cases}$$

	p	q	r	s	
I_1	*true*	*true*	*true*	*true*	is a model of KB
I_2	*false*	*false*	*false*	*false*	not a model of KB
I_3	*true*	*true*	*false*	*false*	is a model of KB
I_4	*true*	*true*	*true*	*false*	is a model of KB
I_5	*true*	*true*	*false*	*true*	not a model of KB

Which of the following is true?

- $KB \models p$, $KB \models q$, $KB \models r$, $KB \models s$
- $KB \models p$, $KB \models q$, $KB \not\models r$, $KB \not\models s$

User's view of Semantics

1. Choose a task domain: intended interpretation.
2. Associate an atom with each proposition you want to represent.
3. Tell the system clauses that are true in the intended interpretation: axiomatizing the domain.
4. Ask questions about the intended interpretation.
5. If $KB \models g$, then g must be true in the intended interpretation.
6. The use can interpret the answer using their intended interpretation of the symbols.

Computer's view of semantics

- The computer doesn't have access to the intended interpretation.
- All it knows is the knowledge base.
- The computer can determine if a formula is a logical consequence of KB.
- If $KB \models g$ then g must be true in the intended interpretation.
- If $KB \not\models g$ then there is a model of KB in which g is false. This could be the intended interpretation.

Role of semantics

In user's mind:

- $l1\_broken$: light $l1$ is broken
- sw_up: switch is up
- $power$: there is power in the building
- $unlit\_l1$: light $l1$ isn't lit
- lit_l2: light $l2$ is lit

In Computer:

$$l1\_broken \leftarrow sw\_up \\ \wedge power \wedge unlit\_l1.$$

$$sw\_up.$$

$$power \leftarrow lit\_l2.$$

$$unlit\_l1.$$

$$lit\_l2.$$

Conclusion: $l1\_broken$

- The computer doesn't know the meaning of the symbols
- The user can interpret the symbols using their meaning

Proofs

- A proof is a mechanically derivable demonstration that a formula logically follows from a knowledge base.
- Given a proof procedure, $KB \vdash g$ means g can be derived from knowledge base KB.
- Recall $KB \models g$ means g is true in all models of KB.
- A proof procedure is sound if $KB \vdash g$ implies $KB \models g$.
- A proof procedure is complete if $KB \models g$ implies $KB \vdash g$.

Bottom-up Ground Proof Procedure

One rule of derivation, a generalized form of *modus ponens*:

If "$h \leftarrow b_1 \wedge \ldots \wedge b_m$" is a clause in the knowledge base, and each b_i has been derived, then h can be derived.

You are forward chaining on this clause.
(This rule also covers the case when $m = 0$.)

Bottom-up proof procedure

$KB \vdash g$ if $g \subseteq C$ at the end of this procedure:

$C := \{\};$
repeat
  select clause "$h \leftarrow b_1 \wedge \ldots \wedge b_m$" in KB such that
    $b_i \in C$ for all i, and $h \notin C$;
  $C := C \cup \{h\}$
until no more clauses can be selected.

Example

$a \leftarrow b \wedge c.$

$a \leftarrow e \wedge f.$

$b \leftarrow f \wedge k.$

$c \leftarrow e.$

$d \leftarrow k.$

$e.$

$f \leftarrow j \wedge e.$

$f \leftarrow c.$

$j \leftarrow c.$

$\{\}$

$\{e\}$

$\{c, e\}$

$\{c, e, f\}$

$\{c, e, f, j\}$

$\{a, c, e, f, j\}$

Soundness of bottom-up proof procedure

If $KB \vdash g$ then $KB \models g$.

- Suppose there is a g such that $KB \vdash g$ and $KB \not\models g$.
- Let h be the first atom added to C that's not true in every model of KB.
- Suppose h isn't *true* in model I of KB.
- There must be a clause in KB of form

$$h \leftarrow b_1 \wedge \ldots \wedge b_m$$

  Each b_i is true in I. h is false in I. So this clause is false in I.
- Therefore I isn't a model of KB. Contradiction: thus no such g exists.

Fixed Point

The C generated at the end of the bottom-up algorithm is called a fixed point.

- further applications of our rule of derivation will not change C.

Let I be the interpretation in which every element of the fixed point is true and every other atom is false.

Claim: I is a model of KB. Proof:

- Assume that I is not a model of KB. Then there must exist some clause $h \leftarrow b_1 \wedge \ldots \wedge b_m$ in KB (having zero or more b_i's) which is false in I.
- This can only occur when h is false and each b_i is true in I.
- But in this case it would have been possible to add h to C.
- Since C is a fixed point, no such I can exist.

I is called a Minimal Model.

Completeness

If $KB \models g$ then $KB \vdash g$.

- Suppose $KB \models g$. Then g is true in all models of KB.
- Thus g is true in the minimal model.
- Thus g is generated by the bottom up algorithm.
- Thus $KB \vdash g$.