Bottom Up: Soundness and Completeness

Computer Science cpsc322, Lecture 21

(Textbook Chpt 5.2)

June, 6, 2017

Lecture Overview

- Recap
- Soundness of Bottom-up Proofs
- Completeness of Bottom-up Proofs

(Propositional) Logic: Key ideas

Given a domain that can be represented with **n propositions** you have ________ interpretations (possible worlds)

2^n

If you do not know anything you can be in any of those

If you know that some **logical formulas** are true (your K.B.……). You know that you can be only in ________________ interpretations in which the KB is true (i.e. the models of KB)

It would be nice to know what else is true in all those…

models what else is logically entailed

PDCL syntax / semantics / proofs

Domain can be represented by three propositions: *p, q, r*

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

Interpretations?

r	q	p
T	T	T
~~T~~	~~T~~	~~F~~
~~T~~	~~F~~	~~T~~
~~T~~	~~F~~	~~F~~
~~F~~	~~T~~	~~T~~
~~F~~	~~T~~	~~F~~
~~F~~	~~F~~	~~T~~
~~F~~	~~F~~	~~F~~

Models?

What is logically entailed ?

r, q, p

Prove $G = \boxed{\cancel{(q \wedge p)}}$

$G \subseteq C$

$\Updownarrow$

$C = \{q, r, p\}$ $KB \vdash_{BU} G$

PDCL syntax / semantics / proofs

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

Interpretations

r	q	p
T	T	T
~~T~~	~~T~~	~~F~~
~~T~~	~~F~~	~~T~~
~~T~~	~~F~~	~~F~~
F	T	T
F	T	F
~~F~~	~~F~~	~~T~~
~~F~~	~~F~~	~~F~~

Models

What is logically entailed?

Prove $G = (q \wedge p)$

$C = \{q\}$

$G \notin C$

$KB \not\vdash_{BU} G$

Lecture Overview

- Recap
- Soundness of Bottom-up Proofs
- Completeness of Bottom-up Proofs

Soundness of bottom-up proof procedure

Generic Soundness of proof procedure:

If G can be proved by the procedure (KB ⊦ G) **then**
G is logically entailed by the KB (KB ⊧ G)

For Bottom-Up proof

if $G \subseteq C$ at the end of procedure
then G is logically entailed by the KB

So BU is sound, if all the atoms in ___C___

are logically entailed by the KB

Soundness of bottom-up proof procedure

Suppose this is not the case.

1. Let *h* be the first atom added to *C* that is not entailed by KB (i.e., that's not true in every model of *KB*)
2. Suppose *h* isn't true in model *M* of *KB*.
3. Since *h* was added to C, there must be a clause in KB of form: $h \leftarrow b_1 \wedge \ldots \wedge b_m$
4. Each b_i is true in *M* *(because of 1.)*. *h* is false in *M*. So…… the clause is false in M
5. Therefore M is not a model
6. Contradiction! thus no such *h* exists.

Lecture Overview

- Recap
- Soundness of Bottom-up Proofs
- Completeness of Bottom-up Proofs

Completeness of Bottom Up

Generic Completeness of proof procedure:

If G is logically entailed by the KB ($KB \models G$)

then G can be proved by the procedure ($KB \vdash G$)

$G \subseteq C$

Sketch of our proof:

1. Suppose $KB \models G$. Then G is true in all models of *KB*.
2. Thus *G* is true in any particular model of KB
3. We will define a model so that if G is true in that model, G is proved by the bottom up algorithm. $G \subseteq C$
4. Thus $KB \vdash_{BU} G$.

Let's work on step 3

3. We will define a model so that if G is true in that model, G is proved by the bottom up algorithm.

$G \subseteq C$

3.1 We will define an interpretation $\mathcal{I}$ so that if G is true in $\mathcal{I}$, G is proved by the bottom up algorithm.

$G \subseteq C$

3.2 We will then show that ……… $\mathcal{I}$ is a model

Let's work on step 3.1

3.1 Define interpretation I so that if G is true in I,

Then $G \subseteq C$.

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

$a \leftarrow e \wedge g.$
$b \leftarrow f \wedge g.$
$c \leftarrow e.$
$f \leftarrow c \wedge e.$
$e.$
$d.$

F	F	T	T	T	T	F
{ a,	b,	c,	d,	e,	f,	g }

C

{ }
{ e }
{ e, d }
{ e, d, c }
{ e, d, c, f }

Let's work on step 3.2

Claim: I is a model of *KB* *(we'll call it the* ***minimal model****)*.

Proof: **Assume** that I is not a model of *KB*.

- **Then** there must exist some clause $h \leftarrow b_1 \land \cdots \land b_m$ in *KB* (having zero or more b_i's) which is false in I.
- The only way this can occur is if $b_1 \cdots b_m$ are true in I (i.e., are in C) and h is false in I (i.e., is not in C)
- **But** if each b_i belonged to C, Bottom Up would have added h to C as well.
- So, there can be no clause in the KB that is false in interpretation I (which implies the claim :–)

Completeness of Bottom Up
(proof summary)

If $KB \models G$ then KB $\vdash_{BU} G$

- Suppose $KB \models G$.
- Then G is true in all the models
- Thus G is true in the minimal model
- Thus $G \subseteq C$
- Thus G is proved by….. BU
- i.e., $KB \vdash_{BU} G$

$$\begin{tikzcd}
KB \models_{BU} G \arrow[rr, bend right, "\text{Completeness}"'] & & KB \vdash_{BU} G \arrow[ll, bend right, "\text{Soundness}"']
\end{tikzcd}$$

RELATION BETWEEN SOUNDNESS & COMPLETENESS

An exercise for you

BU $C = \{d, e, c, f\}$

Let's consider these two alternative proof procedures for PDCL

KB

$a \leftarrow e \wedge g.$

$b \leftarrow f \wedge g.$

$c \leftarrow e.$

$f \leftarrow c$

$e.$

$d.$

X. C_X = {All clauses in KB with empty bodies}

$= \{e, d\}$

Y. C_Y = {All atoms in the knowledge base}

$\{e\ d\ f\ c\ \overset{*}{g}\ \overset{*}{a}\}$

i>clicker

A. Both X and Y are sound and complete

B. Both X and Y are neither sound nor complete

C. X is sound only and Y is complete only $\Leftarrow$

D. X is complete only and Y is sound only

An exercise for you

$BU\ C = \{d, e, c, f\}$

Let's consider these two alternative proof procedures for PDCL

A. C_A = {All clauses in KB with empty bodies}

$= \{e, d\}$

B. C_B = {All atoms in the knowledge base}

$\{e\ d\ f\ c\ \overset{*}{g}\ \overset{*}{a}\}$

KB

$a \leftarrow e \wedge g.$

$b \leftarrow f \wedge g.$

$c \leftarrow e.$

$f \leftarrow c$

$e.$

$d.$

A is sound only and B is complete only

Learning Goals for today' s class

You can:

- Prove that BU proof procedure is sound

- Prove that BU proof procedure is complete

Next class

Midterm, this Thur, June 8,
First 50mins block
we will start at 1pm sharp !

- Using PDC Logic to model the electrical domain
- Reasoning in the electrical domain
- Top-down proof procedure (as Search!)
- Datalog