Lecture 14

Horn formula H is CNF in which each block has at most one positive literal.

H ≡ to one which is in *implicational form*

a conjunction of wff each of which has one of the following three forms:

(1) $\underline{t} \rightarrow p$.

(2) $p_1 \wedge \ldots \wedge p_n \rightarrow q$.

(3) $p_1 \wedge \ldots \wedge p_n \rightarrow \underline{f}$

<u>Example</u> Let

$$H = (p \vee \neg q) \wedge (\neg c \vee \neg p \vee q) \wedge$$

$$(\neg s \vee \neg r) \wedge d.$$

Write H in implicational form.

$$p \vee \neg q \equiv \neg q \vee p \equiv q \to p$$

$$\neg c \vee \neg p \vee q \equiv \neg(c \wedge p) \vee q \equiv c \wedge p \to q$$

$$\neg s \vee \neg r \equiv \neg(s \wedge r) \equiv s \wedge r \to \underline{f}$$

$$d \equiv \neg t \vee d \equiv \underline{t} \to d$$

$$\therefore H = (q \to p) \wedge (c \wedge p \to q) \wedge (s \wedge r \to \underline{f}) \wedge (\underline{t} \to d)$$

We shall now describe a fast algorithm for deciding whether a Horn formula is satisfiable.

Let $H = H_1 \wedge \ldots \wedge H_n$.

For H to be satisfiable, each of the $H_1, \ldots, H_n$ has to be satisfiable for the same truth assignment. We now look at the different shapes the H_i can have.

We try to find the weakest conditions on the H_i that makes them true.

Either this works and we get a truth assignment that satisfies H

or it fails and we have shown that H is a contradiction.

- $H_i = \underline{t} \to p$. The only way for this to be true is for p to be true. We assign p the truth value T and mark <u>all</u> occurrences of p as follows $\dot{p}$.

- $H_i = \dot{p}_1 \wedge \ldots \wedge \dot{p}_n \to q$.
  The only way for this to be true is for q to be true. So, we make all occurrences of q as follows $\dot{q}$.

- $H_i = \dot{p}_1 \wedge \ldots \wedge \dot{p}_n \to \underline{f}$.
  If this occurs, we can stop because H is a contradiction.

Example Determine whether the following is satisfiable or not.

$$H = (r \wedge s \to u) \wedge (p \to q) \wedge (q \to r) \wedge (r \to s) \wedge \boxed{(\underline{t} \to r)} \wedge (p \wedge q \to \underline{f})$$

First mark r and all occurrences of r.

$$H = (\dot{r} \wedge s \to u) \wedge (p \to q) \wedge (q \to \dot{r}) \wedge \boxed{(\dot{r} \to s)} \wedge (\underline{t} \to \dot{r}) \wedge (p \wedge q \to \underline{f})$$

Now mark all occurrences of s

$$H = \boxed{(\dot{r} \wedge \dot{s} \to u)} \wedge (p \to q) \wedge (q \to \dot{r}) \wedge (\dot{r} \to \dot{s}) \wedge (\underline{t} \to \dot{r}) \wedge (p \wedge q \to \underline{f})$$

Now mark all occurrences of u

$$H = (\dot{r} \wedge \dot{s} \to \dot{u}) \wedge (p \to q) \wedge (q \to \dot{r}) \wedge (\dot{r} \to \dot{s}) \wedge (\underline{t} \to \dot{r}) \wedge (p \wedge q \to \underline{f})$$

At this point, there are no further operations we can apply and no contradiction has been found. Thus H is satisfiable.

p	q	r	s	u
F	F	T	T	T

You can check ~~if the abstract tableau generator tree this really does work — or~~ directly. directly that this is a satisfying truth assignment.

Observe that the default truth value setting all atoms = F. Only the atoms marked with a dot (·) ~~A~~ are given the value T.

Example

$$H = (p \wedge q \rightarrow \underline{f}) \wedge (\underline{t} \rightarrow p) \wedge (\underline{t} \rightarrow q)$$

H is a contradiction.

Example

$$(p \wedge q \wedge s \rightarrow \underline{f}) \wedge (q \wedge r \rightarrow \underline{f}) \wedge (s \rightarrow \underline{f})$$

No marking is applied. This is satisfiable.

p	q	r	s
F	F	F	F

Algorithm

Input Horn formula $H = H_1 \wedge \dots \wedge H_n$ in implicational form.

Procedure If there are no occurrences of formulae of the shape $\underline{t} \to p$ then H is satisfiable when all atoms are assigned the truth value F.

- If there are occurrences of formulae of the shape $\underline{t} \to p$ mark all occurrences of p by $\dot{p}$. Do this for all such formulae.
- For any formula of the shape $\underbrace{\dot{p}_1 \wedge \dots \wedge \dot{p}_n}_{\text{all marked}} \to q$ mark q and repeat.
- If any formula of the shape $\dot{p}_1 \wedge \dots \wedge \dot{p}_n \to \underline{f}$ occurs then STOP and say that H is a contradiction.
- Else, once the marking process can no longer be applied, assign the truth value T to all marked atoms and F else. This is a satisfying truth assignment.

Input: Horn formula H in implicational form.
$H = H_1 \wedge \ldots \wedge H_n$

Why this algorithm works

- Assume all atoms are initially assigned the value F. Only change the truth value of an atom to T if it is marked in the course of the algorithm.

- If no wff of the form $\underline{t} \to p$ occurs then the algorithm terminates with no marking having occurred. All atoms will take the initial truth assigned of F and all H_i will take the truth value T.

- Assume that a wff of the form $\underline{t} \to p$. Then the algorithm will mark atoms. These atoms must have the value T if H is to be satisfiable. If $\dot{p}_1 \wedge \ldots \wedge \dot{p}_n \to \underline{f}$ then H is a contradiction.

Suppose that $\dot{p}_1 \wedge \ldots \wedge \dot{p}_n \to \underline{t}$ does not occur by the conclusion of the algorithm. Then the claim is that the updated truth assignment will satisfy H. Look at the different shapes H_i can have.

(1) $\underline{t} \to p$. Then p is assigned the value T by the algorithm and $\underline{t} \to p$ is T.

(2) $p_1 \wedge \ldots \wedge p_n \to q$. Two possibilities.

- $\dot{p}_1 \wedge \ldots \wedge \dot{p}_n \to \ddot{q}$ in which case $p_1, \ldots, p_n, q$ all assigned the value T.

- $p_1 \wedge \ldots \wedge p_n \to q$ not all p_i are marked so at least one is F so LHS is F and RHS is F and $F \to F$ is T

(3) $p_1 \wedge \ldots \wedge p_n \to \underline{f}$.

Note all $p_1, \dots, p_n$ are marked such

leaves are F & $F \to F = T$. ∎

<u>Example</u> Find a satisfying truth assignment of

$$H = (q \to p) \wedge (c \wedge p \to q) \wedge (s \wedge r \to \underline{f}) \wedge (\underline{t} \to d)$$

<u>algorithm</u>

$$H = (q \to p) \wedge (c \wedge p \to q) \wedge (s \wedge r \to \underline{f}) \wedge (\underline{t} \to d)$$

only d is marked. So, a satisfying truth assignment is

c	d	p	q	r	s
F	T	F	F	F	F

Truth Table

c	d	p	q	r	S	$(p \lor \neg q) \land (\neg c \lor \neg p \lor q) \land (\neg S \lor \neg r) \land d$
T	T	T	T	T	T	F
T	T	T	T	T	F	T
T	T	T	T	F	T	T
T	T	T	T	F	F	T
T	T	T	F	T	T	F
T	T	T	F	T	F	F
T	T	T	F	F	T	F
T	T	T	F	F	F	F
T	T	F	T	T	T	F
T	T	F	T	T	F	F
T	T	F	T	F	T	F
T	T	F	T	F	F	F
T	T	F	F	T	T	F
T	T	F	F	T	F	T
T	T	F	F	F	T	T
T	T	F	F	F	F	T
T	F	T	T	T	T	F
T	F	T	T	T	F	F
T	F	T	T	F	T	F
T	F	T	T	F	F	F
T	F	T	F	T	T	F
T	F	T	F	T	F	F
T	F	T	F	F	T	F
T	F	T	F	F	F	F
T	F	F	T	T	T	F
T	F	F	T	T	F	F
T	F	F	T	F	T	F
T	F	F	T	F	F	F
T	F	F	F	T	T	F
T	F	F	F	T	F	F
T	F	F	F	F	T	F
T	F	F	F	F	F	F

c	d	p	q	r	s	
F	T	T	T	T	T	F
F	T	T	T	T	F	T
F	T	T	T	F	T	T
F	T	T	T	F	F	T
F	T	T	F	T	T	F
F	T	T	F	T	F	T
F	T	T	F	F	T	T
F	T	T	F	F	F	T
F	T	F	T	T	T	F
F	T	F	T	T	F	F
F	T	F	T	F	T	F
F	T	F	T	F	F	F
F	T	F	F	T	T	F
F	T	F	F	T	F	T
F	T	F	F	F	T	T
F	T	F	F	F	F	T
F	F	T	T	T	T	F
F	F	T	T	T	F	F
F	F	T	T	F	T	F
F	F	T	T	F	F	F
F	F	T	F	T	T	F
F	F	T	F	T	F	F
F	F	T	F	F	T	F
F	F	T	F	F	F	F
F	F	F	T	T	T	F
F	F	F	T	T	F	F
F	F	F	T	F	T	F
F	F	F	T	F	F	F
F	F	F	F	T	T	F
F	F	F	F	T	F	F
F	F	F	F	F	T	F
F	F	F	F	F	F	F

* This is what the algorithm finds

The fast algorithm for Horn formulae always finds the first satisfying truth assignment from the bottom of the truth table.