Horn Formula

$B ::= \bot \mid \top \mid p$

$A ::= p \mid p \wedge A$

$C ::= A \rightarrow B$

$H ::= C \mid C \wedge H.$

Each instance of C is a Horn clause.

A Horn formula is the conjunction of Horn clauses.

Example.

1. $(p_1 \wedge p_2 \wedge \cdots \wedge p_n) \rightarrow q$

2. $(p \rightarrow q) \wedge (s \wedge t \rightarrow \top) \wedge (p \wedge s \wedge t \rightarrow \bot)$

3. $p_1 \wedge p_2 \wedge p_3 \rightarrow \underline{\neg p_4}$ - Not a Horn formula.

4. $p_1 \wedge \underline{\neg p_2} \rightarrow p_3$ - Not a Horn formula.

$p \equiv \top \rightarrow p$; $\neg p \equiv p \rightarrow \bot$

Note: Horn Satisfiability is a P-complete problem.

Claim. if α is a Horn formula, deciding if α is satisfiable can be solved efficiently (linear time)

Procedure - A marking algorithm.

Algorithm.

Input: A Horn formula α.

Output: Decision whether α is satisfiable.

1. Mark all occurence of $\top$ in α.

2. while there is a conjunct $p_1 \wedge p_2 \wedge \cdots \wedge p_k \rightarrow p'$ of α such that all p_j are marked but p' is not marked.

    - mark p'.

   end while

3. if $\perp$ is marked then return "Unsatisfiable"
   else return "Satisfiable"

Algorithm.

Input: A Horn formula α.

Output: Decision whether α is satisfiable.

1. Mark all occurence of $\top$ in α.

2. while there is a conjunct $P_1 \wedge P_2 \wedge \cdots \wedge P_k \rightarrow P'$ of α
   such that all P_j are marked but P' is not marked.

   - mark P'.

   end while

3. if $\perp$ is marked then return "Unsatisfiable"
   else return "Satisfiable"

Termination: if there are n propositions in α then the while loop is executed atmost $n+1$ times.

Algorithm.

Input: A Horn formula α.

Output: Decision whether α is satisfiable.

1. Mark all occurence of $\top$ in α.

2. while there is a conjunct $p_1 \wedge p_2 \wedge \cdots \wedge p_k \rightarrow p'$ of α such that all p_j are marked but p' is not marked.

   - mark p'.

   end while

3. if $\perp$ is marked then return "Unsatisfiable" else return "satisfiable"

Correctness. By induction on the number of iterations of the while loop, argue that For all v where $v \models \alpha$, for all marked p, $v \models p$.

Base Case: All occurences of $\top$ are marked - nothing else is marked

Induction step. Consider the iteration $k+1$.
$\exists\ p_1 \wedge p_2 \wedge \cdots \wedge p_m \rightarrow p'$ of α s.t p_j is marked $\forall j \in \{1, \ldots m\}$.

Consider any v s.t $v \models \alpha$, By IH, $v \models p_j\ \forall j \in \{1, \ldots m\}$

if $v \models \alpha$ then $v \models p_1 \wedge p_2 \wedge \cdots \wedge p_m \rightarrow p'$. $\therefore v \models p'$.

Algorithm.

Input: A Horn formula α.

Output: Decision whether α is satisfiable.

1. Mark all occurence of $\top$ in α.

2. while there is a conjunct $p_1 \wedge p_2 \wedge \cdots \wedge p_k \rightarrow p'$ of α such that all p_j are marked but p' is not marked.

   - mark p'.

   end while

3. if $\bot$ is marked then return "Unsatisfiable" else return "Satisfiable"

Correctness.

- if $\bot$ is marked, then there exists:

$$\underbrace{p_1 \wedge p_2 \wedge \cdots \wedge p_m \rightarrow \bot}_{v \not\models \;\leftarrow}$$ where p_j is marked $\forall j$.

for all v where $v \models p_j \;\; \forall j$.

- if $\bot$ is not marked, consider valuation v defined as:
  - assign $\top$ to all marked propositions
  - assign $\bot$ to all unmarked propositions

Suppose $v \not\models \alpha \Rightarrow \exists\, \beta = p_1 \wedge p_2 \wedge \cdots \wedge p_m \rightarrow p'$ s.t $v \not\models \beta$

$\Rightarrow v \models p_j \; \forall j$ and $v \not\models p'$ (Semantics of $\rightarrow$)

$\therefore$ For all p_j, p_j is marked. $\Rightarrow p'$ is also marked.

Since $\bot$ is not marked, $p' = \top$ or some proposition.

$\Rightarrow v \models \beta$. A contradiction!