Lecture 5

- HW 1
- Project proposals.

How SAT solvers work

Unit propagation

Example: Consider a clause $(\bar{x} \vee \bar{y} \vee z)$

& say our partial assignment is

$\{x := \text{true}, \; z := \text{false}\}$.

Then, if the clause is to be satisfied,

it has to be the case that $y := \text{false}$

$\bar{y} := \text{true}$.

Solving Horn SAT & 2-SAT in linear time

Yifei's claim for Horn SAT

If all clauses are definite Horn clauses, then the formula is trivially satisfiable

Set all variables to true.

"Definite clause" = "Exactly one positive literal"

If all clauses are definite,
& I set all variables to true,
then formula is trivially satisfied.

Ex: $(x \vee \bar{y} \vee \bar{z})$ and (y) and $(z \vee \bar{y})$

Every clause has a positive literal
By assumption, that is satisfied
So the clause is satisfied. So, the whole formula is sat

"Goal clause" = "Clause with no positive literals"

$$(\bar{x} \vee \bar{y} \vee \bar{z}) \equiv \overline{(x \text{ and } y \text{ and } z)}$$

"Too many variables cannot simultaneously be true" $\equiv$ $\boxed{x \text{ and } y \text{ and } z \Rightarrow \text{false}}$

$\boxed{\text{(Sometimes) By definition, } \bar{x} = x \Rightarrow \text{false.}}$

In other words, at least some variables must be false.

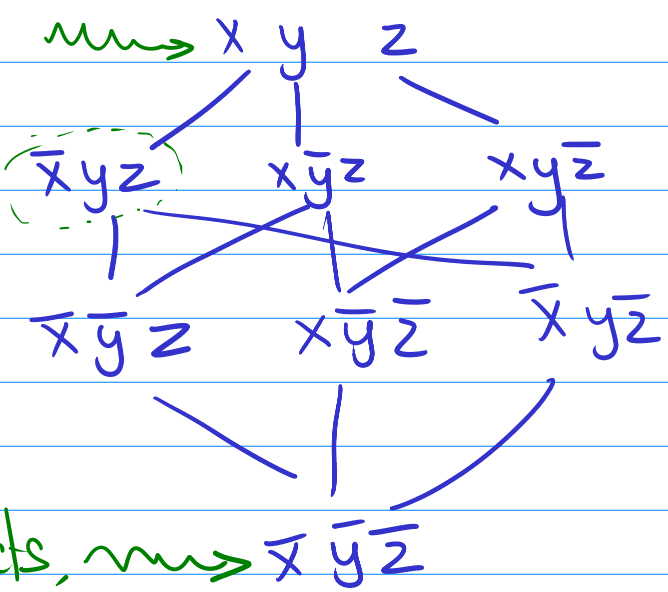
Sojhal's observation

If there are no facts, then the formula is once again trivially satisfiable.

Set all variables to false.

Lattice of assignments

If no goal clauses, this works



⟵ All vars true

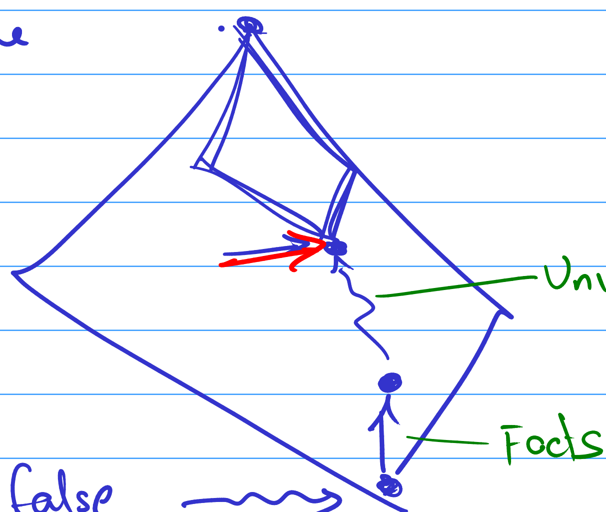
⟵ Two vars true other one is false

⟵ One variable true remaning false

If no facts, this works

⟵ All vars false

All true



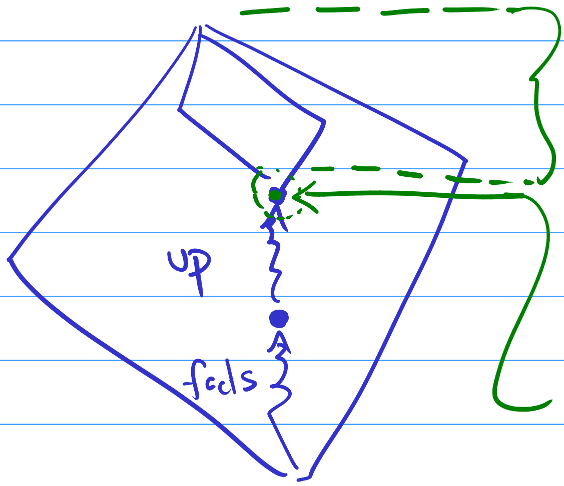
All false

Ex: a and $(a \Rightarrow b)$
and c
and $(a \wedge c \wedge b \Rightarrow d)$
and $(\bar{a} \text{ or } \bar{e})$

Ignore all goals for a moment

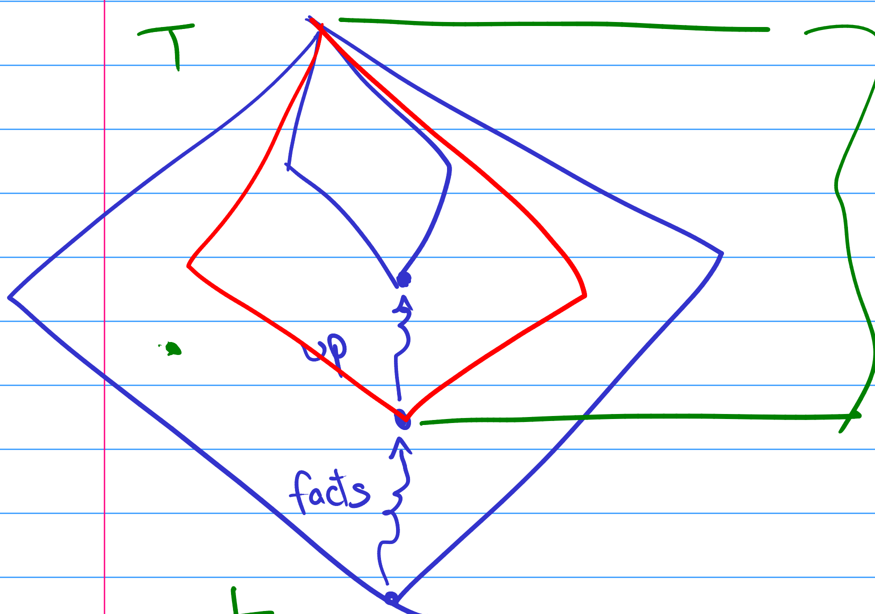
Defn: A fact is a definite clause with no negative literals.

Equivalently A fact is a clause of the form x, for some atomic prop x.



Claim 1: If the formula is satisfiable, then all satisfying assignments in this sublattice

Claim 2: If the formula is satisfiable, then this assignment satisfies the formula.

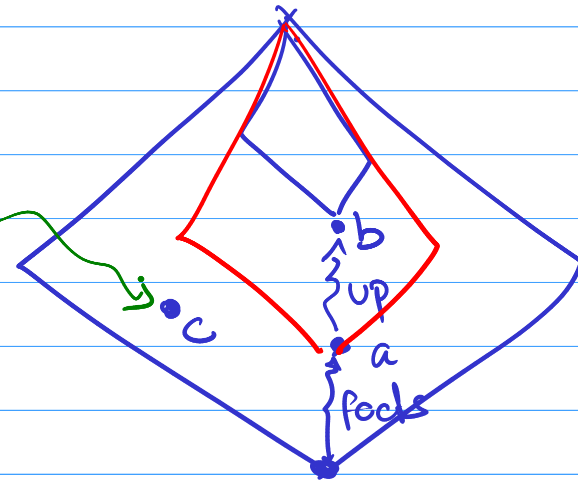


Claim 1': If the formula is satisfiable then all satisfying assignments live here.

Proof of claim 1: Assume not.

Then, there exists a satisfying assignment c such that

$$\underbrace{a \not\leq c}$$



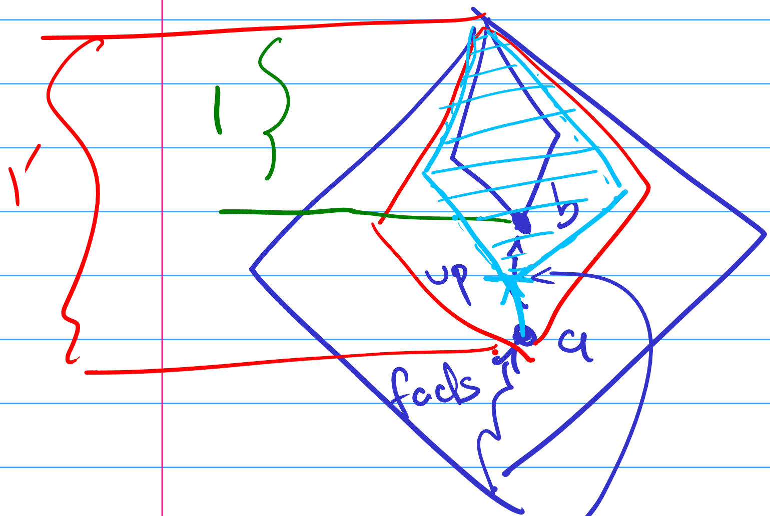
In other words, there is some variable x such that $\underbrace{a_x \text{ is true}}$ but c_x is false

There is a clause x

Then, how can c satisfy the formula?

Contradiction.

Proof of claim 1: Watch unit propagation as it executes



— In each step, UP picks up a Horn clause

$$x \wedge y \wedge z \Rightarrow w$$

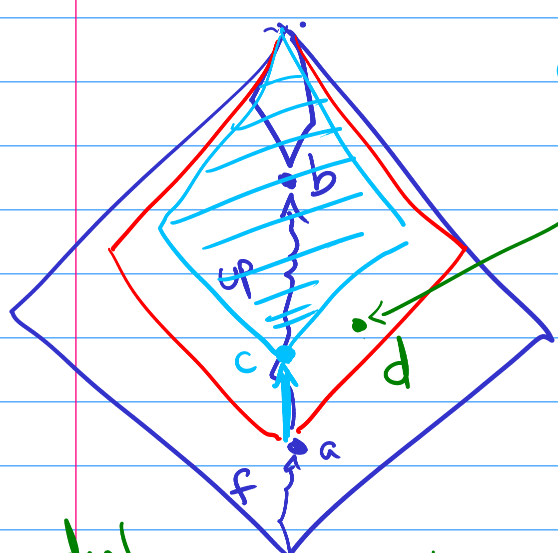
s.t. $x = \text{true}$ $y := \text{true}$ $z = \text{true}$

$a + \{w := \text{true}\}$ — Sets w to true

Claim: All satisfying assignments live inside the light blue sublattice, $a + \{w := \text{true}\}$

Proof: Assume not.

$c = a + \{w := \text{true}\}$



Imagine, for the sake of contradiction, that d satisfied the formula.

Since $c \not\leq d$, it has to be the case that $d_w = \text{false}$

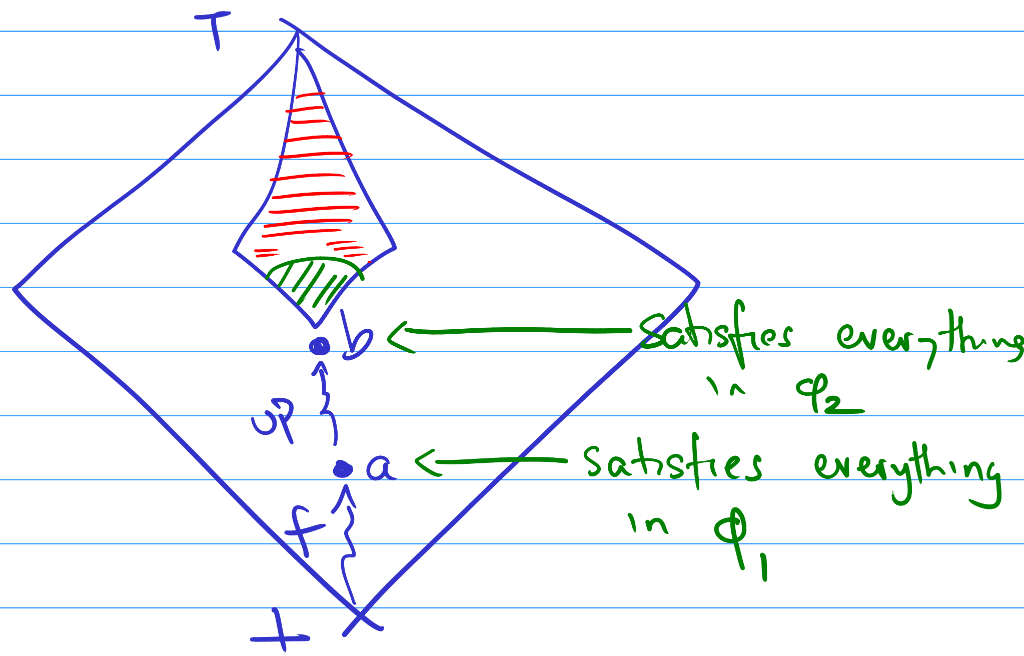
Since $a \leq d$, it has to be the case that $d_x = d_y = d_z = \text{true}$

Contradiction

Then, how can d satisfy the clause $x \wedge y \wedge z \Rightarrow w$?

Claim 2

If the formula is satisfiable, then b satisfies the formula.



$$\varphi = \begin{cases} \varphi_1 & x \text{ and } y \text{ and } w \text{ and } u \quad \Big] \text{ facts} \\ \text{and } \varphi_2 & \text{and } (x \Rightarrow y) \text{ and } (x \wedge y \Rightarrow v) \text{ and } (u \wedge v \wedge w \Rightarrow r) \quad \Big] \text{ non factual definite clauses} \\ \text{and } \varphi_3 & \text{and } (\overline{x} \vee \overline{y} \vee \overline{v}) \text{ and } (\overline{x} \vee \overline{u}) \quad \Big] \text{ goals} \end{cases}$$

"At least a few things must be false."

Ex $\varphi = \varphi_1 \Big[$ x and y and w and u $\Big]$ facts

and $\varphi_2 \Big[$ and $(x \Rightarrow y)$ and $(x \wedge y \Rightarrow v)$ and $(u \wedge v \wedge w \Rightarrow r)$ and $(p \Rightarrow r)$ $\Big]$ non factual definite clauses

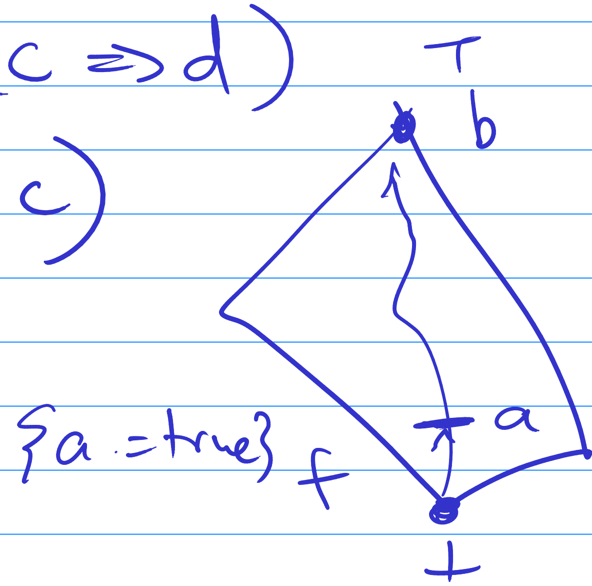
and $\varphi_3 \Big[$ and $(\overline{x} \vee \overline{y} \vee \overline{v})$ and $(\overline{x} \vee \overline{u})$ and $(p \vee \overline{r})$ $\Big]$ goals

"At least a few things must be false."

Ex: a and $(a \Rightarrow b)$ and $(c \Rightarrow d)$

and $(a \wedge b \Rightarrow c)$

and $\overline{c}$



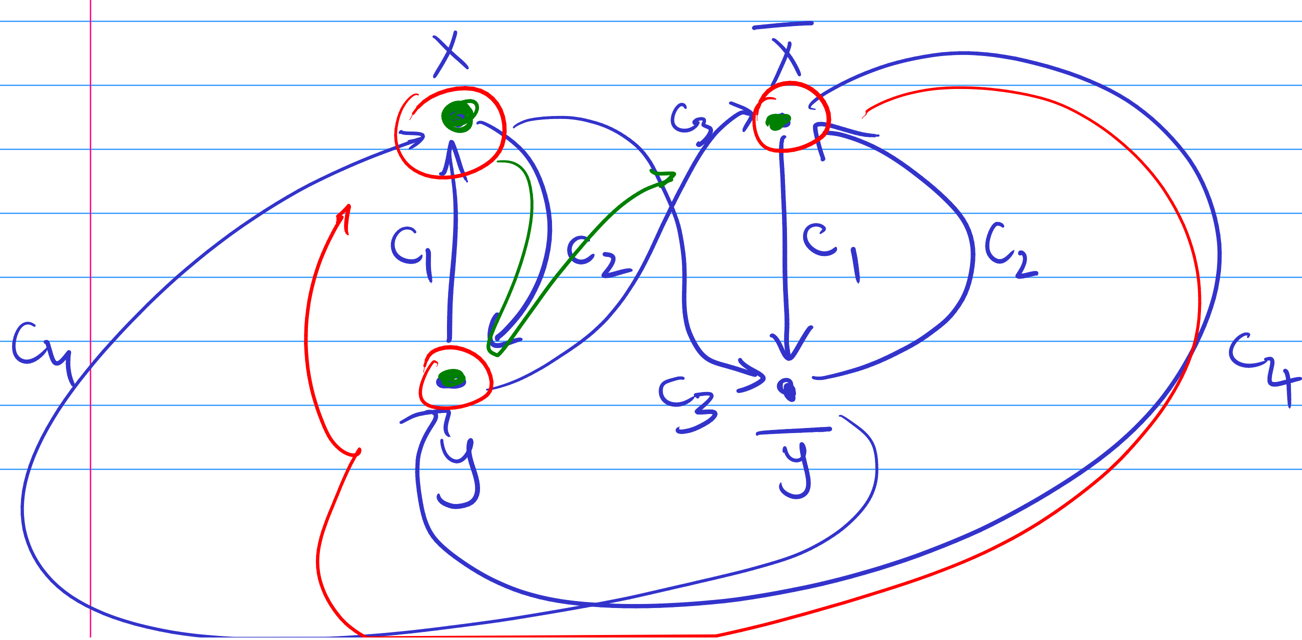
Checking 2-SAT in polynomial time

"2-CNF" = "Every clause has at most 2 literals"

Ex: $(\bar{x} \vee \bar{y})$ and $(\bar{w} \vee \bar{x})$ and $(\bar{y} \vee \bar{w})$

Ex: $(x \vee \bar{y})$ and $(\bar{x} \vee y)$ and $(\bar{x} \vee \bar{y})$ and $(x \vee y)$

$\underbrace{(x \vee \bar{y})}_{C_1}$ and $\underbrace{(\bar{x} \vee y)}_{C_2}$ and $\underbrace{(\bar{x} \vee \bar{y})}_{C_3}$ and $\underbrace{(x \vee y)}_{C_4}$



Algorithm for 2SAT

Input: 2-CNF formula φ

1. Construct the implication graph G_φ
2. If there is a variable x s.t x & $\bar{x}$ are in the same SCC then φ is unsat.
3. Otherwise, φ is sat.