More NP-Complete Problems

Frank

March 17, 2022

1 CNF-SAT

Fact: CNF-SAT $\in$ NPC

Note that a CNF has the form $F = C_1 \wedge C_2 \wedge \ldots \wedge C_m$, where each C_i is a "clause"
and $C_i = l_i^1 \vee l_i^2 \vee \ldots \vee l_i^{k_i}$, where each $l_i^{k_i}$ is a "literal"
and $l = x$ or $\neg x$, where x is a variable

For example $F = (x_1 \vee \neg x_2 \vee x_3) \wedge x_2 \wedge (x_1 \vee \neg x_3)$

Note that every propositional formula has an equivalent CNF form

1.1 3-SAT

Very useful for reductions since it has a lot of "structure"

SAT problem restricted to CNF form with ≤ 3 literals for any clause.

Note that we can also use "exactly 3" here, but it doesn't make a difference because if a clause has less than 3 literals we can simply repeat literals until we have 3 literals.

Thm 9.1: 3-SAT $\in$ NPC

Proof:

- 3-SAT $\in$ NP (trivial, certificate is satisfying assignment)
- We will show CNF-SAT $\leq_m^P$ 3-SAT

  Given CNF formula $F = C_1 \wedge C_2 \wedge \ldots \wedge C_m$

  Construct (in polytime in $|F|$) a CNF formula F' with at most 3 literals per clause s.t. F is satisfiable $\iff$ F' is satisfiable

  If C_j has ≤ 3 literals then $C'_j = C_j$
  If $C_j = l_1 \vee l_2 \vee l_3 \vee \ldots \vee l_k$ ex. $k \geq 4$
  then treat $l_3 \vee \ldots \vee l_k$ as a single formula X
  Claim: $C_j =_{LEQV} (z_1 \vee l_1 \vee l_2) \wedge (\neg z_1 \vee X)$
  $=_{LEQV} (\neg z_1 \rightarrow (l_1 \vee l_2)) \wedge (z_1 \rightarrow X)$
  Given that C_j is true
  If z_1 is true, then X must be true, then one of $l_3, \ldots, l_k$ has to be true
  If z_1 is false, then either l_1 or l_2 must be true
  Now, $(z_1 \vee l_1 \vee l_2)$ has 3 literals, and $(\neg z_1 \vee X)$ has $k - 2 + 1 = k - 1$ literals
  Since we have reduced C_j to a conjunction of a 3 literal clause and a $k-1$ literal clause, we can repeat this process until we have a "chunk" of conjunction of many 3 literal clauses

$C_j =_{LEQV} (z_1 \vee l_1 \vee l_2) \wedge (\neg z_1 \vee l_3 \vee z_2) \wedge (\neg z_2 \vee l_4 \vee z_3) \wedge \ldots \wedge (\neg z_{k-3} \vee l_{k-1} \vee l_k)$, which we can assign as C'_j

Note that $|C'_j| \leq 3 \cdot |C_j|$ which is polynomial in size of C_j

Then $F' = C'_1 \wedge C'_2 \wedge \ldots \wedge C'_m$

Now, since every $C'_j =_{LEQV} C_j$, we have that F satisfiable $\iff F'$ is satisfiable

In addition, F' is constructed in polytime in $|F|$ ($|C'_j| \leq 3 \cdot |C_j|$), thus our reduction is polytime

2 Independent Set

Thm 9.2: IS $\in$ NPC

Proof: IS $\in$ NP (trivial) and 3-SAT $\leq_m^P$ IS (will prove this)

Given a 3-CNF formula F, construct (in polytime in $|F|$) a graph $G = (V, E)$ and $b \in \mathbb{N}$ s.t. F is satisfiable $\iff G$ has IS of b nodes.

Let $F = C_1 \wedge \ldots \wedge C_m$, vars $= x_1, \ldots, x_n$, $C_j = (l_j^1 \vee l_j^2 \vee l_j^3)$

For example

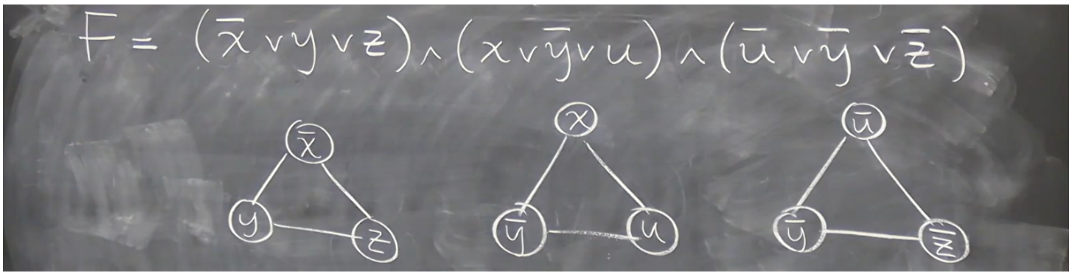

Figure 1: s31

An important observation is that if F is satisfiable, then every clause in F must be true (characteristic of CNF). Thus a satisfying truth assignment must satisfy at least 1 of the 3 literals in every clause. We can simply pick one of the satisfied literals in each clause to be in our IS.

Now, if given a graph, we want to construct the graph such that we create an edge between every literal and its negation (because logically this is not allowed, we cannot satisfy both x and $\bar{x}$)

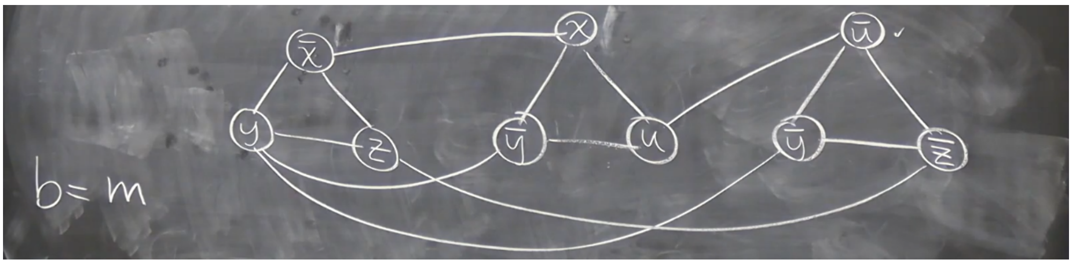

Figure 2: s32

Given arbitary F, construct $G = (V, E)$ as follows

$V = \{N_j^t : 1 \leq j \leq m, 1 \leq t \leq 3\}$

$E = \{\{N_j^1, N_j^2\}, \{N_j^2, N_j^3\}, \{N_j^3, N_j^1\} : 1 \leq j \leq m\}$ (triangle edges)

$\cup\{\{N_j^t, N_{j'}^{t'}\} : 1 \leq j \neq j' \leq m, 1 \leq t, t' \leq 3,$ and $l_j^t, l_{j'}^{t'}$ are opposite literals$\}$ (inter-triangle edges)

$b = m$

Verify that the constructed graph is polynomial in size of input

$|V| = 3m$

$|E| \leq (3m)^2$

$\implies |\langle G = (V, E), b\rangle|$ is poly in $|F|$

$\implies \langle G, b\rangle$ can be constructed from F in polytime

Claim: F is sat $\iff$ G has IS of size $b = m$

$(\implies)$

F is sat $\implies$ let τ be a t.a. that satisfies F

$\implies \forall j, 1 \leq j \leq m, \tau$ satisfies $C_j = (l_j^1 \vee l_j^2 \vee l_j^3)$

Let $l_j^{t_j}$ be s.t. $\tau(l_j^{t_j}) = 1$

Let $\{N_1^{t_1}, N_2^{t_2}, \ldots, N_m^{t_m}\}$, which selects a literal satisfied by τ from each clause. Note that since we are selecting 1 literal per clause, we do not have any triangle edges between any of these nodes. Moreover, we do not have any inter-triangle edges between these nodes because if we do, that would mean a pair of opposite literals are both satisfied at the same time, which obviously cannot be.

Then the set $\{N_1^{t_1}, N_2^{t_2}, \ldots, N_m^{t_m}\}$ is an IS of size m

$(\impliedby)$

G has an IS V' of size $b = m$

V' must be of form $\{N_1^{t_1}, N_2^{t_2}, \ldots, N_m^{t_m}\}$

Let

$$\tau(x) = \begin{cases} 1 & \text{if } \exists j \text{ s.t. } l_j^{t_j} = x \\ 0 & \text{if } \exists j \text{ s.t. } l_j^{t_j} = \bar{x} \\ 0 \text{ or } 1 & \text{otherwise} \end{cases}$$

τ is well-defined and satisfies F

3 More Problems

3.1 CLIQUE

CLIQUE

Instance: $\langle G, b\rangle$

Q: Does G have a clique (set of nodes where every pair of nodes is connected by an edge) of size b

Observation: If $V' \subseteq V$ is a clique of G, then V' is an IS of $\bar{G} = (V, \bar{E})$, where $\bar{E}$ are edges <u>not</u> in E

This observation gives an obvious reduction IS $\leq_m^P$ CLIQUE

Thm 9.3: CLIQUE $\in NPC$

3.2 VERTEX COVER

VERTEX COVER

> Instance: $\langle G, b\rangle$
>
> Q: Does G have a VC (set of nodes s.t. every edge has at least 1 endpoint in one of these nodes) of size b?

Observation: If $V' \subseteq V$ is a v.c. of G, then $V - V'$ is an IS of G (If not, then V' wouldn't cover all edges)

Which gives clear reduction IS $\leq_m^P$ VC

Thm 9.4: VC $\in$ NPC

3.3 So far...

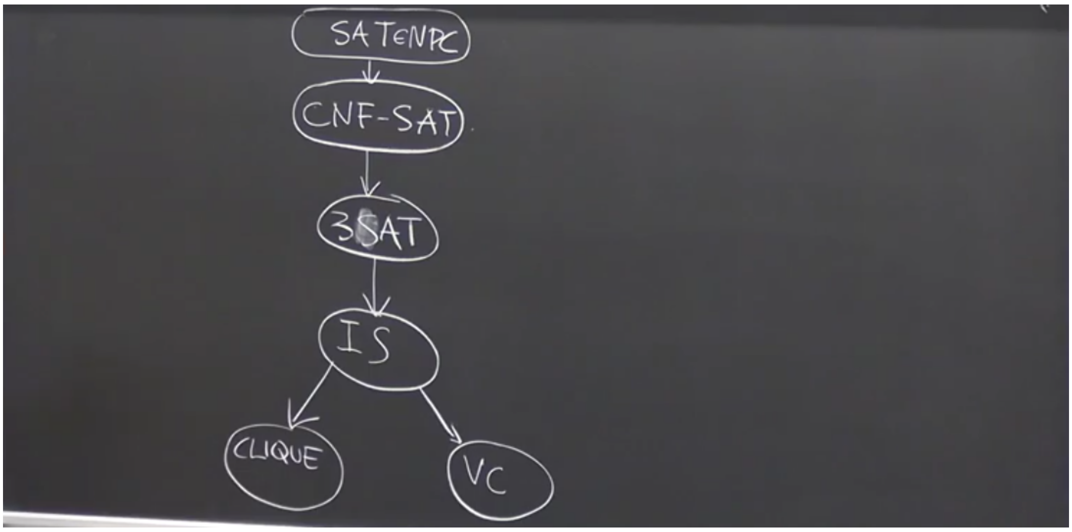

Figure 3: s33

Now, we need to show that SAT is NPC, to justify the other proofs.

4 Cook-Levin Theorem

Thm 9.5: SAT $\in$ NPC

Proof:

1. SAT $\in$ NP (trivial, use correct truth assignment as certificate)
2. $\forall L \in$ NP, $L \leq_m^P$ SAT (We will focus on this)

Fix arbitrary $L \subseteq \Sigma^\star$, in NP

$\implies \exists$ NTM M_L that decides L in polytime

Let $p(n)$ = running time of M_L (maximum height of computation path)

For any $x \in \Sigma^\star$, we construct (in polytime) formula F_x s.t.
$x \in L \iff F_x$ is satisfiable

$x \in L \implies M_L$ on x has an accepting computation path

> M_L on input x has a computation path $C_0 \vdash C_1 \vdash C_2 \vdash \ldots \vdash C_l$ where $l \leq p(|x|)$ and $C_0 = q_0 x$ and $C_l = y q_A z$
>
> $M_L = (Q, \Sigma, \Gamma, \delta, q_0, q_A, q_R)$ is a constant variable that depends on L, so once we fix L, we fix M_L

Visualize a branch of the computation of M_L on x as follows

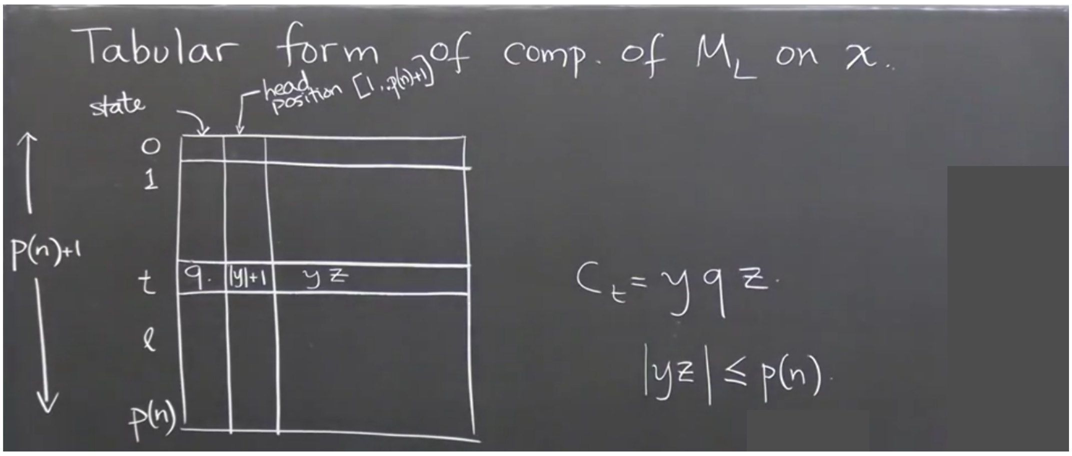

Figure 4: s34

Above is a different way of representing $C_0 \vdash C_1 \vdash C_2 \vdash \ldots \vdash C_l$, where each row represents a computation

Now introduce some <u>propositional vars</u>:

- S_t^q, $\forall t \in [0..p(n)], \forall q \in Q$

$$S_t^q = \begin{cases} 1 & \text{if } C_t \text{ is in state } q \\ 0 & \text{otherwise} \end{cases}$$

  We have $\Theta(p(n))$ of variable of this kind
- H_t^i, $\forall t \in [0..p(n)], \forall i \in [1..p(n)+1]$

$$H_t^i = \begin{cases} 1 & \text{if in } C_t \text{ head is on cell } i \\ 0 & \text{otherwise} \end{cases}$$

  We have $\Theta(p^2(n))$ of variable of this kind
- C_t^{ia}, $\forall t \in [0..p(n)], \forall i \in [1..p(n)+1], \forall a \in \Gamma$

$$C_t^{ia} = \begin{cases} 1 & \text{if in } C_t \text{ cell } i \text{ has } a \\ 0 & \text{otherwise} \end{cases}$$

  We have $\Theta(p^2(n))$ of variable of this kind

Thus we have in total $\Theta(p^2(n))$ variables

At a high level, there will be 4 groups of variables

1. COHERENCE: At every time t (config t):
   (a) M_L is not in two states
   (b) head is not in two places
   (c) each cell does not contain two symbols
2. STARTS-WELL: Row 0 represents q_0x
3. ENDS-WELL: Row $p(n)$ has state q_A
4. MOVE-WELL: In moving from time t to $t+1$:
   (a) only the cell where the head is can change symbols
   (b) new state, symbol in that cell, and new position of the head is as per δ

Now lets define these groups

1. (a) We have $\neg(S_t^p \wedge S_t^q)$, $\forall t \in [0..p(n)]$, $\forall p \neq q \in Q$
   equivalently, $\neg S_t^p \vee \neg S_t^q$
   (b) We have $\neg H_t^i \vee \neg H_t^j$, $\forall t \in [0..p(n)]$, $\forall i \neq j \in [1..p(n)+1]$
   (c) We have $\neg C_t^{ia} \vee \neg C_t^{ib}$, $\forall t \in [0..p(n)]$, $\forall i \in [1..p(n)+1]$, $\forall a \neq b \in \Gamma$
2. We have $S_0^{q_0} \wedge H_0^1 \wedge C_0^{1a_1} \wedge C_0^{2a_2} \wedge \ldots \wedge C_0^{na_n} \wedge \left(\bigwedge_{n+1 \leq i \leq p(n)} C_0^{i\sqcup} \right)$ s.t. $a_1a_2 \ldots a_n = x$
3. We have $S_{p(n)}^{q_A}$
4. (a) $(C_t^{ia} \wedge \neg C_{t+1}^{ia}) \rightarrow H_t^i$, $\forall t$, $\forall i$, $\forall a$ (if content changed, then head must have been at cell i at time t)
   equivalently $\neg C_t^{ia} \vee C_{t+1}^{ia} \vee H_t^i$

(b) Note our transition function is now non-deterministic, meaning it has form $\delta(q,a) \ni (p,b,D)$ (a superset)

We have $(S_t^q \wedge H_t^i \wedge C_t^{ia}) \to \bigvee_{(p,b,D)\in\delta(q,a)} (S_{t+1}^p \wedge H_{t+1}^{i+d} \wedge C_{t+1}^{ib})$, where

$$d = \begin{cases} 1 & \text{if } D = R \\ -1 & \text{if } D = L \text{ and } i > 1 \\ 0 & \text{otherwise} \end{cases}$$

$\forall t$, $\forall q - \{q_A, q_R\}$, $\forall i$, $\forall a$

and $\forall t$, $\forall q \in \{q_A, q_R\}$, $\forall i$, $\forall a$, we have $(S_t^q \wedge H_t^i \wedge C_t^{ia}) \to S_{t+1}^q \wedge H_{t+1}^i \wedge C_{t+1}^{ia}$

(repeat until we reach $p(n)$ steps)

For a DNF form (slightly closer to CNF)

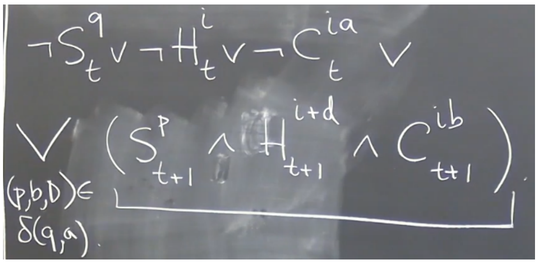

Figure 5: s35

Let F_x^1 = conjunction of type 1 formulas

Let F_x^2 = conjunction of type 2 formulas, and F_x^3, F_x^4

Then $F_x = F_x^1 \wedge F_x^2 \wedge F_x^3 \wedge F_x^4$

But how big is F_x?

1a, we have $\Theta(p(n))$ literals ($|Q|$ is const)

1b, we have $\Theta(p^3(n))$ literals

1c, we have $\Theta(p^3(n))$ literals ($|\Gamma|$ is const)

$\implies |F_x^1| = \Theta(p^3(n))$

$|F_x^2| = \Theta(p(n))$

$|F_x^3| = \Theta(1)$

$|F_x^4| = \Theta(p^2(n))$

$\implies |F_x| = \Theta(p^3(n))$ (note that $p(n)$ is poly by def)

Now, we must show M_L on x has an accepting computation $\iff F_x$ is satisfiable

$(\impliedby)$

Assume M_L on x has accepting comp. $C_0 \vdash C_1 \vdash \ldots \vdash C_l$ where $l \le p(n)$.

Rewrite this in tabular form

This defines t.a. τ that satisfies F_x

For example:

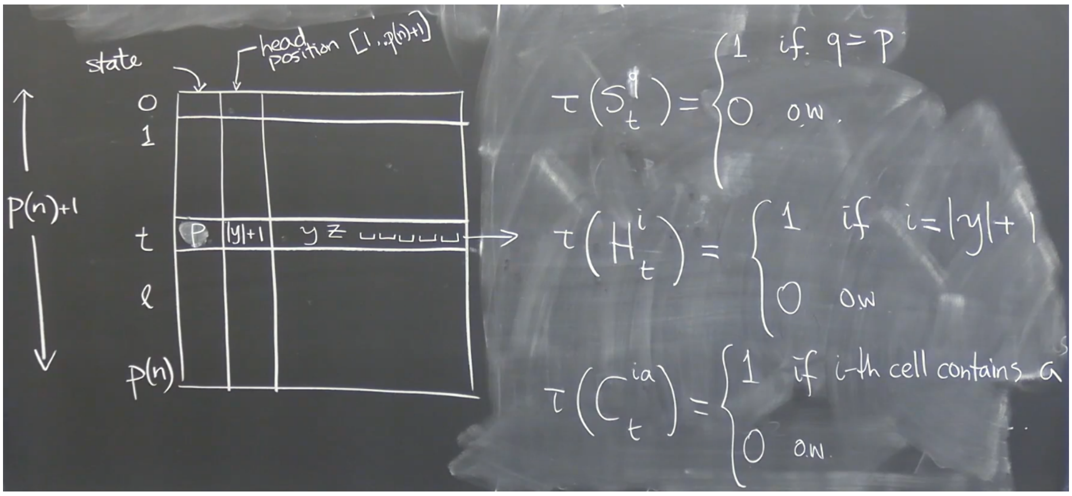

Figure 6: s36

Our t.a. τ is defined to be valid when M_L has an accepting computation.

$(\implies)$

If F_x is satisfiable, let τ be a t.a. that satisfies it.

Use τ to define table and use table to define an accepting comp of M_L on x

This ofcourse, is a sketch of the proof, omitting details.

Some details regarding 4b CND form and its size

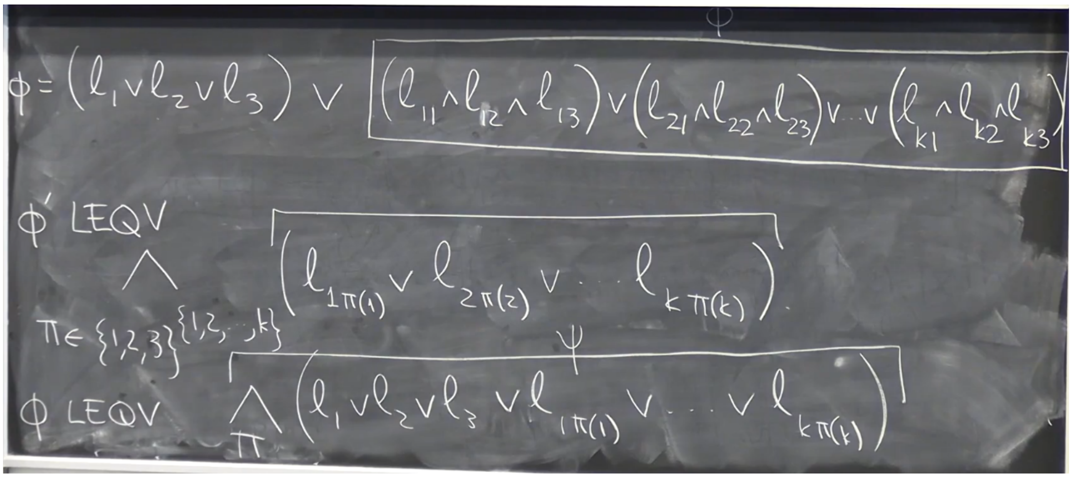

Figure 7: s37

$|\phi| = 3(k+1)$

$|\psi| = 3^k(k+3)$

But k = number of possibilities of NTM and is bounded by $|Q| \cdot |\Gamma| \cdot 2 = \Theta(1)$, independent on input (instance of L, which excludes the actual NTM).