Normal Forms of Boolean Formulas

Conjunction : $\phi_1 \wedge \phi_2$ AND

Disjunction : $\phi_1 \vee \phi_2$

Conjunctive Normal Form

CNF : Conjunction of disjunctions

$$(x_1 \vee x_2 \vee \sim x_{10}) \wedge (\sim x_5 \vee x_9 \vee x_{15} \vee \sim x_{22}) \wedge \cdots$$

(Clauses)

DNF : Disjunction of conjunctions

$$(x_1 \wedge x_2 \wedge \sim x_{15}) \vee (x_{13} \wedge x_{22} \wedge \sim x_{17}) \cdots$$

We assume negations are ALWAYS at literals.

(By de Morgan's law)

Formula ϕ is represented as a TREE with internal nodes $\vee$, $\wedge$ and leaves labeled with literals.

Def: Two formulas are equivalent if they have the same evaluation on all settings of variables.

Any ϕ can be converted to CNF (or DNF) by distributive law. ↳ could make size exponential

$$\text{CNF-SAT} = \{\langle\phi\rangle \mid \phi \text{ is CNF \& satisfiable}\} \in \text{NP-complete}$$

$$\text{DNF-SAT} = \{\langle\phi\rangle \mid \phi \text{ is DNF \& satisfiable}\} \in \text{P}$$

$$k\text{-SAT} = \{\langle\phi\rangle \mid \phi \text{ is } k\text{-CNF \& satisfiable}\}$$

↳ clauses have EXACTLY k literals

Thm: 3-SAT is NP-complete

Thm: 2-SAT $\in$ P

↳ **Proof**: 3SAT $\in$ NP.

We will show CNFSAT $\leq_{p(n)}$ 3SAT

$\langle\phi\rangle$ is CNF

Clause: $x_1 \vee x_2 - x_{k-1} \vee x_k \Rightarrow (x_1 \vee x_2 \vee y_1) \wedge (x_3 \vee \sim y_1 \vee y_2) \wedge (x_4 \vee \sim y_2 \vee y_3) \cdots (x_{k-1} \vee x_k \vee \sim y_{k-2})$

Exercise: Any satisfying assignment to RHS sets some $x_i = 1$

$x_1 \vee x_2 \Rightarrow (x_1 \vee x_2 \vee y) \wedge (x_1 \vee x_2 \vee \sim y)$

$x_1 \Rightarrow (x_1 \vee y \vee z) \wedge (x_1 \vee y \vee \sim z) \wedge (x_1 \vee \sim y \vee \sim z) \wedge (x_1 \vee \sim y \vee z)$

CNF ϕ can be converted in linear time to 3-CNF ψ

s.t. ϕ is satisfiable iff ψ is satisfiable.

~~3S~~ CNFSAT $\leq_{p(n)}$ 3SAT

NP-complete problem

Solving system of linear equations $\in P$

quadratic $\in$ NP-hard

Solving linear system of INTEGER/BINARY equations is NP-hard

Convert

~~Act~~ 3SAT as a system of linear equations with binary variables

$x_1^2 = x_1 \equiv x_1 \in \{0,1\}$

$$\text{co-}\mathcal{C} = \{ \overline{\mathcal{L}} \mid \mathcal{L} \in \mathcal{C} \}$$

↑ class of languages

$$\text{co-NP} = \{ \overline{\mathcal{L}} \mid \mathcal{L} \in \text{NP} \}$$

$$\text{TAUT} = \{ \langle \phi \rangle \mid \phi \text{ is always true} \}$$

$$\text{co-P} = \text{P}$$

Def: $\mathcal{L} \in$ co-NP if $\exists$ polynomial p and a poly time TM M

s.t. $\forall\, x \in \{0,1\}^*$

If $x \notin \mathcal{L}$, $\exists$ certificate y of length $\leq p(|x|)$

s.t. $M(x,y)$ ~~accepts~~ rejects

If $x \in \mathcal{L}$, $\forall$ cert. $y \ldots$, $M(x,y)$ accepts

---

Def: $\mathcal{L}$ is (co-NP)-complete if

(1) $\mathcal{L} \in$ co-NP & (2) $\forall\, \mathcal{L}' \in$ co-NP, $\mathcal{L}' \leq_{p(n)} \mathcal{L}$

Clm: (co-NP)-complete is the same as co-(NP-complete).

NP vs co-NP

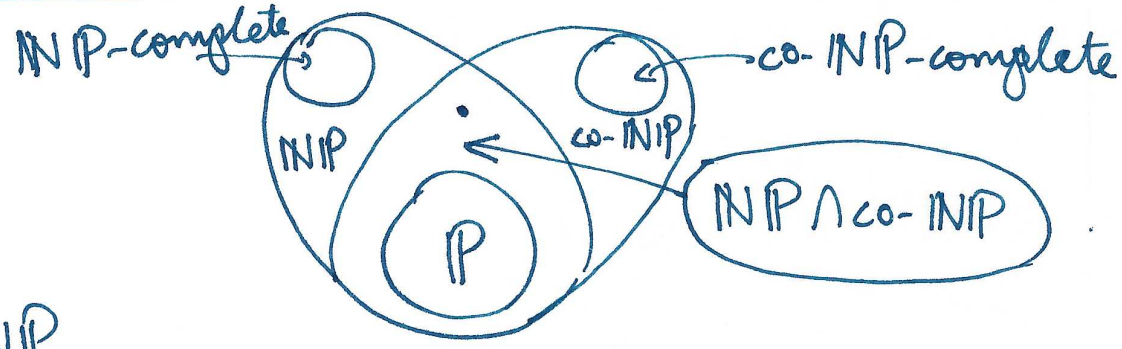
Is $NP = co\text{-}NP$? (We think no.)

Clm: If $NP \neq co\text{-}NP$, $P \neq NP$

($P = co\text{-}P$)

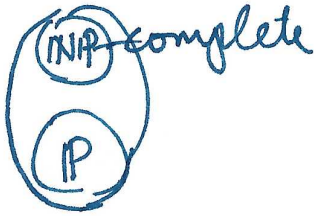
Clm: If $SAT \in co\text{-}NP$, $NP = co\text{-}NP$.

(SAT is NP-complete)

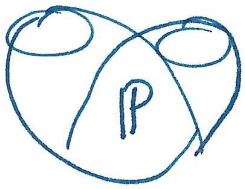


$P = NP$

$NP = co\text{-}NP$, $P \neq NP$



Is $NP \cap co\text{-}NP = P$



FACTORING $= \{ \langle N, k \rangle \mid N$ has a non-trivial factor less than $k \}$

n = size $= \log_2 N$

FACTORING $\in$ NP

The certificate is the non-trivial factor $< k$.

This number can be represented is $\leq \log_2 N$ bits.

The certificate can be checked by division (poly time)

FACTORING $\in$ co-NP (N does NOT have factor $< k$)

The certificate is the prime factorization where all primes are $\geq k$.

Encoding of $p_1^{r_1} p_2^{r_2} \cdots p_k^{r_k} = N$

($k \leq \log_2 N$, all these numbers are $\leq N$.

So encoding has polynomial size)

We need to certify that all p_is are prime.

Can be done in polynomial time.

FACTORING $\in$ NP $\cap$ co-NP.

Clm: If FACTORING is NP-complete, NP = co-NP.

EXP & NEXP

$$EXP = \bigcup_{c \geq 1} DTIME(2^{n^c})$$

$$NEXP = \bigcup_{c \geq 1} NTIME(2^{n^c})$$

Clm: $NP \subseteq EXP$

(Try all certificates)

Thm. If $EXP \neq NEXP$, then $P \neq NP$.

Proof: Padding argument

We prove $P = NP \Rightarrow EXP = NEXP$

Let $L \in NEXP$ $L \in NTIME(2^{n^c})$ $(c \geq 1)$

There is a NTM M deciding L that runs in 2^{n^c} time.

$L_{pad} = \{ \langle x, 1^{2^{|x|^c}} \rangle \mid x \in L \}$