Lecture 16
Computational Complexity
Instructor: Prahladh Harsha

**Today**
- #P
- #P-completeness
- Valiant's Theorem (Permanent is #P-complete)

---

**#P- Counting Class.**

$f : \{0,1\}^* \to \mathbb{N}$ } - complexity of such functions

eg: #SAT : $\varphi \mapsto$ #satisfying assignment

#ST: $G \mapsto$ #spanning trees of graph

Perm: $M$ (w/ $\{0,1\}$-entries) $\mapsto$ permanent $(M)$

#Cycle: $G \mapsto$ #cycles

#Accepting Paths: $(M, w) \mapsto$ #accepting paths

Non-deterministic TM

ie $\#\{y \mid M(w;y) = 1\}$

Path length: $(G, s, t) \mapsto d_G(s,t)$

**Easy Class:**

FP: $f \in$ FP if $\exists$ a deterministic poly time TM "M" s.t $\forall x \in \{0,1\}^*$
$f(x) = M(x)$ (ie output of M is x)

#P: (sharp-P, number-P)

$f \in \#P$ if $\exists$ a polynomial $p$ & a poly TM $M$ s.t

$$f(x) = \#\{w \in \{0,1\}^{p(|x|)} \mid M(x,w) = 1\}$$

$\boxtimes$

① Matrix-tree Theorem / Kirchoff's Thm.

G - undirected graph.

$$L(G) = \mathrm{Diag}(\deg) - \mathrm{Adj}(G)$$

$$= \begin{bmatrix} d_1 & & \\ & \ddots & \\ & & d_n \end{bmatrix} - \mathrm{Adj}(G)$$

$$\#\text{spanning trees}(G) = \det(L_{1,1})$$

Corollary: $(\#ST \in FP)$

② # perfect matchings in a planar graph.

Fisher-Kestelyn-Tempotley:

For every planar graph G, there exists a $\pm$-signing of the edges such that

$$\#\text{perfect matchings} = \sqrt{\det(\text{signed-matrix})}$$

Furthermore, this signing can be obtained efficiently.

---

#SAT, #Hamiltonian Cycles $\in$ #P
- don't believe them to be in FP

- #P - natural defn

Eg: 1 Networks:
G - undirected
every edge can fail w/ prob ½

$$\Pr[G \text{ is connected}] = \frac{\#\text{spanning graphs of } G}{2^{|E|}}$$

2 Machine learning: Hidden Variable Model

$X_1 \dots X_n$ - Hidden Variables

$y_i = \varphi_i(x_1 \dots x_n)$

$Y_1 \dots Y_m$ - Observables

Know: $\varphi_1 \dots \varphi_m$ & observe the $\{y_i\}$'

$\Downarrow$

Estimate the hidden vars

ie, $Pr[x_1=1, x_2=0, \ldots x_n=1 / y_1=1, y_2=0, \ldots y_m=1]$

$= ??$

$$Pr[x_1=1 / y_1=1 \wedge y_2=1 \ldots \wedge y_m=1]$$

$$= \frac{\#\text{sat assgn to } \varphi|_{x_1=1}}{\#\text{sat assgn to } \varphi} \qquad \varphi = \varphi_1 \wedge \varphi_2 \ldots \wedge \varphi_n$$

Obs:

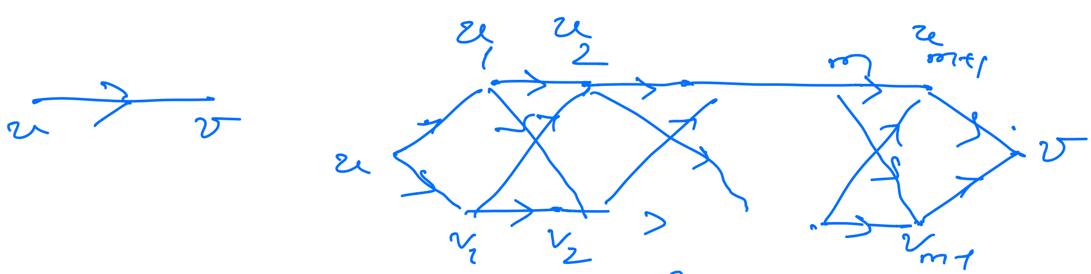
Decision problem is hard, so is counting.

Remark:

Decision might be easy, yet counting might be hard.

Thm: If $\#CYCLE \in FP$ (directed), then $NP = P$

Pf: If $\#CYCLE \in FP$, then Hamiltonian Cycle $\in P$




1 cycle of length n $\longrightarrow$ $(2^m)^n$ - cycles

$m = n \log n$

1 cycle of length $n$ $\longrightarrow$ $(2^m)^n$ - cycles

$n^{n^2}$ - cycles.

All cycles of length $\leq n-1$ $\longrightarrow$ #cycles $< n^{n^2}$

Conclusion: Counting can be much harder than decision.

PP: $L \in PP$ if $\exists$ a polynomial time TM & a poly $p$ s.t

$$x \in L \iff \#\{w \in \{0,1\}^{p(|x|)} \mid M(x,w)=1\} \geq \frac{1}{2} \cdot 2^{p(|x|)}.$$

Lemma: $PP = P \iff \#P = FP$.

Pf: $(\Leftarrow)$ trivial from defn

$(\Rightarrow)$: PP - msb of the count

"Alg compute msb $\Rightarrow$ Alg that computes every bit"

Claim 1: For every 2 TMs $M_0$ & $M_1$, there exists another TM $\dot{M}$ s.t

$\#acc(M,x) = \#$acc paths of m/c $M$ on i/p $x$

$\#acc(M,x) = \#acc(M_0,x) + \#acc(M_1,x)$.

M/C $M$: On input $x = (b,\omega)$

If $b=0$, accept if $M_0(x,\omega)=1$

else accept if $M_1(x,\omega)=1$.

---

<u>Claim</u>: $\forall$ constant $N$, $\exists$ a m/c $M$

$\forall x$, $\#acc(M,x) = N$

---

Alg A that computes msb of count

- we can perform binary search (using above claims) to find the exact count.

---

<u>#P- complete problems</u>

If there exists an alg for #SAT

$\Downarrow$

$\exists$ an alg for every $f \in \#P$

<u>Pf:</u> Cook-Levin Theorem (Reduction is parsimonious)

ie, the #satisfying assignments is preserved.

---

## #P-completeness:

$f$ - #P-complete
- (i) $f \in \#P$
- (ii) #P-hard: $\forall g \in \#P, \quad g \in FP^{f}$

**Thm:** #SAT is #P-complete

(in fact, for any NP-complete problem via parsimonious redns, the corresponding counting problem #L - #P-complete.)

**Thm:** Permanent of $\{0,1\}$-matrices is #P-complete
[Valiant]

$$A = (A_{ij})_{i,j=1}^{n}$$

$$\mathrm{per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} A_{i,\sigma(i)}$$

(like $\det(A)$ but w/o $(-1)^{\mathrm{sign}(\sigma)}$)

## Combinatorial Views of Permanent:

(1) $A$ - adjacency matrix of a bipartite graph $(L, R, E)$

$|L| = |R| = n$

$$A = \begin{array}{cc} & R \\ L & \begin{bmatrix} & & \\ & 1 & \\ & & \end{bmatrix} \end{array} \rightarrow (u,v) \in E$$

$\text{per}(A) = \#$ perfect matchings.

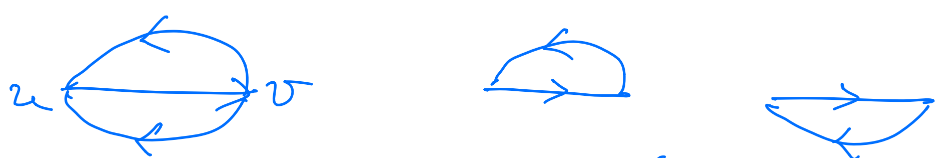
**Cor:** #perfect matchings in bipartite graph is #P-complete

---

(2) #Cycle Covers:

Directed graph :

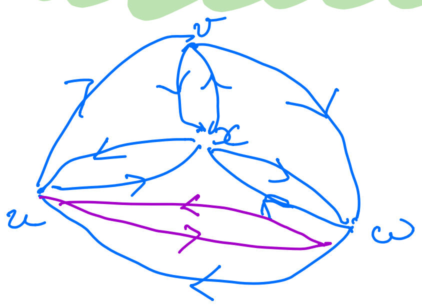
Cycle Cover = Vertex disjoint union of cycles that covers all vertices.




2 cycle covers.

$A$ - adj matrix of a directed graph $G$

$\text{per}(A) = \#$cycle covers of $G$.




Weighted graph

$\mathcal{C}$ - cycle cover $\mathcal{C} = C_1 \dots C_r$

$$wt(\mathcal{C}) = \prod_{\substack{e \in C_i \\ C_i \in \mathcal{C}}} wt(e)$$

$$\#\text{cycle cover}(G) = \sum_{\mathcal{C}\text{-cycle cover}} wt(\mathcal{C})$$

Valiant's Thm:

Redn 3CNF $\to$ Graph (directed)

$\varphi \longmapsto G_\varphi$

$M \cdot \#sat(\varphi) = \#cycle\ cover\ (G_\varphi)$

$\varphi = (x_1 \vee x_5 \vee \overline{x_8})(x_7 \vee \ldots)$

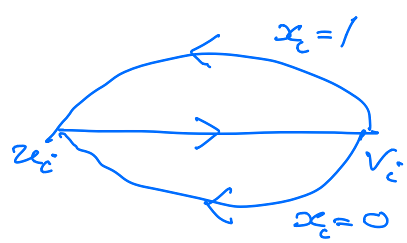
Assumption: We will assume for each var $x_i$, $x_i$ appears an equal number of times positively & negatively.

(by adding $(x_i \vee x_i \vee \overline{x_i})$ or $(\overline{x_i} \vee \overline{x_i} \vee x_i)$)

Reduction:

Variable Gadget

$x_i$




Clause Gadget.

$C = (l_1 \vee l_2 \vee l_3)$

$G_c$ - seven cycle covers corresponding to seven sat assigns

# consistent cycle covers = # satisfying assignments

→ Proof – consistency (equality gadgets) gadgets

(Contd in next lecture)

s.t. all inconsistent cycle covers are annihilated

$$\sum_{\substack{C - \text{inconsistent} \\ \text{cycle cover}}} wt(C) = 0$$