NP – Non-deterministic polynomial

$P \subseteq NP$

As all det. TM can be seen as a non-deter. TM with exactly one choice (or no choice → move to q_{rej})

eg
- checking if 2 nos are relatively prime $\in P$, $\in NP$.
- $0^n 1^n \in P$, $\in NP$
- ww $\in P$, $\in NP$.
- check if a given no. is prime $\in NP$, $\in P$ (complicated proof)

eg: check if graph is connected

$\langle G \rangle$ encoded as (matrix, adjacency list(·))

size = $O(n^2)$ n = # vertices.

NP class (alternate defn)

A lang. $L \in NP$ if $w \in L$ can be

Scanned by CamScanner

verified with the help of a certificate (whose size is polynomial to $|w|$) with a polynomial time soln.

to $|w|$

(Q1). does k-colouring $\in NP$

$L = \{G$ that are 3-colourable$\}$

at present no known poly. time algo to check if a given $G \in L$

But given an assignment of colours to each vertex as certificate we can verify it in $O(m)$ time. $m \to \#$ edges

(Q2) k-clique certificate - clique vertices

verify - if all are connected

$L \in NP$, if $\exists$ non-det. TM that recognises L in $O(|w|^c)$ i.e. it gives yes answers in polynomial time. ('no' answer need not be in pol. time)

If 'no' answer can also be verified in polynomial time.

$\therefore L \in$ co-NP or $\underline{\bar{L} \in NP}$

If $\underbrace{L \in P}_{\text{decides } L} \Rightarrow L \in NP$ & $L \in$ co-NP.

<u>T.P the 2 defn for NP is same.</u>

① Use the path of non-det. TM 'N' as our certificate. Now since N reaches q_{acpt} in pol. time, we can verify certificate in $O(|w|^2)$

② Guess the certificate & run the verifier algorithm. It guis ~~the~~ a path in the non-det TM and combining all these path guis the non-det TM

Travelling salesman problem

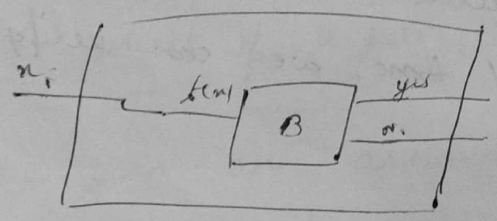
III[r] to finding a least weighted Hamiltonia cycle $\rightarrow$ decision version is $\in NP$

If prob. A is not in P, then B is also not in P

$\Rightarrow$ B is in P $\Rightarrow$ A is in P

$$A \leq_m^P B$$

i.e. assume $B \in P$ ($\exists$ an algo. in pol. time)



$f(x)$ in $|x|^c$

CNF - SAT

it $\in NP$ as given a certificate we can verify it in polynomial time

$\Rightarrow$ if CNF-SAT $\in P$, then $P = NP$?

NP-H class

Problems that are atleast as hard

as any problem in NP

$\therefore \forall x : x \in NP\text{-}H,\ x \le_m^p L$ where $L \in NP\text{-}H$

$\therefore$ if $L \in P$, then $x \in P \Rightarrow NP = P$!

---

① Clique :- Is there a clique of size k

② independent set :- Is there an ind set of size k.

~~Assume~~ clique $\in NP\text{-}H$. Then does ind sets $\in$ NP-H?

let ① $= L_1$ & ② $= L_2$

$\forall x : x \in NP,\ x \le_m^p L_1$

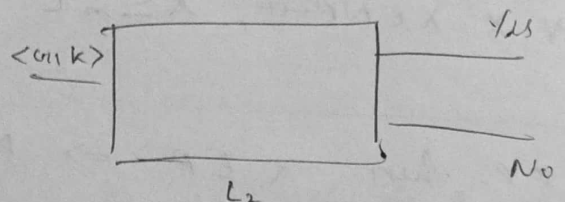
if $L_1 \le_m^p L_2$ then $\forall x : x \in NP, x \le L_2$

& $L_2 \in NP\text{-}H$.

To show $L_1 \le_m^p L_2$.

$\Rightarrow$ make L_1 using L_2 in polynomial time.

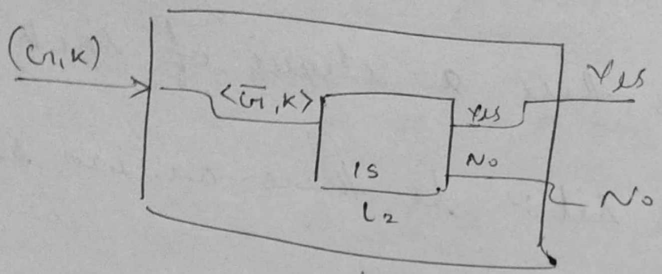
$\therefore$ ~~it~~ just prove that if L_2 runs in poly. time, L_1 also runs in pol. time



G has IS of size k

L_2

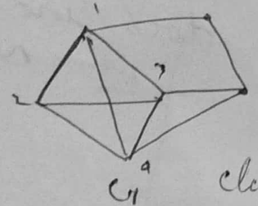
(IS)



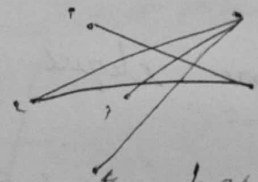
G has clique of size k

L_1

clique



$\Rightarrow \bar{G} =$



clique of size 4

has ind. set of size 4

∴ If G has clique, $\bar{G}$ has ind set

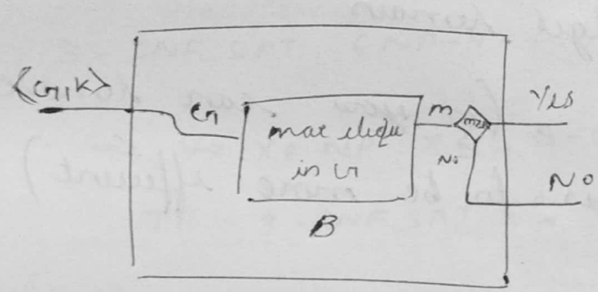
$f(G) = \bar{G}$ can be done in polynomial time

∴ $L_1 \leq_m^p L_2$

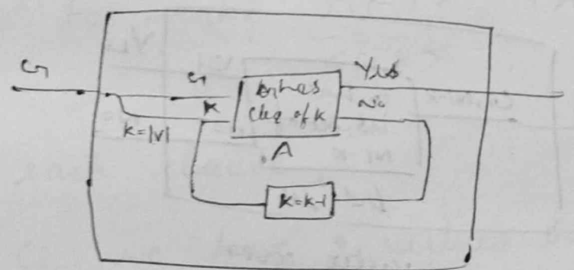
(A) CLIQUE (decision problem)

(B) CLIQUE (optimisation) - what is the max sized clique?

p.T that both are polynomialy equivalent



if B has $O(n^c)$ then A has $O(n^{c'})$



if A has $O(n^c)$ then B has $O(kn^c)$

(C) Find a biggest clique in G.

Remove an edge, check if it has

Scanned by CamScanner

the same sized clique as before.

if NO, keep edge & remove another

if YES, remove another

repeat this till only clique

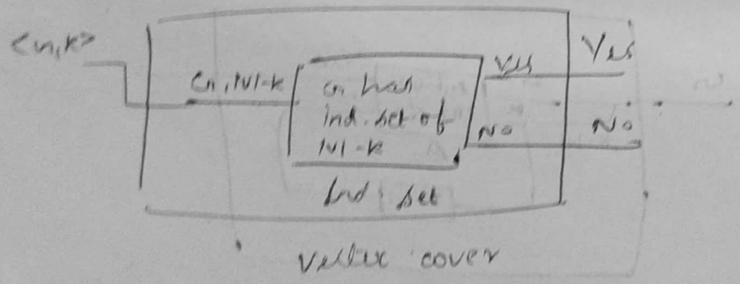
edges remain

(or you can remove

vertices to be more efficient)

Q Vertex cover: set of vertices s.t all

edges have atleast on vertex in it.



if G has vertex cover of size k, then

G also has ind. set of size |V|-k ?

& vice versa

NP-C

if $A \in NP$ & $A \in NP\text{-}H$, then

$A \in NP\text{-}C$

- Show that clique $\in NP\text{-}H$ if

  3-CNF_SAT $\in NP\text{-}H$.

  $\Rightarrow \forall X : X \in NP \quad X \le^P_m$ 3-CNF-SAT

  TP 3-CNF_SAT $\le^P_m$ clique

  i.e. $\exists$ $f(F)$ computable in $O(|F|^c)$ s.t

  $F \in$ CNF-SAT $\Leftrightarrow$ $f(F) \in$ CLIQUE

how to make $f(F)$? (make CNF-SAT
$\langle G, k \rangle$ using CLIQUE
whose i/p is $f(F)$)

for each clause,

C_i : we have 3 vertices in the graph G

let edges be s.t

$\rightarrow$ no edge b/w nodes in same clause

$\longleftrightarrow$ edge b/w nodes in diff clauses
except if both are compliment of

each other.

$f(F)$ is $O(|F|^2)$

Now to prove (i) $F \in CNF \Rightarrow f(F) \in \langle G, k \rangle \in CLIQUE$

$F \in CNF \Rightarrow \exists$ atleast one literal in each clause which is 1

let $X = (x_1, x_2 \ldots x_k)$ be those nodes from each clause. Now no $x_i \in X$ ~~are~~ is a compliment of another $x_j \in X$ (as all are 1 if F is sat)

$\Rightarrow \exists$ edge b/w each $x_i \in X$

$\therefore$ $|X| = k$ (# clause)

$\therefore$ $\langle G, k \rangle \in$ CLIQUE

(ii) $f(F) \in CLIQUE \Rightarrow F \in$ CNF-SAT

let $X = \{x_1, x_2 \ldots x_k\}$ be clique

each $x_i \in$ diff clause. (no edge b/w nodes in same clause)

Now since no edge exist b/w compliments, all $x_i \in X$ are not compliment of $x_j \in X$

$\therefore$ assign 1 to each literal corresponding nodes in X

$\Rightarrow \exists$ atleast one literal with 1 in each clause

$\Rightarrow F \in$ CNF-SAT (F is satisfiable)

Q. S.T 3-SAT is NP-hard if (general) SAT is NP-H.

A. SAT $\in$ NP-H $\Rightarrow \forall X : X \in NP, X \leq_m^P SAT$

TP $SAT \leq_m^P 3\text{-}SAT$

to make $f(F)$ in $O(|F|^k)$

In general sat, let

clause $C_2 = x_2 \vee \bar{x}_3$

$C_3 = x_1 \vee x_2 \vee x_3 \vee x_4$

$$\underbrace{x_2 \vee \bar{x}_3}_{\phi'} \equiv \underbrace{(x_2 \vee \bar{x}_3 \vee z_4) \wedge (x_2 \vee \bar{x}_3 \vee \bar{z}_4)}_{\phi'}$$

→ if ϕ = True ⇒ ϕ' is also true & vice versa

because $\phi' = (Z_4 \vee \phi)(\overline{Z_4} \vee \phi)$

$\phi = T \Rightarrow Z_4 \vee \phi = 1$ &

$\overline{Z_4} \vee \phi = 1 \Rightarrow \phi = 1$

III^{rly} $C_1 = x_1 \equiv \underbrace{(x_1 \vee t_1)}_{\text{III}^r \text{ to } C_2}\underbrace{(x_1 \vee \overline{t_1})}_{\text{III}^r \text{ to } C_2}$

∴ make it 3

if $C_4 = (x_1 \vee x_2 \vee \underbrace{\overline{x_3} \vee x_4}_{t_1})$

$\equiv (x_1 \vee x_2 \vee t_1) \wedge \underbrace{(\cancel{x_1 \vee x_2 \vee} \overline{t_1} \vee \overline{x_3} \vee x_4)}_{\text{always } 1}$

in the sense
satisfiable iff

but if t_1 is independent var,

$x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_5$

$(x_1 \vee x_2 \vee t_1) \wedge (\overline{t_1} \vee x_3 \vee t_2) \wedge (\overline{t_2} \vee x_4 \vee x_5)$

if ϕ is sat, we say ϕ' is also sat
by some assignment of value

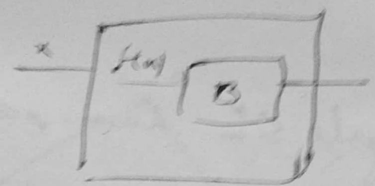
for

for k literal clause,

$x_1 \vee x_2 \vee x_3 \vee (\ ) \cdot x_k \vee x_k$

$A \leq_m^{P} B \Rightarrow$ Algo for A in $O(n^c)$ if

B runs it in $O(n^k)$ time



Q.9 hamiltonian path : Path that goes through all vertex

directed & undirected

b) hamiltonian cycle : cycle that covers all vertices

c) Is there a ham. path from u to v
$\langle G, u, v \rangle$

use c) to build b)

instead of just adding edge b/w u & v, add a vertex w & 2 edge $\langle u, w \rangle$ & $\langle w, v \rangle$

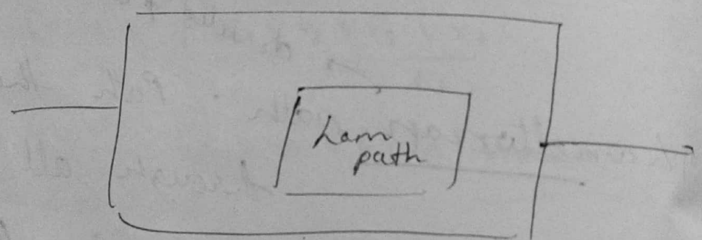
Now, if $\exists$ a hamiltonian cycle

then $\exists$ a path from u to v

else

there does not exist path from u to v

hamiltonian cycle $\leq_m^p$ ham. path.



ham cycle

if it was known that is there a path from u to v to be in P,

then

remove edge u,v

is there a path from u to v

if YES G has ham. cycle.

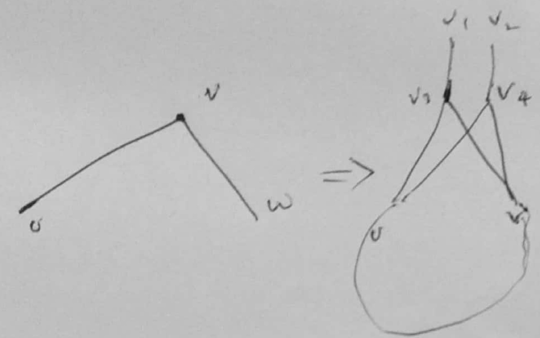
else

pick another edge u,v

Scanned by CamScanner

Now, what if it was a general hamiltonian path problem?

(can't take compliment & give it to ham path eg: if G was complete graph $\bar{G}$ = no edge $\Rightarrow$ no ham path



$\not\Rightarrow$ G has no ham cycle

eg: Set cover: Given a set $\{e_1, e_2, e_3 \ldots\} = U$

& subsets $S_1, S_2 \ldots \subseteq U$

Given k, is there a k subset whose union is U

(prove using vertex cover algo,

U = set of edges

$S_1\ S_2$ etc are edges incident on vertex $1, 2 \ldots k$