- ϕ_{cell}: mi dice che una cella porta con sé un solo simbolo. Si può ottenere così:

$$\phi_{cell}(x) = \bigwedge_{i,j \in [n^k]} \left[\left(\bigvee_{s \in C} x_{i,j,s} \right) \wedge \left(\bigwedge_{\substack{s,t \in C \\ s \neq t}} \left(\overline{x_{i,j,s}} \vee \overline{x_{i,j,t}} \right) \right) \right]$$

$\forall$ coppie di variabili almeno una assume valore F.

- ϕ_{start}: ci dice che la prima riga deve contenere la config. iniziale $\# q_0 w_1 \dots w_n \sqcup \dots$

$$\phi_{start}(x) = x_{1,1,\#} \wedge x_{1,2,q_0} \wedge x_{1,3,w_1} \dots$$
$$\dots \wedge x_{1,n^k,\#}$$

- $\phi_{accept}(x)$: esiste una configurazione accettante.

$$\phi_{accept}(x) = \bigvee_{\nu, i \in [n^k]} x_{\nu,i,q_{acc}}$$

– Φ_{move}: Ci assicura che ogni riga del Tableau sia consistente rispetto alla δ_N a partire dalla riga precedente.

Per assicurare questo, usiamo le finestre 2×3. Diremo che una finestra è lecita se non viola le regole di δ_N.

#	q_0	w_1	w_2
S	q_0	w_1	

NON è LECITA

Se ad es. $\delta_N(q_0, w_1) \Rightarrow (P, S, R)$

#	q_0	w_1	...
#	S	P	w_2 ...

è LECITA

Oss. : Se la prima riga è la configurazione iniziale e ogni finestra del tableau è lecita, allora ogni riga segue dalla precedente in accordo alla δ_N.

È chiaramente vero, segue dalla def. di finestra lecita.

$$\Phi_{move}(x) = \bigwedge_{i,j \in [n^k]} \left(\text{"finestra}(i,j) \text{ LECITA"} \right)$$

finestra (i,j) ha cell (i,j) in alto al centro.

Inoltre:

"funzione (n,j) LECITA" $=$

$$\bigvee_{\substack{e_1,\dots,e_6 \\ \text{funzione lecita}}} \Big(x_{n,j-1,e_1} \wedge x_{n,j,e_2} \wedge x_{n,j+1,e_3} \wedge x_{n+1,j-1,e_4} \wedge x_{n+1,j,e_5} \wedge x_{n+1,j+1,e_6} \Big)$$

$e_1 \quad e_2 \quad e_3$

$e_4 \quad e_5 \quad e_6$

Made with Goodnotes

Dimensione formula: # variabili $O(n^{2k})$ perché ci sono n^{2k} celle e $|C|$ non dipende da n. Anche ϕ_{cell}, ϕ_{move}, ϕ_{acc} hanno dimensione $O(n^{2k})$. ϕ_{start} ha dim. $O(n^k)$. Quindi la formula ha dim. polinomiale. ∎

In realtà non solo SAT, ma anche 3SAT è NP-completo. Esistono altri problemi NP-completi: CLIQUE, 3COL, HAM, ...

Come si fa a dimostrare che altri problemi sono NP-completi? Per riduzione.

Vediamo esempio: $CLIQUE = \{ \langle G, K \rangle :$ G contiene K-clique $\}$

TEO CLIQUE è NP-completo.

Dim. Intanto $CLIQUE \in NP$. Resta da mostrare: $\forall L \in NP$, $L \leq_m^p CLIQUE$.

Mostriamo $3SAT \leq_m^p CLIQUE$ e così il resto segue dal fatto che 3SAT è NP-hard.

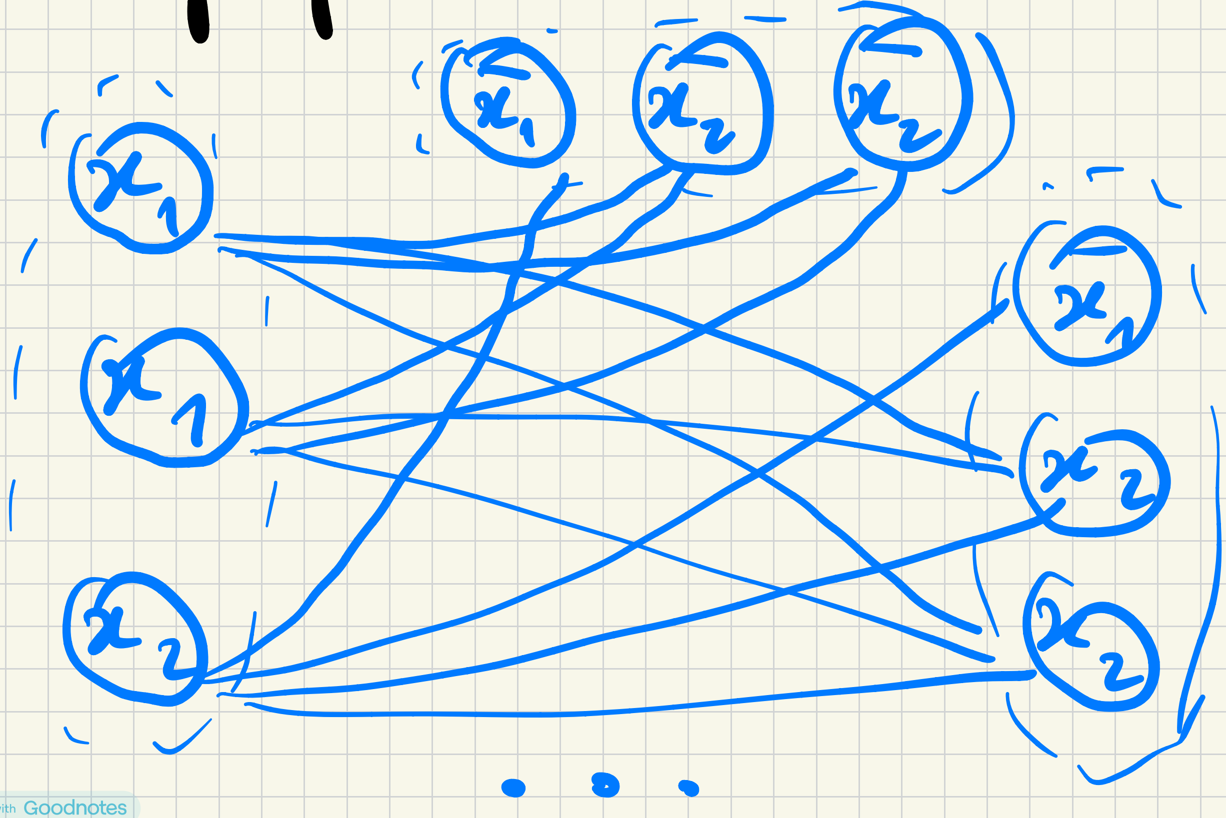
Sia ϕ la formula:

$$\phi = (a_1 \vee b_1 \vee c_1) \wedge \cdots \wedge (a_k \vee b_k \vee c_k)$$

la riduzione genera $\langle G, k \rangle$ in modo che

$\phi \in 3SAT$ sse $\langle G, k \rangle \in CLIQUE$.

Il grafo: contiene k triple $t_1, \dots, t_k$.



$$\phi = (x_1 \vee x_1 \vee x_2) \wedge$$
$$(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_2) \wedge$$
$$(\bar{x}_1 \vee x_2 \vee x_2)$$

Archi di G: Connetto tutto eccetto che

① No archi tra nodi stessa tripla.

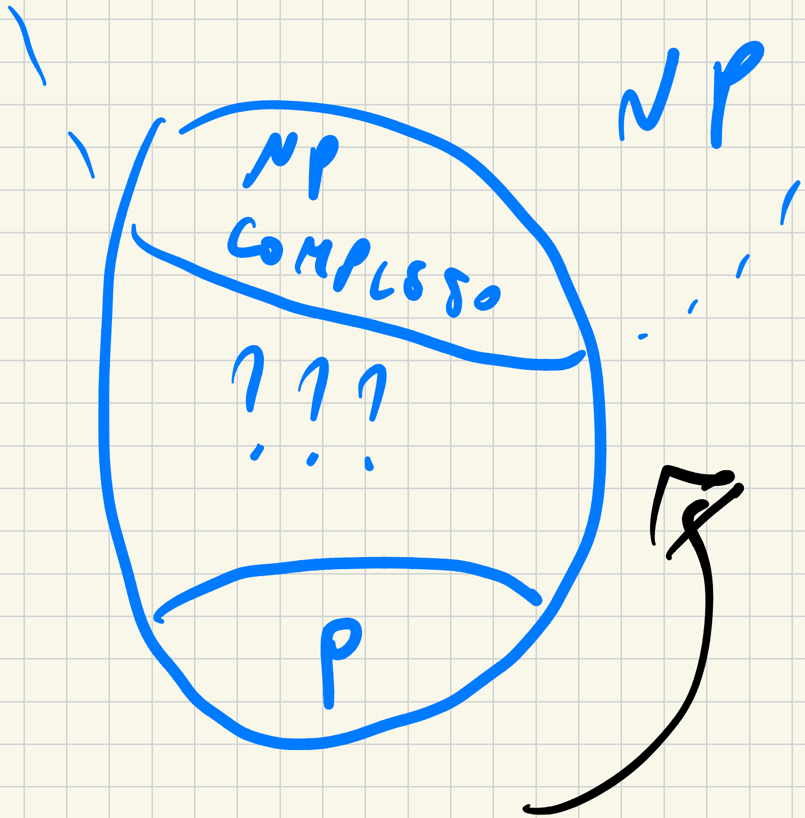
② No archi tra nodi con etichette complementari.

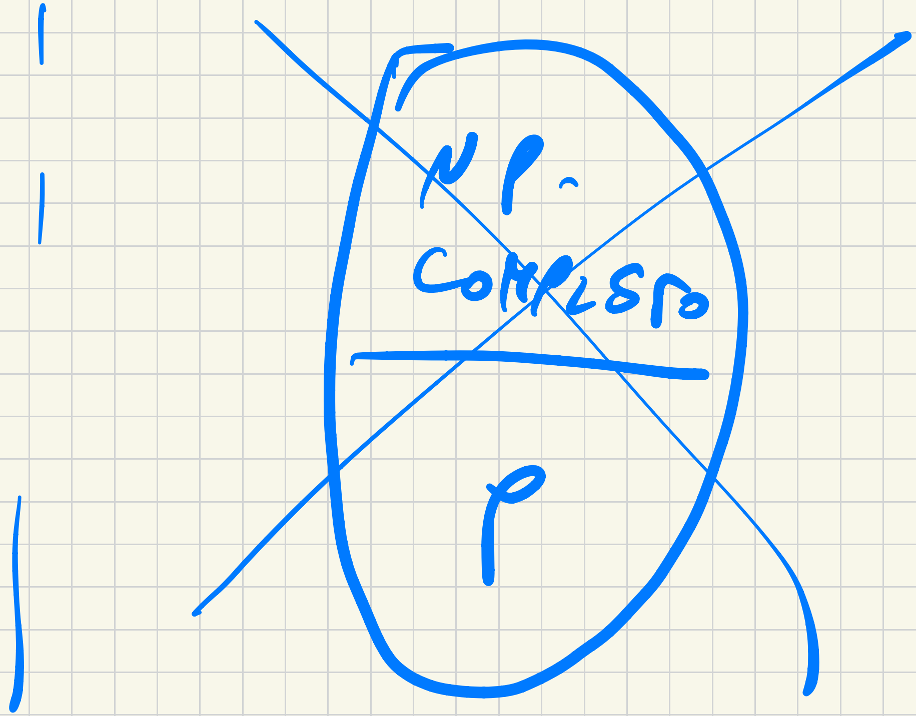
Correttezza: Sia ϕ soddisfacibile, ovvero esiste assegnamento. Ovvero, almeno un letterale è V in ogni clausola. Seleziono il nodo nella tripla t_K corrispondente a un letterale V. # nodi selezionati: K. Siccome le triple sono diverse, c'è sempre un arco tra due nodi a meno che ho selezionato in due triple in modo con etichetta x_i e $\overline{x_i}$. Ma al più uno fra x_i e $\overline{x_i}$ è V.

Altra direzione: Abbiamo G una k-clique. Ma allora i nodi della clique appartengono a triple diverse per costruzione di G. Assegno le variabili in modo che l'etichetta associata sia V. Questo assegnamento soddisfa ϕ

$\hookrightarrow$ (Questo è sempre possibile perché non c'è mai arco $x - \overline{x}$.)

Domanda naturale: Esiste un linguaggio $L \in NP$ ma $L \notin P$ (assumendo $P \neq NP$) t.c. L non è NP-completo? <u>SI</u>





TEO (Ladner)

coNP

Sappiamo NP è equivalente a certificare membership $x \in L$.

coNP: certificare $x \notin L$.

Esempio: Sia $UNSAT = \overline{SAT}$. $UNSAT \in NP$?? Che cosa è il certificato ???

DEF $coNP = \{ L : \overline{L} \in NP \}$

Nota: $coNP \neq \overline{NP}$. $UNSAT \in coNP$.

Vediamo alcuni fatti interessanti.

TEO $SAT \in P \iff UNSAT \in P$.

Dim. Se $SAT \in P$, $\exists$ poly-time TM che decide $\langle \phi \rangle$ accetta sse $\langle \phi \rangle$ soddisfacibile.

Per decidere UNSAT: scambio ACC and REJECT. Funziona anche al contrario. $\blacksquare$

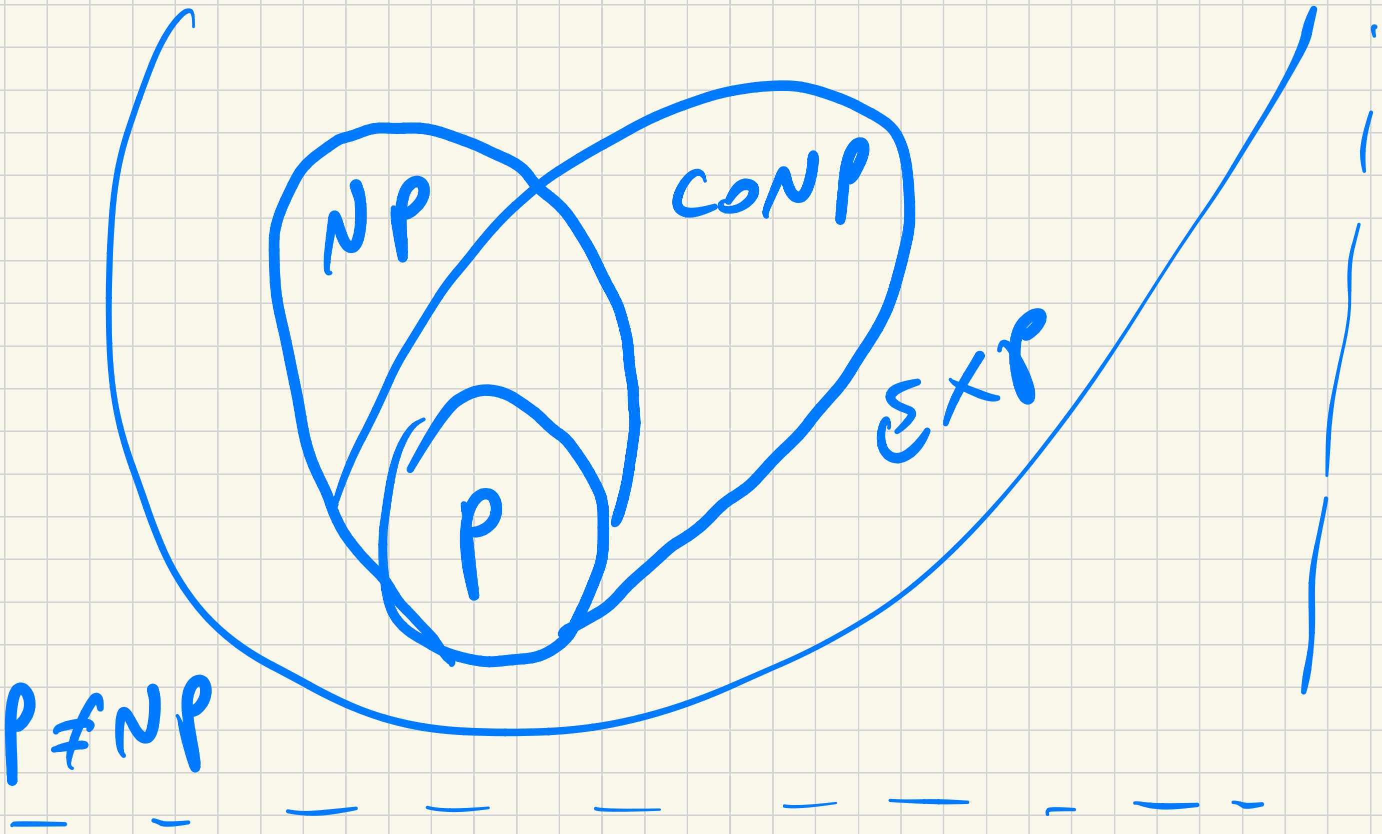
Non è una riduzione $SAT \leq UNSAT$

TEO P è chiusa per complemento: $L \in P \iff \overline{L} \in P$.

Dimostrazione volentieri. Ovvero: $P = coP$.

Similmente: $EXP = coEXP$.

La stessa tecnica non dimostra $NP = coNP$.
Quindi qual è la relazione tra NP e $coNP$?



TEO $coNP \subseteq EXP$.

DIM. Sia $L \in coNP$, allora $\overline{L} \in NP \subseteq EXP$.
ovvero $\overline{L} \in EXP$
e quindi
$L \in coEXP = EXP$ ∎

Similmente:

TEO $P \subseteq coNP$.

DIM. $L \in P \Rightarrow \overline{L} \in coP = P \subseteq NP$

$\Rightarrow L \in coNP$ ∎

Inoltre $NP \neq coNP$ è più forte di $P \neq NP$:

TEO $P = NP \Rightarrow P = coNP \ (= NP)$

DIM. Se $L \in coNP$, allora $\overline{L} \in NP = P$.
Ovvero $\overline{L} \in P$ e quindi $L \in P$. Questo dimostra $coNP \subseteq P$. Siccome $P \subseteq coNP$ allora $P = coNP \ (= NP)$. ∎

Made with Goodnotes

COR $coNP \neq NP \Rightarrow P \neq NP$.

Ci sono anche linguaggi coNP-completi.

TEO $NP = coNP \Leftrightarrow UNSAT \in NP$.

DIM. ($\Rightarrow$) Se $NP = coNP$, allora $UNSAT \in coNP = NP$, ovvero $UNSAT \in NP$.

($\Leftarrow$) Suppongo $UNSAT \in NP$. Mostro $coNP \subseteq NP$ e poi $NP \subseteq coNP$. Il ragionamento è simmetrico, vediamo solo la prima inclusione.

Se $L \in coNP$ devo far vedere $L \in NP$.

Ma sappiamo $L \in coNP \Rightarrow \overline{L} \in NP$.

Pertanto $\bar{L} \leq_m^p SAT$ per NP-completezza.
Infine $\bar{L} \leq_m^p SAT \iff L \leq_m^p UNSAT$
e $UNSAT \in NP$. Ovvero $L \in NP$. ∎

DEF L è coNP-completo se:

- $L \in coNP$
- $\forall A \in coNP$, $A \leq_m^p L$.

TEO UNSAT è coNP-completo.

DIM. Sappiamo già che $UNSAT \in coNP$. Inoltre
$A \leq_m^p UNSAT$ sse $\bar{A} \leq_m^p SAT$.

Made with Goodnotes

Ma $\bar{A} \in NP$ (siccome $A \in coNP$).

$\Rightarrow$ UNSAT è coNP-completo. $\boxtimes$

Altro esempio (esercizio):

TAUTOLOGY = $\{ \langle \phi \rangle$: ogni assegnamento rende ϕ vera $\}$.

è coNP-completo.

In altri termini:

- Se $L \in NP$, $\exists$ poly-time $V(x, y)$ t.c. $\forall x$, $x \in L$ sse $\exists y$ per cui

$V(x, y) = ACC$.

- Se $L \in coNP$ ($\overline{L} \in NP$), $\exists$ poly-time $V(x,y)$ t.c. $\forall x$, $x \notin L$ sse $\exists y$
  ($x \in \overline{L}$)
  per cui $V(x, y) = ACC$.

Se $L \in NP \cap coNP$ posso certificare entrambe! Un fatto curioso: molti linguaggi in $L \in NP \cap coNP$ sono poi risultati essere in P:

- PERFECT MATCHING su grafi. Dato

$G = ((U, V), E)$ bipartito stabilire se ha PM (subset archi E tale che ogni nodo incide esattamente un arco).

- PRIMES : Insieme di numeri primi.

C'è il sospetto che sia sempre vero, ma non lo sappiamo. Esempio:

$$\text{FACTOR} = \{ \langle X, K \rangle : X \text{ ha fattore primo } \leq K \}$$

$\text{FACTOR} \in NP \cap coNP$ ma non sappiamo se $\text{FACTOR} \in P$.