V è eseguibile in tempo $O(|x|^k)$ ovvero polinomiale in $|x|$.

Notare che un tale verificatore implica che $|y| = poly(|x|)$.

DEF NP è l'insieme di linguaggi L che ammettono un verificatore a tempo polinomiale.

PROP. $3COL \in NP$.

DIM. Devo mostrare che esiste un verificatore V associato al linguaggio:

- Suo input $\langle G, y \rangle$
- Interpreta $y = (c_1, \dots, c_n)$ dove $n = \#V$

e $c_i \in \{R, B, Y\}$.

- Per ogni $(i, j) \in E$ REJECT sse $c_i = c_j$.

Chiaramente V ha complessità di tempo polinomiale in $|G| + m$.

Analisi:

- YES CASE: Se $G \in 3COL$, $\exists\, y = (c_1, \dots, c_n)$ che è una valida 3-colorazione di G, ovvero $\forall (i, j) \in E$ $c_i \neq c_j$. Pertanto $V(\langle G, y \rangle) = ACC$.

- NO CASE: Se V accetta per qualche $\langle G, y \rangle$, allora $y = (c_1, \dots, c_n)$ è una 3-colorazione valida di G. ▨

TNM $P \subseteq NP \subseteq EXP$.

Ricordiamo che $P \neq EXP$ per Teorema di gerarchia del Tempo. Quindi o $P \neq NP$ o $NP \neq EXP$, ma non sappiamo quale sia vera.

Dim. Sia $L \in P$, allora devo mostrare $L \in NP$.

$L \in P \Rightarrow \exists$ TM M t.c. $x \in L$ sse $M(x) = Acc$ e inoltre M ha tempo di esecuzione polinomiale.

Ecco il verificatore:

$V(x, y)$: Ignora y ed esegui $M(x)$.

- YES CASE : Se $x \in L$, $V(x, y = \varepsilon) = M(x) = Acc$.

- NO CASE: Se $V(x, y) = ACC$, allora $M(x)$ accetta. Quindi $x \in L$.
D'altra parte, se $L \in NP$ allora chiaramente $L \in EXP$. Basta considerare la TM M che prova tutti i certificati $y \in poly(|x|)$ e ACCETTA sse $V(x, y) = ACC$. Complessità di Tempo $2^{O(n^k)}$ e quindi esponenziale

Definizione alternativa di NP: Tramite il non-determinismo. Qui consideriamo il modello delle TM non-deterministiche.
La complessità di Tempo $t(n)$ di una NTM N

è il massimo numero di passi richiesti che N su ogni x t.c. $|x| = n$ e su ogni rama di computazione.

DEF $NTIME(t(n)) = \{ L : \exists NTM\ N$ t.c. $L(N) = L$ e N ha tempo $O(t(n)) \}$

DEF $NP = \bigcup_{k \in \mathbb{N}} NTIME(n^k)$.

$NEXP = \bigcup_{k \in \mathbb{N}} NTIME(2^{n^k})$

Ad es.: $SAT \in NP$. Basta costruire una NTM $N(\langle \phi \rangle)$ che decide se $\langle \phi \rangle \in SAT$.

$N(\langle \phi \rangle)$:

- Sia n il # di variabili in ϕ.
  Inizializza $x[1 \dots n]$; $i = 0$.
- "Top": $i{+}{+}$; se $i > n$ vai a "Check"
  * goto-both * "Write 0", "Write 1".
- "Write 0": $x[i] = 0$
  goto "Top"
- "Write 1": $x[i] = 1$

Top "Top"

- "Check" : Accetta sse $\phi(x) = 1$.

THM Le due definizioni di NP sono equivalenti.

Dim. ($\Rightarrow$) Se L ha un verificatore V con tempo $poly(|x|) = k \cdot |x|^k$ per $k \in \mathbb{N}$. Definisco NTM $N(x)$ per decidere L:

- Fa la stessa cosa della NTM per SAT, ovvero indovina non deterministicamente y con $|y| \leq k|x|^k$.

- "Check": Accetta sse $V(x, y) = Acc$.

Ogni macchina che tempo $poly(|x|)$ e quindi N ha tempo polinomiale col modello

$$x \in L \quad sse \quad N(x) = Acc.$$

($\Leftarrow$) Se L è decidibile in tempo polinomiale da NTM N, devo fare vedere il verificatore. Siccome N ha tempo $k \cdot |x|^k$ allora questo è upper bound sul numero di istruzioni * fatte both * (scelte non deterministiche). Allora il certificato y identifica l'insieme delle possibili scelte:

- $V(x, y)$ interpreta $y \in \{0,1\}^{k \cdot |x|^k}$

- Simula $N(x)$ e quando N deve fare l'i-esima scelta non-det. usiamo y_i per soddisfacere la scelta det. da effettuare.

Chiaramente $V(\langle x, y \rangle)$ ha complessità di tempo poly ($k > 1$); inoltre:

$x \in L$ sse $N(x)$ accetta sse $\exists y$ tale che $V(x, y) = Acc$. ∎

OPIS : 7 4 9 U B Z Z G

RIDUZIONI

Finora abbiamo visto diversi linguaggi interessanti in NP, per cui non sappiamo se sono anche in P.

Vedremo che tutti questi linguaggi sono in P se uno di loro in particolare è in P.

Le regole sono le RIDUZIONI.

THM Se SAT ∈ P, allora 4-COL ∈ P.

DIM. Data una TM M_{SAT} per SAT con tempo polinomiale devo costruire TM M_{4COL} per decidere in tempo polinomiale 4COL.

La costruzione deve trasformare l'affermazione "G è k-colorabile" in "ϕ_G è soddisfacibile".

Sia $G = (V, E)$ con $\#V = n$. La formula ϕ_G avrà $2n$ variabili: $x_1, x_1', x_2, x_2', \dots, x_n, x_n'$.

dove $x_i, x_i' \in \{0,1\}$ codificano il colore

x_i	x_i'	COLORE
F	F	R
F	V	B
V	F	Y
V	V	G

Devo codificare nella formula il vincolo che per ogni arco $(v,i) \in E$ si abbiano colori diversi.

$$(x_i, x_i') \neq (x_j, x_j')$$

$$\Leftrightarrow \quad \neg\left(x_i \leftrightarrow x_j \;\wedge\; x_i' \leftrightarrow x_j'\right)$$

$$\Leftrightarrow \left(\left(\neg x_i \vee x_j\right) \wedge \left(\neg x_j \vee x_i\right)\right) \wedge \left(\left(\neg x_i' \vee x_j'\right) \wedge \left(\neg x_j' \vee x_i'\right)\right) \wedge \bigwedge \phi_{i,j}$$

(Ovvero $\quad p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$

$$\equiv (\neg p \vee q) \wedge (\neg q \vee p)$$

$$\phi_G = \bigwedge_{(i,j) \in E} \phi_{w_{i,j}}$$

la valutazione è polinomiale chiaramente. Inoltre:

- Se $G \in 4\text{-COL}$, esiste $c = (c_1, \dots, c_n)$ una 4-colorazione, più $x_1, x_1', \dots, x_n, x_n'$ coi loro assegnamenti sono tali che $\phi_G(x) = 1$.

- Se ϕ_G è soddisfacibile $\exists$ assegnamento $x = (x_1, x_1', \dots, x_n, x_n')$ t.c. $\phi_G(x) = 1$ e questo x corrisponde a 4-colorazione $c = (c_1, \dots, c_n)$ t.c. $c_i \neq c_j \ \forall (i,j) \in E$. ☒

DEF Siano A, B linguaggi. $A \leq_m^P B$ se $\exists$ TM $R : \{0,1\}^* \to \{0,1\}^*$ t.c. R è eseguibile in poly-Time $\forall x \in \{0,1\}^*$, $x \in A$ sse $R(x) \in B$.

TEOREMA Se $A \leq_m^P B$ e $B \in P$, allora $A \in P$.

DIM. Data TM M_B per B (siccome $B \in P$) t.c. $L(M_B) = B$, e la funzione R di cui sopra, chiaramente:

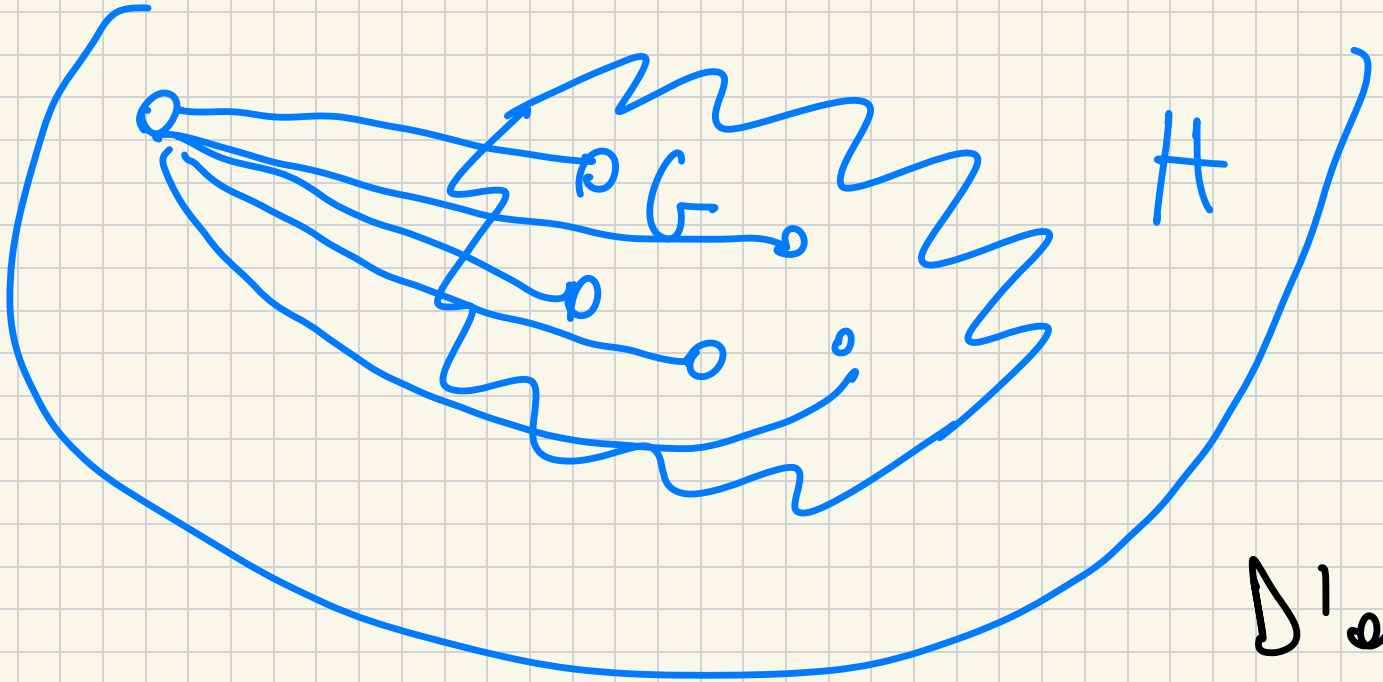
$M_A(x) \equiv M_B(R(x))$ dove $x \in A$

in tempo polinomiale in $|\langle x \rangle|$. ∎

THM $3\text{COL} \leq_m^p 4\text{COL}$.

DIM. Definisco $R(\langle G \rangle) = \langle H \rangle$ t.c.

$G \in 3\text{COL}$ sse $H \in 4\text{COL}$.



H aggiunge un nodo e lo connette a tutti gli altri nodi di G.

D'altra parte:

- Se $G \in 3\text{COL} \Rightarrow H \in 4\text{COL}$ perché posto arbitrario al nuovo nodo il 4° colore.

- Se $H \in 4\,COL \Rightarrow G \in 3\,COL$ perché mediante il colore del nodo in G deve essere l'inverso del colore del nuovo nodo.

TEM $\leq_m^P$ è transitiva: Se $A \leq_m^P B$ e $B \leq_m^P C$, allora $A \leq_m^P C$.

Dim. $\exists$ TM R_1 t.c. $x \in A \iff R_1(x) \in B$ $\forall x$

$\exists$ TM R_2 t.c. $y \in B \iff R_2(y) \in C$ $\forall y$

$\forall x$: $x \in A$ sse $R_1(x) \in B$ sse $R_2(R_1(x)) \in C$

diverso $R(\cdot) \equiv R_2(R_1(\cdot))$

Siccome R_1, R_2 sono a tempo polinomiale, anche R lo è ■

Quindi: Siccome $4COL \leq_m^p SAT$ e $3COL \leq_m^p 4Col$ allora $3COL \leq_m^p SAT$.

Osserviamo che non tutte le riduzioni sono funzioni di tempo polinomiale. Esempio:

$4CHROMA = \{ \langle G \rangle : \chi(G) = 4 \}$

$\chi(G) = 4$ ovvero G è 4-colorabile ma non è 3-colorabile.

In effetti $4CHROMA \leq_T^p SAT$ → Turing

Dato $\langle G \rangle$ consideriamo la seguente procedura:

- Usa "SAT oracle" V la riduzione $R_{3col \to SAT}(\langle G \rangle)$ e restituisce se il grafo non è 3 colorabile. Se non lo è rifiuta.
- Usa "oracolo SAT" su $R_{3col \to SAT}(\langle G \rangle)$ se rifiuta accetta.

Ovvero se posso decidere SAT allora la procedura di sopra decide 3 CHROMA. Ma non è una riduzione mediante funzione.

THM Se $A \leq_m^P B$ e $B \in NP$, allora $A \in NP$.

Dim. Se $\exists$ polynome R t.c. $x \in A$ sse $R(x) \in B$ ed esiste N_B NTM poly-time che decide B, pongo $N_A(x) = N_B(R(x))$.

$x \in A$ sse $R(x) \in B$ sse $N_B(R(x)) = Acc$

sse $N_A(x) = Acc$.

E ancora N_A ha Tempo polinomiale ∎

Un concetto importante: NP-completezza -

DEF Un linguaggio A è NP-difficile se $\forall L \in NP$, $L \leq_m^p A$.

Se A è NP-difficile ed $A \in NP$, allora A è NP-completo.