ALL 21 maggio

Thursday, May 21, 2020 4:11 PM

P è la classe dei problemi decisionali risolubili in tempo poly

NP è la classe dei problemi decisionali verificabili in tempo poly

$ZAINO_{OTT}$ $ZAINO_{DEC}$

Dato un problema di zaino e un valore K esiste una sol. con valore $\geq K$?

$ZAINO_{DEC} \rightarrow ZAINO_{OTT}$

Se sappiamo risolvere $ZAINO_{OTT}$

" " anche $ZAINO_{DEC}$

$P \subseteq NP$ $P \overset{?}{=} NP$ $P \neq NP$ ↑

FATORIZZAZIONE È OPEN

RIDUCIBILITÀ POLINOMIALE

P P' decisionali

$P \rightarrow P'$

$P \to P'$ $\qquad x \to x'$

$P \leq_p P'$

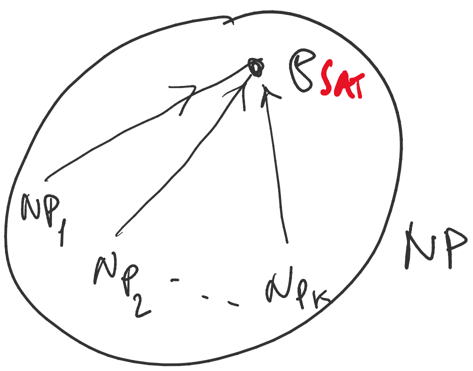
$P \propto P'$

ogni istanza x di P può essere trasformata in un'istanza x' di P' in tempo poly e che P' risponde "true" su x' se e solo se P risponderebbe true su x.

Poiché P' risolve P allora possiamo dire che P' almeno difficile quanto P.

$$P_{DEC} \to P_{OTT}$$

Def: Un problema P è NP-completo se tutti i problemi nella classe NP si possono ridurre poly a P



P è il problema più difficile nella classe

Il primo problema trovato NP-completo:

SODDISFATTIBILITÀ

SATISFIABILITY

SAT

<u>Input</u> : $X = (x_1 \dots x_n)^n$ variabili booleane
insieme C di m clausole
$C = (c_1, \dots, c_m)$ separate da "$\wedge$"
C_i è composta di <u>letterali</u> separati da "$\vee$"

$X = (a, b, c)$

$$F = (a \vee b) \wedge (\bar{a} \vee \bar{b} \vee c) \wedge (\bar{c}) \quad \Theta(m)$$

- <u>F è in forma normale congiuntiva</u>

a = true
$\bar{b}$ = true
$\bar{c}$ = true

a = True
b = false
c = false

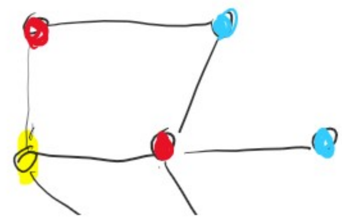
SAT è NP-completo _ Teo di Cook-Levin (1971)

1) SAT $\in$ NP : esiste verifica polinomiale

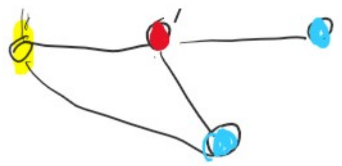
2) Tutti i problemi $NP_1, NP_2 \dots \rightarrow$ SAT

Regole di trasformazione per esprimere tutti i problemi di NP come una formula di SAT.

3-COLOR



r g b



Grafo G è rappresentato da matrice di adiacenza $A = \begin{bmatrix} 0 & 1 & 0 & 1 & \cdot \\ - & - & - & & - \\ & & & & \end{bmatrix}$ $\quad A_{ij} < \begin{array}{ll} a_{ij} & \text{se } A[i,j]=1 \\ \bar{a}_{ij} & \text{se } A[i,j]=0 \end{array}$

$$B_i = (r_i \wedge \bar{g}_i \wedge \bar{b}_i) \vee (\bar{r}_i \wedge g_i \wedge \bar{b}_i) \vee (\bar{r}_i \wedge \bar{g}_i \wedge b_i)$$

$$C_{ij} : \quad a_{ij} \Rightarrow (r_i \wedge \bar{r}_j) \vee (g_i \wedge \bar{g}_j) \vee (b_i \wedge \bar{b}_j)$$

$$\bar{a}_{ij} \vee (r_i \wedge \bar{r}_j) \qquad '' \qquad ''$$

$$\Phi_G = \bigwedge_{1 \le i,j \le n} A_{ij} \wedge \Big(\bigwedge_{1 \le i \le n} B_i\Big) \wedge \Big(\bigwedge_{1 \le i,j \le n} C_{ij}\Big)$$

3COLOR $\rightarrow$ SAT

per tutti gli archi — per tutti i vertici — per tutte le coloros.

Φ_G può essere trasformata in una forma normale congiuntiva

Il teorema di Cook-Levin fa vedere come ogni $P \in NP$ può essere trasformato in una forma normale congiuntiva.

SAT è NP-completo

Un problema P è NP-completo se:

1) $\underline{P \in NP}$ per CL $P \rightarrow$ SAT

1) $P \in NP$ $\quad \varphi$ CL $\quad P \to SAT$

2) $SAT \to P$

$P \equiv SAT$

$$
\begin{tikzcd}
 & SAT \arrow[d] & & \\
 & 3\text{-}SAT \arrow[dl] \arrow[dr] & & \\
3DM \arrow[d] & & VC \arrow[d] \arrow[dr] & \\
PARTITION & & HC & CLIQUE
\end{tikzcd}
$$

PARTITION

Dato un insieme $A = \{a_1 \ldots a_n\}$

esiste $A' \subset A$:

$$\sum_{a_i \in A'} a_i = \sum_{a_i \in A - A'} a_i \quad ? \quad = \sum_{a_i \in A} a_i / 2$$

PARTITION è NP-completo

$A = \{9, 7, 5, 4, 1, 2, 3, 8, 4, 3\}$ $\sum \frac{46}{2} = 23$

Come dimostrare

P è NP-completo

1) Si guardo WIKIPEDIA ✓

2) Garey, Johnson ✓
Computer Intractability

3) dimostrazione

a) $P \in NP$

b) SAT (o un qualsiasi altro altro problema NP-completo) $\to P$

PARTITION è NP-completo

PARTITION $\to P$

SAT $\to$ PARTITION

Riducibilità polinomiale è transitiva

$$P \xrightarrow{t_1} Q \xrightarrow{t_2} R$$

$P \to R$ in tempo $t_1 + t_2$

Def: un problema P è detto NP-hard se la sua versione decisionale è NP-completo

Vogliamo dimostrare che $ZAINO_{OTT}$ è NP-hard

dobbiamo " " $ZAINO_{DEC}$ è NP-completo

$ZAINO_{DEC}$: Esiste una sol. con valore $\geq K$?

$ZAINO_{DEC} \rightarrow ZAINO_{OTT}$

1) Verifica polinomiale $ZAINO_{DEC} \in NP$

<u>Verificapol</u> (V, P, W, b)

b array binario di n elementi:

$b[i] = 1$ se e solo $i \in$ alla soluzione

0	1	0	1	1	0	1

la somma dei pesi $< W$

la somma dei valori è confrontata col max corrente $\Theta(n)$

2) $PARTITION \rightarrow ZAINO_{DEC}$

Restrizione: si considera un problema ridotto dello zaino

$ZAINO_R : p_i = v_i \quad 1 \leq i \leq n$

$$W = \frac{\sum p_i}{2}$$

A': el. nello zaino

$A - A'$; el fuori dallo zaino

$PARTIZIONE \rightarrow ZAINO_R \rightarrow ZAINO_{DEC} \rightarrow ZAINO_{OTT}$

$P \neq NP$

NP-completi

- TEST DI PRIMALITÀ ERA OPEN

NP

- FATTORIZZAZIONE È OPEN