$\mathrm{TIME}(f) = \{ I \mid \exists M.\ \forall x \in \Sigma.\ M(x) \xrightarrow{t} (si, w) \wedge t \leq f \}$

$\mathrm{NTIME}(f) = \{ I \mid \forall x \in I.\ \exists N.\ N(x) \xrightarrow{t} (si, w) \wedge t \leq f \}$

TEOREMA $\quad \mathrm{NTIME}(f) \subseteq \mathrm{TIME}(c^f)$ con $c \in \mathbb{R}^+ \wedge c > 1$

A parole: se una MdT non deterministica decide un problema in tempo f allora posso costruire una MdT deterministica che lo decide in tempo c^f per una qualche costante, ossia che esplora l'albero delle computazioni non deterministiche

Dim: Data N costruisco M:

Definisco il grado di diramazione:

$\mathrm{grado}((q, \sigma)) = \#(q', \sigma' D')$ t.c. $(q, \sigma, q', \sigma', D') \in \Delta_N$

$d := \max \{ \mathrm{grado}(q, \sigma) : (q, \sigma) \in Q \times \Sigma \}$

Per ogni stato q e simbolo σ ordino lessicograficamente

Concretamente, una computazione e' una sequenza di scelte

se la sequenza e' lunga t allora possiamo scrivere la computazione come una successione di numeri naturali minori di t nell'intervallo $[0; d[$

M visita l'albero per livelli, cioe' considera tutte le successioni in ordine crescente

Quante successioni visite? $\quad O(d^{t+1})$

Pongo $c = d$

Q.E.D

Def: N: MdT n.d. a k nastri I/O decide $\mathcal{I}$ in spazio non deterministico f se e solo se

- N decide $\mathcal{I}$
- $\forall x \in \mathcal{I}$. $\exists\, w_1 \dots w_k$ t.c. $N(q_0, \triangleright x \dots \triangleright) \rightarrow^* (s_i, w_1 \dots w_k)$

e $\sum_{i=2}^{k-1} |w_i| \leq f(|x|)$.

$NSPACE(f) := \{ \mathcal{I} \mid N \text{ decide } \mathcal{I} \text{ in s.n.d. } f \}$

Def: $NPSPACE := \bigcup_{k \geq 1} NSPACE(n^k)$

TEOREMA (SAVITCH): NPSPACE = PSPACE

PROBLEMA TSP

n citta', $d(i,j)$ distanza tra le citta' i,j

Si ricorda che una distanza e' una funzione

- riflessiva $d(i,j) = 0 \iff i = j$
- simmetrica $d(i,j) = d(j,i)$
- triangolare $d(i,j) \leq d(i,k) + d(k,j)$

TSP consiste nel trovare il cammino di costo minimo che attraversa le n citta' una e una sola volta

Si rende un problema di decisione chiedendosi se si ha un cammino di costo $\leq B$ valore soglia.

forza bruta:

M: MdT deterministica che genera tutte le soluzioni

1. M genera tutte le permutazioni delle città
   le permutazioni sono $O(n!)$
2. M calcola le distanze in $O(n^3)$

guess & try:

N: MdT non deterministica

1. N scrive sul nastro ausiliario una stringa di n numeri naturali tra 1 e n in $O(n)$ passi
2. N verifica la soluzione e calcola il costo del cammino in $O(n^3)$

FUNZIONE di MISURA APPROPRIATE

$f: \mathbb{N} \to \mathbb{N}$ viene definita di misura appropriata sse è:

- monotona crescente
- $\exists$ M che calcola f in: $\begin{cases} \text{tempo } O(f(|x|) + |x|) \\ \text{spazio } O(f(|x|)) \end{cases}$

Vale il seguente teorema:

TEOREMA di GERARCHIA

se $f: \mathbb{N} \to \mathbb{N}$ è appropriata allora:

- $TIME(f) \subset TIME(f(2n+1)^3)$

   si dimostra facendo vedere che

   $I = \{ x \mid \varphi_x(x) \downarrow \text{ entro } f(|x|) \text{ passi} \}$, $I \in TIME(f^3)$,
   $I \notin TIME(f)$

- $SPACE(f) \subset SPACE(f + \log f)$

COROLLARIO $\mathcal{P} \subset EXP = \bigcup_{k \geq 1} (2^{n^k})$

Dim:

$$\mathcal{P} \subseteq TIME(2^n)$$

Per teorema di gerarchia:

$$\mathcal{P} \subseteq TIME(2^n) \subset TIME((2^{2n+1})^3) \subseteq TIME(2^{n^2})$$

Corollario: $\mathcal{NP} \subseteq EXP$

TEOREMA Sia f di misura appropriata, k una costante allora:

- $SPACE(f) \subseteq NSPACE(f)$
- $TIME(f) \subseteq NTIME(f)$
- $NSPACE(f) \subseteq TIME(k^{\log n + f(n)})$

E vale che:

$LOGSPACE \subseteq \mathcal{P}$

$LOGSPACE \subset \mathcal{P}SPACE$

$PSPACE = NPSPACE$

$\mathcal{NP} \subseteq EXP$

La gerarchia non è superiormente limitata

Teorema: $\forall g$: funzione calcolabile totale

$\exists I \in TIME(f) \wedge I \notin TIME(g)$

con $f > g$ quasi ovunque

TEOREMA di ACCELERAZIONE (o di BLUM)

h: funzione calcolabile totale appropriata

I: problema

M: algoritmo

$\forall h.\ \exists I.\ \forall M$ che decide I in f. $\exists M'$ che decide I in f'.

$f(n) > h(f'(n))$ quasi ovunque

TEOREMA della LACUNA (o di BORODIN)

$\exists f$: funzione calcolabile totale tale che $TIME(f) = TIME(2^f)$.

COMPLESSITA' ASTRATTA

ϕ e' una funzione che misura la complessita' se

$\phi(\gamma, x)$ e' un numero naturale

con γ funzione, x input di γ

e soddisfa i seguenti assiomi:

- $\phi(\gamma, x)\downarrow \iff \gamma(x)\downarrow$
- $\forall \phi, x, k$ $\phi(\gamma, x) = k$ e' decidibile