Teorema di Savitch

Enunciato: Per ogni NTM T con complessità di spazio $s(n)$ esiste una TM T' equivalente con complessità di spazio $s_{T'}(n) = O(s^2(n))$.

Prova. Sia $T=(Q,\Sigma,\Gamma,\delta,q_0, q_a, q_r)$ una NTM.
Modifichiamo T in modo che abbia un'unica configurazione di accettazione: $c_a = q'_a\perp$. Basta introdurre due delimitatori sinistro e destro del nastro e aggiungere un nuovo stato di accettazione, q'_a, che T modificata raggiungerà solo da q_a dopo aver cancellato il contenuto del nastro e riposizionato la testina sulla prima cella.

Ricordiamo che la complessità di spazio, $s(n)$, e la complessità di tempo, $t(n)$, di T soddisfano le seguenti relazioni:

$$n \leq s(n) \text{ e } t(n) \leq 2^{O(s(n))}.$$

Teorema di Savitch - raggiungibilità

Definiamo la funzione

$$\text{CANYIELD}(c,c',t) = \begin{cases} 1 & \text{se } c \Rightarrow_T^* c' \text{ in al più } t \text{ passi} \\ 0 & \text{altrimenti} \end{cases}$$

$x \in L(T)$ sse CANYIELD(c_0,c_a,$2^{ks(n)}$), dove $c_0 = q_0 x$ e k è un'opportuna costante.

Definizione ricorsiva di canyield, in cui t è una potenza di 2

se t = 1

CANYIELD(c,c',t) = 1 sse $c \Rightarrow_T c'$ o $c = c'$

se t > 0,

CANYIELD(c,c',t) = 1 sse

CANYIELD(c, c'',t/2) = 1 e CANYIELD(c'',c',t/2) = 1

per una configurazione c'', di lunghezza al più s(n), infatti in tal caso

$c \Rightarrow_T^* c''$ in al più t/2 passi e $c'' \Rightarrow_T^* c'$ in al più t/2.

la TM che calcola CANYIELD

CANYIELD :

sull'input (c1,c2,t)

if t=1 then

if c2 = c1 then restituisci 1

else (cioè c2 ≠ c1)

if se c1 porta a c2 in un passo then restituisci 1

else restituisci 0

else (cioè se t > 1)

for ogni configurazione c" di lunghezza ≤ s(n)

if CANYIELD(c1, c",t/2) = 1 and CANYIELD(c",c2, t/2) = 1

then restituisci 1 e termina

else restituisci 0.

la TM T’ equivalente alla NTM T

T' =

INPUT x

esegui CANYIELD(c_0,c_a,$2^{ks(n)}$) e accetta se restituisce 1, rifiuta altrimenti.

Teorema di Savitch - complessità 1

Il nastro di T' si comporta come uno stack durante il calcolo di CANYIELD(c_1,c_2,2^m) e contiene gli argomenti delle chiamate:

$c_1,c_2,2^m$	$c_1,c_3,2^{m-1}$	$c_1,c_4,2^{m-2}$	...	$c_1,c_{m+2},2^{m-m}$

T' ha complessità di spazio O(s^2(n)). Infatti CANYIELD(c_0,c_a,$2^{ks(n)}$) è calcolata in spazio O(s^2(n)), visto che la profondità della ricorsione è ks(n) e in ogni chiamata lo stack utilizza O(s(n)) spazio.

Teorema di Savitch - complessità 2

Comunque T' ha bisogno di conoscere anche s(n), che dipende da x.
Modifichiamo T' per costruire una TM che non utilizza la conoscenza di s(n).
T" semplicemente tenta tutti i possibili valori di s(n)=1,2,3,...
Per ogni valore i, $1 \leq i \leq s(n)$, T" usa CANYIELD per verificare se c_a è raggiungibile dalla configurazione iniziale e accetta se la risposta è si.
Così se T' accetta x allora anche T" accetta x quando tenta s(n) = i per qualche $i \leq s(n)$.
Nel caso x <u>non</u> sia accettata bisogna impedire che i sia incrementato indefinitamente, quindi prima di incrementare i a i+1 T" verifica se c'è una configurazione che utilizza i+1 celle di nastro raggiungibile da c_0 .
Se non ce ne sono T" rifiuta. Naturalmente il più grande i per il quale dovrà tentare è s(n).
Lo spazio necessario per ricordare l'i corrente è solo log(s(n)) e questo è l'unico spazio in più che T" usa rispetto a T', quindi anche T" lavora in spazio $O(s^2(n))$.

Classi di complessità

Ricordiamo che (N)SPACE(f(n)) è la classe dei linguaggi decisi da una (N)TM in spazio O(f(n))

$$\mathbf{PSPACE} = \bigcup_{k\geq 0} \mathbf{SPACE}(n^k)$$

$$\mathbf{NPSPACE} = \bigcup_{k\geq 0} \mathbf{NSPACE}(n^k)$$

PSPACE = NPSPACE

PSPACE, NP e EXPTIME

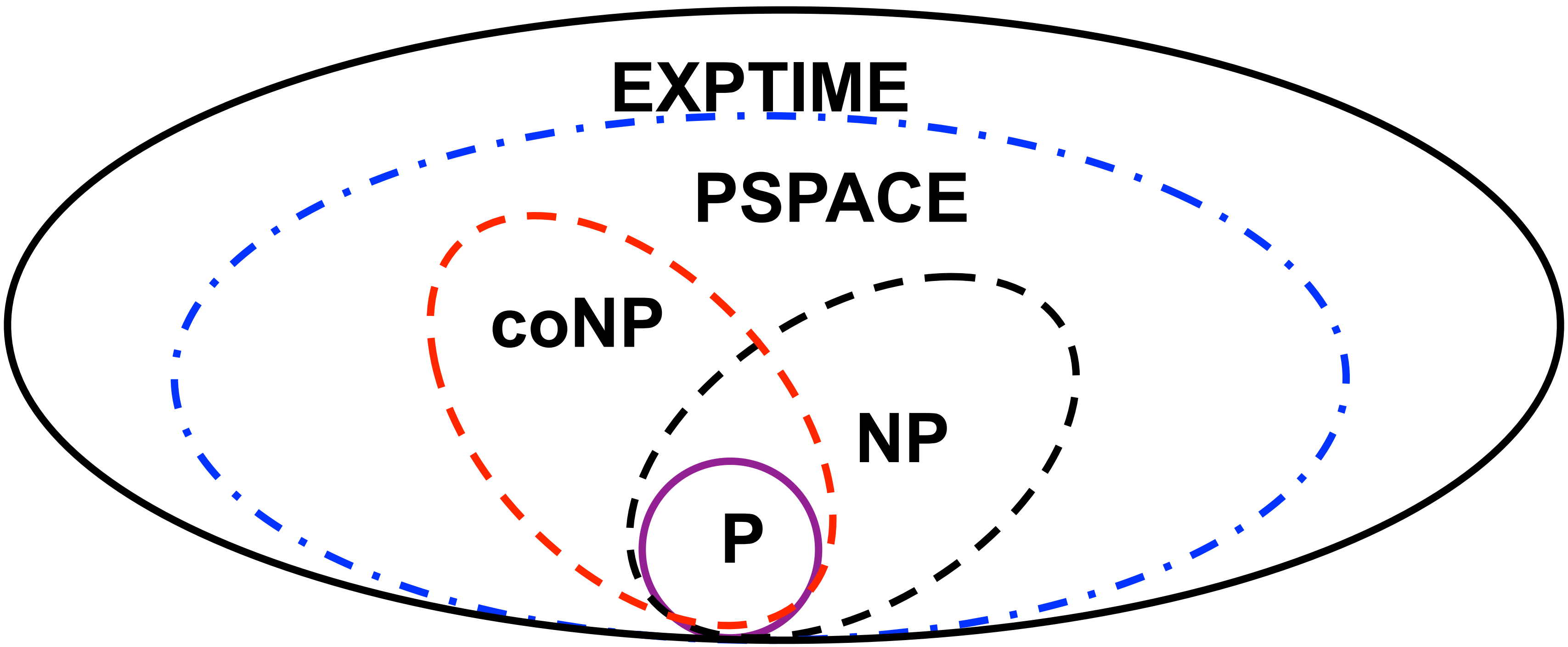
NP $\subseteq$ NPSPACE = PSPACE

PSPACE $\subseteq$ EXPTIME

coNP $\subseteq$ PSPACE

Prova coNP $\subseteq$ PSPACE. Infatti coPSPACE = PSPACE e PSPACE=NPSPACE implica coPSPACE=coNPSPACE. Quindi coNP $\subseteq$ PSPACE

Tra P e EXPTIME?



Si sa solo che P ≠ EXPTIME