Sommario

Proprietà indecidibili delle TM.

Proprietà non Turing riconoscibili delle TM.

$INFIN_{TM}$

Il problema dell'infinito per le TM:
$INFIN_{TM}$ = {<M> | M è in TM e L(M) è infinito} è deciso da qualche TM?

Prova della non decidibilità:

Adottiamo l'alfabeto binario.

Per riduzione da A_{TM}, $A_{TM} \leq_m INFIN_{TM}$.

Ad un input per A_{TM}, < M,w>, associamo <T> , dove T è una TM, in modo tale che w è in L(M) (< M,w> è in A_{TM}) sse L(T) è infinito (e quindi <T> è in $INFIN_{TM}$). Questa associazione è calcolabile, e sia R la macchina che la calcola.

La TM R è così definita:
input <M,w>
output: la TM T così definita:
T = input x
esegui M su w e
se M accetta w, accetta x
se M rifiuta w, rifiuta x.

Se M accetta w allora T accetta ogni parola in {0,1}*, mentre se M non accetta w allora L(T) = ∅, quindi abbiamo ottenuto la riduzione f voluta, calcolata da R:
w è in L(M) (< M,w> è in A_{TM}) sse (R(<M,w>) = <T> è in $INFIN_{TM}$) L(T) è infinito.

Regolarità per TM

REG_{TM} = { <T> | T è una TM e L(T) è regolare} è indecidibile.

Prova. Per riduzione da A_{TM} . Mostriamo che $A_{TM} \leq_m REG_{TM}$.

Ad un input per A_{TM}, < M,w>, associamo <T> , dove T è una TM, in modo tale che w è in L(M) (< M,w> è in A_{TM}) sse L(T) è regolare (e quindi <T> è in REG_{TM}). Questa associazione è calcolabile, e sia R la macchina che la calcola.

Adottiamo l'alfabeto binario.

La TM R è così definita:
input <M,w>
output: la TM T qui definita:
T = input x
    se x = 0^n1^n , per qualche n, allora accetta
    altrimenti (x non è della forma 0^n1^n) esegui M su w e
        se M accetta w, accetta x
        se M rifiuta w, rifiuta x.

Quindi L(T) accetta tutte le parole se M accetta w:
quelle della forma 0^n1^n in ogni caso, quando le riceve in input, e le altre se M accetta w.
Se invece M non accetta w le uniche parole accettate da T sono quelle della forma 0^n1^n , per qualche n. Quindi L(T) è regolare (<T> è in REG_{TM}) sse M accetta w (< M,w> è in A_{TM}).

Non Turing riconoscibilità

Il problema dell'equivalenza per TM:

EQ_{TM} = {<M,M'> | M,M' sono TM e L(M)=L(M')} non è Turing riconoscibile.

<u>Osservazione 1:</u> **se $A \leq_m B \Leftrightarrow \neg A \leq_m \neg B$**

<u>Osservazione 2:</u> **A_{TM} è Turing riconoscibile mentre $\neg A_{TM}$ non lo è.**

<u>Osservazione 3:</u> **Per provare che un qualsiasi problema B non è Turing riconoscibile dovremmo dimostrare $\neg A_{TM} \leq_m B$, ma sfruttando l'osservazione 1 possiamo ridurre A_{TM} al complemento di B, e cioè dimostrare che $A_{TM} \leq_m \neg B$.**

Equivalenza: EQ_{TM}

EQ_{TM} = {<M,M'> | M,M' sono TM e L(M)=L(M')} non è Turing riconoscibile.

Prova della non Turing riconoscibilità:

Riduciamo A_{TM} all'inequivalenza, cioè al complemento dell'equivalenza per TM:

$$A_{TM} \leq_m \neg EQ_{TM} .$$

Ad un input per A_{TM}, <M,w>, associamo <T1,T2>, la codifica di due TM T1 e T2, in modo tale che w è in L(M) sse L(T1)≠L(T2).
Questa associazione è calcolabile, e sia R la macchina che la calcola.

La TM R è così definita:
input <M,w>
output: la coppia di TM <T1,T2> così definite:
T1 = input x
rifiuta.
T2 = input x
esegui M su w
se M accetta w, accetta
se M rifiuta w , rifiuta.

L(T1)= ∅

Se w è in L(M) allora L(T2) = {0,1}*
mentre se w non è in L(M) allora L(T2) = ∅

quindi abbiamo ottenuto la riduzione voluta: w è in L(M) sse L(T1)≠L(T2).

L'inequivalenza: $\neg Q_{TM}$

$\neg EQ_{TM}$ = {<M,M'> | M,M' sono TM e L(M)≠L(M')} è decidibile? NO!

Turing riconoscibile? NO!

Riduciamo A_{TM} al complemento dell'inequivalenza per TM, cioè all'equivalenza: $A_{TM} \leq_m EQ_{TM}$.
Ad un input per A_{TM}, <M,w>, associamo <T1,T2> la codifica di due TM T1e T2 tali che w è in L(M) sse L(T1)=L(T2). Questa associazione è calcolabile, e sia R la macchina che la calcola.

La TM R è così definita:
input <M,w>
output: la coppia di TM <T1,T2> così definite:
T1 = input x
accetta.
T2 = input x
esegui M su w
se M accetta w, accetta
se M rifiuta w , rifiuta.

L(T1)= $\{0,1\}^*$

Se w è in L(M) allora L(T2) = $\{0,1\}^*$
mentre se w non è in L(M) allora L(T2) = $\emptyset$

quindi abbiamo ottenuto la riduzione voluta: w è in L(M) sse L(T1)=L(T2).

Completezza di A_{TM}

A_{TM} = {<M,w> | M è TM e w in L(M)} è completo per $\mathcal{L}$(TM)

Un linguaggio è completo rispetto a una classe di linguaggi se appartiene a quella classe e se ogni linguaggio della classe si riduce ad esso. Nella fattispecie si deve dimostrare che ogni linguaggio Turing riconoscibile si riduce a A_{TM}.

Riduciamo L ad A_{TM} cioè mostriamo che $L \leq_m A_{TM}$ per ogni linguaggio in $\mathcal{L}$(TM).
Ad un input per L, <w>, associamo <T,w> la codifica di una TM T che riconosce L. Questa associazione è calcolabile, e sia R la macchina che la calcola.

La TM R è così definita:
input w
output: la coppia <T,w>, dove T è una TM tale che L(T) = L.

Banalmente w è in L sse <T,w> è in A_{TM}.