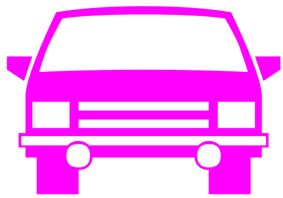
macchine di Turing (MT)



macchine di Turing

- la macchina di Turing e' un automa con testina di scrittura/lettura su nastro illimitato bidirezionale
- ad ogni istante la macchina si trova in uno stato di un insieme finito K
- la funzione di transizione δ porta la macchina da uno stato ad un altro, facendo anche scrivere e spostare la testina
- determinismo

macchine di Turing

- " ....e' incredibile quanto poco occorra per calcolare tutto ciò che e' calcolabile...." (C.Papadimitriou)
- descrivono formalmente il concetto di algoritmo
- sono in grado di simulare ogni linguaggio di programmazione

problema di Hilbert

- nel 1900 David Hilbert, in un intervento ad un congresso, presentò 23 problemi come sfida per il secolo successivo
- il decimo problema riguardava gli algoritmi: dato un polinomio trovare un algoritmo che stabilisca se esso ha una radice intera
  - es. $6x^3yz^2+3xy^2-x^3-10$ ha una radice intera con $x=5$, $y=3$ e $z=0$
- nel suo intervento Hilbert assumeva che l'algoritmo esistesse, il problema era trovarlo

problema di Hilbert

- ora sappiamo (dal 1970) che non esiste nessun algoritmo per risolvere il problema posto da Hilbert
- ma in quel periodo non c'era nessuno strumento idoneo a trattare la questione
- mancava proprio la definizione di algoritmo

problema di Hilbert

- ma se per esempio il polinomio e' x^2+4x+9 ? e' facile stabilire se ha radici intere
- me lo hanno insegnato al liceo!
- il fatto che non esista un algoritmo che risolva il problema generale non implica che non esistano algoritmi che risolvano sotto-problemi particolari
- per equazioni di secondo grado in una sola variabile il problema di Hilbert diventa facile

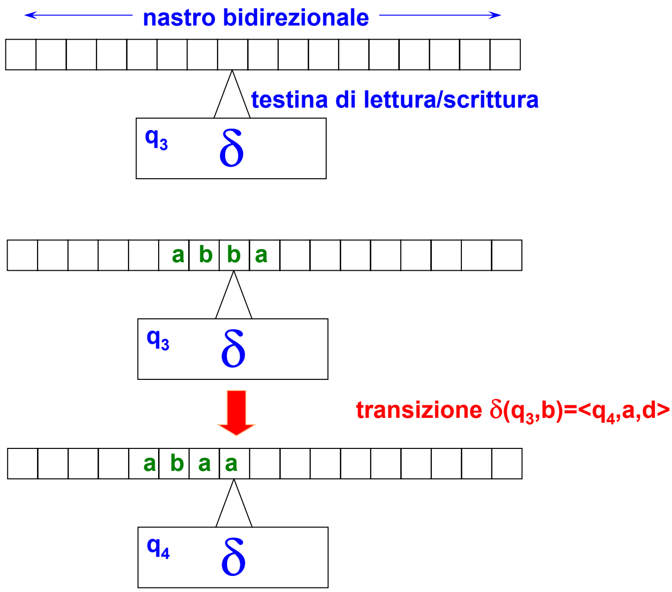
macchine di Turing (MT)

macchina di Turing:

$M=\langle\Sigma,\underline{b},K,q0,F,\delta\rangle$

Σ alfabeto di simboli
$\underline{b}$ carattere speciale, spazio bianco
K insieme finito di stati
F insieme di stati finali
δ funzione parziale di transizione
$\delta : K \times (\Sigma\cup\{\underline{b}\}) \rightarrow K \times (\Sigma\cup\{\underline{b}\}) \times \{d,s,i\}$
d,s,i indicano spostamento a destra, a sinistra e immobilità della testina
$\Sigma_{\underline{b}} = \Sigma\cup\{\underline{b}\}$
le macchine di turing sono usate sia per accettare stringhe sia per calcolare funzioni: riconoscitori e trasduttori

macchine di turing



funzionamento delle MT configurazioni e computazioni

- configurazione di una macchina:
  contenuto del nastro + posizione della testina + stato corrente
- la conoscenza di una configurazione e della funzione di transizione implica la conoscenza della configurazione successiva
- il nastro, pur essendo infinito, ha un numero finito di caratteri $\neq \underline{b}$
- rappresentazione di una configurazione (esempio):

  $$abb\underline{b}bq_iab\underline{bb}bb$$

  q_i e' immediatamente prima del carattere su cui è posizionata la testina

funzionamento delle MT
configurazioni e computazioni

- configurazione di una macchina:
  stringa appartenente al linguaggio
  $$(\Sigma_{\underline{b}})^* \cdot K \cdot (\Sigma_{\underline{b}})^+$$
- normalmente si fa l'ipotesi che all'inizio della computazione il nastro contenga l'input, il resto del nastro contenga $\underline{b}$ e la testina sia sul primo carattere dell'input
- configurazione iniziale:
  stringa appartenente al linguaggio
  $$\{q_0\} \cdot (\Sigma_{\underline{b}})^+$$
- configurazione accettante:
  stringa appartenente al linguaggio
  $$(\Sigma_{\underline{b}})^* \cdot F \cdot (\Sigma_{\underline{b}})^+$$

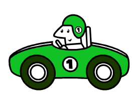
funzionamento delle MT
configurazioni e computazioni

- osservazione: la funzione di transizione si presta ad una rappresentazione tabellare o ad una rappresentazione mediante grafo di transizione
- l'applicazione della funzione di transizione ad una configurazione si chiama passo o mossa
- relazione di transizione per macchina di Turing: relazione binaria sulle configurazioni $\vdash$
  $$c_i \vdash c_{i+1}$$
- computazione per macchina di Turing: sequenza
  $$c_1 \vdash c_2 \vdash \ldots \vdash c_i \vdash \ldots$$
  <u>eventualmente</u> <u>infinita</u> di configurazioni

funzionamento delle MT
configurazioni e computazioni

- usiamo la chiusura riflessiva e transitiva di $\vdash$, indicata con $\vdash^*$, per denotare computazioni finite
- c' $\vdash^*$ c" indica che c" e' ottenibile da c' tramite un numero finito di transizioni da c' a c"
- una computazione finita $c_1 \vdash c_2 \vdash \ldots\ldots \vdash c_n$ e' massimale se non esiste una configurazione c tale che $c_n \vdash c$
  - la configurazione c_n è la configurazione finale della computazione massimale

un giro in macchina con Turing



funzionamento delle MT

MT multinastro

MT non deterministiche

descrizione linearizzata delle MT

MT universale

funzionamento delle MT
riconoscimento di linguaggi

- una MT accetta una stringa x se, quando ha come configurazione iniziale q_0x, ha luogo una computazione massimale la cui ultima configurazione è accettante
- l'insieme L(M) della stringhe accettate da una MT M è il linguaggio accettato da M
- un linguaggio è Turing-riconoscibile se esiste una MT che lo accetta

funzionamento delle MT
riconoscimento di linguaggi

- una MT rifiuta una stringa x se, quando ha come configurazione iniziale q_0x, ha luogo una computazione massimale la cui ultima configurazione non è accettante
- quando facciamo partire una MT su una stringa ci sono tre possibilità: la MT accetta, rifiuta, o cicla per sempre (non si ferma) (non ha luogo una computazione massimale)

funzionamento delle MT
riconoscimento di linguaggi

- una MT M decide il linguaggio L(M) che accetta se quando le proponiamo una stringa che non è di L(M) termina sempre la computazione
- una MT che decide un linguaggio è un decisore
- un linguaggio è Turing-decidibile (o semplicemente decidibile) quando c'è una MT che lo decide

funzionamento delle MT
riconoscimento di linguaggi

- ogni linguaggio Turing-decidibile è anche Turing-riconoscibile ma non è detto il viceversa

funzionamento delle MT
calcolo di funzioni

- macchina di Turing trasduttrice: le configurazioni finali sono di tipo xqy, dove x e' il contenuto del nastro all'inizio della computazione
- convenzione: una macchina trasduttrice calcola la funzione f(x) tale che

$$q_0 x \vdash^* x\underline{b}q_F f(x)$$

funzionamento delle MT

esempio: MT che accetta 0^n1^n $(n \geq 1)$

nastro all'inizio:

$\underline{b}0000011111\underline{b}$

la MT esegue ripetutamente le seguenti operazioni:

- rimpiazza lo 0 piu' a sinistra con una X

  $\underline{b}X000011111\underline{b}$
- si muove a destra verso l'1 piu' a sinistra
- rimpiazza l'1 piu' a sinistra con una Y

  $\underline{b}X0000Y1111\underline{b}$
- si muove a sinistra verso la X piu' a destra
- si muove di una cella sullo 0 piu' a sinistra

se cercando un 1 trova un $\underline{b}$
allora rifiuta

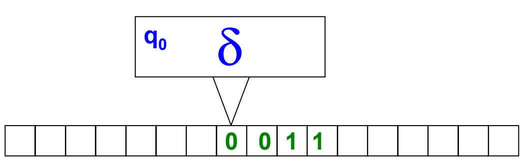
se dopo aver modificato un 1 in Y non trova nessuno 0
allora se non c'e' rimasto nessun 1 accetta

esempio (continua): MT che accetta 0^n1^n ($n\geq 1$)

- q_0 stato usato prima della sostituzione 0 X
- q_1 per muoversi a destra verso il primo 1
- q_2 per muoversi a sinistra verso le X
- q_3 per verificare che non rimane nessun 1
- q_4 stato di accettazione

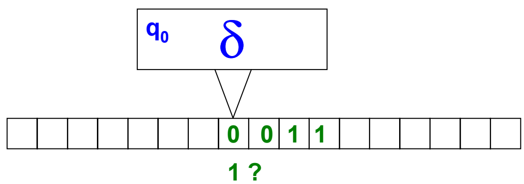
esempio di MT

$\underline{\delta(q_0,0)=\{<q_1,X,d>\}}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\delta(q_1,0)=\{<q_1,0,d>\}$	$\delta(q_1,1)=\{<q_2,Y,s>\}$
$\delta(q_1,Y)=\{<q_1,Y,d>\}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\delta(q_2,X)=\{<q_0,X,d>\}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



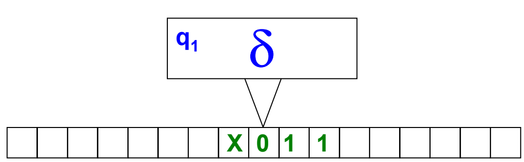
esempio di MT

$\underline{\delta(q_0,0)=\{<q_1,X,d>\}}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\delta(q_1,0)=\{<q_1,0,d>\}$	$\delta(q_1,1)=\{<q_2,Y,s>\}$
$\delta(q_1,Y)=\{<q_1,Y,d>\}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\delta(q_2,X)=\{<q_0,X,d>\}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



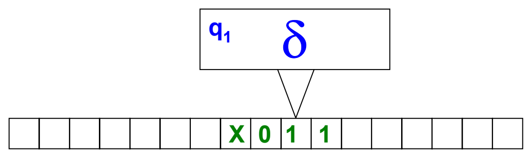
esempio di MT

$\delta(q_0,0)=\{<q_1,X,d>\}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\underline{\delta(q_1,0)=\{<q_1,0,d>\}}$	$\delta(q_1,1)=\{<q_2,Y,s>\}$
$\delta(q_1,Y)=\{<q_1,Y,d>\}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\delta(q_2,X)=\{<q_0,X,d>\}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



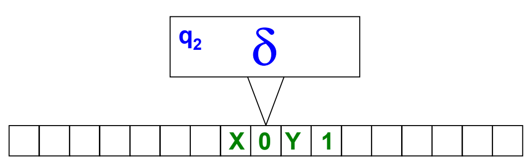
esempio di MT

$\delta(q_0,0)=\{\langle q_1,X,d\rangle\}$	$\delta(q_0,Y)=\{\langle q_3,Y,d\rangle\}$
$\delta(q_1,0)=\{\langle q_1,0,d\rangle\}$	$\underline{\delta(q_1,1)=\{\langle q_2,Y,s\rangle\}}$
$\delta(q_1,Y)=\{\langle q_1,Y,d\rangle\}$	$\delta(q_2,0)=\{\langle q_2,0,s\rangle\}$
$\delta(q_2,X)=\{\langle q_0,X,d\rangle\}$	$\delta(q_2,Y)=\{\langle q_2,Y,s\rangle\}$
$\delta(q_3,Y)=\{\langle q_3,Y,d\rangle\}$	$\delta(q_3,\underline{b})=\{\langle q_4,\underline{b},d\rangle\}$



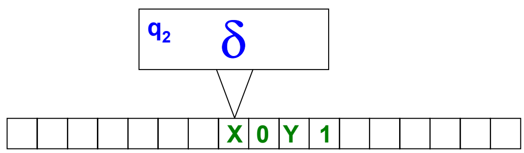
esempio di MT

$\delta(q_0,0)=\{\langle q_1,X,d\rangle\}$	$\delta(q_0,Y)=\{\langle q_3,Y,d\rangle\}$
$\delta(q_1,0)=\{\langle q_1,0,d\rangle\}$	$\delta(q_1,1)=\{\langle q_2,Y,s\rangle\}$
$\delta(q_1,Y)=\{\langle q_1,Y,d\rangle\}$	$\underline{\delta(q_2,0)=\{\langle q_2,0,s\rangle\}}$
$\delta(q_2,X)=\{\langle q_0,X,d\rangle\}$	$\delta(q_2,Y)=\{\langle q_2,Y,s\rangle\}$
$\delta(q_3,Y)=\{\langle q_3,Y,d\rangle\}$	$\delta(q_3,\underline{b})=\{\langle q_4,\underline{b},d\rangle\}$



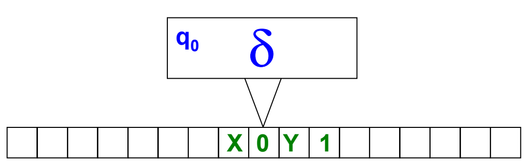
esempio di MT

$\delta(q_0,0)=\{<q_1,X,d>\}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\delta(q_1,0)=\{<q_1,0,d>\}$	$\delta(q_1,1)=\{<q_2,Y,s>\}$
$\delta(q_1,Y)=\{<q_1,Y,d>\}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\underline{\delta(q_2,X)=\{<q_0,X,d>\}}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



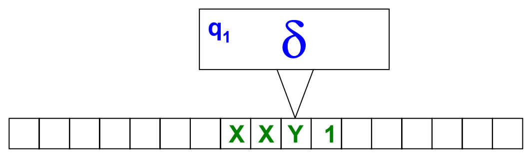
esempio di MT

$\underline{\delta(q_0,0)=\{<q_1,X,d>\}}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\delta(q_1,0)=\{<q_1,0,d>\}$	$\delta(q_1,1)=\{<q_2,Y,s>\}$
$\delta(q_1,Y)=\{<q_1,Y,d>\}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\delta(q_2,X)=\{<q_0,X,d>\}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



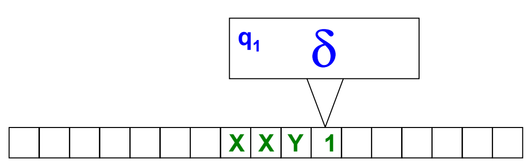
esempio di MT

$\delta(q_0,0)=\{<q_1,X,d>\}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\delta(q_1,0)=\{<q_1,0,d>\}$	$\delta(q_1,1)=\{<q_2,Y,s>\}$
$\underline{\delta(q_1,Y)=\{<q_1,Y,d>\}}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\delta(q_2,X)=\{<q_0,X,d>\}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



esempio di MT

$\delta(q_0,0)=\{<q_1,X,d>\}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\delta(q_1,0)=\{<q_1,0,d>\}$	$\underline{\delta(q_1,1)=\{<q_2,Y,s>\}}$
$\delta(q_1,Y)=\{<q_1,Y,d>\}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\delta(q_2,X)=\{<q_0,X,d>\}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



esempio di MT

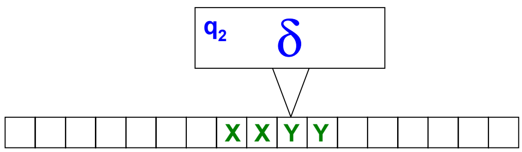
$\delta(q_0,0)=\{\langle q_1,X,d\rangle\}$ $\delta(q_0,Y)=\{\langle q_3,Y,d\rangle\}$

$\delta(q_1,0)=\{\langle q_1,0,d\rangle\}$ $\delta(q_1,1)=\{\langle q_2,Y,s\rangle\}$

$\delta(q_1,Y)=\{\langle q_1,Y,d\rangle\}$ $\delta(q_2,0)=\{\langle q_2,0,s\rangle\}$

$\delta(q_2,X)=\{\langle q_0,X,d\rangle\}$ $\underline{\delta(q_2,Y)=\{\langle q_2,Y,s\rangle\}}$

$\delta(q_3,Y)=\{\langle q_3,Y,d\rangle\}$ $\delta(q_3,\underline{b})=\{\langle q_4,\underline{b},d\rangle\}$



esempio di MT

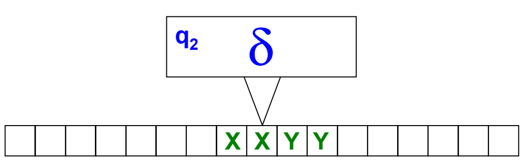
$\delta(q_0,0)=\{\langle q_1,X,d\rangle\}$ $\delta(q_0,Y)=\{\langle q_3,Y,d\rangle\}$

$\delta(q_1,0)=\{\langle q_1,0,d\rangle\}$ $\delta(q_1,1)=\{\langle q_2,Y,s\rangle\}$

$\delta(q_1,Y)=\{\langle q_1,Y,d\rangle\}$ $\delta(q_2,0)=\{\langle q_2,0,s\rangle\}$

$\underline{\delta(q_2,X)=\{\langle q_0,X,d\rangle\}}$ $\delta(q_2,Y)=\{\langle q_2,Y,s\rangle\}$

$\delta(q_3,Y)=\{\langle q_3,Y,d\rangle\}$ $\delta(q_3,\underline{b})=\{\langle q_4,\underline{b},d\rangle\}$



esempio di MT

$\delta(q_0,0)=\{<q_1,X,d>\}$

$\delta(q_1,0)=\{<q_1,0,d>\}$

$\delta(q_1,Y)=\{<q_1,Y,d>\}$

$\delta(q_2,X)=\{<q_0,X,d>\}$

$\delta(q_3,Y)=\{<q_3,Y,d>\}$

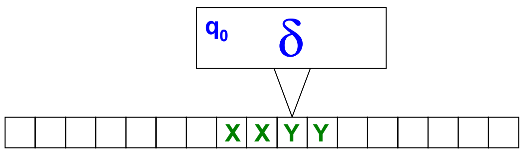
$\underline{\delta(q_0,Y)=\{<q_3,Y,d>\}}$

$\delta(q_1,1)=\{<q_2,Y,s>\}$

$\delta(q_2,0)=\{<q_2,0,s>\}$

$\delta(q_2,Y)=\{<q_2,Y,s>\}$

$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



esempio di MT

$\delta(q_0,0)=\{<q_1,X,d>\}$

$\delta(q_1,0)=\{<q_1,0,d>\}$

$\delta(q_1,Y)=\{<q_1,Y,d>\}$

$\delta(q_2,X)=\{<q_0,X,d>\}$

$\underline{\delta(q_3,Y)=\{<q_3,Y,d>\}}$

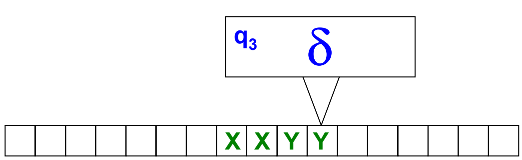
$\delta(q_0,Y)=\{<q_3,Y,d>\}$

$\delta(q_1,1)=\{<q_2,Y,s>\}$

$\delta(q_2,0)=\{<q_2,0,s>\}$

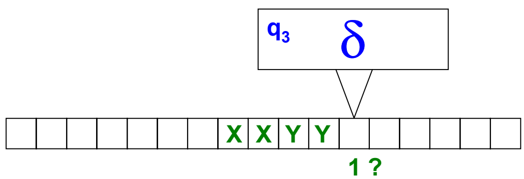
$\delta(q_2,Y)=\{<q_2,Y,s>\}$

$\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}$



esempio di MT

$\delta(q_0,0)=\{<q_1,X,d>\}$	$\delta(q_0,Y)=\{<q_3,Y,d>\}$
$\delta(q_1,0)=\{<q_1,0,d>\}$	$\delta(q_1,1)=\{<q_2,Y,s>\}$
$\delta(q_1,Y)=\{<q_1,Y,d>\}$	$\delta(q_2,0)=\{<q_2,0,s>\}$
$\delta(q_2,X)=\{<q_0,X,d>\}$	$\delta(q_2,Y)=\{<q_2,Y,s>\}$
$\delta(q_3,Y)=\{<q_3,Y,d>\}$	$\boxed{\delta(q_3,\underline{b})=\{<q_4,\underline{b},d>\}}$



macchine di Turing multinastro (MTM)

macchina di Turing a k nastri:

$$M^k=<\Sigma,\underline{b},K,q0,F,\delta(k)>$$

Σ alfabeto di simboli

$\underline{b}$ carattere speciale, spazio bianco

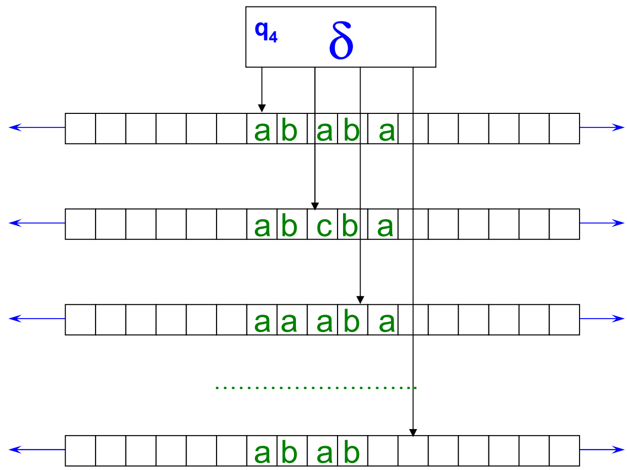
K insieme finito di stati

F insieme di stati finali

$\delta^{(k)}$ funzione di transizione

$$\delta^{(k)} : K \times (\Sigma_{\underline{b}})^k \rightarrow K \times (\Sigma_{\underline{b}})^k \times \{d,s,i\}^k$$

MTM



MTM - configurazioni, transizioni e computazioni

- configurazione:

  $$q\#\alpha_1\uparrow\beta_1\#\alpha_2\uparrow\beta_2\#\ldots.\#\alpha_k\uparrow\beta_k$$

  q è lo stato, la stringa $\alpha_i\uparrow\beta_i$ rappresenta la situazione sul nastro i

  α_i è eventualmente vuota e il primo carattere di β_i è il carattere attualmente osservato
- configurazione accettante: q appartiene a F
- configurazione iniziale:

  $$q_0\#\uparrow\beta_1\#\uparrow Z_0\#\ldots.\#\uparrow Z_0$$

  l'input sta sul primo nastro e gli altri contengono il carattere Z_0
- transizioni e computazioni: analoghi alle MT

equivalenza tra MTM e MT

- strumento di lavoro: MT a nastro suddiviso in tracce

  se il nastro ha h tracce la testina può leggere/scrivere h caratteri contemporaneamente
- la corispondenza tra MT e MT a nastro suddiviso in tracce è immediata
- osservazione: se sulle tracce sono usati gli alfabeti $\Sigma_1,\Sigma_2,....,\Sigma_h$, una MT corrispondente ha un alfabeto Σ con $|\Sigma|\geq|\Sigma_1\times\Sigma_2\times....\times\Sigma_h|$

macchine di Turing multinastro equivalenza tra MTM e MT

teorema: data una MTM $M=<\Sigma,\underline{b},K,q0,F,\delta^{(k)}>$ a k nastri esiste una MT che simula t passi di M in $O(t^2)$ passi usando un alfabeto di cardinalità $O((2|\Sigma|)^k)$

dimostrazione:

- costruiamo una MT $M'=<\Sigma',\underline{b},K',q_0',F',\delta'>$ con nastro suddiviso in 2k tracce che simula M
- poi costruiamo una MT M" equivalente a M'
- le k tracce di posto pari di M' rappresentano i k nastri di M
- le k tracce di posto dispari di M' rappresentano con il carattere "↓" la posizione delle testine sui k nastri di M

nastro a 2k tracce per simulare una macchina di Turing a k nastri

								↓											
$\underline{b}$	a	b	c	b	b	b	b	c	c	c	c	c	b	b	b	b	b	b	$\underline{b}$
	↓																		
$\underline{b}$	c	b	b	b	a	b	a	b	b	b	b	b	c	b	b	b	b	b	$\underline{b}$
					↓														
$\underline{b}$	b	b	b	b	c	b	b	b	b	b	a	b	c	b	b	$\underline{b}$	b	b	$\underline{b}$
									↓										
$\underline{b}$	b	b	b	b	a	c	b	b	a	b	b	b	b	b	b	b	b	b	$\underline{b}$
													↓						
$\underline{b}$	b	b	a	b	b	b	b	b	a	a	a	a	b	b	b	b	b	b	$\underline{b}$

dimostrazione (MTM→MT)

- il nastro di M' all' inizio della computazione si presenta con tutte le tracce dispari "vuote" tranne la prima
- per simulare la funzione di transizione di M che è del tipo:

  $\delta^{(k)}(q_i,\sigma_{i1},..,\sigma_{ik}) = <q_j,\sigma_{j1},..,\sigma_{jk},z_{j1},..,z_{jk}>$

  la δ' deve:

  rintracciare le posizioni dei marcatori,

  scrivere e spostare i marcatori,

  cambiare stato
- quindi per ogni passo di M, M' deve eseguire una sequenza di un numero di passi proporzionale alla distanza (numero di caselle) tra i due marcatori più lontani

nastro iniziale di M' con input acca

								↓											
b	b	b	b	b	b	b	b	a	c	c	a	b	b	b	b	b	b	b	b
								↓											
b	b	b	b	b	b	b	b	Z_0	b	b	b	b	b	b	b	b	b	b	b
								↓											
b	b	b	b	b	b	b	b	Z_0	b	b	b	b	b	b	b	b	b	b	b
								↓											
b	b	b	b	b	b	b	b	Z_0	b	b	b	b	b	b	b	b	b	b	b
								↓											
b	b	b	b	b	b	b	b	Z_0	b	b	b	b	b	b	b	b	b	b	b

dimostrazione(MTM→MT)

- dopo t passi due marcatori possono essersi allontanati di al più O(t) caselle
- se M esegue t passi, M' ne esegue $O(t^2)$
- M" esegue gli stessi passi di M'

- per ciò che riguarda la cardinalità dell'alfabeto di M", abbiamo da codificare con un solo alfabeto stringhe di 2k simboli così composte:

  k simboli appartengono a $\{\underline{b}, \downarrow\}$

  1 simbolo appartiene a $\Sigma \cup \{\underline{b}\}$

  k-1 simboli appartengono a $\Sigma \cup \{\underline{b}, Z_0\}$

  $|\Sigma''| = 2^k(|\Sigma|+1)(|\Sigma|+2)^{k-1} = O((2|\Sigma|)^k)$

esempi di MTM

esempio: MTM per riconoscere $xc\underline{x}$ con $x \in \{a,b\}$

usiamo 2 nastri: uno di input monodirezionale a sola lettura e uno di lavoro che usiamo come pila

durante la scansione di x, fino a c, x viene copiata sul nastro di lavoro

durante la scansione di $\underline{x}$ si confrontano i caratteri con quelli sul nastro di lavoro

configurazione iniziale della MTM:

$q_0\#\uparrow z\#\uparrow Z_0$

esempio (continua): MTM per riconoscere $xc\underline{x}$ con $x \in \{a,b\}$

- 3 stati:
  - q_0 per scandire x
  - q_1 per scandire $\underline{x}$
  - q_2 stato finale
- copiatura iniziale:
  - $\delta(q_0,a,Z_0)=<q_0,a,d>$
  - $\delta(q_0,b,Z_0)=<q_0,b,d>$
- osservazione: la funzione di transizione è descritta in forma semplificata a causa della natura del primo nastro
- copiatura a regime:
  - $\delta(q_0,a,\underline{b})=<q_0,a,d>$
  - $\delta(q_0,b,\underline{b})=<q_0,b,d>$
- passaggio dalla copiatura alla verifica:
  - $\delta(q_0,c,\underline{b})=<q_1,\underline{b},s>$

esempio (continua): MTM per riconoscere xc$\underline{x}$ con $x \in \{a,b\}$

- verifica positiva:
  - $\delta(q_1,a,a)=<q_1,a,s>$
  - $\delta(q_1,b,b)=<q_1,b,s>$
- accettazione:
  - $\delta(q_1,\underline{b},\underline{b})=<q_2,\underline{b},i>$
- computazione con input bacab:

q_0	#↑bacab	#↑Z_0	⊢
q_0	#b↑acab	#b↑$\underline{b}$	⊢
q_0	#ba↑cab	#ba↑$\underline{b}$	⊢
q_1	#bac↑ab	#b↑a	⊢
q_1	#baca↑b	#↑ba	⊢
q_1	#bacab↑$\underline{b}$	#↑$\underline{b}$ba	⊢
q_2	#bacab$\underline{b}$↑$\underline{b}$	#↑$\underline{b}$ba	

esempio (continua): MTM per riconoscere xc$\underline{x}$ con $x \in \{a,b\}$

- computazione con input acb:

q_0	#↑acb	#↑Z_0	⊢
q_0	#a↑cb	#a↑$\underline{b}$	⊢
q_1	#ac↑b	#↑a	

esempi di MTM

esempio: MTM per riconoscere $x\underline{x}$ con $x\in\{a,b\}$

usiamo 3 nastri: uno di input monodirezionale e due di lavoro

copiamo la stringa sui due nastri di lavoro

poi scandiamo i due nastri in senso contrario ed effettuiamo i confronti

configurazione iniziale della MTM:

$q_0\#\uparrow z\#\uparrow Z_0\#\uparrow Z_0$

copiatura iniziale:

$\delta(q_0,a,Z_0,Z_0)=<q_0,a,a,d,d>$

$\delta(q_0,b,Z_0,Z_0)=<q_0,b,b,d,d>$

copiatura a regime:

$\delta(q_0,a,\underline{b},\underline{b})=<q_0,a,a,d,d>$

$\delta(q_0,b,\underline{b},\underline{b})=<q_0,b,b,d,d>$

esempio (continua): MTM per riconoscere $x\underline{x}$ con $x\in\{a,b\}$

termine della copiatura:

$\delta(q_0,\underline{b},\underline{b},\underline{b})=<q_0,\underline{b},\underline{b},s,s>$

riposizionamento della testina:

$\delta(q_0,\underline{b},a,a)=<q_0,a,a,s,i>$

$\delta(q_0,\underline{b},b,a)=<q_0,b,a,s,i>$

$\delta(q_0,\underline{b},a,b)=<q_0,a,b,s,i>$

$\delta(q_0,\underline{b},b,b)=<q_0,b,b,s,i>$

fine del riposizionamento della testina:

$\delta(q_0,\underline{b},\underline{b},a)=<q_1,\underline{b},a,d,i>$

$\delta(q_0,\underline{b},\underline{b},b)=<q_1,\underline{b},b,d,i>$

verifica:

$\delta(q_1,\underline{b},a,a)=<q_1,a,a,d,s>$

$\delta(q_1,\underline{b},b,b)=<q_1,b,b,d,s>$

$\delta(q_1,\underline{b},\underline{b},\underline{b})=<q_2,\underline{b},\underline{b},i,i>$

macchine di Turing non deterministiche (MTND)

macchina di Turing non deterministica:

$M=<\Sigma,\underline{b},K,q_0,F,\delta_N>$

Σ alfabeto di simboli

$\underline{b}$ carattere speciale, spazio bianco

K insieme finito di stati

F insieme di stati finali

δ_N funzione parziale di transizione

δ_N: $K\times(\Sigma\cup\{\underline{b}\}) \rightarrow P(K \times (\Sigma\cup\{\underline{b}\}) \times \{d,s,i\})$

d,s,i indicano spostamento a destra, a sinistra e immobilità della testina; $\Sigma_{\underline{b}} = \Sigma\cup\{\underline{b}\}$

ogni configurazione può dare luogo a più transizioni verso altre configurazioni

MTND - funzionamento

- la macchina può eseguire contemporaneamente più transizioni
- grado di non determinismo di una macchina M

$$\nu(M) = \max|\delta_N(q_i,\sigma_j)|$$

- il comportamento di una macchina non deterministica può essere rappresentato con un albero di computazione

  nodi: configurazioni

  archi: transizioni
- osservazione: il grado di non determinismo coincide con il massimo numero di figli di un nodo dell'albero di computazione

MTND - funzionamento

- una MTND M accetta una stringa x se nell'albero di computazione di M su x è possibile trovare almeno una computazione deterministica accettante
- una MTND è un decisore (decide un linguaggio) se tutte le computazioni deterministiche associate al suo albero di computazione sono massimali

equivalenza tra MT e MTND

le MTND sono più efficienti ma non più potenti computazionalmente delle MT

teorema: per ogni MTND M con grado di non determinismo d esiste una MT M' equivalente che simula k passi di M in $O(kd^k)$ passi

dimostrazione:

- l'albero di computazione di M viene visitato in ampiezza da M' (perche' non in profondita'?)
- M' ha 3 nastri

  nastro 1: contiene l'input

  nastro 2: viene usato per generare tutte le sequenze finite composte da cifre comprese tra 1 e d; le sequenze piu' corte sono generate prima; le sequenze con lo stesso numero di cifre sono generate in ordine numerico cresecente

  nastro 3: nastro di lavoro

dimostrazione(MTND→MT)

- simulazione:

  per ogni sequenza generata sul nastro 2, M' copia l'input sul nastro 3

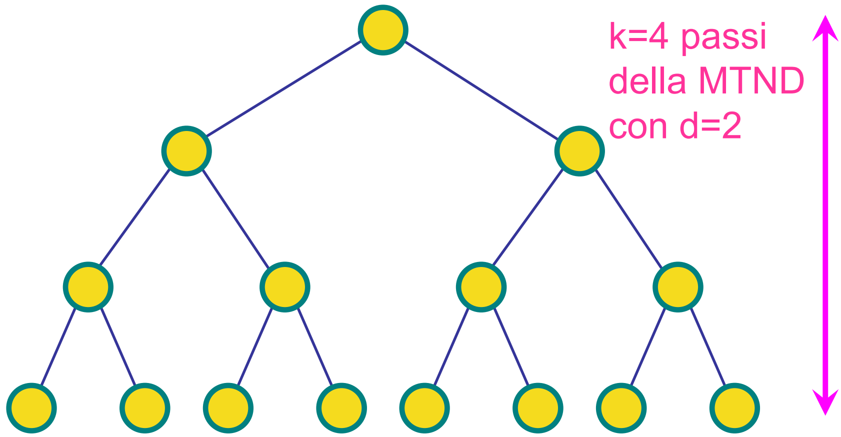
  le transizioni di ogni insieme $\delta_N(q,\sigma)$ sono numerate da 1 a d

  ogni sequenza di lunghezza s sul nastro 2 è in corrispondenza con una computazione di M di s passi

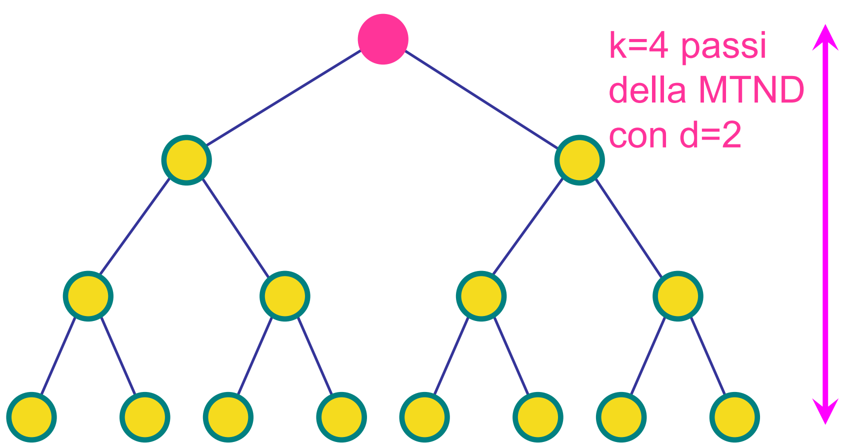
  gli s numeri di ogni sequenza (compresi tra 1 e d) sono usati per scegliere ad ogni passo una transizione tra le d possibili

  es. se s=4 e d=2 e la sequenza e' 2122 M' sceglie per la prima mossa la seconda transizione disponibile, per la seconda mossa la prima, ecc....

costo della simulazione

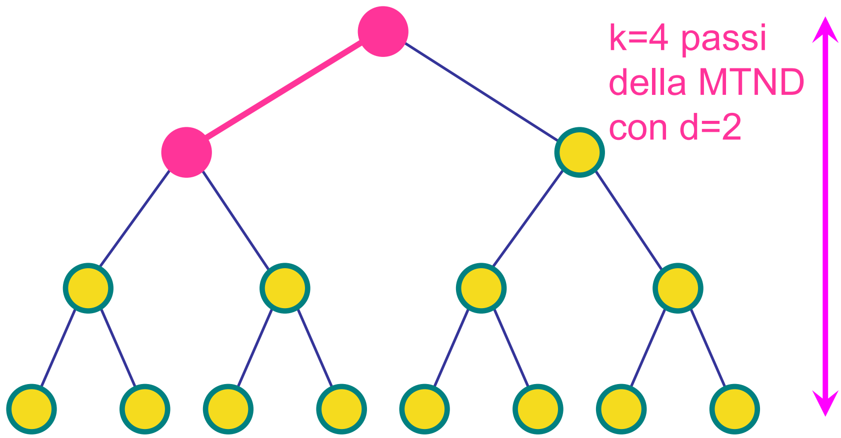


costo della simulazione



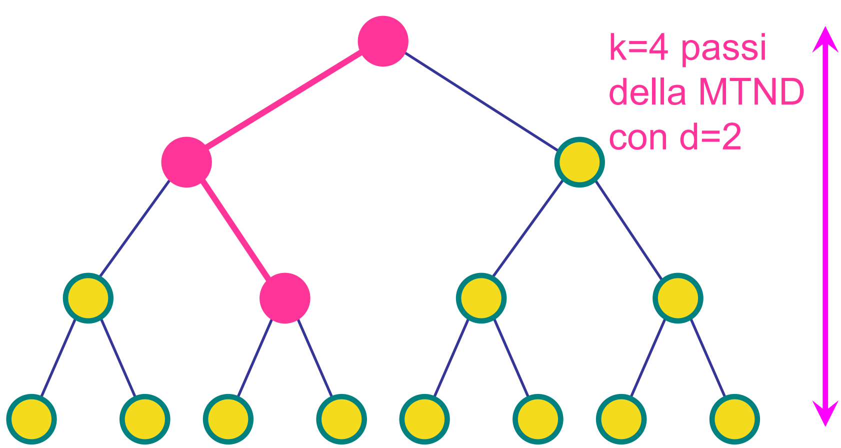
per simulare il primo passo:
tutti i cammini di lunghezza 0

costo della simulazione



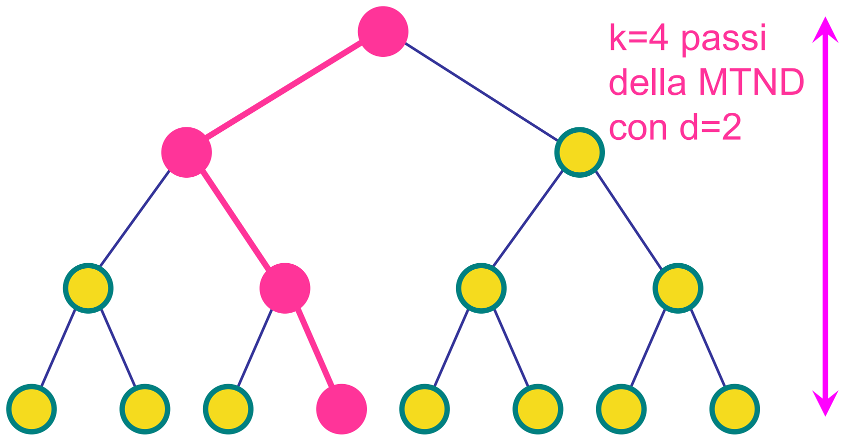
per simulare il secondo passo:
tutti i cammini di lunghezza 1

costo della simulazione



per simulare il terzo passo:
tutti i cammini di lunghezza 2

costo della simulazione



per simulare il quarto passo:
tutti i cammini di lunghezza 3

dimostrazione(MTND→MT)

se su qualche foglia dell'albero di computazione di M c'e' uno stato finale, allora M' lo raggiunge in tempo finito

altrimenti M' non raggiunge mai uno stato finale

se M termina in k passi M' ha bisogno di

$$O\left(\sum_{j=0}^{k} jd^j\right) = O(kd^k) \text{ passi}$$

osservazione: polinomialità ed esponenzialità nella simulazione

alla ricerca del modello di calcolo minimale: ulteriori riduzioni delle MT

teorema: per ogni MT esiste una MT con nastro semi-infinito equivalente

dimostrazione: simulazione con nastro a 3 tracce

il nastro infinito viene piegato in due

teorema: per ogni MT $M=<\Sigma,\underline{b},K,q_0,F,\delta>$ esiste una MT $M'=<\Sigma',\underline{b},K',q_0',F',\delta'>$ con $|\Sigma'|=1$ equivalente

dimostrazione: codifica dei caratteri di $\Sigma_{\underline{b}}$ con i caratteri di $\Sigma'_{\underline{b}}$

descrizione linearizzata delle MT

uso di macchine per realizzare nuove macchine:

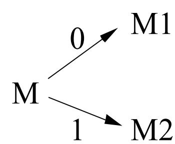
composizione delle macchine usando diramazioni condizionate

esempio: siano date le MT: M1 scrive 1, M2 scrive 0, M calcola $q_0 x\sigma_i \vdash^* x q_F \sigma_i$ con $\sigma_i \in \Sigma$ e $\Sigma=\{0,1\}$ e $x \in \Sigma^*$

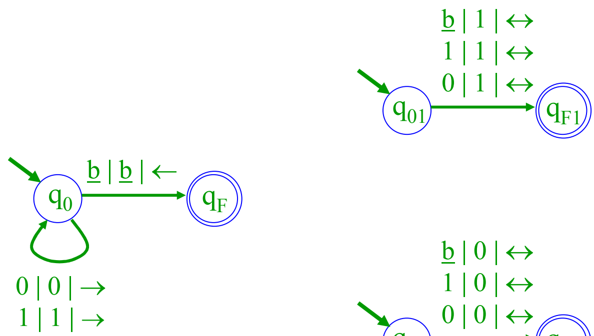
se, quando M si ferma, vogliamo far scrivere 1 al posto di σ_i possiamo collegare lo stato finale di M con quello iniziale di M1

se invece vogliamo far partire M1 o M2 a seconda che σ_i sia 0 o 1 possiamo fare una diramazione condizionata

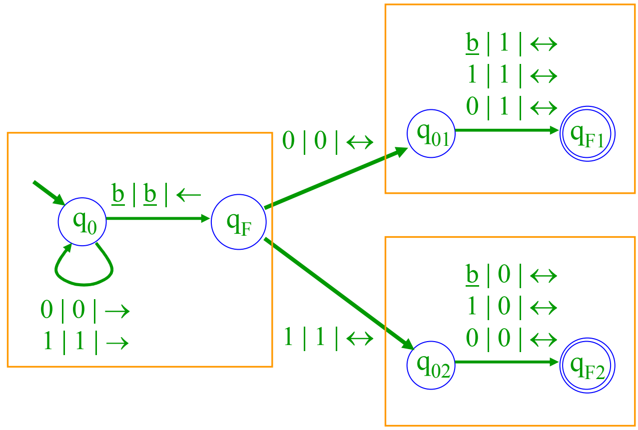
rappresentazione schematica:



composizione delle MT per diramazione condizionata



composizione delle MT per diramazione condizionata



descrizione linearizzata delle MT

diramazione verso varie macchine:

siano

$M=\langle\Sigma,\underline{b},K,q_0,F,\delta\rangle$ e $M_i=\langle\Sigma,\underline{b},K_i,q_{0i},F_i,\delta_i\rangle$

$(i=1,....,n)$, $n\geq|\Sigma|$, $\Sigma=\{\sigma_1,....,\sigma_r\}$

n+1 MT con K ed i K_i disgiunti a coppie

$M'=\langle\Sigma,\underline{b},K',q_0',F',\delta'\rangle$

è ottenuta dalle M e M_i per diramazione se

$K' = K\cup K_1\cup....\cup K_n$

$q_0'=q_0$

$F' = F_1\cup....\cup F_n$

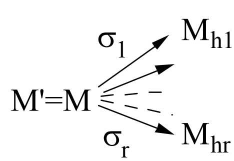
δ' è l'unione delle funzioni di transizione

per ogni $q\in F$ $\delta'(q,\sigma_j)=\langle q_{0hj},\sigma_j,i\rangle$

descrizione linearizzata delle MT

diramazione verso varie macchine

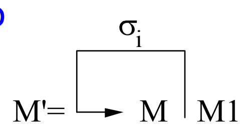
rappresentazione schematica:



un interessante caso particolare:

M coincide con una delle macchine verso cui si dirama e tutte le altra coincidono

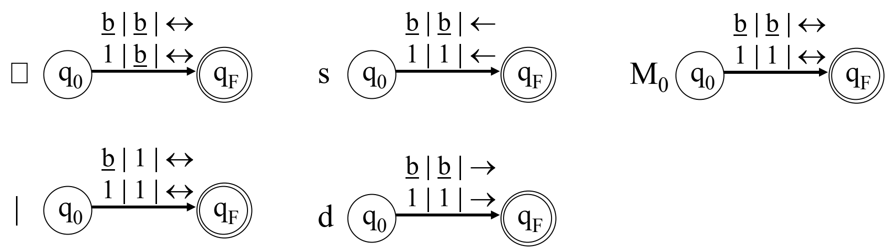
rappresentazione schematica:



MT elementari con 1 nastro e 2 caratteri:

| scrive 1, □ scrive $\underline{b}$, d sposta la testina a destra, s sposta la testina a sinistra, M_0 non fa nulla

MT elementari



□ = scrive un b
| = scrive un 1

s = mossa a sinistra
d = mossa a destra

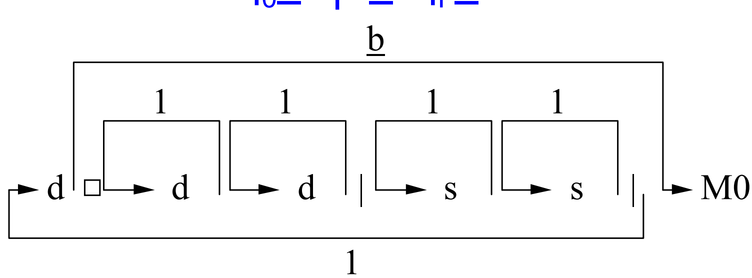
M_0 = ferma

descrizione linearizzata delle MT

esempio: sintesi di una macchina complessa con macchine elementari

sintetizziamo la MT che esegue

$$q_0\underline{b}x \vdash^* \underline{b}xq_F\underline{b}x$$



descrizione linearizzata delle MT

teorema: ogni MT è ottenibile per composizione per diramazione di macchine elementari e ogni macchina ottenuta per composizione di macchine elementari è una MT

dimostrazione:

la composizione di diagrammi di stato di MT restituisce diagrammi di stato di MT, cio' prova la seconda parte

per provare la prima parte associamo ad ogni stato la composizione di alcune macchine elementari

(| □ d s M0)

dim. (MT→macchine elementari)

allo stato q_i della MT con $\delta(q_i,1)=<q_h,\sigma_h,t_h>$ e $\delta(q_i,\underline{b})=<q_k,\sigma_k,t_k>$ $\sigma_h,\sigma_k \in \{1,\underline{b}\}$ $t_h,t_k \in \{s,d,i\}$ associamo:

$$\begin{tikzcd} & ShTh \\ M0 \arrow[ur, "1"] \arrow[dr, "\underline{b}"'] & \\ & SkTk \end{tikzcd}$$

con $S_h=|$ se $\sigma_h=1$ $S_h=□$ se $\sigma_h=\underline{b}$

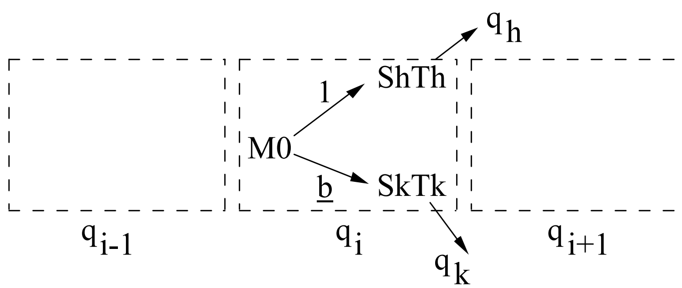
$S_k=|$ se $\sigma_k=1$ $S_k=□$ se $\sigma_k=\underline{b}$

$T_h=d$ se $t_h=d$ $T_h=s$ se $t_h=s$ $T_h=i$ se $t_h=i$

$T_k=d$ se $t_k=d$ $T_k=s$ se $t_k=s$ $T_k=i$ se $t_k=i$

dim. (MT→macchine elementari)

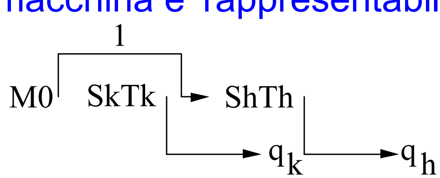
colleghiamo inoltre:



descrizione linearizzata delle MT

- osservazione: per realizzare una macchina con n stati occorrono al piu' 5n macchine elementari
- osservazione: ogno stato e' rappresentabile nella forma linearizzata

  quindi ogni macchina e' rappresentabile in forma lienarizzata



- convenzione: l'ultima macchina in un diagramma è M0
- a questo punto e' possibile attribuire un nome ad ogni macchina ed usare i nomi per comporre altre macchine

MT universale

una MT $U=<\Sigma,\underline{b},K',q_0',F',\delta'>$, che calcola la funzione u, si dice universale quando data una qualunque MT $M=<\Sigma,\underline{b},K,q_0,F,\delta>$, esiste una codifica c_M di M in Σ tale che se se M calcola

$$m: (\Sigma^*)^n \to \Sigma^*$$

con $m(x_1,\dots,x_n)=y$
se $f_M(x_1\underline{b}\dots\underline{b}x_n\underline{b})=\underline{b}y\underline{b}$,
altrimenti indefinita

allora

$$u(c_M,x_1,\dots,x_n)=m(x_1,\dots,x_n)$$

U puo' simulare qualunque MT

dimostriamo l'esistenza di una U che sa simulare MT con $|\Sigma|=1$ e nastro seminfinito

osservazione: l'ipotesi non e' restrittiva!

MT universale

teorema: esiste una MT $U=<\Sigma',\underline{b},K',q_0',F',\delta'>$ tale che data ogni MT $M=<\{1\},\underline{b},K,q_0,F,\delta>$

$$f_U(c_M\underline{b}x) = \underline{b}^h f_M(x)\underline{b}^k \quad h,k\geq 0 \quad x\in\{1,\underline{b}\}^*$$

dimostrazione:

codifica di M con $\Sigma=\{1\}$

ulteriore riduzione sull'insieme delle macchine necessarie

eliminazione dei salti condizionati su $\underline{b}$

simulazione

dimostrazione (simulazione di M con U)

ulteriore riduzione sull'insieme delle macchine necessarie

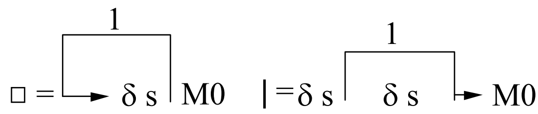
due macchine fondamentali:

δ cambia il carattere letto e va a destra

s va a sinistra

abbiamo che

$d = \delta\, s\, \delta\, M_0$



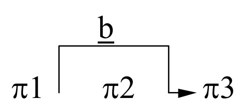
inoltre eliminiamo M_0 ovunque ma non in ultima posizione

rimangono solo δ e s

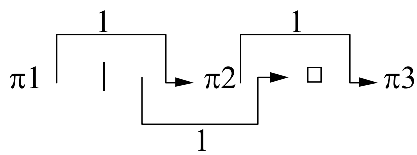
dimostrazione (simulazione di M con U)

eliminazione dei salti condizionati su <u>b</u>

ad esempio



diventa



c_M e' quindi organizzata come segue

1	per	s
11	per	δ
1....1(n+3)	per	salto all'istruzione n

dimostrazione (simulazione di M con U)

inoltre

una istruzione è separata con b dalla successiva

caratteri speciali per indicare l'istruzione attualmente in esecuzione ed il carattere attualmente esaminato da M: 1' e b'

un salto ad istruzione non presente arresta la computazione

configurazione iniziale

1....1b1....1b1....1b111b............b

c_M x

configurazione finale

b.........................bb............b

$f_M(x)$

MT universale

- osservazione: possiamo interpretare la MT universale come un calcolatore e M e x come un programma e i suoi dati
- osservazione: possiamo interpretare la MT universale come un interprete di un linguaggio di programmazione