Automi e Linguaggi Formali

Macchine di Turing, indecidibilità e problemi intrattabili

Lamberto Ballan
lamberto.ballan@unipd.it

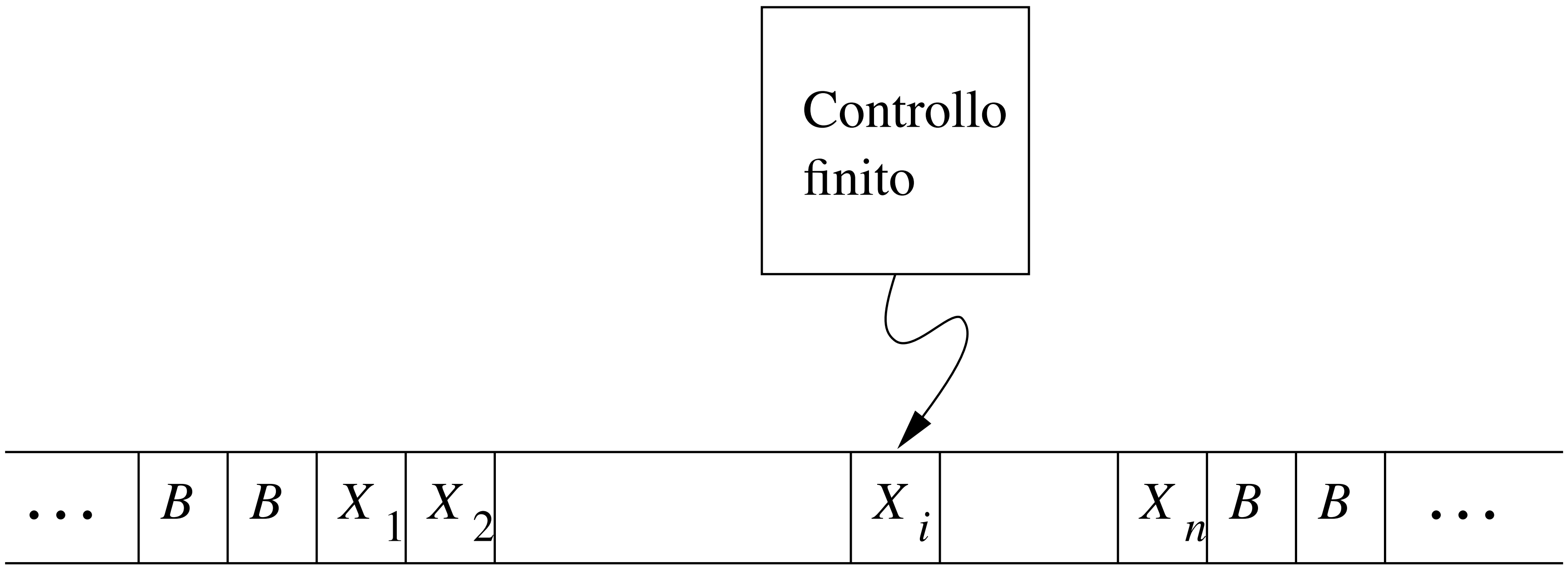
Università degli Studi di Padova

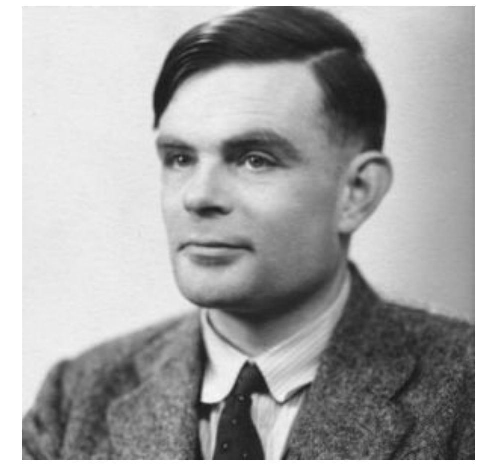
Padova, 12 Maggio 2017

Comunicazioni / calendario delle lezioni

- Settimana prossima faremo lezione soltanto di martedì (13:30-15-30) e venerdì (13:30-15:30)
- Prima dell'inizio della settimana assegnerò degli esercizi sul Moodle del corso; durante la lezione del prossimo venerdì svolgeremo/correggeremo alcuni degli esercizi
- Ufficio: Dipartimento di Matematica, Torre Archimede, III piano corridoio CD, stanza 322 (è lo stesso del Prof. Bresolin)

La macchina di Turing (1937)





Una macchina di Turing esegue una **mossa** in funzione del suo stato e del simbolo che si trova nella cella visitata dalla testina di lettura del nastro.

In una mossa, la macchina di Turing (TM) compie tre azioni:

- Cambia stato
- Scrive un simbolo di nastro nella cella che visita
- Muove la testina verso sinistra o verso destra

Definiamo una TM che calcola la *sottrazione propria*

- Vediamo un altro esempio in cui abbiamo una TM con "output"
  - oggi si pensa alle TM come dispositivi per riconoscere linguaggi; Turing considerava la sua macchina come un calcolatore di funzioni a valori interi
- La seguente TM calcola la *sottrazione propria* tra m e n, cioè:

$$m \dot{-} n = \max(m - n, 0)$$

- I numeri interi sono rappresentati in codice unario (come blocchi di un singolo carattere, *0* o *1*)
- La macchina computa cambiando le lunghezze dei blocchi o costruendo nuovi blocchi altrove

Definiamo una TM che calcola la *sottrazione propria*

- La macchina *M* parte con un nastro su cui é presente la stringa di input 0^m10^n preceduta e seguita da *B;* si arresta con $0^{m \dot{-} n}$ sul nastro, circondata da *B*

- Partendo dall'estremità sinistra dell'input, la TM:

  - trova lo 0 residuo più a sinistra e lo sostituisce con *B*

  - si sposta verso destra alla ricerca di un 1; trovato un 1 continua a destra finché non trova 0 e lo sostituisce con 1

  - torna verso sinistra cercando lo 0 più a sinistra (lo identifica quando trova *B* e poi muove di una cella a destra)

Definiamo una TM che calcola la *sottrazione propria*

- La TM si arresta in uno dei seguenti casi:
  - si arresta se cercando uno 0 a destra trova un B; quindi gli *n* 0 sono già stati trasformati in 1 e *n+1* di *m* simboli 0 in *B*
    - allora la TM sostituisce gli *n+1* simboli 1 con *B* e il primo *B* alla loro sinistra con 0
  - se iniziando il ciclo non riesce a trovare uno 0 da mutare in B; quindi i primi *m* simboli 0 sono già stati trasformati in B
    - allora la TM sostituisce tutti gli 1 e 0 residui con B e finisce con un nastro completamente vuoto

Definiamo una TM che calcola la *sottrazione propria*

- La TM che calcola tale operazione é specificata da

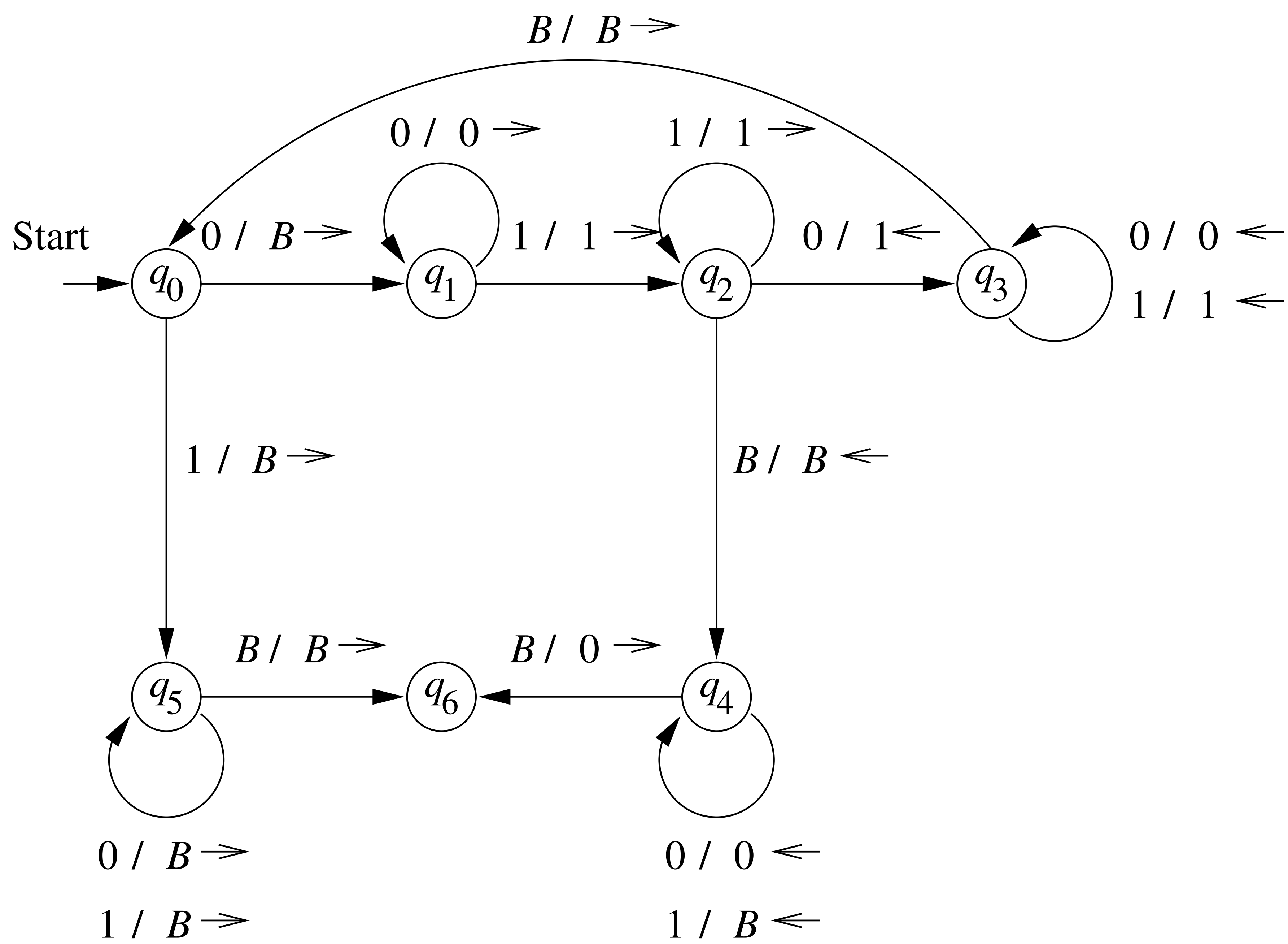
$$M = (\{q_0, q_1, \ldots, q_6\}, \{0, 1\}, \{0, 1, B\}, \delta, q_0, B)$$

- La TM é definita dalla tabella di transizione:

	0	1	B
$\rightarrow q_0$	(q_1, B, R)	(q_5, B, R)	
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
$*q_6$			

Definiamo una TM che calcola la *sottrazione propria*

Il diagramma di transizione è



Esercizi

- Scrivere le ID della TM appena definita quando il nastro di input contiene la sequenza di simboli (in codifica unaria):
  - 0010 *(cioè $2 \dot{-} 1 = 1$)*
  - 0100 *(cioè $1 \dot{-} 2 = 0$)*
  - 000010 *(cioè $4 \dot{-} 1 = 3$)*

Il linguaggio di una TM, le TM e l'arresto

1. Una TM **si arresta** se entra in uno stato q guardando un simbolo di nastro X e non ci sono mosse possibili, cioè $\delta(q, X)$ non è definita
2. Se una TM accetta una stringa, possiamo assumere che si arresti (basta rendere indefinito $\delta(q, X)$ per ogni q accettante)
3. Se non accetta, non possiamo fare in modo che si arresti
4. Linguaggi **ricorsivi**: esiste una TM che si arresta su ogni stringa (sia accettata che non accettata)
5. Linguaggi **ricorsivamente enumerabili**: esiste una TM che si arresta se la stringa è accettata
6. Problema **decidibile**: esiste una TM che si arresta sempre

TM che accetta L delle stringhe con stesso num 0,1

- **Esercizio:** Definire la TM che accetta il linguaggio L costituito dalle stringhe con stesso numero di 0 e 1 (es. 8.2.2a del libro)
  - {01,10,0011,0101,1001,1010,1100,0110,…}

- La TM usa i simboli *X* e *Y* per sostituire "coppie" di 0 e 1
  - *X* garantisce che non ci sono 0 o 1 alla sua sinistra (quindi la testina non può muoversi a sinistra di una *X*)
  - Y invece può avere 0 o 1 alla sua sinistra