

# Algoritmo Generale

## STEP 1: Analisi del Problema

Input: "Data una TM  $M$ , determinare se  $M$  ha la proprietà  $P$ "

Output: Linguaggio  $L_P = \{\langle M \rangle \mid M \text{ ha la proprietà } P\}$

Template di definizione:

$L_P = \{\langle M \rangle \mid M \text{ è una TM e } [\text{CONDIZIONE SPECIFICA}]\}$

## STEP 2: Classificazione della Proprietà

Checklist di identificazione:

- ☐ **Proprietà del linguaggio riconosciuto**  $\rightarrow$  Riduzione da  $A_{TM}$
- ☐ **Proprietà temporale/computazionale**  $\rightarrow$  Riduzione da  $HALT_{TM}$
- ☐ **Proprietà di vuotezza**  $\rightarrow$  Riduzione da  $E_{TM}$
- ☐ **Proprietà di finitudine**  $\rightarrow$  Riduzione da  $A_{TM}$  o  $HALT_{TM}$

## STEP 3: Scelta della Riduzione

Matrice decisionale:

Se  $P$  riguarda contenuto di  $L(M)$ :  $A_{TM} \leq_m L_P$

Se  $P$  riguarda terminazione:  $HALT_{TM} \leq_m L_P$

Se  $P$  riguarda  $L(M) = \emptyset$ :  $E_{TM} \leq_m L_P$

ALTERNATIVAMENTE  $\rightarrow EQ_{TM} = M_1$  ed  $M_2$  che unite accettano il linguaggio.

## STEP 4: Template di Costruzione della Macchina $M'$

Pattern di Esempio 1 (da  $A_{TM}$ ):

$F$  = Funzione di riduzione = Su input  $\langle M, w \rangle$

$M'$  = "Su input  $x$ : (ugualmente: "Simula  $M'$  su  $x$ ")

1. [Ignora  $x$  / Usa  $x$  secondo necessità]
2. Simula  $M$  su  $w$
3. Se  $M$  accetta  $w \rightarrow$  [COMPORTAMENTO CHE SODDISFA  $P$ ]
4. Se  $M$  rifiuta  $w \rightarrow$  [COMPORTAMENTO CHE NON SODDISFA  $P$ ]"

## Pattern di Esempio 2 (da HALT\_TM):

```
F = Funzione di riduzione = Su input  $\langle M, w \rangle$   
M' = "Su input x: (ugualmente: "Simula M' su x")  
    1. Simula M su w per  $|x|$  passi  
    2. Se M termina  $\rightarrow$  [COMPORTAMENTO CHE SODDISFA P]  
    3. Se M non termina  $\rightarrow$  [COMPORTAMENTO CHE NON SODDISFA P]"
```

## STEP 5: Template di Analisi dei Casi

### Struttura standard:

```
Caso 1:  $\langle M, w \rangle \in A_{TM}$   
 $\Rightarrow$  [Conseguenza su M']  
 $\Rightarrow$  [Effetto su  $L(M')$ ]  
 $\Rightarrow$  [Verifica proprietà P]  
 $\Rightarrow \langle M' \rangle \in L_P$   
  
Caso 2:  $\langle M, w \rangle \notin A_{TM}$   
 $\Rightarrow$  [Conseguenza su M']  
 $\Rightarrow$  [Effetto su  $L(M')$ ]  
 $\Rightarrow$  [Verifica negazione di P]  
 $\Rightarrow \langle M' \rangle \notin L_P$ 
```

## STEP 6: Template di Verifica

### Checklist di correttezza:

```
✓  $f(\langle M, w \rangle) = \langle M' \rangle$  è computabile  
✓  $\langle M, w \rangle \in A_{TM} \Rightarrow \langle M' \rangle \in L_P$   
✓  $\langle M, w \rangle \notin A_{TM} \Rightarrow \langle M' \rangle \notin L_P$   
✓ Equivalenza:  $\langle M, w \rangle \in A_{TM} \Leftrightarrow \langle M' \rangle \in L_P$ 
```

---

## Pattern Specifici per Proprietà Comuni

### Tipo 1: "M accetta almeno una stringa di tipo T"

```
M' ignora input, simula M su w:  
- Se M accetta w  $\rightarrow$  M' accetta tutte le stringhe di tipo T  
- Se M rifiuta w  $\rightarrow$  M' non accetta mai
```

## Tipo 2: "M accetta solo stringhe di tipo T"

M' su input x:

- Se x non è di tipo T  $\rightarrow$  rifiuta subito
- Se x è di tipo T  $\rightarrow$  simula M su w, accetta sse M accetta w

## Tipo 3: "L(M) ha proprietà quantitativa Q"

M' costruita per avere:

- $L(M')$  con proprietà Q  $\Leftrightarrow$  M accetta w
- $L(M')$  senza proprietà Q  $\Leftrightarrow$  M non accetta w

## Tipo 4: "M scrive pattern specifico su nastro"

M' su input x:

- Simula M su w
- Se M accetta  $\rightarrow$  scrive il pattern richiesto e accetta
- Se M rifiuta  $\rightarrow$  non scrive mai il pattern

---

## Formula di Conclusione Standard

Poiché  $A_{TM}$  è indecidibile e abbiamo mostrato  $A_{TM} \leq_m L_P$  tramite una riduzione computabile, segue che  $L_P$  è indecidibile.  $\square$

**Nota:** Sostituire  $A_{TM}$  con  $HALT_{TM}$ ,  $E_{TM}$ , etc. secondo la riduzione scelta.