Automi e Linguaggi Formali – Appello 5/7/2021

Seconda parte – Decidibilità e complessità

Esercizio 1: Simulazione di MT a testine multiple

Teorema

Qualsiasi macchina di Turing a testine multiple può essere simulata da una macchina di Turing deterministica a nastro singolo.

Dimostrazione

Definizione formale della MT a testine multiple: Una MT a k testine $M_k = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$ dove: $\delta : Q \times \Gamma^k \to Q \times \Gamma^k \times \{L,R\}^k$

Costruzione della MT simulante M_1:

Idea: Codificare le k testine su un singolo nastro usando marcatori speciali.

Alfabeto esteso: $\Gamma' = \Gamma \cup \{\hat{\gamma} \mid \gamma \in \Gamma\} \cup \{\#\}$

dove $\hat{\gamma}$ rappresenta il simbolo γ con testina sopra, e # è separatore.

Codifica del nastro: I contenuti dei k nastri:

```
Nastro 1: a₁a₂a₃...
Nastro 2: b₁b₂b₃...
...
Nastro k: z₁z₂z₃...
```

vengono codificati come:

```
#a₁a₂â₃...#b₁b̂₂b₃...#...#z₁z₂ẑ₃...#
```

Algoritmo di simulazione:

Fase 1 - Scansione e lettura:

1. Scansiona il nastro da sinistra a destra

2. Per ogni sezione i-esima (tra #), trova il simbolo marcato $\hat{\gamma}_i$
3. Memorizza nella tabella degli stati: $(\gamma_1, \gamma_2, ..., \gamma_k)$

Fase 2 - Calcolo della transizione: 4. Calcola $\delta(q\_current, \gamma_1, ..., \gamma_k) = (q\_new, \beta_1, ..., \beta_k, d_1, ..., d_k)$

Fase 3 - Aggiornamento: 5. Scansiona nuovamente il nastro 6. Per ogni sezione i:

- Sostituisci $\hat{\gamma}_i$ con β_i (rimuove marcatura)
- Sposta la marcatura secondo d_i
- Gestisci estensioni del nastro se necessario

Correttezza: La configurazione del nastro di M_1 codifica univocamente la configurazione di M_k. Ogni transizione di M_k è simulata correttamente da una sequenza finita di transizioni di M_1.

Complessità: Se M_k termina in tempo T(n), allora M_1 termina in tempo $O(k^2T(n))$. ■

Esercizio 2: Indecidibilità di L_2

Teorema

Il linguaggio $L_2 = \{\langle M, w\rangle \mid M$ accetta la stringa ww^R$\}$ è indecidibile.

Dimostrazione

Dimostriamo per riduzione da ACCEPT_TM = {⟨M, w⟩ | M accetta w}, che è indecidibile.

Costruzione della riduzione:

Data un'istanza $\langle M_0, w_0\rangle$ di ACCEPT_TM, costruiamo ⟨M', w'⟩ per L_2:

Costruzione di M':

```
M' = "Su input x:
1. Ignora completamente l'input x
2. Simula M₀ su w₀
3. Se M₀ accetta w₀, allora accetta
4. Se M₀ rifiuta w₀, allora rifiuta
5. Se M₀ non termina su w₀, allora non terminare"
```

Scelta di w': Poniamo w' = ε (stringa vuota). Quindi w'w'^R = ε.

Verifica della correttezza:

Caso 1: $\langle M_0, w_0 \rangle \in$ ACCEPT_TM

- M_0 accetta $w_0 \Longrightarrow$ M' accetta ogni input (incluso ε)
- $\Longrightarrow$ M' accetta w'w'^R = ε
- $\Longrightarrow \langle M', \varepsilon \rangle \in L_2$

Caso 2: $\langle M_0, w_0 \rangle \notin$ ACCEPT_TM

- M_0 non accetta $w_0 \Longrightarrow$ M' non accetta alcun input
- $\Longrightarrow$ M' non accetta ε
- $\Longrightarrow \langle M', \varepsilon \rangle \notin L_2$

Conseguenza: Se L_2 fosse decidibile, potremmo decidere ACCEPT_TM, contraddicendo la sua indecidibilità.

Quindi L_2 è indecidibile. ■

Esercizio 3: PebbleDestruction è NP-hard

Teorema

PebbleDestruction è NP-hard.

Dimostrazione

Dimostriamo per riduzione dal **Circuito Hamiltoniano** (NP-completo).

Costruzione della riduzione:

Dato G = (V, E) con |V| = n, costruiamo un'istanza di PebbleDestruction:

Grafo G':

- V' = V ∪ {s} (s è nuovo vertice "sink")
- E' = E ∪ {(v, s) | v ∈ V}

Distribuzione ciottoli:

- p'(v) = 3 per ogni v ∈ V
- p'(s) = 0

- Totale: 3n ciottoli
- Obiettivo: ridurre a 1 ciottolo

Direzione (⇒): Se G ha circuito Hamiltoniano $C = v_1 \to v_2 \to \ldots \to v_n \to v_1$

Strategia di soluzione:

Fase 1 - Consolidamento lungo C (n mosse):

```
Per i = 1 to n:
  Da vᵢ rimuovi 2 ciottoli, aggiungi 1 a vᵢ₊₁
```

Risultato: ogni v_i ha 2 ciottoli (totale 2n)

Fase 2 - Raccolta in v_1 (n-1 mosse):

```
Seguendo C⁻¹: consolidare tutti i ciottoli in v₁
```

Risultato: v_1 ha n ciottoli

Fase 3 - Trasferimento finale (n-1 mosse):

```
Da v₁ trasferisci tutto a s
```

Risultato: 1 ciottolo in s

Totale mosse: $3n-2 < 3n-1$ (limite energetico) ✓

Direzione (⇐): Se PebbleDestruction ha soluzione ma G non ha circuito Hamiltoniano

Senza struttura Hamiltoniana, ogni strategia di consolidamento richiede > 3n-1 mosse per gestire:

- Ponti critici
- Vertici di taglio
- Back-tracking necessario

Questo viola il vincolo energetico (ogni mossa riduce il totale di 1).

Conclusione: Circuito Hamiltoniano $\leq_p$ PebbleDestruction

Quindi PebbleDestruction è NP-hard. ■