Tutorato di Automi e Linguaggi Formali

Homework 11: NP-Completezza ed NP-Hard

Gabriel Rovesti
Corso di Laurea in Informatica - Università degli Studi di Padova

Tutorato 11 - 04-06-2025

1 Classi di Complessità P e NP

Esercizio 1. Considerare i seguenti problemi:

- $PATH = \{\langle G, s, t\rangle \mid G$ è un grafo che contiene un cammino da s a $t\}$
- $HAMPATH = \{\langle G, s, t\rangle \mid G$ è un grafo che contiene un cammino Hamiltoniano da s a $t\}$
- $RELPRIMES = \{\langle x, y\rangle \mid \gcd(x, y) = 1\}$

a) Dimostrare che $PATH \in P$ fornendo un algoritmo deterministico polinomiale esplicito e analizzandone la complessità.

b) Dimostrare che $HAMPATH \in NP$ costruendo un verificatore polinomiale. Specificare il formato del certificato e l'algoritmo di verifica.

c) Dimostrare che $RELPRIMES \in P$ utilizzando l'algoritmo di Euclide. Analizzare la complessità in termini della rappresentazione binaria degli input.

d) Spiegare perché $HAMPATH$ è considerato intrattabile mentre $PATH$ è trattabile, nonostante la loro apparente similarità.

Esercizio 2. Sia $DOMINO[2] = \{D \mid D$ è un insieme di tessere del domino tale che si possono disporre alcune tessere in fila in modo che ogni numero compaia esattamente due volte$\}$.

a) Dimostrare che $DOMINO[2] \in NP$ costruendo un verificatore polinomiale appropriato.

b) Confrontare la complessità di $DOMINO[2]$ con quella di $DOMINO[1]$ (disporre tutte le tessere in fila con numeri adiacenti corrispondenti). Spiegare perché $DOMINO[1] \in P$ mentre $DOMINO[2]$ è presumibilmente intrattabile.

c) Analizzare come piccole modifiche nella definizione di un problema possano cambiarne drasticamente la complessità computazionale.

2 Riduzioni Polinomiali

Esercizio 3. Dimostrare le seguenti riduzioni polinomiali fondamentali:

a) $CircuitSAT \leq_p SAT$: Costruire una riduzione che trasformi un circuito booleano in una formula proposizionale equivalente. Specificare la trasformazione per ogni tipo di porta logica (AND, OR, NOT).

b) $SAT \leq_p 3SAT$: Fornire un algoritmo che converta una formula booleana arbitraria in una formula in 3-CNF soddisfacibile se e solo se la formula originale è soddisfacibile.

c) Dimostrare che entrambe le riduzioni operano in tempo polinomiale, analizzando la crescita della dimensione dell'output rispetto all'input.

d) Spiegare l'importanza di queste riduzioni nella teoria della NP-completezza.

Esercizio 4. Considerare la riduzione $3SAT \leq_p MAXINDSET$ (Insieme Indipendente Massimo).

a) Descrivere dettagliatamente la costruzione del grafo G a partire da una formula ϕ in 3-CNF. Specificare come vengono creati i vertici e gli archi.

b) Dimostrare che ϕ è soddisfacibile se e solo se G ha un insieme indipendente di dimensione m (dove m è il numero di clausole in ϕ).

c) Analizzare la complessità temporale della riduzione in termini del numero di variabili e clausole.

d) Utilizzare questa riduzione per concludere che $MAXINDSET$ è NP-completo.

3 Problemi NP-Completi su Grafi

Esercizio 5. Sia $VERTEXCOVER = \{\langle G, k\rangle \mid G$ ha una copertura di vertici di dimensione al più $k\}$.

a) Dimostrare che $VERTEXCOVER \in NP$ fornendo un verificatore polinomiale esplicito.

b) Dimostrare che $MAXINDSET \leq_p VERTEXCOVER$ utilizzando la relazione: I è un insieme indipendente in G se e solo se $V \setminus I$ è una copertura di vertici di G.

c) Concludere che $VERTEXCOVER$ è NP-completo utilizzando la riduzione precedente.

d) Confrontare questo risultato con il fatto che il problema della copertura di archi (maximum matching) è in P. Spiegare la differenza fondamentale.

Esercizio 6. Considerare il problema del circuito Hamiltoniano: $HAMCYCLE = \{\langle G \rangle \mid G$ ha un circuito Hamiltoniano$\}$.

a) Dimostrare che $HAMCYCLE \in NP$ costruendo un verificatore appropriato.

b) Spiegare perché $HAMCYCLE$ è NP-completo mentre il problema del circuito Euleriano è in P. Analizzare le differenze strutturali tra i due problemi.

c) Discutere l'algoritmo di Fleury per il circuito Euleriano e spiegare perché un approccio simile non funziona per il circuito Hamiltoniano.

d) Analizzare le implicazioni pratiche di questa differenza di complessità per problemi di routing e ottimizzazione.

4 Strategie per Dimostrare NP-Completezza

Esercizio 7. Considerare il problema $3COLORING = \{\langle G \rangle \mid G$ è un grafo colorabile con 3 colori$\}$.

a) Seguire lo schema standard di dimostrazione di NP-completezza:

- Dimostrare che $3COLORING \in NP$
- Costruire una riduzione $3SAT \leq_p 3COLORING$
- Provare la correttezza della riduzione
- Analizzare la complessità temporale

b) Spiegare perché $2COLORING \in P$ mentre $3COLORING$ è NP-completo. Costruire un algoritmo polinomiale per $2COLORING$.

c) Discutere le strategie per scegliere il problema di partenza nelle riduzioni, utilizzando le linee guida presentate nella lezione.

Esercizio 8. Definire il problema $SUBSETSUM = \{(S, t) \mid S$ è un insieme di interi positivi e esiste un sottoinsieme $S' \subseteq S$ tale che $\sum_{x \in S'} x = t\}$.

a) Dimostrare che $SUBSETSUM$ è NP-completo costruendo una riduzione appropriata da $3SAT$.

b) Nella costruzione della riduzione, spiegare come:

- Rappresentare variabili booleane come interi
- Codificare clausole nella somma target
- Garantire che la riduzione preservi soddisfacibilità

c) Analizzare l'esistenza di algoritmi pseudo-polinomiali per $SUBSETSUM$ e discutere la differenza tra complessità forte e debole.

d) Confrontare con il problema della somma esatta vs. il problema della somma approssimata.

5 Problemi Avanzati di NP-Completezza

Esercizio 9. Definire il problema $PARTITION = \{S \mid S$ è un insieme di interi positivi che può essere partizionato in due sottoinsiemi con la stessa somma$\}$.

a) Dimostrare che $PARTITION$ è NP-completo riducendo da $SUBSETSUM$.

b) Analizzare la relazione tra $PARTITION$ e altri problemi classici di programmazione dinamica.

c) Discutere l'esistenza di algoritmi pseudo-polinomiali per $PARTITION$ e le implicazioni per la complessità parametrizzata.

d) Estendere l'analisi al problema $3PARTITION$ e discutere perché è fortemente NP-completo.