Automi e Linguaggi Formali

a.a. 2017/2018

LT in Informatica
29 Maggio 2018

Università degli Studi di Padova

I grafi e le loro applicazioni

- I grafi sono strutture dati che vengono usate estensivamente in informatica
- Ci sono migliaia di problemi computazionali che sono importanti per le applicazioni e che si possono modellare con i grafi.
- In questa lezione vedremo che cos'è un grafo, e studieremo alcuni problemi sui grafi che sono interessanti per la loro classe di complessità.

Definizioni di base

Un grafo è definito da un'insieme di nodi (o vertici) e da un'insieme di archi che collegano i nodi. Introduciamo subito una prima distinzione tra due tipi di grafo.

Definition (Grafo non orientato)

Un grafo non orientato (detto anche indiretto) G è una coppia (V, E) dove:

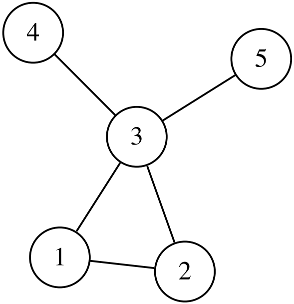
- $V = \{v_1, v_2, \ldots, v_n\}$ è un insieme finito e non vuoto di vertici;
- $E \subseteq \{\{u, v\} \mid u, v \in V\}$ è un insieme di coppie non ordinate, ognuna delle quali corrisponde ad un arco non orientato del grafo.

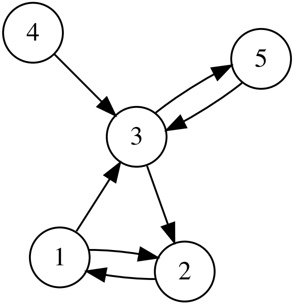
Definition (Grafo orientato)

Un grafo orientato (detto anche diretto) G è una coppia (V, E) dove:

- $V = \{v_1, v_2, \ldots, v_n\}$ è un insieme finito e non vuoto di vertici;
- $E \subseteq V \times V$ è un insieme di coppie ordinate (u, v) tali che $u \neq v$, ognuna delle quali corrisponde ad un arco orientato del grafo.

Due esempi di grafo





Il problema del massimo insieme indipendente

- Sia $G = (V, E)$ un grafo arbitrario.
- Un insieme indipendente in G è un sottoinsieme I dei vertici tali che per ogni coppia di vertici in I, non c'è nessun arco che li collega

Il problema del massimo insieme indipendente

- Sia $G = (V, E)$ un grafo arbitrario.
- Un insieme indipendente in G è un sottoinsieme I dei vertici tali che per ogni coppia di vertici in I, non c'è nessun arco che li collega

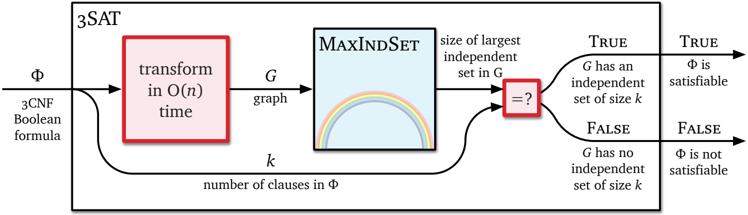
Problema del Massimo Insieme Indipendente (MaxIndSet)

Input: un grafo arbitrario G

Output: la dimensione k dell'insieme indipendente più grande in G

MaxIndSet è NP-hard!

Dimostriamo che **MaxIndSet** è NP-hard con una riduzione da **3SAT**



Data una formula arbitraria in 3CNF, come per esempio

$$(a \lor b \lor c) \land (b \lor \overline{c} \lor \overline{d}) \land (\overline{a} \lor c \lor d) \land (a \lor \overline{b} \lor \overline{c})$$

costruiamo un grafo $G = (V, E)$ tale che:

Da 3SAT a MaxIndSet

Data una formula arbitraria in 3CNF, come per esempio

$$(a \vee b \vee c) \wedge (b \vee \overline{c} \vee \overline{d}) \wedge (\overline{a} \vee c \vee d) \wedge (a \vee \overline{b} \vee \overline{c})$$

costruiamo un grafo $G = (V, E)$ tale che:

- V contiene 3 vertici per clausola, 1 per letterale

Da 3SAT a MaxIndSet

Università degli Studi di Padova

Data una formula arbitraria in 3CNF, come per esempio

$$(a \vee b \vee c) \wedge (b \vee \overline{c} \vee \overline{d}) \wedge (\overline{a} \vee c \vee d) \wedge (a \vee \overline{b} \vee \overline{c})$$

costruiamo un grafo $G = (V, E)$ tale che:

- V contiene 3 vertici per clausola, 1 per letterale
- Gli archi in E sono di due tipi

Da 3SAT a MaxIndSet

Data una formula arbitraria in 3CNF, come per esempio

$$(a \vee b \vee c) \wedge (b \vee \overline{c} \vee \overline{d}) \wedge (\overline{a} \vee c \vee d) \wedge (a \vee \overline{b} \vee \overline{c})$$

costruiamo un grafo $G = (V, E)$ tale che:

- V contiene 3 vertici per clausola, 1 per letterale
- Gli archi in E sono di due tipi
- Archi di clausola che collegano letterali nella stessa clausola

Da 3SAT a MaxIndSet

Data una formula arbitraria in 3CNF, come per esempio

$$(a \vee b \vee c) \wedge (b \vee \overline{c} \vee \overline{d}) \wedge (\overline{a} \vee c \vee d) \wedge (a \vee \overline{b} \vee \overline{c})$$

costruiamo un grafo $G = (V, E)$ tale che:

- V contiene 3 vertici per clausola, 1 per letterale
- Gli archi in E sono di due tipi
- Archi di clausola che collegano letterali nella stessa clausola
- Archi di consistenza che collegano un letterale con la sua negazione

Correttezza della riduzione

- Se k è il numero di clausole nella formula, allora un insieme indipendente in G può contenere al più k elementi.
- Dimostriamo che G contiene un insieme indipendente di dimensione esattamente k se e solo se la formula Φ è soddisfacibile:

Correttezza della riduzione

- Se k è il numero di clausole nella formula, allora un insieme indipendente in G può contenere al più k elementi.
- Dimostriamo che G contiene un insieme indipendente di dimensione esattamente k se e solo se la formula Φ è soddisfacibile:
  - $\Rightarrow$ Se la formula è soddisfacibile allora esiste un assegnamento delle variabili che la rende vera. Scegliamo un sottoinsieme S di k vertici di G, uno per ogni clausola, in modo che il letterale corrispondete sia vero. Si può far vedere che S è un insieme indipendente per G.

Correttezza della riduzione

- Se k è il numero di clausole nella formula, allora un insieme indipendente in G può contenere al più k elementi.
- Dimostriamo che G contiene un insieme indipendente di dimensione esattamente k se e solo se la formula Φ è soddisfacibile:
  - $\Rightarrow$ Se la formula è soddisfacibile allora esiste un assegnamento delle variabili che la rende vera. Scegliamo un sottoinsieme S di k vertici di G, uno per ogni clausola, in modo che il letterale corrispondete sia vero. Si può far vedere che S è un insieme indipendente per G.
  - $\Leftarrow$ Supponiamo che G contenga un insieme indipendente S di dimensione k. Ogni vertice di S deve stare in un triangolo diverso. Se assegnamo il valore Vero ai letterali presenti in S otteniamo un assegnamento che rende vera la formula.

Correttezza della riduzione

- Se k è il numero di clausole nella formula, allora un insieme indipendente in G può contenere al più k elementi.
- Dimostriamo che G contiene un insieme indipendente di dimensione esattamente k se e solo se la formula Φ è soddisfacibile:
  - $\Rightarrow$ Se la formula è soddisfacibile allora esiste un assegnamento delle variabili che la rende vera. Scegliamo un sottoinsieme S di k vertici di G, uno per ogni clausola, in modo che il letterale corrispondete sia vero. Si può far vedere che S è un insieme indipendente per G.
  - $\Leftarrow$ Supponiamo che G contenga un insieme indipendente S di dimensione k. Ogni vertice di S deve stare in un triangolo diverso. Se assegnamo il valore Vero ai letterali presenti in S otteniamo un assegnamento che rende vera la formula.
- La costruzione del grafo richiede un tempo polinomiale.

Un problema apparentemente simile

- Sia $G = (V, E)$ un grafo arbitrario.
- Un accoppiamento o insieme di archi indipendenti in G è un sottoinsieme M degli archi tali che non c'è nessun vertice in comune tra due archi.

Un problema apparentemente simile

- Sia $G = (V, E)$ un grafo arbitrario.
- Un accoppiamento o insieme di archi indipendenti in G è un sottoinsieme M degli archi tali che non c'è nessun vertice in comune tra due archi.

Problema del Massimo Accoppiamento (MaxMatch)

Input: un grafo arbitrario G

Output: la dimensione k dell'accoppiamento più grande in G

Un problema apparentemente simile

- Sia $G = (V, E)$ un grafo arbitrario.
- Un accoppiamento o insieme di archi indipendenti in G è un sottoinsieme M degli archi tali che non c'è nessun vertice in comune tra due archi.

Problema del Massimo Accoppiamento (MaxMatch)

Input: un grafo arbitrario G

Output: la dimensione k dell'accoppiamento più grande in G

Algoritmo di Edmonds

Il problema dell'accoppiamento massimo è risolvibile in tempo polinomiale! Più precisamente, in tempo $O(|V|^2|E|)$

Altri esempi di dualità

Problemi NP-Hard

Problemi in P

Altri esempi di dualità

Problemi NP-Hard

- Copertura di vertici
  Trovare il minimo sottoinsieme S di vertici tali che ogni arco ha almeno un'estremità in S

Problemi in P

Altri esempi di dualità

Problemi NP-Hard

- Copertura di vertici
  Trovare il minimo sottoinsieme S di vertici tali che ogni arco ha almeno un'estremità in S

Problemi in P

- Copertura di archi
  Trovare il minimo sottoinsieme M di archi tali che ogni vertice è adiacente ad un arco in M

Altri esempi di dualità

Problemi NP-Hard

- Copertura di vertici
  Trovare il minimo sottoinsieme S di vertici tali che ogni arco ha almeno un'estremità in S
- circuito Hamiltoniano
  Trovare un ciclo che visita ogni vertice esattamente una volta

Problemi in P

- Copertura di archi
  Trovare il minimo sottoinsieme M di archi tali che ogni vertice è adiacente ad un arco in M

Altri esempi di dualità

Problemi NP-Hard

- Copertura di vertici
  Trovare il minimo sottoinsieme S di vertici tali che ogni arco ha almeno un'estremità in S
- circuito Hamiltoniano
  Trovare un ciclo che visita ogni vertice esattamente una volta

Problemi in P

- Copertura di archi
  Trovare il minimo sottoinsieme M di archi tali che ogni vertice è adiacente ad un arco in M
- Circuito Euleriano
  Trovare un ciclo che visita ogni arco esattamente una volta

Altri esempi di dualità

Problemi NP-Hard

- Copertura di vertici
  Trovare il minimo sottoinsieme S di vertici tali che ogni arco ha almeno un'estremità in S
- circuito Hamiltoniano
  Trovare un ciclo che visita ogni vertice esattamente una volta
- 3-colorazione di un grafo
  Trovare un modo per colorare i vertici di un grafo con tre colori tale che vertici adiacenti sono di colore diverso

Problemi in P

- Copertura di archi
  Trovare il minimo sottoinsieme M di archi tali che ogni vertice è adiacente ad un arco in M
- Circuito Euleriano
  Trovare un ciclo che visita ogni arco esattamente una volta

Altri esempi di dualità

Problemi NP-Hard

- Copertura di vertici
  Trovare il minimo sottoinsieme S di vertici tali che ogni arco ha almeno un'estremità in S
- circuito Hamiltoniano
  Trovare un ciclo che visita ogni vertice esattamente una volta
- 3-colorazione di un grafo
  Trovare un modo per colorare i vertici di un grafo con tre colori tale che vertici adiacenti sono di colore diverso

Problemi in P

- Copertura di archi
  Trovare il minimo sottoinsieme M di archi tali che ogni vertice è adiacente ad un arco in M
- Circuito Euleriano
  Trovare un ciclo che visita ogni arco esattamente una volta
- 2-colorazione di un grafo
  Trovare un modo per colorare i vertici di un grafo con due colori tale che vertici adiacenti sono di colore diverso

Come scegliere il problema giusto

- Se il problema richiede di assegnare bit agli oggetti: **SAT**
- Se il problema richiede di assegnare etichette agli oggetti prese un piccolo insieme, o di partizionare gli oggetti in un numero costante di sottoinsiemi: **3Color** o k**Color**
- Se il problema richiede di organizzare un insieme di oggetti in un ordine particolare: **Circuito Hamiltoninano**
- Se il problema richiede di trovare un piccolo sottoinsieme che soddisfi alcuni vincoli: **MinVertexCover**
- Se il problema richiede di trovare un sottoinsieme grande che soddisfi alcuni vincoli: **MaxIndependentSet**
- Se il numero 3 appare in modo naturale nel problema, provare **3SAT** o **3Color** (No, questo non è uno scherzo.)
- Se tutto il resto fallisce, prova **3SAT** o anche **CircuitSAT**!