Esame Ricerca Operativa – 24/01/2019 – Soluzioni

Problema 1 : modello ILP

Sia $G = (V, E)$ un grafo completo i cui archi sono colorati con tre colori. Denotiamo con E_i, per $i = 1, 2, 3$ l'insieme di archi del colore i, sicchè $E = E_1 \cup E_2 \cup E_3$.

- Diciamo che un sottografo completo $G' = (S, E')$ è *bilanciato* se E' contiene come minimo il 30% e come massimo il 36% di archi di ciascun colore. Ossia, detto $E'_i := E' \cap E_i$ per ogni $i = 1, 2, 3$, si ha
$$30\% \ |E'| \leq |E'_i| \leq 36\% \ |E'| \qquad \forall i$$
- Dato $k \in \mathbb{N}$, diciamo che un sottografo completo $G' = (S, E')$ è *k-stretto* se il colore più rappresentato in E' contiene al massimo k archi in più del colore meno rappresentato, ossia
$$\max_i |E'_i| - \min_i |E'_i| \leq k$$

Si risolva *uno* a scelta tra i seguenti problemi:

Primo problema (voto complessivo max 28) : Si scriva un modello di programmazione lineare intera per determinare un sottografo completo bilanciato con $|S|$ massima possibile.

Secondo problema (voto complessivo max 30L) : Si scriva un modello di programmazione lineare intera per determinare il minimo k tale che esiste un sottografo completo k-stretto con almeno 10 nodi.

Soluzione.

Introduciamo delle variabili binarie x_i per ogni $i \in V$ per selezionare i nodi presi in S. Inoltre introduciamo delle variabili y_{ij}, per $ij \in E$. Il senso di queste variabili è che
$$E'_t := \{ij \in E_t : y_{ij} = 1\}$$

Primo modello Il modello relativo al primo problema è il seguente.

La funzione obiettivo è
$$\max \sum_{i \in V} x_i \tag{1}$$

subject to

$$y_{ij} \leq x_i \qquad \forall ij \in E \tag{2}$$

$$y_{ij} \leq x_j \qquad \forall ij \in E \tag{3}$$

$$y_{ij} \geq x_i + x_j - 1 \qquad \forall ij \in E \tag{4}$$

$$\sum_{ij \in E_s} y_{ij} \leq 0.36 \sum_{ij \in E} y_{ij} \qquad \forall s = 1, 2, 3 \tag{5}$$

$$\sum_{ij \in E_s} y_{ij} \geq 0.3 \sum_{ij \in E_t} y_{ij} \qquad \forall s = 1, 2, 3 \tag{6}$$

I vincoli (2),(3),(4), garantiscono che per ogni $ij \in E$ si ha $y_{ij} = 1$ sse $x_i = x_j = 1$. I vincoli (5),(6) garantiscono che il sottografo G' è bilanciato.

Secondo modello. Nel secondo modello possiamo usare le stesse variabili che nel primo, più una variabile intera k. La funzione obiettivo è

$$\min k \tag{7}$$

Ci servono dei vincoli per essere sicuri che la differenza tra il numero di archi di colore diverso non è mai superiore a k. Possiamo usare (2), (3),(4) e in più richiedere

$$\sum_{ij \in E_s} y_{ij} - \sum_{ij \in E_t} y_{ij} \leq k \qquad \forall s, t \in \{1, 2, 3\} : s \neq t \tag{8}$$

(Si noti che uesto vincolo ha lo stesso effetto che chiedere

$$|\sum_{ij \in E_s} y_{ij} - \sum_{ij \in E_t} y_{ij}| \leq k \qquad \forall 1 \leq s < t \leq 3$$

e quindi ci garantisce che il grafo è k-stretto.)

Infine il vicolo sul numero minimo di nodi è

$$\sum_{i \in V} x_i \geq 10 \tag{9}$$

Il modello finale è minimizzare (7) con vincoli (2), (3),(4), (8), (9).

Problema 2 : Domande Vero/Falso

1. ([1pt]$\boxed{V \mid F}$) Sia $G = (\{1, \dots, n\}, A)$ una qualsiasi rete di flusso in cui la capacità di ogni arco è un numero intero. Allora il valore del massimo flusso tra 1 e n deve essere intero.

   VERO: La capacità di ogni taglio è intera in quanto somma di interi. Quindi anche la min capacità(che è uguale al massimo flusso) è intera.

2. ([2pt]$\boxed{V \mid F}$) Sia $G = (\{1, \dots, n\}, A)$ una rete di flusso in cui la capacità di ogni arco è un numero irrazionale e tutte le capacità sono diverse tra loro. Allora il valore del massimo flusso tra 1 e n non può essere intero.

   FALSO: Possiamo avere capacità irrazionali la cui somma è un intero. Ad esempio supponiamo 3 nodi e $c(1,2) = \sqrt{2}$, $c(1,3) = 2 - \sqrt{2}$, $c(2,3) = \pi$. Il taglio di capacità minima separa $1, 2$ da 3 e ha capacità 2.

3. ([1pt]$\boxed{V \mid F}$) Ogni matrice 0/1 che ha esattamente due '1' in ogni colonna è totalmente unimodulare.

   FALSO: Ad esempio la matrice

   $$\begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

   ha determinante = 2.

4. Sia $G = (V, E)$ un grafo connesso pesato sugli archi, con $m := |E| \geq 3$. Denotiamo gli archi con $e_1, \dots, e_m$ e supponiamo che

   $$c(e_1) \leq c(e_2) \leq \cdots \leq c(e_m) \tag{10}$$

   Allora:

   (a) ([1pt]$\boxed{V \mid F}$) Se $c(e_j) \neq c(e_k)$ per ogni coppia di archi e_j, e_k con $1 < j < k \leq m$ allora e_1 appartiene a ogni minimo albero di supporto di G.

   VERO: Ci possono essere due archi entrambi di costo minimo, i.e. $c(e_1) = c(e_2)$, ma non tre. L'algoritmo di Kruskal li prende entrambi in quanto non creano cicli.

   (b) ([1pt]$\boxed{V \mid F}$) Se $c(e_1) = c(e_2) = c(e_3)$ allora allora e_1 non appartiene ad almeno un minimo albero di supporto di G.

   FALSO: Potrebbe essere che e_1 è un taglio di G e quindi c'e' in tutti gli alberi (ad es. è l'unico arco verso un certo nodo)

(c) ([1pt] $\boxed{V \mid F}$) Se $c(e_m) > \sum_{i=1}^{m-1} c(e_i)$ allora e_m non appartiene ad almeno un minimo albero di supporto di G.

FALSO: Per lo stesso motivo del punto precedente

(d) ([1pt] $\boxed{V \mid F}$) Sia $e_m = ij$. Se $\min\{d(i), d(j)\} \geq 2$ allora e_m non appartiene ad alcun minimo albero di supporto di G.

FALSO: Per lo stesso motivo del punto precedente. Ad es. se gli archi in G sono 12,23, 34 con $c(2,3)$ massimo

5. ([2pt] $\boxed{V \mid F}$) Il poliedro delle soluzioni ammissibili di un problema di programmazione lineare (il primale) e quello delle soluzioni ammissibili del suo duale sono sempre diversi.

FALSO: Sia A una matrice antisimmetrica, ossia $A = -A^t$. Ad esempio

$$\begin{pmatrix} 0 & 1 & -2 \\ -1 & 0 & 3 \\ 2 & -3 & 0 \end{pmatrix}$$

Prendiamo come primale

$$\min\{0^t x : Ax \geq 0, x \geq 0\}$$

il duale e'

$$\max\{0^t y : A^t y \leq 0, y \geq 0\}$$

ed essendo $A^t y \leq 0$ sse $(-A)y \leq 0$ sse $Ay \geq 0$ si ha l'asserto.

NOTA: qualcuno ha risolto il problema nel modo appena visto, altri hanno trovato una "scorciatoia" meno generale, ossia hanno considerato l'unico caso in cui sia il primale che il duale sono non ammissibili. In questo caso i poliedri uguali (se di poliedri si può parlare) sono l'insieme vuoto. Ho considerato corretta anche questa soluzione, ma la soluzione in cui si trovato il controesempio non vuoto è stata premiata con un voto maggiore.

6. ([1pt] $\boxed{V \mid F}$) Sia P il poliedro definito dai vincoli di un modello di programmazione lineare intera per un'istanza di dimensione n di un problema di ottimizzazione. Siano $f(n)$ il numero di faccette e $v(n)$ il numero di vertici di P. Se $f(n)$ è un polinomio in n allora $v(n)$ non può essere una funzione esponenziale in n.

FALSO: Si pensi all'ipercubo Q_n, che ha $2n$ faccette, definite dai vincoli $0 \leq x_i \leq 1$ per $i = 1, \ldots, n$. Esso però ha 2^n vertici in quanto ogni punto di $\{0, 1\}^n$ è un vertice di Q_n.

7. ([1pt] $\boxed{V \mid F}$) Trovare un cover di cardinalità massima su un grafo non bipartito è un problema NP-hard.

FALSO: la soluzione ottima è prendere tutti i nodi. Trovare il cover *minimo* è NP-hard...

8. ([2pt] $\boxed{V}\boxed{F}$) Sia $c = (c_{ij})$ la matrice dei costi di un problema di assegnamento su un grafo bipartito di $2n$ nodi (dove $c(i,j)$ è il costo di assegnare il lavoratore i al lavoro j. Sia M la soluzione ottima per i costi c. Dati $2n$ numeri $a_1, \ldots, a_n, b_1, \ldots, b_n$, si considerino dei nuovi costi c' definiti da

$$c'(i,j) = c(i,j) + a_i - b_j$$

Allora M rimane la soluzione ottima anche rispetto ai costi c'.

VERO: Ogni soluzione L per i costi c' ha come valore $c'(L) = c(L) + \sum_i a_i - \sum_j b_j$ e quindi ha lo stesso valore che ha per i costi c, a meno di una costante

9. ([1pt] $\boxed{V}\boxed{F}$) I seguenti problemi sono tutti polinomiali: (i) trovare un matching perfetto di costo minimo in grafi non-bipartiti; (ii) trovare un taglio di capacità minima tra due nodi prefissati di una rete di flusso; (iii) trovare un cammino elementare di lunghezza massima tra due nodi di un grafo aciclico.

VERO: Attenzione al fatto che il problema (iii) è NP-hard su grafi generali, ma qui stiamo supponendo il grafo essere aciclico.

10. ([2pt] $\boxed{V}\boxed{F}$) Sia G un grafo connesso di 30 nodi e 60 archi in cui ci sono 20 archi di valore 1, 20 di valore 2 e 20 di valore 3. Allora per ogni arco e di costo 2, esiste almeno un minimo albero di supporto che contiene e.

FALSO: Ad esempio, si costruisca G creando un albero di costo 20 su 21 nodi, aggiungendo i 20 archi di costo 2 tra questi stessi nodi e usando gli archi di costo 3 per toccare i rimanenti 9 nodi.