OTTIMIZZAZIONE COMBINATORIA
Prova scritta del 20 aprile 2017

1. Rispondete alle domande seguenti distinguendo le risposte vere da quelle false. Penalità per risposte errate.

A	*a*)	Un grafo intervallo non può avere clique di tre elementi.	*vero*	*falso*
	b)	Aggiungendo $\mathbf{I}_{nxn}$ alla matrice d'incidenza nodi-archi di un grafo di n nodi si ha sempre una matrice totalmente unimodulare.	*vero*	*falso*
	c)	Due alberi binari con le foglie in comune e le radici collegate da un arco formano un grafo cubico.	*vero*	*falso*
	d)	Siano $G = (V, E)$, $V = S_1 \cup S_2$ ma $S_1 \cap S_2 \neq \emptyset$. Allora G non è bipartito	*vero*	*falso*
B	*a*)	La matrice di adiacenza **A** di un ciclo C è totalmente unimodulare se e solo se il ciclo ha un numero pari di nodi	*vero*	*falso*
	b)	Un grafo intervallo non è mai bipartito	*vero*	*falso*
	c)	Esistono almeno due grafi cubici distinti che contengono una clique di quattro elementi	*vero*	*falso*
	d)	Ogni grafo 3-colorabile è necessariamente bipartito	*vero*	*falso*

Gruppo A. Tutte le frasi sono false: (*a*) perché tre intervalli con un punto in comune corrispondono a una clique del grafo intervallo associato; (*b*) perché il grafo potrebbe essere simmetrico e non bipartito; (*c*) perché le foglie dei due alberi, una volta messe in comune, hanno grado 2 e non 3; infine (*d*) perché anche se la bipartizione di V non è data da S_1 e S_2, $\{S_1 - S_2, S_2\}$ è comunque una bipartizione in insiemi stabili.

Gruppo B. La frase (*c*) è falsa se il grafo in questione è connesso (l'unico grafo cubico connesso contenente K_4 è K_4), ma se non è connesso è vera: quindi in generale è vera. Le altre tre frasi sono false: (*a*) la matrice **A** associata a C_6 non è totalmente unimodulare. Le altre tre frasi sono tutte false: (*b*) perché basta considerare P_3; (*d*) perché K_3 è 3-colorabile ma non è bipartito.

2. Occorre mettere in fila quattro azioni non sovrapponibili della durata di un'ora ciascuna. L'azione k dovrebbe essere completata entro il tempo d_k riportato in ore nella seguente tabella, ma è ammesso un ritardo se ciò non fosse possibile.

	Gruppo A				*Gruppo B*			
azione	1	2	3	4	1	2	3	4
d_k	2	2	3	2	2	1	2	3

Si vogliono assegnare le azioni a quattro ore consecutive in modo da minimizzare il numero di azioni completate in ritardo. Formulate il problema come programmazione lineare 01 utilizzando variabili x_{kt} che assegnano alla generica azione k l'ora t alla quale l'azione è terminata (avendosi quattro azioni si avrà quindi $t = 1, \ldots, 4$), più altre variabili per indicare se un'azione è o no in ritardo. Scrivete tutti i vincoli del problema e la sua funzione obiettivo.

Le variabili x_{kt} assegnano a ogni azione k esattamente un tempo di completamento:

$$x_{k1} + x_{k2} + x_{k3} + x_{k4} = 1 \qquad \text{per } k = 1, \ldots, 4$$

Inoltre, non potendo sovrapporre due azioni, ciascun tempo t corrisponderà a una sola azione:

$$x_{1t} + x_{2t} + x_{3t} + x_{4t} = 1 \qquad \text{per } t = 1, \ldots, 4.$$

Per indicare se l'azione k è in ritardo, utilizziamo una nuova variabile y_k a valori in $\{0, 1\}$. Adottiamo la convenzione che $y_k = 1$ se e solo se k è in ritardo. Le y_k sono perciò legate alle x_{kt} dai vincoli seguenti:

$$y_k \geq x_{kt} \qquad \text{per ogni } t > d_k$$

Riferendoci ad esempio alla tabella del **Gruppo A** avremo:

$$y_1 \geq x_{13} \qquad y_1 \geq x_{14}$$

$y_2 \geq x_{23}$ $\qquad$ $y_2 \geq x_{24}$

$y_3 \geq x_{34}$

$y_4 \geq x_{43}$ $\qquad$ $y_4 \geq x_{44}$

L'obiettivo si scrive

$$\min \quad y_1 + y_2 + y_3 + y_4$$

Si noti che le diseguaglianze scritte non escludono $y_k = 1$ quando $x_{kt} = 0$ per $t > d_k$. In altre parole l'azione potrebbe essere dichiarata in ritardo pur essendo assegnata a un tempo che non comporta ritardi. Tuttavia $y_k = 1$ non sarebbe in questo caso una scelta ottima, potendosi porre $y_k = 0$ senza violare alcun vincolo.
In alternativa, tenendo conto dei primi vincoli di assegnamento, le variabili y possono esprimersi come

$$y_k = \sum_{t>d_k} x_{kt}$$

Infatti: o l'azione k è assegnata a un tempo $t \leq d_k$, e allora la sommatoria a secondo membro vale 0, oppure no, e allora per via dei vincoli di assegnamento di cui s'è detto la sommatoria vale 1.

3. **Gruppo A**. Come va cambiata la formulazione del Problema 2 se si vuole minimizzare la somma dei ritardi delle azioni considerate?

Gruppo B. Come va cambiata la formulazione del Problema 2 se si vuole minimizzare il più grande ritardo fra quelli delle azioni considerate?

Indichiamo con r_k il ritardo dell'azione k. Questo è per definizione il massimo di due termini: il primo è pari a 0, il secondo è pari alla differenza fra il tempo t nel quale l'azione è terminata e la data di consegna d_k stabilita. In altre parole: $r_k = \text{t} - d_k$ se $t > d_k$, 0 altrimenti. Il massimo fra due numeri è per definizione il più piccolo numero che è $\geq$ di entrambi i numeri dati. Quindi

$r_k \geq 0$

$r_k \geq tx_{kt} - d_k$ $\qquad$ per ogni k e t

Notiamo che per via dei vincoli di assegnamento esiste un solo valore di t per il quale $x_{kt} = 1$: in corrispondenza a esso la seconda disequazione diventa $r_k \geq t - d_k$, mentre per tutti gli altri valori si ha $r_k \geq - d_k$. Per il **Gruppo A** l'obiettivo si scrive semplicemente

$$\min \quad r_1 + r_2 + r_3 + r_4$$

Per il **Gruppo B**, indicando con r il massimo ritardo fra tutti gli r_k e applicando lo stesso ragionamento, possiamo scrivere l'obiettivo come

$$\min \quad r$$

soggetto a

$r \geq r_k$ $\qquad$ per ogni azione k

4. Guardando la formulazione scritta in risposta al Problema 2 ci si accorge che lo stesso può formularsi come massimo insieme stabile pesato. Su quale grafo, e quali pesi vanno adottati per i suoi vertici?

Ponendo $z_k = 1 - y_k$ diamo implicitamente alle z_k il ruolo di contare le azioni svolte in tempo. L'obiettivo, equivalente a quello del Problema 2, consiste nel massimizzarne il numero:

$$\max \quad z_1 + z_2 + z_3 + z_4$$

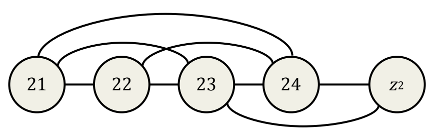
Usando la definizione delle nuove variabili, il vincolo che le lega alle x_{kt} si scrive $1 - z_k \geq x_{kt}$, ovvero

$x_{kt} + z_k \leq 1$ $\qquad$ per ogni $t > d_k$

Notiamo poi che i vincoli di assegnamento delle x_{kt} differiscono dai vincoli clique dell'insieme stabile per il fatto che sono di eguaglianza anziché di $\leq$. Rilasciandoli abbiamo

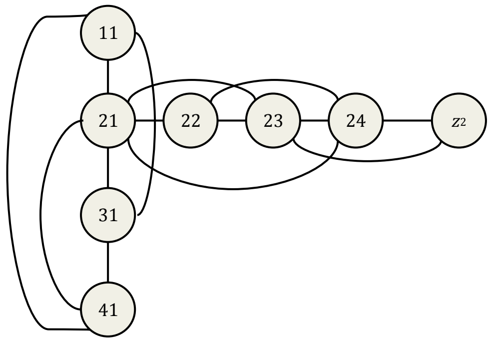
$x_{k1} + x_{k2} + x_{k3} + x_{k4} \leq 1$ $\qquad$ per $k = 1, \ldots, 4$.

$x_{1t} + x_{2t} + x_{3t} + x_{4t} \leq 1$ $\qquad$ per $t = 1, \ldots, 4$.

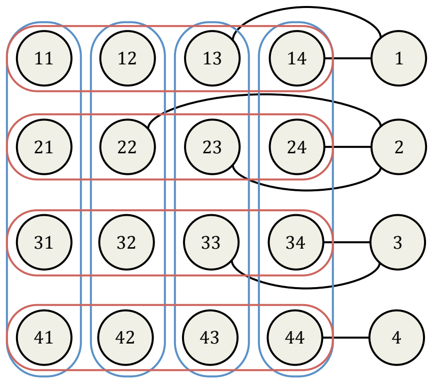


Il grafo avrà quindi un nodo per ciascuna delle variabili x e z introdotte. Per un dato k, l'aspetto sarà quello rappresentato a lato (la figura si riferisce a $k = 2$).

Il nodo più a destra corrisponde alla variabile z_2, gli altri alle varia bili x_{21}, …, x_{24}: questi ultimi formano una clique di quattro nodi. Gli archi fra il nodo z_2 e i nodi 23 e 24 implicano che se si sceglie di porre uno di questi ultimi nell'insieme stabile soluzione (il che vuol dire che l'azione 2 termina alla terza o quarta ora), allora non si può includere nell'insieme anche il nodo z_2 (cioè l'azione 2 non può essere svolta in tempo).
Per t fissato, i nodi associati alle x_{kt} sono anch'essi legati in una clique da 4: di seguito mostriamo la clique corrispondente a t = 1 insieme ai nodi relativi all'azione k = 2. Il grafo va completato con le clique verticali associate ai tempi t = 2, 3, 4, collegandone i nodi orizzontalmente a formare altre clique, e con i nodi z_1, z_3, z_4 opportunamente collegati ai nodi kt.
Notiamo però che i nodi z_k formano un insieme stabile di 4 elementi: quindi con l'obiettivo sopra scritto una soluzione ottima metterebbe tutte le x a 0 e tutte le z a 1, il che non è evidentemente corretto. Per evitarlo occorre scegliere i pesi dei nodi kt in modo da obbligare la soluzione ottima a sceglierne esattamente uno per ogni k ed esattamente uno per ogni t.
E' facile constatare che se si dà peso 5 a questi nodi, una soluzione ottima soddisferà necessariamente i vincoli di assegnamento con il segno "=", scegliendo esattamente quattro nodi di tipo kt.
Infatti, ogni soluzione con meno di quattro nodi di tipo kt pesa al massimo $3\cdot 5 + 4 = 19$, mentre ogni soluzione con quattro nodi di tipo kt ha peso almeno pari a 20.



Organizzando i nodi kt in una griglia quadrata e rappresentando le clique derivate dai vincoli di assegnamento solo come insiemi di nodi (senza quindi specificarne gli archi), il grafo del **Gruppo B** ha complessivamente l'aspetto mostrato nella figura seguente.



OTTIMIZZAZIONE COMBINATORIA **GRUPPO B**
Prova scritta del 15 settembre 2015

Cognome: |__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|
Nome: |__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|__|
Matricola: |__|__|__|__|__|__|

1. Rispondete alle domande seguenti distinguendo le risposte vere da quelle false:

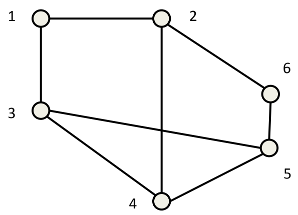
a) Esistono grafi intervallo non bipartiti e grafi bipartiti non intervallo.

b) Matching, clique, insieme stabile, edge-cover e trasversale si risolvono tutti in tempo polinomiale se il grafo è un albero.

c) Un grafo r-regolare è euleriano se e solo se r è un numero pari.

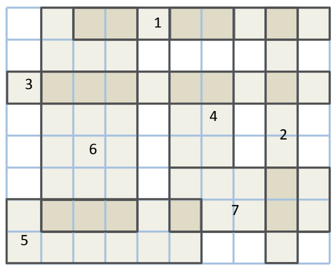
d) Sia $c_u \in \mathbb{R}$ associato $u \in U$, e $\Im = \{X \subseteq U\colon U - X$ albero ricoprente di $G\}$. Si può usare il greedy per trovare l'insieme $X^* \in \Im$ di peso minimo.

vero	*falso*
vero	*falso*
vero	*falso*
vero	*falso*



2. Dato il grafo $G = (V, E)$ di figura, sia S un sottoinsieme di V e sia $N(S)$ l'insieme dei vertici adiacenti ad almeno un vertice di S. Si vuole trovare un insieme S che massimizzi $|N(S) - S|$. Formulare il problema come programmazione lineare 0-1 sul grafo dato, eliminare il vincolo di interezza e scriverne il duale. Si ha garanzia che soluzioni dei due PL siano intere? Se sì, perché?

3. Il disegno accanto mostra 7 rettangoli numerati progressivamente. Le aree più scure indicano le intersezioni fra i rettangoli. Formulate su un opportuno grafo il problema di massimizzare l'area delle intersezioni scegliendole in modo che nessun rettangolo dei sette originali sia intersecato due volte (calcolate le aree basandovi sulla griglia indicata nel disegno).
Descrivete un algoritmo per risolvere il problema nel caso in esame e mostratene alcuni passi.



4. In un problema di KNAPSACK 0-1 con penalità, il generico oggetto k è associato a un ingombro a_k, un profitto p_k e un costo c_k. Il profitto dello zaino, che rispetta la solita disequazione $a_1x_1 + \ldots a_nx_n \leq b$, a detratto del più grande fra i costi degli elementi in esso contenuti. Ad esempio, se una soluzione $\mathbf{x}$ contiene i soli elementi 1 e 2, il suo valore sarà $p(\mathbf{x}) = p_1 + p_2 - \max\{c_1, c_2\}$. Consideriamo il problema

$$\max \quad 3x_1 + 2x_2 + 7x_3 + 4x_4 + 6x_5 - c$$
$$6x_1 + 5x_2 + 5x_3 + x_4 + 4x_5 \leq 11 \quad \text{dove} \quad c \geq c_k\, x_k \ \text{ per ogni oggetto } k$$
$$x_k \in \{0, 1\}$$

Come si vede, se $x_k = 1$ allora il profitto è detratto di una quantità pari ad almeno c_k, e massimizzando si ottiene l'effetto di sottrarre esattamente la penalità più grande fra quelle associate agli oggetti selezionati. Risolvete questo problema con un algoritmo di branch-and-bound per le penalità c_k sotto riportate.

c_1	c_2	c_3	c_4	c_5
4	2	7	4	8