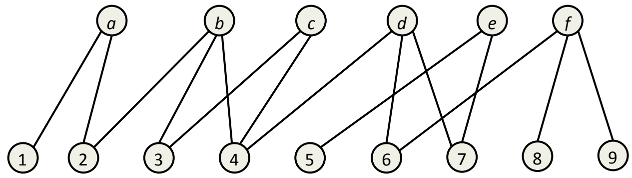
1. Per misurare la qualità della ricerca di un dipartimento universitario, un'agenzia indipendente richiede ai ricercatori che ne fanno parte di sottoporre a valutazione almeno due dei lavori scientifici pubblicati nel quadriennio. Il dipartimento sarà penalizzato in misura pari al numero di ricercatori che ne presenteranno meno di due. La regola prevede inoltre che ogni lavoro scientifico eseguito in collaborazione da più ricercatori del dipartimento possa essere presentato da uno solo degli autori.
Il grafo bipartito $G = (V, E)$ di figura, con V partizionato nei due sottoinsiemi R (i ricercatori, sopra) e L (i lavori da loro prodotti, sotto), contiene l'arco uv se e solo se il ricercatore $u \in R$ ha partecipato alla redazione del lavoro $v \in L$. Ad esempio, il ricercatore b ha collaborato ai lavori 2, 3 e 4, mentre il lavoro 6 ha come autori i ricercatori d e f. Come si vede, non vi sono ricercatori con zero pubblicazioni nel quadriennio.



Il direttore del dipartimento vuole proporre ai ricercatori di assegnarsi i lavori in modo che la lista presentata all'agenzia minimizzi il numero di ricercatori che ne espongono uno soltanto. Formulate questo problema di ottimizzazione combinatoria in termini di PL 01.

Sia $a_{uv} = 1$ se $uv \in E$ e $a_{uv} = 0$ altrimenti. Definiamo due tipi di variabile decisionale 01: $x_{uv} = 1$ se e solo se il lavoro v è attribuito al ricercatore u; $x_u = 1$ se il ricercatore u presenta un solo lavoro. Ogni lavoro dev'essere assegnato a uno dei suoi autori, quindi:

$$\sum_{u \in R} a_{uv} x_{uv} = 1 \qquad \text{per ogni lavoro } v \in L$$

Inoltre se $x_u = 0$ allora il ricercatore u deve presentare almeno 2 lavori, mentre se $x_u = 1$ deve presentarne (almeno) 1:

$$\sum_{v \in L} a_{uv} x_{uv} \geq 2 - x_u \qquad \text{per ogni ricercatore } u \in R$$

L'obiettivo consiste nel minimizzare il numero di ricercatori che presentano un solo lavoro:

$$\min \quad \sum_{u \in R} x_u$$

Nel caso in esame il problema si scrive quindi

$$\begin{array}{llllll}
\min & x_a + x_b + \ldots + x_f & & & & \\
 & x_{a1} = 1 & x_{a2} + x_{b2} = 1 & x_{b3} + x_{c3} = 1 & x_{b4} + x_{c4} + x_{d4} = 1 & \ldots \\
 & x_{a1} + x_{a2} + x_a = 2 & x_{b2} + x_{b3} + x_{b4} + x_b = 2 & x_{c3} + x_{c4} + x_c = 2 & & \ldots \\
 & x_{a1}, x_{a2}, \ldots, x_{f9}, x_a, \ldots, x_f \in \{0, 1\} & & & &
\end{array}$$

2. Con riferimento al grafo bipartito di figura, considerate la funzione $f(S) = |\{uv \in E: u \in S\}|$ definita per ogni $S \subseteq R$: ad esempio, $f(\{a\}) = 2$, $f(\{a, b\}) = f(\{a, b, c\}) = 4$. Definito il matroide uniforme $(R, \mathfrak{I})$ associato l'intero positivo k, formulate come PL 01 il problema di massimizzare $f(S)$ fra tutti gli $S \in \mathfrak{I}$.

Introduciamo le variabili binarie x_u che codificano il vettore caratteristico dell'insieme S; in altre parole, $x_u = 1$ se e solo se $u \in S$. Un matroide uniforme sull'insieme universo R ha come famiglia $\mathfrak{I}$ la classe di tutti i sottoinsiemi S di R con cardinalità non superiore a k. Quindi un S ammissibile è associato a un vettore $\mathbf{x}$ per il quale

$$\sum_{u \in R} x_u \le k$$

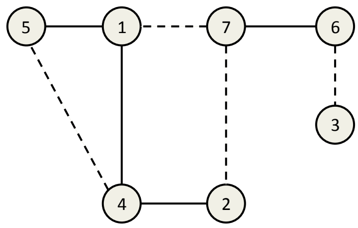
Per scrivere $f(S)$ occorre definire nuove variabili binarie y_v per ogni $v \in L$, convenendo $y_v = 1$ solo se v è adiacente ad almeno un nodo $u \in S \subseteq R$. Questo comporta l'introduzione dei vincoli

$$y_v \le \sum_{u \in N(v)} x_u \qquad \text{per ogni } v \in L$$

In questo modo, se nell'intorno di v c'è almeno un nodo che appartiene all'insieme S prescelto, allora si potrà porre a 1 la variabile y_v, in caso contrario no. Ciò posto, l'obiettivo si scrive

$$\max \quad f(S) = \sum_{v \in L} y_v$$

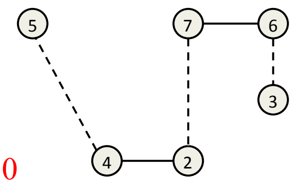
3. L'insieme S degli archi tratteggiati nel grafo G mostrato qui accanto ha una proprietà che lo mette in relazione con la cardinalità α del più grande insieme stabile di G. Mostrate, iniziando da una dicotomia sulla variabile x_1, come si possono usare insiemi con la stessa proprietà di S in un algoritmo di branch-and-bound per calcolare più rapidamente il valore di α.



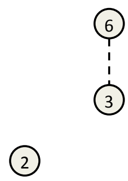
L'insieme S è un edge-cover di G, e come tale la sua cardinalità fornisce una limitazione superiore al numero di stabilità α di G. In questo caso abbiamo quindi $\alpha \le 4$. Un algoritmo di branch-and-bound può iniziare determinando un insieme stabile massimale in modo euristico. Si può ad esempio procedere tramite l'algoritmo Greedy: scegliamo il nodo 1 (eliminando quindi 4, 5, 7), poi il nodo 2 e quindi il nodo 3 (eliminando 6). L'insieme {1, 2, 3} così ottenuto ha tre elementi e non possiamo quindi dichiararlo ottimo usando il bound dato dal nostro edge-cover.

Eseguiamo allora un branching su x_1. Scegliendo $x_1 = 0$ ($x_1 = 1$) riduciamo il problema alla determinazione del più grande insieme stabile nel grafo di sinistra (di destra) nella figura qui sotto.

$x_1 = 0$



$x_1 = 1$



Il grafo di sinistra ammette un matching perfetto (e quindi un edge-cover, indicato ancora una volta a tratteggio) con tre elementi. L'algoritmo Greedy applicato a questo grafo restituisce l'insieme {5, 2, 6} ovvero l'insieme {4, 7, 3}: in entrambi i casi si ha $\alpha = 3$, quindi la soluzione del Greedy condizionata a $x_1 = 0$ è ottima.

Il grafo di destra ha un nodo isolato, 2, quindi non ammette edge-cover. Ma il nodo 2 può appartenere a qualunque insieme stabile di questo grafo, quindi intanto possiamo fissare $x_2 = 1$. Nel grafo rimanente, formato dai soli nodi 3 e 6, il più piccolo edge-cover ha un solo elemento, e così anche il più grande insieme stabile (ancora una volta, il Greedy può fornire due soluzioni distinte, l'insieme {3} o l'insieme {6}). In definitiva unendo a questo insieme i nodi 1 e 2, le cui variabili sono state fissate a 1, otteniamo per il grafo di destra un insieme stabile di tre elementi. Quindi le soluzioni {5, 2, 6}, {4, 7, 3}, {1, 2, 3}, {1, 2, 6} sono tutte ottime.