1. Accoppiamento

Definizione. *Dato un grafo (non orientato) $G = (N, E)$, un sottoinsieme M di archi, tale che ogni nodo del grafo è incidente in al più un arco di M, viene detto* accoppiamento*, (*matching*). I nodi incidenti in M vengono detti nodi* accoppiati*, mentre gli altri nodi vengono detti* esposti*.* ∎

Definizione. *Se il grafo è bipartito, un accoppiamento viene anche detto* assegnamento*.* ∎

Il motivo per cui l'accoppiamento viene detto anche assegnamento nel caso bipartito è ovviamente dovuto al fatto che un accoppiamento realizza una corrispondenza fra due sottoinsiemi di due insiemi dati.

Definizione. *Un accoppiamento in cui tutti i nodi sono accoppiati viene detto* perfetto*.* ∎

Definizione. *Il* problema dell'accoppiamento di cardinalità *consiste nel trovare un accoppiamento di massima cardinalità.* ∎

Se per ogni arco è assegnato un costo c_e, $e \in E$, e se il peso di un insieme M di archi viene semplicemente definito come $c(M) := \sum_{e \in M} c_e$, allora si può definire:

Definizione. *Dato un grafo completo, il* problema dell'accoppiamento pesato *consiste nel trovare un accoppiamento perfetto di costo minimo.* ∎

Definizione. *Dato un grafo bipartito completo, il* problema dell'assegnamento pesato *consiste nel trovare un accoppiamento perfetto di costo minimo.* ∎

Ovviamente nei problemi pesati il grafo deve avere un numero pari di nodi e nel caso bipartito i due insiemi di nodi devono avere la stessa cardinalità.

Anche se i problemi di accoppiamento bipartito sono un caso particolare di quelli definiti su un grafo generale, è tuttavia utile distinguere l'accoppiamento bipartito da quello non bipartito. Non solo un accoppiamento bipartito corrisponde ad un assegnamento e quindi riveste una speciale importanza in alcuni problemi di ricerca operativa, ma avviene anche che la particolare struttura semplifichi in modo essenziale il problema.

Un problema di assegnamento pesato può anche essere definito da una tabella $n \times n$ di pesi, ad esempio:

3	8	2	9	3
7	5	5	1	2
5	3	8	8	2
2	2	5	4	3
6	5	2	5	6

con il compito di scegliere n elementi, esattamente uno per ogni riga e uno per ogni colonna, in modo da minimizzare la somma dei pesi scelti. Nell'esempio l'ottimo è costituito dai pesi evidenziati in grassetto.

3	8	2	9	**3**
7	5	5	**1**	2
5	**3**	8	8	2
2	2	5	4	3
6	5	**2**	5	6

2. Cammini aumentanti

Nello sviluppo di algoritmi per problemi di accoppiamento uno dei concetti chiave è quello di cammino aumentante. Sia $G = (N, E)$ un grafo assegnato (indifferentemente se completo o non, bipartito o generico) e sia $M \subset E$ un accoppiamento assegnato.

Definizione.
– *Un cammino semplice che parte da un nodo esposto e consiste alternativamente di archi accoppiati e archi non accoppiati viene detto* cammino alternante.
– *Un cammino alternante viene detto* cammino aumentante *se termina con un nodo esposto.* ∎

È opportuno notare che le definizioni dipendono dal particolare accoppiamento M. L'importanza dei cammini aumentanti viene espressa dal seguente teorema:

Teorema. *Un accoppiamento è di cardinalità massima se e solo se non esistono cammini aumentanti.*
Dimostrazione: È evidente che se esiste un cammino aumentante è possibile ottenere un nuovo accoppiamento M' tale che $|M'| = |M| + 1$, semplicemente rendendo liberi gli archi accoppiati del cammino e viceversa, e quindi M non può essere ottimo.

Viceversa sia M un accoppiamento per il quale non esistono cammini aumentanti e sia M' un altro accoppiamento. Consideriamo l'insieme di archi $E' := M' \setminus M \cup M \setminus M'$. In generale E' è un insieme sconnesso di archi e, siccome il grado di ogni nodo rispetto a E' è al massimo due, consiste di circuiti e/o cammini disgiunti. Due archi di E' incidenti nello stesso nodo devono necessariamente appartenere l'uno a M e l'altro a M'. Quindi in tutti i cammini e i circuiti di E' gli archi di M e M' si alternano; inoltre i circuiti devono contenere lo stesso numero di archi di M e di M'. Siccome M non ha cammini aumentanti ogni cammino di E' deve contenere alle sue estremità archi di M che quindi risultano in numero maggiore di quelli di M' da cui $|M| \geq |M'|$. Dato che M' è un qualsiasi accoppiamento, M è ottimo. ∎

Il teorema fornisce pertanto un metodo per ottenere un accoppiamento di cardinalità massima: dato un accoppiamento corrente, si tratta di trovare un cammino aumentante, aggiornare l'accoppiamento corrente che chiameremo *accoppiamento aumentato da* M e proseguire iterativamente fino a quando non esistono più cammini aumentanti. Il seguente teorema (dato senza dimostrazione) facilita la ricerca di cammini aumentanti:

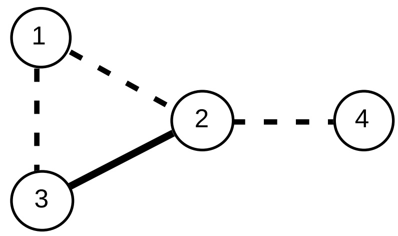
Teorema. *Sia s un nodo esposto in un accoppiamento M e non esista nessun cammino aumentante da s relativamente all'accoppiamento M. Allora non esistono cammini aumentanti da s neppure relativamente ad accoppiamenti aumentati da M.* ∎

Esercizio. Sia $\hat{m}$ la cardinalità di un accoppiamento massimo e sia M un accoppiamento. Si dimostri che esistono $\hat{m} - |M|$ cammini aumentanti disgiunti nei nodi. ∎

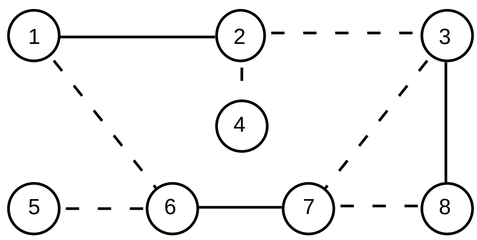
In base al teorema la ricerca di cammini aumentanti può essere fatta in modo sistematico a partire da un nodo esposto alla volta. Un nodo dal quale non esistono cammini aumentanti può non essere più preso in considerazione.

Il problema si sposta allora sulla determinazione di un cammino aumentante da un nodo esposto assegnato. Siccome l'esistenza di un cammino (qualsiasi) fra un nodo ed un insieme assegnato di nodi può essere determinata tramite una ricerca sul grafo, possiamo pensare di usare la stessa tecnica limitata però ai cammini alternanti.

La ricerca di un cammino aumentante su un grafo generico presenta alcuni problemi. Ad esempio nel seguente grafo



la ricerca del cammino aumentante a partire dal nodo 1, raggiungerebbe prima i nodi 2 e 3, poi però i cammino aumentante $1 \to 3 \to 2 \to 4$ non sarebbe più visibile per ché il nodo 2, essendo già stato selezionato, non è pi]'u raggiungibile da 3. L'inconveniente si può superare con una tecnica di ricerca diversa, ad esempio non 'bloccando' i nodi pari del cammino e selezionando solo quelli dispari. Si generano però problemi di altro tipo. Nel seguente grafo



partendo dal nodo 5 e 'saltando' i nodi pari del cammino si selezionano nell'ordine i nodi 5, 7 (da 5 saltando 6), 3 (da 7 saltando 8), 8 (da 7 saltando 3), 1 (da 3 saltando 2), 6 (da 3 saltando 7), 2 (da 6 saltando 1). A questo punto, arrivata la ricerca al nodo 2, adiacente ad un nodo esposto (diverso da quello di partenza) sembrerebbe che si è trovato un cammino aumentante. In realtà il cammino, includendo a anche i nodi saltati è $5 \to 6 \to 7 \to 8 \to 3 \to 7 \to 6 \to 1 \to 2 \to 4$, che presenta un ciclo con dei nodi ripetuti. Quindi non può essere un cammino aumentannte.

Il modo per superare questo inconveniente non è del tutto banale e non viene qui presentato. Si fa solo notare come in entrambi gli esempi il problema è derivato dalla presenza di circuiti dispari. Siccome un grafo è bipartito se e solo se tutti i circuiti sono pari, la ricerca di un cammino aumentante (con complessità $O(m)$, in entrambe le modalità viste) non presenta problemi. Tenuto conto che tale ricerca viene ripetuta al massimo $O(n)$ volte in base al precedente teorema abbiamo il seguente risultato:

Teorema. *Un assegnamento di massima cardinalità si può trovare, eseguendo ricerche di cammini aumentanti, con complessità $O(n\,m)$.* ∎

Si può tuttavia far meglio di così trasformando il problema in un problema di massimo flusso. Si aggiungano al grafo bipartito $G = (N_1, N_2, E)$ un nodo sorgente s ed un nodo pozzo t, si aggiungano archi da s a ogni nodo di N_1 e da ogni nodo di N_2 a t. Tutti gli archi abbiano intervallo di capacità $[0, 1]$. È immediato verificare che la massimizzazione del flusso da s a t è equivalente a realizzare un assegnamento di cardinalità massima in G. Gli archi accoppiati sono esattamente quelli con flusso unitario.

L'algoritmo del massimo flusso nel caso particolare di capacità $[0, 1]$ ha complessità inferiore e inoltre si può dimostare che bastano meno di n cicli aumentanti per il particolare problema di massimo flusso derivato dal problema dell'assegnamento. Questo permette di risolvere un problema di assegnamento di cardinalità con complessità $O(m\sqrt{n})$.

3. Descrizione poliedrale dei problemi dell'assegnamento e dell'accoppiamento

Sia dato un grafo (non necessariamente completo) con dei costi assegnati agli archi e si supponga di voler determinare l'assegnamento di costo massimo (il costo di un assegnamento è la somma dei costi dei singoli archi dell'assegnamento). Per affrontare il problema definiamo le seguenti variabili

$$x_e = \begin{cases} 1 & \text{se l'arco } e \text{ fa parte dell'assegnamento} \\ 0 & \text{altrimenti} \end{cases} \tag{1}$$

Quindi le variabili x definiscono, a seconda dei valori che assumono, tutti i sottoinsiemi di archi. Per ottenere solo sottoinsiemi corrispondenti ad assegnamenti si devono imporre dei vincoli. Siccome per ogni nodo al più un arco del tagio indotto dal nodo deve essere presente si può imporre il seguente vincolo

$$\sum_{e \in \delta(i)} x_e \le 1 \qquad \forall i \in N \tag{2}$$

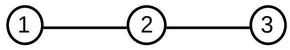
Soluzioni che soddisfano (1) e (2) sono soltanto gli accoppiamenti. Siccome però il vincolo (1) è difficile da trattare, si può pensare di rilassare (1) nel più generico vincolo

$$0 \le x_e \le 1 \tag{3}$$

che ammette anche soluzioni frazionarie, che non possiamo interpretare come sottoinsiemi di archi. L'insieme ammissibile dato dai vincoli (2) e (3) è un poliedro, dato che tutti i vincoli sono diseguaglianze e/o eguaglianze di tipo lineare.

La minimizzazione (o alternativamente massimizzazione) di una funzione lineare su un poliedro definito da diseguaglianze (con eventualmente anche eguaglianze) è il problema detto della Programmazione Lineare. Si tratta di uno dei problemi più studiati e per il quale esistono algoritmi particolarmente efficaci, che sono in grado di risolvere istanze con migliaia di variabili e di vincoli. Inoltre, se si minimizza (o massimizza) una funzione lineare, allora l'ottimo può essere cercato fra i vertici e gli algoritmi di Programmazione Lineare forniscono sempre soluzioni di vertice. Quindi l'interesse maggiore ricade sui punti del poliedro che sono anche vertici. La domanda fondamentale è perciò: che relazione c'è fra i vertici del poliedro dato da (2) e (3) e gli accoppiamenti ammissibili di un grafo?

Prima di rispondere a questa domanda è utile capire con piccoli esempi la struttura del problema. Si consideri il semplice grafo (bipartito):



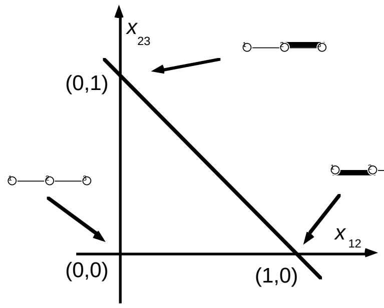
per il quale i vincoli (2) sono

$$x_{12} \le 1, \qquad x_{12} + x_{23} \le 1, \qquad x_{23} \le 1$$

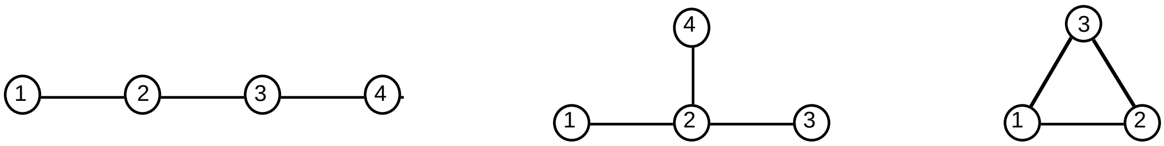
tenendo conto di (3) ed eliminando i vincoli ridondanti (perché già implicati da altri) rimangono le tre diseguaglianze

$$x_{12} \ge 0, \qquad x_{23} \ge 0, \qquad x_{12} + x_{23} \le 1$$

che danno luogo all'insieme ammissibile indicato in figura (il triangolo), i cui vertici hanno coordinate 0-1 e sono quindi identificabili con sottoinsiemi di archi. Come si vede i tre vertici corrispondono esattamente agli unici tre accoppiamenti possibili (includendo anche l'accoppiamento vuoto)



Come secondo esempio consideriamo grafi di tre archi (i primi due sono bipartiti)



Per il primo grafo i vincoli (2) sono

$$x_{12} \leq 1, \qquad x_{12} + x_{23} \leq 1, \qquad x_{23} + x_{34} \leq 1, \qquad x_{34} \leq 1$$

che, insieme a (3) danno luogo all'insieme non ridondante di vincoli

$$x_{12} \geq 0, \qquad x_{23} \geq 0, \qquad x_{34} \geq 0, \qquad x_{12} + x_{23} \leq 1, \qquad x_{23} + x_{34} \leq 1 \tag{4}$$

Per il secondo grafo i vincoli (2) sono

$$x_{12} \leq 1, \qquad x_{12} + x_{23} + x_{34} \leq 1, \qquad x_{34} \leq 1, \qquad x_{24} \leq 1$$

che, insieme a (3) danno luogo all'insieme non ridondante di vincoli

$$x_{12} \geq 0, \qquad x_{23} \geq 0, \qquad x_{34} \geq 0, \qquad x_{12} + x_{23} + x_{34} \leq 1 \tag{5}$$

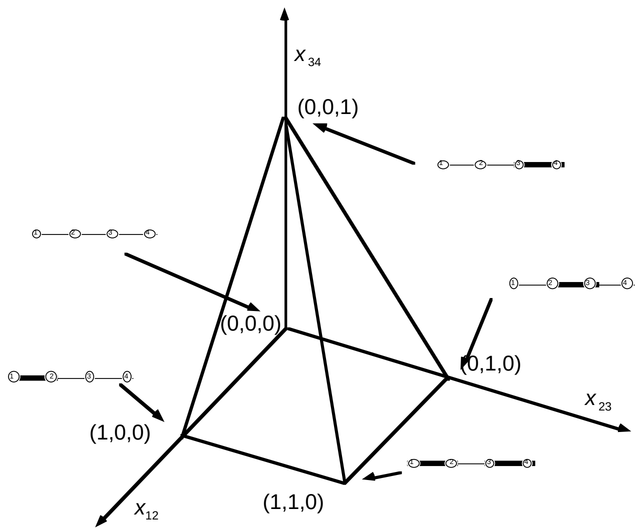
Per il terzo grafo i vincoli (2) sono

$$x_{12} + x_{13} \leq 1, \qquad x_{12} + x_{23} \leq 1, \qquad x_{13} + x_{23} \leq 1$$

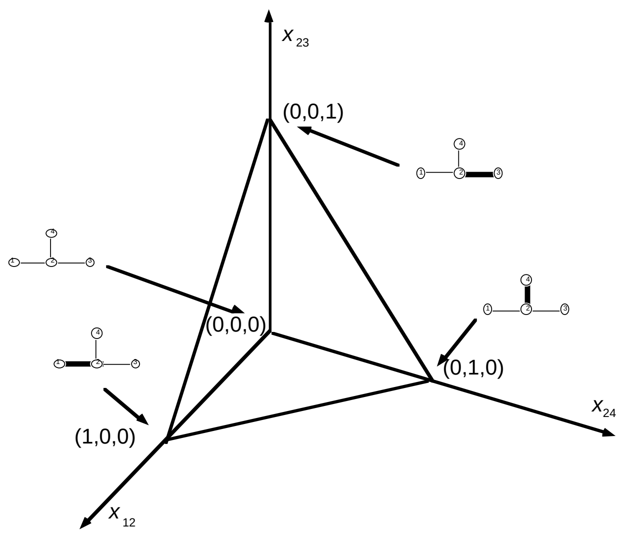
che, insieme a (3) danno luogo all'insieme non ridondante di vincoli

$$x_{12} \geq 0, \qquad x_{23} \geq 0, \qquad x_{34} \geq 0, \qquad x_{12} + x_{13} \leq 1, \qquad x_{12} + x_{23} \leq 1, \qquad x_{13} + x_{23} \leq 1 \tag{6}$$

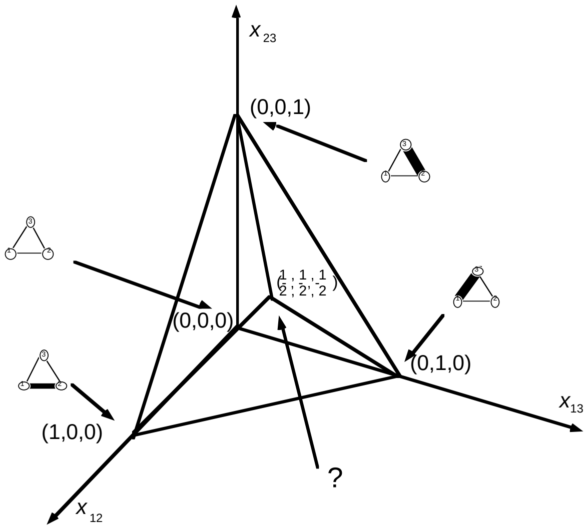
I vincoli (4) generano il seguente poliedro, i cui vertici sono interi e corrispondono agli accoppiamenti indicati in figura. Come si vede, c'è un'esatta corrispondenza biunivoca fra i vertici e tutti gli accoppiamenti possibili.



I vincoli (5) generano il seguente poliedro, anch'esso con vertici interi e accoppiamenti corrispondenti. Anche in questo caso c'è un'esatta corrispondenza biunivoca fra i vertici e tutti gli accoppiamenti possibili.



I vincoli (6) generano un poliedro che presenta, oltre ai vertici interi corrispondenti a tutti gli accoppiamenti anche un vertice di coordinate frazionarie, e che ovviamente non corrisponde ad alcun accoppiamento.



Un importante teorema afferma che per grafi bipartiti il poliedro definito da (2) ha vertici a coordinate intere corrispondenti a tutti gli assegnamenti possibili. Quindi risolvendo un problema di assegnamento con la Programmazione Lineare si ha la garanzia di ottenere una soluzione intera.

Invece, per un grafo generico può succedere che un algoritmo di Programmazione Lineare fornisca una soluzione frazionaria non traducibile come accoppiamento. In questi casi si possono seguire due strade (o anche entrambe): si ripristina il vincolo d'interezza $x_e \in \{0, 1\}$ per una particolare variabile frazionaria x_e, ad esempio imponendo una volta $x_e := 0$ ed un'altra $x_e := 1$ e risolvendo *due* problemi. Si noti che ognuno dei due problemi potrebbe a sua volta richiede la soluzione di altri due problemi e così di seguito portando immediatamente ad un numero esponenziale di problemi da risolvere. Con opportuni accorgimenti, che verranno esposti più avanti, si riesce a contenere il numero di problemi da risolvere e rendere quindi praticabile questa tecnica. Questo approccio prende il nome *branch-and-bound*.

Nel secondo metodo si possono aggiungere vincoli che eliminino la soluzione frazionaria senza però eliminare soluzioni intere. Ad esempio la soluzione $(1/2, /1/2, 1/2)$ può essere eliminata aggiungendo il vincolo

$$x_{12} + x_{13} + x_{23} \leq 1$$

Infatti in un insieme di tre nodi al più un arco può essere accoppiato. In generale per ogni insieme dispari di nodi S un vincolo del tipo

$$\sum_{e \in E(S)} x_e \leq \frac{|S| - 1}{2} \tag{7}$$

è valido in quanto non esclude alcuna soluzione intera. Un teorema molto importante afferma che il poliedro definito dalle seguenti diseguaglianze

$$\begin{aligned} \sum_{e \in \delta(i)} x_e &\leq 1 && \forall i \in N \\ \sum_{e \in E(S)} x_e &\leq \frac{|S| - 1}{2} && \forall S \subset N, \text{ dispari} \\ x_e &\geq 0 \end{aligned} \tag{8}$$

ha vertici interi. Questo non significa però che si può risolvere direttamente il problema di PL con vincoli dati da (8). Infatti sono presenti in (8) un numero esponenziale di vincoli e non è pensabile né formulare né tanto meno risolvere un problema di Pl di tali dimensioni. Ciò che si può invece fare computazionalmente è usare solo quei vincoli (7) che servono effettivamente. A questo scopo si risolve inizialmente solo il problema

$$\begin{aligned} \max \quad & \sum_e c_e\, x_e \\ & \sum_{e \in \delta(i)} x_e \le 1 \qquad \forall i \in N \\ & \qquad x_e \ge 0 \end{aligned} \tag{9}$$

Se la soluzione è intera, il problema è ovviamente risolto e non c'è altro da fare. Se invece la soluzione è frazionaria, il teorema citato assicura che almeno uno dei vincoli (7) è violato dalla soluzione. Si tratta di identificarlo e aggiungerlo a (9). Quindi il problema dell'accoppiamento si risolve risolvendo la successione di problemi:

$$\begin{aligned} \max \quad & \sum_s c_e\ x_e \\ & \sum_{e \in \delta(i)} x_e = 1 && \forall i \in N \\ & \sum_{e \in E(S)} x_e \le (|S|-1)/2 && S = S_1, S_2, \dots S_q \\ & \qquad x_e \ge 0 && \forall e \in E \end{aligned} \tag{10}$$

per $q := 0, 1, \dots,$ e insiemi S_k dispari, generati in modo che la soluzione ottima $\hat{x}^q$ del problema q-mo non sia ammissibile per il problema $(q+1)$-mo.

La successione di problemi (10) termina con l'ottimo. L'identificazione di una diseguaglianza violata da una soluzione frazionaria non è in generale semplice. Tuttavia, nella maggior parte dei casi i valori frazionari sono 1/2, e allora non possono che disporsi secondo un circuito dispari. Che i valori si dispongano secondo un circuito è abbastanza ovvio. Che il circuito sia dispari è meno ovvio. Infatti nessun vertice può avere valori frazionari su un circuito pari, in quanto esiste sempre un $\varepsilon > 0$ per cui la soluzione si può esprimere come combinazione convessa della soluzione ottenuta aggiungendo e sottraendo ε ad archi alterni e della soluzione ottenuta in maniera analoga scambiando gli archi. Il circuito pertanto identifica un insieme S sul quale si ha $\sum_{e \in S} \hat{x}_e = |S|/2$, per cui la soluzione non è ammissibile per il vincolo $\sum_{e \in S} x_e \le (|S|-1)/2$. Tuttavia è bene tenere a mente che soluzioni frazionarie con valori diversi da 1/2 si possono presentare. In questi casi, se non si vuole fare ricorso a tecniche più complesse di identificazione delle diseguaglianze violate, si può affrontare il problema con una tecnica branch-and-bound.

Si veda nelle figure 1, 2, 3 e 4, la risoluzione di un'istanza euclidea con 50 nodi. La prima soluzione, ottenuta con il solo vincolo di grado nei nodi ha valore 211.5 e presenta quattro circuiti dispari con valore 1/2 sugli archi dei circuiti. Aggiungendo la diseguaglianza relativa all'insieme $\{19, 29, 32\}$ si ottiene la soluzione in figura 2 di valore 213.5. Si aggiunge la diseguaglianza relativa a $\{12, 21, 30\}$ e si ottiene la soluzione in figura 3 ancora di valore 213.5. Aggiungendo infine la diseguaglianza relativa a $\{21, 41, 48\}$ si ottiene l'ottimo (figura 4) di valore 215.

Si consideri il problema dell'accoppiamento di massima cardinalità e lo si formuli come un problema di PL intera

$$\begin{aligned} m = \max \quad & \sum_{e \in E} x_e \\ & \sum_{e \in \delta(i)} x_e \le 1 && i \in N \\ & \qquad x_e \in \{0, 1\} && e \in E \end{aligned} \tag{11}$$

Figura 1

Figura 2

Figura 3

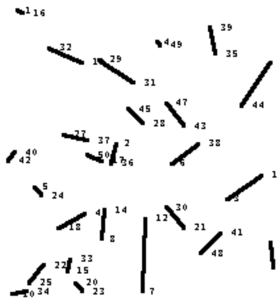


Figura 4

si rilassi il vincolo d'interezza in

$$
\begin{aligned}
v = \max \; & \sum_{e \in E} x_e \\
& \sum_{e \in \delta(i)} x_e \leq 1 \qquad i \in N \\
& \qquad\; x_e \geq 0 \qquad e \in E
\end{aligned}
\tag{12}
$$

Ovviamente si ha $m \leq v$. Il duale di (12) è

$$
\begin{aligned}
d = \min \; & \sum_{i \in N} y_i \\
& y_i + y_j \geq 1 \qquad (i,j) \in E \\
& \qquad\; y_i \geq 0 \qquad i \in N
\end{aligned}
\tag{13}
$$

Per la dualità si ha $m \leq v = d$. Si restringa l'insieme ammissibile di (13) in

$$
\begin{aligned}
c = \min \; & \sum_{i \in N} y_i \\
& y_i + y_j \geq 1 \qquad (i,j) \in E \\
& \quad y_i \in \{0,1\} \qquad i \in N
\end{aligned}
\tag{14}
$$

e quindi $m \leq v = d \leq c$. Si noti che (14) è un problema di minima copertura di nodi. Quindi si ottiene il risultato che il massimo accoppiamento è sempre limitato superiormente dalla minima copertura di nodi. Questo risultato si ottiene agevolmente ragionando direttamente sul problema: infatti per ogni accoppiamento almeno uno dei due nodi dell'accoppiamento deve essere presente in ogni copertura, altrimenti l'arco non sarebbe coperto. Tuttavia il passaggio tramite la PL permetterà di provare l'uguaglianza nel caso speciale di grafi bipartiti.