



CORSO DI LAUREA TRIENNALE IN MATEMATICA  
UNIVERSITÀ DI PISA

**Dal problema di massimo  
accoppiamento al problema  
di trasporto ottimo:  
due generalizzazioni del problema  
di assegnamento**

RELATORE:  
Dario Trevisan

CANDIDATA:  
Arianna Carelli

---

ANNO ACCADEMICO: 2020/2021

# Indice

<b>Introduzione</b>	<b>2</b>
<b>1 Problema di massimo accoppiamento</b>	<b>3</b>
1.1 Problema di massimo accoppiamento per grafi bipartiti . . .	4
1.2 Problema di massimo accoppiamento: una generalizzazione .	6
1.3 Metodo ungherese . . . . .	10
1.3.1 Algoritmo di Hopcroft & Karp . . . . .	13
<b>2 Problema di trasporto ottimo</b>	<b>20</b>
2.1 Problema di assegnamento e rilassamento di Kantorovich . .	20
2.2 Matrici di permutazione e politopo di Birkhoff . . . . .	22
2.3 Caso 1-dimensionale . . . . .	24
2.4 Regolarizzazione entropica . . . . .	26
2.5 Algoritmo di Sinkhorn-Knopp . . . . .	27
2.5.1 Cenni sulla convergenza dell'algoritmo di Sinkhorn- Knopp . . . . .	29
2.6 Sperimentazioni numeriche . . . . .	30
2.6.1 Sperimentazioni relative al teorema di Birkhoff . . . .	30
2.6.2 Sperimentazioni relative al teorema dell'accoppiamen- to monotono . . . . .	34
2.7 Problema duale . . . . .	35
<b>Appendice</b>	<b>38</b>

# Introduzione

Il problema di assegnamento è un problema classico di ottimizzazione combinatoria che trova impiego in diverse applicazioni. In questo lavoro di tesi, esponiamo alcuni risultati e algoritmi inerenti a due generalizzazioni del problema di assegnamento: il problema di massimo accoppiamento su grafi non orientati e non pesati, e il problema di trasporto ottimo secondo Kantorovich.

Nella prima parte, illustriamo due risultati classici caratterizzanti i problemi di programmazione lineare associati rispettivamente al problema di accoppiamento nel caso bipartito (Sezione 1.1), e successivamente, al problema di accoppiamento nel caso generale (Sezione 1.3). Riportiamo dunque lo schema algoritmico (Algoritmo 1) esposto nel libro di Lovász e Plummer [1, §1.2] denominato metodo ungherese, e illustriamo in dettaglio l'Algoritmo 2 di Hopcroft e Karp, studiando in particolare il caso di grafi bipartiti, nel quale risulta molto efficiente.

Nella seconda parte introduciamo la formulazione del problema di trasporto ottimo di Kantorovich, osservando come il problema di assegnamento possa essere visto come un'istanza particolare di tale formulazione. Dunque, definiamo il politopo di Birkhoff ed enunciamo il teorema omonimo 2.1, notando che rappresenta un caso particolare del teorema di massimo accoppiamento di grafi bipartiti di stessa dimensione 1.1. Inoltre, analizziamo il caso speciale unidimensionale ponendo determinate ipotesi sull'insieme di punti, sui vettori di probabilità associati e sulla matrice costo, deducendo una caratterizzazione del piano di trasporto ottimale. Introduciamo poi, un metodo generale per risolvere il problema di Kantorovich: la regolarizzazione entropica (2.8). Dunque enunciamo e dimostriamo il risultato 2.5 che giustifica l'introduzione di tale regolarizzazione. Dalla 2.6 giustifichiamo l'impiego dell'algoritmo di Sinkhorn-Knopp come algoritmo risolutivo per la determinazione della soluzione ottima del problema regolarizzato. In seguito, riportiamo i risultati di alcune sperimentazioni in cui è stato utilizzato l'algoritmo di Sinkhorn per verificare gli enunciati del teorema di Birkhoff e del teorema di accoppiamento monotono 2.3. Infine, introduciamo la definizione del problema duale di Kantorovich (2.17), dimostrando l'uguaglianza dei valori ottimi dei problemi primale e duale con l'ausilio di tecniche basilari di ricerca operativa, riportate in appendice.

# Capitolo 1

## Problema di massimo accoppiamento

In questo capitolo, vediamo alcuni risultati caratterizzanti il problema di massimo accoppiamento [1], dapprima nel caso specifico di grafi bipartiti e successivamente nel caso generale. In particolare, forniamo due risultati caratterizzanti i vertici del politopo del problema di programmazione lineare associato al problema di massimo accoppiamento, sia nel caso di grafi bipartiti 1.1, sia per grafi non orientati 1.3. Infine, riportiamo lo schema algoritmico del metodo ungherese (Algoritmo 1) e illustriamo nel dettaglio l'algoritmo di Hopcroft e Karp (Algoritmo 2), utilizzati rispettivamente per determinare un accoppiamento massimo nel caso di grafi bipartiti e di grafi non orientati e non pesati. Infine, osserviamo che l'algoritmo di Hopcroft e Karp risulta molto efficiente nel caso di grafi bipartiti.

**Definizione 1.1.** Dato un grafo  $G = (V, E)$  non orientato, un *accoppiamento* (*matching*)  $M$  in  $G$  è un sottoinsieme di archi tale che ogni vertice del grafo è incidente in al più un arco di  $M$ . La cardinalità di  $M$  viene detta *dimensione* (*size*) dell'accoppiamento. Infine, i nodi incidenti in  $M$  si definiscono *accoppiati* (*matched*), altrimenti si dicono *esposti* (*exposed*).

**Definizione 1.2.** Un accoppiamento di  $G$  si dice *massimo* (*maximum*) se contiene il massimo numero di archi possibile. Un accoppiamento di  $G$  si dice *perfetto* (*perfect*) se ogni vertice è incidente con un arco dell'accoppiamento.

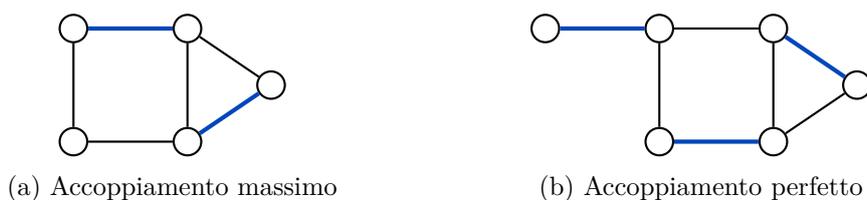


Figura 1.1

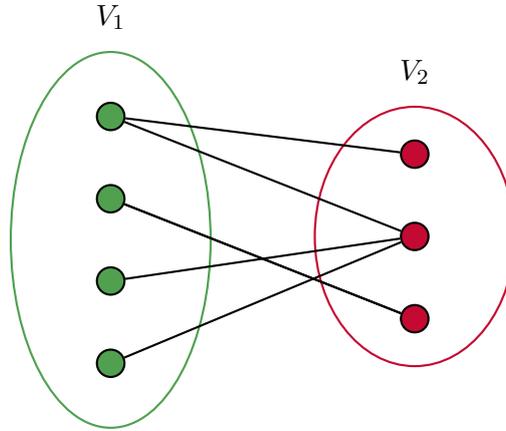


Figura 1.2: Grafo bipartito

**Definizione 1.3.** Un grafo  $G = (V, E)$  si dice *bipartito* (*bipartite*) se l'insieme di vertici  $V$  può essere partizionato in due sottoinsiemi  $V_1$  e  $V_2$  disgiunti, tali che  $V = V_1 \sqcup V_2$  e ogni arco  $(v_1, v_2) \in E$  è tale che  $v_1 \in V_1$  e  $v_2 \in V_2$ .

## 1.1 Problema di massimo accoppiamento per grafi bipartiti

Sia  $G$  un grafo bipartito con bipartizione  $V = U \sqcup W$ . Si indica con  $\mathbb{R}^E$  l'insieme di vettori  $|E|$ -dimensionali le cui entrate sono indicizzate dagli archi di  $G$ . Ogni sottoinsieme  $F \subseteq E$  può essere descritto dal suo *vettore di incidenza*  $q^F$ , definito come

$$q^F = (\alpha_e : e \in E) \in \mathbb{R}^E \quad \text{dove} \quad \alpha_e := \chi_F(e).$$

**Osservazione 1.** Un vettore  $x \in \mathbb{R}^E$  di entrate 0-1 è un vettore di incidenza di un accoppiamento in  $G$  se e solo se  $x(\nabla(v)) \leq 1$  per ogni  $v \in V$ , dove  $\nabla(v)$  è l'insieme degli archi con esattamente un estremo incidente in  $v$ . Infatti, se  $x$  è il vettore di incidenza di un sottoinsieme  $F \subseteq E$ , allora  $x(\nabla(v))$  conta il numero di archi in  $F$  incidenti in  $v$ .

Inoltre, la cardinalità di  $F$  può essere espressa semplicemente in termini del suo vettore di incidenza  $x = q^F$  come

$$|F| = x(E) = x \cdot \mathbf{1}, \tag{1.1}$$

dove  $\mathbf{1}$  è il vettore  $|F|$ -dimensionale di entrate poste uguali a uno.

Dunque, il *problema di massimo accoppiamento* in  $G$  è equivalente a trovare un vettore  $x$  di entrate 0-1 che massimizzi (1.1), soggetto ai vincoli

$x(\nabla(v)) \leq 1$  e  $x_e \geq 0$  dove  $v \in V$  e  $e \in E$ .

In forma più compatta, il problema di massimo accoppiamento può essere formulato come il seguente problema di programmazione lineare:

$$\begin{aligned} \max \quad & \mathbf{1} \cdot x \\ & x \geq 0 \\ & Ax \leq \mathbf{1}, \end{aligned} \tag{1.2}$$

in cui  $A = (a_{ve})$  è la matrice di dimensione  $|V| \times |E|$  denominata *matrice di incidenza vertice-arco di  $G$* , dove  $a_{ve} = 1$  se l'arco  $e$  è incidente nel vertice  $v$  e  $a_{ve} = 0$  altrimenti.

**Osservazione 2.** La formulazione (1.2) del problema di massimo accoppiamento non è equivalente alla formulazione originale. Infatti, non sono presenti i vincoli di integrità, necessari per far sì che  $x$  sia un vettore di incidenza degli archi.

Tuttavia, nel caso di grafi bipartiti possiamo mostrare che tra le soluzioni ottime di (1.2) c'è un vettore 0-1. Più formalmente, si ha il seguente

**Teorema 1.1.** Sia  $G = (V, E)$  un grafo bipartito. Allora i vertici del politopo determinato dalle disuguaglianze in (1.2) sono vettori 0-1. In particolare, sono esattamente i vettori di incidenza degli accoppiamenti.

*Dimostrazione:* Per il teorema fondamentale della programmazione lineare 2.4, tra i punti del politopo determinato da (1.2) che massimizzano il problema, almeno uno è un vertice. Dunque basta dimostrare che i vertici del politopo sono vettori 0-1, in particolare, per i vincoli in (1.2) è sufficiente mostrare che i vertici sono interi.

Si ricorda che un vertice del politopo in  $\mathbb{R}^q$  può essere ottenuto, in modo unico, come intersezione di  $q$  facce del politopo; questo significa selezionare  $q$  righe linearmente indipendenti della matrice  $(A \mid I)^T$  e trasformare le disuguaglianze dei vincoli in uguaglianze e, infine, risolvere il sistema di equazioni lineari risultante.

Eliminando le variabili che sono subito poste uguali a zero, otteniamo

$$A_1 x' = 1$$

dove  $A_1$  è una sottomatrice quadrata della matrice di incidenza  $A$ .

Per la regola di Cramer, le entrate della soluzione sono numeri razionali con denominatore pari a  $\det A_1$ . Dunque, il fatto che la soluzione sia intera segue direttamente dal seguente lemma 1.2 □

**Lemma 1.2.** Se  $G$  è un grafo bipartito, allora ogni sottomatrice quadrata della matrice di incidenza  $A$  ha determinante 0 oppure  $\pm 1$ .

*Dimostrazione:* Sia  $Q$  una qualsiasi sottomatrice quadrata  $k \times k$  di  $A$ . Dimostriamo il risultato per induzione su  $k$ .

*Passo base.* Per  $k = 1$  la tesi segue banalmente.

*Passo induttivo.* Sia  $k > 1$ . Se  $Q$  ha una colonna di soli zeri, applicando la regola di Laplace segue che  $\det Q = 0$ . Se  $Q$  ha una colonna con un solo 1, allora si sviluppa con Laplace su tale colonna e si procede per induzione. Ora supponiamo che ogni colonna di  $Q$  contenga almeno due entrate pari a 1. Le righe di  $Q$  corrispondono ai vertici di  $G$  e dunque possono essere partizionate in due classi, rispettando il partizionamento di  $V$ . Possiamo assumere senza perdita di generalità che le prime  $t$  righe corrispondano a una classe e le rimanenti  $k - t$  all'altra classe. Allora, ogni colonna di  $Q$  ha un 1 nelle prime  $t$  righe e un 1 nelle rimanenti  $k - t$  righe. Questo implica che la somma delle prime  $t$  righe è uguale alla somma delle rimanenti  $k - t$  righe, che è uguale al vettore  $k$ -dimensionale di entrate uguali a 1. Dunque le righe di  $Q$  sono linearmente dipendenti e perciò  $\det Q = 0$ .  $\square$

**Osservazione 3** (Problema di assegnamento). Il problema di massimo accoppiamento di grafi bipartiti *completi* (ogni vertice di una bipartizione è adiacente a ogni vertice della seconda bipartizione) non pesati può essere visto come un'istanza particolare del problema di assegnamento. Quest'ultimo può essere formulato nel modo seguente.

*Siano dati in input  $n$  impiegati e  $m$  incarichi con  $n > m$ . Ogni impiegato può essere assegnato a qualsiasi incarico, richiedendo il pagamento di un costo  $c_{i,j}$  dipendente dall'assegnamento impiegato/incarico. L'obiettivo è di eseguire più incarichi possibili, assegnando ciascun incarico ad al più un impiegato e ciascun impiegato al più un incarico, in modo che il costo totale di impiego sia minimo.*

Più in generale, il problema di assegnamento può essere visto come problema di massimo accoppiamento di grafi bipartiti completi pesati, in cui la somma dei pesi degli archi dell'accoppiamento è minima.

## 1.2 Problema di massimo accoppiamento: una generalizzazione

Nel caso più generale di grafi non bipartiti, il teorema 1.1 non rimane valido. Per esempio, se  $G$  è un triangolo, il punto  $(1/2, 1/2, 1/2)$  è un vertice frazionario del politopo descritto da (1.2). Tuttavia, rimane vero che il vettore di incidenza di ogni accoppiamento appartiene al politopo (1.2).

**Definizione 1.4.** Si definisce *politopo di accoppiamento frazionario* (*fractional matching polytope*), e si denota con  $FM(G)$ , il politopo determinato dai vincoli di (1.2).

**Definizione 1.5.** Un *2-accoppiamento* (*2-matching*) di un grafo  $G$  è un assegnamento di pesi scelti tra 0, 1 o 2 agli archi di  $G$ , in modo che la somma dei pesi degli archi incidenti a ogni punto sia al più 2. Un *2-accoppiamento* si dice *semplice* (*basic*) se i suoi archi di peso 1 formano solo cicli dispari.

Il seguente teorema dà una caratterizzazione dei vertici del politopo  $FM(G)$ .

**Teorema 1.3** (Balinski (1965)). Sia  $w$  un 2-accoppiamento semplice di  $G$ . Allora  $w/2$  è un vertice di  $FM(G)$ , e ogni vertice di  $FM(G)$  ha questa forma.

Prima di dimostrare il teorema 1.3 è necessario premettere alcune definizioni e risultati.

**Definizione 1.6.** Dato un grafo  $G = (V, E)$ , si dice *grado* (*degree*) del vertice  $v \in V$ , e si indica con  $\deg_G(v)$ , il numero di archi incidenti in  $v$ . Inoltre, si definisce *grado massimo* (*maximum degree*) di  $G$  la quantità  $\max_{v \in V} \deg_G(v)$ .

Nel caso di grafi orientati si definiscono il *grado entrante* (*indegree*) e il *grado uscente* (*outdegree*) di  $v \in V$ , come il grado di  $v$  ristretto rispettivamente ai suoi archi entranti e uscenti.

**Definizione 1.7.** Un grafo  $G = (V, E)$  si dice *k-regolare* (*k-regular*) se ogni vertice ha grado  $k$  o, più generalmente, si dice che il grafo è *regolare*.

**Definizione 1.8.** Dato un grafo  $G = (V, E)$ , si definisce *2-fattore* (*2-factor*) un sottoinsieme di archi  $F$ , tale che ogni vertice sia incidente con esattamente due archi di  $F$ .

**Teorema 1.4** (P. Hall (1935)). Sia  $G = (A, B)$  un grafo bipartito. Allora esiste un accoppiamento di  $A$  in  $B$  (ovvero un accoppiamento in cui tutti i vertici di  $A$  sono estremi di un arco dell'accoppiamento) se e solo se

$$|\Gamma(X)| \geq |X| \quad \text{per ogni } X \subseteq A,$$

dove  $\Gamma(X)$  è l'insieme dei vertici adiacenti a elementi di  $X$ .

*Dimostrazione:* Si osserva che se esiste un accoppiamento di  $A$  in  $B$  la disuguaglianza è valida per ogni  $X \subseteq A$ .

Supponiamo ora che valga  $|\Gamma(X)| \geq |X|$  per ogni  $X \subseteq A$ . Procediamo per induzione sulla cardinalità di  $A$ .

*Passo base.* Se  $|A| = \{0, 1\}$  il risultato vale banalmente.

*Passo induttivo.* Si distinguono due casi.

- (a) Supponiamo che per ogni  $X \subset A$ ,  $X \neq \emptyset$ , valga  $|X| < |\Gamma(X)|$ . Siano  $a, b$  vertici adiacenti con  $a \in A$ . Sia  $G' = G \setminus \{a, b\}$  e sia  $X \subset A \setminus \{a\}$ . Se  $X = \emptyset$ , allora  $|X| = 0 = |\Gamma_{G'}(X)|$ . Dunque supponiamo  $X \neq \emptyset$ . Dato che  $X \neq A$ , per ipotesi vale  $|X| < |\Gamma_{G'}(X)|$  e dunque

$$|\Gamma_{G'}(X)| \geq |\Gamma_G(X)| - 1 \geq |X|.$$

Per ipotesi induttiva esiste un accoppiamento  $M'$  di  $G'$  che ricopre ogni vertice di  $A \setminus \{a\}$ . Ma allora  $M = M' \cup \{(a, b)\}$  è un accoppiamento di  $A$  in  $B$  come voluto.

- (b) Se  $|A| = |\Gamma(A)|$ , la tesi è banale. Supponiamo ora che esista  $A' \subset A$ ,  $A' \neq \emptyset$  con  $|\Gamma_G(A')| = |A'|$ . Procediamo dividendo  $G$  in due sottografi. Si definiscono i sottografi indotti  $G_1$  e  $G_2$  rispettivamente da  $A' \cup \Gamma(A')$  e  $G \setminus G_1$ . Infine, si mostra che  $G_1$  e  $G_2$  soddisfano l'ipotesi induttiva.

Supponiamo  $X \subseteq A'$ . Allora  $\Gamma_G(X) \subseteq \Gamma_G(A')$ , dunque  $\Gamma_{G_1}(X) = \Gamma_G(X)$  e quindi

$$|\Gamma_{G_1}(X)| = |\Gamma_G(X)| \geq |X|.$$

Ora assumiamo che valga  $X \subseteq A \setminus A'$  in  $G_2$ . Allora

$$|\Gamma_G(X \cup A')| = |\Gamma_{G_2}(X) \cup \Gamma_G(A')|,$$

e quindi

$$\begin{aligned} |\Gamma_{G_2}(X)| &= |\Gamma_G(X \cup A')| - |\Gamma_G(A')| \\ &\geq |X \cup A'| - |\Gamma_G(A')| \\ &= |X \cup A'| - |A'| = |X|, \end{aligned}$$

dato che  $X \cap A' = \emptyset$  e  $\Gamma_{G_2}(X) \cap \Gamma_G(A') = \emptyset$ .

Applicando l'ipotesi induttiva a  $G_1$  e  $G_2$  segue che devono esistere due accoppiamenti  $M_1$  e  $M_2$  rispettivamente di  $A'$  in  $\Gamma_G(A')$  e di  $A \setminus A'$  in  $B \setminus \Gamma_G(A')$ . L'insieme  $M = M_1 \cup M_2$  è l'accoppiamento voluto.  $\square$

**Lemma 1.5.** Se  $G$  è un grafo bipartito regolare allora ammette un accoppiamento perfetto.

*Dimostrazione:* Sia  $G = (U \cup V, E)$  un grafo bipartito regolare. Allora vale la proprietà

$$|U| = |V|.$$

Infatti, sia  $d$  il valore del grado di un qualunque vertice di  $G$  e poniamo  $|U| = n$  e  $|V| = m$ . Per assurdo, se fosse  $n > m$  (il viceversa è analogo) visto che il grafo è  $d$ -regolare dovremmo distribuire  $d \cdot n$  archi da  $U$  a  $V$ . Ma  $n > m$ , dunque esiste un vertice in  $V$  con grado almeno  $d + 1$ .

Concludiamo utilizzando il teorema di Hall 1.4, dunque dimostriamo che vale  $|\Gamma(X)| \geq |X|$  per ogni  $X \subseteq U$ .

Siano  $E(X)$  ed  $E(\Gamma(X))$  gli insiemi degli archi incidenti rispettivamente a  $X$  e  $\Gamma(X)$ . Si osserva che ogni arco incidente in  $X$  è anche incidente in  $\Gamma(X)$ , perciò  $|E(X)| \leq |E(\Gamma(X))|$ . Dalla regolarità del grafo valgono le seguenti uguaglianze

$$|E(X)| = d \cdot |X|, \quad |E(\Gamma(X))| = d \cdot |\Gamma(X)|.$$

Da cui si ottiene  $|X| \leq |\Gamma(X)|$ . □

**Definizione 1.9.** Dato un grafo  $G = (V, E)$ , si definisce *colorazione valida* (*valid coloring*) degli archi un assegnamento di interi, o *colori*, agli archi di  $G$  in modo che nessun arco di stesso colore sia adiacente.

Il più piccolo numero di colori necessario per produrre una valida colorazione degli archi si chiama *indice cromatico* (*chromatic index*) e si indica con  $\chi_e(G)$ .

**Lemma 1.6.** Sia  $G$  un grafo bipartito regolare di grado  $\Delta$ . Allora vale la seguente

$$\chi_e(G) = \Delta.$$

*Dimostrazione:* La dimostrazione segue dal lemma 1.5 eliminando progressivamente  $\Delta$  accoppiamenti perfetti. Infatti, per il lemma 1.5 esiste un accoppiamento perfetto  $M$ , ai cui archi possiamo associare un colore, cosicché  $M$  sia una valida colorazione di  $G$ . Quindi si eliminano gli archi di  $M$  e il grafo diventa  $(\Delta - 1)$ -regolare. Dunque si ripete il procedimento. □

**Definizione 1.10.** Un grafo  $G$  si dice *connesso* (*connected*) se per ogni coppia di vertici  $u, v \in V$  esiste un cammino in  $G$  che li collega.

**Teorema 1.7** (Petersen). Sia  $G$  un grafo  $(2r)$ -regolare. Allora l'insieme degli archi  $E$  può essere decomposto nell'insieme di  $r$  2-fattori con insiemi di archi disgiunti.

*Dimostrazione:* Innanzitutto, notiamo che è sufficiente considerare grafi connessi; infatti, nel caso di grafi non connessi basta restringersi alle componenti connesse (la proprietà di regolarità viene ereditata). Per il teorema di Eulero, un grafo connesso  $k$ -regolare con  $k$  pari ammette un *ciclo euleriano* (ovvero un cammino chiuso che tocca tutti gli archi una sola volta). Attraversando il ciclo euleriano otteniamo un'orientazione  $D$  di  $G$  tale che

ogni vertice ha grado entrante e uscente pari a  $r$ . Successivamente, rimpiazziamo ogni vertice  $v \in V(D)$  con due vertici  $v', v''$  e per ogni arco orientato  $(u, v) \in E$  disegniamo un arco da  $u'$  a  $v''$ . Dato che  $D$  ha grado entrante e uscente uguale a  $r$  il bigrafo risultante  $G'$  è  $r$ -regolare. Dal lemma 1.6 gli archi di  $G'$  possono essere decomposti in  $r$  accoppiamenti perfetti. A questo punto, se identifichiamo  $v'$  e  $v''$  per ogni  $v$ , ritroviamo il grafo  $G$ , e questi  $r$  accoppiamenti perfetti di  $G'$  saranno mappati in  $r$  2-fattori di  $G$  che partizionano di archi.  $\square$

A questo punto riportiamo la dimostrazione del teorema di Balinski 1.3.  
*Dimostrazione:* Il punto  $w/2$  è l'unico di  $FM(G)$  che massimizza la funzione obiettivo avente coefficienti

$$\begin{cases} c_e = w_e & \text{se } w_e > 0 \\ c_e = -1 & \text{se } w_e = 0. \end{cases}$$

Dunque è un vertice.

Sia ora  $y = (y_e : e \in E)$  un vertice di  $FM(G)$ . Mostriamo che  $y$  è della forma  $n/2$  dove  $n \in \{0, 1, 2\}^E$ . Sia  $2t$  un denominatore comune delle entrate di  $y$ . Definiamo un grafo ausiliario  $G'$  rimpiazzando ogni arco di  $G$  con  $2ty_e$  copie parallele. Per ipotesi  $y$  soddisfa  $y(\nabla(v)) \leq 1$ , da cui  $G$  ha *massimo grado* al più  $2t$ .

Ora includiamo  $G'$  in un grafo  $G''$   $2t$ -regolare. Per il teorema di Petersen 1.7,  $G''$  è l'unione di  $t$  2-fattori, dunque  $G'$  è l'unione di  $t$  sottografi, ognuno di essi aventi *massimo grado* al più 2. Questo implica che  $G$  ha  $t$  2-accoppiamenti  $w_1, \dots, w_t$  tali che  $2ty = w_1 + \dots + w_t$ . Poniamo  $x_i = w_i/2$ , allora

$$y = \frac{x_1 + \dots + x_t}{t}, \quad x_1, \dots, x_t \in FM(G).$$

Dato che per ipotesi  $y$  è un vertice di  $FM(G)$ , segue che

$$y = x_1 = \dots = x_t.$$

Dunque,  $y$  ha entrate del tipo  $n/2$  con  $n \in \{0, 1, 2\}$  e quindi  $2y$  è un 2-accoppiamento.

Resta da dimostrare che i vertici di  $FM(G)$  sono 2-accoppiamenti semplici. Questo segue notando che i 2-accoppiamenti non semplici possono essere rappresentati come combinazione convessa di quelli semplici e dunque non possono essere vertici.  $\square$

### 1.3 Metodo ungherese

**Definizione 1.11.** Sia  $G$  un grafo e  $M$  un accoppiamento in  $G$ . Un cammino  $P = v_1, v_2, \dots, v_m$  si dice *cammino  $M$ -alternante* ( *$M$ -alternating path*) se  $v_i, v_{i+1} \in M$  ma  $v_{i+1}, v_{i+2} \notin M$  per  $1 \leq i \leq m - 2$ .

**Osservazione 4.** Se  $G$  contiene un cammino  $M$ -alternante  $P$  che collega due vertici esposti allora  $M$  non può essere un accoppiamento massimo. Infatti, si può ottenere un accoppiamento  $M'$  più grande scartando gli archi di  $P \cap M$  e aggiungendo quelli di  $P \setminus M$ . Questo segue osservando che un cammino  $M$ -alternante  $P$  per poter collegare due vertici esposti deve essere incidente in essi tramite archi non appartenenti a  $M$ . Per l'alternanza degli archi, se indichiamo con  $n$  il numero di archi di  $P \cap M$ , allora il numero di archi di  $P \setminus M$  è  $n + 1$ .

L'osservazione sopra introduce naturalmente la seguente definizione

**Definizione 1.12.** Un cammino  $M$ -alternante che collega due vertici esposti si definisce  $M$ -aumentante ( *$M$ -augmenting path*).

**Teorema 1.8.** Sia  $M$  un accoppiamento in un grafo  $G$ . Allora  $M$  è un accoppiamento massimo se e solo se non esiste alcun cammino  $M$ -aumentante in  $G$ .

*Dimostrazione:* La condizione necessaria è stata dimostrata nell'osservazione 4. Vediamo la sufficienza. Dobbiamo mostrare che se  $M$  non ammette alcun cammino aumentante allora è massimo.

Sia  $M'$  un accoppiamento in  $G$ . Per l'argomento precedente non esiste alcun cammino  $M'$ -aumentante. Definiamo ora  $H$  come il sottografo di  $G$  avente  $E(H) = M \triangle M'$ , dove  $\triangle$  indica l'operazione di differenza simmetrica. Studiamo le possibili componenti connesse di  $H$ . Innanzitutto, osserviamo che ogni vertice ha grado 0, 1 oppure 2 in  $H$ . I vertici di grado 0 sono disgiunti e quindi sono componenti connesse banali. Se una componente ha tutti i vertici di grado 2 allora deve essere un ciclo pari che alterna tra archi di  $M$  e archi di  $M'$ . Se una componente ha un vertice di grado 1 deve essere un cammino alternante tra archi di  $M$  e  $M'$ . Visto che sia  $M$  che  $M'$  non hanno cammini aumentanti in  $G$ , le componenti connesse devono essere tali da cominciare con un arco in un accoppiamento e terminare con un arco nell'altro accoppiamento.

In ogni caso, ogni componente non banale di  $H$  ha esattamente metà dei suoi archi da  $M$  e l'altra metà da  $M'$ . Dunque, segue che  $|M \setminus M'| = |M' \setminus M|$ , da cui

$$|M| = |M \cap M'| + |M \setminus M'| = |M \cap M'| + |M' \setminus M| = |M'|$$

e quindi  $M$  è un accoppiamento massimo di  $G$ . □

Sia  $G$  un bigrafo con bipartizione  $(A, B)$ , e sia  $M$  un accoppiamento in  $G$ . Poniamo  $A_1$  e  $B_1$  gli insiemi dei vertici esposti in  $A$  e  $B$  rispettivamente. Vogliamo trovare un cammino  $M$ -aumentante che collega  $A_1$  a  $B_1$ .

Premettiamo la seguente.

**Definizione 1.13.** Dato un grafo  $G = (V, E)$ , una foresta (forest)  $F = (V(F), E(F))$  di  $G$  è un sottografo aciclico di  $G$ . La foresta  $F$  si dice *massimale*, se ogni arco  $e \in E \setminus E(F)$  chiude un ciclo in  $F$ .

Costruiamo una foresta massimale  $F$  in  $G$  con le seguenti proprietà:

- (i) Ogni vertice  $b$  di  $F$  in  $B$ , ha grado 2 e uno degli archi di  $F$  incidente in  $b$  appartiene a  $M$ ;
- (ii) Ogni componente connessa di  $F$  contiene un vertice di  $A_1$ .

Si osserva che  $A_1 \subseteq V(F)$  perché ogni vertice di  $A_1$  non in  $F$  può essere aggiunto a  $V(F)$ .

**Definizione 1.14.** Dato un grafo  $G = (V, E)$ , si definiscono

- $\tau(G) := \max \{|M| \mid M \text{ è un accoppiamento}\}$ ;
- $\nu(G) := \min \{|P| \mid P \text{ è una copertura dei vertici}\}$ .

**Lemma 1.9.** Siano  $G, M, A, B, A_1, B_1$  e  $F$  definiti come sopra, allora  $M$  è un accoppiamento massimo se e solo se nessun vertice di  $B_1$  è adiacente ad un vertice di  $F$ .

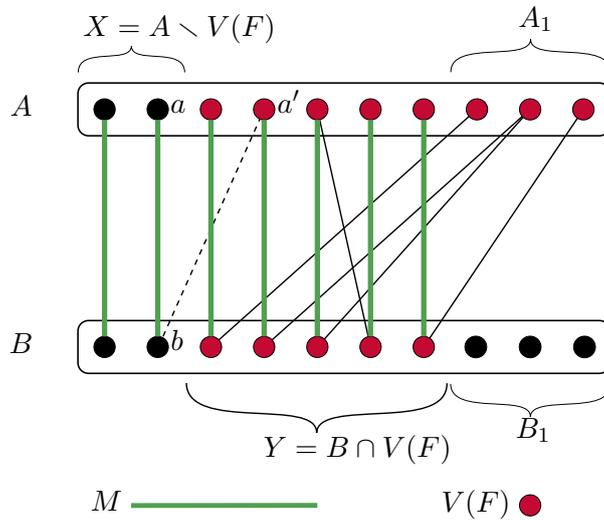


Figura 1.3

*Dimostrazione:* Supponiamo anzitutto che ci sia un vertice  $x$  di  $F$  adiacente a un vertice  $b_1 \in B_1$ . Allora  $x$  è un vertice in  $A$ , e per la proprietà (ii) e dalla massimalità di  $F$  esiste un cammino  $P$  che collega  $x$  a un vertice  $a \in A_1$ . Allora  $P + xb_1$  è un cammino  $M$ -aumentante e dunque  $M$  non è un accoppiamento massimo.

Viceversa, supponiamo che nessun vertice di  $F$  sia collegato a un vertice di  $B_1$ . Definiamo  $X = A \setminus V(F)$  e  $Y = B \cap V(F)$ . Dimostriamo che  $X \cup Y$  è una *copertura di vertici*, ovvero un sottoinsieme di  $V$  tale che ogni arco ha almeno un estremo in tale insieme, e che  $|X \cup Y| = |M|$ .

Mostriamo anzitutto che ogni arco di  $M$  ha almeno un estremo in  $X \cup Y$ . Per definizione di  $X, Y$  segue immediatamente che  $M$  ricopre  $X \cup Y$ . Sia ora  $m$  un generico arco di  $M$ . Se  $m \in E(F)$ , allora ha un estremo in  $Y$ , viceversa ha un estremo in  $X$ . Ora mostriamo che ogni arco di  $M$  ha esattamente un estremo incidente in  $X \cup Y$ , da cui segue che  $|X \cup Y| = |M|$ . Si nota che nessun arco di  $M$  collega due vertici in  $X \cup Y$ , perché se  $xy \in M$ ,  $y \in V(F) \cap B$ , per la proprietà (i) segue che  $x \in V(F)$ , assurdo. Dunque  $|X \cup Y| \leq |M|$ .

Resta da dimostrare che  $X \cup Y$  è una copertura di vertici. Supponiamo per assurdo che esista un arco  $ab$  che non sia coperto da  $X \cup Y$ , dove  $a \in A$  e  $b \in B$ . Allora  $a \in V(F)$  e  $b \notin V(F)$ . Inoltre, per ipotesi,  $b \notin B_1$ . Dunque l'accoppiamento  $M$  copre  $b$ , supponiamo con l'arco  $a'b$ . Inoltre  $a' \neq a$  visto che se  $a$  è accoppiato da  $M$ , deve esserlo in  $Y$ , perché ogni arco di  $M$  ha esattamente un estremo in  $X \cup Y$ . Ma ora  $F$  può essere estesa a una foresta  $F'$  contenente il cammino  $aba'$ , contraddicendo la massimalità. Perciò  $X \cup Y$  ricopre  $E$  e abbiamo

$$\nu(G) \leq |X \cup Y| = |M|.$$

Inoltre vale la seguente

$$\tau(G) \leq \nu(G).$$

Infatti, sia  $M$  un qualsiasi accoppiamento e  $U$  una qualsiasi copertura dei vertici. Per ogni vertice  $uv \in M$ , almeno uno tra  $u, v$  è in  $U$ , altrimenti l'arco  $uv$  non sarebbe coperto da  $U$ . Dunque segue la disuguaglianza e  $M$  è un accoppiamento massimo.  $\square$

L'Algoritmo 1 riporta uno schema algoritmico basato sulla dimostrazione precedente per trovare un accoppiamento massimo di un grafo bipartito.

### 1.3.1 Algoritmo di Hopcroft & Karp

Seppur l'Algoritmo 1 fornisca un esempio implementativo per determinare un accoppiamento massimo di un grafo bipartito, esso non rappresenta una scelta algoritmica computazionalmente efficiente.

Riportiamo dunque l'algoritmo di Hopcroft e Karp, che tutt'oggi risulta il metodo più efficace per determinare un accoppiamento massimo di grafi bipartiti. In primo luogo, presentiamo l'algoritmo nel caso generale di grafi non pesati e successivamente nel caso bipartito, notando come, in quest'ultimo caso, la geometria del grafo permetta di migliorarne considerevolmente la velocità.

**Algoritmo 1** METODO UNGHERESE

*Step 0.* Viene dato in input un qualsiasi accoppiamento  $M$ .

*Step 1.* Si costruisce una foresta massimale  $F$  che rispetta le proprietà (i) e (ii) sopra.

*Step 2.* Se esiste un arco che collega l'insieme  $V(F) \cap A$  a un vertice di  $B_1$ , possiamo trovare un accoppiamento più grande  $M'$  come fatto nella dimostrazione del Lemma 1.9. Ripeti *Step 1*.

*Step 3.* Se nessun arco collega  $V(F) \cap A$  a  $B_1$ , allora per il Lemma 1.9,  $M$  è un accoppiamento massimo. Fine.

*Caso generale.* Sia  $G = (V, E)$  un grafo finito, non orientato, senza cicli, archi multipli (archi distinti che sono incidenti negli stessi vertici) e vertici isolati. Ricordiamo che dall'osservazione 4, segue la seguente proprietà

**Lemma 1.10.** Se  $M$  è un accoppiamento in  $G$  e  $P$  un cammino aumentante relativo a  $M$ , allora  $M \triangle P$  è un accoppiamento, e  $|M \triangle P| = |M| + 1$ .

**Teorema 1.11.** Siano  $M$  e  $N$  due accoppiamenti. Se  $|M| = r$  e  $|N| = s$ ,  $s > r$ , allora  $M \triangle N$  contiene almeno  $s - r$  cammini aumentanti *disgiunti sui vertici* relativi a  $M$ .

Si ricorda che due insiemi di archi  $E, F$  di un grafo  $G$  sono *disgiunti sui vertici* se non esiste alcun vertice di  $G$  che è incidente sia a un arco di  $E$  che a un arco di  $F$ .

*Dimostrazione:* Consideriamo il grafo  $\bar{G} = (V, M \triangle N)$ . Dato che  $M, N$  sono accoppiamenti, ogni vertice di  $G$  è incidente in al più un vertice di  $M \setminus N$  e al più in un vertice di  $N \setminus M$ . Dunque ogni componente connessa di  $G$  è una delle seguenti

- (i) un vertice isolato;
- (ii) un ciclo di lunghezza pari, con archi alternativamente in  $M \setminus N$  e in  $N \setminus M$ ;
- (iii) un cammino i cui archi sono alternativamente in  $M \setminus N$  e in  $N \setminus M$ .

Denotiamo le componenti connesse di  $\bar{G}$  come  $C_1, C_2, \dots, C_g$ , dove  $C_i := (V_i, E_i)$ . Poniamo

$$\delta(C_i) := |E_i \cap N| - |E_i \cap M|.$$

Allora,  $\delta(C_i) \in \{\pm 1, 0\}$ , e  $\delta(C_i) = 1$  se e solo se  $C_i$  è un cammino aumentante relativo a  $M$ . Segue

$$\sum_i \delta(C_i) = |N \setminus M| - |M \setminus N| = |N| - |M| = s - r.$$

Dunque, ci sono almeno  $s - r$  componenti  $C_i$  di  $\overline{G}$  tali che  $\delta(C_i) = 1$ . Tali componenti sono disgiunte sui vertici, e ognuna è un cammino aumentante relativo a  $M$ , da cui segue la tesi.  $\square$

**Corollario 1.12.** [3, §2, Corollario 2] Sia  $M$  un accoppiamento in  $G$ . Supponiamo che  $|M| = r$  e che la cardinalità di un accoppiamento massimo sia  $s, s > r$ . Allora esiste un cammino aumentante relativo a  $M$  di lunghezza  $\leq 2\lceil r/(s - r) \rceil + 1$ .

**Definizione 1.15.** Sia  $M$  un accoppiamento in  $G$ . Un cammino  $P$  si dice *minimo* cammino aumentante relativo a  $M$  se ha la cardinalità minore rispetto ad altri cammini aumentanti relativi a  $M$ .

**Teorema 1.13.** Sia  $M$  un accoppiamento in  $G$ ,  $P$  un minimo cammino aumentante rispetto a  $M$  e  $P'$  un cammino aumentante rispetto a  $M \triangle P$ . Allora,

$$|P'| \geq |P| + |P \cap P'|.$$

*Dimostrazione:* Sia  $N := M \triangle P \triangle P'$ . Allora,  $N$  è un accoppiamento e per il Lemma 1.10 segue che  $|N| = |M| + 2$ , dunque, per il Teorema 1.13,  $M \triangle N$  contiene due cammini aumentanti con vertici disgiunti relativi a  $M$ ; chiamiamoli  $P_1$  e  $P_2$ . Dato che

$$M \triangle N = M \triangle M \triangle P \triangle P' = P \triangle P',$$

si ha  $|P \triangle P'| \geq |P_1| + |P_2|$ . Ma  $|P_1| \geq |P|$  e  $|P_2| \geq |P|$ , visto che  $P$  è il cammino aumentante minimo. Dunque

$$|P \triangle P'| \geq |P_1| + |P_2| \geq 2|P|.$$

Inoltre, dall'identità

$$|P \triangle P'| = |P| + |P'| - |P \cap P'|,$$

segue

$$|P'| \geq |P| + |P \cap P'|.$$

$\square$

**Corollario 1.14.** [3, §2, Corollario 3-4] Vale la disuguaglianza

$$|P_i| \leq |P_{i+1}|.$$

Inoltre, per ogni  $i, j, i \neq j$  tali che  $|P_i| = |P_j|$ , si ha che  $P_i$  e  $P_j$  sono disgiunti sui vertici.

**Teorema 1.15.** Sia  $s$  la cardinalità di un accoppiamento massimo. Allora il numero di interi distinti nella sequenza

$$|P_0|, |P_1|, \dots, |P_i|, \dots$$

è al più  $2\lfloor\sqrt{s}\rfloor + 2$ .

*Dimostrazione:* Sia  $r = \lfloor s - \sqrt{s} \rfloor$ . Allora  $|M_r| = r$  e, dal Corollario 1.12 segue

$$|P_r| \leq 2\lfloor s - \sqrt{s} \rfloor / (s - \lfloor s - \sqrt{s} \rfloor) + 1 \leq 2\lfloor\sqrt{s}\rfloor + 1.$$

Dunque, per ogni  $i < r$ ,  $|P_i|$  è uno dei  $\lfloor\sqrt{s}\rfloor + 1$  interi positivi dispari minori o uguali di  $2\lfloor\sqrt{s}\rfloor + 1$ . Inoltre,  $|P_{r+1}|, \dots, |P_s|$  contribuisce al più  $s - r = \lceil\sqrt{s}\rceil$  interi distinti. Dunque, il numero totale di interi distinti è minore o uguale a

$$\lfloor\sqrt{s}\rfloor + 1 + \lceil\sqrt{s}\rceil \leq 2\lfloor\sqrt{s}\rfloor + 2.$$

□

**Osservazione 5.** Come conseguenza del Corollario 1.14 e del Teorema 1.15, la computazione della sequenza  $\{M_i\}$  si divide in al più  $2\lfloor\sqrt{s}\rfloor + 2$  fasi, nelle quali i cammini aumentanti trovati hanno stessa lunghezza e sono disgiunti sui vertici.

I risultati esposti finora permettono di introdurre uno schema algoritmico per determinare un accoppiamento massimo di grafi non pesati.

---

**Algoritmo 2** ALGORITMO A

---

*Step 0.*  $M \leftarrow \emptyset$ .

*Step 1.* Sia  $l(M)$  la lunghezza di un cammino aumentante minimo relativo a  $M$ . Trova un insieme massimale di cammini  $\{Q_1, Q_2, \dots, Q_t\}$  che rispetti le seguenti proprietà

- (a) per ogni  $i$ ,  $Q_i$  è un cammino aumentante relativo a  $M$  e  $|Q_i| = l(M)$ ;
- (b) i cammini  $Q_i$  sono disgiunti sui vertici.

Stop se non esiste alcun cammino che soddisfi la proprietà  $\rightarrow M$  è un accoppiamento massimo (Teorema 1.8).

*Step 2.*  $M \leftarrow M \triangle Q_1 \triangle Q_2 \triangle \dots \triangle Q_t$ ; ripeti *Step 1*.

---

L'implementazione esposta nell'Algoritmo 2 suggerisce di raccogliere i cammini aumentanti relativi a una stessa fase in un solo step. La geometria del caso bipartito permette una tale ottimizzazione.

*Caso bipartito.* Sia  $G = (X \cup Y, E)$  un grafo bipartito e  $M$  un accoppiamento su  $G$ . Discutiamo l'implementazione dello *Step 1* dell'Algoritmo 2.

Per prima cosa, direzioniamo il grafo  $G$  in modo che i cammini aumentanti relativi a  $M$  siano orientati. Ossia, orientiamo ogni arco in  $E \setminus M$  in modo che gli archi siano entranti in  $X$ , e ogni arco in  $M$  in modo che gli archi siano entranti in  $Y$ . Il grafo orientato risultante viene indicato come  $\bar{G} = (V, \bar{E})$ , dove

$$\begin{aligned} \bar{E} = & \{(y, x) \mid \{x, y\} \in E \setminus M, x \in X, y \in Y\} \\ & \cup \{(x, y) \mid \{x, y\} \in M, x \in X, y \in Y\}. \end{aligned}$$

Successivamente, estraiamo un sottografo  $\hat{G}$  di  $\bar{G}$ , con la proprietà che i cammini diretti di  $\hat{G}$  che partono da un vertice esposto di  $X$  e terminano in un vertice esposto di  $Y$  sono in corrispondenza biunivoca con i cammini aumentanti minimi in  $G$  relativi a  $M$ .

Sia  $L_0$  l'insieme dei vertici esposti di  $X$  e siano

$$\begin{aligned} E_i &:= \{(u, v) \mid (u, v) \in \bar{E}, v \in L_i, u \notin L_0 \cup L_1 \cup \dots \cup L_i\}, \\ L_{i+1} &:= \{u \mid (u, v) \in E_i, v \in V\} \end{aligned}$$

per  $i = 0, 1, 2, \dots$

Poniamo

$$i^* := \min \{i \mid L_i \cap \{\text{vertici esposti di } Y\} \neq \emptyset\}.$$

Allora definiamo  $\hat{G} = (\hat{V}, \hat{E})$ , dove

$$\begin{aligned} \hat{V} &= L_0 \cup L_1 \cup \dots \cup L_{i^*-1} \cup (L_{i^*} \cap \{\text{vertici esposti di } Y\}), \\ \hat{E} &= E_0 \cup E_1 \cup \dots \cup E_{i^*-2} \cup \{(u, v) \mid v \in L_{i^*-1}, u \in \text{vertici esposti di } Y\}. \end{aligned}$$

Le proprietà seguenti di  $\hat{G}$  sono immediate:

- (i) i vertici appartenenti a *livelli* pari, ossia  $L_0, L_2, \dots$ , sono contenuti in  $X$ , viceversa i livelli dispari  $L_1, L_3, \dots$  sono contenuti in  $Y$ ;
- (ii) se  $(u, v) \in \hat{E}$ , allora per qualche  $i$ ,  $u \in L_{i+1}$  e  $v \in L_i$ ;
- (iii)  $\hat{G}$  è aciclico;
- (iv) i cammini aumentanti minimi relativi a  $M$  sono in corrispondenza biunivoca con i cammini di  $\hat{G}$  che iniziano nell'insieme dei vertici esposti di  $Y$  e terminano nell'insieme dei vertici esposti di  $X$ . Inoltre, tutti questi cammini hanno lunghezza  $i^*$ .

A fini implementativi aggiungiamo due vertici al grafo:  $s$  la *sorgente* e  $t$  il *pozzo*, in modo che  $s$  sia collegato a tutti i vertici esposti di  $Y$  tramite archi uscenti da  $s$  e  $t$  sia collegato a tutti i vertici esposti di  $X$  tramite archi entranti in  $t$ .

L'Algoritmo 3 rappresenta un modo per determinare un insieme di cammini massimali disgiunti sui vertici (ad eccezione dei punti esterni) da  $s$  a  $t$  in un grafo aciclico arbitrario  $H$ . Il metodo di ricerca per determinare tale insieme di cammini è standard e di tipo *depth-first*.

A seguire scriviamo la notazione necessaria per comprendere la procedura dell'algoritmo. Per ogni vertice  $u$  indichiamo con  $LIST(u)$  l'insieme contenente, in ordine arbitrario, i vertici  $v$  tali che  $(u, v)$  sia un arco di  $H$ . Inoltre, vengono utilizzate due strutture dati ausiliarie: una pila (**stack**), ovvero una struttura di tipo *last in-first out* che è inizialmente vuota, e un insieme  $B$  di vertici anch'esso inizialmente vuoto. Inoltre, introduciamo:

- *Variabili*

TOP: primo elemento della pila (STACK)

FIRST

- *Operazioni*

PUSH  $x$ : inserisci l'elemento  $x$  nella STACK

POP: elimina un elemento dalla STACK

DELETE: cancella il primo elemento dalla lista  $LIST(TOP)$

PRINT: esegui POP finché la STACK non è vuota e stampa ogni elemento prima di eseguire POP

- *Predicati*

EMPTY: STACK è vuota

NULL:  $LIST(TOP)$  è vuota

Riportiamo le seguenti osservazioni: ogni volta che l'istruzione 3.3 dell'Algoritmo 3 viene eseguita:

- la pila contiene la sequenza dei vertici di un cammino da  $s$  a  $t$ ;
- per ogni vertice  $u \notin \{s, t\}$  vale una delle seguenti
  - $u \in B$  e  $u$  è un cammino da  $s$  a  $t$  già stampato;
  - $u \in B$  e  $u$  è nella pila;
  - $u \in B$  e  $u$  non appare in alcun cammino da  $s$  a  $t$  che è disgiunto da un cammino da  $s$  a  $t$  già stampato;

---

**Algoritmo 3** ALGORITMO B

---

```
1: PUSH s
2: while STACK ≠ EMPTY do
3:   while LIST(TOP) ≠ NULL do
4:     FIRST = first element of LIST(TOP)
5:     if FIRST ∉ B then
6:       PUSH FIRST
7:       if TOP ≠ t then
8:         B ← B ∪ {TOP}
9:       else
10:        PRINT, PUSH s
11:      end if
12:    else
13:      DELETE
14:    end if
15:  end while
16: POP
17: end while
```

---

- (d)  $u \notin B$ ,  $u$  non appare nella pila e in ogni altro cammino da  $s$  a  $t$  stampato precedentemente e  $u$  non è stato cancellato da nessuna lista.

L'algoritmo termina quando  $s$  è eliminato dalla pila e dopo che la lista  $LIST(s)$  risulta vuota. Le osservazioni riportate precedentemente, implicano che alla terminazione dell'algoritmo, non esiste alcun cammino da  $s$  a  $t$  disgiunto da quelli già stampati.

Ogni blocco **while** dell'algoritmo contiene o un'operazione **POP** o **DELETE**. Dato che nessun vertice viene tolto dalla pila più di una volta (**POP**) o cancellato (**DELETE**) dalla lista più di una volta, il tempo di esecuzione dell'algoritmo è limitato da una costante. Se indichiamo con  $n$  il numero di vertici e con  $m$  il numero di archi, tale costante risulta essere  $m + n$ .

Visto che l'esecuzione dello Step 1 dell'algoritmo A richiede al più  $O(m + n)$  operazioni, l'esecuzione dell'intero algoritmo di Hopcroft e Karp richiede al più

$$O((m + n)\sqrt{s}) = O(n^{5/2})$$

passi, dove  $m, s$  sono stimati rispettivamente come  $|m| \leq n^2, s \leq n$ .

## Capitolo 2

# Problema di trasporto ottimo

In questo capitolo, riportiamo alcuni risultati riguardanti il problema di trasporto ottimo di Kantorovich [4]. Studiamo dapprima il problema definito sul politopo di Birkhoff, successivamente il problema di accoppiamento monotono e, infine, il problema regolarizzato da un termine entropico. Dunque esponiamo l'algoritmo di Sinkhorn-Knopp e i cenni di convergenza del metodo. Riportiamo inoltre, alcune sperimentazione che impiegano l'algoritmo di Sinkhorn per verificare gli enunciati del teorema di Birkhoff e il teorema di accoppiamento monotono. Infine definiamo il problema duale di Kantorovich.

### 2.1 Problema di assegnamento e rilassamento di Kantorovich

Ricordando quanto riportato nell'Osservazione 3, introduciamo più formalmente il problema di assegnamento.

Siano  $X$  e  $Y$  due insiemi di cardinalità  $n$  ed  $m$  rispettivamente. Supponiamo ora  $n = m$ . Sia  $C = (c_{i,j})$  la matrice *costo*, dove  $c_{i,j}$  è il costo di assegnamento dell'elemento  $x_i \in X$  a  $y_j \in Y$ . Si definisce *problema di assegnamento* il seguente problema di minimo

$$\min_{\sigma \in \mathcal{S}_n} \frac{1}{n} \sum_{i=1}^n C_{i,\sigma(i)}, \quad (2.1)$$

dove  $\mathcal{S}_n$  è l'insieme di permutazioni di  $n$  elementi.

Analizziamo ora un rilassamento del problema di assegnamento dovuto a Kantorovich. L'idea è di rilassare la natura deterministica del problema di assegnamento, ovvero il vincolo che la *massa* di un punto  $x_i$  sia assegnata interamente a un solo elemento  $y_{\sigma(i)}$ , per considerare, invece, un problema di

tipo probabilistico, dove la massa del punto  $x_i$  possa essere divisa e assegnata a più punti di  $Y$ . Inoltre, viene rilassata l'ipotesi che prevede  $n = m$ . Questa flessibilità è espressa dall'utilizzo di una matrice di probabilità congiunta chiamata *matrice di trasporto* (*coupling matrix*),  $P \in \mathbb{R}_+^{n \times m}$  con  $P = (p_{i,j})$ , dove l'elemento  $p_{i,j}$  descrive la quantità di massa trasportata dal punto  $x_i$  al punto  $y_j$ .

Si definisce *vettore di probabilità* di dimensione  $n$  ogni elemento appartenente al seguente insieme

$$\Sigma_n := \left\{ a \in \mathbb{R}_+^n : \sum_{i=1}^n a_i = 1 \right\}.$$

**Definizione 2.1.** Dati  $a \in \mathbb{R}_+^n$  e  $b \in \mathbb{R}_+^m$  vettori di probabilità, si definisce l'insieme

$$U(a, b) := \left\{ P \in \mathbb{R}_+^{n \times m} : P\mathbf{1}_m = a \text{ e } P^T\mathbf{1}_n = b \right\}, \quad (2.2)$$

dove

$$P\mathbf{1}_m = \left( \sum_j P_{i,j} \right)_i \in \mathbb{R}^n \quad \text{e} \quad P^T\mathbf{1}_n = \left( \sum_i P_{i,j} \right)_j \in \mathbb{R}^m.$$

In particolare, si osserva che l'insieme delle matrici  $U(a, b)$  è un politopo convesso, in quanto intersezione finita di  $n + m$  vincoli di uguaglianza, chiamato *politopo di trasporto*.

**Definizione 2.2.** Nel caso particolare in cui  $a, b = \mathbf{1}_n/n$ , il politopo  $U(\mathbf{1}_n/n, \mathbf{1}_n/n)$  prende il nome di *politopo di Birkhoff* e si indica con  $\mathcal{B}_n$ .

Dati i vettori  $a, b$  e la matrice costo  $C$  definiti sopra, si chiama *problema di trasporto ottimo di Kantorovich* il seguente problema di minimo

$$L_C(a, b) := \min_{P \in U(a, b)} \langle C, P \rangle := \sum_{i,j} c_{i,j} p_{i,j}. \quad (2.3)$$

Il problema di Kantorovich può essere visto come modello del seguente problema di allocazione delle risorse.

**Nota 6.** Supponiamo che un operatore gestisca  $n$  magazzini ed  $m$  fabbriche. Ogni magazzino contiene una considerevole quantità di materie prime che viene utilizzata nelle fabbriche per mantenere attiva la produzione.

Più formalmente, indicizziamo l'insieme dei magazzini con  $i$ , e l'insieme delle fabbriche con  $j$ . Ogni magazzino  $i$  contiene  $a_i$  unità di merce, inoltre ogni fabbrica necessita della quantità di merce  $b_j$  per essere operativa. Per trasferire le risorse dal magazzino  $i$  alla fabbrica  $j$ , l'operatore si affida a una compagnia di trasporto che richiede il costo  $c_{i,j}$  per muovere una singola unità di risorse dal luogo  $i$  al luogo  $j$ .

Assumiamo che la ditta di trasporto possenga il monopolio sul trasporto della merce, e applichi la stessa tariffa a ogni acquirente: il costo di trasportare  $a$  unità di materie prime da  $i$  a  $j$  è uguale a  $a \cdot c_{i,j}$ .

Dunque, per allocare le risorse necessarie alle fabbriche a costo minimo, l'operatore deve risolvere il problema di ottimizzazione (2.3).

Infine, osserviamo che la soluzione ottima di (2.3), quantifica per ogni coppia  $i, j$  l'ammontare di materiale  $p_{i,j}$  che deve essere trasportato dal magazzino  $i$  alla fabbrica  $j$ .

**Osservazione 7** (Problema di assegnamento come problema di Kantorovich). Data una permutazione  $\sigma \in \mathcal{S}_n$ , indichiamo con  $P_\sigma$  la corrispondente *matrice di permutazione* definita come segue

$$(P_\sigma)_{i,j} = \begin{cases} 1/n & \text{se } j = \sigma(i) \\ 0 & \text{altrimenti} \end{cases}$$

per ogni  $(i, j)$  con  $1 \leq i, j \leq n$ .

In tal caso si può verificare che vale

$$\langle C, P_\sigma \rangle = \frac{1}{n} \sum_{i=1}^n c_{i,\sigma(i)}.$$

L'uguaglianza sopra mostra che il problema di assegnamento (2.2) può essere riformulato come problema di Kantorovich (2.11) dove la matrice di trasporto  $P$  ha il vincolo di essere una matrice di permutazione. Vale a dire

$$\min_{\sigma \in \mathcal{S}_n} \frac{1}{n} \sum_{i=1}^n c_{i,\sigma(i)} = \min_{\sigma \in \mathcal{S}_n} \langle C, P_\sigma \rangle.$$

## 2.2 Matrici di permutazione e politopo di Birkhoff

L'insieme delle matrici di permutazione è incluso strettamente nel politopo di Birkhoff. Infatti, ogni matrice di permutazione soddisfa  $P\mathbf{1}_n = \mathbf{1}_n$  e  $P^T\mathbf{1}_n = \mathbf{1}_n$ , mentre  $\mathbf{1}_n\mathbf{1}_n^T/n^2$  è una matrice di trasporto ma non una matrice di permutazione. Dunque, la quantità  $\langle C, P \rangle$  è necessariamente minore quando si considerano tutte le matrici del politopo di Birkhoff anziché solo quelle di permutazione, ossia

$$L_c(\mathbf{1}_n, \mathbf{1}_n) \leq \min_{\sigma \in \mathcal{S}_n} \langle C, P_\sigma \rangle.$$

Tuttavia, quando i vettori di probabilità sono vettori di probabilità uniforme, ovvero  $a = b = \mathbf{1}_n/n$ , la proposizione seguente mostra che questi problemi risultano nello stesso ottimo, ossia esiste una matrice di permutazione  $P_\sigma$  che minimizza il problema di Kantorovich (2.3). Dunque la soluzione del problema di Kantorovich è intera quando vengono considerati problemi di assegnamento.

**Proposizione 2.3.** Se  $m = n$  e  $a = b = \mathbf{1}_n/n$ , allora esiste una soluzione ottima  $P_\sigma^*$  del problema (2.3). In particolare, la soluzione  $P_\sigma^*$  è una matrice di permutazione ed è associata a una permutazione ottima  $\sigma^* \in \mathcal{S}_n$  del problema di assegnamento (2.1).

*Dimostrazione:* Per il corollario 2.2 del teorema di Birkhoff 2.1, i vertici di  $U(\mathbf{1}_n/n, \mathbf{1}_n/n)$  corrispondono all'insieme delle matrici di permutazione. Si conclude usando il teorema fondamentale della programmazione lineare 2.4.  $\square$

**Teorema 2.1** (Birkhoff). Ogni matrice quadrata  $n$ -dimensionale bistocastica  $A = (a_{i,j})$  è una combinazione convessa di matrici di permutazione.

*Dimostrazione:* Si dimostra che per induzione sul numero di entrate non nulle di  $A$  che denotiamo con  $k(A)$ .

Dato che ogni riga di  $A$  contiene almeno un'entrata non nulla,  $k(A) \geq n$ .

- Se  $k(A) = n$ , allora  $A$  è una matrice di permutazione.
- Sia ora  $k(A) > n$ . Supponiamo che ogni matrice bistocastica  $B$  con  $k(B) \leq k(A)$  sia una combinazione convessa di matrici di permutazione. Dato che  $A$  è una matrice bistocastica [2, §4, Corollario 4.2] esiste una permutazione  $\pi \in \mathcal{S}_n$  tale che  $a_{j,\pi(j)} > 0$  dove  $1 \leq j \leq n$ . Si definiscono le quantità

$$t_0 := \min_{1 \leq j \leq n} a_{j,\pi(j)}, \quad \Pi_{i,j}^{(0)} := \begin{cases} 1 & \text{se } i = \pi(j) \\ 0 & \text{altrimenti} \end{cases}$$

- se  $t_0 = 1$ , allora  $A$  coincide con  $\Pi_{i,j}^{(0)}$ , e dunque è una matrice di permutazione;
- se  $t_0 < 1$ , allora la matrice  $B := (A - t_0 \Pi_{i,j}^{(0)}) / (1 - t_0)$  è bistocastica con  $k(B) < k(A)$ . Per ipotesi induttiva,  $B$  è una combinazione convessa di matrici di permutazione, così è  $A$ .

$\square$

**Osservazione 8.** L'insieme delle matrici di permutazione è contenuto nell'insieme dei vertici del politopo di Birkhoff, vale a dire che ogni matrice di permutazione non può essere espressa come combinazione convessa propria di matrici bistocastiche.

Infatti, mostriamo anzitutto che una combinazione convessa di matrici bistocastiche può essere ridotta a una combinazione convessa di due sole matrici bistocastiche. Ovvero mostriamo che

$$\lambda_0 A_0 + \lambda_1 A_1 + \dots + \lambda_n A_n = tA + (1 - t)B, \quad (2.4)$$

dove  $\sum_{i=0}^n \lambda_i = 1$  e inoltre  $A, B, A_i$  sono matrici bistocastiche per ogni  $1 \leq i \leq n$ . Osserviamo che

$$\lambda_0 A_0 + \lambda_1 A_1 + \dots + \lambda_n A_n = \lambda_0 A_0 + (1 - \lambda_0) \cdot \left( \sum_{i=1}^n \frac{\lambda_i}{1 - \lambda_0} A_i \right).$$

Inoltre,  $\sum_{i=1}^n \frac{\lambda_i}{1 - \lambda_0} A_i$  è una matrice bistocastica, dunque l'uguaglianza (2.4) vale scegliendo

$$t = \lambda_0, \quad A = A_0 \quad \text{e} \quad B = \sum_{i=1}^n \frac{\lambda_i}{1 - \lambda_0} A_i.$$

Dal teorema di Birkhoff e dall'osservazione precedente si ottiene il seguente.

**Corollario 2.2.** Le matrici di permutazione sono i vertici del politopo di Birkhoff.

**Osservazione 9** (Problema di massimo accoppiamento e problema Kantorovich). Si nota che il teorema di Birkhoff è un'istanza particolare del teorema di massimo accoppiamento di grafi bipartiti 1.1.

Infatti, il problema di trasporto ottimo definito sul politopo di Birkhoff coincide con il problema di assegnamento (2.1), che, a sua volta, può essere visto come problema di massimo accoppiamento di grafi bipartiti completi pesati in cui le bipartizioni hanno stessa cardinalità.

## 2.3 Caso 1-dimensionale

Nel caso particolare in cui viene considerata un'istanza del problema di trasporto definita su  $\mathbb{R}$ , ponendo alcune ipotesi sui vettori di probabilità e sulla matrice costo, la soluzione ottima del problema di trasporto ottimo di Kantorovich coincide con la soluzione del problema di assegnamento.

Più formalmente, supponiamo quanto segue:

- i campioni  $X = \{x_1, \dots, x_n\}$  e  $Y = \{y_1, \dots, y_n\}$  sono sottoinsiemi finiti di stessa cardinalità della retta reale;
- i vettori di probabilità  $r, c$  definiti rispettivamente su  $X$  e  $Y$  sono uniformi, ovvero  $r = c = \mathbf{1}_n/n$ ;
- la matrice costo  $M$  è della forma seguente

$$M := \begin{bmatrix} |x_1 - y_1|^p & \dots & |x_1 - y_n|^p \\ \vdots & \vdots & \vdots \\ |x_n - y_1|^p & \dots & |x_n - y_n|^p \end{bmatrix}$$

con  $p \geq 1$ .

Sotto queste ipotesi, il problema di trasporto ottimo, che in tal caso coincide con il problema di assegnamento, ammette una semplice soluzione: l'idea infatti, è quella di riordinare i campioni  $X, Y$  in modo che gli elementi siano crescenti, dunque accoppiare il più piccolo elemento di  $X$  al più piccolo elemento di  $Y$  e così via. Per dimostrare quanto appena detto introduciamo la seguente notazione.

Dati  $n \geq 1$  punti distinti  $(z_i)_{i=1}^n$  in  $\mathbb{R}$ , denotiamo con  $k$ -esima statistica d'ordine ( $k$ -th smallest value) e la indichiamo con  $z_{(k)}$ , il valore tale che

$$z_{(k)} = \min_i \{z_i : z_i > z_{(k-1)}\}.$$

Si osserva che le statistiche d'ordine estremo sono

$$z_{(1)} = \min_i \{z_i\}, \quad z_{(n)} = \max_i \{z_i\}.$$

L'insieme  $z_{(1)}, z_{(2)}, \dots, z_{(n)}$  viene detto *statistica d'ordine* (*order statistics*).

**Teorema 2.3** (Accoppiamento monotono). Sia  $n \geq 1$  e siano  $X = (x_i)_{i=1}^n, Y = (y_i)_{i=1}^n$  insiemi di punti distinti in  $\mathbb{R}$ . Allora esiste un'unica permutazione  $\sigma^\dagger \in \mathcal{S}_n$  tale che

$$\{(x_i, y_{\sigma^\dagger(i)})\}_{i=1}^n = \{(x_{(i)}, y_{(i)})\}_{i=1}^n,$$

Inoltre, per ogni  $p \geq 1$ ,

$$\begin{aligned} \sum_{i=1}^n |x_i - y_{\sigma^\dagger(i)}|^p &= \sum_{i=1}^n |x_{(i)} - y_{(i)}|^p \\ &= \min_{\sigma \in \mathcal{S}_n} \sum_{i=1}^n |x_i - y_{\sigma(i)}|^p \stackrel{\star}{=} n \cdot \min_{P_\sigma} \langle P_\sigma, M \rangle. \end{aligned} \quad (2.5)$$

Dove  $\star$  indica che il piano di trasporto  $\{(x_i, y_{\sigma^\dagger(i)})\}_{i=1}^n$  è ottimo per Kantorovich.

*Dimostrazione:* La prima parte dell'enunciato è banale, in quanto i punti sono supposti distinti.

Il secondo enunciato si dimostra induttivamente usando la seguente disuguaglianza: dati  $x < x', y < y'$ , allora

$$|x - y|^p + |x' - y'|^p \leq |x - y'|^p + |x' - y|^p. \quad (2.6)$$

Assumendo che  $\bar{\sigma}$  determini un minimo di (2.5), se  $\bar{\sigma}$  non accoppia  $x := x_{(1)}$  a  $y := y_{(1)}$  ma a un altro punto  $y' > y$  (e dunque anche  $y$  è accoppiato a un altro punto  $x' > x$ ), usando la (2.6) con  $x, y, x', y'$  segue che esiste un accoppiamento che abbina  $x$  con  $y$  e  $x'$  con  $y'$  e che negli altri casi è

definito come  $\bar{\sigma}$ . Ovvero esiste un accoppiamento che determina un costo minore. Dunque possiamo assumere che  $\bar{\sigma}$  accoppia  $x_{(1)}$  a  $y_{(1)}$ ; rimuovendoli e usando l'ipotesi induttiva segue la tesi.

Resta da dimostrare la disuguaglianza (2.6). Senza perdita di generalità possiamo assumere che  $x = 0$  e che  $y > 0$ , dunque risulta

$$y^p + |x' - y'|^p \leq (y')^p + |x' - y|^p,$$

per ogni  $x' > 0$ ,  $0 < y < y'$ , che è equivalente a

$$y^p - |y - x'|^p \leq (y')^p - |y' - x'|^p$$

per ogni  $x' > 0$ ,  $0 < y < y'$ , in quanto la funzione  $y \mapsto y^p - |y - x'|^p$  è crescente per ogni  $x' > 0$ . Derivando rispetto alla variabile  $y$  si ottiene

$$p \left( y^{p-1} - \operatorname{sgn}(y - x') |y - x'|^{p-1} \right),$$

che è banalmente positiva se  $y < x'$  mentre per ogni  $y > x'$  è positiva in quanto  $z \mapsto z^{p-1}$  è crescente (qui stiamo usando l'ipotesi  $p \geq 1$ ).  $\square$

**Nota 10.** Il risultato in 2.3 vale più in generale per misure di probabilità e sottoinsiemi di  $\mathbb{R}$  arbitrari, come riportato in [5, §2.2, Teorema 2.9]. Inoltre, riusciamo a esprimere il costo di tale problema di trasporto in funzione dei quantili delle funzioni di ripartizione delle misure di probabilità considerate [5, §2.2, Proposizione 2.17].

## 2.4 Regolarizzazione entropica

Nel caso generale viene introdotta una regolarizzazione alla formulazione originaria del problema di trasporto ottimo in modo che il calcolo della soluzione sia computazionalmente più accessibile.

**Definizione 2.4.** Si definisce *entropia discreta* di una matrice di trasporto  $P$ , la quantità

$$H(P) := - \sum_{i,j} p_{i,j} \log(p_{i,j}). \quad (2.7)$$

Si osserva che la funzione  $H(P)$  è *1-strettamente concava* poiché la sua Hessiana è  $\partial^2 H(P) = -\operatorname{diag}(1/p_{i,j})$  dove  $p_{i,j} \leq 1$  per ogni  $i, j$ .

Si definisce *regolarizzazione entropica* del problema di trasporto ottimo di Kantorovich l'espressione

$$L_C^\varepsilon(a, b) := \min_{P \in U(a,b)} \langle P, C \rangle - \varepsilon H(P). \quad (2.8)$$

Si osserva che la funzione obiettivo in (2.8) è  $\varepsilon$ -strettamente convessa, dunque il problema (2.8) ha soluzione unica. Inoltre, si ha la seguente

**Proposizione 2.5.** L'unica soluzione  $P_\varepsilon$  di (2.8) converge alla soluzione ottima di massima entropia tra l'insieme di tutte le soluzioni ottimali del problema di Kantorovich, ossia

$$P_\varepsilon \xrightarrow{\varepsilon \rightarrow 0} \operatorname{argmin} \{-H(P) : \langle P, C \rangle = L_C(a, b)\}, \quad (2.9)$$

quindi in particolare

$$L_C^\varepsilon(a, b) \xrightarrow{\varepsilon \rightarrow 0} L_C(a, b).$$

*Dimostrazione:* Sia  $(\varepsilon_l)_{l \in \mathbb{N}}$  una successione tale che  $\varepsilon_l \rightarrow 0$  e  $\varepsilon_l > 0$  per ogni  $l \in \mathbb{N}$ . Si denota con  $P_l$  la soluzione di (2.8) per  $\varepsilon = \varepsilon_l$ . Dato che  $U(a, b)$  è limitato, per il teorema di Bolzano-Weierstrass [9, §5.43, Teorema 34.1] possiamo estrarre una sottosuccessione convergente a una matrice  $P^*$ . Per semplicità di notazione indichiamo tale sottosuccessione come  $P_l$ . Dato che  $U(a, b)$  è chiuso,  $P^* \in U(a, b)$ .

Sia  $P$  tale che  $\langle C, P \rangle = L_C(a, b)$ . Dall'ottimalità di  $P$  e  $P_l$  per i rispettivi problemi di ottimizzazione ( $\varepsilon = 0$  per  $P$  e  $\varepsilon_l$  per  $P_l$ ), si ha

$$0 \leq \langle C, P_l \rangle - \langle C, P \rangle \leq \varepsilon_l (H(P_l) - H(P)). \quad (2.10)$$

Dato che  $H$  è continua, prendendo il limite per  $l \rightarrow +\infty$  l'espressione (2.10) mostra che  $\langle C, P^* \rangle = \langle C, P \rangle$ , dunque  $P^*$  è una soluzione ottima di (2.3). Inoltre, dividendo per  $\varepsilon_l$  la (2.10) e passando al limite si nota che  $H(P) \leq H(P^*)$  il che mostra che  $P^*$  è una soluzione di (2.9).

Dato che la funzione  $-H$  è strettamente convessa, se indichiamo con  $P_0^*$  l'unica soluzione di (2.9) si ha  $P_0^* = P^*$ .  $\square$

## 2.5 Algoritmo di Sinkhorn-Knopp

Si ha la seguente scomposizione della soluzione di (2.8).

**Proposizione 2.6.** [4, §4.2, Proposizione 4.3] La soluzione di (2.8) è unica e ha la seguente forma

$$p_{i,j} = u_i k_{i,j} v_j \quad \text{per ogni } (i, j) \text{ tale che } 1 \leq i \leq n, 1 \leq j \leq m, \quad (2.11)$$

per alcuni vettori  $(u, v) \in \mathbb{R}_+^n \times \mathbb{R}_+^m$ , dove la matrice  $K = (k_{i,j}) = (e^{-c_{i,j}})$ .

La fattorizzazione della soluzione ottimale esibita in (2.11) può essere riscritta nella forma

$$P = \operatorname{diag}(u) K \operatorname{diag}(v). \quad (2.12)$$

L'espressione (2.12) giustifica l'utilizzo dell'algoritmo di Sinkhorn-Knopp per il calcolo di  $P$ .

Infatti, data una matrice  $A$  di dimensione  $n \times m$  con entrate strettamente positive, lo scopo dell'algoritmo di Sinkhorn-Knopp è di generare una sequenza di matrici diagonali

$$(D^{(0)}, E^{(0)}), (D^{(1)}, E^{(1)}), \dots, (D^{(k)}, E^{(k)}), \dots$$

con  $\dim(D^{(k)}) = n$  e  $\dim(E^{(k)}) = m$  per ogni  $k \geq 0$ , tale che il prodotto

$$D^{(k)} D^{(k-1)} \dots D^{(0)} A E^{(0)} \dots E^{(k-1)} E^{(k)}$$

converga a una matrice  $B$  tale che se  $B\mathbf{1}_m = p$  e  $\mathbf{1}_n B^T = q$ , dove  $p := (p_i)_{i \leq n}$  e  $q := (q_j)_{j \leq m}$ , allora  $p_1 + \dots + p_n = q_1 + \dots + q_m$ . Ossia  $B$  si può scrivere nella forma

$$B = \text{diag}(r) A \text{diag}(c).$$

Per descrivere l'algoritmo più formalmente, introduciamo l'operatore

$$\begin{aligned} \mathcal{D}: \mathbb{R}^n &\longrightarrow \mathbb{R}^{n \times n} \\ x &\longmapsto \text{diag}(x). \end{aligned}$$

Inizializziamo

$$D^{(0)} = E^{(0)} = I$$

e definiamo

$$\begin{aligned} r^{(k)} &= D^{(k-1)} A E^{(k)} \mathbf{1}_m \\ (c^{(k)})^T &= \mathbf{1}_n^T D^{(k-1)} A E^{(k-1)}, \end{aligned} \tag{2.13}$$

dove

$$D_k = \mathcal{D}(p_i/r_i^{(k)}), \quad E_k = \mathcal{D}(q_j/c_j^{(k)}).$$

Determiniamo ora una forma più compatta per la relazione iterativa (2.13).

Le identità  $B\mathbf{1}_m = p$  e  $\mathbf{1}_n^T B = q$  danno le relazioni

$$\begin{aligned} \text{diag}(r) A \text{diag}(c) \mathbf{1}_m &= p \\ \text{diag}(c)^T A \text{diag}(r) \mathbf{1}_n &= q. \end{aligned} \tag{2.14}$$

Le equazioni (2.14) possono essere semplificate ulteriormente. Infatti, dato che  $\text{diag}(c) \mathbf{1}_m = c$  e la moltiplicazione di  $\text{diag}(r)$  per  $Kc$ , rispettivamente  $\text{diag}(r) \mathbf{1}_n = r$  e di  $\text{diag}(c)$  e  $K^T r$  è

$$r \odot (Kc) = p \quad \text{e} \quad c \odot (K^T r) = q, \tag{2.15}$$

dove  $\odot$  corrisponde alla moltiplicazione elemento per elemento di vettori.

Un modo intuitivo per manipolare le equazioni (2.15) consiste nel risolverle iterativamente, modificando prima  $r$  in modo che soddisfi la prima

equazione di (2.15) e successivamente  $c$  in modo che soddisfi la seconda equazione in (2.15).

Questi due aggiornamenti definiscono l'iterazione dell'*algoritmo di Sinkhorn*

$$r^{(k+1)} := \frac{P}{Kc^{(k)}} \quad \text{e} \quad c^{(k+1)} := \frac{q}{K^T r^{(k+1)}}, \quad (2.16)$$

dove l'operatore divisione utilizzato in (2.16) agisce elemento per elemento, e il vettore iniziale  $c^{(0)}$  è pari a  $\mathbf{1}_m$ .

Di seguito verrà riportato un cenno al procedimento da seguire per dimostrare l'algoritmo di Sinkhorn-Knopp.

### 2.5.1 Cenni sulla convergenza dell'algoritmo di Sinkhorn-Knopp

Introduciamo la seguente notazione

- $\mathbb{R}_+^n := \{(x_1, \dots, x_n) : x_i \geq 0, 1 \leq i \leq n\}$ ;
- sia  $X \subset \mathbb{R}_+^n$ . Indichiamo con  $\mathbb{P}(X)$  lo spazio *proiettivo* di  $X$ ;
- $E_A = \{B : A \sim B, b_{i,j} > 0\}$ , dove le matrici  $A \sim B$  sono equivalenti se esistono due matrici diagonali  $X, Y$  tali che  $A = XBY$ .

Di seguito, vengono riportate schematicamente le idee principali usate nella dimostrazione della convergenza dell'algoritmo di Sinkhorn-Knopp.

1. Si introduce la metrica di Hilbert sullo spazio  $\mathbb{R}_+^n$  [7, §2.1, Definizione 2.2] che verrà utilizzata per definire la metrica  $\mu$  sullo spazio  $E_A$  [7, §2.1, Definizione 2.6]. Successivamente, si mostra che lo spazio  $(\mathbb{P}(E_A), \mu)$  è completo [7, §2.1, Definizione 2.8].
2. Si dimostra il teorema di Birkhoff-Hopf [7, §2.2, Teorema 2.18], il quale fornisce un bound sull'iterazione definita dall'algoritmo di Sinkhorn-Knopp.
3. Dai punti 1 e 2 segue la convergenza dell'algoritmo di Sinkhorn [7, §2.3.1, Proposizione 2.25.].

**Nota 11.** [4, §4.2, Remark 4.6] Eseguendo un'analisi della complessità ed assumendo per semplicità  $n = m$ , è stato mostrato che definendo

$$\varepsilon := \frac{4 \log(n)}{\tau},$$

sono necessarie  $O\left(\|C\|_\infty^3 \log(n) \tau^{-3}\right)$  iterazioni dell'algoritmo di Sinkhorn-Knopp per assicurare la computazione di una matrice di trasporto  $\hat{P} \in U(a, b)$  tale che

$$\langle \hat{P}, C \rangle \leq L_C(a, b) + \tau.$$

Dunque, l'algoritmo computa una soluzione  $\tau$ -approssimata del problema di trasporto ottimo originario in  $O(n^2 \log(n) \tau^{-3})$  operazioni.

## 2.6 Sperimentazioni numeriche

In questa sezione, riportiamo i risultati di alcune sperimentazioni che mostrano l'impiego dell'algoritmo di Sinkhorn-Knopp 4 nel verificare le tesi del teorema di Birkhoff 2.1 e del teorema di accoppiamento monotono 2.3.

---

### Algoritmo 4 ALGORITMO DI SINKHORN-KNOPP

---

```

procedure SINKHORN-KNOPP( $r, c, M, \lambda$ )
   $I = (r > 0); r = r(I); M = M(I, :); K = \exp(-\lambda M);$ 
   $u = \text{ones}(\text{length}(r), N) / \text{length}(r);$ 
   $\tilde{K} = \text{diag}(1./r)K;$ 
  while  $u$  cambia o qualsiasi altro ragionevole criterio di stop do
     $u = 1./(\tilde{K}(c./(K'u)));$ 
  end while
   $v = c./(K'u);$ 
   $d = \text{sum}(u.*(K.*M)v);$ 
end procedure

```

---

### 2.6.1 Sperimentazioni relative al teorema di Birkhoff

Consideriamo un insieme di punti  $X$  di cardinalità  $d = 10$  campionato sulla circonferenza unitaria (Figura 2.1). Al campione  $X$  è associato un vettore di probabilità uniforme  $d$ -dimensionale, ovvero definito come  $\mathbf{1}_d/d$ . La matrice dei costi  $M$  viene posta uguale a

$$M := \begin{bmatrix} * & |x_1 - x_2|^p & \dots & |x_1 - x_n|^p \\ |x_2 - x_1|^p & * & & \vdots \\ \vdots & & \ddots & |x_{n-1} - x_n|^p \\ |x_n - x_1|^p & \dots & |x_n - x_{n-1}|^p & * \end{bmatrix},$$

dove gli elementi  $*$  sulla diagonale sono molto grandi, in modo da costringere i punti a muoversi.

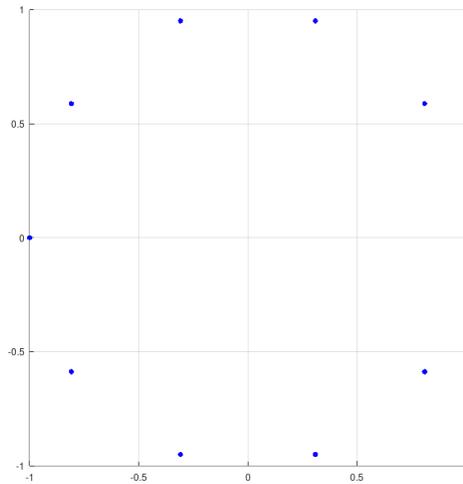
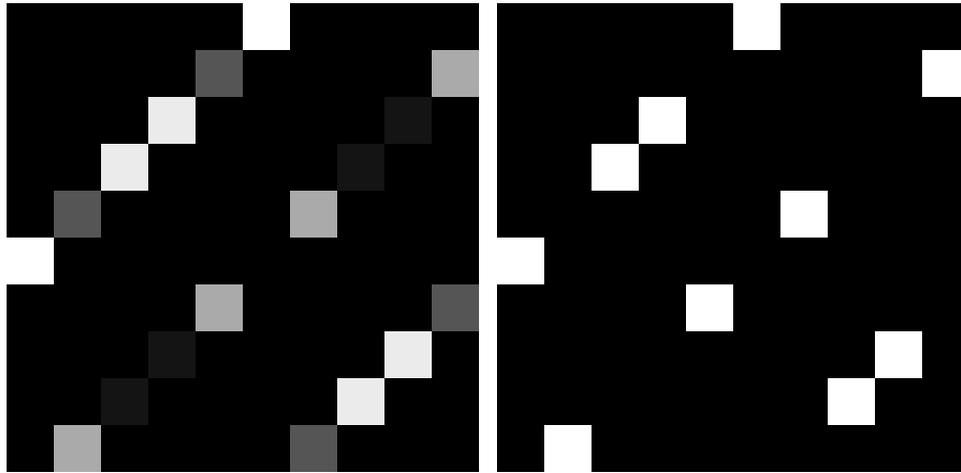


Figura 2.1: Punti equidistanti sulla circonferenza unitaria

Ad ogni sperimentazione variano i valori  $p$  e  $\lambda$ , dove  $p$  è il parametro relativo alla matrice  $M$  e  $\lambda := 1/\varepsilon$  è il parametro relativo alla regolarizzazione del problema di Kantorovich.



(a)  $p = 0.2, \lambda = 10$

(b)  $p = 0.2, \lambda = 100$

Figura 2.2

In Figura 2.2 sono riportate le matrici di trasporto determinate dall'algoritmo nel caso in cui  $p = 0.2$ . Notiamo che per  $\lambda = 100$ , il problema regolarizzato si avvicina al problema originale, e risulta in una matrice di permutazione come riportato nell'enunciato del teorema di Birkhoff 2.1. Nel caso in cui  $p = 2$ , l'algoritmo di Sinkhorn-Knopp restituisce in output la matrice mostrata in Figura 2.4a. Si osserva che in questo caso la soluzione ottima non è unica, infatti le matrici di trasporto in Figura 2.4c e 2.4b

$$\begin{bmatrix} 0 & 1/2 & 0 & \cdots & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 & & 0 \\ 0 & 1/2 & 0 & 1/2 & & \vdots \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & & & 1/2 & 0 & 1/2 \\ 1/2 & 0 & \cdots & 0 & 1/2 & 0 \end{bmatrix}$$

(a)

$$\begin{bmatrix} 0 & 2/3 & 0 & \cdots & 0 & 1/3 \\ 1/3 & 0 & 2/3 & 0 & & 0 \\ 0 & 1/3 & 0 & 2/3 & & \vdots \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & & & 1/3 & 0 & 2/3 \\ 2/3 & 0 & \cdots & 0 & 1/3 & 0 \end{bmatrix}$$

(b)

$$\begin{bmatrix} 0 & 1 & 0 & & \cdots & 0 \\ & \ddots & 1 & 0 & & \vdots \\ & & 0 & 1 & & \\ \vdots & & & & \ddots & \ddots & 0 \\ 0 & & & & & \ddots & 1 \\ 1 & 0 & \cdots & & & & 0 \end{bmatrix}$$

(c)

Figura 2.3: Matrici relative alle immagini della Figura 2.4.

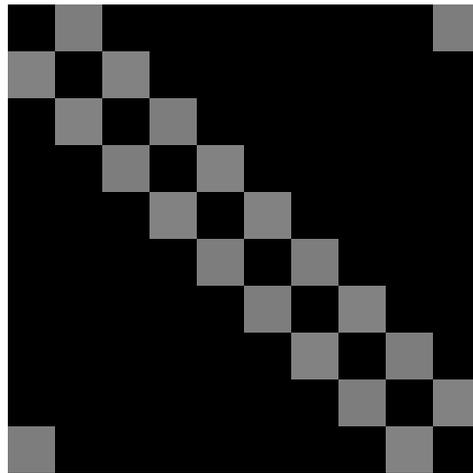
sono ottime. Tuttavia l'algoritmo restituisce la soluzione con maggiore entropia (Proposizione 2.5), che si può verificare essere quella in Figura 2.4a. Le immagini mostrate in Figura 2.4 corrispondono rispettivamente alle matrici (2.3a), (2.3b), (2.3c). Calcoliamo dunque l'entropia della matrici in Figura 2.3 usando la formula (2.4).

(a)  $H(P^{(a)}) = -10 \cdot \log(1/2) \sim 6.93;$

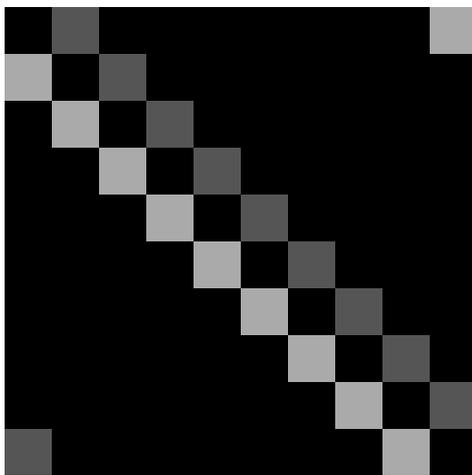
(b)  $H(P^{(b)}) = -10 \cdot (1/3 \log(1/3) + 2/3 \log(2/3)) \sim 6.36;$

(c)  $H(P^{(c)}) = 0.$

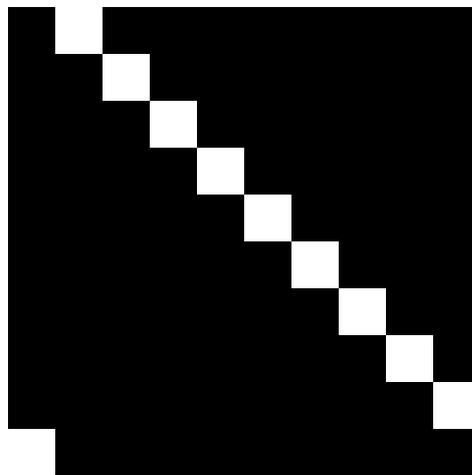
Consideriamo ora un insieme di punti  $X$  di cardinalità  $d = 10$  campionato sulla circonferenza unitaria (Figura 2.5a) in modo che i punti risultino vicini a coppie. Il vettore di probabilità associato a  $X$  e la matrice dei costi  $M$  sono gli stessi riportati nella sperimentazione precedente. In tal caso, la soluzione mostrata in Figura 2.5b data in output dall'algoritmo di Sinkhorn-Knopp risulta coincidere con la caratterizzazione ottima fornita dal teorema di Birkhoff 2.1.



(a)  $p = 2, \lambda = 100$



(b)



(c)

Figura 2.4

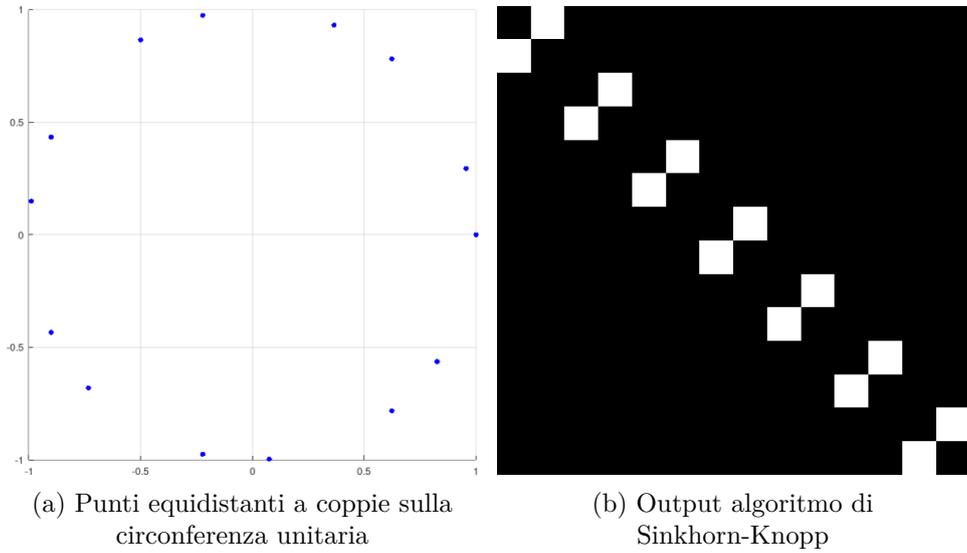


Figura 2.5

### 2.6.2 Sperimentazioni relative al teorema dell'accoppiamento monotono

Consideriamo gli insiemi di punti  $X, Y$  interi di cardinalità  $d = 10$  campionati sull'intervallo  $[1, 21]$  come mostrato in Figura 2.6.

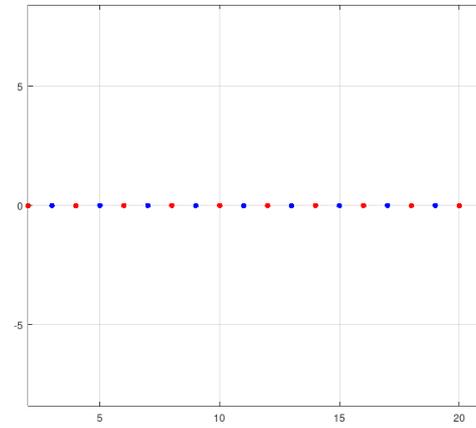


Figura 2.6

Ai campioni sono associati due vettori di probabilità uniforme  $d$ -dimensionale, ovvero definiti come  $\mathbf{1}_d/d$ . La matrice dei costi  $M$  viene posta uguale a

$$M := \begin{bmatrix} |x_1 - y_1|^p & \cdots & |x_1 - y_n|^p \\ \vdots & \ddots & \vdots \\ |x_n - y_1|^p & \cdots & |x_n - y_n|^p \end{bmatrix}.$$

Ad ogni sperimentazione variamo il valore  $p = 1, \dots, 10$  del parametro relativo alla matrice  $M$ . Come riportato nel Teorema 2.3 si ha la seguente matrice ottima di trasporto.

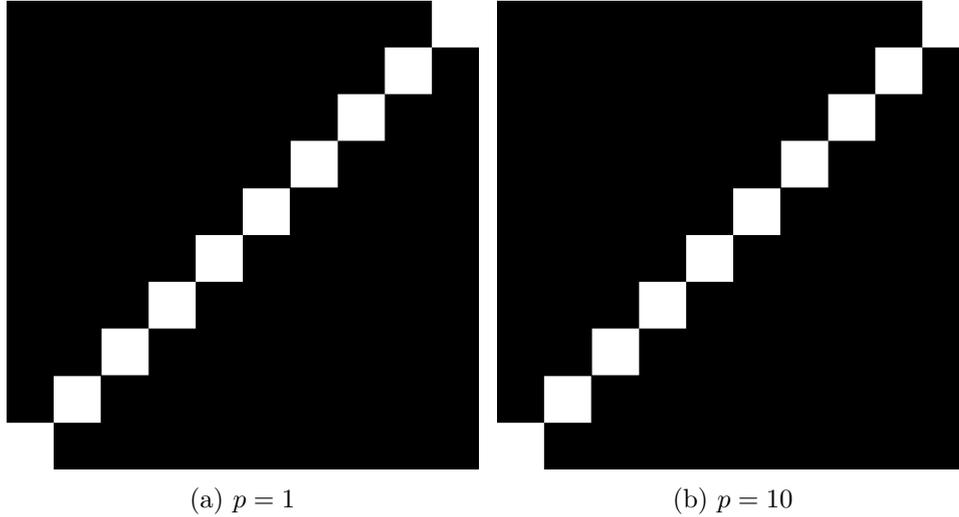


Figura 2.7

## 2.7 Problema duale

Il problema di Kantorovich (2.3) viene naturalmente con il suo problema duale. Per il teorema fondamentale della dualità forte 2.7 gli ottimi di questi due problemi coincidono.

**Proposizione 2.7.** Il problema di Kantorovich (2.3) ammette il problema duale

$$L_C(a, b) = \max_{(f, g) \in R(C)} \langle f, a \rangle + \langle g, b \rangle, \quad (2.17)$$

dove l'insieme dei valori ammissibili duali associati alla matrice costo  $C$  è

$$R(C) := \{(f, g) \in \mathbb{R}^n \times \mathbb{R}^m : f \oplus g \leq C\}. \quad (2.18)$$

Dove l'operatore  $\oplus$  trasforma i vettori  $f$  e  $g$  in una matrice di  $\mathbb{R}^{n \times m}$  che ha per entrate tutte le possibili somme degli elementi di  $f$  e  $g$ . Vale a dire

$$f \oplus g := f \cdot \mathbf{1}_m^T + \mathbf{1}_n \cdot g^T.$$

Le variabili duali  $f, g$  vengono definite *potenziali di Kantorovich*.

*Dimostrazione:* Per quanto riportato nella tabella 2.1, l'espressione

$$\max_{(f, g) \in R(C)} \langle f, a \rangle + \langle g, b \rangle$$

rappresenta il problema duale del problema di Kantorovich. Dunque dal teorema della dualità forte 2.7 segue la tesi.  $\square$

**Nota 12.** La definizione del problema duale può essere formulata anche nel caso del problema di trasporto ottimo regolarizzato (2.8), come riportato in [6, §2, Proposizione 2.1].

**Nota 13.** Completiamo l'esempio riportato nella Nota 6 aggiungendo un'interpretazione al problema duale. Ricordiamo che l'operatore desidera movimentare, a costo minimo, una certa quantità di risorse dai magazzini alle fabbriche. L'operatore può farlo risolvendo l'Equazione (2.3), seguendo le istruzioni riportate nella piano di trasporto  $P^*$ , e pagando  $\langle P^*, C \rangle$  alla compagnia di trasporto.

*Appalto a terzi.* Supponiamo che l'operatore non abbia le possibilità computazionali per risolvere il problema in (2.3). Dunque, decide di affidarsi a un fornitore esterno. Il fornitore, decide di suddividere l'incarico in due fasi: la prima consiste nel ritirare la merce dai magazzini e la seconda nell'inviare tale merce alle fabbriche. Alla prima fase applica il prezzo  $f_i$  per raccogliere le unità di merce dal magazzino  $i$  (senza contare dove la materia prima verrà inviata), e un prezzo  $g_j$  per spedire un'unità di materia prima alla fabbrica  $j$  (senza contare da quale magazzino deriva).

In totale, dato che ci sono esattamente  $a_i$  unità nel magazzino  $i$  e  $b_j$  necessarie alla fabbrica  $j$ , il prezzo totale richiesto dal fornitore per risolvere il problema logistico dell'operatore è il seguente

$$\langle f, a \rangle + \langle g, b \rangle.$$

*Decisione dei prezzi.* Il sistema di prezzi usato dal fornitore può risultare piuttosto sconveniente: infatti, se il fornitore applica un vettore prezzo  $f$  per i magazzini e un vettore prezzo  $g$  per le fabbriche, allora il costo totale non cambia se simultaneamente vengono diminuite le entrate in  $f$  e aumentate le entrate in  $g$  della stessa quantità, dato che la quantità totale di risorse in tutti i magazzini è uguale a quella spedita alle fabbriche. In altre parole, il fornitore può dare l'illusione di proporre un contratto vantaggioso semplicemente diminuendo il costo di raccolta delle risorse e aumentando quello di spedizione. Sapendo ciò, il fornitore, spera di aumentare la sua rendita alzando quanto possibile sia  $f$  che  $g$ .

*Controllo prezzi.* In assenza di un fornitore concorrente, l'operatore deve pensare a un modo veloce per controllare se la tariffa richiesta dal fornitore è equa oppure no. Ricordiamo che  $f_i$  è il prezzo richiesto per prelevare l'unità  $i$  e  $g_j$  per consegnarne una a  $j$ . Vista la funzione prezzo del venditore, osserviamo che trasferire un'unità di materia prima da  $i$  a  $j$  costa esattamente  $f_i + g_j$ . Ma l'operatore sa anche che il costo di trasporto di un'unità da  $i$  a  $j$  è esattamente il costo richiesto dalla compagnia di trasporto  $c_{i,j}$ .

Di conseguenza, se per ogni coppia  $i, j$  il prezzo totale  $f_i + g_j$  è strettamente maggiore di  $c_{i,j}$  il fornitore sta richiedendo più di un prezzo equo per il trasporto, e il venditore dovrebbe rifiutare l'offerta.

*Prezzi ottimo come problema duale.* È dunque nell'interesse dell'operatore controllare se per ogni coppia  $i, j$  i prezzi offerti dal fornitore verificano  $f_i + g_j \leq c_{i,j}$ . Supponiamo che l'operatore controlli che i prezzi proposti dal fornitore verifichino queste  $n \times m$  disuguaglianze, l'operatore può concludere che la proposta del venditore è conveniente? La risposta è affermativa: con un semplice calcolo l'operatore può concludere che è nel suo interesse accettare l'offerta del venditore, infatti, vale la seguente catena di disuguaglianze

$$\sum_{i,j} p_{i,j} c_{i,j} \geq \sum_{i,j} p_{i,j} (f_i + g_j) = \left( \sum_i f_i \sum_j p_{i,j} \right) + \left( \sum_i g_j \sum_j p_{i,j} \right) = \langle f, a \rangle + \langle g, b \rangle$$

e dunque, ogni tentativo di eseguire l'operazione senza l'appoggio di un venditore esterno risulta sconveniente.

Sapendo ciò, il fornitore può scegliere i prezzi  $f, g$  in modo da massimizzare  $\langle f, a \rangle + \langle g, b \rangle$ , soddisfacendo però, le disuguaglianze  $f_i + g_j \leq c_{i,j}$  per ogni coppia  $i, j$ , in modo che la sua offerta venga accettata dall'operatore. Dunque il problema di ottimizzazione che deve risolvere il fornitore si traduce nel problema (2.17).

# Appendice

## Richiami di programmazione lineare

### Geometria della PL

Un semispazio chiuso in  $\mathbb{R}^n$  può essere descritto algebricamente come l'insieme delle soluzioni di una disequazione lineare in  $n$  variabili

$$\alpha^T x \leq \beta, \quad \alpha \in \mathbb{R}^n, \beta \in \mathbb{R}.$$

Un *poliedro* di  $\mathbb{R}^n$  è l'intersezione di un numero finito di semispazi chiusi di  $\mathbb{R}^n$ . Ogni poliedro  $P$  è quindi rappresentabile come

$$P = \{x \in \mathbb{R}^n : Ax \leq b\},$$

dove  $A \in \mathbb{R}^{m \times n}$  e  $b \in \mathbb{R}^m$ . Si osserva che  $P$  è un insieme convesso, in quanto intersezione finita di semispazi chiusi, che sono insiemi convessi.

**Definizione 2.8.** Un punto  $x \in \mathbb{R}^n$  si dice *combinazione convessa* di  $x^1, \dots, x^m \in \mathbb{R}^n$  se esistono dei coefficienti  $\lambda_1, \dots, \lambda_m$  tali che

$$x = \sum_{i=1}^m \lambda_i x^i, \quad \lambda_i \in [0, 1], \quad 1 \leq i \leq m, \quad \sum_{i=1}^m \lambda_i = 1.$$

Le combinazione convessa è *propria* se  $0 < \lambda_i < 1$  per ogni  $1 \leq i \leq m$ .

Inoltre, si definisce *inviluppo convesso* di un insieme  $K$ , indicato con  $\text{conv}(K)$ , l'insieme di tutte le possibili combinazioni convesse di elementi di  $K$ .

**Definizione 2.9.** Un punto  $x \in \mathbb{R}^n$  si dice *combinazione conica* di  $x^1, \dots, x^m \in \mathbb{R}^n$  se esistono dei coefficienti  $\lambda_1, \dots, \lambda_m$  tali che

$$x = \sum_{i=1}^m \lambda_i x^i, \quad \lambda_i \geq 0, \quad 1 \leq i \leq m.$$

Le combinazione conica è *propria* se  $\lambda_i > 0$  per ogni  $1 \leq i \leq m$ .

Inoltre, si definisce *inviluppo conico* di un insieme  $K$ , indicato con  $\text{cono}(K)$ , l'insieme di tutte le possibili combinazioni coniche di elementi di  $K$ .

**Definizione 2.10.** Un *vertice* di un poliedro è un punto dello stesso che non può essere espresso come combinazione convessa propria di altri punti del poliedro.

**Definizione 2.11.** Una *direzione di recessione* per un poliedro  $P$  è un vettore  $d$  tale che

$$x + \lambda d \in P, \quad \forall x \in P, \forall \lambda \geq 0.$$

L'insieme delle direzioni di recessione di  $P$  si indica con  $\text{rec}(P)$ .

Una *direzione di linealità* per un poliedro  $P$  è un vettore  $d$  tale che

$$d \in \text{rec}(P), \quad -d \in \text{rec}(P).$$

L'insieme delle direzioni di linealità di  $P$  si indica con  $\text{lineal}(P)$ .

### Teorema fondamentale della PL

Un *problema di programmazione lineare in forma standard* è un problema della forma

$$\begin{aligned} \max \quad & c^T x & (\mathcal{P}) \\ & Ax \leq b \end{aligned}$$

dove  $x \in \mathbb{R}^n$  è il vettore incognito,  $c \in \mathbb{R}^n$  vettore *costo*,  $b \in \mathbb{R}^m$ ,  $A \in \mathbb{R}^{m \times n}$  matrice dei vincoli, e la funzione  $c^T x$  è detta *funzione obiettivo*. L'insieme  $\Omega := \{x \in \mathbb{R}^n : Ax \leq b\}$  è chiamata regione ammissibile. Il valore  $z(P) = \max \{c(x) : x \in \Omega\}$  si dice *valore ottimo* di  $(\mathcal{P})$ . Un elemento  $x^* \in \Omega$  tale che  $c(x^*) = z(P)$  si dice *soluzione ottima* di  $(\mathcal{P})$ .

**Teorema 2.4** (Fondamentale della PL). Sia  $P = \{x \in \mathbb{R}^n : Ax \leq b\}$  un poliedro non vuoto tale che  $\text{lineal}(P) = \{0\}$ . Siano  $v^1, \dots, v^m$  i vertici di  $P$ , e siano  $e^1, \dots, e^p$  tali che

$$P = \text{conv}(\{v^1, \dots, v^m\}) + \text{cono}(\{e^1, \dots, e^p\}).$$

Si consideri il problema di PL  $(\mathcal{P})$ . Allora  $\mathcal{P}$  ha ottimo finito se e solo se  $c^T e^j \leq 0$ , con  $j = 1, \dots, p$ . In tal caso esiste  $k \in \{1, \dots, m\}$  tale che il vertice  $v^k$  è una soluzione ottima di  $(\mathcal{P})$ .

*Dimostrazione:* Il problema  $(\mathcal{P})$  può essere riformulato come

$$\max \sum_{i=1}^m \lambda_i (c^T v^i) + \sum_{j=1}^p \gamma_j (c^T e^j),$$

dove  $\sum_{i=1}^m \lambda_i = 1$  con  $\lambda_i \geq 0$ ,  $\gamma_j \geq 0$  e  $1 \leq i \leq m$ ,  $1 \leq j \leq p$ .

( $\Rightarrow$ ) Se esistesse un indice  $j \in \{1, \dots, p\}$  tale che  $c^T e^j > 0$ , facendo crescere  $\gamma_j$  a infinito, il valore ottimo tenderebbe a infinito.

( $\Leftarrow$ ) Sia  $x$  una qualsiasi soluzione ammissibile, ovvero  $x \in P$  :

$$c^T x = \sum_{i=1}^m \lambda_i (c^T v^i) + \sum_{j=1}^p \gamma_j \overbrace{(c^T e^j)}^{\leq 0}$$

$$\sum_{i=1}^m \lambda_i (c^T v^i) \leq c^T v^k \left( \underbrace{\sum_{i=1}^m \lambda_i}_{=1} \right) = c^T v^k,$$

dove  $v^k$  è un vertice tale che

$$c^T = \max \{ c^T v^i : i = 1, \dots, m \}.$$

Ovvero esiste un vertice che è soluzione ottima di ( $\mathcal{P}$ ).

□

### Problema duale

Al problema *primale* ( $\mathcal{P}$ ) in forma standard associamo il seguente problema di PL

$$\begin{aligned} \min \quad & y^T b & (\mathcal{D}) \\ & y^T A = c \\ & y \geq 0 \end{aligned}$$

che definiamo *problema duale in forma standard*. Poiché ogni problema di PL si può trasformare in un problema standard di tipo ( $\mathcal{P}$ ) [8, §2.2], ogni problema di PL ha un suo problema duale.

La Tabella 2.1 riassume la corrispondenza fra problema primale e duale. In particolare, fornisce una procedura per trasformare il problema primale nel suo duale e viceversa.

**Teorema 2.5** (Dualità debole). Se i poliedri  $P$  e  $D$  sono non vuoti, allora

$$c^T x \leq y^T b \quad \forall x \in P, \forall y \in D.$$

*Dimostrazione:* Per ogni  $x \in P$  e  $y \in D$  si ha

$$c^T x = y^T A x \leq y^T b.$$

□

Primale	Duale
$c$	$b$
$b$	$c$
$x$	$y$
$A_i x \leq b_i$	$y_i \geq 0$
$A_i x \geq b_i$	$y_i \leq 0$
$A_i x = b_i$	$y_i \leq 0$
$x_j \geq 0$	$y^T A^j \leq c_j$
$x_j \leq 0$	$y^T A^j \geq c_j$
$x_j \leq 0$	$y^T A^j = c_j$

Tabella 2.1

**Teorema 2.6** (Farkas). [8, §2.3, Teorema 2.3.3] Data una matrice  $A \in \mathbb{R}^{m \times n}$  e un vettore  $c \in \mathbb{R}^n$ , i due sistemi

$$\begin{cases} y^T A = c^T \\ y \geq 0 \end{cases} \quad (2.19)$$

e

$$\begin{cases} Ad \leq 0 \\ c^T d < 0 \end{cases} \quad (2.20)$$

sono mutualmente esclusivi, cioè (2.19) ammette soluzioni se e solo se (2.20) è impossibile.

**Teorema 2.7** (Dualità forte). Se i poliedri  $P$  e  $D$  sono non vuoti, allora

$$-\infty < z(P) = z(D) < +\infty. \quad (2.21)$$

*Dimostrazione:* Dal teorema della dualità debole segue che il problema  $(P)$  ha valore ottimo finito e quindi, per il teorema fondamentale della PL, esiste una soluzione ottima  $\bar{x}$ . Consideriamo l'insieme dei vincoli attivi in  $\bar{x}$ , ovvero i vincoli soddisfatti in forma di uguaglianza:

$$\mathcal{A}(\bar{x}) = \{i : A_i \bar{x} = b_i\}$$

ed indichiamo con  $A_{\mathcal{A}(\bar{x})}$  la sottomatrice di  $A$  ottenuta scegliendo le righe  $A_i$  con  $i \in \mathcal{A}(\bar{x})$ . Consideriamo un vettore  $d \in \mathbb{R}^n$  tale che  $A_{\mathcal{A}(\bar{x})} d \leq 0$ . Se per ogni  $i \notin \mathcal{A}(\bar{x})$  si ha che  $A_i d \leq 0$ , cioè  $d$  è una direzione di recessione per il poliedro  $P$ , allora

$$\bar{x} + \lambda d \in P \quad \forall \lambda \geq 0.$$

Se invece, esiste almeno un indice  $i \notin \mathcal{A}(\bar{x})$  tale che  $A_i d > 0$ , allora

$$\bar{x} + \lambda d \in P \quad \forall \lambda \in [0, \bar{\lambda}],$$

dove

$$\bar{\lambda} = \min \left\{ \frac{b_i - A_i \bar{x}}{A_i d}, \quad i \notin \mathcal{A}(\bar{x}), \quad A_i d > 0 \right\} > 0.$$

In entrambi i casi, esistono punti ammissibili del tipo  $\bar{x} + \lambda d$  con  $\lambda > 0$ .

Poiché  $\bar{x}$  è una soluzione ottima di  $(\mathcal{P})$ , la direzione  $d$  non può essere una *direzione di crescita* per la funzione obiettivo primale, cioè deve valere  $c^T d \leq 0$ . Quindi abbiamo dimostrato che se  $A_{\mathcal{A}(\bar{x})} d \leq 0$ , allora  $c^T d \leq 0$ . Ossia il sistema

$$\begin{cases} A_{\mathcal{A}(\bar{x})} d \leq 0 \\ c^T d > 0 \end{cases}$$

è impossibile. Dunque, per il teorema di Farkas, il sistema

$$\begin{cases} u^T A_{\mathcal{A}(\bar{x})} u = c^T \\ u \geq 0 \end{cases}$$

ammette almeno una soluzione  $\bar{u}$ . Definiamo ora il vettore  $\bar{y}$  come

$$\bar{y}_i := \begin{cases} \bar{u}_i & \text{se } i \in \mathcal{A}(\bar{x}) \\ 0 & \text{se } i \notin \mathcal{A}(\bar{x}) \end{cases}.$$

Allora il vettore  $\bar{y}$  appartiene al poliedro duale  $D$ , infatti

$$\bar{y}^T A = \bar{u}^T A_{\mathcal{A}(\bar{x})} = c^T, \quad \bar{y} \geq 0.$$

Inoltre il valore della funzione obiettivo duale in  $\bar{y}$  coincide con il valore della funzione obiettivo primale in  $\bar{x}$ , infatti

$$\bar{y}^T b = \bar{u}^T b_{\mathcal{A}(\bar{x})} = \bar{u}^T A_{\mathcal{A}(\bar{x})} \bar{x} = c^T \bar{x}.$$

Quindi per il teorema della dualità debole  $\bar{y}$  è una soluzione ottima di  $(\mathcal{D})$  ed i valori ottimi di  $(\mathcal{P})$  e  $(\mathcal{D})$  coincidono.  $\square$

# Bibliografia

- [1] L. Lovász, M.D. Plummer. *Matching Theory*. North Holland Mathematics Studies, 1986.
- [2] T. Ando. *Majorization, Doubly Stochastic Matrices, and Comparison of Eigenvalues*. Linear Algebra and Its Applications, 118:163–248, 1989.
- [3] J. E. Hopcroft, R. M. Karp. *An algorithm for maximum matching in bipartite graphs*. SIAM J. Comput. 2(4):225-231, 1973.
- [4] G. Peyré, M. Cuturi. *Computational Optimal Transport*. arXiv:1803.00567, 2020.
- [5] F. Santambrogio. *Optimal Transport for Applied Mathematicians*. Birkhäuser Basel, 2015.
- [6] A. Genevay, G. Peyré, M. Cuturi, F. Bach. *Stochastic Optimization for Large-Scale Optimal Transport*. arXiv:1605.08527v1, 2016.
- [7] V. D’Onofrio. *Sinkhorn’s algorithm for optimal transport problem*. Tesi di laurea triennale, a.a. 2017/2018.
- [8] M. Pappalardo, M. Passacantando. *Ricerca Operativa*. Pisa University Press, 2012.
- [9] G. Prodi. *Analisi Matematica*. Bollati Boringhieri, 1970.