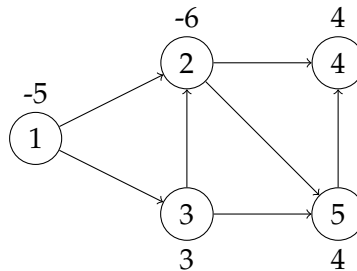


1 Lezione del 05-11-24

1.1 Ottimizzazione sui grafi

Veniamo adesso all'**ottimizzazione su grafi**, cioè la massimizzazione o minimizzazione del flusso su grafi definiti come insiemi $G = (N, A)$, di $\text{card}(N) = n$ nodi e $\text{card}(A) = m$ archi, dove $A \subseteq N \times N$ (dalla commutatività del prodotto cartesiano si parla di grafi **orientati**). Si potrà avere ad esempio:



Etichettiamo con il **bilancio** di flusso ogni nodo, adottando la convenzione dei **segni positivi** per i *pozzi* (i nodi che assorbono flusso) e **segni negativi** per le *sorgenti* (i nodi che erogano flusso). A questo punto, il vettore soluzione x_{ij} rappresenterà le unità di flusso su ogni arco (i e j sono indici rispettivamente dei nodi di partenza e arrivo). Come sempre, il vettore soluzione è ordinato **lessicograficamente**.

Quello che vogliamo fare è quindi esprimere i **vincoli** a cui è sottoposto flusso sugli archi del grafo. Conviene quindi guardare sempre al bilancio in termini di flusso di ogni nodo, prendendo gli entranti (cioè i nodi che *approvigionano* il nodo) come positivi e gli uscenti (cioè i nodi che si *ri forniscono* dal nodo) come negativi (da qui il segno negativo alle sorgenti). Ad esempio, avremo per l'esempio precedente:

$$\begin{cases} -x_{12} - x_{13} = -5 \\ x_{12} + x_{32} - x_{24} - x_{25} = -6 \\ x_{13} - x_{32} - x_{35} = 3 \\ x_{24} + x_{54} = 4 \\ x_{25} + x_{35} - x_{54} = -4 \\ x_{ij} \geq 0 \end{cases}$$

in formato duale standard.

La situazione diventa un problema di LP quando si introduce un **costo di movimentazione** su ogni arco: a questo punto può essere interessante calcolare, se ogni unità di flusso "paga" per ogni unità il costo dell'arco su cui scorre, il **flusso di costo minimo** sul grafo. Questo si traduce nella funzione costo:

$$\min 2x_{12} + 3x_{13} + 4x_{24} + 6x_{25} + 7x_{32} + 5x_{35} + 8x_{54} \equiv c^T \cdot x$$

e quindi dà il sistema completo:

$$\begin{cases} \min 2x_{12} + 3x_{13} + 4x_{24} + 6x_{25} + 7x_{32} + 5x_{35} + 8x_{54} \\ -x_{12} - x_{13} = -5 \\ x_{12} + x_{32} - x_{24} - x_{25} = -6 \\ x_{13} - x_{32} - x_{35} = 3 \\ x_{24} + x_{54} = 4 \\ x_{25} + x_{35} - x_{54} = -4 \\ x_{ij} \geq 0 \end{cases}$$

1.1.1 Matrici di incidenza

Diamo una definizione più generale dell'insieme di vincoli necessari a definire un problema di ottimizzazione su un grafo. Avevamo che vogliamo, preso il vettore di bilanci b_i , eguagliare il bilancio del flusso effettivo su un nodo in funzione degli archi entranti ed uscenti con questo vettore. Questo si esprime nella forma:

$$\sum_{(e,i) \in A} x_{ei} - \sum_{(i,u) \in A} x_{iu} = b_i$$

dove si è preso con A l'insieme di tutti gli archi, con $(e,i) \in A$ gli indici dei nodi che formano (*incidono su*) archi entranti al nodo, e con $(i,q) \in A$ gli indici dei nodi che formano archi uscenti dal nodo, cioè la definizione operativa di bilancio del flusso che avevamo usato nell'esempio. Più formalmente, abbiamo che i vincoli di un problema di ottimizzazione su un grafo rispettano la forma:

$$Ex = b$$

dove E è la **matrice di incidenza** del grafo.

Definizione 1.1: Matrice di incidenza

Per un grafo di $\text{card}(N) = n$ nodi e $\text{card}(A) = m$ archi, la matrice di incidenza E è la matrice $n \times m$ che mette in relazione ogni nodo con ogni arco, con i valori e_{ij} :

- $e_{ij} = 0$: il nodo non partecipa all'arco;
- $e_{ij} = -1$: il nodo è la partenza dell'arco;
- $e_{ij} = 1$: il nodo è l'arrivo dell'arco.

Riassumiamo la differenza fra **matrice di incidenza** e **matrice di adiacenza** (che abbiamo già usato, ad esempio nei problemi di assegnamento di costo minimo e di trasporto). La prima mette in relazione nodi con archi, la seconda nodi con nodi. Entrambe sono utili alla modellizzazione di problemi di tipo diverso. A scopo esplicativo, si riportano le matrici di incidenza E e di adiacenza A dell'esempio precedente:

$$E = \begin{array}{c|ccccccc} & (1,2) & (1,3) & (2,3) & (2,4) & (2,5) & (3,5) & (4,5) \\ \hline 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 1 & -1 & -1 & 0 & 0 \\ 3 & 0 & 1 & -1 & 0 & 0 & -1 & 0 \\ 4 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 5 & 0 & 0 & 0 & 0 & 1 & 1 & -1 \end{array} \quad A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Abbiamo quindi che la forma più generale di un problema di ottimizzazione su un grafo è:

$$\begin{cases} \min c^T \cdot x \\ Ex = b \\ x \geq 0 \end{cases}$$

dove segnaliamo è importante il grafo sia **connesso**:

Definizione 1.2: Grafo connesso

Un grafo si dice connesso quando l'insieme degli archi A contiene un albero di copertura $T \subseteq A$.

Questa definizione (che in verità si potrebbe intendere come caratterizzazione) si fonda sull'idea di albero di copertura (che in verità abbiamo già usato nei problemi del commesso viaggiatore in ILP), che definiamo come segue:

Definizione 1.3: Albero di copertura

Dato un grafo $G = (N, A)$, un albero di copertura è un sottoinsieme di archi $T \subseteq A$ tale che il sottografo (N, T) è connesso e non presenta cicli. Un nodo di questo grafo su cui incide solo un arco si dice **foglia**.

Notiamo che su ogni colonna della matrice di adiacenza si ha esattamente un 1 e un -1, e tutti gli altri elementi valgono 0 (in altre parole, un arco è formato da un nodo di partenza e un nodo di arrivo). Questo significa che la somma delle righe della matrice è nulla, ergo per definizione queste sono fra di loro linearmente dipendenti. Si ha quindi che:

Teorema 1.1: Rango della matrice di incidenza

La matrice di incidenza di un problema di ottimizzazione su un grafo con n nodi ha rango $n - 1$.

La proposizione precedente rispetto alla dipendenza lineare delle righe della matrice di adiacenza è solo metà delle ipotesi necessarie a dimostrare quanto sopra: il resto viene dalla definizione di albero di copertura. Dimostriamo quindi che il rango vale $n - 1$, mostrando come si possono prendere come minori della matrice gli alberi di copertura costruiti sul grafo in questione. Per formare un albero di copertura si prendono i minimi $n - 1$ archi necessari a collegare fra di loro tutti i nodi, ergo si prendono le $n - 1$ colonne della matrice di adiacenza del grafo corrispondenti agli archi che formano l'albero per formare un minore E_T . Da queste, permutando appositamente righe e colonne, si può sempre ottenere una forma del tipo:

$$E_T = \left(\begin{array}{c|ccc} \pm 1 & 0 & \dots & 0 \\ * & E_S & & \end{array} \right)$$

dove la prima riga rappresenta una foglia z , e la matrice E_S la matrice di adiacenza del sottoalbero di copertura ottenuto su $N \setminus z$. Questa forma, per ipotesi induttiva, ci dice che la matrice di incidenza dell'albero è **triangolare inferiore**, da cui ricaviamo due cose:

- La matrice di adiacenza dell'albero di copertura, ergo il minore, è invertibile, da cui il rango $n - 1$;
- La matrice di adiacenza dell'albero ha determinante ± 1 , ergo è **unimodulare**. Questo ha dei risvolti importanti per quanto riguarda l'ILP, in quanto vedremo che una soluzione di LP per un problema di flusso minimo corrisponde con la soluzione del problema di ILP corrispondente (a patto di bilanci b_i interi).

1.1.2 Calcolo delle basi

Abbiamo appurato che la matrice di incidenza di un problema di ottimizzazione su un grafo ha rango $n - 1$ e presenta minori invertibili, rappresentanti matrici di incidenza di alberi di copertura. Visto che il problema stesso era stato espresso in forma duale standard, e che quello che facciamo nel prendere minori di rango $n - 1$ è effettivamente scegliere $n - 1$ colonne della matrice di incidenza, si ha che effettivamente ogni albero di copertura rappresenta una **base** del poliedro.

Teorema 1.2: Caratterizzazione di base di matrici di incidenza

ei ha che, per una matrice di incidenza E di un problema di ottimizzazione su grafi, un albero di copertura coincide con una data base T ottenuta scegliendo le $n - 1$ colonne degli archi che formano l'albero, ovvero:

$$T \text{ è un albero di copertura} \Leftrightarrow T \text{ è una base}$$

Vediamo un modo per calcolare una soluzione di base data una base T . Iniziamo con l'impostare, come era sempre valso per il duale standard, a 0 le variabili non di base. Adesso si può fare una visita posicipata per foglie (considerando la base come un albero). Vorremo che questa foglia rispetti:

$$E_T x_T = b$$

Partendo dalle foglie, si può mettere sull'arco corrispondente (e quindi sulla x_{ij} corrispondente) la capacità di quella foglia, e poi tagliarla dall'albero. Il procedimento si ripete sulle foglie rimaste, fino ad arrivare alla radice dell'albero.

Così facendo troveremo un vettore soluzione x_{ij} , su cui notiamo se almeno una componente è < 0 , la soluzione di base non è ammissibile (significa che un nodo sorgente chiede più di quanto può erogare).

1.1.3 Reti sbilanciate

Notiamo che finora abbiamo assunto l'ipotesi:

$$\sum_{i=1}^n b_i = 0$$

Potremmo avere invece che questa sommatoria è *negativa* o *positiva*, ergo il grafo **sbilanciato** in **positivo** o in **negativo**. Nel primo caso, si ha più ricchezza che produzione, e il problema è quindi vuoto. Nel secondo, si ha più produzione che richiesta, e bisogna quindi introdurre un **nodo fittizio**, di capacità pari all'**eccedente** $\sum_{i=1}^n b_i$ e costo nullo, dove verrà a riversarsi (in condizioni di ottimo) il flusso in eccesso.