



SAPIENZA
UNIVERSITÀ DI ROMA

Ottimizzazione su reti

Esercizi 2 — Alberi e cammini

Esercizi svolti sui capitoli del volume 2, alberi e cammini: ogni esercizio è seguito dalla sua soluzione completa.

Fabio Furini

fabiofurini.github.io

Dipartimento di Ingegneria informatica, automatica e gestionale — Sapienza Università di Roma

Versione online e materiale scaricabile:
fabiofurini.github.io/ottimizzazione-su-reti/
github.com/fabiofurini/ottimizzazione-su-reti

Anno accademico 2026–2027

Indice

2	Alberi ricoprenti di costo minimo	2
3	Cammini minimi	12



Alberi ricoprenti di costo minimo

Regole utilizzate in questi esercizi

Le definizioni e gli algoritmi del capitolo «Minimum albero ricoprente», per un grafo non orientato connesso $G = (V, E)$ con $n = |V|$, $m = |E|$ e pesi w_e .

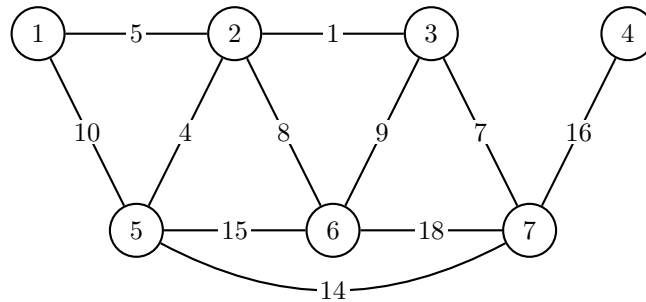
- (M1) **Proprietà di taglio.** Se nessun lato di F attraversa $\delta(S)$ e F è contenuto in un'estensione minima albero, aggiungendo un lato più leggero di $\delta(S)$ si mantiene questa proprietà; un lato più leggero unico di $\delta(S)$ è in ogni albero di copertura minimo.
- (M2) **Proprietà del ciclo.** Il lato più pesante di un ciclo può essere tralasciato; un lato rigorosamente più pesante degli altri lati di un ciclo non si trovano in un albero di copertura minimo.
- (M3) **Ottimalità.** Uno albero ricoprente T è ottimale se e solo se $w_e \geq w_f$ per ogni $e \notin T$ e ogni f del suo ciclo fondamentale $C(T, e)$; è l'unico ottimo se e solo se tutte queste disuguaglianze esistono rigoroso.
- (M4) **Kruskal.** I lati sono scanditi con peso non decrescente, legature di valore in ordine lessicografico di $\{u, v\}$, $u < v$; un lato è accettato se i suoi punti finali si trovano in componenti diversi di (V, F) .
- (M5) **Prim.** Da $s = 1$: selezionare $v \notin S$ con il minimo $d(v)$, legare per l'indice più piccolo, aggiungere $\{pred(v), v\}$; quindi $d(u) := w_{vu}$, $pred(u) := v$ per ogni $u \notin S$ adiacente a v con $w_{vu} < d(u)$.

Esercizio 2.1 — Algoritmo di Kruskal

Consideriamo il grafo non orientato $G = (V, E)$ con $V = \{1, 2, \dots, 7\}$ e gli archi 11 con i pesi

lato	$\{1, 2\}$	$\{1, 5\}$	$\{2, 3\}$	$\{2, 5\}$	$\{2, 6\}$	$\{3, 6\}$
w_e	5	10	1	4	8	9
lato	$\{3, 7\}$	$\{4, 7\}$	$\{5, 6\}$	$\{5, 7\}$	$\{6, 7\}$	
w_e	7	16	15	14	18	

disegnato qui sotto.



Domande

1. Ordinare i lati come in (M4).
2. Esegui l'algoritmo di Kruskal: per ogni lato scansionato, indica se è accettato o rifiutato e fornisci i componenti di (V, F) .
3. Disegna l'albero di copertura minimo e indica il suo peso.
4. Per ogni lato scartato, dare un ciclo di cui è il lato più pesante.

Soluzione

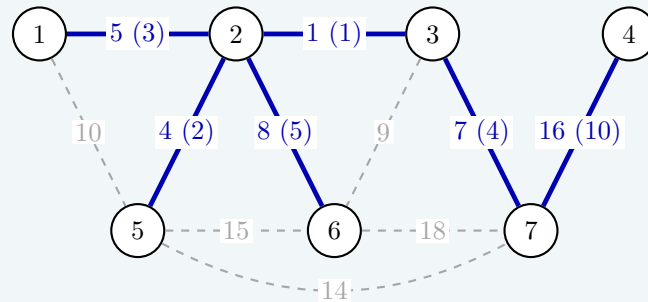
Usiamo le regole (M1)–(M5) del riquadro all'inizio di questi esercizi.

1. Con peso non decrescente: $\{2, 3\}$, $\{2, 5\}$, $\{1, 2\}$, $\{3, 7\}$, $\{2, 6\}$, $\{3, 6\}$, $\{1, 5\}$, $\{5, 7\}$, $\{5, 6\}$, $\{4, 7\}$, $\{6, 7\}$ con pesi 1, 4, 5, 7, 8, 9, 10, 14, 15, 16, 18.
2. Ogni riga rappresenta un lato scansionato:

k	lato e_k	w_{e_k}	decisione e componenti di (V, F)
1	$\{2, 3\}$	1	accetta: $\{1\}, \{2, 3\}, \{4\}, \{5\}, \{6\}, \{7\}$
2	$\{2, 5\}$	4	accetta: $\{1\}, \{2, 3, 5\}, \{4\}, \{6\}, \{7\}$
3	$\{1, 2\}$	5	accetta: $\{1, 2, 3, 5\}, \{4\}, \{6\}, \{7\}$
4	$\{3, 7\}$	7	accetta: $\{1, 2, 3, 5, 7\}, \{4\}, \{6\}$
5	$\{2, 6\}$	8	accetta: $\{1, 2, 3, 5, 6, 7\}, \{4\}$
6	$\{3, 6\}$	9	rifiuta (3 e 6 sono entrambi in $\{1, 2, 3, 5, 6, 7\}$)
7	$\{1, 5\}$	10	rifiuta (1 e 5 sono entrambi in $\{1, 2, 3, 5, 6, 7\}$)
8	$\{5, 7\}$	14	rifiuta (5 e 7 sono entrambi in $\{1, 2, 3, 5, 6, 7\}$)
9	$\{5, 6\}$	15	rifiuta (5 e 6 sono entrambi in $\{1, 2, 3, 5, 6, 7\}$)
10	$\{4, 7\}$	16	accetta: $\{1, 2, 3, 4, 5, 6, 7\}$

L'algoritmo si ferma dopo 10 dei lati 11, quando $|F| = 6$. I lati $\{6, 7\}$ non vengono mai scansionati.

3. L'albero è $F = \{\{1, 2\}, \{2, 3\}, \{2, 5\}, \{2, 6\}, \{3, 7\}, \{4, 7\}\}$, di peso $5+1+4+8+7+16 = 41$; su ciascun lato dell'albero, il peso e tra parentesi il passo della sua accettazione:



4. Un lato rifiutato unisce due vertici già collegati da F , e il percorso di F tra di loro utilizza solo lati scansionati prima di esso, quindi più leggeri: $\{3, 6\}$, di peso 9, chiude il ciclo $3, 2, 6, 3$ con alberi di lati di pesi 1, 8; $\{1, 5\}$, di peso 10, chiude il ciclo $1, 2, 5, 1$ con lati ad albero di pesi 5, 4; $\{5, 7\}$, di peso 14, chiude il ciclo $5, 2, 3, 7, 5$ con lati ad albero di pesi 4, 1, 7; $\{5, 6\}$, di peso 15, chiude il ciclo $5, 2, 6, 5$ con lati ad albero di pesi 4, 8. Per (M2) nessuno di questi archi si trova in un albero di copertura minimo.

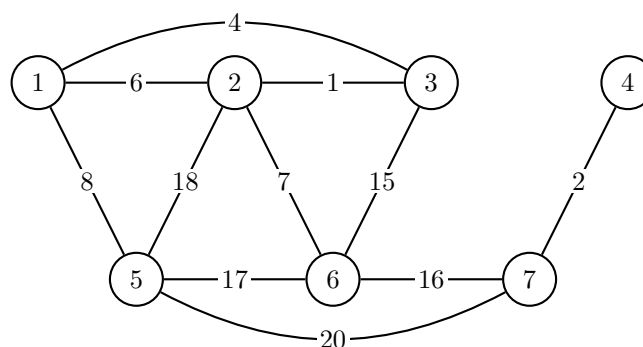
L'algoritmo di Kruskal accetta $\{2, 3\}$, $\{2, 5\}$, $\{1, 2\}$, $\{3, 7\}$, $\{2, 6\}$, $\{4, 7\}$ in questo ordine e trova l'albero di copertura minimo di peso 41; è unico, poiché i pesi sono distinti.

Esercizio 2.2 — Algoritmo di Prim

Consideriamo il grafo non orientato $G = (V, E)$ con $V = \{1, 2, \dots, 7\}$ e gli archi 11 con i pesi

lato	$\{1, 2\}$	$\{1, 3\}$	$\{1, 5\}$	$\{2, 3\}$	$\{2, 5\}$	$\{2, 6\}$
w_e	6	4	8	1	18	7
lato	$\{3, 6\}$	$\{4, 7\}$	$\{5, 6\}$	$\{5, 7\}$	$\{6, 7\}$	
w_e	15	2	17	20	16	

disegnato qui sotto.



Domande

1. Esegui l'algoritmo di Prim da $s = 1$: dopo ogni iterazione, fornisci il vertice selezionato, il lato aggiunto e le etichette $d(v)$ e $pred(v)$ dei vertici esterni a S .

2. Quali etichette diminuiscono durante l'esecuzione e perché?
3. Disegna l'albero di copertura minimo e indica il suo peso.
4. In quale ordine l'algoritmo di Kruskal accetta gli stessi lati?

Soluzione

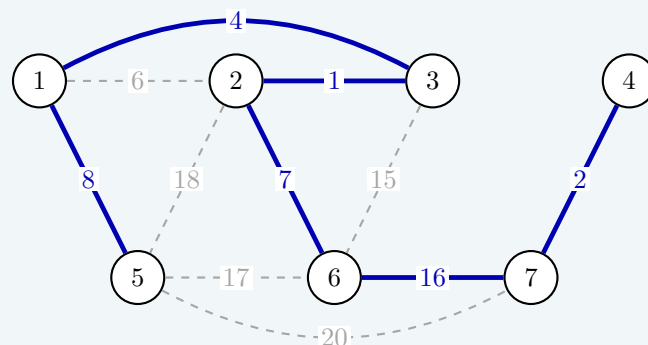
Usiamo le regole (M1)–(M5) del riquadro all'inizio di questi esercizi.

1. Ogni riga mostra $d(v)$ dopo l'aggiornamento, con $pred(v)$ tra parentesi; un trattino segna i vertici in S :

esso.	lato selezionato e aggiunto	$d(2)$	$d(3)$	$d(4)$	$d(5)$	$d(6)$	$d(7)$
0	1	–	6 (1)	4 (1)	∞	8 (1)	∞
1	3	{1, 3}	1 (3)	–	∞	8 (1)	15 (3)
2	2	{2, 3}	–	–	∞	8 (1)	7 (2)
3	6	{2, 6}	–	–	∞	8 (1)	–
4	5	{1, 5}	–	–	∞	–	16 (6)
5	7	{6, 7}	–	–	2 (7)	–	–
6	4	{4, 7}	–	–	–	–	–

I vertici entrano in S nell'ordine 1, 3, 2, 6, 5, 7, 4.

2. Un'etichetta diminuisce quando il vertice appena inserito ha un lato più chiaro verso u : all'iterazione 1, $d(2)$ scende da 6 a 1 (lato {2, 3}); all'iterazione 1, $d(6)$ scende da ∞ a 15 (lato {3, 6}); all'iterazione 2, $d(6)$ scende da 15 a 7 (lato {2, 6}); all'iterazione 3, $d(7)$ scende da ∞ a 16 (lato {6, 7}); all'iterazione 5, $d(4)$ scende da ∞ a 2 (lato {4, 7}).
3. L'albero è $F = \{\{1, 3\}, \{1, 5\}, \{2, 3\}, \{2, 6\}, \{4, 7\}, \{6, 7\}\}$, di peso 38:



4. L'algoritmo di Kruskal accetta i lati con peso non decrescente: {2, 3}, {4, 7}, {1, 3}, {2, 6}, {1, 5}, {6, 7}. L'albero è lo stesso, perché i pesi sono distinti e l'albero di copertura minimo è unico; cambia solo l'ordine.

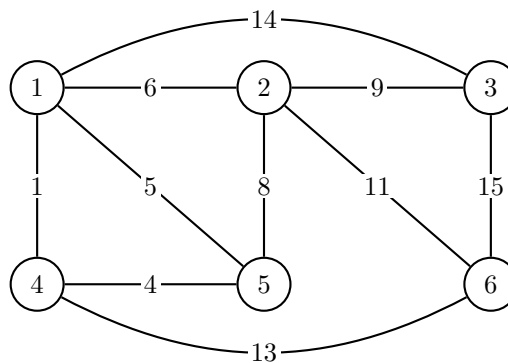
L'algoritmo di Prim da 1 aggiunge {1, 3}, {2, 3}, {2, 6}, {1, 5}, {6, 7}, {4, 7}; l'albero di copertura minimo pesa 38, come rilevato dall'algoritmo di Kruskal.

Esercizio 2.3 — Proprietà di taglio e ciclo

Consideriamo il grafo non orientato $G = (V, E)$ con $V = \{1, 2, \dots, 6\}$ e gli archi 10 con i pesi

lato	$\{1, 2\}$	$\{1, 3\}$	$\{1, 4\}$	$\{1, 5\}$	$\{2, 3\}$	$\{2, 5\}$
w_e	6	14	1	5	9	8
lato	$\{2, 6\}$	$\{3, 6\}$	$\{4, 5\}$	$\{4, 6\}$		
w_e	11	15	4	13		

disegnato qui sotto. Consideriamo anche lo albero ricoprente $T_0 = \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{1, 5\}, \{3, 6\}\}$.



Domande

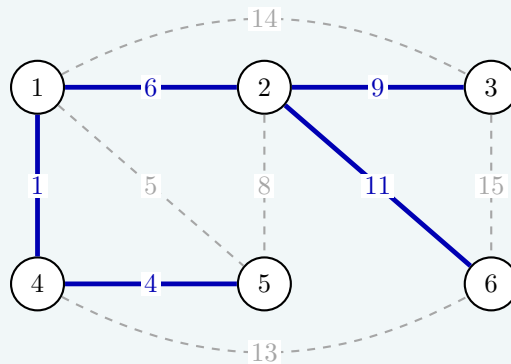
1. Trova il taglio $\delta(\{1, 2\})$. Quale dei suoi archi appartiene ad ogni albero di copertura minimo?
2. Trova un ciclo che contiene il lato $\{2, 5\}$ e usalo per mostrare che $\{2, 5\}$ non si trova in alcun albero di copertura minimo.
3. T_0 è un albero ricoprente minimo? Elenca le coppie (e, f) , $e \notin T_0$, $f \in C(T_0, e)$, che violano la condizione del ciclo (M3).
4. Migliora T_0 con lo scambio della coppia violante con la differenza maggiore $w_f - w_e$ e assegna il nuovo peso. Qual è il peso di un albero di copertura minimo?

Soluzione

Usiamo le regole (M1)–(M5) del riquadro all’inizio di questi esercizi.

1. $\delta(\{1, 2\}) = \{\{1, 3\}, \{1, 4\}, \{1, 5\}, \{2, 3\}, \{2, 5\}, \{2, 6\}\}$, con pesi 14, 1, 5, 9, 8, 11. Il lato $\{1, 4\}$ è l’unico più leggero, quindi appartiene ad ogni albero di copertura minimo di (M1).
2. Il ciclo $2, 1, 4, 5, 2$ utilizza i lati $\{1, 2\}, \{1, 4\}, \{4, 5\}, \{2, 5\}$, con pesi 6, 1, 4, 8. Il lato $\{2, 5\}$, di peso 8, è rigorosamente il più pesante, quindi non ha un albero di copertura minimo di (M2).

3. T_0 pesa 41. Per ogni lato e esterno a T_0 confrontiamo w_e con i lati del suo ciclo fondamentale; le coppie violanti, con $w_f > w_e$, sono $(\{2, 3\}, \{1, 3\})$, $(\{2, 6\}, \{1, 3\})$, $(\{2, 6\}, \{3, 6\})$, $(\{4, 5\}, \{1, 5\})$, $(\{4, 6\}, \{1, 3\})$, $(\{4, 6\}, \{3, 6\})$. Per (M3), T_0 non è ottimale.
4. La differenza più grande è data da $e = \{2, 3\}$ e $f = \{1, 3\}$, con $w_f - w_e = 14 - 9 = 5$: il lato f giace su $C(T_0, e)$, i cui lati sono $\{1, 2\}, \{1, 3\}, \{2, 3\}$. Lo scambio fornisce lo albero ricoprente $\{\{1, 2\}, \{1, 4\}, \{1, 5\}, \{2, 3\}, \{3, 6\}\}$ di peso $41 - 5 = 36$. L'algoritmo di Kruskal fornisce l'albero di copertura minimo $\{\{1, 2\}, \{1, 4\}, \{2, 3\}, \{2, 6\}, \{4, 5\}\}$, di peso 31:



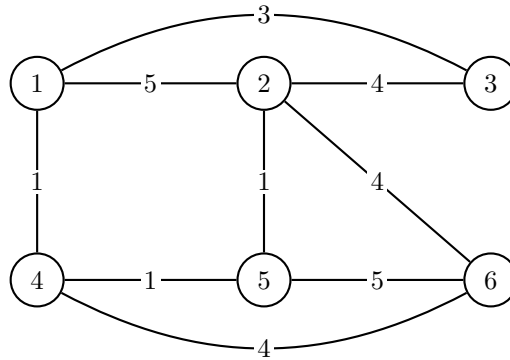
$\{1, 4\}$ viene forzato da un taglio, $\{2, 5\}$ viene escluso da un ciclo; T_0 (peso 41) viola la condizione del ciclo e il peso ottimale 31.

Esercizio 2.4 — Unicità e legami

Consideriamo il grafo non orientato $G = (V, E)$ con $V = \{1, 2, \dots, 6\}$ e gli archi 9 con i pesi

lato	$\{1, 2\}$	$\{1, 3\}$	$\{1, 4\}$	$\{2, 3\}$	$\{2, 5\}$	$\{2, 6\}$
w_e	5	3	1	4	1	4
lato	$\{4, 5\}$	$\{4, 6\}$	$\{5, 6\}$			
w_e	1	4	5			

disegnato qui sotto.



Domande

1. Trova un albero di copertura minimo T con l'algoritmo di Kruskal e il suo peso.
2. Utilizzando i tagli fondamentali di T , trova gli lati che appartengono a ogni albero di copertura minimo.
3. Utilizzando i cicli fondamentali di T , trovare gli archi che non appartengono ad alcun albero di copertura minimo.
4. L'albero ricoprente minimo è unico? Fornirne un altro, se esiste, e il numero minimo di albero ricoprente.

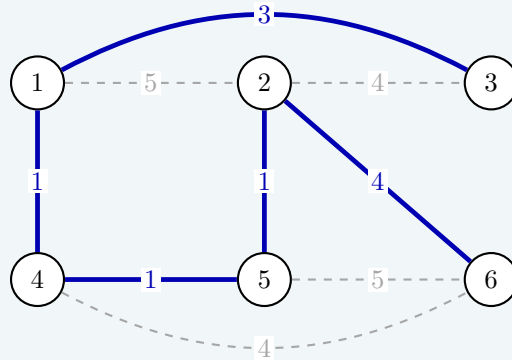
Soluzione

Usiamo le regole (M1)–(M5) del riquadro all'inizio di questi esercizi.

1. Algoritmo di Kruskal, con legature di valore in ordine lessicografico:

k	lato e_k	w_{e_k}	decisione e componenti di (V, F)
1	$\{1, 4\}$	1	accetta: $\{1, 4\}, \{2\}, \{3\}, \{5\}, \{6\}$
2	$\{2, 5\}$	1	accetta: $\{1, 4\}, \{2, 5\}, \{3\}, \{6\}$
3	$\{4, 5\}$	1	accetta: $\{1, 2, 4, 5\}, \{3\}, \{6\}$
4	$\{1, 3\}$	3	accetta: $\{1, 2, 3, 4, 5\}, \{6\}$
5	$\{2, 3\}$	4	rifiuta (2 e 3 sono entrambi in $\{1, 2, 3, 4, 5\}$)
6	$\{2, 6\}$	4	accetta: $\{1, 2, 3, 4, 5, 6\}$

$T = \{\{1, 3\}, \{1, 4\}, \{2, 5\}, \{2, 6\}, \{4, 5\}\}$, di peso 10.



2. Un lato di T si trova in ogni albero di copertura minimo se e solo se è strettamente il più leggero del suo taglio fondamentale (il componente S_e di $T \setminus \{e\}$ con il punto finale più piccolo):

$e \in T$	w_e	S_e	pesi degli altri lati di $\delta(S_e)$	rigorosamente più leggeri?
$\{1, 3\}$	3	$\{1, 2, 4, 5, 6\}$	4	sì
$\{1, 4\}$	1	$\{1, 3\}$	5, 4	sì
$\{2, 5\}$	1	$\{2, 6\}$	5, 4, 4, 5	sì
$\{2, 6\}$	4	$\{1, 2, 3, 4, 5\}$	4, 5 e n.: $\{4, 6\}$	
$\{4, 5\}$	1	$\{1, 3, 4\}$	5, 4, 4	sì

I lati in ogni albero di copertura minimo sono $\{1, 3\}, \{1, 4\}, \{2, 5\}, \{4, 5\}$.

3. Un arco esterno a T non è in alcun albero di copertura minimo se e solo se è strettamente il più pesante del suo ciclo fondamentale:

$e \notin T$ e w_e e lati degli alberi di $C(T, e)$ e i loro pesi e rigorosamente più pesanti?				
$\{1, 2\}$	5	$\{1, 4\}, \{4, 5\}, \{2, 5\}$	1, 1, 1	sì
$\{2, 3\}$	4	$\{2, 5\}, \{4, 5\}, \{1, 4\}, \{1, 3\}$	1, 1, 1, 3	sì
$\{4, 6\}$	4	$\{4, 5\}, \{2, 5\}, \{2, 6\}$	1, 1, 4 e n.: $\{2, 6\}$	
$\{5, 6\}$	5	$\{2, 5\}, \{2, 6\}$	1, 4	sì

I lati nell'albero senza copertura minima sono $\{1, 2\}, \{2, 3\}, \{5, 6\}$.

4. Alcune disuguaglianze non sono strette, quindi per (M3) l'ottimo non è unico. Ad esempio $\{\{1, 3\}, \{1, 4\}, \{2, 5\}, \{4, 5\}, \{4, 6\}\}$ pesa anche 10. Enumerando gli alberi ricoprente 64, esattamente 2 hanno peso 10.

Il peso minimo è 10, raggiunto dagli albero ricoprente 2; i legami nei tagli e nei cicli fondamentali di un albero ottimale indicano quali lati sono forzati e quali sono esclusi.

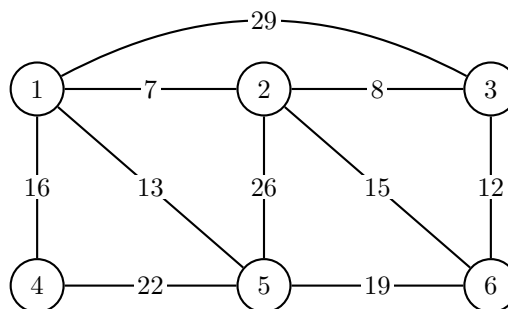
Esercizio 2.5 — Modellazione con alberi di copertura minimi

Un'azienda desidera collegare i propri edifici 6 $1, 2, \dots, 6$ con una rete in fibra. È possibile creare un collegamento diretto tra le coppie seguenti, e w_e è il suo costo (in migliaia di euro):

lato	$\{1, 2\}$	$\{1, 3\}$	$\{1, 4\}$	$\{1, 5\}$	$\{2, 3\}$	$\{2, 5\}$
w_e	7	29	16	13	8	26

lato	$\{2, 6\}$	$\{3, 6\}$	$\{4, 5\}$	$\{5, 6\}$
w_e	15	12	22	19

Ogni edificio deve poter raggiungere tutti gli altri, possibilmente attraverso altri edifici.



Ogni collegamento e funziona anche con probabilità p_e , indipendentemente dagli altri:

collegamento	$\{1, 2\}$	$\{1, 3\}$	$\{1, 4\}$	$\{1, 5\}$	$\{2, 3\}$
p_e	0.97	0.90	0.91	0.84	0.94

collegamento	$\{2, 5\}$	$\{2, 6\}$	$\{3, 6\}$	$\{4, 5\}$	$\{5, 6\}$
p_e	0.96	0.93	0.85	0.81	0.98

Domande

1. Modella la rete più economica come un problema grafo e risolvi.
2. Per motivi di sicurezza, gli edifici 2 e 6 devono essere collegati direttamente. Come trovi la rete più economica e quanto costa?
3. Una rete con collegamenti $n - 1$ funziona se tutti i suoi collegamenti funzionano. Modella la rete più affidabile come un problema di albero di copertura minimo, risolvi e fornisci la sua probabilità di funzionamento.

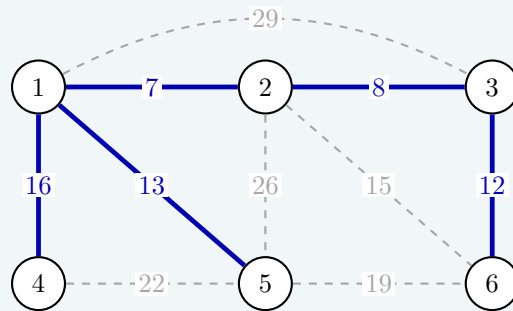
Soluzione

Usiamo le regole (M1)–(M5) del riquadro all'inizio di questi esercizi.

1. Gli edifici sono i vertici e gli eventuali collegamenti i lati di G , con pesi w_e . Una rete più economica è connessa e non ha ciclo: la rimozione di un lato di un ciclo la mantiene connessa e riduce il costo, poiché i costi sono positivi. Quindi è un albero

di copertura minimo. L'algoritmo di Kruskal dà

$$T = \{\{1, 2\}, \{1, 4\}, \{1, 5\}, \{2, 3\}, \{3, 6\}\}, \quad w(T) = 56.$$



2. Ogni albero ricoprente che contiene $\{2, 6\}$ cambia il suo peso della stessa quantità se cambia w_{26} . Impostando $w_{26} = 0$, al di sotto di ogni altro peso, l'algoritmo di Kruskal accetta prima $\{2, 6\}$ e restituisce l'albero più economico che lo contiene: $\{\{1, 2\}, \{1, 4\}, \{1, 5\}, \{2, 3\}, \{2, 6\}\}$, con i costi originali $w = 59$, cioè 3 maggiore di T .
3. La rete funziona con probabilità $\prod_{e \in T} p_e$. Massimizzarlo equivale a minimizzare $\sum_{e \in T} (-\log p_e)$, un problema di albero di copertura minimo con pesi $-\log p_e$. Questi pesi hanno l'ordine di $-p_e$, e l'algoritmo di Kruskal utilizza solo l'ordine: analizza i collegamenti con affidabilità non crescente e fornisce

$$\{\{1, 2\}, \{1, 4\}, \{2, 3\}, \{2, 5\}, \{5, 6\}\}, \quad \prod_e p_e = 0.97 \cdot 0.91 \cdot 0.94 \cdot 0.96 \cdot 0.98 \approx 0.7806.$$

Rete più economica: un albero ricoprente minimo di costo 56; con il collegamento forzato $\{2, 6\}$ costa 59; la rete più affidabile è un albero ricoprente massimo per p_e .

Cammini minimi

Regole utilizzate in questi esercizi

Le definizioni e gli algoritmi del capitolo «[Percorsi più brevi](#)», per una grafo $G = (N, A)$ con lunghezze d_{ij} , $n = |N|$, $m = |A|$ e sorgente $s = 1$.

(S1) **Dijkstra** ($d_{ij} \geq 0$). Il nodo temporaneo con l'etichetta più piccola diventa permanente (indice più basso sui pareggi); per ogni arco (i, j) con j temporaneo, se $d(i) + d_{ij} < d(j)$ allora $d(j) := d(i) + d_{ij}$ e $pred(j) := i$.

(S2) **Bellman–Ford**. Passa k calcola

$$d^k(j) = \min \left\{ d^{k-1}(j), \min_{(i,j) \in A} (d^{k-1}(i) + d_{ij}) \right\};$$

il predecessore cambia solo se l'etichetta diminuisce strettamente (il più piccolo i nelle legature di valore). Arresta quando $d^k = d^{k-1}$; un cambio passaggio n significa ciclo negativo raggiungibile da s .

(S3) **Ciclo negativo**. Se $d^n(j) < d^{n-1}(j)$, i predecessori leggono all'indietro da j , una riga per passare dalla riga n , eseguire una passeggiata con archi n che contengono un ciclo negativo.

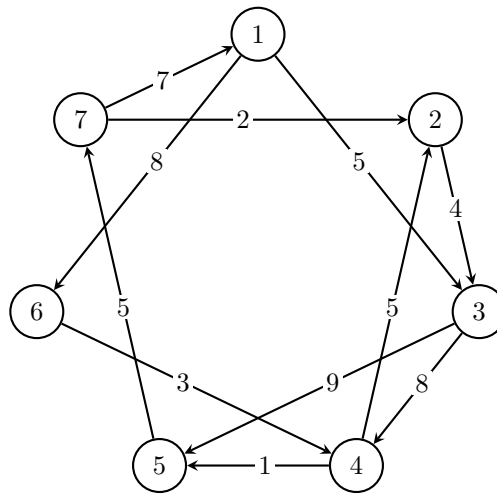
(S4) **Grafi aciclici**. Ordine topologico rimuovendo un nodo senza inserire archi (indice più piccolo); i nodi vengono elaborati in questo ordine, ognuno rilassando i propri archi uscenti. Le lunghezze $-d_{ij}$ danno i percorsi più lunghi.

(S5) **Potenziali**. $\bar{c}_{ij} = d_{ij} - \pi_i + \pi_j$; dalle etichette si ottiene $\pi_i = -d(i)$. Potenziali ammissibili, per cui $\bar{c}_{ij} \geq 0$, e un percorso da s a t formato da archi con costo ridotto nullo certificano che il percorso è minimo, con lunghezza $\pi_s - \pi_t$. Il duale di $SP(s, t)$ è $\max\{\pi_s - \pi_t : \pi_i - \pi_j \leq d_{ij}\}$.

Esercizio 3.1 — Algoritmo di Dijkstra

Consideriamo il grafo diretto $G = (N, A)$ con $N = \{1, 2, \dots, 7\}$, gli archi 11 e le lunghezze d_{ij} della tabella, disegnata sotto; la fonte è $s = 1$.

arco	(1, 3)	(1, 6)	(2, 3)	(3, 4)	(3, 5)	(4, 2)	(4, 5)	(5, 7)	(6, 4)	(7, 1)	(7, 2)
d_{ij}	5	8	4	8	9	5	1	5	3	7	2



Domande

1. Esegui l'algoritmo di Dijkstra da $s = 1$: fornisci le etichette dopo ogni iterazione e l'ordine in cui i nodi diventano permanenti.
2. Disegna l'albero del percorso più breve e fornisci un percorso più breve da 1 a 7 con la sua lunghezza.
3. Con i potenziali $\pi_i = -\tilde{d}(i)$, calcolare i costi ridotti ed elencare gli archi stretti.

Soluzione

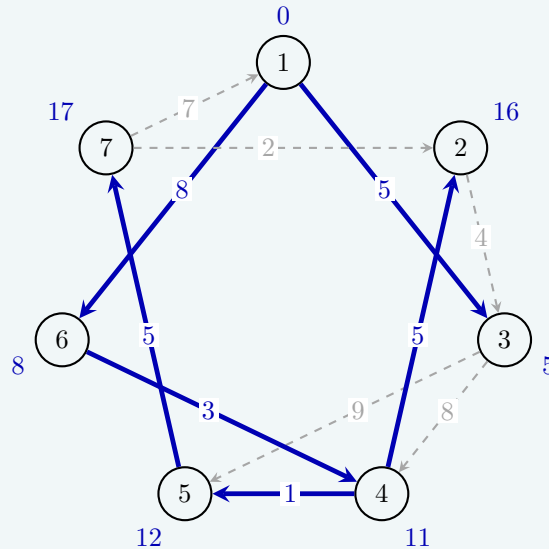
Usiamo le regole (S1)–(S5) del riquadro all'inizio di questi esercizi.

1. Ogni riga è un'iterazione; etichette permanenti in grassetto, etichetta che è diminuita con il nuovo predecessore tra parentesi:

esso.	i	$d(1)$	$d(2)$	$d(3)$	$d(4)$	$d(5)$	$d(6)$	$d(7)$
1 e 1 e 0 e ∞ e 5 (1) e ∞ e ∞ e 8 (1)	∞							
2	3	0	∞	5	13 (3)	14 (3)	8	∞
3	6	0	∞	5	11 (6)	14	8	∞
4	4	0	16 (4)	5	11	12 (4)	8	∞
5 e 5 e 0 e 16 e 5 e 11 e 12 e 8 e 17 (5)								
6	2	0	16	5	11	12	8	17
7	7	0	16	5	11	12	8	17

I nodi diventano permanenti nell'ordine 1, 3, 6, 4, 5, 2, 7 e $\tilde{d} = (0, 16, 5, 11, 12, 8, 17)$.

2. Gli archi $(pred(j), j)$ sono (4, 2), (1, 3), (6, 4), (4, 5), (1, 6), (5, 7) (blu); accanto ad ogni nodo la sua distanza:



Il percorso più breve da 1 a 7 è 1, 6, 4, 5, 7, di lunghezza $8 + 3 + 1 + 5 = 17$.

3. I potenziali sono $\pi = (0, -16, -5, -11, -12, -8, -17)$ e $\bar{c}_{ij} = d_{ij} - \pi_i + \pi_j$:

arco	(1, 3)	(1, 6)	(2, 3)	(3, 4)	(3, 5)	(4, 2)	(4, 5)	(5, 7)	(6, 4)	(7, 1)	(7, 2)
d_{ij}	5	8	4	8	9	5	1	5	3	7	2
$\bar{c}_{ij}$	0	0	15	2	2	0	0	0	0	24	3

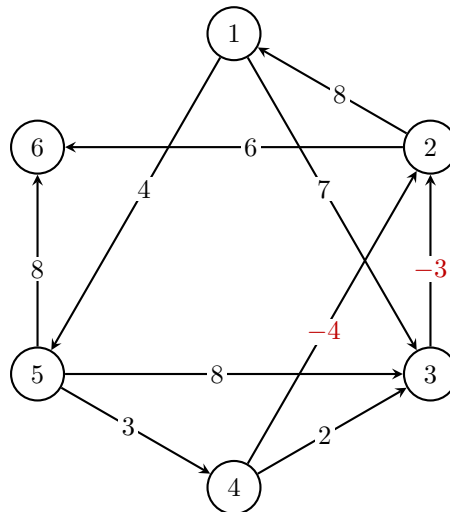
Tutti i costi ridotti sono non negativi e gli archi stretti sono (1, 3), (1, 6), (4, 2), (4, 5), (5, 7), (6, 4): esattamente gli archi dell'albero del percorso più breve.

L'algoritmo di Dijkstra dà $\tilde{d} = (0, 16, 5, 11, 12, 8, 17)$; i potenziali $\pi_i = -\tilde{d}(i)$ sono ammissibili, e i loro archi stretti formano l'albero dei cammini più brevi.

Esercizio 3.2 — Bellman–Ford con archi negativi

Consideriamo il grafo diretto $G = (N, A)$ con $N = \{1, 2, \dots, 6\}$, gli archi 10 e le lunghezze d_{ij} della tabella, disegnata sotto; la fonte è $s = 1$.

arco	(1, 3)	(1, 5)	(2, 1)	(2, 6)	(3, 2)	(4, 2)	(4, 3)	(5, 3)	(5, 4)	(5, 6)
d_{ij}	7	4	8	6	-3	-4	2	8	3	8



Domande

1. Eseguire l'algoritmo Bellman-Ford da $s = 1$: fornire le etichette d^k di ogni passaggio e il numero di passaggi.
2. Disegna l'albero del percorso più breve e fornisci un percorso più breve da 1 a 6.
3. Esegui l'algoritmo di Dijkstra sulle stesse lunghezze: quali etichette sono sbagliate e perché?

Soluzione

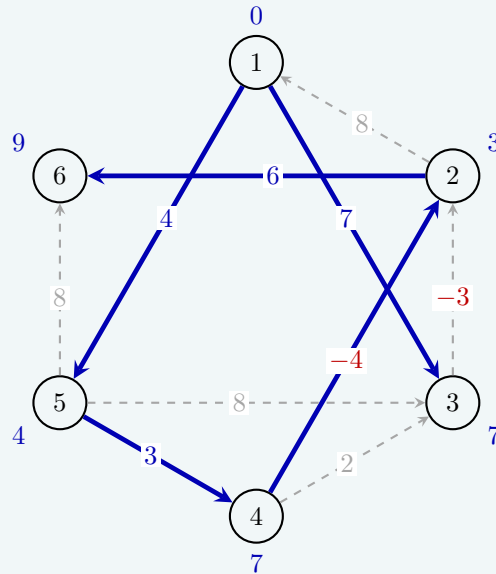
Usiamo le regole (S1)–(S5) del riquadro all'inizio di questi esercizi.

1. Ogni riga è un passaggio; accanto ad un'etichetta diminuita, il nuovo predecessore tra parentesi:

k	$d^k(1)$	$d^k(2)$	$d^k(3)$	$d^k(4)$	$d^k(5)$	$d^k(6)$
0	0	∞	∞	∞	∞	∞
1	0	∞	7 (1)	∞	4 (1)	∞
2 e 0 e 4 (3) e 7 e 7 (5) e 4 e 12 (5)						
3	0	3 (4)	7	7	4	10 (2)
4	0	3	7	7	4	9 (2)
5	0	3	7	7	4	9

Passaggio 5 ripete le etichette del passaggio 4: l'algoritmo si interrompe dopo che 5 passa con $\tilde{\mathbf{d}} = (0, 3, 7, 7, 4, 9)$.

2. Gli ultimi predecessori danno gli archi (4, 2), (1, 3), (5, 4), (1, 5), (2, 6) (blu):



Il percorso più breve da 1 a 6 è 1, 5, 4, 2, 6, di lunghezza 9.

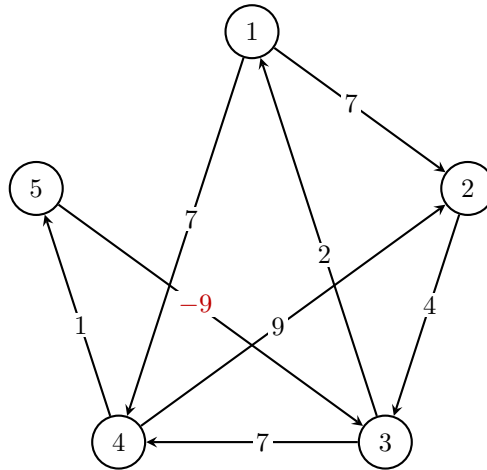
3. L'algoritmo di Dijkstra rende permanenti i nodi nell'ordine 1, 5, 3, 2, 4, 6 e restituisce (0, 4, 7, 7, 4, 10): le etichette dei nodi 2, 6 sono sbagliate. L'arco negativo (4, 2) termina con un nodo che è già permanente quando viene selezionata la sua coda, quindi (S1) non lo usa mai, sebbene $\tilde{d}(4) + d_{42} = 7 - 4 = 3 < 4$.

Con archi negativi e nessun ciclo negativo, l'algoritmo Bellman-Ford si arresta dopo che 5 passa con $\tilde{d} = (0, 3, 7, 7, 4, 9)$, mentre l'algoritmo di Dijkstra fallisce.

Esercizio 3.3 — Rilevamento di un ciclo negativo

Consideriamo il grafo diretto $G = (N, A)$ con $N = \{1, 2, \dots, 5\}$, gli archi 8 e le lunghezze d_{ij} della tabella, disegnata sotto; la fonte è $s = 1$.

arco	(1, 2)	(1, 4)	(2, 3)	(3, 1)	(3, 4)	(4, 2)	(4, 5)	(5, 3)
d_{ij}	7	7	4	2	7	9	1	-9



Domande

1. Eseguire l'algoritmo Bellman-Ford da $s = 1$ per i passaggi 5. G ha un ciclo negativo raggiungibile da 1?
2. Trova un ciclo negativo leggendo i predecessori all'indietro e indica la sua lunghezza.
3. Qual è il valore ottimale del modello LP a sorgente singola SP(1)?

Soluzione

Usiamo le regole (S1)–(S5) del riquadro all'inizio di questi esercizi.

1. Ogni riga è un passaggio; accanto ad un'etichetta diminuita, il nuovo predecessore tra parentesi:

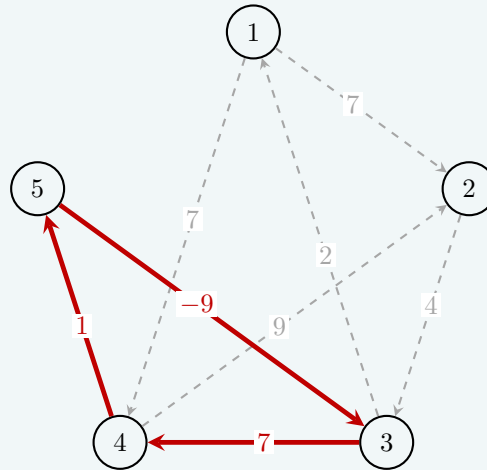
k	$d^k(1)$	$d^k(2)$	$d^k(3)$	$d^k(4)$	$d^k(5)$
0	0	∞	∞	∞	∞
1	0	7 (1)	∞	7 (1)	∞
2	0	7	11 (2)	7	8 (4)
3	0	7	-1 (5)	7	8
4	0	7	-1	6 (3)	8
5	0	7	-1	6	7 (4)

Nel passaggio $n = 5$ l'etichetta del nodo 5 diminuisce da 8 a 7: $d^5 \neq d^4$ e un ciclo negativo è raggiungibile da 1.

2. Lettura all'indietro dal nodo 5 (voci in grassetto): la riga 5 dà 4, la riga 4 dà 3, la riga 3 dà 5, la riga 2 dà 4, la riga 1 restituisce 1. La passeggiata è

$$1, 4, 5, 3, 4, 5, \quad \text{di lunghezza } 7 = d^5(5),$$

e il suo primo nodo ripetuto chiude il ciclo 4, 5, 3, 4, di lunghezza $1 - 9 + 7 = -1$ (rosso):



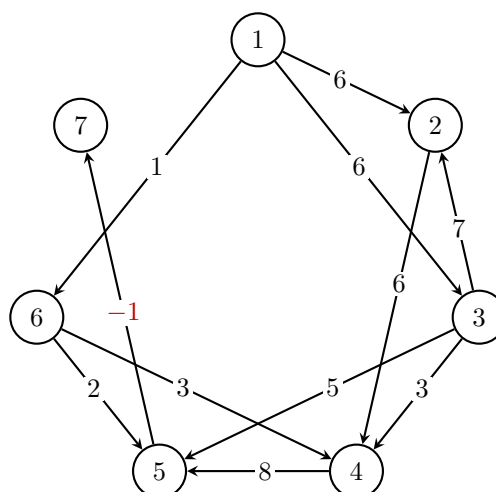
3. Il modello è fattibile, poiché ogni nodo è raggiungibile da 1, e G ha un ciclo negativo: aggiungendo λ unità di flusso attorno ad esso mantiene gli equilibri e abbassa il valore di λ . Il modello SP(1) è illimitato (lo conferma Gurobi).

Una modifica nel passaggio $n = 5$ rivela un ciclo negativo: il cammino letto all'indietro nella tabella contiene il ciclo 4, 5, 3, 4 di lunghezza -1 e il modello LP è illimitato.

Esercizio 3.4 — Cammini minimi in un grafo aciclico

Consideriamo il grafo diretto $G = (N, A)$ con $N = \{1, 2, \dots, 7\}$, gli archi 11 e le lunghezze d_{ij} della tabella, disegnata sotto; la fonte è $s = 1$.

arco	(1, 2)	(1, 3)	(1, 6)	(2, 4)	(3, 2)	(3, 4)	(3, 5)	(4, 5)	(5, 7)	(6, 4)	(6, 5)
d_{ij}	6	6	1	6	7	3	5	8	-1	3	2



Domande

1. Mostrare che G è aciclico trovandone l'ordine topologico con (S4).
2. Calcola i percorsi più brevi da $s = 1$ con l'algoritmo per grafi aciclici e disegna l'albero dei percorsi più brevi.
3. Calcola il percorso più lungo da 1 a 7 e la sua lunghezza.

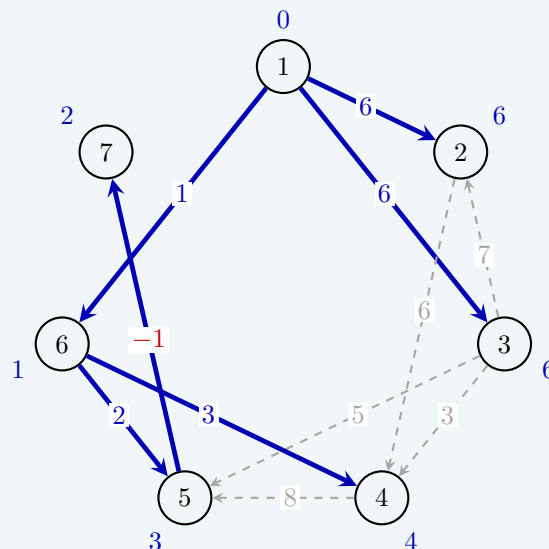
Soluzione

Usiamo le regole (S1)–(S5) del riquadro all'inizio di questi esercizi.

1. Togliendo ogni volta il nodo senza inserire archi di indice minore si ottiene l'ordine 1, 3, 2, 6, 4, 5, 7. Ogni arco avanza in questo ordine, quindi G non ha un ciclo diretto.
2. Ogni riga elabora un nodo e mostra le etichette successive:

gradino e nodo i	$d(1)$	$d(2)$	$d(3)$	$d(4)$	$d(5)$	$d(6)$	$d(7)$
1	1	0	6	6	∞	∞	1
2	3	0	6	6	9	11	1
3	2	0	6	6	9	11	1
4	6	0	6	6	4	3	1
5	4	0	6	6	4	3	1
6	5	0	6	6	4	3	1
7	7	0	6	6	4	3	1

Quindi $\tilde{d} = (0, 6, 6, 4, 3, 1, 2)$; i predecessori danno gli archi (1, 2), (1, 3), (6, 4), (6, 5), (1, 6), (5, 7) (blu):



Il percorso più breve da 1 a 7 è 1, 6, 5, 7, di lunghezza 2.

3. Con le lunghezze $-d_{ij}$ lo stesso algoritmo fornisce le lunghezze del percorso più lunghe (0, 13, 6, 19, 27, 1, 26).

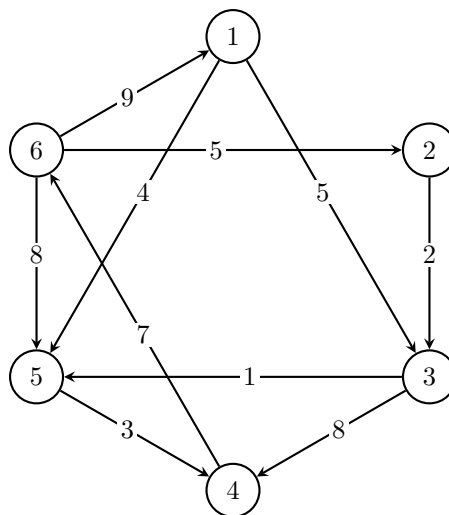
Il percorso più lungo da 1 a 7 è 1, 3, 2, 4, 5, 7, di lunghezza 26.

Nell'ordine topologico 1, 3, 2, 6, 4, 5, 7 ogni etichetta è definitiva quando il suo nodo viene elaborato: percorso più breve verso 7 di lunghezza 2, percorso più lungo di lunghezza 26.

Esercizio 3.5 — Potenziali come certificato di ottimalità

Consideriamo il grafo diretto $G = (N, A)$ con $N = \{1, 2, \dots, 6\}$, gli archi 10 e le lunghezze d_{ij} della tabella, disegnata sotto; la fonte è $s = 1$.

arco	(1, 3)	(1, 5)	(2, 3)	(3, 4)	(3, 5)	(4, 6)	(5, 4)	(6, 1)	(6, 2)	(6, 5)
d_{ij}	5	4	2	8	1	7	3	9	5	8



L'albero $\bar{T}$ con gli archi (6, 2), (1, 3), (3, 4), (1, 5), (4, 6) fornisce le etichette $\bar{d} = (0, 25, 5, 13, 4, 20)$, le lunghezze dei suoi percorsi da 1.

Domande

1. Scrivere il modello duale $D(\text{SP}(1, 6))$ del modello LP del percorso più breve da 1 a 6.
2. Calcola i costi ridotti dei potenziali $\bar{\pi}_i = -\bar{d}(i)$. Il percorso di $\bar{T}$ da 1 a 6 è il percorso più breve?
3. Vengono forniti i potenziali $\tilde{\pi} = (0, -19, -5, -7, -4, -14)$. Dimostrare che sono fattibili e trovare un percorso da 1 a 6 che certificano come il più breve.
4. Quali sono i valori ottimali di $\text{SP}(1, 6)$ e del suo duale?

Soluzione

Usiamo le regole (S1)–(S5) del riquadro all’inizio di questi esercizi.

1. Esiste una variabile libera π_i per nodo e un vincolo $\pi_i - \pi_j \leq d_{ij}$ per arco:

$$\begin{aligned}
 & \max \quad \pi_1 - \pi_6 \\
 & \text{subject to} \quad \pi_1 - \pi_3 \leq 5 \quad \text{arc } (1, 3) \\
 & \quad \pi_1 - \pi_5 \leq 4 \quad \text{arc } (1, 5) \\
 & \quad \pi_2 - \pi_3 \leq 2 \quad \text{arc } (2, 3) \\
 & \quad \pi_3 - \pi_4 \leq 8 \quad \text{arc } (3, 4) \\
 & \quad \pi_3 - \pi_5 \leq 1 \quad \text{arc } (3, 5) \\
 & \quad \pi_4 - \pi_6 \leq 7 \quad \text{arc } (4, 6) \\
 & \quad \pi_5 - \pi_4 \leq 3 \quad \text{arc } (5, 4) \\
 & \quad \pi_6 - \pi_1 \leq 9 \quad \text{arc } (6, 1) \\
 & \quad \pi_6 - \pi_2 \leq 5 \quad \text{arc } (6, 2) \\
 & \quad \pi_6 - \pi_5 \leq 8 \quad \text{arc } (6, 5) \\
 & \quad \pi_i \in \mathbb{R}, \quad i \in \{1, \dots, 6\}.
 \end{aligned}$$

2. Con $\bar{\pi} = (0, -25, -5, -13, -4, -20)$ i costi ridotti sono:

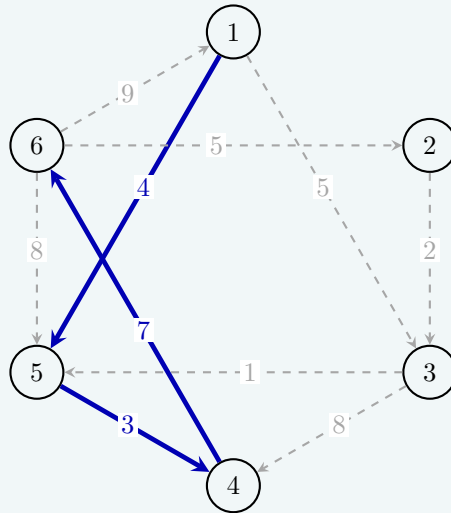
arco	(1, 3)	(1, 5)	(2, 3)	(3, 4)	(3, 5)	(4, 6)	(5, 4)	(6, 1)	(6, 2)	(6, 5)
d_{ij}	5	4	2	8	1	7	3	9	5	8
$\bar{c}_{ij}$	0	0	22	0	2	0	-6	29	0	24

L’arco (5, 4) ha $\bar{c}_{54} = 3 - (-4) + (-13) = -6 < 0$: i potenziali non sono realizzabili, e inserendo il nodo 4 attraverso (5, 4) si ottiene un percorso di lunghezza $7 < 13 = \bar{d}(4)$. Il percorso 1, 3, 4, 6 di $\bar{T}$ ha lunghezza 20, maggiore della distanza 14: non è il più breve.

3. I costi ridotti di $\tilde{\pi}$ sono tutti non negativi:

arco	(1, 3)	(1, 5)	(2, 3)	(3, 4)	(3, 5)	(4, 6)	(5, 4)	(6, 1)	(6, 2)	(6, 5)
d_{ij}	5	4	2	8	1	7	3	9	5	8
$\tilde{c}_{ij}$	0	0	16	6	2	0	0	23	0	18

Il percorso 1, 5, 4, 6 utilizza solo archi stretti, quindi per (S5) è il percorso più breve, di lunghezza $\tilde{\pi}_1 - \tilde{\pi}_6 = 14$.



4. Per dualità debole il vettore di incidenza di 1,5,4,6, di valore 14, e $\tilde{\pi}$, di valore $\tilde{\pi}_1 - \tilde{\pi}_6 = 14$, sono entrambi ottimali: $z(\text{SP}(1,6)) = z(D(\text{SP}(1,6))) = 14$ (Gurobi lo conferma).

I potenziali con un costo ridotto negativo indicano un percorso più breve; potenziali realizzabili con percorso stretto lo certificano: la distanza da 1 a 6 è 14.