



SAPIENZA
UNIVERSITÀ DI ROMA

Ottimizzazione su reti

Volume 1 — Grafi

I grafi orientati e non orientati e il loro linguaggio: rappresentazioni al calcolatore, matrice di incidenza nodo-arco, cammini, cicli e connessione, alberi, visite in ampiezza e in profondità.

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

1	Grafi	2
1.1	Grafi non orientati e orientati	3
1.2	Rappresentazioni di un grafo	7
1.3	Connettività	10
1.4	Alberi e foreste	11
1.5	Il rango della matrice di incidenza	15
1.6	Ricerca su grafi	17
1.7	Errori tipici	21
1.8	In sintesi	22



Grafi

Prerequisiti

- Algebra lineare: matrici, combinazioni lineari, vettori linearmente indipendenti e rango di una matrice.
- Insiemi, relazioni e classi di equivalenza; la notazione $O(\cdot)$ per il tempo di esecuzione di un algoritmo.
- Non è necessaria alcuna conoscenza della teoria dei grafi: ogni nozione è definita in questo capitolo.

Obiettivi

Alla fine di questo capitolo sarai in grado di:

- usare le nozioni di base dei grafi orientati e non orientati: adiacenza, grado, passeggiate, cammini e cicli;
- rappresentare un grafo mediante una matrice di adiacenza, liste di adiacenza, una lista di archi o una matrice di incidenza nodo-arco, e confrontare lo spazio richiesto da ciascuna rappresentazione;
- trovare le componenti connesse e le componenti fortemente connesse di un grafo;
- riconoscere un albero attraverso caratterizzazioni equivalenti e calcolare il rango della matrice di incidenza;
- eseguire la ricerca in ampiezza e in profondità, costruire l'albero dei predecessori e dimostrare che la ricerca individua i nodi raggiungibili in tempo $O(|N| + |A|)$.

1.1 Grafi non orientati e orientati

Definizione 1.1: grafo non orientato

Un **grafo non orientato** $G = (V, E)$ è formato da un insieme finito V di **vertici** e da un insieme E di **lati**, ciascuno dei quali è un sottoinsieme di V con due elementi:

$$E \subseteq \{ \{u, v\} : u, v \in V, u \neq v \}.$$

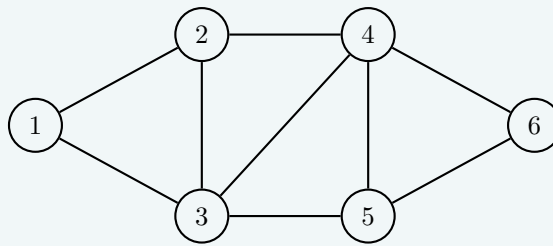
Scriviamo $n = |V|$ e $m = |E|$ e i vertici sono etichettati $1, 2, \dots, n$.

- I vertici u e v sono gli **estremi** del lato $\{u, v\}$. Il lato è **incidente** a u e a v ; di conseguenza, u e v sono **adiacenti**.
- L'**intorno** di v è $N(v) = \{u \in V : \{u, v\} \in E\}$; l'insieme dei lati incidenti a v è $\delta(v) = \{ \{u, v\} \in E : u \in V \}$ e il **grado** di v è $\deg(v) = |\delta(v)|$.

Esempio 1.1: un grafo non orientato

Consideriamo il grafo $G = (V, E)$ con $n = 6$ vertici e $m = 9$ lati:

$$V = \{1, 2, 3, 4, 5, 6\}, \quad E = \{ \{1, 2\}, \{1, 3\}, \{2, 3\}, \{2, 4\}, \{3, 4\}, \{3, 5\}, \{4, 5\}, \{4, 6\}, \{5, 6\} \}.$$



L'intorno del vertice 3 è $N(3) = \{1, 2, 4, 5\}$, quindi $\deg(3) = 4$. I gradi dei sei vertici sono

$$\deg(1) = 2, \quad \deg(2) = 3, \quad \deg(3) = 4, \quad \deg(4) = 4, \quad \deg(5) = 3, \quad \deg(6) = 2.$$

Proposizione 1.2: lemma della stretta di mano

Per ogni grafo non orientato $G = (V, E)$ abbiamo

$$\sum_{v \in V} \deg(v) = 2m.$$

Dimostrazione. La somma conta le coppie (v, e) con $v \in V$, $e \in \delta(v)$. Ogni lato $e = \{u, v\}$ appare esattamente in due coppie, (u, e) e (v, e) , quindi il numero di coppie è $2m$. $\square$

- In [Esempio 1.1](#) la somma dei gradi dà $2 + 3 + 4 + 4 + 3 + 2 = 18 = 2 \cdot 9$.

Definizione 1.3: grafo diretto

Un **grafo orientato** $G = (N, A)$ è dato da un insieme finito N di **nodi** e un insieme A di **archi**, ogni arco essendo una coppia ordinata di nodi distinti:

$$A \subseteq \{(i, j) : i, j \in N, i \neq j\}.$$

Scriviamo $n = |N|$ e $m = |A|$ e i nodi sono etichettati $1, 2, \dots, n$.

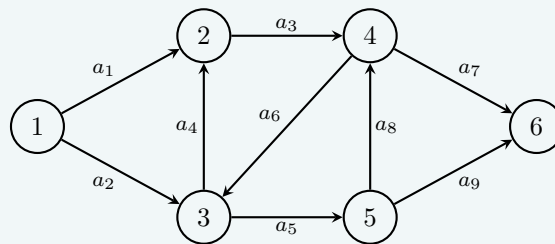
- L'arco (i, j) va dalla sua **coda** i alla sua **testa** j ; j è un **successore** di i e i è un **predecessore** di j .
 - La **stella uscente** $\delta^+(i) = \{(i, j) \in A\}$ contiene gli archi uscenti da i ; la **stella entrante** $\delta^-(i) = \{(j, i) \in A\}$ contiene gli archi entranti in i .
 - Il **grado uscente** di i è $\deg^+(i) = |\delta^+(i)|$ e il **grado entrante** è $\deg^-(i) = |\delta^-(i)|$.
- Gli archi (i, j) e (j, i) sono diversi: si chiamano **antiparallelo** e possono appartenere entrambi a A .
 - Il **grafo non orientato sottostante** di $G = (N, A)$ ha il lato $\{i, j\}$ per ogni arco (i, j) o (j, i) di A : si ottiene ignorando l'orientamento degli archi.

Esempio 1.2: il grafo di riferimento

Consideriamo il grafo diretto $G = (N, A)$ con nodi $n = 6$ e archi $m = 9$:

$$N = \{1, 2, 3, 4, 5, 6\}, \quad A = \{(1, 2), (1, 3), (2, 4), (3, 2), (3, 5), (4, 3), (4, 6), (5, 4), (5, 6)\}.$$

Gli archi sono numerati $a_1, a_2, \dots, a_9$ in questo ordine (lessicografo), come in figura.



I gradi uscenti e quelli entranti sono:

nodo i	1	2	3	4	5	6
$\deg^+(i)$	2	1	2	2	2	0
$\deg^-(i)$	0	2	2	2	1	2

Il grafo non orientato sottostante è il grafo di [Esempio 1.1](#). Chiameremo questo grafo il **grafo di riferimento**: tutti gli esempi del capitolo lo utilizzano.

- Ogni arco ha una coda e una testa, quindi $\sum_{i \in N} \deg^+(i) = \sum_{i \in N} \deg^-(i) = m$. Nel grafo di riferimento entrambe le somme sono 9.

1.1.1 Passeggiate, percorsi e cicli

Definizione 1.4: passeggiata, cammino e ciclo

Sia $G = (V, E)$ un grafo non orientato.

- Una **passeggiata** da v_0 a v_k è una sequenza $v_0, e_1, v_1, e_2, \dots, e_k, v_k$ di vertici e lati con $e_t = \{v_{t-1}, v_t\} \in E$ per tutti i $t \in \{1, 2, \dots, k\}$; la sua **lunghezza** è k .
- La passeggiata è **chiusa** se $v_0 = v_k$. È un **cammino semplice** se i vertici $v_0, v_1, \dots, v_k$ sono distinti.
- Un **ciclo** è una passeggiata chiusa con $k \geq 3$ i cui vertici $v_0, v_1, \dots, v_{k-1}$ sono distinti.

Un grafo senza cicli è **aciclico**. In un grafo semplice una passeggiata è identificata dai suoi vertici $v_0, v_1, \dots, v_k$.

Definizione 1.5: passeggiata, cammino e ciclo diretto

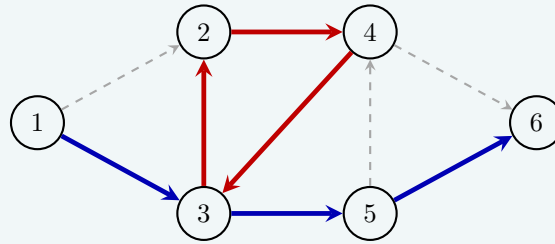
Sia $G = (N, A)$ un grafo diretto.

- Una **passeggiata diretta** da i_0 a i_k è una sequenza di nodi $i_0, i_1, \dots, i_k$ con $(i_{t-1}, i_t) \in A$ per tutti gli $t \in \{1, 2, \dots, k\}$: ogni arco viene percorso dalla coda alla testa.
- È un **cammino semplice diretto** se i suoi nodi sono distinti, e un **ciclo diretto** se $k \geq 2$, $i_0 = i_k$ e i nodi $i_0, i_1, \dots, i_{k-1}$ sono distinti.
- Una passeggiata, un cammino semplice o un ciclo del grafo non orientato sottostante è rispettivamente una passeggiata, un cammino semplice o un ciclo di G : gli archi possono essere attraversati in entrambe le direzioni. Un arco è percorso **in avanti** se si va dalla coda alla testa e **all'indietro** altrimenti.

Esempio 1.3: passeggiate, percorsi e cicli nel grafo di riferimento

Nel grafo di riferimento:

- 1, 3, 2, 4, 3, 5 è una passeggiata diretta di lunghezza 5; non è un cammino semplice perché il nodo 3 compare due volte;
- 1, 3, 5, 6 è un cammino semplice diretto di lunghezza 3 (in blu nella figura);
- 2, 4, 3, 2 è un ciclo diretto di lunghezza 3 (rosso in figura);
- 1, 2, 3 è un cammino semplice del grafo sottostante, ma non un cammino diretto: l'arco (1, 2) è percorso in avanti e (3, 2) all'indietro.


Lemma 1.6: ogni passeggiata contiene un cammino semplice

Se in un grafo esiste una passeggiata (diretta) da u a v , allora esiste anche un cammino semplice (diretto) da u a v , i cui vertici appartengono alla passeggiata.

Dimostrazione. Tra i cammini da u a v che utilizzano solo i vertici del cammino dato, sia $v_0, v_1, \dots, v_k$ uno dei minimi lunghezza. Se due gli indici $s < t$ avevano $v_s = v_t$, l'eliminazione di $v_{s+1}, \dots, v_t$ darebbe il cammino $v_0, \dots, v_s, v_{t+1}, \dots, v_k$ da u a v , più corto di $t - s \geq 1$. Quindi i vertici sono distinti e la passeggiata è un percorso. Vale lo stesso argomento per passeggiate guidate, poiché la cancellazione mantiene ogni arco percorso dalla coda alla testa. $\square$

- In [Esempio 1.3](#), eliminando il pezzo chiuso 3, 2, 4, 3 dalla passeggiata 1, 3, 2, 4, 3, 5 esce il percorso diretto 1, 3, 5.

1.1.2 Grafi semplici e multigrafi

- Un grafo è **semplice** se non ha **cappi** (lati $\{v, v\}$ o archi (i, i)) né **lati o archi paralleli** (due copie della stessa coppia). I grafi di [Definizioni 1.1](#) e [1.3](#) sono semplici, perché E e A sono insiemi di coppie di elementi distinti.
- Un **multigrafo** ammette cappi e lati o archi paralleli. In questo modulo tutti i grafi sono semplici e ogni arco è identificato dalla coppia (i, j) formata dalla sua coda e dalla sua testa.

Un grafo non orientato $G = (V, E)$ ha lati $\{u, v\}$; un grafo diretto $G = (N, A)$ ha archi (i, j) dalla coda i a la testa j . Un percorso ha vertici distinti e ogni passeggiata contiene un percorso con gli stessi punti finali.

1.2 Rappresentazioni di un grafo

Definizione 1.7: matrice di adiacenza

La **matrice di adiacenza** di un grafo orientato $G = (N, A)$ è la matrice $n \times n$ $\mathbf{M}$ con voci

$$m_{ij} = \begin{cases} 1 & \text{se } (i, j) \in A, \\ 0 & \text{altrimenti,} \end{cases} \quad \forall i, j \in \{1, 2, \dots, n\}.$$

La matrice di adiacenza di un grafo non orientato ha $m_{uv} = m_{vu} = 1$ per ogni lato $\{u, v\}$: è simmetrica.

Definizione 1.8: matrice di incidenza nodo-arco

Sia $G = (N, A)$ un grafo diretto con archi $a_1, a_2, \dots, a_m$. La **matrice di incidenza nodo-arco** è la $n \times m$ matrice $\mathbf{B}$ con una riga per nodo e una colonna per arco, le cui voci sono

$$b_{ik} = \begin{cases} +1 & \text{se } i \text{ è la coda di } a_k, \\ -1 & \text{se } i \text{ è la testa di } a_k, \\ 0 & \text{altrimenti,} \end{cases} \quad \forall i \in \{1, 2, \dots, n\}, \forall k \in \{1, 2, \dots, m\}.$$

- Ogni colonna di $\mathbf{B}$ ha esattamente uno $+1$ (la coda) e uno -1 (la testa), quindi la somma delle righe di $\mathbf{B}$ è il vettore zero.
- La riga i ha uno $+1$ per ogni arco di $\delta^+(i)$ e uno -1 per ogni arco di $\delta^-(i)$: ha $\deg^+(i) + \deg^-(i)$ voci diverse da zero.
- La **matrice di incidenza vertici-lati** di un grafo non orientato ha un 1 in entrambi i punti finali di ogni lato.

Esempio 1.4: le matrici del grafo di riferimento

La matrice di adiacenza $\mathbf{M}$ e la matrice di incidenza nodo-arco $\mathbf{B}$ del grafo di riferimento di [Esempio 1.2](#) vengono mostrati con l'indice di ogni riga (un nodo) e di ogni colonna (un nodo in $\mathbf{M}$, un arco in $\mathbf{B}$):

		1	2	3	4	5	6
	1	0	1	1	0	0	0
	2	0	0	0	1	0	0
$\mathbf{M} :$	3	0	1	0	0	1	0
	4	0	0	1	0	0	1
	5	0	0	0	1	0	1
	6	0	0	0	0	0	0

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9
$B :$	1	1	0	0	0	0	0	0	0
	2	-1	0	1	-1	0	0	0	0
	3	0	-1	0	1	1	-1	0	0
	4	0	0	-1	0	0	1	1	-1
	5	0	0	0	0	-1	0	0	1
	6	0	0	0	0	0	-1	0	-1

Ad esempio, la colonna $a_6 = (4, 3)$ ha $+1$ nella riga 4 (la coda) e -1 nella riga 3 (la testa). Riga 6 di M è zero, perché il nodo 6 non ha successori.

Definizione 1.9: elenchi di adiacenze ed elenchi di archi

Sia $G = (N, A)$ un grafo diretto.

- Le **liste di adiacenza** memorizzano, per ogni nodo i , la lista $L(i)$ dei suoi successori, cioè le teste degli archi di $\delta^+(i)$. In questo modulo ogni lista è ordinata per indice crescente.
- **Arc list** memorizza due array di lunghezza m , con la coda e la testa di ogni arco.

Per un grafo non orientato la lista $L(v)$ contiene l'intorno $N(v)$ e ogni arco appare in due liste.

Esempio 1.5: le liste del grafo di riferimento

Le liste di adiacenza e la lista di archi del grafo di riferimento sono:

i	$L(i)$	arco	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9
1	2, 3										
2	4										
3	2, 5	coda	1	1	2	3	3	4	4	5	5
4	3, 6	testa	2	3	4	2	5	3	6	4	6
5	4, 6										
6	(vuoto)										

Gli elenchi del grafo non orientato sottostante sono $L(1) = \{2, 3\}$, $L(2) = \{1, 3, 4\}$, $L(3) = \{1, 2, 4, 5\}$, $L(4) = \{2, 3, 5, 6\}$, $L(5) = \{3, 4, 6\}$ e $L(6) = \{4, 5\}$.

1.2.1 Spazio e tempo di accesso

- La tabella mette a confronto le quattro rappresentazioni di un grafo orientato. Lo spazio è il numero di voci memorizzate; le due operazioni sono il test di un arco e la scansione dei successori di un nodo.

rappresentazione e spazio ed è $(i, j) \in A$?	successori di i		
matrice di adiacenza	n^2	$O(1)$	$O(n)$
matrice di incidenza nodo-arco	nm	$O(m)$	$O(m)$
elenco archi	$2m$	$O(m)$	$O(m)$
elenchi di adiacenze	$n + m$	$O(\deg^+(i))$	$O(\deg^+(i))$

- Per il grafo di riferimento le quattro rappresentazioni memorizzano le voci 36, 54, 18 e 15.
- Un grafo diretto semplice ha $m \leq n(n - 1)$. Nelle reti delle applicazioni m è molto più piccolo di n^2 (il grafo è **sparse**) e le liste di adiacenza sono la rappresentazione scelta per gli algoritmi.

1.2.2 La matrice di incidenza e la conservazione del flusso

- Dato un valore x_{ij} per ogni arco, la riga i di $\mathbf{Bx}$ è ciò che lascia il nodo i meno ciò che vi entra:

$$(\mathbf{Bx})_i = \sum_{(i,j) \in \delta^+(i)} x_{ij} - \sum_{(j,i) \in \delta^-(i)} x_{ji}, \quad \forall i \in N.$$

- Con $x_{ij} = 1$ per ogni arco del grafo di riferimento, $\mathbf{Bx}$ è il vettore delle differenze $\deg^+(i) - \deg^-(i)$:

$$\mathbf{Bx} = (2, -1, 0, 0, 1, -2)'$$

La matrice di incidenza nodo-arco ha +1 nella coda e -1 nella testa di ogni arco; gli elenchi di adiacenza memorizzano a tracciare un grafo nello spazio $n + m$ e fornire i successori di un nodo nel tempo $O(\deg^+(i))$.

1.3 Connettività

Definizione 1.10: grafo connesso e componenti connessi

Sia $G = (V, E)$ un grafo non orientato.

- Due vertici u e v sono **connessi** se G ha un percorso da u a v ; il grafo G è **connected** se ogni due vertici sono collegati.
- Un **componente connesso** di G è un sottoinsieme massimo $U \subseteq V$ di vertici connessi a coppie.

Un grafo orientato è **debolmente connesso** se il suo grafo non orientato sottostante è connesso e il suo grafo connesso i componenti sono quelli del grafo non orientato sottostante.

Definizione 1.11: raggiungibilità e componenti fortemente connessi

Sia $G = (N, A)$ un grafo diretto.

- Un nodo j è **raggiungibile** da i se G ha un percorso diretto da i a j ; l'insieme dei nodi raggiungibile da i , i compreso, è indicato con $R(i)$.
- I nodi i e j sono **fortemente connessi** se $j \in R(i)$ e $i \in R(j)$; il grafo è **fortemente connesso** se ogni due nodi sono fortemente connessi.
- Un **componente fortemente connesso** di G è un sottoinsieme massimo $S \subseteq N$ di coppie fortemente connesse nodi connessi.

Proposizione 1.12: i componenti sono classi di equivalenza

Essere connessi è una relazione di equivalenza su V ed essere fortemente connessi è una relazione di equivalenza su N . Da qui la partizione di componenti connessi V e la partizione di componenti fortemente connessi N .

Dimostrazione. Dimostriamo le tre proprietà per la connessione forte; il caso non orientato è lo stesso con i percorsi al posto di percorsi diretti.

1. *Riflessivo*: il percorso i di lunghezza 0 mostra che $i \in R(i)$.
2. *Simmetrico*: la definizione è simmetrica in i e j .
3. *Transitivo*: un percorso diretto da i a j seguito da uno da j a h è una passeggiata diretta da i a h , che contiene un percorso diretto da i a h . Lo stesso vale da h a i .

Le classi di equivalenza sono i sottoinsiemi massimi di elementi correlati a coppie, cioè i componenti. $\square$

- Il **grafo invertito** $G' = (N, A')$ ha l'arco (j, i) per ogni arco $(i, j) \in A$. Il nodo i è raggiungibile da j in G se e solo se j è raggiungibile da i in G' ; indichiamo con $R'(i)$

l'insieme dei nodi raggiungibili da i in G' .

- Con [Definizione 1.11](#), il componente fortemente connesso che contiene i è

$$S(i) = R(i) \cap R'(i).$$

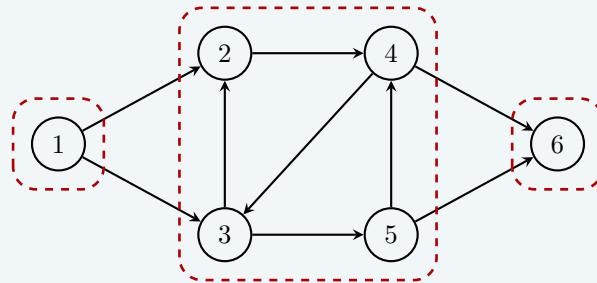
Esempio 1.6: componenti del grafo di riferimento

Nel grafo di riferimento:

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

Nel grafo invertito $R'(3) = \{1, 2, 3, 4, 5\}$ e $R(3) = \{2, 3, 4, 5, 6\}$, quindi $S(3) = \{2, 3, 4, 5\}$.
Il fortemente connesso i componenti sono

$$\{1\}, \quad \{2, 3, 4, 5\}, \quad \{6\}.$$



Il grafo è debolmente connesso, con una componente connessa, ma non è fortemente connesso: nessun nodo è raggiungibile from node 6.

Le componenti connesse e le componenti fortemente connesse sono le classi di due relazioni di equivalenza. Il Il componente fortemente connesso di i è $R(i) \cap R'(i)$: i nodi raggiungibili da i che possono anche raggiungere i .

1.4 Alberi e foreste

Definizione 1.13: albero, foresta e albero ricoprente

Sia $G = (V, E)$ un grafo non orientato.

- G è un **foresta** se è aciclico e un **albero** se è connesso e aciclico. Un vertice di grado 1 è un **foglia**.
- Un **albero ricoprente** di un grafo connesso G è un albero $T = (V, F)$ con $F \subseteq E$; un **spanning forest** di G è una foresta (V, F) , $F \subseteq E$, con gli stessi componenti connessi di G .

Per un grafo orientato $G = (N, A)$ le nozioni si applicano al grafo non orientato sottostante: un sottoinsieme di archi $F \subseteq A$ è un albero (una foresta) se gli lati corrispondenti formano un albero (una foresta).

Lemma 1.14: foglie

Ogni albero con vertici $n \geq 2$ ha almeno due foglie.

Dimostrazione. L'albero è connesso con i vertici $n \geq 2$, quindi ha un lato. Sia $v_0, v_1, \dots, v_k$ il percorso massimo lunghezza, con $k \geq 1$. Un vicino $u \neq v_1$ di v_0 si troverebbe sul percorso, chiudendo un ciclo, o si estenderebbe il percorso, contro la sua massimalità. Quindi $\deg(v_0) = 1$ e allo stesso modo $\deg(v_k) = 1$. $\square$

Lemma 1.15: aggiunta di un lato

Lascia che $G = (V, E)$ abbia componenti collegati k e lascia che $e = \{u, v\} \notin E$ con $u, v \in V$.

1. Se u e v si trovano in componenti diversi di G , il grafo $(V, E \cup \{e\})$ ha componenti $k - 1$ e nessuno scorrere e .
2. Se u e v si trovano nello stesso componente di G , il grafo $(V, E \cup \{e\})$ ha componenti k e un scorrere e .

Dimostrazione. 1. Il lato e unisce i componenti di u e v in uno solo e lascia invariati gli altri componenti. attraverso e darebbe, senza e , un percorso da u a v in G : impossibile.

2. I componenti non cambiano. Un percorso $u = v_0, v_1, \dots, v_k = v$ di G ha $k \geq 2$, poiché $\{u, v\} \notin E$; con e chiude il ciclo $v_0, v_1, \dots, v_k, v_0$. $\square$

Corollario 1.16: lati e componenti

Un grafo con vertici n , lati m e componenti connessi k soddisfa $k \geq n - m$, con uguaglianza se e solo se il grafo è aciclico.

Dimostrazione. Iniziare da $(V, \emptyset)$, che dispone di componenti n , e inserire i lati m uno alla volta. Di [Lemma 1.15](#), ogni inserimento diminuisce il numero di componenti di uno, oppure lascia rimane invariato e chiude un ciclo. Quindi $k \geq n - m$.

- Se il grafo ha un ciclo C , l'ultimo lato di C da inserire unisce due vertici già collegati dal resto di C : lascia il numero invariato e $k > n - m$.
- Se il grafo è aciclico nessun inserimento chiude un ciclo, perché un ciclo di un grafo intermedio è un ciclo di il grafo finale: $k = n - m$. $\square$

Teorema 1.17: caratterizzazioni di un albero

Sia $G = (V, E)$ un grafo non orientato con vertici n e archi m . Le seguenti affermazioni sono equivalenti:

- (a) G è un albero (connesso e aciclico);

- (b) G è collegato e $m = n - 1$;
- (c) G è aciclico e $m = n - 1$;
- (d) per ogni due vertici u e v esiste esattamente un percorso da u a v ;
- (e) G è collegato e la rimozione di qualsiasi lato lo disconnette.

Dimostrazione. Sia k il numero di componenti collegate; G è connesso se e solo se $k = 1$.

- **(a) $\Rightarrow$ (b),(c).** G è aciclico con $k = 1$, quindi $m = n - k = n - 1$ di [Corollario 1.16](#).
- **(b) $\Rightarrow$ (a).** Qui $k = 1 = n - m$: l'uguaglianza vale e G è aciclico per lo stesso corollario.
- **(c) $\Rightarrow$ (a).** G è aciclico, quindi $k = n - m = 1$ con lo stesso corollario: G è connesso.

Per $(a) \Leftrightarrow (d)$, connesso significa almeno un percorso tra ogni due vertici; lo mostriamo aciclico significa al massimo uno.

- Se G ha un ciclo $v_0, v_1, \dots, v_{k-1}, v_0$, i vertici v_0 e v_1 sono uniti da due percorsi: il lato $\{v_0, v_1\}$ e $v_0, v_{k-1}, \dots, v_2, v_1$.
- Lascia che $P \neq Q$ siano due percorsi da u a v . Sia w l'ultimo vertice della loro parte iniziale comune, e x il primo vertice dopo w su P che si trova anche su Q (v è un candidato). I pezzi di P e di Q di w a x condividono solo w e x e non sono entrambi costituiti dal lato singolo $\{w, x\}$, poiché P e Q lasciano w lungo lati diversi. Insieme formano un ciclo.

Per $(a) \Leftrightarrow (e)$, collegare G e $e = \{u, v\} \in E$. Applicazione [Lemma 1.15](#) a $(V, E \setminus \{e\})$ e e : il lato e giace su un ciclo di G se e solo se u e v sono collegati in $(V, E \setminus \{e\})$, cioè se e solo se rimuovendo e si lascia il grafo collegato. Quindi ogni rimozione disconnette G se e solo se su un ciclo non si trova alcun lato. $\square$

Corollario 1.18: foreste che si estendono

Ogni grafo con vertici n e componenti connessi k ha una foresta estesa, con lati $n - k$. In particolare, ogni grafo connesso ha un albero coprente, con archi $n - 1$.

Dimostrazione. Mentre il grafo ha un ciclo, rimuovi un lato $e = \{u, v\}$ di un ciclo. Il resto del ciclo collega ancora u e v , quindi aggiungere nuovamente e non modifica i componenti ([Lemma 1.15](#), parte 2): rimozione non li cambia neanche. Il grafo finale è aciclico con gli stessi componenti k , quindi è anche si estende sulla foresta e ha lati $n - k$ di [Corollario 1.16](#). $\square$

1.4.1 Cicli fondamentali

Proposizione 1.19: ciclo fondamentale

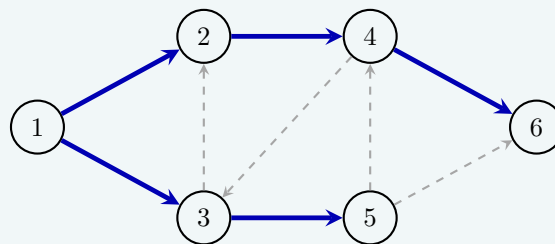
Sia $T = (V, F)$ uno albero ricoprente di un grafo connesso $G = (V, E)$ e sia $e = \{u, v\} \in E \setminus F$. Il grafo $(V, F \cup \{e\})$ contiene esattamente un ciclo, il **ciclo fondamentale** $C(T, e)$, formato da e e dal percorso unico da u a v in T .

Dimostrazione. I vertici u e v sono collegati in T , quindi il lato e e il percorso da u a v in T formano un ciclo (Lemma 1.15, parte 2). Ogni ciclo di $(V, F \cup \{e\})$ contiene e , poiché T è aciclico; la rimozione di e da esso lascia un percorso da u a v in T , che è unico per Teorema 1.17(d). $\square$

- Un grafo connesso con un albero ricoprente T ha cicli fondamentali $m - n + 1$, uno per ogni arco esterno a T .
- In un grafo orientato il ciclo fondamentale di un arco $(i, j) \notin F$ viene percorso nella direzione di (i, j) : gli archi di T percorsi dalla coda alla testa sono in avanti, gli altri all'indietro.

Esempio 1.7: un albero ricoprente del grafo di riferimento

Gli archi $F = \{(1, 2), (1, 3), (2, 4), (3, 5), (4, 6)\}$ formano un albero ricoprente T del grafo di riferimento (blu in figura): sono archi $5 = n - 1$ e i loro lati formano un grafo connesso.



Gli archi $m - n + 1 = 4$ esterni a T (tratteggiati) danno i cicli fondamentali, scritti dalla coda dell'arco:

arco	ciclo fondamentale	archi rovesci
(3, 2)	3, 2, 1, 3	(1, 2)
(4, 3)	4, 3, 1, 2, 4	(1, 3)
(5, 4)	5, 4, 2, 1, 3, 5	(2, 4), (1, 2)
(5, 6)	5, 6, 4, 2, 1, 3, 5	(4, 6), (2, 4), (1, 2)

Ad esempio, il ciclo di (3, 2) utilizza (3, 2) in avanti, (1, 2) all'indietro (da 2 a 1) e (1, 3) in avanti.

Un albero è connesso e aciclico; due qualsiasi delle proprietà “connesso”, “aciclico”, “ $n - 1$ lati” implicano la terzo. Aggiungendo ad un albero di copertura un arco esterno si crea esattamente un ciclo, il ciclo fondamentale.

1.5 Il rango della matrice di incidenza

- Sia $G = (N, A)$ un grafo diretto con matrice di incidenza nodi-arco $\mathbf{B}$. Indichiamo con $\mathbf{B}_a$ il colonna dell'arco a , e da $\mathbf{e}_i$ il vettore unitario i -esimo di $\mathbb{R}^n$, in modo che $\mathbf{B}_{(i,j)} = \mathbf{e}_i - \mathbf{e}_j$.

Lemma 1.20: colonne di una foresta

Se gli archi $F \subseteq A$ formano una foresta, le colonne $\mathbf{B}_a$, $a \in F$, sono linearmente indipendenti.

Dimostrazione. Per induzione su $|F|$; per $F = \emptyset$ non c'è nulla da dimostrare. Lascia che $|F| \geq 1$ e $\sum_{a \in F} \lambda_a \mathbf{B}_a = \mathbf{0}$.

1. Un componente della foresta con almeno un arco è un albero con almeno 2 nodi, quindi ha una foglia ℓ (Lemma 1.14); lascia che $a' \in F$ sia l'unico arco di F incidente a ℓ .
2. La riga ℓ della combinazione contiene solo il termine di a' , che è $\pm \lambda_{a'}$; quindi $\lambda_{a'} = 0$.
3. Gli archi $F \setminus \{a'\}$ formano ancora una foresta, e $\sum_{a \in F \setminus \{a'\}} \lambda_a \mathbf{B}_a = \mathbf{0}$: per induzione tutti gli λ_a sono zero. $\square$

Lemma 1.21: colonne di un ciclo

Se gli archi $C \subseteq A$ formano un ciclo, le colonne $\mathbf{B}_a$, $a \in C$, sono linearmente dipendenti.

Dimostrazione. Lascia che il ciclo visiti $v_0, v_1, \dots, v_k = v_0$ lungo gli archi $c_1, c_2, \dots, c_k$ e lascia che $\varepsilon_t = +1$ se c_t è avanti e $\varepsilon_t = -1$ se è indietro. In entrambi i casi $\varepsilon_t \mathbf{B}_{c_t} = \mathbf{e}_{v_{t-1}} - \mathbf{e}_{v_t}$ e la somma dei telescopi:

$$\sum_{t=1}^k \varepsilon_t \mathbf{B}_{c_t} = \mathbf{e}_{v_0} - \mathbf{e}_{v_k} = \mathbf{0}.$$

I coefficienti ε_t non sono zero, quindi le colonne sono linearmente dipendenti. $\square$

Teorema 1.22: rango della matrice di incidenza

Sia $G = (N, A)$ un grafo diretto il cui grafo non orientato sottostante ha componenti connessi k . Poi

$$\text{rank}(\mathbf{B}) = |N| - k.$$

Dimostrazione. Sia $N_1, N_2, \dots, N_k$ gli insiemi di nodi dei componenti e $n = |N|$.

- $\text{rank}(\mathbf{B}) \geq n - k$. Per Corollario 1.18 il grafo ha uno spanning foresta con archi $n - k$, le cui colonne sono linearmente indipendenti per Lemma 1.20.
- $\text{rank}(\mathbf{B}) \leq n - k$. Ogni arco ha entrambi gli estremi nello stesso componente,

quindi per ogni $h \in \{1, 2, \dots, k\}$ le righe dei nodi di N_h si sommano al vettore zero. I vettori k $\mathbf{y}^h$ con $y_i^h = 1$ per $i \in N_h$ e 0 altrimenti soddisfare $(\mathbf{y}^h)' \mathbf{B} = \mathbf{0}'$.

- Questi vettori k hanno supporti disgiunti, quindi sono linearmente indipendenti: lo spazio nullo sinistro di $\mathbf{B}$ ha dimensione almeno k e $\text{rank}(\mathbf{B}) \leq n - k$. $\square$

Corollario 1.23: basi della matrice di incidenza

Lascia che il grafo non orientato sottostante di $G = (N, A)$ sia connesso. Quindi $\text{rank}(\mathbf{B}) = |N| - 1$, la matrice ottenuta da l'eliminazione di una riga qualsiasi di $\mathbf{B}$ ha il rango di riga completo e un insieme di colonne $|N| - 1$ di $\mathbf{B}$ è lineare indipendente se e solo se i suoi archi formano un albero coprente.

Dimostrazione. Il rango è $n - 1$ di Teorema 1.22 con $k = 1$.

- La somma delle righe dà zero, quindi ogni riga è meno la somma delle altre: le rimanenti righe $n - 1$ si estendono sulla riga spazio, che ha dimensione $n - 1$, e sono linearmente indipendenti.
- Se gli archi $n - 1$ formano un albero ricoprente, le loro colonne sono indipendenti da Lemma 1.20. Se le loro colonne sono indipendenti, gli archi non contengono cicli Lemma 1.21: sono aciclici con archi $n - 1$, quindi un albero ricoprente di Teorema 1.17(c). $\square$

Esempio 1.8: il rango sul grafo di riferimento

- Il grafo di riferimento è debolmente connesso, quindi $\text{rank}(\mathbf{B}) = 6 - 1 = 5$; Eliminazione gaussiana sulla matrice di Esempio 1.4 lo conferma.
- Le cinque colonne dello albero ricoprente di Esempio 1.7, a_1, a_2, a_3, a_5, a_7 , sono linearmente indipendente.
- Il ciclo diretto $2, 4, 3, 2$ utilizza $a_3 = (2, 4)$, $a_6 = (4, 3)$, $a_4 = (3, 2)$, tutto avanti e $\mathbf{B}_{a_3} + \mathbf{B}_{a_6} + \mathbf{B}_{a_4} = \mathbf{0}$.
- Senza gli archi $(4, 6)$ e $(5, 6)$ il grafo ha le due componenti $\{1, 2, 3, 4, 5\}$ e $\{6\}$, e il rango della sua matrice di incidenza è $6 - 2 = 4$.

Il rango della matrice di incidenza nodo-arco è il numero di nodi meno il numero di componenti connessi. Nell'a grafo connesso una riga è ridondante e le basi di $\mathbf{B}$ sono esattamente gli albero ricoprente.

1.6 Ricerca su grafi

Definizione 1.24: ricerca grafo

Dato un grafo diretto $G = (N, A)$ e un nodo $s \in N$, **source**, una **graph search** trova l'insieme $R(s)$ dei nodi raggiungibili da s . Durante la ricerca:

- un nodo è **etichettato** quando la ricerca lo raggiunge e rimane etichettato;
- ogni nodo etichettato $j \neq s$ registra il suo **predecessore** $pred(j)$, il nodo da cui è stato raggiunto;
- i nodi etichettati ancora da elaborare vengono mantenuti in una lista, $LIST$;
- un arco (i, j) è **ammissibile** se i è etichettato e j non lo è.

Algoritmo

Algorithm: Ricerca generica nel grafo

Input: un grafo diretto $G = (N, A)$ dato dalle sue liste di adiacenza, una fonte $s \in N$

Output: i nodi etichettati, uguali a $R(s)$, e i predecessori $pred(j)$

- Passaggio 0 (inizializzazione).** Etichetta s e imposta $LIST := \{s\}$; tutti gli altri nodi non sono etichettati.
- Passaggio 1 (selezione).** Se $LIST = \emptyset$, STOP. Altrimenti seleziona un nodo $i \in LIST$:
- il nodo *primo* di $LIST$ nella ricerca in ampiezza (BFS),
- il nodo *ultimo* di $LIST$ nella ricerca approfondita (DFS).
- Passaggio 2 (etichettatura).** Se i ha un arco ammissibile (i, j) , prendi il primo in $L(i)$:
- etichetta j , imposta $pred(j) := i$ e aggiungi j alla fine di $LIST$.
- Altrimenti rimuovere i da $LIST$. Vai al passaggio 1.

- Nel BFS la lista $LIST$ è una **queue** (first in, first out): i nodi vengono elaborati nell'ordine in cui sono etichettati.
- Nel DFS la lista $LIST$ è un **stack** (last in, first out): la ricerca parte sempre dal più nodo recentemente etichettato ancora in $LIST$ e torna indietro solo quando quel nodo non ha archi ammissibili.
- L'ordine degli elenchi di adiacenza decide quale arco ammissibile viene preso per primo; con elenchi ordinati in ordine crescente indice, come in questo modulo, il risultato è unico.

1.6.1 L'albero predecessore

Proposizione 1.25: albero predecessore

Ad ogni iterazione, gli archi $(pred(j), j)$, per i nodi etichettati $j \neq s$, formano un albero sui nodi etichettati, **albero predecessore**. Per ogni nodo etichettato j contiene un percorso diretto da s a j .

Dimostrazione. Sia r il numero di nodi etichettati.

1. Ogni nodo etichettato $j \neq s$ ha un predecessore, etichettato prima di j . A partire da j e passando a predecessore, i nodi sono etichettati sempre prima, quindi la sequenza termina con s .
2. Letto al contrario, è un percorso diretto da s a j lungo gli archi $(pred(h), h)$.
3. Quindi gli archi $r - 1$ collegano i nodi etichettati r e formano un albero per [Teorema 1.17\(b\)](#). $\square$

Teorema 1.26: correttezza della ricerca

Al termine della ricerca generica, BFS e DFS inclusi, i nodi etichettati sono esattamente i nodi di $R(s)$.

Dimostrazione. Ogni nodo etichettato è raggiungibile da s , da [Proposizione 1.25](#). Viceversa, lasciamo $j \in R(s)$ e lascia che $s = i_0, i_1, \dots, i_k = j$ sia un percorso diretto; presupponiamo per assurdo che j non sia etichettato.

1. Sia i_t il primo nodo senza etichetta del percorso; $t \geq 1$, poiché s è etichettato.
2. Il nodo i_{t-1} è etichettato, quindi è entrato in $LIST$, e alla fine lo ha lasciato.
3. Un nodo esce da $LIST$ solo quando non ha arco ammissibile. Ma (i_{t-1}, i_t) è ammissibile fino alla fine, perché le etichette non si tolgono mai: un controsenso. $\square$

Teorema 1.27: tempo di esecuzione della ricerca

Con gli elenchi di adiacenza, la ricerca generica viene eseguita nel tempo $O(|N| + |A|)$.

Dimostrazione. Ogni nodo mantiene un puntatore alla posizione corrente nella sua lista $L(i)$; il puntatore si sposta in avanti oltre l'etica successori e non torna mai indietro, poiché le etichette non vengono mai rimosse.

- Ogni iterazione etichetta un nodo o rimuove un nodo da $LIST$. Ogni nodo viene etichettato al massimo una volta e rimosso a più una volta, quindi ci sono al massimo $2n$ iterazioni, ciascuna con $O(1)$ che funziona oltre al puntatore.
- Il puntatore del nodo i si sposta al massimo $\deg^+(i)$ volte in totale e

$$\sum_{i \in N} \deg^+(i) = m.$$

Il tempo totale è $O(n + m) = O(|N| + |A|)$. □

- Con la matrice di adiacenza i successori di un nodo si trovano scansionando una riga di lunghezza n , e la ricerca richiede tempo $O(n^2)$.

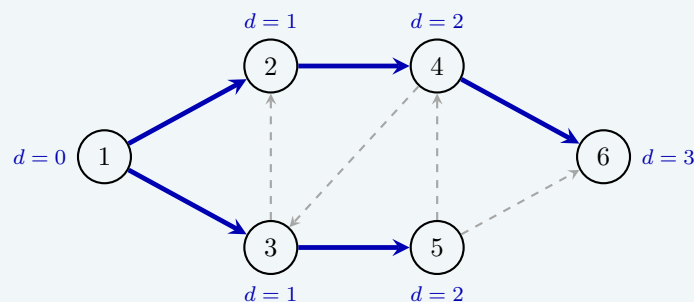
1.6.2 La visita in ampiezza sul grafo di riferimento

Esempio 1.9: BFS dal nodo 1

Eseguiamo il BFS sul grafo di riferimento da $s = 1$, con gli elenchi di adiacenza di [Esempio 1.5](#). Ogni riga rappresenta un'iterazione e dopo di essa viene visualizzato *LIST*:

iterazione e i selezionato e azione	<i>LIST</i>
1	1 etichetta 2, $pred(2) = 1$ 1, 2
2	1 etichetta 3, $pred(3) = 1$ 1, 2, 3
3	1 rimuovi 1 2, 3
4	2 etichetta 4, $pred(4) = 2$ 2, 3, 4
5	2 rimuovi 2 3, 4
6	3 etichetta 5, $pred(5) = 3$ (2 è etichettato) 3, 4, 5
7	3 rimuovi 3 4, 5
8	4 etichetta 6, $pred(6) = 4$ (3 è etichettato) 4, 5, 6
9	4 rimuovi 4 5, 6
10	5 rimuovi 5 (4 e 6 sono etichettati) 6
11	6 rimuovi 6 (vuoto)

L'ordine di etichettatura è 1, 2, 3, 4, 5, 6: ogni nodo è raggiungibile da 1. L'albero precedente ha gli archi (1, 2), (1, 3), (2, 4), (3, 5), (4, 6); è lo albero ricoprente di [Esempio 1.7](#). Accanto a ciascuno nodo, in figura è riportato il numero $d(j)$ di archi del percorso dell'albero da 1 a j .



Proposizione 1.28: BFS trova i percorsi più brevi in numero di archi

Nell'albero precedente del BFS, lasciare $d(s) = 0$ e $d(j) = d(pred(j)) + 1$ per ogni $j \neq s$ etichettato. Quindi, per ogni $j \in R(s)$, $d(j)$ è il numero minimo di archi di un percorso diretto da s a j .

Dimostrazione. Dimostriamo tre fatti.

1. Se $LIST = u_1, u_2, \dots, u_r$, allora $d(u_1) \leq d(u_2) \leq \dots \leq d(u_r) \leq d(u_1) + 1$.
Vale a Passaggio 0. Un nuovo nodo j viene etichettato da u_1 e aggiunto con $d(j) = d(u_1) + 1$; rimuovendo u_1 si mantiene la proprietà, perché $d(u_r) \leq d(u_1) + 1 \leq d(u_2) + 1$.
2. I nodi sono etichettati in ordine non decrescente di d . Ogni nuovo nodo ha $d(j) = d(u_1) + 1 \geq d(u_r)$, e u_r è l'ultimo nodo etichettato prima di j .
3. Per ogni arco (i, j) con i etichettato, $d(j) \leq d(i) + 1$. Se j è etichettato dopo i , allora j è etichettato prima che i lasci $LIST$, da un nodo u nella parte anteriore di $LIST$, etichettato non più tardi di i : $d(j) = d(u) + 1 \leq d(i) + 1$. If j is labelled before i , then $d(j) \leq d(i)$ di fatto 2.

Sia $s = i_0, i_1, \dots, i_k = j$ un percorso diretto con il numero minimo k di archi. I suoi nodi sono etichettati (Teorema 1.26), e di fatto 3, $d(j) \leq d(i_{k-1}) + 1 \leq \dots \leq d(s) + k = k$. L'albero il percorso da s a j è un percorso diretto con archi $d(j)$, quindi $d(j) \geq k$. $\square$

- In Esempio 1.9, i valori $d = (0, 1, 1, 2, 2, 3)$ sono il numero minimo di archi dal nodo 1: ad esempio, i percorsi diretti più brevi al nodo 6 hanno archi 3, come 1, 2, 4, 6.

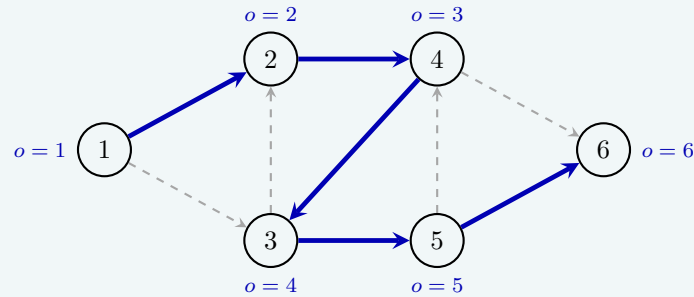
1.6.3 La ricerca approfondita sul grafo di riferimento

Esempio 1.10: DFS dal nodo 1

Eseguiamo il DFS sul grafo di riferimento da $s = 1$: il nodo selezionato è sempre l'ultimo di $LIST$.

iterazione e i selezionato e azione	$LIST$	
1	1	etichetta 2, $pred(2) = 1$
2	2	etichetta 4, $pred(4) = 2$
3	4	etichetta 3, $pred(3) = 4$
4	3	etichetta 5, $pred(5) = 3$ (2 è etichettato)
5	5	etichetta 6, $pred(6) = 5$ (4 è etichettato)
6	6	rimuovi 6
7	5	rimuovi 5
8	3	rimuovi 3
9	4	rimuovi 4 (3 e 6 sono etichettati)
10	2	rimuovi 2
11	1	rimuovi 1 (2 e 3 sono etichettati)

L'ordine di etichettatura è 1, 2, 4, 3, 5, 6 e l'albero precedente ha gli archi (1, 2), (2, 4), (4, 3), (3, 5), (5, 6): un unico percorso diretto. Accanto a ciascun nodo, la figura mostra la sua posizione $o(j)$ nell'ordine di etichettatura.



- Entrambe le ricerche etichettano tutti i nodi, come garantisce [Teorema 1.26](#), ma i loro alberi differire. L'albero DFS raggiunge il nodo 6 con archi 5, mentre l'albero BFS utilizza 3.

1.6.4 Utilizzo della ricerca

- **Componenti connessi.** In un grafo non orientato, una ricerca da un vertice senza etichetta ne etichetta l'intero componente; ripetendolo da ogni vertice ancora non etichettato si trovano tutti i componenti in tempo $O(n + m)$.
- **Componenti fortemente connessi.** Una ricerca in G e una nel grafo invertito G' restituiscono $R(i)$ e $R'(i)$, quindi $S(i) = R(i) \cap R'(i)$; ripeterlo per un nodo di ogni componente ancora sconosciuto richiede $O(n(n + m))$ ora.
- **Spanning tree.** Su un grafo connesso non orientato, l'albero predecessore di una ricerca da qualsiasi vertice è un albero di copertura.

Una ricerca sul grafo etichetta esattamente i nodi raggiungibili dalla sorgente, in tempo $O(|N| + |A|)$, e i suoi predecessori formano un albero. Il BFS (una coda) fornisce percorsi con il numero minimo di archi; il DFS (uno stack) va prima in profondità.

1.7 Errori tipici

Attenzione

- **Scambio dei segni della matrice di incidenza.** La voce è $+1$ in coda e -1 in testa: la colonna di $(4, 3)$ ha $+1$ nella riga 4.
- **Chiamare una passeggiata un percorso.** Un percorso ha vertici distinti; una passeggiata può ripeterli e contiene sempre a percorso con gli stessi punti finali.
- **Confusione tra connettività debole e forte.** Il grafo di riferimento è debolmente connesso, ma ha tre componenti fortemente connessi.
- **Conteggio solo degli lati.** Un grafo con lati $n - 1$ non deve essere un albero: deve anche essere connesso, oppure aciclico.
- **Credere che la matrice di incidenza abbia rango di riga completo.** Il suo rango è $|N| - k$: in un grafo connesso una riga è sempre ridondante.

- **Percorsi più brevi previsti dal DFS.** Solo l'albero BFS fornisce il numero minimo di archi e solo quando ogni arco conta uno.
- **Dimenticare l'ordine degli elenchi.** L'ordine di etichettatura e l'albero precedente dipendono dall'ordine in cui vengono scansionati i successori.

1.8 In sintesi

In sintesi

- Un grafo non orientato $G = (V, E)$ ha archi $\{u, v\}$, un grafo diretto $G = (N, A)$ ha archi (i, j) ; i gradi sommano a $2m$, i gradi in uscita e in entrata a m .
- Le liste di adiacenza necessitano di spazio $n + m$; la matrice di incidenza nodo-arco ha $+1$ in coda e -1 in testa, e la riga i di $\mathbf{Bx}$ è il deflusso meno l'afflusso del nodo i .
- Le componenti connesse e fortemente connesse sono classi di equivalenza; $S(i) = R(i) \cap R'(i)$.
- Un albero è connesso e aciclico, con lati $n - 1$ e un percorso unico tra ogni due vertici; un lato esterno uno albero ricoprente chiude un ciclo fondamentale.
- $\text{rank}(\mathbf{B}) = |N| - k$, e in un grafo connesso le basi di $\mathbf{B}$ sono gli albero ricoprente.
- La ricerca generica trova $R(s)$ nel tempo $O(|N| + |A|)$; il BFS fornisce il numero minimo di archi.

Il grafo di riferimento in sintesi.

nodi e archi	$n = 6, m = 9$
componenti collegati	1
componenti fortemente collegati	$\{1\}, \{2, 3, 4, 5\}, \{6\}$
$\text{rank}(\mathbf{B})$	5
BFS da 1 e ordina 1, 2, 3, 4, 5, 6; $d = (0, 1, 1, 2, 2, 3)$	
DFS da 1 e ordina 1, 2, 4, 3, 5, 6	