30/04/2024 | LEZ 29

APPLICAZIONI DEL PROBLEMA DI MAX FLOW

VEDREMO 4 PROBLEMI

1° BIPARTITE MATCHING

DEF

SIA $G=(V,E)$ GRAFO ∅ ORIENTED:

- MATCHING M ($\subseteq E$) t.c. ∀ NODO CHE APPARE IN M = ATTACCATO DA 1 SOLO ARCO.
- MATCHING PERFETTO SE SU M CI SONO TUTTI I NODI

PROBLEMA = ∃ MATCHING ∪ MAX $|M|$

X RENDERLO + EZ LO APPLICHIAMO AD UN PARTICOLARE GRAFO.

GRAFO BIPARTITO ⟹ GRAFO / IN 2 SET DI NODI A e B DOVE ∀ NODO DI A HA ARCHI SOLO VERSO NODI DI B (E VICEVERSA)

NUOVO PROBLEMA = CONTROL, SE M È PERFECT.

COME SFRUTTARE IL MAX FLOW?

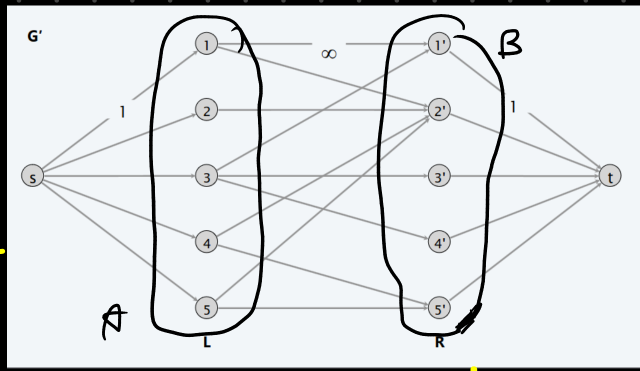
TRANSFORMO IL PROBLEMA A MIO VANTAGGIO.

RIFORMULAZIONE

1. NEW G', DIRETTO, = G CON ADD 2 NODI: s e t
2. DIREZIONE: DA A e B → [s → A → B → t]

③ e TRA A → B → CAPACITÀ $= \infty$

④ e // s → A / B → t → CAPACITÀ $= 1$



TEOREMA ⇝ CHE SARÀ IL TRUCCO

∃ BIEZIONE TRA

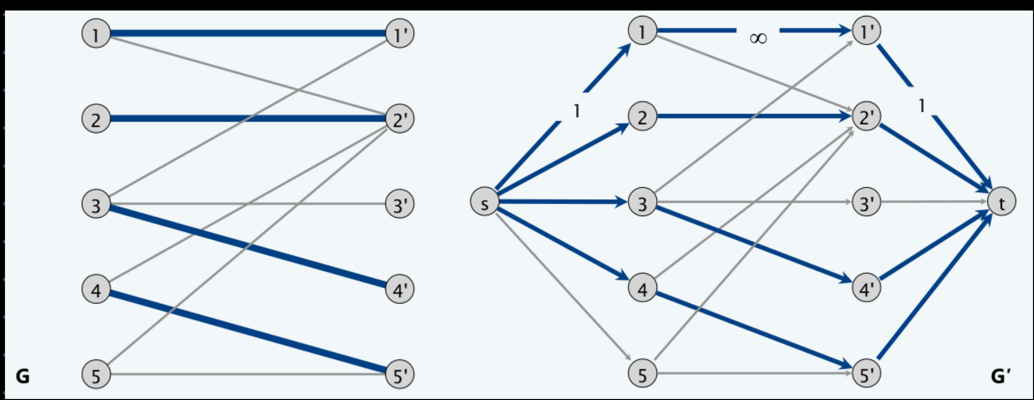
FLOW $f = k$ ⟷ MATCHING M CON $|M| = k$

DIM

1° PARTE (→):

SIA M di $|M| = k$:

- CONSIDERO UN f CHE ∀ ARCO ∈ M. MANDA 1 UNITÀ DI FLUSSO.
- f è FLOW DI VAL k



2° PARTE (←)

SIA f FLOW $= k$ e SIA M:

- ∀ NODO IN M APPARE IN AL PIÙ 1 ARCO ∈ M.
- X LEMMA FLOW-VALUE [CON CUT $(S \cup A, t \cup B)$]

$$\mathrm{VAL}\, F(x) = \sum_{e \text{ out } A} f(e) - \sum_{e \text{ in } A} f(e) = k$$

QUINDI:

DA G', ? $MAX(VAL(F)) = MAX(|M|)$.

COSTO

USANDO FORD-FULKERSON

$O(m f^*) \rightarrow f^* \leq n \rightarrow$ ESSENDO OGNI ARCO 1 UNITÀ DI CAP e n PUÒ ESSERE COLL. AD AL PIÙ $n-1$ ARCHI

TIPO SINGOLO AUGMENTATION

$\Downarrow$

$O(m\, n)$

2° APPLICAZIONE: ARCHI DISGIUNTI

SIA G GRAFO DIRETTO e 2 NODI $\in G$, s e t

DEF: 2 CAMMINI SONO DISGIUNTI IF NON HANNO ARCHI IN COMUNE

PROBLEMA: ? MAX NUM. DI PATH DISGIUNTI.

RIFORMULAZIONE MAX-FLOW

- DA G ASSEGNO $\forall e \rightarrow c(e) = 1$

BIEZIONE TRA:

f VALORE $k \iff k$ CAMMINI DISGIUNTI

DIM

1° PARTE ($\Rightarrow$):

- SIANO $P_1, \dots, P_k$, K CAMMINI EDGE-DISJ. IN G

- SETTIAMO
$$F(e) = \begin{cases} 1 & e \in \text{GENERICO } P_J \\ 0 & \text{ELSE} \end{cases}$$

- P SONO DISGIU → RISPETTA REGOLA OF $C(e)=1$; RISPETTA PROP. CONSERVAZIONE FLUSSO

$\Downarrow$

$f = K$

2° PARTE ($\Leftarrow$)

~ SIA f = FLOW VAL. K
~ SIA $f(u,v) = 1$

· ALLORA, X FLOW CONSERVA. $\leadsto \exists (v,h) \to f(v,h)=1$

~ CONTINUO FINO AD ARRIVARE t
~ RIPETO U NUOVO EDGE
~ CREO K PATH NON X FORZA SEMPLICI

AA QUI

POSSO RISOLVERE PROBLEM, POSSO USARE UNA FORMULAZIONE BY MAX-FLOW. ↳ USING. FORD

TIME = $O(n \cdot m)$

WHY?

~ F RIMANE INTERO
~ F^* MAX = MAX N. OF PATH.

3° APPLICAZIONE: ELABORAZIONE IMG.

PROBLEMA: / IMMAGINI IN REGIONI COERENTI.

DEF. PROBL. BACKGROUND / FOREGROUND SEGMENTATION.
(OGG. DESIGNATO)

- OGNI PIXEL VIENE ASSOCIATO O IN → BACKG.
  ↳ FOREG.
- V = SET PIXEL / E = PAIO DI PIXEL VICINI
  i IN PIXEL i
- $a_i \geq 0$ ⇝ PROB. DI ASSOCI. i IN FOREGROUND
- $b_i \geq 0$ ⇝ // // // BACKGROUND.
- $P_{ij} > 0$ ⇝ COSTO IN ASSOCIARE 1 IN FORE. E OTHER IN BACK.

GOAL:

- PRECISIONE ⇝ IF $a_i > b_i$ → SCELTA OVVIA
- FLUIDITÀ ⇝ IF MANY NEIGH. ARE IN F./B. ALLORA ANCHE i VA IN F./B.
- ∃ PARTITION (A, B) CHE MAX:
  ↓ FOR. ↘ BACK.

$$: \sum_{i \in A} a_i + \sum_{j \in B} b_j - \sum_{\substack{(i,j) \in E \\ |A \cap \{i,j\}| = 1}} P_{ij}$$

↳ 1 DEI DUE IN A

MIN-CUT TRANSFORM

① MINIMIZATION

- MAXIMIZING:

$$\sum_{i\in A} a_i + \sum_{j\in B} b_j - \sum_{\substack{(i,j)\in E \\ |A\cap\{i,j\}|=1}} p_{ij}$$

- È UGUALE A MINIMIZZARE IL COMPLEMENTARE:

$$\left(\sum_{i\in V} a_i + \sum_{j\in V} b_j\right) - \sum_{i\in A} a_i - \sum_{j\in B} b_j + \sum_{\substack{(i,j)\in E \\ |A\cap\{i,j\}|=1}} p_{ij} =$$

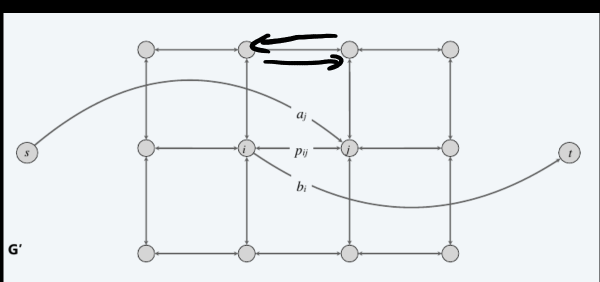
$$= \boxed{\sum_{i\in B} a_i + \sum_{j\in A} b_j + \sum_{\substack{(i,j)\in E \\ |A\cap\{i,j\}|=1}} p_{ij}}$$

② FORMAZIONE GRAFO

FOREGRO. BACKGROU.

- FORMULAZIONE PROB-MIN-CUT ⇝ ~ $G'=(V',E')$ → $V' = \text{PIXEL} = \text{NODI} + \{s,t\}$
  → E = ~ e tra pixel → 2 archi ⇄
  ~ ARCHI s ⇝ p_i
  ~ ARCHI p_i ~ t

ESEMPIO



③ FLUSSO

- SIA CUT (A,B) IN G' DOVE A = FOREGROUND
- ALLORA $CAP(A,B) = \sum_{i \in B} a_i + \sum_{j \in A} b_j + \sum_{\substack{(i,j) \in E \\ i \in B,\, j \in A}} p_{ij}$

↓

OBBIETTIVO DA MINIMIZZARE.

4° APPLICAZ. : BASEBALL ELIMINATION

QUESTION

i		team	wins	losses	to play	ATL	PHI	NYM	MON
0		Atlanta	83	71	8	–	1	6	1
1		Philly	80	79	3	1	–	0	2
2		New York	78	78	6	6	0	–	0
3		Montreal	77	82	3	1	2	0	–

CHI HA + CHANCE DI FINIRE SEASON U + WINS?

CAPIRE CHI HA CHANCE, DIPENDE DA
~ QUANTE VITTORIE HANNO
~ CHI GIOCANO CONTRO

FORMUL. PROBLEMA

- S = SET TEAMS
- $z \in S \rightarrow$ TEAM DISTINTO
- TEAM x HA VINTO w_x GIOCHI
- TEAM x e y GIOCANO TRA DI LORO r_{xy} VOLTE

Q: CON QUESTI DATI, POSSIAMO CAPIRE SE, UNA SQUADRA Z PUÒ FINIRE U T WIN?

FORMUL. MAX-FLOW

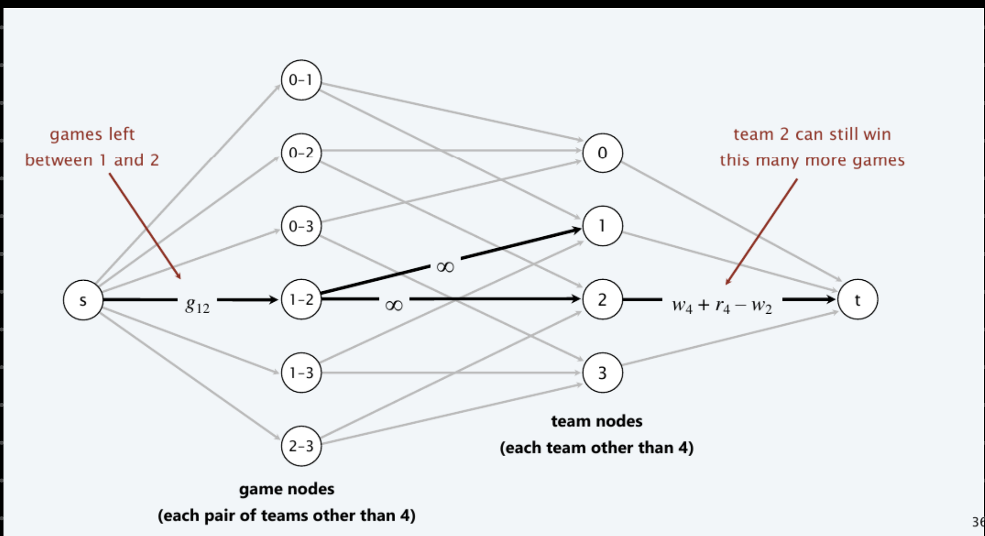
SIA $S = \{0,1,2,3,4\}$ $z=4$.

- ASSUMIAMO CHE:

  Z HA VINTO ALL REMAIN PLAY → $w_4 + r_4$ WINS

- QUINDI:
  - U PLAY REMAINED, ALL OTHER DEVONO FARE:

    WIN $\leq w_4 + r_4$



STRUTTURA PROBLEMA

DA QUI:

- 4 NOT ELIMINATO SE F SATURA TUTTI ARCHI USCENTI DA 2.
- DOVE SE ANCHE 1 (x,t) DI ESSI CON VIENE RIEMPITO, $C((x,t))=0 \rightsquigarrow$ Z NON HA POSSIBILITÀ.