17 April 2023
MAT 2250

Theorem 8.9 says that a matching $G = (V, E)$ is maximal iff there is no M-alternating path.

So the idea is that if we want to find a maximal matching, then we can start from ϕ, gradually add more edges if there is still alternating path.

How can we do it in practice?

3 basic questions:

Q1: Is $m(G) = |S|$?

Q2: $G = (V = S + T, E)$ bipartite graph, $w_1 : S \to R_{\geq 0}$ weight for vertices in S.

Find a matching M with maximal weights: $w_1(M) = \sum_{s \geq v : a \text{ vertex in } M} w_1(v)$

Q3: (Similar to Q2, but the weights are for edges): $G = (V = S + T, E)$, $w_2 : E \to R_{\geq 0}$, find a matching M with maximal weight $w_2(M) = \sum_{k \in M} w_2(k)$

Q1 follows from Q2 or Q3

(define $w_1(x) = 1 \quad \forall x \in S$
$w_2(k) = 1 \quad \forall k \in E$)

So we only need to work with Q2 & Q3. (Forget about Algorithm 8.10 in the book)

First, consider Q2.

Definition (Transversal subset of S):
$X \subseteq S$ (can be ϕ) is transversal if $\exists$ a matching M of G such that $X \subseteq V(M)$ = the set of vertices of M. $\mathcal{T}$ = the set of all transversals.

Thm 8.11: (Solution to Q2):
$G = (V = S + T, E)$ bipartite graph
$\mathcal{T}$ = the set of transversals.
Then $(S, \mathcal{T})$ forms a matroid.

Hence, Q2 can be solved by finding a maximal basis for $(S, \mathcal{T})$ (use Thm 7.13, but with minimal replaced by maximal).

Proof:

Need to check 3 axioms:

(1) $\phi \in \mathcal{T}$

(2) $A \in \mathcal{T}$, $B \subseteq A \Rightarrow B \in \mathcal{T}$

(3) $A, B \in \mathcal{T}$, $|B| = |A| + 1$
$\Rightarrow \exists\, v \in B \setminus A$ such that $A \cup \{v\} \in \mathcal{T}$.

We only give a proof for (3).
Let M_A be a matching with $A \subseteq V(M_A)$
M_B be a matching with $B \subseteq V(M_B)$
From the proof of Thm 8.9, we can find an M_A-alternating path

$P = u_0 \to \cdots \to u_j \to u_{j+1}$ is odd in S.

& u_0 or u_{j+1} is

(& by definition u_0, u_{j+1} are M_A-unsaturated)

We can add 1 more edge from M_B into M_A. (Q.E.D.)

For Q3: We use the idea from the end of § 8.1.

First, some reductions:

Reduction 1: We may assume that $|S| = |T| = n$. (e.g. if $|S| < |T|$, add more vertices into S.)

Reduction 2: Can assume that for every $u \in S$ & every $v \in T$, there is an edge uv. (If not, then add an edge with weight $= 0$.). $G = K_{n,n}$ = complete bipartite (n,n) graph.

Reduction 3: Instead of maximum weight problem, we can do minimum weight problem, with the requirement that the matching M has $|M| = n$. (Basically, if $|M| < n$ $\Rightarrow$ add some more edges with weight 0 into M & still have a matching).

Take any $\alpha \geq \max_{k \in E} w(k)$
& choose a new weight:
$w'(k) = \alpha - w(k)$

Q3': $G = K_{n,n}$

Find a matching M with $|M| = n$ & minimum weight.

Reduction to Linear Optimization (a special case of what's considered in Chapter 15 in the book): $\downarrow$ Linear Programming

Input: $S = \{s_1, \ldots, s_n\}$
$T = \{t_1, \ldots, t_n\}$
$W: S \times T \to \mathbb{R}_{\geq 0}$ (weight)
$w_{ij} = w(s_i, t_j)$
W is an $n \times n$ matrix with non-neg entries.

Output:
$x = (x_{ij})$ a permutation matrix
such that
$\sum w_{ij} x_{ij} \to \min.$

Claim: A matching M in $K_{n,n}$ $\Leftrightarrow$ a permutation matrix with $|M| = n$.

Proof: A matching M with $|M| = n$ is a way to choose a set of n edges in $K_{n,n}$, each pair by no common vertices. Define $x_{ij} = 1$ if $s_i t_j$ is an edge in M, $= 0$ otherwise, then $x = (x_{ij})_{i,j}$ is a permutation matrix. (Q.E.D.)

Algorithm 8.12:

Step 0:
Subtract each row of the matrix W by the minimum value on that row.
Subtract each column ——— minimum value on that column.

Say
$$W = \begin{pmatrix} 9 & 11 & 12 & 11 \\ 6 & 3 & 8 & 5 \\ 7 & 6 & 13 & 11 \\ 9 & 10 & 10 & 7 \end{pmatrix}$$

After subtracting on rows:
$$\Rightarrow W_1 = \begin{pmatrix} 0 & 2 & 3 & 2 \\ 3 & 0 & 5 & 2 \\ 1 & 0 & 7 & 5 \\ 2 & 3 & 3 & 0 \end{pmatrix}$$

After subtracting on columns:
$$\Rightarrow W_2 = \begin{pmatrix} 0 & 2 & 2 & 2 \\ 3 & 0 & 2 & 2 \\ 1 & 0 & 4 & 5 \\ 2 & 3 & 0 & 0 \end{pmatrix}$$

Step 1: Find [minimal zero-covering]

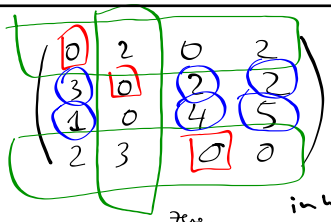
$W \to$ get graph $G_W = (S + T, E)$

$S =$ rows
$T =$ columns
if $w_{ij} = 0 \Rightarrow (i,j)$ is an edge in E

A zero-covering for W is a diagonal whose entries are 0, minimal zero-covering is a ——— whose length is ———.
For W in previous Step a minimal covering is $(1,1), (3,2), (4,3)$.

Step 2: If the cardinality of the minimal zero-covers is n, then we are done because of Thm 8.5 (# max matching = min vertex cover), we have a matching with length n & weight $= 0$.

If not, then we can transform W into a new W' with the cardinality of the minimal zero-covers for W' is $>$ that for W. So after at most n steps we will stop.



We know a zero-diagonal ($\Rightarrow$ matching of G_W. By Theorem 8.5 $\Rightarrow$ $\exists$ a vertex cover D for G_W, s.t. $|D| < n$.
(like, s_1, s_4, t_2)

Step 3: Let $\alpha = \min w_{ij}$ in W
$w_{ij} \notin$ the vertex cover D.
(the blue ones)

(in this case $\alpha = 1$)

Subtract α from the rows not covered in D.

Add α from the columns covered in D.

(We want non-negative positive numbers).

$\alpha = 1$

$$\begin{pmatrix} 0 & 2 & 0 & 2 \\ 3 & 0 & 2 & 2 \\ 1 & 0 & 4 & 5 \\ 2 & 3 & 0 & 0 \end{pmatrix} \Rightarrow \begin{pmatrix} 0 & 2 & 0 & 2 \\ 2 & -1 & 1 & 1 \\ 0 & -1 & 3 & 4 \\ 2 & 3 & 0 & 0 \end{pmatrix}$$

$+\alpha$

$\Rightarrow$ one more zero $\rightarrow$

$$\begin{pmatrix} 0 & 3 & 0 & 2 \\ 2 & 0 & 1 & 1 \\ 0 & 0 & 3 & 4 \\ 2 & 4 & 0 & 0 \end{pmatrix}$$

Now: $(3,1)$, $(2,2)$, $(1,3)$, $(4,4)$ is a zero diagonal, with length $n = 4 \Rightarrow$ we are done.

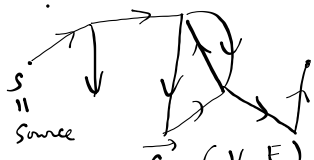
Lemma: If we do step 3, then for W': we have weight $(W') <$ weight (W).

& the change of weight is an integer, hence we cannot go infinitely many times & must stop eventually.

Proof: W' has: ① integer coefficients (because every time we subtract/add a minimum of integer numbers). ② non-negative numbers (look at columns in D & not in D separately). ③ weight of W' < that of W (# of rows not in D is more than # of columns in D.) Q.E.D. □

§8.3 Flows

A flow network: A directed graph G with a source s & a terminal t



$\vec{G} = (V, E)$

Want to go from s to t subject to some constraints, which is defined by capacity $c : E \to \mathbb{N}_{>0}$.

(For simplicity, here $c \in \mathbb{N}$, but we can generalise to $\mathbb{R}_{>0}$).