Shortest path

Max Flow　　　integral data

Min Cost Flow

We wanted integral optimal sol'ns.

IPs, polynomial time

- Integer programming is NP-hard, but for these problems, we can find an integral opt. sol'n in poly time.

- T

$\max \{ f(s) \}$

$f(e) \leq u(e)$

flow in = flow out

~~$f(e)$ integral~~

$f(e) \geq 0$

solved the LP

$\Downarrow$

you'd get an integral sol'n

(basic feasible sol'ns are integer valued)

Unimodular

Apr 21-4:22 PM

Def A matrix A is <u>unimodular</u> if every basis matrix B of A has $\det(B) = \pm 1$

(Basis matrix is a $p \times p$ submatrix w/ linearly ind. columns)

Thm Let A be an integer matrix w/ linearly independent rows. Then the following 3 conditions are equivalent [2]

1) A is unimodular
2) Every basic feasible solution $Ax = b$, $x \geq 0$ is integral for any integer vector b
3) Every basis matrix B has an integer inverse B^{-1}

$Ax = b$

A: graph matrix

b: capacities (supplies, demands)

Totally unimodular (TU) A is TU if each square submatrix has determinant $= -1, 0,$ or $+1$.

TU $\Rightarrow$ unimodular

Thm The node-arc incidence matrix of a directed network is TU.

Pf

$$\begin{array}{cc} & \begin{array}{cccccc} (v,w) & & \text{arcs} & & & \end{array} \\ \begin{array}{c} A = \\ \text{nodes} \end{array} \begin{array}{c} v \\ \\ w \\ \\ \end{array} & \begin{bmatrix} -1 & -1 & +1 & & -1 & \dots \\ & & & -1 & +1 & \\ +1 & & & & & \\ & +1 & & & & \end{bmatrix} \end{array}$$

$A \cdot f = 0$

$A \cdot f = \begin{pmatrix} \\ \\ \end{pmatrix}$ supply dem vector

By induction on k, the size of submatrix

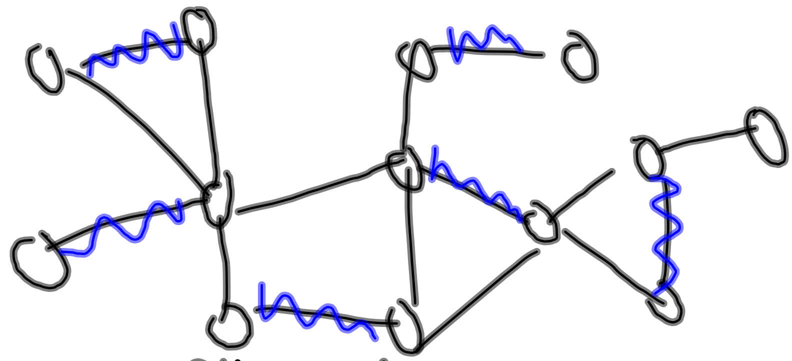
$k = 1$ ✓

Inductive step, assume for $k \times k$ submatrices

$$\begin{array}{cc} & \text{arc} \\ \text{arc} & \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix} \end{array} \begin{bmatrix} f \end{bmatrix} \le \begin{bmatrix} u \end{bmatrix}$$

$$\begin{array}{cc} & \text{arc} \\ \text{node} & \begin{bmatrix} \; \end{bmatrix} \\ & \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \end{array}$$

Non-bipartite matching



$$x_{ij} = \begin{cases} 1 & \text{if } (i,j) \in M \\ 0 & \text{o.w} \end{cases}$$

not TU.

$$\max \sum_{(i,j)\in E} x_{ij}$$

$$\text{s.t.} \sum_{(i,j)\in E} x_{ij} \le 1 \quad \forall i \in V$$

$$x_{ij} \in \{0,1\}$$

Consider F by $(k+1)\times(k+1)$ submatrix

3 cases

1) F has a column of all 0's
2) Each column of F has exactly 2 non-zeros
3) Some column has exactly one non-zero

1) $\det(F) = 0$

2) each column has $+1$ -1
sum all the rows, you get $0, 0, \dots 0$
$\Rightarrow$ rows are not lin ind.
$\Rightarrow \det(F) = 0$

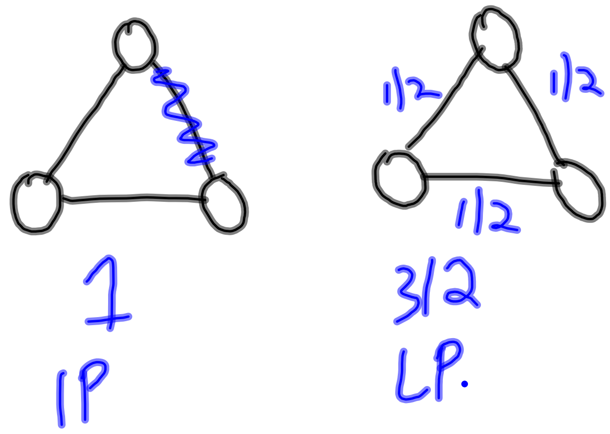
3) $\begin{bmatrix} \pm 1 & & & \\ & 1 & & 1 \\ & & -1 & -1 \\ & -1 & 1 & \end{bmatrix}$

remove the col & row w/ the single ± 1 from F to obtain F', a $k \times k$ submatrix
$\det(F') = 0, -1, +1$
$\det(F) = \pm \det(F') = 0, -1, +1$

Find a graph for which opt. integer sol'n
$\neq$ opt. non-integer sol'n.

max $\sum x_{ij}$

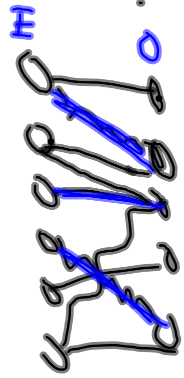
$\sum x_{ij} \le 1$



not a flow problem

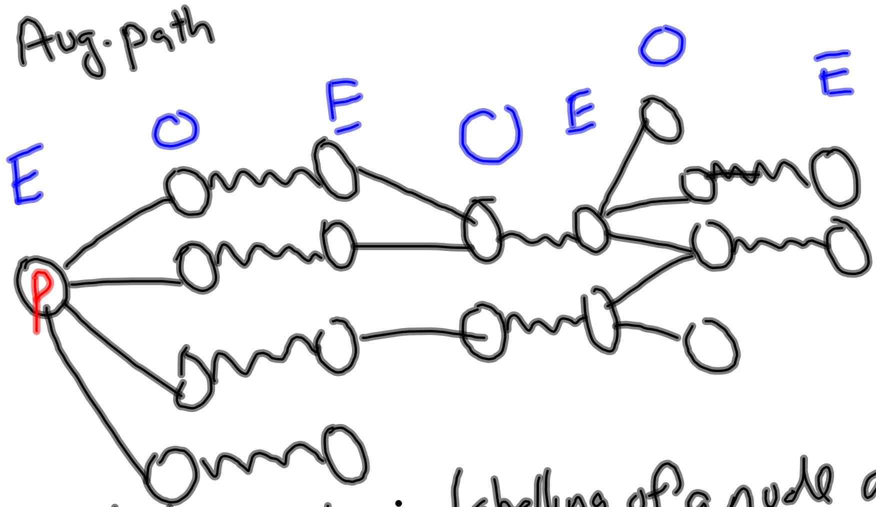
Develop a poly time alg. for matching.

Follow non-bipartite alg but fix it where it breaks.



Bipartite case:

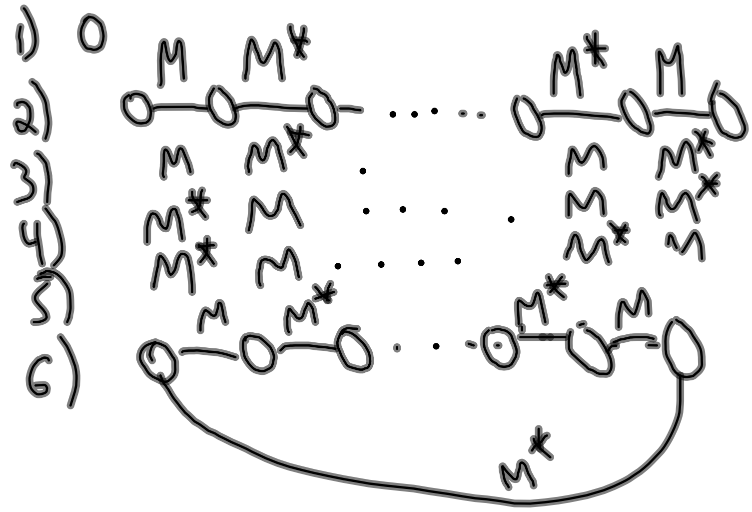
Aug. path



unique label property: labelling of a node as even/odd is independent of the particular search/paths

an aug. path is an alt path starting at a free vertex, ending at a free vertex, end is labelled odd.

Lemma Consider two matchings M, M^*, let $A = M \oplus M^*$. Connected components of A can be of one of six types

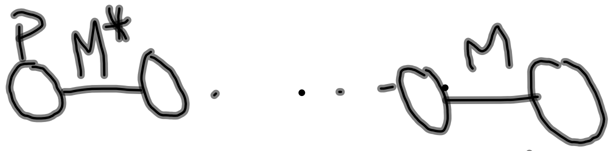


Lemma (Aug. path) If p is unmatched in a matchr M, and there is no aug. path starting at p, then p is unmatched in some maximum matching.

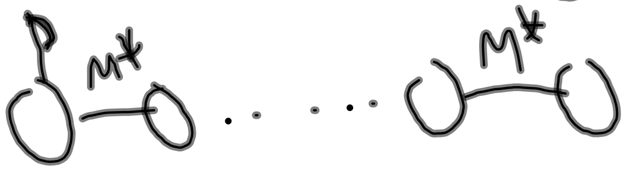
Pf M, consider maximum matching M^*.
If p is unmatched in M^*, we're done.
If not, consider $M \oplus M^*$

Apr 21-5:19 PM

p is unmatched in M, matched in M*



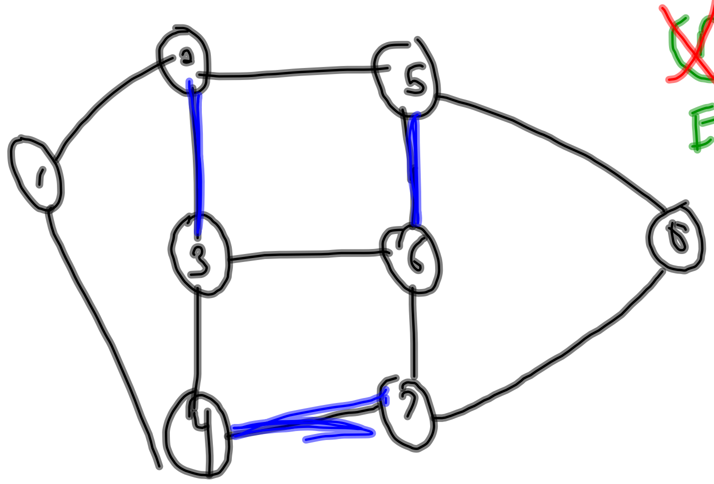
← use M instead of M*

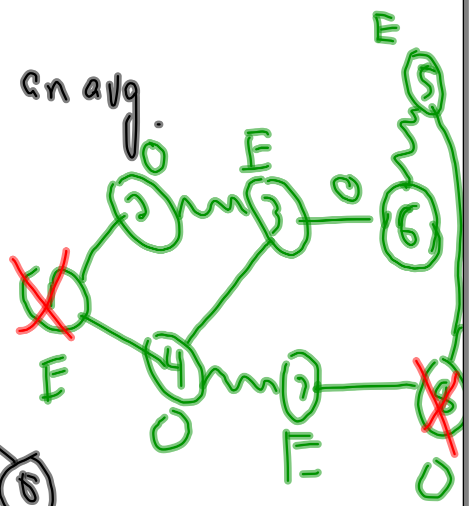


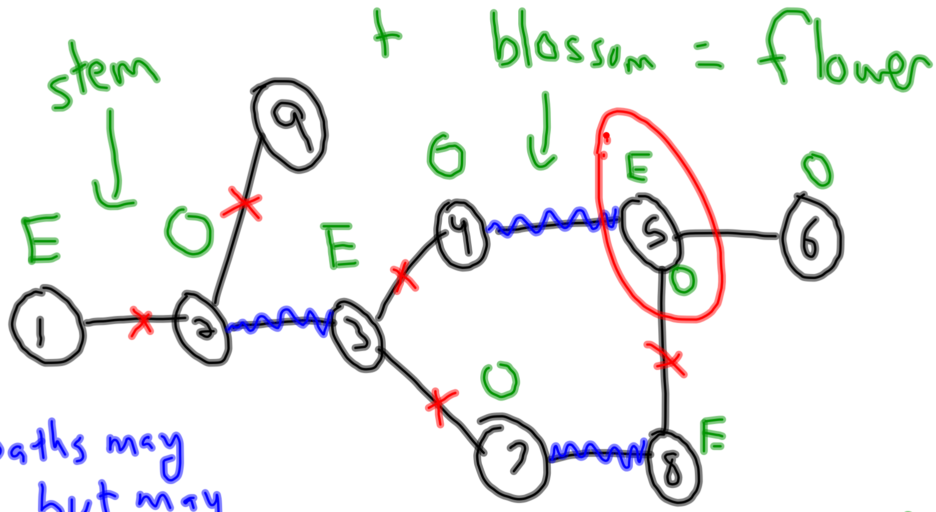
← there is an aug. path from p, w/r/t M.

$\boxtimes$

unique label prop ⇒ we can find an aug. path, if one exists







- aug. paths may exist, but may be hard to find

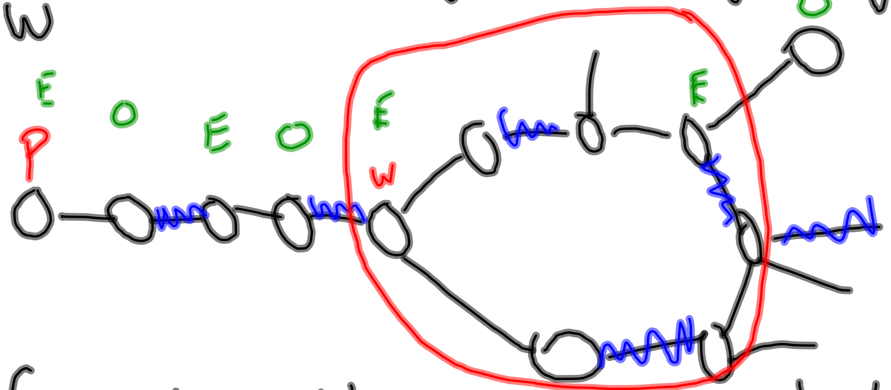
1, 2, $\underbrace{3, 7, 8, 5, 4, 3}$, 2, 9

1, 2, 9

This is the "only" complication.

Stem = even length alt. path starting at root p, ending at w (p=w is possible)

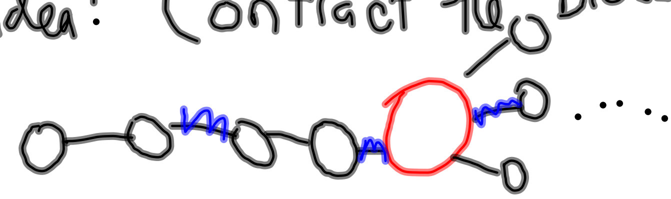
blossom odd length alt. cycle starting & ending at w



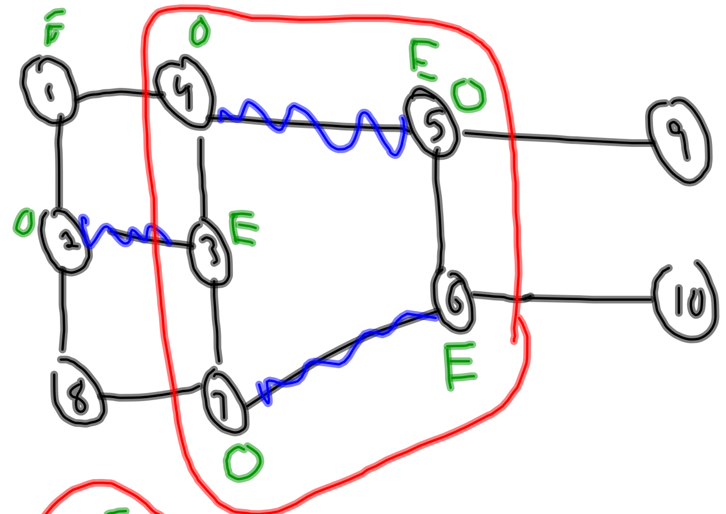
Claim Every node in a blossom is reachable by both an odd and even length alt. path

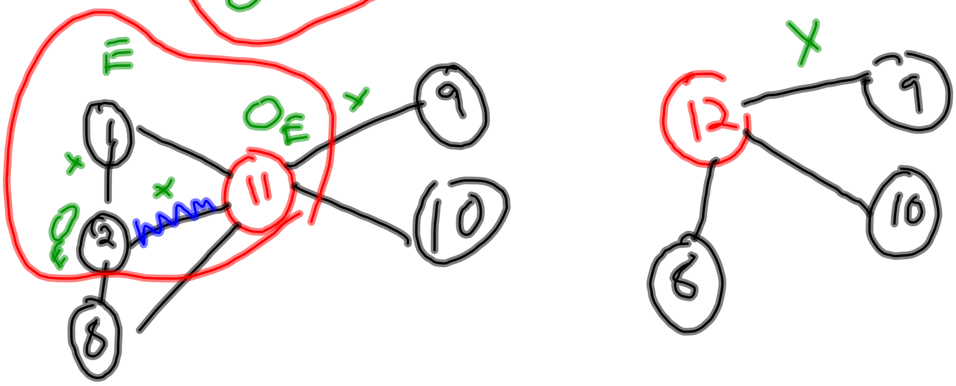
Idea: label the whole blossom as "even"

Impl. of idea: Contract the blossom



contract (v_1, v_2) $\cong$ replace v_1 & v_2 by a supernode v' where v' where v' has an edge to any neighbor of v_1 or v_2





12 · 9 Aug path

1-2-11-9

1-2-3-4-5-9

12 · 8

1-2-11-8

1-2-3-4-5-6-7

-8

10 · 8

1-2-10-8

1-2-3-7-6-8

Correctness of alg.

G^c (contracted graph) after one blossom

1) Aug path in G^c $\Rightarrow$ Aug. path in G
2) Aug path in G $\Rightarrow$ Aug path in G^c

1) to get an aug path in G, take aug. path in G^c & go around the cycle in appropriate direction

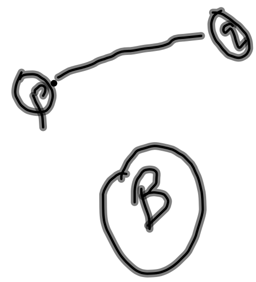
Apr 23-5:01 PM

To show

2) if G has an aug. path P from $p \to q$
the G^c has an aug path from $p \to q$

Assume p & q are the only unmatched nodes in G
Let B be the blossom

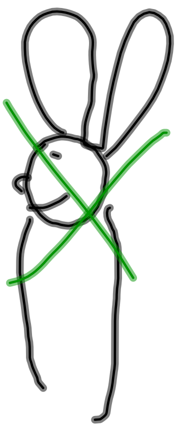
if $P \cap B = \emptyset$ then easy

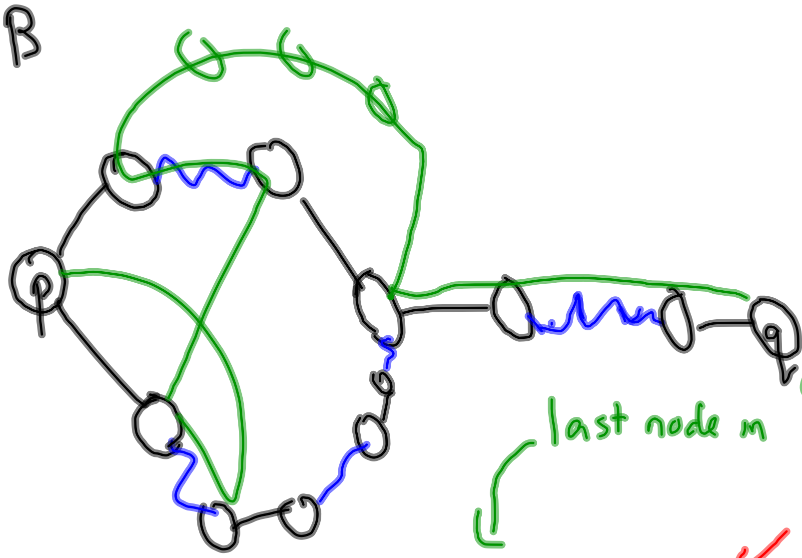


we consider $P \cap B \neq \emptyset$
2 subcases a) node p is in B
b) " " not in B

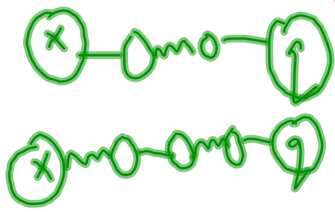
Apr 23-5:06 PM

4) p in B



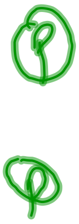


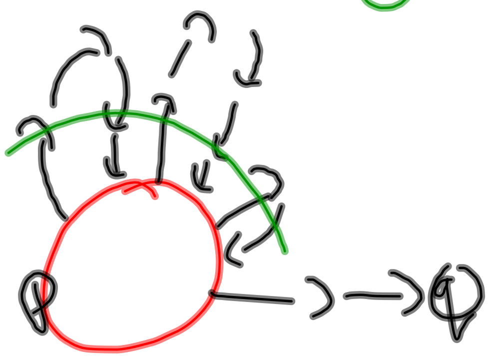
last node in P in B

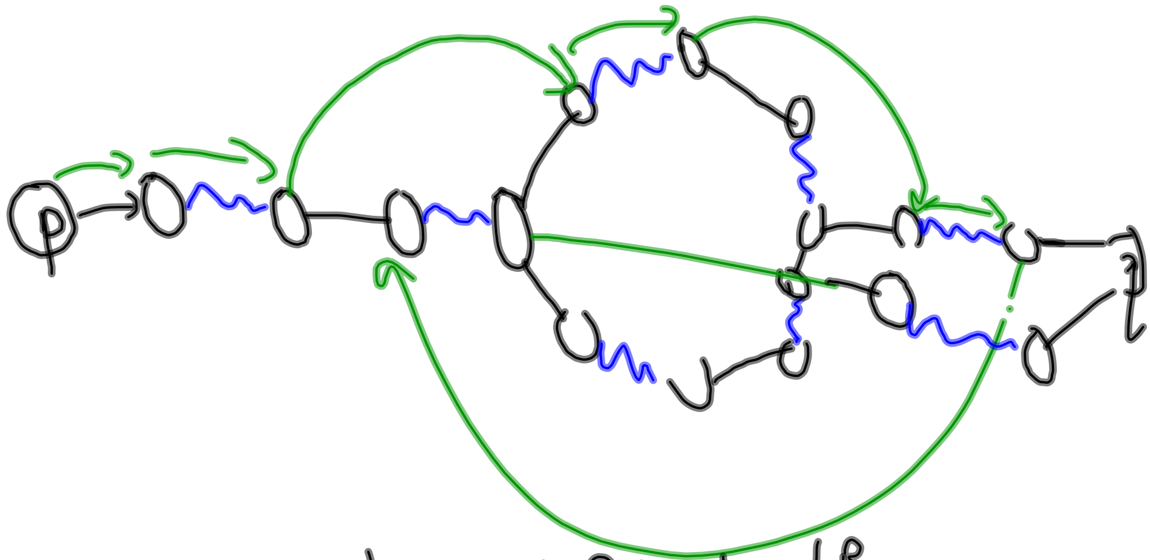


p & x are the same node in G^c

can't happen because it leaves B on a matched edge.







form a path $p \to$ first int. w/ B

$\to$ last int. w/ B

$\to$ to q.

Polynomial time

n iterations (find. aug. path)

Search for an aug path.

if find blossum

contract

n times

} $O(m)$ time to search

$O(n^2 m)$ alg.

$O(\sqrt{n}\,m)$ is possible.