9/17/14

All pairs shortest path problem.

Given a matrix $D = \| l_{ij} \|$ of edge lengths

Find a shortest path between every pair of vertices - assume no negative cycles. n vertices

Floyd's Algorithm

Initialise : $d_{ij} := l_{ij} \quad \forall i, j$

For $k = 1$ to n do

  For $i = 1$ to n do

    For $j = 1$ to n do

      $d_{ij} = \min\{ d_{ij}, d_{ik} + d_{kj} \}$

od od od

Claim: after k outer loop steps d_{ij} is the minimum length of a walk (path) from i to j, all of whose interior vertices are in $\{1, 2, \ldots, k\}$

Proof of Claim: by induction on k.

$k=0$: d_{ij} = length of edge (i,j)

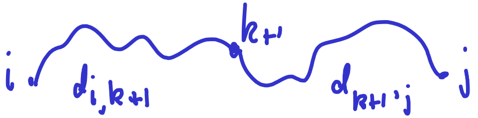
Suppose true for some $k \geq 0$.

Consider the walks from i to j that use $\{1, 2, \dots, k+1\}$ inside.

(a) Walks that do not use $k+1$; (b) Walks that use $k+1$

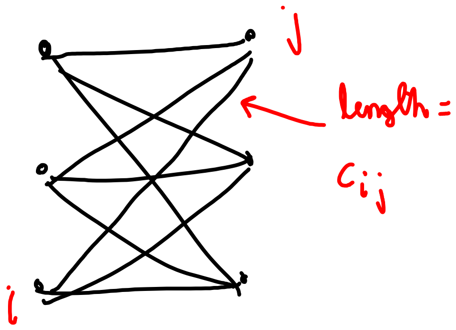
Can assume we only use $k+1$ once.

Min length $= d_{ij}$



Assigment Problem

n people & n tasks : c_{ij} = cost of assigning person i to task j.

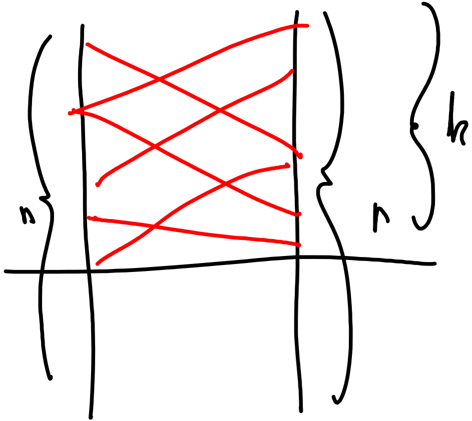


Problem: assign person i to task $\pi(i)$ s.t.

(i) Each task has one assigned person

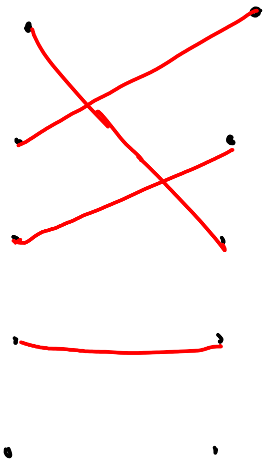
(ii) The sum $\sum_i c_{i,\pi(i)}$ is minimised

Suppose we have solved the $k \times k$ problem



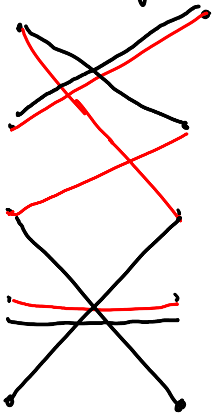
Given this solution, I reduce finding solution to the $(k+1) \times (k+1)$ problem to a shortest path problem.

$k=4$

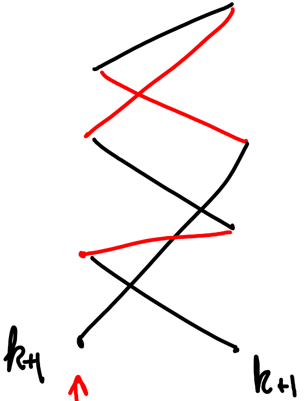


Matching M

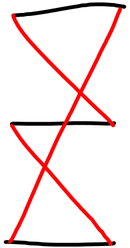
Let M' be any $(k+1) \times (k+1)$ matching



Look at $M \oplus M' = (M \setminus M') \cup (M' \setminus M)$



plus



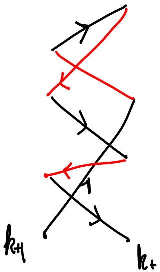
plus more cycles

Removing 

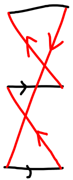
only vertices of degree 1 in $M \oplus M'$: others have degree 0 or 2

Orient all edges: M from right to left; rest from left to right

length $= -$ cost length $=$ cost



plus



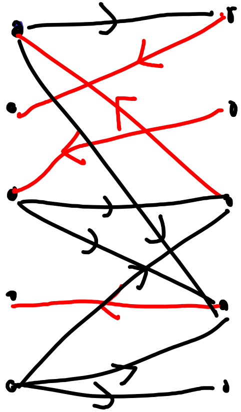
Removin

$M \oplus M' =$

$P - (k+1) \to (k+1)$

plus

cycle $C_1\, C_2 \ldots$



etc

So we should just minimise $\ell(P)$.

$\ell(C) < 0$

$\Rightarrow$

M not optimal.

$$\mathrm{cost}(M') - \mathrm{cost}(M)$$

$$= \ell(P) + \sum_i \underbrace{\ell(C_i)}_{\geq 0}$$

Need to minimise