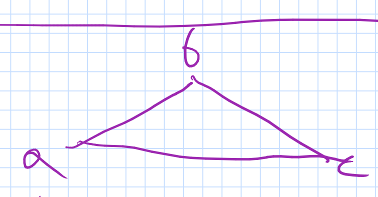
L18: Approximation algorithms 2 — 10/26/2021

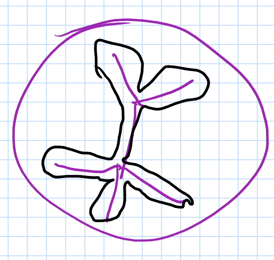
TSP with the triangle inequality



$u, v \in X$ $\quad d(u,v) \geq 0$

$d(a,c) \leq d(a,b) + d(b,c)$

MST



G TSP

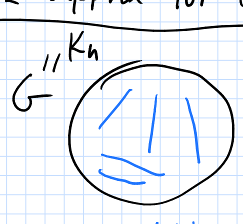
$w(MST) \leq w(OPT\_TSP)$

C: shortcut "doubled" MST walk

$w(\text{opt TSP}) \leq w(C) \leq 2w(MST) \leq 2w(\text{opt TSP})$

3/2-approx for TSP

G "K_n"



even number of vertices

weights on the edge

perfect matching = matching that engages all the vertices in the graph $n/2$

Q: Min cost perfect matching?

Thm min cost perfect matching can be computed in polynomial time. [Hungarian method]

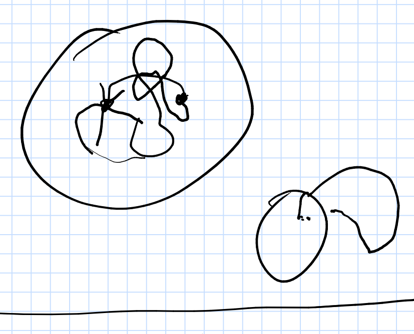
Bipartite

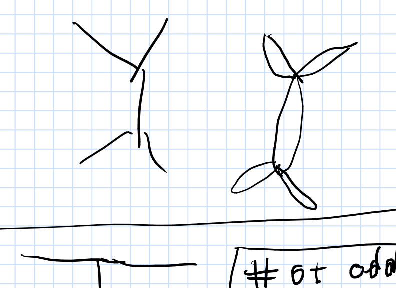
Eulerian cycle

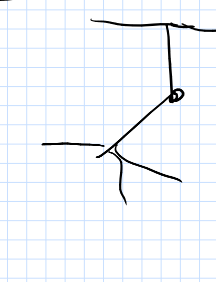
Cycle that visits every edge exactly once.

G can compute EC in linear time.

Thm A graph is Eulerian iff it is connected and all its vertices have even degree.





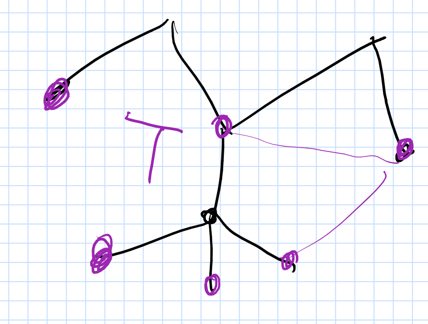


$\boxed{\text{\# of odd vertices in a graph is even.}}$

$$\sum_{v \in V} d(v) = 2|E|$$

$$\sum_{v \in V_{odd}} d(v) = 2m - \sum_{v \in V_{even}} d(v) = \text{even}$$

$\Rightarrow |V_{odd}|$ is even



$V(T)_{odd}$ = Build complete graph over them

$H = MST +$ min cost perfect matching (of odd vertices of MST)

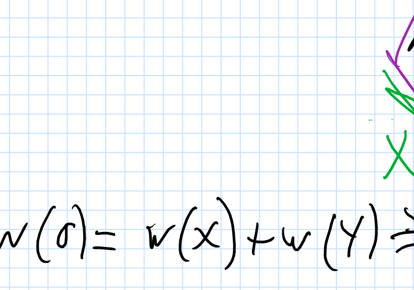
Compute EC for H

Shortcut it

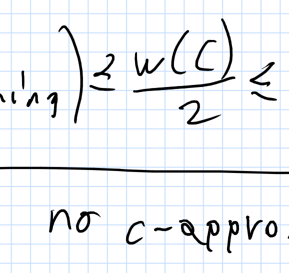
Output resulting cycle.

$\boxed{\text{Cost of the perfect matching edge added is at most } w(\text{opt TSP})/2.}$

Proof $\quad \pi = \text{opt TSP}$



$len(\sigma) \leq len(\pi)$



$w(\sigma) = w(X) + w(Y) \geq 2w(\text{min cost perfect matching})$

$w(\text{min cost perfect matching}) \leq \frac{w(C)}{2} \leq \frac{w(\text{opt TSP})}{2}$ ∎

$c > 1$ s.t. no c-approx is possible

1.499999-approx for TSP with triangle inequality.

Set cover

set system
hypergraph

(U, F) — U: ground set; F: family of subsets of U

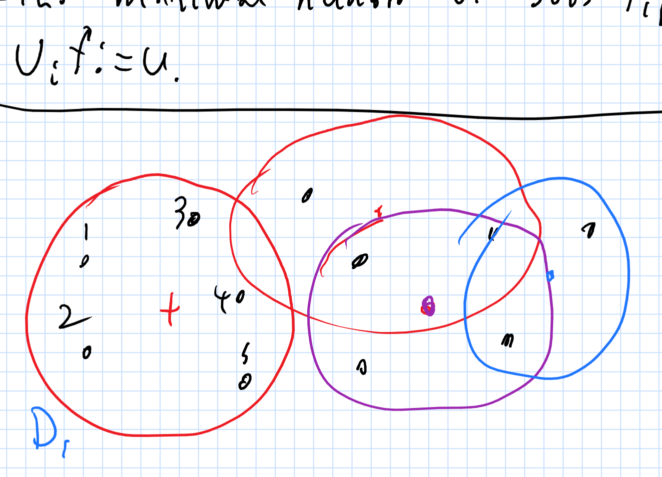
$F \subseteq 2^U$

$n = |U|$

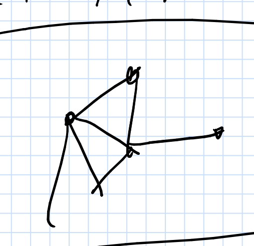
$m = |F|$

Find minimal number of sets $f_1, f_2, \ldots f_k \in F$

$\cup_i f_i = U$.



$f_1 = D_1 \cap P = \{1,2,3,4,5\}$



How to approximate

Greedy algorithm

$U_0 \leftarrow U$, $i = 0$ $\qquad O(n^2 m)$

while $U_i \neq \emptyset$ do

$\quad f_i \leftarrow$ set in F covering largest # of elements in U_i

$\quad U_{i+1} \leftarrow U_i \setminus f_i$

K: size of the optimal solution

$Opt = \{o_1, o_2, \ldots, o_k\} \subseteq F$ $\quad \bigcup_{o \in opt} o = U$

Theorem

The above algorithm computes a cover of size $O(k \log n)$.

Proof

$n_0 = |U_0| = n$, $\quad n_i = |U_i|$

U covered by k sets (opt solution)

$\cup|$

U_i can be covered by k set (opt solution)

n_i elements.

f_i: covers at least $\frac{n_i}{k}$ elements

$n_{i+1} \leftarrow n_i - |\text{\# of elements in } U_i \text{ covered by } f_i|$

$$n_{i+1} = n_i - \underbrace{|f_i \cap U_i|}_{\geq \frac{n_i}{k}} \leq n_i - \frac{n_i}{k} = \left(1 - \frac{1}{k}\right) n_i$$

$$n_{i+1} \leq \left(1 - \frac{1}{k}\right) n_i \leq \exp\left(-\frac{1}{k}\right) n_i \leq \exp\left(-\frac{2}{k}\right) n_{i-1}$$

$1 - x \leq e^{-x}$ $\qquad \leq \ldots \leq \exp\left(-\frac{i}{k}\right) n$

$i := \lceil k \ln n \rceil + 1$

$$n_i \leq \exp\left(-\frac{\lceil k \ln n \rceil + 1}{k}\right) n < \exp\left(-\frac{k \ln n}{k}\right) n = \frac{1}{n} \cdot n = 1$$

$\Rightarrow$ alg performs $\lceil k \ln n \rceil + 1$ iteration

$\Rightarrow$ outputs a cover of size $O(k \log n)$.

Max Exact 3SAT

$F = \underbrace{(l_1 \vee l_2 \vee l_3)}_{1 - \frac{1}{8} = \frac{7}{8}} \wedge (\ldots) \ldots$

$X_i = 1 \Leftrightarrow$ ith clause is sat by random assignment

$$E\left[\text{satisfied clauses}\right] = E\left[\sum_{i=1}^m X_i\right] = \sum_{i=1}^m E[X_i] = \sum_{i=1}^m \frac{7}{8} = \frac{7}{8} m$$