6. Approximation Algorithms

Introductory Example

Vertex cover: Find min-size $C \subseteq V$ st $C \cap e \neq \emptyset \;\; \forall e \in E$

```
GREEDY-VC1( G=(V,E) )
  Put C = ∅ , E' = E
  WHILE ( E' ≠ ∅ )
      v = vertex of max-degree in G' = (V,E')
      C = C ∪ {v}
      remove all edges incident to v from E'
  return C
```

```
GREEDY-VC2( G=(V,E) )
  Put C = ∅ , E' = E
  WHILE ( E' ≠ ∅ )
      choose arbitrary e = (u,w) ∈ E'
      C = C ∪ {u,w}
      remove all edges incident to u & w from E'
  return C
```

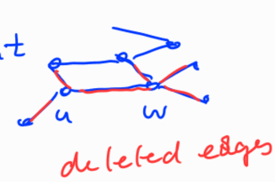
Which version is "better", i.e. closer to optimum?
Intuitively, "version 1", but this is not true!

GREEDY-VC2

RUNTIME: $O(|E|)$

& C is indeed a VC of G since we only remove edges from E' for which their "endvertices" are in C.

Now consider opt vertexcover C^* of G
& let $\tilde{E}$ be the set of edge we have have chosen in GREEDY-VC2
The edges in $\tilde{E}$ are pairwisely vertex disjoint



$|C^*| \geq |\tilde{E}|$ since C^* contains at least 1 vertex of those pairw.-vertex disj edges

By construnction: $C = 2|\tilde{E}| \leq 2|C^*|$

$\Rightarrow$ The vertex cover C returned by GREEDY-VC2 contains at most twice the vertices as an opt. vertexcover contains "Approximation guarantee"!

GREEDY-VC 1:

construct graph $G = (V, E)$ as follows

$V = L \dot\cup R$, $|L| = r$

$R = R_1 \dot\cup R_2 \dot\cup \dots \dot\cup R_r$, $|R_j| = \lfloor \frac{r}{j} \rfloor$

Every vertex in R_j has precisely j edges to vertices in L

& no two vertices in R_j have the same neighbor in L

& edges are "uniformly distributed":

$L = \{v_1 \dots v_r\}$

$R_1 = \{w_1 \dots w_r\} \to w_1 - v_1, \dots, w_r - v_r$

$R_2 = \{u_1 \dots u_{\lfloor \frac{r}{2} \rfloor}\} \to u_1 <^{v_1}_{v_2}, \; u_2 <^{v_3}_{v_4}, \; \dots$

$\vdots$

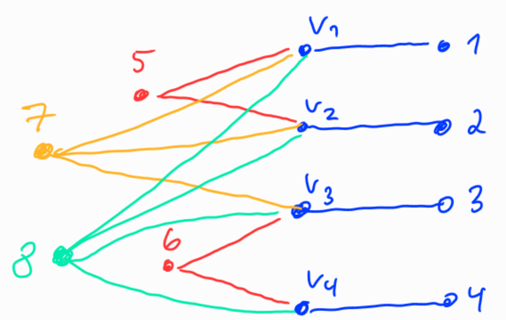
Exmpl: $r = 4$ $\quad L = \{v_1, v_2, v_3, v_4\}$

$|R_1| = \frac{4}{1} = 4$, $R_1 = \{1, 2, 3, 4\}$

$|R_2| = \frac{4}{2} = 2$ $\quad R_2 = \{5, 6\}$

$|R_3| = \lfloor \frac{4}{3} \rfloor = 1$ $\quad R_3 = \{7\}$

$|R_4| = \lfloor \frac{4}{4} \rfloor = 1$ $\quad R_4 = \{8\}$



$\Rightarrow \forall v \in L : \deg(v) \le r$

$\forall v \in R_i : \deg(v) = i$

possible greedy-choice: 8, 7, 6, 5, 4, 3, 2, 1

opt VC: v_1, v_2, v_3, v_4

$\overset{2}{\Rightarrow} |C| \le |C^*|$ NO!

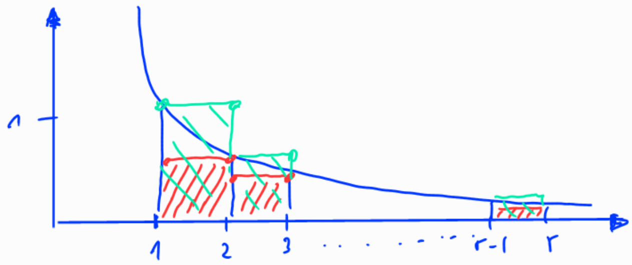
We show that there is no "approx. guarantee".

To this end, we show first:

$$r \cdot \ln(r) \le |V| \le r \cdot \ln(r) + 2r$$

By construction: $V = r + \sum_{j=1}^{r} \lfloor \frac{r}{j} \rfloor$ ⊗

Consider $f(x) = \frac{1}{x}$



Area: $\frac{1}{2}$ $\frac{1}{3}$ ... $\frac{1}{r}$

Area: 1 $\frac{1}{2}$ ... $\frac{1}{r-1}$

$$\Rightarrow \underbrace{\sum_{i=2}^{r} \frac{1}{i} \le \int_1^r \frac{1}{x}\,dx}_{\Rightarrow \sum_{i=1}^{r} \frac{1}{i} \le \ln(r)+1} = \underbrace{\ln(r) \le \sum_{i=1}^{r-1} \frac{1}{i}}_{\Rightarrow \ln(r) \le \sum_{i=1}^{r} \frac{1}{i}}$$

$$\Rightarrow \ln(r) \underset{⊗}{\le} \sum_{i=1}^{r} \frac{1}{i} \underset{⊗}{\le} \ln(r) + 1$$

Hence,

$$r \cdot \ln(r) \overset{⊗}{\le} r \cdot \sum_{i=1}^{r} \frac{1}{i} = r + \sum_{i=1}^{r} \frac{r}{i} - r \le r + \sum_{i=1}^{r} \lfloor \frac{r}{i} \rfloor = |V| \le r + \sum_{i=1}^{r} \frac{r}{i} =$$

$$= r\left(1 + \sum_{i=1}^{r} \frac{1}{i}\right) \underset{⊗}{\le} r\left(1 + \ln(r) + 1\right) = r \cdot \ln(r) + 2r$$

By construction of G: L opt VC.

R possible greedy VC

But: $\ln(r) - 1 = \frac{r\ln(r) - r}{r} \le \frac{|V| - r}{r} = \frac{|V| - |L|}{|L|} = \frac{|R|}{|L|} \le \frac{r\ln(r) + 2r - r}{r} = \ln(r) + 1$

$\Rightarrow \frac{R}{L} \sim \ln(r)$, i.e. $|R| \sim \ln(r)|L|$

$\Rightarrow$ GREEDY_VC1 may yield solutions that are worth than $\ln(r) \cdot$ opt sol. !

$\Rightarrow$ in general want to find poly-time alg that approximate a given solution in a "good" way.

Def: Π denotes opt. problem & $I \in \Pi$ an instance
A denotes algorithm for Π & $A(I)$ the value returned by applying A with input I
OPT(I) denotes optimal value for I

Alg. A has approximation ratio $\rho \in \mathbb{R}_{\geq 1}$
$\forall I \in \Pi$ it holds that:

$$\frac{1}{\rho} \, OPT(I) \leq A(I) \leq \rho \, OPT(I)$$

NOTE: for min-problems: left „≤" is always satisfied

$\Rightarrow$ it suffices to show that $\frac{A(I)}{OPT(I)} \leq \rho$

for max problems: right "≤" always satisfied

$\Rightarrow$ it suffices to show that $\frac{OPT(I)}{A(I)} \leq \rho$

A is optimal, if $\rho = 1$

An algorithm with approximation ratio ρ is called ρ- Approximation algorithm.

eg: GREEDY-VC2 is 2-approximat. alg.
GREEDY-VC1 does not have bounded approx. ratio, since $ln(r)$ not constant!

Traveling-Salesperson Proplem (TSP)

> **Input:** complete graph $G=(V,E)$ with edge weight $w: E \to \mathbb{N}_0$ & integer k
>
> **Q:** Is there a cycle C in G that visits each vertex precisely once s.t. $\sum_{e \in C} w(e) \leq k$?

Notation: for $e=(ab) \in E$ write $w(ab) := w(e)$
[instead of $w((ab))$]

Δ-TSP $\subseteq$ TSP where all weights satisfy $w: E \to \mathbb{N}_0$

triangle (Δ)-inequality: $\forall a,b,c: \; w(ab) \leq w(ac) + w(bc)$

Δ-TSP exmpl.: G drawn on plane $\mathbb{R}^2$
& $w(a,b)$ = euklidian distance.

Exercise(!) show Δ-TSP is still NP-hard.

[$M = \max_{e \in E} w(e) + 1$, put $w^*(e) = w(e) + M$

(1) $w^*(e)$ satisf Δ-ineq: $w^*(ab) = w(ab) + M \leq 2M \overset{w(e) \geq 0!}{\leq} w(ac) + M + w(bc) + M$
$= w^*(ac) + w^*(bc)$

(2) opt tour wrt w
$\widehat{=}$ opt. wrt "$w^* = w + M$".]

In what follows, we show that for TSP there is no poly-time approx. alg for TSP but for Δ-TSP there is one.

Notation: For $A \subseteq E$ write $w(A) := \sum_{e \in A} w(e)$ // $H \subseteq G$: $w(H) = \sum_{e \in E(H)} w(e)$

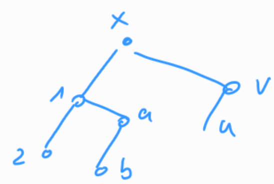
IDEA of Approx. alg for Δ-TSP:

1) find MST T^* → gives lower bound on optimal tour
2) Modify T^* to some tour & show that is tour has the desired approx. ratio.

To this end, we need:

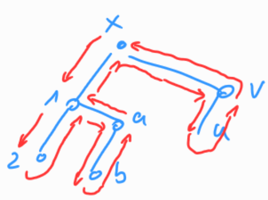
```
T = tree
x ∈ V(T)
v.visited = false ∀ v ∈ V(T).

PREORDER_WALK (x)
    IF (x.visited = false)
        PRINT x ; x.visited = true
        FOR (all (xy) ∈ E(T) with y.visited = false)
            PREORDER_WALK (y)
```



print: x, 1, 2, a, b, v, u

<u>FULL-WALK</u> of T lists the vertices of PREORDER-WALK when they are first visited and also whenever they are returned to after a visit to a subtree.



preorder: x, 1, 2, a, b, v, u

FULL WALK: x, 1, 2, 1, a, b, a, 1, x, v, u, v, x

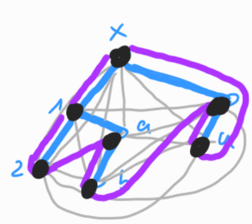
first occurrence gives preorder walk

APPROX_TSP (G, w)

T^* = MST of (G, w)

H = "preorder-walk + x" in T^*

return Hamiltonian cycle H



G = complete graph

MST T^* of G

H = preorder

Theorem 6.1: APPOX_TSP is a poly-time 2-Approx. alg for Δ-TSP.

Proof: runtime, clear (Exerc.)

Let H^* be optimal Hamilton. cycle
T^* be MST

$\Rightarrow w(T^*) \leq w(H^*)$ // since $H^* - e$ is spanning tree & T^* MST $\Rightarrow w(T^*) \leq w(H^*) - w(e)$ $\forall e \in E$

Let F be full walk in T^* (based on preorder walk H)

$\Rightarrow W(F) = 2\, w(T^*) \leq 2\, w(H^*)$ // each edge of T^* traversed twice in F

It holds for H = "preorder + x" that

$$w(H) \leq w(F)$$

SINCE: F

can successively by "shortcutting" double edges
eg: $(ca) - (ac) - (cb) - (bc)$
by replacing $(ac)-(cb)$
by (ab)

by Δ-ineq.: in $w(H)$ we have $w(ab) \leq$
in $w(F)$ we have $w(ac) + w(bc)$

$\Rightarrow \quad w(H) \leq w(F) \leq 2\, w(H^*)$

$\square$

Theorem 6.2 If $P \neq NP$, then there is no poly-time ρ-approx. alg. for TSP $\forall \rho \geq 1$.

proof: Assume, for contradiction, es exists poly-time ρ-approx. alg A for TSP & some $\rho \geq 1$.

Let I' be an instance of HAMILTON-CYCLE with input $G = (V, E)$

Construct instance I of TSP as follows:

put: $H = K_{|V|}$, $w: E(H) \to \mathbb{N}_0$

$$w(ab) := \begin{cases} 1, & (ab) \in E(G) \\ \rho \cdot |V|, & \text{else} \end{cases}$$

Claim: G has Hamiltonian cycle $\iff$ $A(I) \leq \rho \cdot |V|$

return value of A applied on I

„$\Rightarrow$" IF G has Ham. cycle C

$\Rightarrow$ this is an optimal tour in TSP of weight $w(C) = |V|$

Since A is ρ-approx. alg $\Rightarrow$ $A(I) \leq \rho \, OPT(I) = \rho \cdot |V|$

„$\Leftarrow$" by contraposition:

IF G has no Ham. cycle

$\Rightarrow$ Every tour in H has at least one edge e with $w(e) = \rho \cdot |V|$

$\Rightarrow$ $A(I) \geq \underbrace{|V| - 1}_{\text{tree}} + \underbrace{\rho \cdot |V|}_{+\text{ extra edge}} > \rho \cdot |V|$

in „opt" case only one edge of weight $\rho|V|$

$/\!\!\diamond$

$\Rightarrow$ Deciding whether G has a Ham. cycle or not can be done with poly-time Alg A.

$\Rightarrow$ $P = NP$ $\lightning$

$/\square$

Approximationschemes

Approx. alg provide results whose deviation from the optimal solution is "in some way fix". That is, one cannot obtain a better solution "efficiently"
But maybe we can achieve increasingly better solutions by using stepwisely more computation time!
That is, the approximation f should convert to 1, if runtime goes to "infinity."
This leads to the concept of "Approximationschemes"

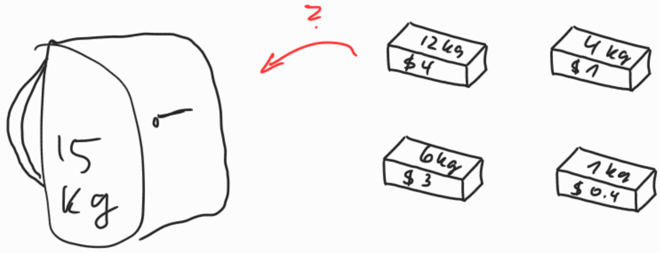
Def: Let π be an optimization problem.
A polynomialtime Approximationscheme (PTAS) for π
is an Algorithm A that takes as input
instance $I \in \pi$ & value ε $(0 < \varepsilon < 1)$
s.t. A is, for each fixed ε, a polynomial-time $(1+\varepsilon)$-approx. alg (in size of I)

eg: $O(n^{\frac{1}{\varepsilon}})$

PTAS is FPTAS (full polynomial-time Approx. scheme) if it is also polynomial in $\frac{1}{\varepsilon}$

eg: $O(n^{10}(\frac{1}{\varepsilon})^2)$

In what follows, we show that a version of the "knapsack" problem has a PTAS.



SIMPLE-KNAPSACK (SK)

INPUT: item-set $W = \{1 \dots n\}$
capacity C
weight of items $w_1 \dots w_n \in \mathbb{N}$, $w_i \le C$
integer B

Q: Is there a subset $M \subseteq W$ st $B \le \sum_{i \in M} w_i \le C$?

This problem is NP-complete.

Let us now consider the following greedy-alg.

GREEDY-SK($W = \{1..n\}$, $w_1 \dots w_n$, C) // $w_1 \ge w_2 \ge \dots \ge w_n$

```
M = ∅
cost = 0
FOR (i = 1 ... n) DO
    IF (cost + w_i ≤ C)
        M = M ∪ {i}
        cost = cost + w_i
return M
```

Proposition 6.3 GREEDY-SK is a polynomial 2-approx. alg. for SK.

proof: GREEDY-SK returns feasible solution, (since $\le C$) & runs in $O(n)$ time.

Let $M \subseteq W$ be a solution returned by GREEDY-SK,
Since $w_i \le C$ $\forall i \Rightarrow M \ne \emptyset$ & in particular $1 \in M$

Notation: $w(M) = \sum_{i \in M} w_i$

Let $j+1$ be the smallest index in W that is <u>not</u> in M, ie
ie. $1, \dots j \in M$, but $j+1 \notin M$.

<u>2 cases:</u> (i) $2 \notin M \Rightarrow j = 1 \Rightarrow w_1 + w_2 > C$

Since $w_1 \geq w_2 \Rightarrow 2w_1 \geq w_1 + w_2 > C$

$\Rightarrow w(M) \geq w_1 > \frac{C}{2}$

$$\Rightarrow \text{"}\frac{OPT(I)}{A(I)}\text{"} = \frac{\text{opt-cost}}{w(M)} \leq \frac{C}{\frac{C}{2}} = 2$$

(ii) $2 \in M \Rightarrow j \geq 2 \Rightarrow w(M) + w_{j+1} > C \geq \text{opt-cost}$ (I)

since $\sum_{i=1}^{j} w_j + w_{j+1} > C$
otherwise $j+1 \in M$!

Since $w_1 \geq w_2 \geq \dots \geq w_n$, it holds that

$$w_{j+1} \leq w_j = \frac{j \cdot w_j}{j} \leq \frac{w_1 + w_2 + \dots + w_j}{j} \leq \frac{C}{j} \quad \text{(II)}$$

since $\{1..j\} \subseteq M$
& $w(M) \leq C$

$$\Rightarrow w(M) \overset{(I)}{>} C - w_{j+1} \overset{(II)}{\geq} C - \frac{C}{j} \geq \frac{C}{2}$$

$j \geq 2$

$$\Rightarrow \text{"}\frac{OPT(I)}{A(I)}\text{"} = \frac{\text{opt-cost}}{w(M)} \leq \frac{C}{\frac{C}{2}} = 2$$

In summary, GREEDY-SK is a polytime 2-approx. alg.

$\square$

We now derive PTAS for GREEDY-SK.

SK-scheme $(W=\{1..n\}, w_1 .. w_n, C, \varepsilon)$ // $w_1 \geq w_2 \geq .. \geq w_n$

$k_\varepsilon = \lceil \frac{1}{\varepsilon} \rceil$

FOR (all subsets $M \subseteq \{1..n\}$ with $|M| \leq k_\varepsilon$ & $\sum_{i \in M} w_i \leq C$) DO

extend M via GREEDY-SK to M^*

[that is we greedily extend $M = \{i_1 .. i_j\}$ by successively checking if we can add $i_j+1, i_j+2 \ldots n$]

return one of the M^* for which $\sum_{i \in M^*} w_i$ is maximum.

Exmpl: $W = \{1,2,3\}$, $C = 5$, $\varepsilon = \frac{1}{2}$ ⟹ Aim: Find in polytime solution with approx. ratio $1+\varepsilon = 1{,}5$

weight value: 6 2 1

⟹ $k_\varepsilon = \lceil \frac{1}{\varepsilon} \rceil = 2$ ⟶ all subset $M \subseteq \{1,2,3\}$ with $|M| \leq k_\varepsilon = 2$

⟶ $M = \{1\}$ ⟶ weight $w(M) = 6 \not\leq C = 5$

$M = \{2\}$ ⟶ $w(M) = 2 \leq 5$
⟶ greedy ⟶ $M^* = \{2,3\}$

$M = \{3\}$ ⟶ $w(M) = 1 \leq 5$
⟶ greedy ⟶ $M^* = \{3\}$

⟶ $M = \{1,2\}$ ⟶ $w(M) = 8 \not\leq 5$

⟶ $M = \{1,3\}$ ⟶ $w(M) = 7 \not\leq 5$

⟶ $M = \{2,3\}$ ⟶ $w(M) = 3 \leq 5$
⟶ greedy ⟶ $M^* = \{2,3\}$

⟹ return $M^* = \{2,3\}$ with weight $w(M^*) = 3$

Theorem 6.4

SK-scheme is a PTAS for SK.

proof: RUNTIME FOR loop:

subsets of size $\leq k_\varepsilon$

$$= \sum_{0 \leq i \leq k_\varepsilon} \binom{n}{i} = \sum_{0 \leq i \leq k_\varepsilon} \frac{n!}{(n-i)!\,i!} \leq \sum_{0 \leq i \leq k_\varepsilon} (n-i+1)(n-i+2)\cdots n \leq \sum_{0 \leq i \leq k_\varepsilon} n^i = \frac{n^{k_\varepsilon+1}-1}{n-1} \in O(n^{k_\varepsilon})$$

in each step of the FOR-loop, we call GREEDY-SK

⟹ Total runtime: $O(n^{k_\varepsilon} \cdot n) = O(n^{\lceil \frac{1}{\varepsilon} \rceil + 1})$

Approx. ratio: let $w_1 \geq w_2 \geq \ldots \geq w_n$ for $W = \{1 \ldots n\}$

let $M_{opt} = \{i_1 \ldots i_p\}$, $i_1 < i_2 < \ldots < i_p$ be an opt. solution

2 cases: (i) $p \leq k_\varepsilon$ $\Rightarrow$ M_{opt} cannot be further extended with GREEDY-SK (otherwise M_{opt} would not be optimal)

$\Rightarrow M^* = M_{opt}$

(ii) $p > k_\varepsilon$: $M = \{i_1 \ldots i_{k_\varepsilon}\} \subseteq M_{opt}$ is one of the subsets that is considered in the FOR-loop.

$\Rightarrow$ 2 subcases $M^* = M_{opt}$ or $M^* \neq M_{opt}$.

Assume $M^* \neq M_{opt}$. $M_{opt} = \{i_1 \ldots i_{k_\varepsilon}, i_{k_\varepsilon+1} \ldots i_p\}$, $i_1 < \ldots < i_{k_\varepsilon} < \ldots < i_p$

$\Rightarrow \exists\, i_q \in M_{opt} \setminus M^*$ st $i_q > i_{k_\varepsilon} \geq k_\varepsilon$

$\Rightarrow w(M^*) + w_{i_q} > C \geq w(M_{opt})$ (x)

Since $w_{i_q} \notin M^* \Rightarrow$ there was a subset $M' \subseteq M^*$ st $w(M') + w_{i_q} > C$

the first k_ε - Elements of M_{opt} could be $\{1 \ldots k_\varepsilon\}$ in which case $i_{k_\varepsilon} = k_\varepsilon$ & otherwise $i_{k_\varepsilon} > k_\varepsilon$

Since $i_1 \ldots i_{k_\varepsilon}$ & $i_q \in M_{opt}$ it holds that

$$w_{i_q} = \frac{(k_\varepsilon+1)\, w_{i_q}}{(k_\varepsilon+1)} \leq \frac{w_{i_1} + w_{i_2} + \ldots + w_{i_{k_\varepsilon}} + w_{i_q}}{k_\varepsilon+1} \leq \frac{w(M_{opt})}{k_\varepsilon+1} \quad (y)$$

$$\Rightarrow \text{"}\frac{OPT(I)}{A(I)}\text{"} = \frac{w(M_{opt})}{w(M^*)} \overset{(x)}{<} \frac{w(M_{opt})}{w(M_{opt}) - w_{i_q}} \overset{(y)}{<} \frac{w(M_{opt})}{w(M_{opt}) - \frac{w(M_{opt})}{k_\varepsilon+1}}$$

$$= \frac{1}{1 - \frac{1}{k_\varepsilon+1}} = 1 + \frac{1}{k_\varepsilon} \leq 1 + \varepsilon$$

$k_\varepsilon = \lceil \frac{1}{\varepsilon} \rceil$

$\square$

There are also FPTAS for SK with runtime $O(n \log \frac{1}{\varepsilon} + \frac{1}{\varepsilon^4})$