§8 Approximation Algorithms

- metric Travelling Salesperson
  - *Christofides*: approximation ratio 2
- Knapsack
  - Strongly polyn.-time / Dynamic Programming
  - *Fully Polynomial-Time Approximation Scheme*
- Limits of Approximability

Approximating metric TSP

MTSP = { $\langle G,\underline{w},k\rangle$ | G with metric edge weights $\underline{w}:V\times V\to\mathbb{N}$ admits a Hamiltonian circuit of weighted length $\leq k$ }

Input: $\underline{w}:V\times V\to\mathbb{N}$ edge weights symmetric and s.t. triangle inequality holds: $w(a,c)\leq w(a,b)+w(b,c)$

Sought: Tour (permutation π of V) of least weight

Decision problem **MTSP** still $\mathcal{NP}$-complete

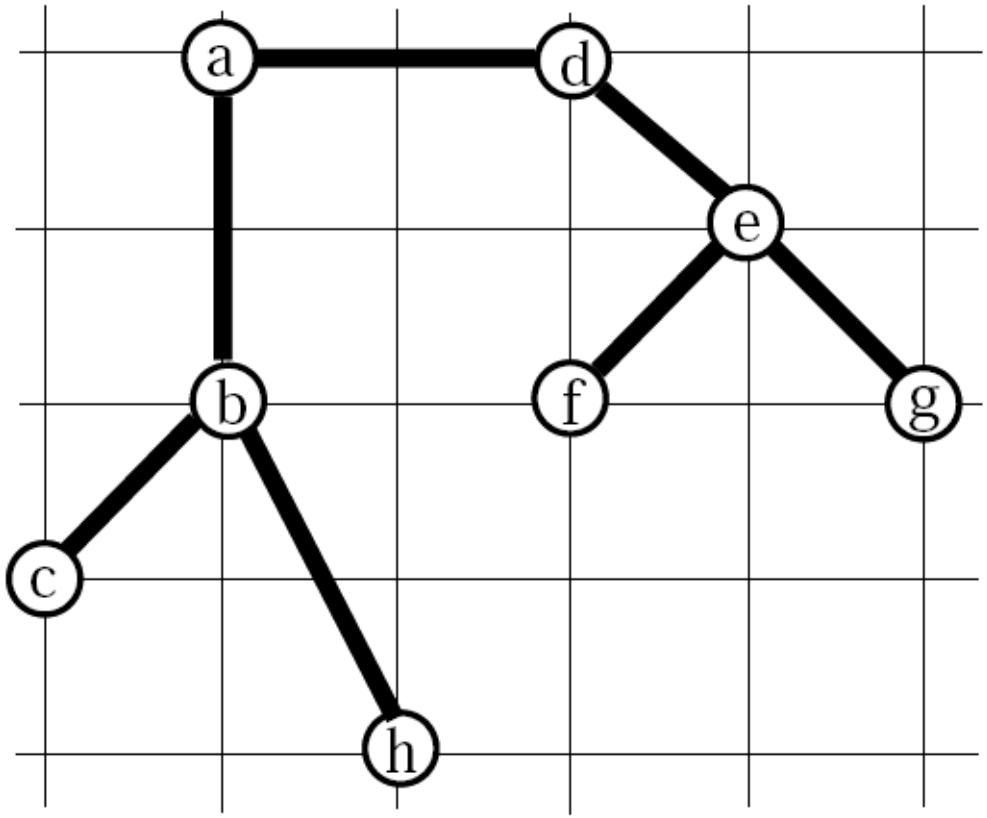
[Christofides'76]

Polytime approximating **minMTSP** up to factor 2:

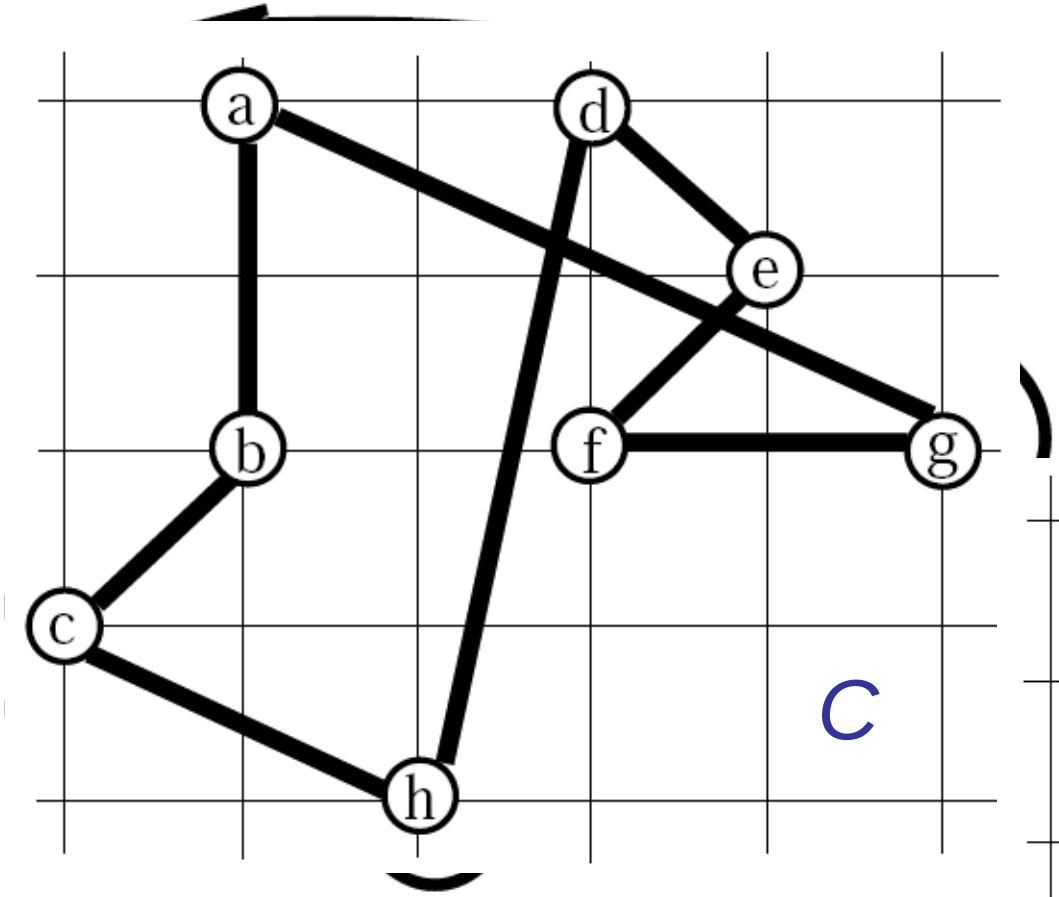
1. Compute minimum spanning tree T of (G,w).
2. Traverse T depth-first pre-order

ETSP with $V\subseteq\mathbb{R}^d$, $w(\underline{a},\underline{b})=\|\underline{a}-\underline{b}\|_2$ $\mathcal{NP}$-hard, but **in** $\mathcal{NP}$ **?**

Example: 1. Compute MST *T*



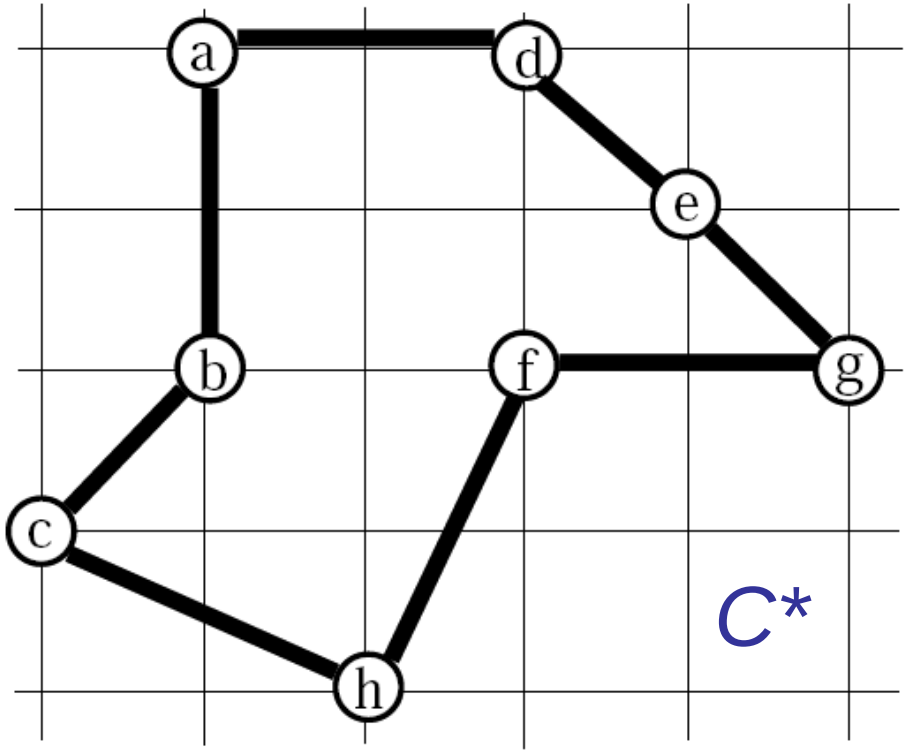
Example: 2. Traverse *T* in pre-order



C

Tour: a,b,c,h,d,e,f,g,a

optimal: a,b,c,h,f,g,e,d,a



*C**

Proof of Approximation Ratio

w weights with 3-inequality, T is MST traversed in-order

Let F denote the sequence of edges pursued in-order, C the tour thus obtained, C^* (unknown) optimal tour.

For edges $e_1,\ldots e_k$ abbreviate $L(e_1,\ldots e_k) := w(e_1)+\ldots+w(e_k)$

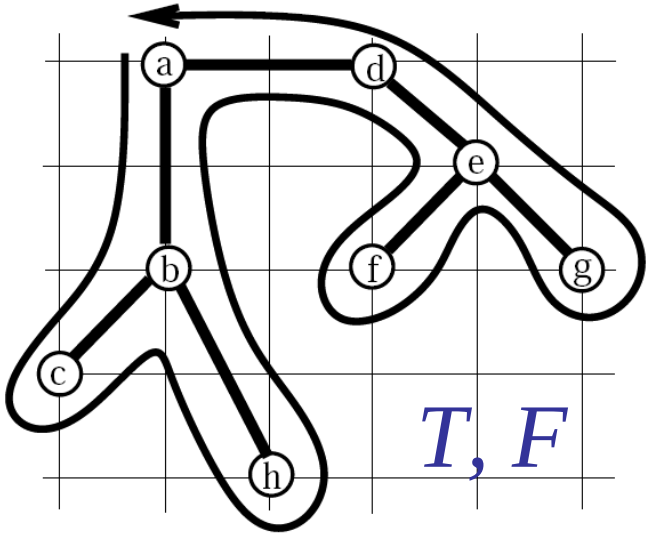
(i) $L(T) \le L(C^*)$, because removing any edge from C^* yields a spanning tree of cost $\le L(C^*)$

Every edge of T appears precisely twice in F:

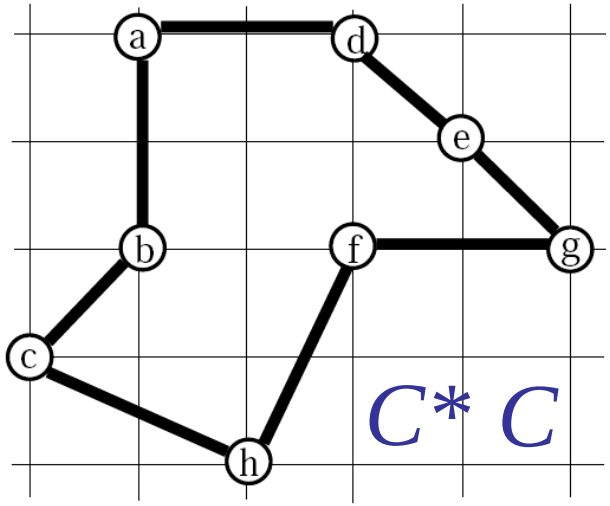
(ii) $L(F) = 2 \cdot L(T)$

(iii) $L(C) \le L(F)$ because 3-inequal.

$\Rightarrow L(C) \le L(F) = 2 \cdot L(T) \le 2 \cdot L(C^*)$



T, F



C^* C

Approximation Schemes

Input: n packets, values $v_1,\dots v_n \in \mathbb{N}$ and weights $w_1,\dots w_n \in \mathbb{N}$ and c) value bound V.

Question: Is there a subset $S \subseteq \{1,\dots n\}$ s.t. values $\sum_{p\in S} v_p \geq V$ subject to weight bound $\sum_{p\in S} w_p \leq W$

Now: Find S' s.t. $\sum_{p\in S'} w_p \leq W$ and $\sum_{p\in S'} v_p \geq V\cdot(1-\varepsilon)$

Or: Find S'' s.t. $\sum_{p\in S''} w_p \leq W\cdot(1+\varepsilon)$ and $\sum_{p\in S''} v_p \geq V$

Algorithm: guaranteed approximation ratio $1\pm\varepsilon$

Discrete optimization $\rightarrow$ decision often $\mathcal{NP}$-hard

Try approximating maxim./minim. up to relative error

Dynamic Programming: *Knapsack*

For $S \subseteq \{1,\dots n\}$ write $w(S)=\sum_{p\in S} w_p$ and $v(S)=\sum_{p\in S} v_p$

Goal: Given W, determine $V := \max \{ v(S): w(S)\leq W \}$

Consider $T(v,m) := \min \{ w(S) : S\subseteq\{1,\dots m\}, v(S)\geq v \}$

Note: i) $T(0,n) \leq T(1,n) \leq \dots \leq T(V,n) \leq W < T(V+1,n)$

ii) $V = \max \{ v : T(v,n) \leq W \}$

iii) $T(v,m) = 0$ for $v\leq 0$

iv) $T(v,0) = \infty$ for $v>0$

v) $T(v,m) =$
$\min \{ T(v,m\text{-}1) ,$
$w_m + T(v\text{-}v_m , m\text{-}1) \}$

w.l.o.g.
$0<w_p\leq W$
$0<v_p\leq V$

v \ m	0	1	…	n
0	0	0	0	0
1	∞			
2	∞			
⋮	∞			T

runtime poly(n+V)

FPTAS for *Knapsack*

Scaling Lemma a) For $0 \le \underline{v}' \le \underline{v}$, $V(\underline{v}') \le V(\underline{v})$

b) and for $\underline{v} \le \underline{d} \le (k,\ldots k)$: $V(\underline{v}-\underline{d}) \ge V(\underline{v}) - n \cdot k$

c) Also, $V(k \cdot \underline{v}) = k \cdot V(\underline{v})$

$v - k < \lfloor v/k \rfloor \cdot k \le v$

Scaling Method: Fix $k \in \mathbb{N}$ and let $v_p' := \lfloor v_p / k \rfloor$

Compute $V' := k \cdot V(v_1', \ldots, v_n')$ in time $\mathrm{poly}(n + V/k)$. So

$V' = V(\lfloor \underline{v}/k \rfloor \cdot k) \ge V(\underline{v} - k \cdot \underline{1}) \ge V - n \cdot k = V \cdot (1 - n \cdot k / V) \overset{!}{\ge} V \cdot (1-\varepsilon)$

for $k := \approx \lfloor \varepsilon \cdot \sum_p v_p / n^2 \rfloor \le \varepsilon \cdot V/n$ $\quad 0 < v_p \le V \Rightarrow V \le \sum_p v_p \le nV$ $\quad V/k \le O(n^2/\varepsilon + 1)$

Theorem: For every given $\varepsilon > 0$, can approximate Knapsack up to error $1-\varepsilon$ in time polynom. in $n + 1/\varepsilon$

$$V(v_1, \ldots v_n) := \max\{ \textstyle\sum_{p \in S} v_p : S \subseteq \{1..n\}, \sum_{p \in S} w_p \le W \}$$

Limits of Approximation

Theorem: No polynom.-time algorithm can approximate the general TSP up to some constant unless $\mathcal{P}=\mathcal{NP}$.

Proof: Suppose $\mathcal{A}$ approximates TSP up to factor $c\in\mathbb{N}$.

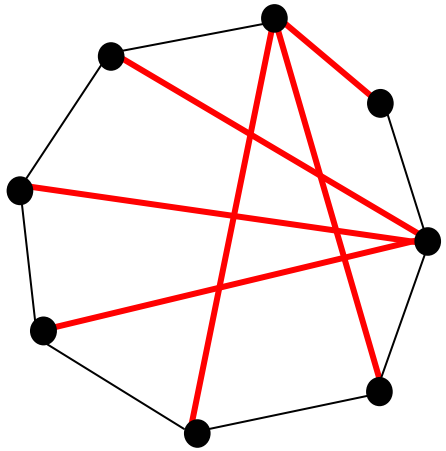
Turn $\mathcal{A}$ into algorithm $\mathcal{B}$ for HC:

Algorithm $\mathcal{B}$, input graph $G=(V,E)$, $n:=|V|$.

Define $w(u,v) := 1$ for $\{u,v\}\in E$;

$w(u,v) := n\cdot c$ for $\{u,v\}\notin E$.

No triangle-inequality…



$\langle G\rangle\in\mathbf{HC} \Rightarrow w$ contains Hamiltonian cycle of weight n

$\Rightarrow$ algorithm $\mathcal{A}$ finds some of weight $\leq n\cdot c$

$\langle G\rangle\notin\mathbf{HC} \Rightarrow$ any Hamiltonian cycle has weight $> n\cdot c$

HC := { $\langle G\rangle$ | graph G contains a Hamiltonian cycle }

TSP := { $\langle G,w,k\rangle$ | (G,w) contains a Hamiltonian cycle of weight $\leq k$ }

§8 Summary

- metric Travelling Salesperson
  - *Christofides*: approximation ratio 2
- Knapsack
  - Strongly polyn.-time / Dynamic Programming
  - *Fully Polynomial-Time Approximation Scheme*
- Limits of Approximability